

# **Methods for Processing and Analyzing Protein Structure Collections for Data-Driven Structure-Property Relationship Modeling**

Cumulative Dissertation

to receive the degree

*Dr. rer. nat.*

at the Faculty of Mathematics, Informatics and Natural Sciences

Universität Hamburg

submitted to the

Department of Informatics of

Universität Hamburg

**Jochen Sieg**

born in Giessen

Hamburg, December 2023





1. Reviewer: Prof. Dr. Matthias Rarey  
2. Reviewer: Prof. Dr. Andrew Torda  
3. Reviewer: Prof. Dr. Gerhard Wolber  
Date of thesis defense: 22.03.2024



# Kurzfassung

Die effektive Vorhersage der Eigenschaften von Biomolekülen könnte entscheidende Forschungsfragen beantworten: Welches Biomolekül wäre ein wirksames Arzneimittel für eine bestimmte Krankheit? Wird eine Mutation bei einem Patienten pathologisch sein? Welches Biomolekül kann Materialien wie Kunststoffe abbauen?

Das Paradigma der Struktur-Eigenschafts-Beziehung ist ein zentrales Konzept, welches beschreibt, dass die Struktur eines Biomoleküls seine Eigenschaften bestimmt. Insbesondere für Proteine, die sogenannten Bausteine des Lebens, hat die Zahl der hochwertigen dreidimensionalen Strukturdaten in den letzten Jahren enorm zugenommen. Datengetriebene Vorhersagemethoden, wie maschinelles Lernen, sind eine viel versprechende Wahl, um Eigenschaften mittels der Strukturdaten vorherzusagen. Solche datengetriebenen Methoden unterliegen jedoch Datenlimitierungen und benötigen Proteinrepräsentationen, die der Natur und den Eigenschaften der Proteine angemessen sind. In dieser Arbeit wurden Methoden zur Analyse und Verarbeitung von Datensätzen entwickelt, um datengetriebene Eigenschaftsvorhersagen zu verbessern.

Zunächst wurde eine auf maschinellem Lernen basierende Interpretierbarkeitsmethode entwickelt, um prädiktive Feature in einem Datensatz für bestimmte Eigenschaftsvorhersagen zu analysieren. Die Technik wurde zuerst zur Analyse von Unbiasing-Strategien in Benchmark-Datensätzen für strukturbasiertes virtuelles Screening bei der Arzneimittelentwicklung eingesetzt. Daraufhin wurde sie mit dem Shapley Value System erweitert und verwendet, um stabilisierende Proteinanpassungen für das Protein-Engineering zu interpretieren. Neben wichtigen domänenspezifischen Trends haben die Analysen gezeigt, dass Datenlimitierungen ein tiefgreifender Engpass in der Modellierung von Struktur-Eigenschafts-Beziehungen sind. Mehr Daten zu akquirieren ist oft nicht möglich. Eine effektive Alternative kann die Prozessierung von existierenden Daten sein, um bessere Proteinrepräsentationen für die jeweilige Aufgabe zu erhalten. Es wurden zwei Prozessierungsmethoden entwickelt, welche relevante Proteinvariabilitäten mittels Strukturensambles beschreiben. Die erste Methode enumeriert alternative Konformationen anhand von AltLoc-Annotationen, um die Proteinflexibilität zu repräsentieren. Die

zweite Methode konstruiert Strukturensamples mittels der Ähnlichkeit von 3D Mikro-Umgebungen von Aminosäureresten, um die strukturellen Änderungen durch Einzelmutationen zu repräsentieren. Beide Methoden können auf gesamte Proteinstruktursammlungen angewendet werden und essentielle Daten und verbesserte Repräsentationen von Proteinen für eine Vielzahl von Eigenschaftsvorhersagen, Methodenentwicklung und molekulares Modeling bereitstellen.

# Abstract

Effective prediction of the properties of biomolecules could answer crucial research questions: Which biomolecule would be an effective drug for a particular disease? Will a mutation in a patient be pathologic? Which biomolecule can break down materials like plastics?

The structure-property relationship paradigm is a central concept describing that the biomolecule's structure determines its properties. Especially for proteins, the so-called building blocks of life, high-quality three-dimensional structure data has increased tremendously in the last years. Data-driven prediction methods, like machine learning, are a promising choice to predict properties from the structure data. However, such data-driven methods are subject to data limitations and need protein representations adequate for proteins' nature and properties. In this work, methods were developed to analyze and process data sets for improving data-driven property prediction.

First, a machine learning-based interpretability method was developed to analyze predictive features on a data set for a given property-prediction task. The technique was first applied to analyze unbiasing strategies in benchmark data sets for structure-based virtual screening in drug discovery. Then, it was extended with the Shapley Values framework and used to interpret stabilizing protein adaptations for protein engineering. Besides important domain-specific trends, the analyses demonstrated that data limitations are a profound bottleneck in structure-property modeling. Obtaining more data is often not possible. An effective alternative can be to process the existing data to derive better protein representations for the task at hand. Two processing methods that describe relevant protein variabilities using structure ensembles were developed. The first method enumerates alternative conformations from AltLoc annotations to represent proteins' inherent flexibility. The second method constructs structure ensembles through the similarity of residue 3D micro-environments to represent the structural changes upon single mutations. Both methods can be applied to entire protein structure collections and provide essential data and an improved representation of proteins for various property-prediction tasks, method development, and molecular modeling.



# Acknowledgments

I want to acknowledge and thank the people who were instrumental in supporting and inspiring me throughout my doctorate.

First, I want to thank Matthias Rarey for allowing me to pursue my doctorate in his group and for providing all the excellent support, guidance and feedback over the last few years. I also want to thank Florian Flachsenberg for continuously giving me great advice, particularly during the beginning of my doctorate. I also thank Joel Graef for providing me with the initial version of the Latex template for preparing this thesis. Further, I thank the BMBF for funding my work.

I am thankful for many fruitful discussions which, in one way or another, contributed to or supported my projects. Many conversations with Patrick Penner, Florian Flachsenberg, Louis Bellmann, Joel Graef, Christiane Ehrt, Wolf-Guido Bolick, Torben Guter-muth, Jonathan "Eddi" Pletzer-Zelgert and Annika Jochheim helped me to shape my research. I also have to give my gratitude to Rainer Fährrolfes, Kai Sommer, Emanuel Ehmki, Konrad Diedrich, Christian Meyenburg, Uschi Dolfus, and all the other mem-bers of the Center of Bioinformatics for being excellent colleagues. It was a pleasure to work with you. Finally, beyond the workplace, I owe my gratitude to my family and friends, whose incredible and persistent support was pivotal in shaping the course of my journey.





# Contents

<b>1</b>	<b>Introduction</b>	1
1.1	Proteins and Structure Data Collections . . . . .	2
1.2	Protein Properties . . . . .	4
1.3	Structure-Property Relationship Modeling . . . . .	5
1.3.1	Similarity Search Methods . . . . .	6
1.3.2	Machine Learning Methods . . . . .	6
1.4	Challenges in Data-Driven Modeling . . . . .	9
1.4.1	Data Limitations . . . . .	9
1.4.2	Protein Representations . . . . .	11
1.4.3	Interpretability . . . . .	12
1.5	This Work . . . . .	13
1.6	Overview of Scientific Contributions . . . . .	15
<b>2</b>	<b>Publications</b>	17
2.1	Analyzing Benchmark Data Sets for Virtual Screening . . . . .	17
2.1.1	Motivation . . . . .	17
2.1.2	Preliminary Work . . . . .	18
2.1.3	Unbiasing Strategies in Benchmark Data Sets . . . . .	19
2.1.4	Methodological Summary . . . . .	21
2.1.5	Evaluation of Unbiasing Strategies . . . . .	22
2.1.6	Outlook . . . . .	25
2.2	Analyzing Structure Data Sets for Protein Adaptations . . . . .	27
2.2.1	Motivation . . . . .	27
2.2.2	Data Set Creation . . . . .	28
2.2.3	Analytical Workflow . . . . .	29
2.2.4	Methodological Summary . . . . .	31
2.2.5	Analyzing Important Features For Protein Adaptations . . . . .	32

2.2.6 Outlook . . . . .	34
2.3 Alternate Location Enumeration . . . . .	35
2.3.1 Motivation . . . . .	35
2.3.2 AltLocs in the PDB . . . . .	36
2.3.3 Handling AltLocs with AltLocEnumerator . . . . .	37
2.3.4 Outlook . . . . .	43
2.4 3D Micro-Environment Similarity Search . . . . .	44
2.4.1 Motivation and Idea . . . . .	44
2.4.2 Approaches for Searching Locally Similar Protein Structures . . . . .	45
2.4.3 Searching Single Mutation Structure Pairs with MicroMiner . . . . .	47
2.4.4 Outlook . . . . .	55
<b>3 Summary</b>	<b>59</b>
<b>References</b>	<b>61</b>
<b>References of the Cumulative Dissertation</b>	<b>81</b>
<b>Appendix</b>	<b>83</b>
<b>A Scientific Contributions</b>	<b>83</b>
A.1 Publications . . . . .	83
A.2 Conference Contributions . . . . .	85
A.2.1 Talks . . . . .	85
A.2.2 Poster . . . . .	86
<b>B Software Architecture</b>	<b>87</b>
B.1 NAOMI . . . . .	87
B.2 AltLocEnumerator . . . . .	87
B.3 MicroMiner . . . . .	88
B.4 Feature Interpretability Method . . . . .	89
<b>C Software Usage</b>	<b>93</b>
C.1 AltLocEnumerator . . . . .	93
C.1.1 Command Line Tool . . . . .	93
C.2 MicroMiner . . . . .	96
C.2.1 Command Line Tool . . . . .	96
C.2.2 Evaluation and Application Scripts . . . . .	101
C.2.3 Web server . . . . .	102

<b>D Journal Articles</b>	103
D.1 In Need of Bias Control: Evaluating Chemical Data for Machine Learning in Structure-Based Virtual Screening . . . . .	103
D.2 Analyzing structural features of proteins from deep-sea organisms . . . . .	135
D.3 Modeling with Alternate Locations in X-ray Protein Structures . . . . .	179
D.4 Searching similar local 3D micro-environments in protein structure databases with MicroMiner . . . . .	201
D.5 ProteinsPlus: a comprehensive collection of web-based molecular modeling tools . . . . .	221



# Chapter 1

## Introduction

A fundamental approach to viewing the physical world in the life sciences is through data collected in experiments. This data can be, for example, the three-dimensional structure of a protein solved with X-ray crystallography [1], measures of mutation effects on protein function with deep mutational scanning [2] or assessments of the interaction of proteins and small molecules using diverse ligand binding assays [3, 4]. Usually, experimental methods generate the most reliable and highest quality data considered the gold standard for computational methods [5–7]. However, experimental data generation is often expensive in time and resources [8, 9] and has limitations in applicability [10, 11]. Computational methods can achieve faster, cheaper, and novel results through predictions or make experimental processes more efficient [8, 9, 12–15].

The computational modeling of the complex relationships between protein structure and relevant properties, like inhibition upon binding or mutation effects, is central to numerous disciplines, including drug design and discovery [8, 9] and protein engineering and design [12, 16, 17]. A major concept is that the protein’s structure is a determinant of numerous relevant protein properties, like the function [15, 18, 19] or binding partners [20, 21]. This concept will be termed the structure-property relationship in this thesis. Various modeling approaches have been developed over the years to map structures to properties and predict the unknown properties of new protein structures [20–25]. Data-driven methods, like machine learning, are promising techniques that can learn to associate protein structure and properties automatically from patterns in data sets [26, 27]. In the last years, many successful data-driven methods have been proposed, applying innovative concepts and borrowing ideas from other domains to structural bioinformatics and adjacent life science fields; examples are protein structure prediction [28, 29], protein-ligand binding scoring [30], and methods predicting various elemental and practically relevant properties of biomolecules [31, 32].

Today, high-quality protein structure data is abundant [33–35]. Simultaneously, the progress in modern machine learning methods shows excellent potential in various domains [28, 36–38]. Consequently, high expectations have been raised to improve property prediction of biomolecules [15, 26, 27, 39]. However, several domain-specific challenges arising from the available data must be considered to deploy such data-driven methods effectively to drug discovery and protein engineering tasks.

In this dissertation, three software methods were developed, and two in-depth analyses were conducted to address central challenges in data-driven structure-property relationship modeling. The first software method is a machine learning-based interpretability method developed for and applied to two specific analysis studies. The analysis results reveal detailed trends in data sets for structure-based virtual screening and proteins' high-pressure adaptations. They also stress profound limitations that need to be overcome to progress. In contrast, the other two developed tools more universally address data limitations and shortcomings in representations of proteins during modeling. They use structure ensembles to represent protein flexibility and structural changes upon single mutations to improve various property-prediction tasks, method development, and molecular modeling.

The following sections of this chapter introduce the relevant concepts for data-driven structure-property relationship modeling. First, the current state of protein structure data and relevant protein properties are described. Then, an overview of data-driven methods for structure-property modeling is provided, and their limitations regarding the underlying data sets, protein representations, and interpretability are described. Finally, the structure of the rest of this work is presented.

### 1.1 Proteins and Structure Data Collections

Proteins are essential building blocks of life [40]. With their diverse functions, they play an irreplaceable and ubiquitous role in the processes of organisms [40]. The functional diversity of proteins includes, for example, the catalysis of chemical reactions, transport processes, signal transduction, motion, and immunological defense and protection against foreign substances [40].

The study of proteins is of fundamental and substantial practical importance in many scientific research and industrial areas. Due to their ubiquitous involvement in physiological and pathological processes, the intended alteration of protein function is a goal in drug design [41]. A classical pharmaceutical concept is the targeted inhibition of pathologically relevant proteins with small molecules to achieve a therapeutic effect [41]. Proteins are also used as therapeutic agents, as so-called biopharmaceuticals

or biologics [42, 43] with antibodies being a notable example [44]. Proteins are also essential in biotechnology due to their specific and diverse functions. In particular, enzymes - proteins that catalyze a chemical reaction - are often used as biocatalysts in biotechnological processes [45]. The properties of proteins can be tailored for industrial applications [45, 46]. For this purpose, proteins can be optimized using protein engineering methods to achieve, for example, higher enzymatic activity, a broader substrate spectrum, and higher protein stability [46].

Proteins are flexible, three-dimensional biomolecules [47]. Their physical structure comprises interacting amino acids that create specific structural and chemical environments crucial for the protein’s properties like the function [48–51]. While many proteins fold into a particular structure, in their native biological environments, proteins exist as an ensemble of energetically accessible conformations [47]. Protein structure data representing these states is essential for understanding protein properties. Since the first solved protein structure of Myoglobin in 1958 [52], structure determination techniques have constantly improved. The most common methods are X-ray crystallography, nuclear magnetic resonance spectroscopy (NMR) and cryogenic electron microscopy (Cryo-EM) [33]. Cryo-EM methods have been especially improved in the last years and enable the structure determination of large multimeric protein complexes in atomic resolution [53–55]. More than 200,000 structures of proteins could be determined over the years, which are curated and made openly accessible by the Protein Structure Database (PDB) [33]. As a result of these efforts, the research community now has access to a large and growing set of high-resolution experimental protein structure data.

Besides experimental structure determination, enormous improvements have been achieved in the *in silico* protein structure prediction based on the protein sequence in the last few years [28, 29]. This progress is attributed [28] to the extensive data collection and curation of the PDB and the methodological advancements in deep learning [36] in the areas of computer vision [56, 57] and computational linguistics [37]. In addition, the community-wide blind prediction challenge in CASP [58] provided a well-defined and accepted prospective evaluation for structure prediction. Today, in addition to the approximately 200,000 experimental protein structures in the PDB, more than 214 million predicted structures are available in the AlphaFold Protein Structure Database [34] and more than 617 million in the ESM Metagenomic Atlas [35]. Overall, this makes high-quality structures for almost all folded proteins publicly available [59]. The unprecedented variety of protein structures available has the substantial potential to improve multiple areas in structural bioinformatics, especially structure-property relationship modeling for property prediction.

## 1.2 Protein Properties

The term 'protein properties' is used differently throughout the life science disciplines. Therefore, the following describes how the term is used in this thesis.

This thesis focuses on computational methods for drug discovery and protein engineering. Protein properties in this context are considered various attributes and characteristics of a particular protein, including structural, functional, chemical, biological, pharmacological, and dynamic properties. Generally, all properties whose prediction helps address relevant life science problems are of interest. Popular examples are the protein function [60], potential binding sites [61], binding partners [62, 63], binding affinity of biomolecules [64], flexibility [65, 66], hydrophobicity [67], and stability [68, 69]. Understanding these protein properties is invaluable for growing biological and chemical knowledge and facilitating biotechnological and pharmaceutical advancement.

For example, predicting proteins' function can help elucidate life from the molecular level. A protein's molecular function, biological role, and location of activity in the organism build a basis to characterize biological systems and whole organisms [70]. The diversity of functional properties resulted in specific prediction tasks, like the prediction of subcellular locations [71] or predicting the specific chemical reactions enzymes catalyze [25].

Predicting the binding of proteins and other molecules is integral to industrial drug discovery and academic research [8]. In structure-based drug discovery, new drugs for a therapeutic protein target are searched by first identifying a suitable binding site on the target protein structure and then searching for small molecules binding the site with a desired affinity, bioactivity, and molecular properties using techniques like virtual screening [61, 72]. Protein property prediction methods are employed at every step of this process and various other stages in the drug discovery pipeline [8, 61, 72].

It is highly relevant for various applications to determine how alterations like protein mutations impact properties [46]. Protein engineering aims to find protein variants with improved properties [46]. Prediction methods are frequently applied to guide experimental redesign through the vast protein sequence space to optimize properties, like enzymes' substrate conversion, enantioselectivity, or light absorption [73]. Similar approaches guide the optimization and design of therapeutic antibodies for specifically binding a target antigen's epitope. Antibodies are the largest class of biologics and comprise a considerable part of blockbuster drugs [74]. Furthermore, mutations have significant pharmaceutical relevance because they can cause drug resistance or lead to diseases like cancer [75, 76]. Thus, predicting mutation effects on binding and stability



can help to unravel the molecular roots of drug resistance and diseases and build a basis for personalized medicine [69, 77, 78].

### 1.3 Structure-Property Relationship Modeling

As described above, the foundation for many protein property prediction methods is the structure-property relationship paradigm, which describes that the protein’s three-dimensional structure determines the protein’s properties. The application of this paradigm can be challenging because structure-property relationships are often complex, and their exact mechanisms still need to be understood [51, 68, 79–82]. It is not uncommon that some relevant structural factors are known, for example, noncovalent interactions. However, this knowledge is often incomplete, and the interplay of multiple factors is intricate [68, 82, 83]. Even predicting the impact on properties due to minor structural changes can be challenging [51, 81, 84]. Consequently, an explicit specification of the relationships’ rules is usually intractable. For this reason, data-driven methods are appealing because they do not require explicit knowledge of the relationship. Such modeling techniques range from similarity-based approaches, which transfer annotations of well-characterized protein data to unknown data points using similarity, to predictive modeling approaches using statistical modeling and machine learning. Both these approaches are described in detail in the following subsections.

Notable, the field of protein-structure relationship modeling is similar to the Quantitative Structure-Activity Relationships (QSAR) modeling field [85]. Both disciplines employ data-driven and statistical methods to predict the properties of biomolecules. However, QSAR focuses primarily on small molecules and applications in drug discovery [85].

In recent decades, protein sequence data was abundant while structure data was comparatively limited [34], and many prominent methods for protein property prediction use sequence-property relationship modeling. Notable sequence-based tools are the famous BLAST [86] and FASTA [87] programs, which perform similarity searches for homology detection against sequence databases. Machine learning tools like PredictProtein [88] or RaptorX-Property [89] predict structural protein properties, like secondary structure, solvent accessibility, and disordered regions based on the sequence. The tools presented in the following for applications on structure also often exist analogously for protein sequences. With its rising availability, structure data will likely become a more attractive and informative resource for calculating or predicting such properties directly from the structures.

### 1.3.1 Similarity Search Methods

An established approach to elucidating properties of uncharacterized proteins is by transferring annotations from similar well-characterized proteins [90]. Typically, a similarity search is conducted for a given query protein against a database of well-annotated proteins. Protein similarity can be described through evolutionary origin and relation. Proteins that share a common ancestor are called homologs. This relationship can be inferred and detected statistically through excess similarity in sequence or structure [90]. Homology detection was already very successful on sequences [86, 87, 90–94]. However, the structure is more conserved over the evolutionary time frame than the sequence [90]. The increase in protein structure data enables the application of similarity search to structure databases for detecting even more distant evolutionary relationships [59]. Structure-based methods that can perform similarity searches against structure databases for homology detection are, for example, DALI [95], mTM-align [96], or Foldseek [59]. In addition, tools focusing on specific sites of interest exist, for example, binding site similarity search tools, like SIENA [97].

In contrast to detecting homology, accurately inferring properties from homologous proteins is more complicated. For example, while there is generally the notion that homology implies a similar function, the functional similarity is hard to quantify [90]. Often assessing functionally critical residues, like the active site, is necessary to reduce errors when transferring annotations, especially for sequence-based homology detection [90]. Homology-based transfer becomes increasingly unsuited when the goal is to predict significant impacts on the property after minor protein modifications. For example, even single mutations can greatly affect the protein’s properties, like hindering binding or destabilizing the protein. Consequently, they can lead to a loss of function, drug resistance, and diseases [76].

### 1.3.2 Machine Learning Methods

Machine learning methods are a widely applied approach to protein property prediction. They can automatically use patterns in a data set to derive a mathematical model without explicitly specifying the details of the relationship, making them an appealing approach to model structure-property relationships [26, 27].

#### 1.3.2.1 Modeling with Machine Learning

Most machine learning methods applied to property prediction are probably supervised. Supervised methods learn the parameters of an adaptive statistical model by minimizing

the model’s prediction error for a target variable on a data set [98]. The model’s parameters are optimized during the training phase based on a training data set of known input samples and their associated target values. Trained models are usually evaluated on a test set of data with known target values, held out during the training phase to assess the model’s ability to predict unknown data points. Generally, supervised machine learning applications can be divided into classification and regression problems. In classification problems, the target variable is a finite number of discrete categories. The task is to assign the input sample to one of the categories [98]. In contrast, in regression problems, the target variable can be one or more continuous variables. Other important fields are unsupervised machine learning and reinforcement learning, which are not discussed in this work [98].

For structure-property relationship modeling, the input to a supervised machine learning model can be some representation of the protein’s three-dimensional structure, and the target variable could be experimentally measured property values [27]. The input structures must be described numerically as so-called features [27, 98]. There are various ways to define features for protein structures, for example, using physicochemical characteristics and geometric arrangements of atoms and residues [73, 99]. Geometric descriptions can include the torsion angles, distances, volumes, and surface, while physicochemical characterizations can be charge, hydrophobicity or functional groups [24, 99–101]. Features that describe known critical drivers of binding and folding, like noncovalent interactions, are frequently used [102]. Such interaction features are usually a combination of geometric and chemical descriptions. For example, hydrogen bonds can be described by the distance and angle of the donor and acceptor [103]. Subsequently, calculated hydrogen bonds can be combined to calculate secondary structure elements of the structure [104], compute protonation states [103], and more. While these features must be calculated before the machine learning fitting procedure with external tools or algorithms, features can also be derived during training.

Deep learning [36] methods try to automatically discover meaningful features from more basic features like atom positions and chemical elements with representation learning. In neural network methods, multiple processing layers try to learn multiple and increasingly abstract levels of representations from the raw features. A typical example is the task of object detection in an image [36]. The raw feature representation of an image is an array of pixels. The first neural network layer would learn features corresponding to edges in specific orientations and locations in the image. The second layer would detect edges in relative arrangements and the next would connect these edge arrangements into partial objects. In contrast, the last layer would predict which

object is in the image based on the combinations of the object parts. The key benefit of deep learning systems is that input features must not be designed by a human but can be learned automatically from the data with representation learning [36, 105].

### 1.3.2.2 Examples of Machine Learning Methods

This section provides an exemplary overview of structure-property relationship modeling approaches with machine learning, their methodological ideas, and the research questions to which they are applied.

Knowledge about the location of functional sites in a protein, like binding sites, is crucial for function prediction, drug discovery, and protein engineering. For example, the P2Rank [101] tool predicts ligand binding sites in protein structures using a Random Forest [106] classifier. The model scores local chemical environments of surface probes for their ligandability, the potential that the chemical environment would likely facilitate a small molecule ligand. Neighboring well-scoring probes are clustered to form the binding site prediction [101]. Subsequently, a binding site can be the basis for predicting other relevant properties. Tools like the DoGSiteScorer [24] predict pockets and estimate their druggability, which resembles the potential that the pocket can be addressed with a small molecule for pharmaceutical purposes. The tool uses a Support Vector Machine [107] with geometric and physicochemical features of the pocket. For example, a reliable estimate of a pocket’s druggability is relevant to prioritizing protein targets in drug discovery [24].

Predicting binding partners of proteins is essential for multiple applications. The MaSIF tool [108] predicts potential ligand and protein binding sites and protein-protein complexes by combining molecular surface interaction fingerprints with geometric deep learning. The fingerprints are learned from surface patches on the structure and their geometric and chemical features. Matching such fingerprints for site complementarity between biomolecules can enable efficient interaction prediction [108].

In drug discovery, structure-based virtual screening is an established approach to rank small molecules from a database by their potential to bind in the pocket of a target protein [109]. For this task, machine learning is used to assess the binding of different small molecules. Typically, three-dimensional poses of small molecules are generated with molecular docking tools, like FlexX [110], GLIDE [111] or AutoDock Vina [112], and subsequently scored and ranked with an additional machine learning-based scoring function. For example, the GNINA [30] method is a 3D-grid-based convolutional neural network and structure-based scoring function. GNINA is inspired by the object

detection task for images described above. Instead of detecting objects based on arrangements of edges in the image, the GNINA network tries to automatically learn more abstract representations of protein-ligand interactions that correlate with binding from protein-ligand complexes [30]. Another tool, DIFFDOCK [113], is a diffusion generative model that directly predicts a 3D structure of a bound protein-ligand complex without using classical molecular docking tools or a definition of the binding site. DIFFDOCK treats molecular docking as a generative modeling problem and learns a distribution over ligand poses defined by their translational, rotational, and torsional degrees of freedom to predict fitting ligand poses for a given protein structure [113].

Besides predicting the properties of a given protein, it is also vital to accurately predict how this property changes when there is a mutation in the protein. For example, the mCSM tool [99] predicts thermodynamic stability and affinity changes upon mutations as well as disease-related mutations using supervised machine learning. The features used by the algorithm describe the local environment of the mutation position in the structure through graph-based signatures and changes in pharmacophore features induced by the mutation. In the DynaMut [114] tool, the mCSM approach was combined with normal mode analysis to consider protein flexibility in the mutation effect prediction. Further, a 3D grid-based convolutional neural network by Torng et al. [115] predicts the likelihood of the 20 proteinogenic amino acid residues occupying a given local 3D environment in the structure. Such models can guide protein design and engineering [116].

## 1.4 Challenges in Data-Driven Modeling

Data-driven methods, like machine learning, are appealing because they enable predictions while avoiding the explicit modeling of complex structure-property relationships. However, challenges that must be overcome for their effective applications come from their inherent data dependence, the black box character of many data-derived relationships, and incomplete representations in the modeling process.

### 1.4.1 Data Limitations

Data scarcity and paucity are common problems. With small data set sizes, relationship modeling becomes more difficult. Much protein structure data is now available, but significantly less experimental protein property data. For example, similarity search methods, like homology detection, depend on the availability of databases of well-characterized proteins annotated with properties, like their function [18, 19, 117].

However, only a fraction of all proteins' properties have been and will be characterized experimentally [25]. This limitation also applies to machine learning methods, which need a certain amount of data to effectively capture the underlying patterns and variations of the property in the population [118], especially more expressive deep learning methods [36, 105].

Further, the available data is subject to sampling bias. Depending on the data collection procedures, specific population subsets can be over or underrepresented in the available data. For example, experimental structure determination of membrane proteins has been challenging [11]. Therefore, membrane proteins are underrepresented in current experimental structure databases [119, 120]. In addition, data collection is usually driven by specific research interests and projects. For example, in the hit-to-lead phase in drug discovery, hit compounds binding the target protein are optimized in focused explorations, also called molecular series, to identify similar binding compounds with properties suited for a drug [121]. The protein-ligand binding data collected from such efforts can be narrowly clustered, which makes predictions outside the cluster difficult for models built from that data [118, 122, 123].

Another frequently occurring challenge is imbalanced data, where a small minority of the samples hold interesting target values, like biomolecules with desired property profiles. In contrast, most samples have uninteresting target values, i.e., undesired property profiles. Imbalanced data sets challenge most standard learning algorithms and evaluation approaches, especially with small sample sizes [124]. While handling imbalanced data is still an active area of research, different approaches aim to alleviate imbalance, ranging from data sampling techniques to cost-sensitive learning algorithms and novelty detection methods [124, 125].

Data quality and experimental uncertainty are important factors to consider. The determined three-dimensional protein structure is a model derived from experimental measurements [126]. For example, a crystallographer derives the atomic structure model from fitting the electron density from an X-ray crystallography experiment. Insufficient density and resulting ambiguities in the structure modeling process can lead to uncertainties and missing atoms, especially in more flexible areas like the side chains of surface residues, solvents, and loops. Even entire domains can be missing [126]. Structure files from the PDB provide additional information on experimental uncertainties of a structure model, like resolution, alternate locations, and more. Further, how well the single atoms of the structure model match the electron density can be quantified and assessed with tools like the EDIA scorer, which scores individual atoms for their electron density support considering neighboring atoms and unaccounted density [127].

Similarly, experimental measures of protein properties are subject to uncertainty and noise [128, 129]. Analysis of deviations in repeated experiments or measurements of the same systems allows for estimating these uncertainties. However, comparing experiments is complicated because most public data comes from different laboratories and was measured with various conditions, protocols, and assays. Still, estimation approaches exist; for example, Kramer et al. [128] estimated uncertainty of protein-ligand affinity measures in terms of  $K_i$  values, and Montanucci et al. [129] evaluated the uncertainty of  $\Delta\Delta G$  protein stability changes upon single mutations. Experimental uncertainties constitute a natural upper bound to predictive models [128, 129].

Finally, data limitations have not only implications on building the predictive models but also on evaluating them. Test data is subject to the same limitations. The ability of the test set to represent a prospective evaluation well can be restricted under these limitations. Therefore, it is not only challenging to build models but also challenging to recognize working solutions.

### 1.4.2 Protein Representations

Proteins must be sufficiently represented for the property prediction task. Most protein structural data in standard databases like the PDB is typically deposited as single structures. However, richer representations than single structures can add enabling information for property prediction. A famous example of an ensemble-based protein representation from the domain of protein structure prediction is multiple sequence alignments (MSAs), which are the input to AlphaFold2 [28] and RoseTTAFold [29]. MSAs are sets of sequences folding into a similar structure from which co-evolutionary signals can be extracted to derive structural constraints, a cornerstone for structure prediction [28, 29, 130]. Therefore, representing proteins for structure-property modeling not only with a single structure but with an ensemble is a formidable challenge, especially given the recent abundance of structure data.

Structure ensembles can represent protein flexibility, which is not well expressed with a single rigid structure. Protein flexibility is indispensable for adequately describing structure-property relationships since it influences function, stability, energetic properties, and the binding of small molecules [131]. Generally, protein flexibility can be represented in multiple ways. X-ray structures have experimentally derived annotations describing the uncertainty of atom positions. B-factors describe the attenuation of X-ray scattering due to thermal motion, with which the flexibility of atoms, side chains, and regions can be identified and analyzed [132]. The electron density in X-ray experiments can also describe multiple discrete locations for an atom, residue, or larger



parts of the structure. Such states are modeled as alternative locations, also called AltLocs. However, AltLocs are usually ignored by practitioners and users of the structure models [126]. Protein flexibility can also be explored with methods like molecular dynamics (MD) simulations, which predict the spatial positions of atoms over time utilizing molecular mechanics force fields [66]. However, MD simulations are limited to small time frames by numerical stability issues and high computational demand [66]. Furthermore, structure ensembles of the same protein in different conformations can be assembled from a database. For example, the SIENA [97] tool can compile structure ensembles of ligand binding sites clustered by the sites' rigid regions, highlighting flexible parts of the binding site. From these approaches, database searches and the enumeration of AltLoc conformations can provide experimentally validated protein conformations as structure ensembles.

Another critical challenge is identifying changes in protein properties in a desired direction, which is crucial for finding an inhibitor for a target protein in drug discovery or identifying mutations that increase the thermodynamic stability of an enzyme in protein engineering. However, a single structure only represents a single state, which is insufficient to represent the change in the structure upon binding or mutations. Solvent displacement and considerable conformational changes can occur upon binding [79], but large conformational changes are hard to model. Similarly, the structural changes upon mutation between wild-type and mutant are essential determinants of the mutation effect [84, 133]. A single structure can not fully represent the different states of such events and variabilities. Again, structure ensembles representing both states can improve protein representations in these cases.

Generally, structure ensembles can be compiled through sequence similarity tools, like BLAST [86] or MMseqs2 [92]. However, these tools primarily focus on homology detection. They can not be directly used for relevant downstream tasks, such as the structural analyses of 3D sites, like ligand binding or mutation sites. Dedicated tools like SIENA [97] can directly provide binding site ensembles for analyzing binding site flexibility or structural differences between apo and holo-structures. Methods for comparing structural sites, such as ligand binding sites, are well-established [61], whereas dedicated analysis methods for other sites, such as mutation sites, are mostly unavailable.

### 1.4.3 Interpretability

Due to their implicit nature, predictive models can be somewhat black boxes that are not directly interpretable [134]. This applies in particular to many competitive machine learning models. However, knowledge about the learned relationship can be



invaluable. Typically employed machine learning models do not differentiate between correlation and causation. Disclosing what a model has learned helps to analyze if the model is generalizing and is not biased or misinformed [134]. Explanations help to build trust in a prediction model and help experts make model-informed decisions [134]. In addition, explanations can be used to find clues for deciphering unknown structure-property relationships.

Most interpretability approaches aim to provide explanations regarding feature importance [134] by determining features and sometimes feature values most influencing predictions.

Interpretability approaches can be broadly grouped into global and local explanation approaches, also called model and instance approaches [134, 135]. Local approaches try to extract explanations for single instances and their corresponding prediction. In contrast, global approaches extract explanations from a model’s predictions for a whole data set. Some models are considered white box models since they are easier for humans to interpret. These are usually simpler models, like small decision trees and linear models. Surrogate approaches try to fit white box surrogate models with high accuracy on top of a black box model. LIME [136] and SHAP [137] are popular examples of local surrogate methods [134]. Further, interpretability approaches can either be model-specific or agnostic [134]. For example, Random Forest [106] has a model-specific approach integrated by design and provides interpretable variable importance from internally derived estimates. Another popular global but model-agnostic approach are Partial Dependence Plots [138], which aim to display how individual features contribute to the model by varying each feature [134].

The application of interpretability approaches still poses a challenge since there is no one-fits-all solution. Usually, more exhaustive interpretability approaches are more demanding in computational resources and the quality of explanations is hard to quantify [134]. Therefore, careful analysis and application-dependent decisions are often required.

## 1.5 This Work

This work addresses challenges in data-driven structure-property modeling common to multiple fields in protein property prediction. The four main publications of this cumulative dissertation present three software methods and two in-depth analysis studies.

This doctorate project started with two analysis studies during which a machine learning-based interpretability method was developed to aid the analyses. The studies’ findings disclosed crucial domain-specific patterns in data sets for structure-based

virtual screening and the high-pressure adaptations of proteins. The overarching conclusions from the analysis studies were that data limitations often restrict structure-property modeling and prediction. As an attempt in the second half of this doctorate project to overcome such shortcomings and effectively and more fundamentally improve multiple facets of structure-property modeling, two dedicated software methods were developed that exploit the vast existing protein structure collections and annotations to improve protein representations with structure ensembles.

The following chapter presents the four main publications of this cumulative dissertation.

Section 2.1 considers analyzing and evaluating unbiasing strategies of benchmark data sets in structure-based virtual screening for drug design. An interpretability method was developed, which comprehensively uses feature selection with wrapper methods to quantify feature importance on a whole data set using baseline machine learning methods. Various feature sets were analyzed with the method and trends identified. The approach can be classified as a global interpretability method [134] that explains an ensemble of multiple baseline models instead of only one model. Therefore, the developed method comprehensively characterizes important features and depicts how well a collection of baseline methods performs on a particular data set and prediction task. This method was the basis for analyzing benchmark data sets' unbiasing strategies, with which essential data limitations, like the unsuitability of these unbiasing strategies for machine learning, could be characterized. In particular, it could be shown that machine learning methods learn combinations of simple, highly distinguishable features between active and decoy small molecules, even though these features were expected to be uninformative because of the unbiasing strategies.

Section 2.2 applies the developed interpretability method to interpreting high-pressure protein adaptation for protein engineering. For this purpose, a data set of matched protein structure pairs from high-pressure and other habitats was collected, and a diverse collection of protein features was computed. The interpretability method was extended with individual feature attributions using the Shapley value framework on the evaluated feature sets. The attribution was used to interpret feature importance based on the features' average performance increase in the feature selection with wrapper methods experiment. The extended method was applied to compare features important on different data subsets to isolate and characterize protein features potentially related to proteins' high-pressure adaptations.

Section 2.3 presents the AltLocEnumerator method developed to enumerate experimentally observed alternative protein conformations based on AltLoc annotations to

represent the protein’s flexibility through structure ensembles. The method efficiently extracts AltLoc annotations from protein structure files, checks valid overall protein conformations, and provides the resulting protein conformations as an ensemble. Additionally, filters and options help to focus large conformation ensembles to a manageable size for subsequent processing and structure-based tasks.

Section 2.4 describes the MicroMiner tool, which implements a new method for searching similar local 3D micro-environments in protein structure databases. A novel perspective is introduced for searching and compiling structure ensembles of local residue-centered structural protein sites, termed ‘residue 3D micro-environments’. The method was applied to the scientific key application of structural single mutation analysis. MicroMiner was used to extract all amino acid pairs in protein structures, exemplifying the structural changes of single mutations from the PDB. Subsequently, the extracted data was used to annotate existing mutation effect measures with structures for the mutant to combine the mutation effect with the structural change upon mutation.

## 1.6 Overview of Scientific Contributions

This cumulative dissertation consists of five peer-reviewed publications. The four first-author publications (including one shared first-author publication) and the one co-authored publication are summarized with individual contributions in Appendix A.1.

The four first-author publications are:

**J. Sieg**, F. Flachsenberg, and M. Rarey. “In need of bias control: evaluating chemical data for machine learning in structure-based virtual screening”. In: *Journal of chemical information and modeling* 59.3 (2019), pp. 947–961.

Note that a subchapter of the above publication was produced in J. Sieg’s Master’s thesis [139], which is not part of this dissertation. See section 2.1.2 for a description of which parts are preliminary work not conducted during this doctorate project.

**J. Sieg**, C. C. Sandmeier, J. Lieske, A. Meents, C. Lemmen, W. R. Streit, and M. Rarey. “Analyzing structural features of proteins from deep-sea organisms”. In: *Proteins: Structure, Function, and Bioinformatics* 90.8 (2022), pp. 1521–1537.

T. Gutermuth, **J. Sieg**, T. Stohn, and M. Rarey. “Modeling with Alternate Locations in X-ray Protein Structures”. In: *Journal of Chemical Information and Modeling* 63.8 (2023), pp. 2573–2585.

The publication above is a shared first-authored publication. T. Gutermuth and J. Sieg contributed equally to this work.

**J. Sieg** and M. Rarey. “Searching similar local 3D micro-environments in protein structure databases with MicroMiner”. In: *Briefings in Bioinformatics* 24.6 (2023), bbad357.

The co-authored publication is:

K. Schöning-Stierand, K. Diedrich, C. Ehrt, F. Flachsenberg, J. Graef, **J. Sieg**, P. Penner, M. Poppinga, A. Ungethüm, and M. Rarey. “Proteins Plus: a comprehensive collection of web-based molecular modeling tools”. In: *Nucleic Acids Research* 50.W1 (2022), W611–W615.

## Chapter 2

# Publications

### 2.1 Analyzing Benchmark Data Sets for Virtual Screening

This section presents an analysis of benchmark data sets for structure-based virtual screening [D1], which was part of this doctorate project. The goal was to evaluate whether benchmark data sets designed for conventional structure-based methods, such as docking with empirical and knowledge-based scoring functions [102], are also suitable for scoring functions based on standard machine learning methods. For this purpose, a new interpretability method was developed based on feature selection to evaluate to what extent common data set unbiasing techniques work for machine learning methods.

#### 2.1.1 Motivation

Virtual screening is a field in the early stages of drug design aimed at prioritizing large *in silico* molecular libraries for the likelihood that the molecules are active at a particular protein target [109]. A distinction can be made between ligand-based and structure-based virtual screening. In ligand-based screening, the prediction is made based on the similarity of a molecule with unknown activity to molecules with known activity. Frequently applied methods include substructure searches, quantitative structure-activity relationship (QSAR) analysis, pharmacophore analysis, and 3D shape matching [140]. On the other hand, structure-based virtual screening models the interactions of a small molecule with unknown activity to the protein target based on the 3D protein structure [109]. The most popular method is molecular docking, in which potential 3D binding poses of the query molecule in the protein binding pocket are calculated, and their complementarity or affinity is assessed [140]. In recent years, machine learning and deep learning methods have been applied more frequently to virtual screening, for

example, in scoring functions for the ranking of docking poses [102, 141] and are seen as promising techniques in the field [141].

Retrospective evaluation of virtual screening methods using benchmark data sets has become a standard procedure [123, 142–144]. Popular data sets include the Directory of Useful Decoys (DUD) [142], the Directory of Useful Decoys - Enhanced (DUD-E) [144], the Maximum Unbiased Validation-dataset (MUV) [143], the Demanding Evaluation Kits for Objective in silico Screening (DEKOIS) [145], the Community Structure-Activity Resource (CSAR) [146], and PDBbind [147]. These data sets are frequently used to evaluate how well a method can prioritize active molecules in an extensive set of inactive molecules. Although most of the data sets were designed with a focus on conventional scoring functions, such as empirical and knowledge-based scoring functions, they are also increasingly used to train and test more potent data-driven methods from the machine learning [148, 149] and deep learning field [30, 150–152].

Whether these data sets and their assumptions are suited for training and validating machine learning methods was evaluated in this doctorate project by building a method to measure how well their unbiasing strategies work with machine learning [D1].

### 2.1.2 Preliminary Work

A part of the publication [D1] is not part of the doctorate project described in this dissertation because they were created before the doctorate period. Specifically, the observations and analyses described in [D1] in section "4. NON-CAUSAL BIAS IN LITERATURE" and the resulting conclusions were made outside the doctoral project. In the following, the analysis and resulting conclusions not included in the doctorate project are shortly summarized for a clear distinction.

It was observed that only the ligand features were sufficient for the structure-based deep learning method DeepVS [151] to achieve competitive performance on the DUD and DUD-E data sets. Therefore, DeepVS was re-implemented with different descriptors expressing only ligand information. The key finding was that molecular 2D features of small molecules are discriminative features in the DUD and DUD-E datasets that separate the set of active molecules from the set of decoys with high accuracy. Topological descriptors, such as substructure fingerprints, can describe these 2D features. It was shown that the difference in 2D features is so concise that the protein structure information of DeepVS is hardly used by the deep learning method, although it is a structure-based method. Moreover, the separation of actives and decoys based on 2D features is detectable on the whole data set, irrespective of the associated protein targets of the small molecules, which allows for the artificial distinction of actives and

decoys of unrelated targets. It was concluded that the decoy selection strategy based on property-matched decoys used in DUD and DUD-E leads to this artificial performance of 2D features. The authors of DUD and DUD-E applied the 2D dissimilarity filter of the decoy matching protocol across the entire data set, resulting in the artificial separation of actives and decoys on all protein targets collectively. Therefore, virtual screening methods that use 2D features have an artificial advantage on these data sets. Especially for highly flexible machine learning methods with black box characters, like deep learning methods, it is hard to determine which signals of the input are picked up by the model. Care must be taken because standard procedures such as cross-validation are insufficient to detect bias when using these datasets [D1].

In this doctorate project, multiple analyses building up from this preliminary work were performed to analyze further the suitability of the assumptions made by the data sets compilation protocols for the application with machine learning methods. The goal was to build a method to measure which data sets are better suited for machine learning methods than others.

### 2.1.3 Unbiasing Strategies in Benchmark Data Sets

Standard benchmark data sets in structure-based virtual screening were created to minimize biases such as 'artificial enrichment' [153] and 'analogue bias' [122]. Artificial enrichment describes that the decoy set or inactives set differs significantly from the active molecules in simple properties, for example, molecular weight, number of hydrogen bond donors or acceptors (see Table 2.1). On the other hand, analogue bias describes that actives with the same chemotype are overrepresented in data sets, for example, because the actives come from the same molecular series. The three data sets DUD, DUD-E, and MUV evaluated here implement strategies to minimize such biases. A short overview of the data sets unbiasing strategies is given in the following.

DUD [142]	DUD-E [144]	MUV [143]
molecular weight	molecular weight	
number of hydrogen bond acceptors	number of hydrogen bond acceptors	number of hydrogen bond acceptors
number of hydrogen bond donors	number of hydrogen bond donors	number of hydrogen bond donors
number of rotatable bonds	number of rotatable bonds	
logP	logP	logP
	net charge	
		number of all atoms
		number of heavy atoms
		number of boron atoms
		number of bromine atoms
		number of carbon atoms
		number of chlorine atoms
		number of fluorine atoms
		number of iodine atoms
		number of nitrogen atoms
		number of oxygen atoms
		number of phosphorus atoms
		number of sulfur atoms
		number of chiral centers
		number of ring systems
5 features	6 features	17 features

**Table 2.1:** List of the unbiased features of DUD, DUD-E and MUV. This table was taken from [D1].



The DUD data set [142] was created to benchmark docking methods. The compilation protocol uses so-called property-matched decoys. Specifically, experimentally validated active molecules are paired with assumed inactives, also called decoys, from the ZINC database [154]. The decoys are selected to be similar to the actives regarding simple properties (see Table 2.1) to reduce artificial enrichment. Simultaneously, to ensure that the selected decoys are not active, a dissimilarity filter was applied such that each decoy is dissimilar to any active using CACTVS fingerprints [155]. The DUD-E data set [144] extends DUD and addresses some of its shortcomings. For example, artificial enrichment was further reduced by including net charges in the matched properties (see Table 2.1). In addition, analogue bias was addressed by including only the cluster representatives of the scaffold clustered sets of actives. In contrast, the data sets of the MUV collection are initially designed for ligand-based virtual screening but can also be used for benchmarking structure-based methods [143]. MUV contains both experimentally validated actives and inactives from the PubChem database [156]. Molecules are selected to reduce artificial enrichment and analogue bias by enforcing a common spread between actives and other actives as well as actives and inactives in a 17-dimensional descriptor space of simple features (see Table 2.1). This spread should ensure that an active molecule can not be classified correctly based on its nearest neighbor in this simple feature space.

### 2.1.4 Methodological Summary

The interpretability method developed in this doctorate is based on feature selection. Feature selection methods provide a toolbox for analyzing the multivariate feature distributions of a data set considering the target variable [157]. While their main task is to select features for building a particular machine learning model, they can also help to understand which features are important for a task and which models and features are sufficient on a particular data set. Consequently, feature selection methods can be used to evaluate the prediction performance of unbiased features (see Table 2.1) and check whether the unbiasing holds for machine learning methods.

Feature selection with wrapper methods describes approaches that use a particular machine learning model as a black box to evaluate feature subsets for their predictive power in an evaluation experiment [157]. Accordingly, in the wrapper methodology, models are trained and tested with different feature sets, for example, using cross-validation. The number of possible feature sets that can be evaluated is the number of combinations of the individual features. For  $n$  features, the number of feature sets is defined by

$$\sum_{k=1}^n \binom{n}{k} = 2^n - 1$$

Due to the exponential number of feature sets and the relatively demanding computation process of training and evaluating a model with each feature set, this methodology becomes computationally expensive as the number of features increases. To analyze higher dimensional data sets, greedy strategies like backward elimination or forward selection can be applied, which heuristically traverse the space of feature sets [157].

In this work, wrapper methods were chosen because they evaluate the performance of combinations of features rather than considering individual features separately. This analysis focuses not on building an optimal predictive model but on evaluating whether and to what extent standard machine learning methods can exploit unbiased features of standard benchmark data sets. In this scenario, good performance means that the feature unbiasing does not work as intended since close to random performance is expected through the unbiasing procedure.

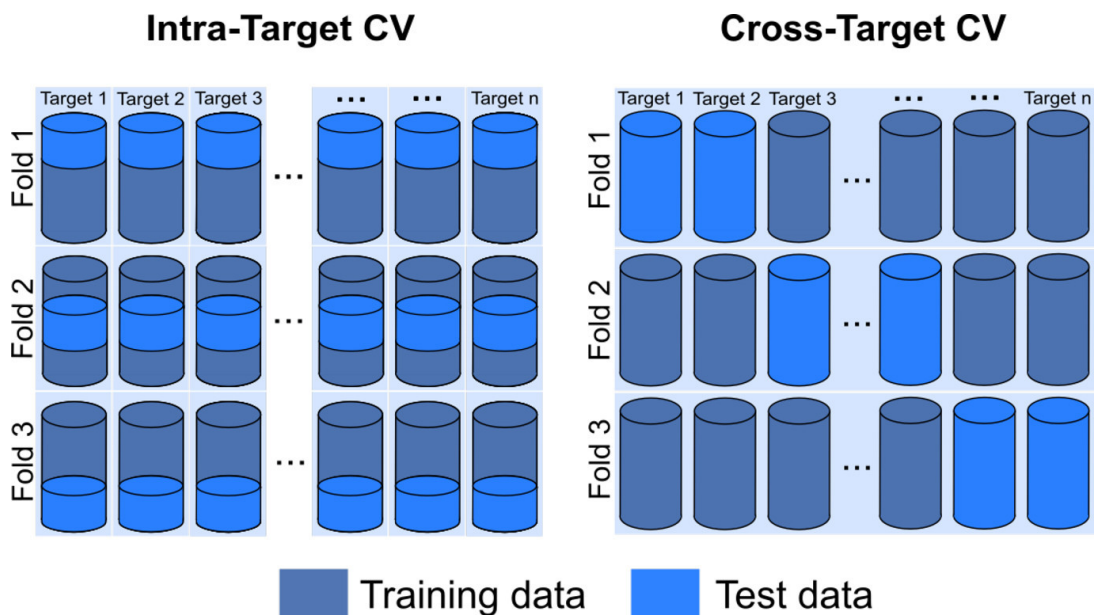
The two cross-validation (CV) scenarios illustrated in Figure 2.1 were used for evaluation. In intra-target CV, the data set of molecules for a single target is split into training and test sets. In contrast, in cross-target CV, the molecules of a specific target are either collectively assigned to the training or test data set. The second split is more challenging since a model needs to predict across potentially unrelated targets.

The Random Forest classifier and Logistic Regression were used as machine learning algorithms. For DUD and DUD-E it was feasible to evaluate each possible feature set. For MUV, backward elimination was employed because of the high number of feature sets. MUV was omitted from the cross-target CV experiment because it contains many duplicate inactives across targets. Removing them would yield an arbitrary data set not resembling the MUV unbiasing.

### 2.1.5 Evaluation of Unbiasing Strategies

The results of the feature selection analysis are shown in Figure 2.2, exemplarily for Random Forest. However, especially for DUD and DUD-E, the unbiasing techniques do not work well with machine learning, which is illustrated by the excellent prediction performance when their unbiased features are used.

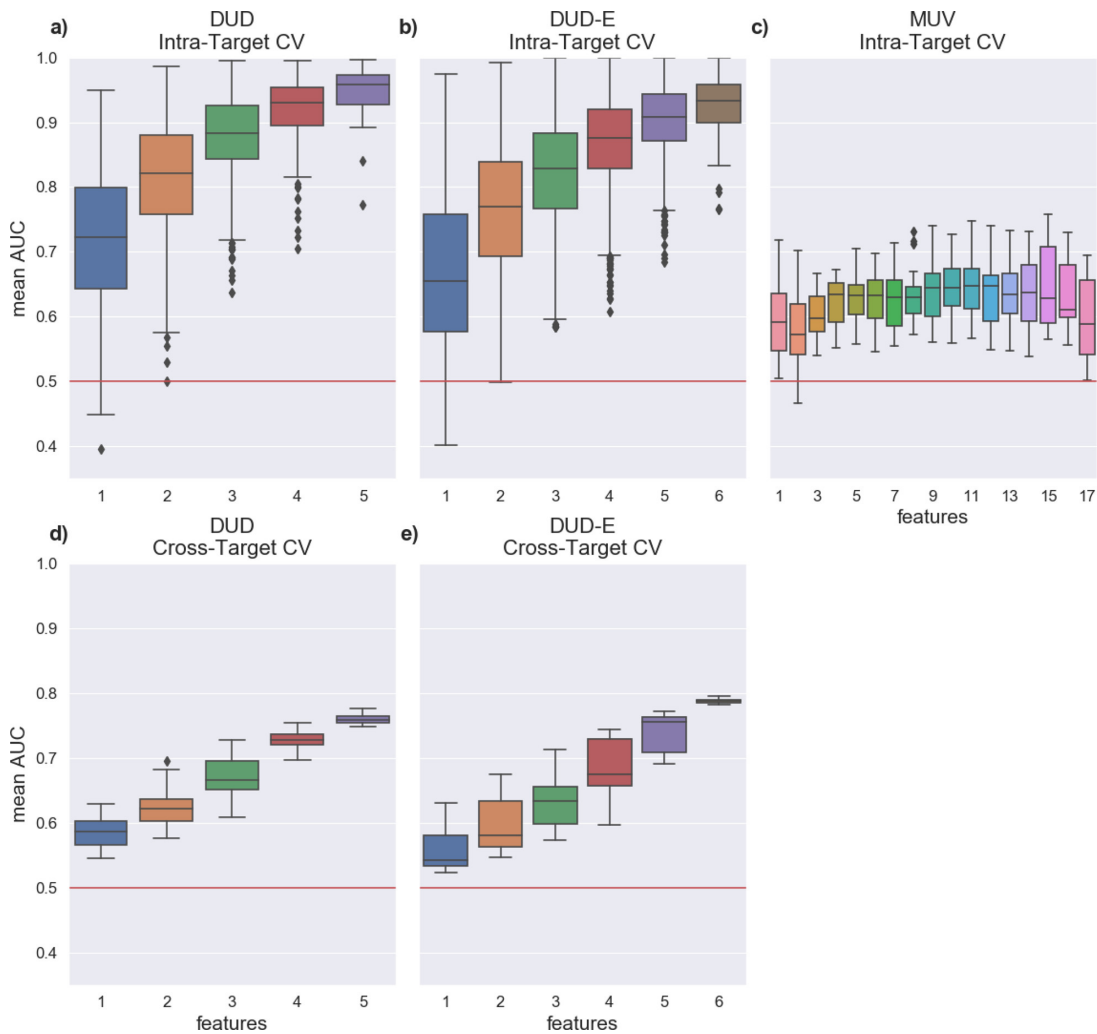
The analysis showed that even single unbiased features result in good prediction performance much better than a random guess, especially for the intra-target CV (see Figure 2.2a and b). This was surprising because the distributions of individual features were expected to be matched between actives and decoys. In the example of the number



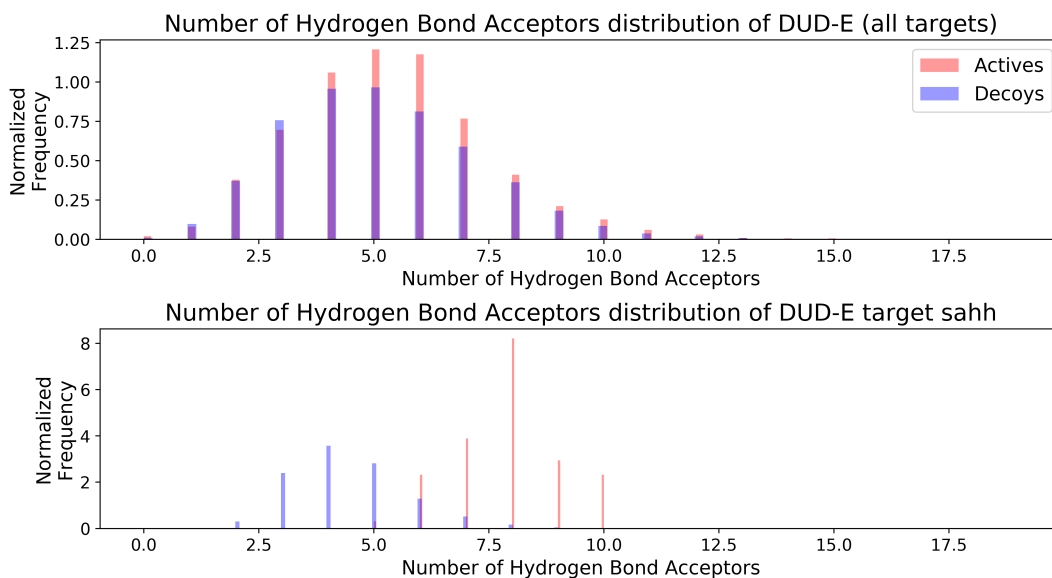
**Figure 2.1:** Cross validation (CV) scenarios used in the experiments, namely intra-target CV and cross-target CV, are depicted schematically. Intra-target CV evaluates prediction performance on a single target, while cross-target CV evaluates predictions to new targets. This figure was taken and adapted from [D1].

of the hydrogen bond acceptors, shown in the first row in Figure 2.3, the mean and variance seem to be matched on the level of the whole data set. However, on the level of individual targets, the distributions can still be distinct, as shown for target sahh in DUD-E in the second row in Figure 2.3. Furthermore, even when the feature distributions are better matched and not linearly separable, non-linear machine learning methods like Random Forest can separate actives from decoys if test actives are more similar to training actives than training decoys. The contour plots in Figure 2.4 illustrate this on the target pnp in DUD.

Another observation for DUD and DUD-E is a positive correlation between the number of unbiased features used and the performance (see Figure 2.2a, b, d, e). This suggests standard machine learning methods can capture synergies when combining multiple unbiased features. Surprisingly, almost perfect prediction performance can be achieved in the intra-target CV and competitive performance on the cross-target CV. These results demonstrate that simple and unbiased features of small molecules can have a considerable influence when evaluating structure-based methods on these benchmark data sets. Especially, the performance in the cross-target CV discloses artificial and overoptimistic evaluation performance, which illustrates that the unbiasing strategies are not suited for machine learning.



**Figure 2.2:** Evaluation of the unbiased features with Random Forest. The first row shows the results of the intra-target CV, and the second row shows the results of the cross-target CV. On the x-axis, the number of features is shown, and on the y-axis, the mean ROC AUC of the CVs. A box illustrates the range of performance in terms of ROC AUC overall targets when  $x$  features are used. The red line marks the random performance of a ROC AUC value of 0.5. This figure was taken from [D1].



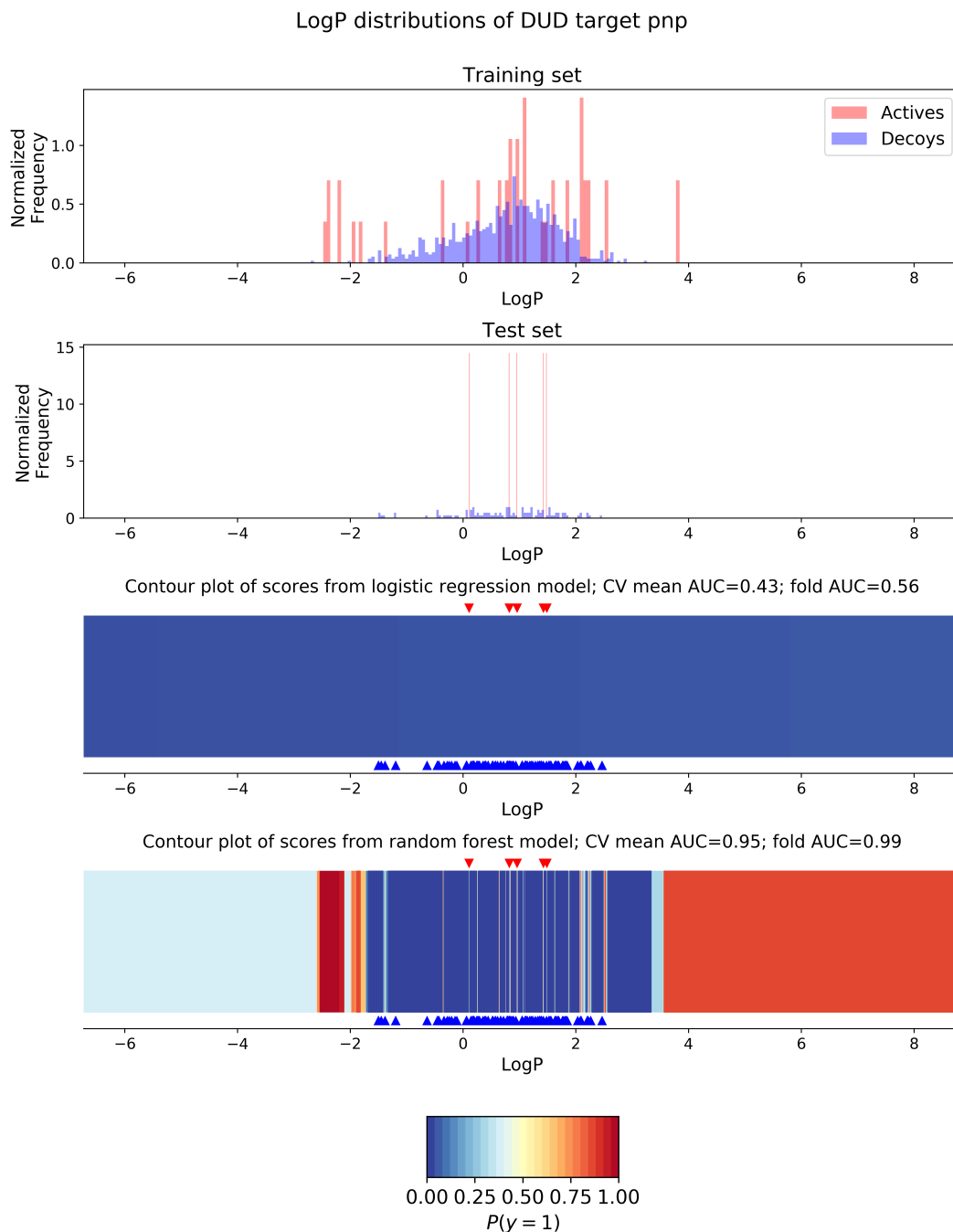
**Figure 2.3:** Comparison of the distributions of the matched property 'Number of Hydrogen Bond Acceptors'. The first row shows the distributions over all targets in DUD-E. The second row shows the distributions only on the example of the target sahh. This figure was taken and adapted from [D1].

On MUV, the performance of unbiased features is substantially lower, but the predictions are still better than a random guess. In contrast to DUD and DUD-E, no correlation between the performance and the number of features can be observed, probably because the MUV unbiasing strategy tries to reduce bias in a 17-dimensional feature space in contrast to matching distributions of single features.

In conclusion, the described analysis suggests that the benchmark data sets' unbiasing strategies are unsuitable for machine learning. While the unbiasing strategy of MUV might be better suited than DUD and DUD-E, it also has serious limitations [123]. The feature selection method developed here enabled the detailed analysis of the unbiasing strategies, determining which and how many features are important for solving the target task on the data set. With the method, the extent of bias could be assessed. Therefore, when using these data sets to evaluate a new method, it is important not to compare the method's performance to random predictions but to the baseline performance given by the unbiased features.

### 2.1.6 Outlook

After publication of [D1], other studies [158] were published supporting the described findings and came to similar conclusions.



**Figure 2.4:** Example of the linearly non-separable case in the intra-target CV experiment with the single feature LogP on the protein target pnp of DUD. The first and second rows depict the feature distribution on the training and testing set of one fold of the CV, respectively. The third and fourth rows show Logistic Regression and Random Forest contour plots. The red triangles on the top of the contour plots mark the LogP value for the test actives, and the blue triangles at the bottom mark the LogP values of the decoys in the test set. This figure was taken and adapted from [D1]. Contour plots were recomputed using the same script and data as in [D1], but with a different random seed, exhibiting negligible differences.

The analysis in this doctorate project concluded that new data sets suitable for machine learning are needed to overcome the identified problems. An inherent and challenging problem is the data limitations in currently available data sets, for example, from the unavailability of inactives and the similarity of actives. New data sets are published that try to address these biases, such as LIT-PCBA [159], which provides baseline performances for different methodologies and uses an extended version of the MUV bias scoring [123] for building bias-reduced training and test splits for machine learning. However, other studies suggested that these unbiasing scoring functions do not help to improve the ability to generalize [160]. Alternatively, instead of integrating unbiasing strategies in the data selection protocol of data sets, subfields of representation learning, like algorithmic fairness [161], try to integrate unbiasing strategies into the learning process of machine learning models.

Building suitable benchmark data sets in the future might be achieved through larger community efforts that systematically generate new data, the industry making inactive data available, careful curation, and comparing to baseline methods instead of random performance. In addition, community challenges like D3R [5] or CACHE [6] aim to provide a competition platform supplying prospective testing to foster innovative methodology. Ultimately, to progress in the field, not only the methodology but also the techniques and data sets used for validating new methods have to improve.

## 2.2 Analyzing Structure Data Sets for Protein Adaptations

In the second doctorate project [D2] of this work, protein structures were analyzed for adaptations to the conditions of their source organisms' habitat. The scientific goal of this study was to decipher protein adaptations to high pressure by a comparative analysis of protein features. The interpretability method described in section 2.1.4 was extended and applied to the analysis of protein adaptations. The methodological enhancement uses the Shapley values framework on the output of the before-developed feature selection procedure [D1] to attribute importance to individual features. The resulting per-feature importance estimates were applied to isolate predictive features and interpret their relationship to protein adaptations.

### 2.2.1 Motivation

Proteins functional under extreme conditions are highly relevant for improving biotechnological processes in fields like pharmacology, agriculture, and biofuel production [162]. Such proteins occur naturally in extremophilic organisms inhabiting environments with

prevailing extreme conditions. The deep sea is one of the largest terrestrial extreme environments. It is particularly interesting because it simultaneously poses its inhabitants under multiple extremes, including extreme temperature, pressure, and salt concentration ranges [162–164]. Driven by environmental metagenomic efforts, recent years have seen an increased understanding of life forms from extreme environments, and a growing number of genome and protein data became available [162]. Despite the progress, the effective engineering of proteins to function under extreme conditions remains a long-standing challenge [68, 162].

Accurately modeling the structure-property relationship is crucial for rational design and computational methods for designing and engineering proteins. However, the knowledge of this relationship is limited for many protein adaptations. For example, high-pressure adaptations are not well-described, while protein adaptations to high temperatures are described the most extensively [162, 165]. Still, a profound understanding of high-temperature adaptations enabling large-scale protein engineering is yet to be derived [68]. It is presumed that high-temperature adaptations are a complex and context-dependent combination of multiple factors that are difficult to disentangle [68]. In addition, most extremophilic organisms are exposed to multiple extremes in their environments. Therefore, their proteins potentially exhibit multifactorial adaptations [166], making it even more challenging to disentangle the protein features of specific stability adaptations.

Optimizing proteins for extreme conditions is a crucial challenge for protein engineering. The currently available protein data of extremophiles might hold valuable clues that can be exploited to unravel the factors responsible for adaptations, but the data was not analyzed at scale yet. Comparing not only the proteins of organisms of two environments but multiple different environments might help decipher multifactorial and individual adaptations to extremes.

The scientific question addressed in this project [D2] was whether the currently available structural protein data of deep-sea organisms could be used to generate new insights into protein adaptations to high hydrostatic pressure. Unique to this study is the generation of a large-scale protein structure data set of orthologous protein pairs from organisms of multiple environments from which relevant protein characteristics were isolated using machine learning-based feature importance attribution.

### 2.2.2 Data Set Creation

A data set of matched protein structure pairs was created. The idea is to pair a protein of a deep-sea organism with a similar related protein from an organism of a different



environment. In this approach, the source environments of the proteins are used as surrogates for the property of stability towards extreme conditions. This assumption is necessary since there is little data on experimentally determined protein pressure stability. The creation of pairs of similar proteins should emphasize the structures' remaining and likely nuanced differences for the analysis of adaptations whose characteristics are believed to be subtle.

The data set was generated by collecting the names of deep-sea organisms from the literature. Then, the available experimental structures of deep-sea organisms in the PDB [33] were extracted. The individual chains of the structures were used to find potential orthologous chains in the PDB using the sequence similarity search and comparison tools HHsearch [93] and needle [167] and the structure comparison tool TM-align [168] to generate structure chain pairs with a high probability of evolutionary relationship. The chains from deep-sea organisms are called deep-sea structures, and the paired chains are called decoy structures in the following.

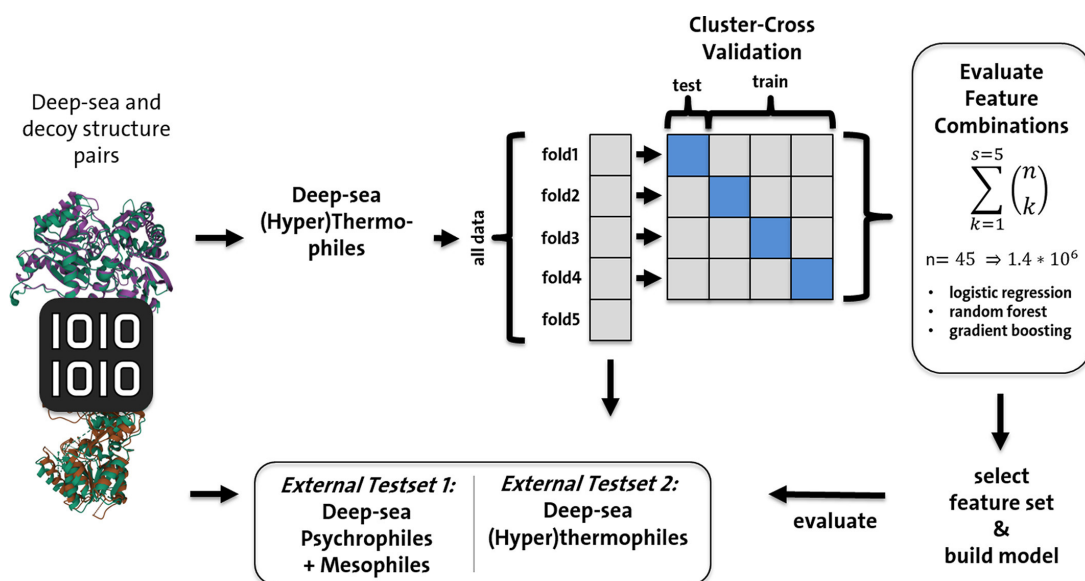
### 2.2.3 Analytical Workflow

The generated data set was analyzed for high-pressure adaptations with the workflow illustrated in Figure 2.5. This top-down approach aimed to detect global protein adaptations of the population of deep-sea proteins in contrast to adaptations in certain protein families or organisms. Machine learning-based feature selection was employed in a cluster CV scenario. The protein pairs were clustered by sequence similarity while keeping the pair relation. Multiple clusters were grouped into five folds. Therefore, the models should pick up features descriptive of protein characteristics in different protein families during training, which can then be directly assessed on the test set.

The workflow steps from Figure 2.5 are summarized in the following. A detailed description can be found in [D2].

The compiled data set was split based on the deep-sea organisms' optimal growth temperature in the first step. The pairs with structures from deep-sea (hyper)thermophiles are used for feature selection, while the scarcely available pairs of structures from deep-sea psychrophiles (prefer cold temperature) and deep-sea mesophiles (prefer moderate temperature) were used as an external test set. The pairs from deep-sea (hyper)thermophiles were used in the five-fold cluster CV, and one of these folds was set aside as a second external test set.

An additional partitioning was performed based on the decoy structure's source organism used in the CV. The data was split into four overlapping subsets listed and described in Table 2.2.



**Figure 2.5:** The workflow for analyzing protein adaptations in deep-sea proteins. The data set of matched protein pairs was split based on the optimal growth temperature preferences of the deep-sea source organism. Structure pairs of deep-sea (hyper)thermophiles were used for feature selection, while pairs from psychro- and mesophiles were used as external test sets. Exemplarily, it is illustrated how 1.4 million structure feature combinations are evaluated. The best-performing sets are then validated on the two external test sets. This figure was taken from [D2].

set name	short description
DecoyAll	All decoy organisms
MesoModel	Mesophilic model organisms
ThermoAll	All thermophilic organisms
ThermoModel	Thermophilic model organisms

**Table 2.2:** Overview of overlapping subsets based on the decoy structures' source organisms. The 'DecoyAll' set is the complete set of deep-sea/decoy structure pairs from deep-sea (hyper)thermophiles. The 'MesoModel' set comprises decoy structures from mesophilic model organisms, for example, *Homo sapiens* and *Escherichia coli*, 'ThermoAll' contains pairs with thermophiles from the literature, and the 'ThermoModel' set is a subset of the 'ThermoAll' set holding only well-studied model thermophiles, like *Thermus thermophilus* and *Thermotoga maritima*. This table was taken and adapted from [D2].

25 sequential and 45 structural features were computed on each protein chain of the data set. Sequential features were, for example, the relative frequency of amino acid residues and their physicochemical properties. In contrast, features computed on the structure describe noncovalent molecular interactions, the secondary structure, the solvent-accessible surface (SAS), the buriedness of residues and waters, volume, rigidity, and flexibility.

The three machine learning algorithms, Logistic Regression, Random Forest, and Gradient Boosting, were employed to classify the structures as either 'deep-sea' or 'decoy'. Feature selection with wrapper methods is applied to evaluate the performance of different feature sets with the CV, like in section 2.1. The best-performing models and feature sets are evaluated on the external test sets as an additional validation step. Since the exponential number of feature combinations makes the enumeration of all feature sets in the feature selection infeasible, only feature combinations of up to five features are evaluated. For the 45 structural features these are  $\sum_{k=1}^{s=5} \binom{45}{k} = 1,385,979$  feature sets.

To interpret which features are important and potentially related to protein adaptations, the feature selection with wrapper methods procedure from section 2.1.4 was extended to quantify the importance of individual features with the enhancement described in the following section 2.2.4.

### 2.2.4 Methodological Summary

The method developed in [D2] extends the interpretability method from section 2.1.4 with the Shapley value framework [169] from cooperative game theory for estimating the importance of individual features. Initially, Shapley values were developed to quantify the contributions of individual players in a cooperative game. The Shapley value of an individual player is the average of the player's marginal contributions over all possible permutations the coalition can be formed [169]. Thus, a player's contribution is described as the average change in the coalition's value when the player participates. This concept can be used to attribute contributions single features make in combination with other features in a machine learning prediction task. The set of all features is denoted as  $N$ , and the set  $S \mid S \subseteq N$  is a certain combination of features. The Shapley value of a feature  $f_i \mid f_i \in N$  is the weighted sum of the marginal contributions feature  $f_i$  makes when included in the feature set  $S$ :

$$Sh_{f_i}(v) = \sum_{S \subseteq N \setminus \{f_i\}} \frac{|S|!(n - |S| - 1)!}{n!} (v(S \cup \{f_i\}) - v(S)),$$

where  $n = |N|$  is the total number of features, and the function  $v$  is a function mapping a feature set to a real number representing its value. The Shapley values concept can be directly applied to the feature selection methodology with wrapper methods described in section 2.1.4 to quantify the contributions of individual features. In this scenario, the value of collaboration described through  $v$  is the feature set's mean performance in the CV experiment.

The exponential scaling of feature sets makes the computation of Shapley values infeasible when a larger number of features is used. For example, this issue can be avoided with sampling approaches through which Shapley values can be estimated in polynomial time [170]. In this project, analogously to the sampling approach, the contribution of each feature  $f_i$  was computed by considering only the marginal contributions from a sample of all possible coalitions, specifically all feature sets up to five features. Features with high contributions indicate that the feature is important for the classification task, which can be used to interpret protein adaptations.

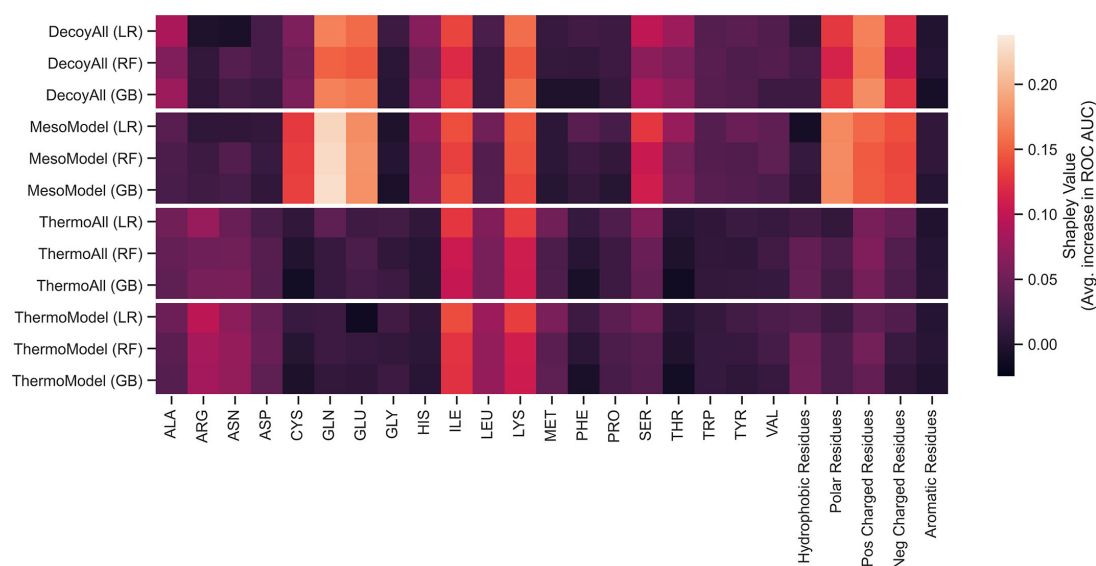
### 2.2.5 Analyzing Important Features For Protein Adaptations

The evaluation results of important features are illustrated in Figure 2.6 and 2.7. Features with high contributions indicate that these features are important for predicting deep-sea proteins across different protein clusters or families.

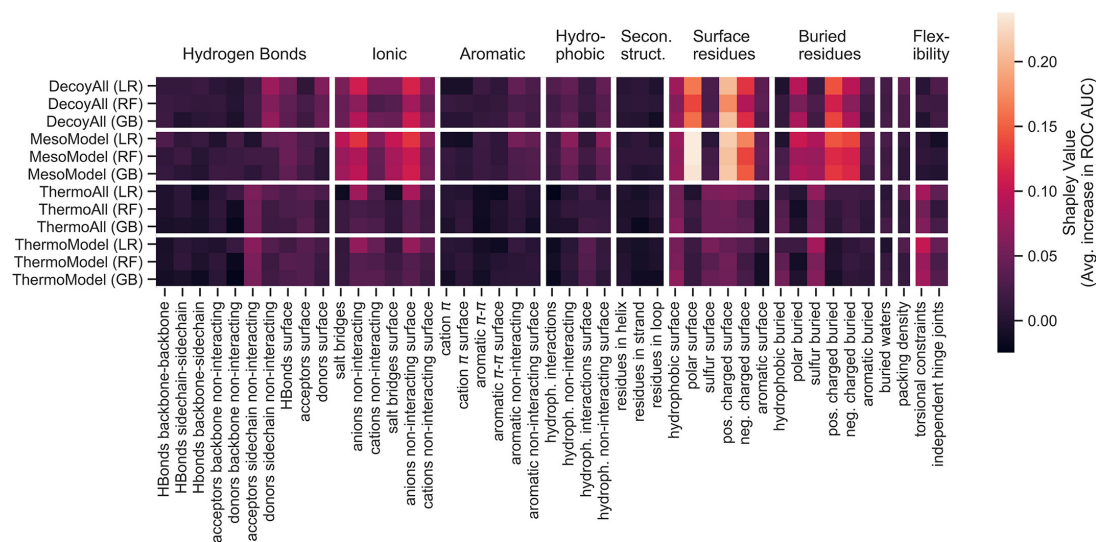
The results in Figure 2.6 and 2.7 show that the features considered important by the three machine learning algorithms within each data set are relatively similar. Therefore, the selection of the machine learning method does not seem critical, and even the simpler Logistic Regression algorithm captures signals similar to the non-linear tree ensemble methods.

Viewing the results in the context of the compared proteins' source organisms' environments is essential to interpret protein adaptations to high pressure. Specifically, strong contributing features can be observed in the experiments of the 'DecoyAll' and 'MesoModel' data sets, namely the relative frequencies of GLN, GLU, ILE, LYS, SER, positively and negatively charged residues, polar residues, and CYS for the 'MesoModel' data set for the sequence and the portion of the polar surface, positively and negatively charged surface, the buried polar residues, buried positively and negatively charged residues and the number of noninteracting anions of the whole protein and on its surface. The distribution of charged and polar residues are the most contributing features in both sequence and structure features. Intriguingly, these features correspond well to the characteristics usually described as thermal adaptations: an increased proportion of charged and a reduced number of polar residues [68, 82, 166, 171]. This exact

## 2.2 Analyzing Structure Data Sets for Protein Adaptations



**Figure 2.6:** Heatmap of the contributions of each sequence feature overall data sets and machine learning algorithms. The color illustrates the average increase in prediction performance in terms of mean ROC AUC in the cluster CV if the feature is included. On the x-axis, the names of the features are listed. The y-axis depicts the data sets and the machine learning algorithm used. The algorithms are Logistic Regression (LR), Random Forest (RF), and Gradient Boosting (GB). This figure was taken from [D2].



**Figure 2.7:** Heatmap of the contributions of each structure feature overall data sets and machine learning algorithms. The color illustrates the average increase in prediction performance in terms of mean ROC AUC in the cluster CV if the feature is included. On the x-axis, the names of the features are listed. The y-axis depicts the data sets and the machine learning algorithm used. The algorithms are Logistic Regression (LR), Random Forest (RF), and Gradient Boosting (GB). This figure was taken from [D2].

same trends were observed in the results of this doctorate project. Considering that the deep-sea protein data comes from (hyper)thermophilic organisms, it can not be ruled out that the features highlighted as important in the results describe thermal and not pressure or other adaptations. While adaptations to high temperature and high pressure might be related and dependent, this cannot be concluded based on the results of the 'DecoyAll' and 'MesoModel' experiments alone.

For disentangling adaptations to temperature from other adaptations in deep-sea proteins, the 'ThermoAll' and 'ThermoModel' data sets were employed to compare deep-sea proteins to proteins of thermophiles. Features important in all four data sets might describe distinct properties of deep-sea proteins that can not be attributed to thermal adaptations. The results on the 'ThermoAll' and 'ThermoModel' data sets show notably less prominent trends than those on the first two. The most important sequence features are the proportion of LYS and ILE, which, interestingly, are, on average, increased in deep-sea proteins over all four data sets. This enrichment was described before by others [172, 173], but a mechanism for how this residue preference might adapt deep-sea proteins still needs to be clarified. On the structure level, the most distinctive contributions come from noninteracting anions of the whole protein and at the surface. Both are increased on average in deep-sea proteins. However, the contributions of the features on the 'ThermoAll' and 'ThermoModel' are not strongly pronounced, suggesting that the features ranked here as the most important features are likely not traits of the whole population of deep-sea proteins but only of a subset.

In conclusion, a clear pattern or mechanism for high-pressure adaptations could not be derived. However, the results of the systematic analysis provide clues on which features might be linked to an adaptive mechanism. The features highlighted on the 'DecoyAll' and 'MesoModel' data sets might illustrate the apparent differences already extensively described in the literature on thermal adaptations [68, 82, 166, 171]. When the 'ThermoAll' and 'ThermoModel' data sets are also considered, a small set of rather subtle and hard-to-interpret features emerges. Therefore, the results suggest that pressure adaptations might be only present in a subset of deep-sea proteins, which is in accordance with current beliefs [162, 174] but does not provide detailed new insights into a particular adaptive mechanism. Further analysis and results reaching the same conclusions are described in detail in [D2].

### 2.2.6 Outlook

The major limitation hindering the analysis of protein adaptations using comparative studies still seems to be the available data and its annotation. Even though there has

been recent progress in data collection, the experimental structure data available is scarce, mostly from (hyper)thermophiles and imbalanced towards individual organisms. In addition, given the complexity and entanglement of extreme adaptations, using the habitat as a surrogate for protein properties might be a bottleneck whose resolution requires a tremendous and community-wide effort.

The next steps to further investigate high-pressure adaptations in proteins could be analyzing individual protein families and organisms using the clues derived in this study. Certain protein classes are suspected to be more likely to hold adaptations, for example, enzymes involved in energy metabolism [162]. In addition, alternative data sources might be helpful, like considering protein sequences databases or predicted structures. However, whether these data sources are helpful must be investigated. While current metagenomic efforts seem to step-wise improve the knowledge of extremophiles, unraveling the molecular mechanism and adaptations within these organisms and their proteins poses an additional challenge.

## 2.3 Alternate Location Enumeration

In the third project of this doctorate [D3], a new method was developed for automatically handling alternate locations (AltLocs) in protein structures and structural complexes. AltLocs are experimentally derived structure annotations describing discrete conformations of regions of the structure, like single atoms, side chains, or larger parts. In this doctorate project, a new method was co-developed to efficiently and automatically enumerate AltLoc conformations for a given structure. The method was implemented in the AltLocEnumerator tool which provides the generated conformations to the user as a structure ensemble. AltLocEnumerator can automatically generate valid structure ensembles representing experimental evidenced protein flexibility through a simple and function-rich interface ready for various structure-based tasks.

### 2.3.1 Motivation

Proteins are usually represented as a single rigid structure that insufficiently expresses the inherent protein flexibility. However, flexibility is vital for many essential tasks, for example, docking or binding free energy estimation in drug discovery [175, 176]. Protein flexibility can be described through structure ensembles of the protein collected from a structure database [176], estimated with computational methods like molecular dynamics simulations [175] or derived from experimental indicators, like B-factors [132]. AltLocs encompass another source of structural conformations supported with



experimental evidence and are directly available from single structure files. However, despite 42% [D3] of structures in the PDB annotated with AltLocs, they are often overlooked. AltLoc annotations describe discrete conformations of parts of the structure, like different side chain conformations, derived by crystallographers from ambiguities in the experimental data, i.e., the electron density [126]. The negligence of AltLocs by molecular modelers and other practitioners might be due to a lack of accessible methods to handle AltLocs efficiently, automatically, and correctly. It is common practice to arbitrarily select only a single AltLoc conformation either by picking the first appearing conformation when parsing the structure file or selecting all AltLoc conformations labeled with a certain identifier, like 'A' [D3]. However, this leads to a single rigid structure when there is actually an exponential number of conformations of the full protein structure when all AltLoc conformations are combined. Some modeling tools allow manually selecting specific AltLocs, e.g., ChimeraX [177] or PyMOL [178], which can become tedious.

AltLocs can be crucial in almost all structure-based tasks. Even though structural changes encoded in AltLoc conformations are usually relatively small, even small conformational changes can significantly impact the structure and interactions of biomolecules. For example, side chain conformations can make a difference in small molecule docking. A specific AltLoc conformation can enable noncovalent interactions, like hydrogen bonds or polar interactions, while the alternative can not [D3]. Further, AltLoc conformations can help to express structural variations for developing and evaluating side chain prediction methods widely used in protein design, protein docking, or structure optimization [179].

At the start of this project, no method existed that enables practitioners to exploit the protein flexibility encoded in AltLoc annotations automatically without human intervention while checking for reasonable overall structure conformations and efficiently handling the exponential number of potential structure conformations.

### 2.3.2 AltLocs in the PDB

AltLocs describe alternative atom coordinates and are annotated to a structure by crystallographers when interpreting the electron density. The usage and annotation of AltLocs throughout the structures in the PDB are not fully consistent. Perhaps because an explicit guideline on when and how to annotate AltLocs is unavailable, leaving space for interpretation. This, however, complicates the automatic processing of AltLocs.

AltLocs are annotated in PDB structure files in the ATOM/HETATM section as additional entries labeled with alphabetic characters, often starting with 'A'. AltLocs



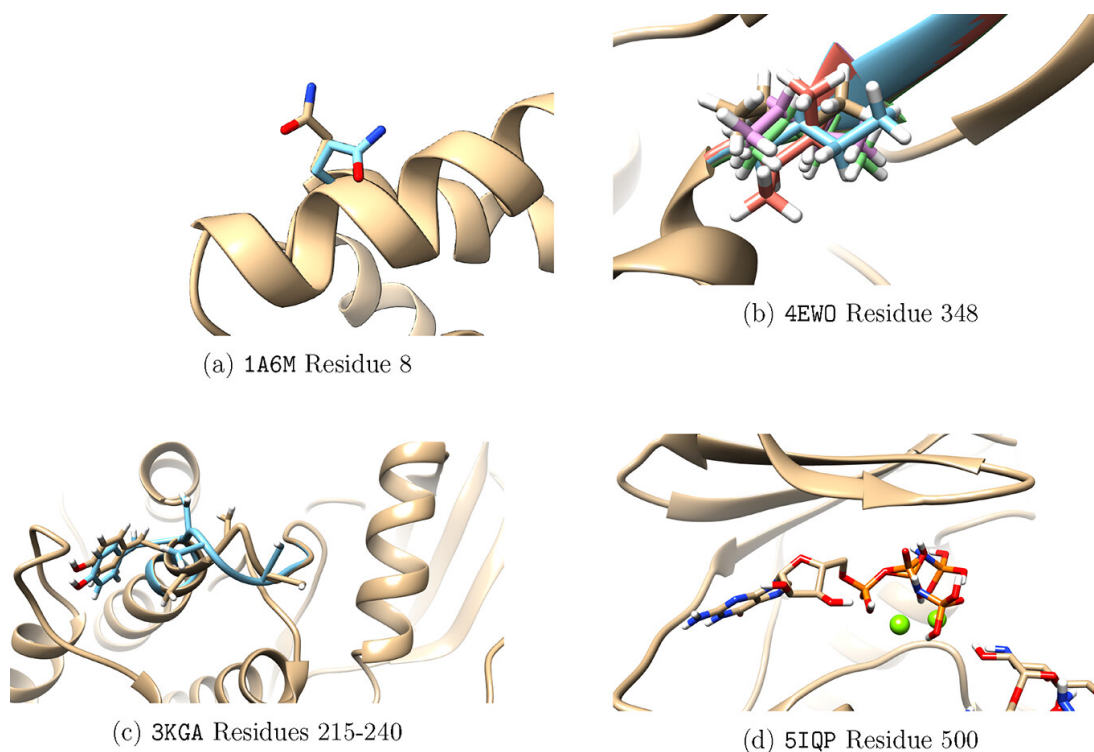
can be annotated for all entities of a structural complex in the PDB structure file, including proteins, nucleic acids, and other molecules. Each alternative atom coordinate is assigned an occupancy value between zero and one, reflecting the AltLoc conformation's frequency experimentally observed in the crystal [126, D3]. However, it can be challenging for crystallographers to determine the exact differences in occupancy, resulting in uniform values for all conformations. All occupancy values over the AltLoc conformations of an atom should sum up to one [126]. While AltLocs are atom-level annotations, they are usually used on a per-residue basis, for example, to describe alternative side chain conformations. In addition, while theoretically, an arbitrary number of conformations can be annotated, most AltLocs describe only two discrete states [D3, 179].

Interestingly, the number of AltLocs annotated in PDB structure files correlates positively with the resolution [D3, 179]. This implies that poorly resolved structures do not provide a sufficient experimental basis for determining AltLocs. In contrast, in high-resolution structures, AltLocs can be deduced more reliably [D3, 179]. Improving structure determination methods will likely make more high-resolution structures containing more AltLoc annotations available.

Figure 2.8 shows examples of AltLocs in the PDB. Figure 2.8a illustrates the simple case of two side chain conformations. In this case, only the side chain atoms have alternative coordinates annotated. However, AltLocs can describe more than two conformations and include both side chain and backbone atoms as depicted in Figure 2.8b. Further, as illustrated in Figure 2.8c the AltLoc conformations of multiple residues can depend on each other. Here, the AltLoc annotations describe two loop conformations. Finally, AltLoc conformations can also appear in HETATM entries, including small molecules, cofactors, and water molecules. For example, Figure 2.8d shows two AltLoc conformations in the GNP ligand in complex with an aminoglycoside kinase.

### 2.3.3 Handling AltLocs with AltLocEnumerator

AltLocEnumerator was implemented as a command line tool within the NAOMI ChemBio Suite [180–182]. The tool takes a structure in PDB/mmCIF format as input. It automatically generates a particular single structure conformation or a structure ensemble from the AltLoc annotations in the input file. AltLocs of structural complexes of proteins, nucleic acid, and other molecules are handled automatically, and the user can select different AltLoc enumeration strategies (see Table 2.3). The enumeration strategies represent common use cases, like obtaining only a single or all structure conformations with the best occupancy estimate ((i), (ii)). Also, all possible valid structure



**Figure 2.8:** Examples of AltLocs in the PDB. a) The AltLocs describe two side chain conformations, which is a frequent case. b) AltLocs of a residue with five different conformations differing in the side chain and backbone atoms. c) Two AltLocs of a peptide fragment containing parts of a helix and loop. The AltLocs of multiple sequential residues depend on each other. d) Two AltLoc conformations in a ligand molecule. This figure was taken from [D3].

conformations can be enumerated (iii) and exported as a structure ensemble. In addition, two simple and often employed strategies are provided, which select AltLoc conformations using a single AltLoc identifier ((iv), (v)) to generate a specific structure conformation.

**Table 2.3:** List of strategies to enumerate AltLoc conformations implemented in AltLocEnumerator. This table was taken from [D3].

	output conformations	name	description
(i)	single	best occupancy score	Generation of a single structure conformation with the maximal occupancy score.
(ii)	multiple	all best occupancy score	Generation of all structure conformations with the same maximum occupancy score.
(iii)	multiple	enumerate all	Enumeration of all possible valid structure conformations.
(iv)	single	first encounter	Selection of a structure conformation based on the first encountered AltLoc identifier while reading the file.
(v)	single	specific AltLoc-ID	Selection of a structure conformation with a user-specified AltLoc identifier. For example, all AltLoc conformations with identifier 'B'.

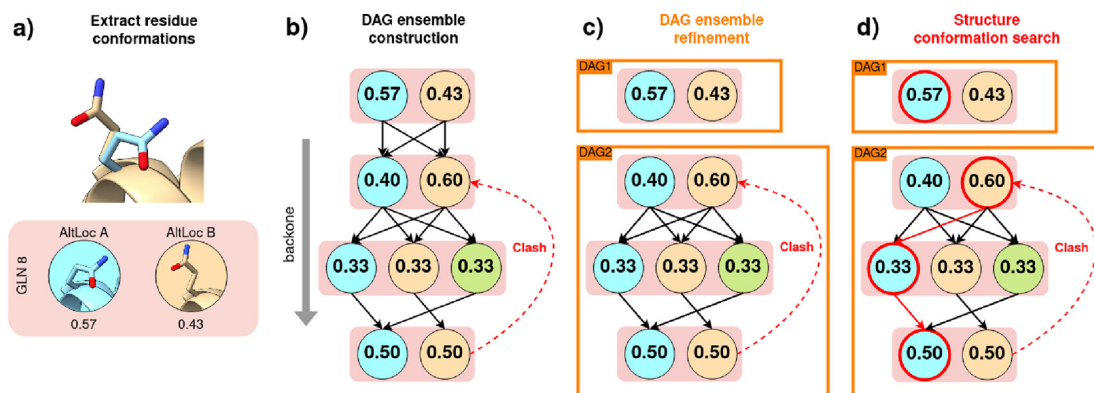
The AltLocEnumerator tool provides some additional options. The number of enumerated structure conformations can be limited for strategies (ii) and (iii). Limitation can be necessary when there is a combinatorial explosion of residue AltLoc conformations. In addition, local sites or a subset of residues can be specified to focus the AltLoc enumeration on a set of residues, for example, a binding site. This option can also help to reduce the problem of combinatorial explosions. Further options include stripping HETATMs before AltLoc handling, which allows building some structure conformations that would clash with HETATMs. Finally, a root mean square deviation (RMSD) filter can greedily select a diverse set of structure conformations by filtering out all structure conformations below a user-defined RMSD threshold to already accepted structure conformations.

### 2.3.3.1 Methodological Summary

The newly developed method behind AltLocEnumerator efficiently combines AltLoc conformations of individual residues with a branch-and-bound algorithm. Only valid structure conformations are generated by selecting the AltLoc combinations that do not introduce backbone clashes or chain breaks. The algorithm is illustrated in Figure 2.9. The general idea is to handle and represent AltLoc annotations as residue conformations by grouping alternate locations of atoms from the same residues together. These residue AltLoc conformations are extracted from the input structure file (see Figure 2.9a) and occupancy values are assigned to the residue AltLoc conformations as the mean of the occupancy values of the atoms with AltLocs in the residue.

A data structure is constructed from the residue AltLoc conformations, named AltLoc-DAG, which is an ensemble of directed acyclic graphs (DAGs) (see Figure 2.9b). The AltLoc-DAG represents a reduced search space of all possible combinations of residue AltLoc conformations containing only compatible residue conformations. Nodes represent residue conformations and are weighted by their occupancy. The DAG is organized hierarchically in layers, each holding the AltLoc conformations of a particular residue. An edge is added when two residues are successive in the macromolecular sequence, and no chain break would be introduced by selecting the two conformations. In addition, clashes between all residue conformations are tracked as indicated by the dashed red arrow in Figure 2.9b. Subsequently, the AltLoc-DAG is refined into separated independent DAGs (see Figure 2.9b and c).

Valid structure conformations can be determined from the AltLoc-DAG (see Figure 2.9d) by employing the first three search strategies ((i), (ii) and (iii)) listed in Table 2.3. A valid structure conformation can be found by traversing each DAG from the top to



**Figure 2.9:** Workflow of the method behind AltLocEnumerator for enumerating valid structure conformations from AltLoc annotations. a) depicts the extraction of residue conformations and their occupancy values from the structure file. b) illustrates the construction of the AltLoc-DAG data structure, an ensemble of DAGs. The AltLoc-DAG represents a reduced search space for the extraction of valid structure conformations. c) shows the refinement of the AltLoc-DAG and splitting independent residue AltLoc conformations in separate DAGs. d) demonstrates the search for a valid structure conformation with a traversal on the AltLoc-DAG. This figure was taken from [D3].

the bottom layer. Any path without a node pair with a clash represents a valid structure conformation. Strategies (i) and (ii) employ a greedy depth-first search-like traversal scoring the occupancy values of the residue conformations to find the valid structure conformation with the best occupancy score or all valid structure conformations with the best occupancy score, respectively. In contrast, strategy (iii) enumerates all paths in the AltLoc-DAG (ignoring occupancy but considering clashes), providing all possible valid structure conformations.

For further methodological details, see [D3].

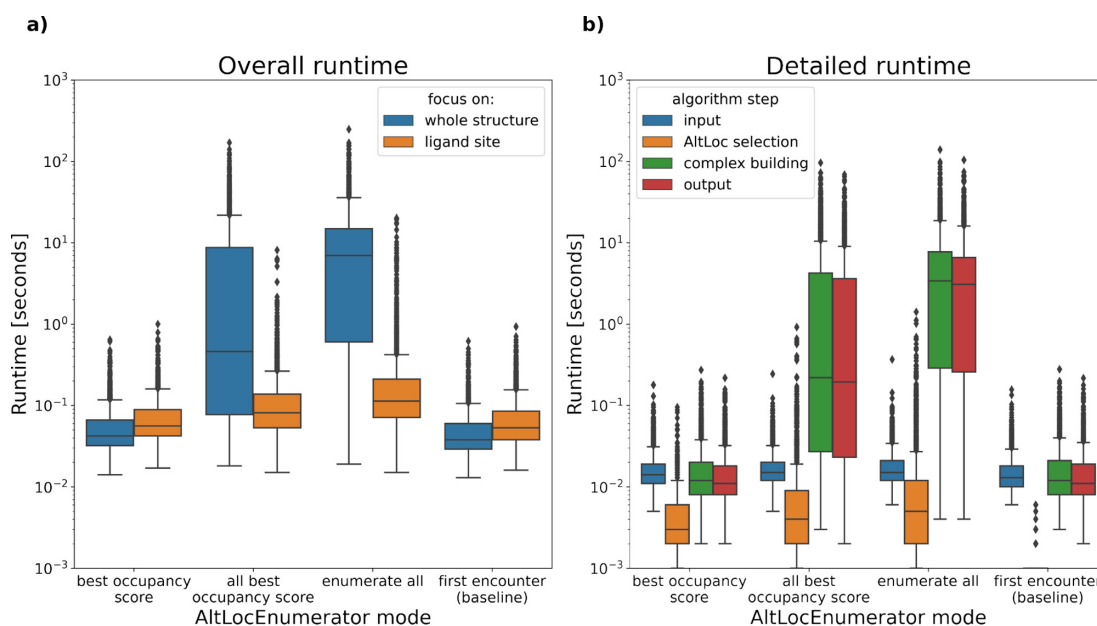
### 2.3.3.2 Evaluation

AltLocEnumerator was evaluated on the binding sites of the structures with AltLocs in the sc-PDB [183]. A naive approach was implemented with scripts that parses the binding site residues with AltLocs from the structure file and computes the number of structure conformations by taking the product of residue AltLoc conformations. The naive approach combines all residue conformations without considering dependencies between separate residue AltLoc conformations. The comparison showed that AltLocEnumerator extracts the same AltLoc annotated residues as the naive approach. However, the number of structure conformations differed in 15.56% of the cases. Of these cases, most differed because the maximum number of structure conformations to enumerate (513) was reached by AltLocEnumerator. The remaining 370 cases were checked manually and differed only due to expected or acceptable reasons. For example,

the conformations reported by the naive approach but not by AltLocEnumerator had backbone breaks or clashes with HETATMs. We found cases where AltLocs were used to model alternative amino acids instead of conformations. Also, the naive approach reports combinations with AltLocs representing only a single residue conformation, which AltLocEnumerator ignores.

The runtime of AltLocEnumerator was also evaluated on the sc-PDB subset with AltLocs in the binding sites. The results are depicted in Figure 2.10. The results show that using the 'best occupancy score' strategy with AltLocEnumerator does not take considerably more time than the much simpler baseline strategy of selecting AltLocs with the first encountered AltLoc identifier (Figure 2.10a). Consequently, with AltLocEnumerator the single structure conformation with the best occupancy score can be selected without significant compromise in runtime. Specifying a ligand binding site has an adverse effect on the runtime when only a single structure conformation should be generated. However, for generating structure ensembles the focus on the ligand site can speed up the computation probably because the overhead of processing the local site specification is not beneficial when only a single structure conformation is generated. In Figure 2.10b it can be seen that the core algorithm of AltLocEnumerator takes the least time while handling the input and building and validating generated structure conformations with NAOMI, as well as writing them to disc takes the most time. Particularly, in the strategies generating structure ensembles, the last two steps are taking the most time. These results show that the developed algorithm is efficient enough to replace the naive default strategy 'first encounter' with the new 'best occupancy score' strategy when a single structure conformation is required. In addition, structure ensembles of AltLoc conformations can be generated flexibly for the whole structure or a local site with AltLocEnumerator in just a few seconds.

It was also evaluated whether considering AltLocs in the redocking of protein-ligand complexes has a positive effect on the generated ligand poses. This experiment combined AltLocEnumerator and the docking tool JAMDA [184]. The analysis showed that in many cases, considering AltLocs yields ligand poses with better RMSD in the top ranks than the baseline structure conformation. However, this improvement could only be observed when more poses were considered for the AltLoc approach than for the baseline approach. When accounting for this numerical advantage by considering the same number of poses for both approaches, no significant change in performance compared to the baseline was observed. The analysis shows that better poses can be generated using AltLocs but cannot automatically be selected from the ranking.



**Figure 2.10:** Results of AltLocEnumerator runtime evaluation on the sc-PDB. a) shows the runtime of four different enumeration strategies implemented in AltLocEnumerator (strategies (i) to (iv)). The runtime is evaluated for the whole structure and the ligand site separately. b) detailed runtime for each step in the tool when using the whole structure. The 'input' time describes the time needed to process the input data. 'AltLoc selection' illustrates the time taken for the newly developed algorithm to construct the AltLoc-DAG and perform the search (except for the 'first encounter' strategy). 'complex building' describes the time needed by NAOMI to build and validate the complex data structure and 'output' is the time needed for writing the generated structure conformations as PDB files on the disc. This figure was taken from [D3].

### 2.3.4 Outlook

With AltLocEnumerator the protein flexibility described through AltLoc conformations can now be handled automatically for any structure-based downstream application. Since AltLocs have often been ignored in the past, applying AltLocEnumerator to investigate the influence of AltLoc conformations in diverse tasks in drug discovery and protein engineering would be an interesting next step. In addition, the tool can be integrated into existing workflows to incorporate AltLoc-based protein flexibility. A convenient starting point could be workflows where structure ensembles already handle protein flexibility, for example, in ensemble docking [185]. Further technical developments could include the integration of additional criteria to select conformations. For example, interactions, like hydrogen bonds could be scored or the electron density support of conformations could be considered using methods like the EDIA scorer [127]. Furthermore, more strategies could be tested that generate diverse conformation ensembles, for example, different clustering algorithms.



## 2.4 3D Micro-Environment Similarity Search

In the fourth doctorate project [D4], a new method for searching similar local 3D micro-environments in protein structure databases was developed and implemented in the tool MicroMiner. The method introduces a novel perspective for searching and compiling structure ensembles of local residue-centered structural protein sites, termed residue 3D micro-environments. In general, micro-environment similarity search constitutes a basis for exploring and analyzing the local details characterizing structure and function. This project focused on the scientific key application of structural single mutation analysis. With MicroMiner, millions of amino acid pairs illustrating the structural changes upon single mutations could be extracted from the PDB. Specifically, more than  $255 \cdot 10^6$  pairs for monomers and more than  $45 \cdot 10^6$  pairs for protein-protein interfaces were extracted. With these pairs, existing experimental data of mutation effects, like  $\Delta\Delta G$  measurements and affinity changes, are annotated with structures for the mutant. For this existing data, only a structure for the wild-type was usually available, insufficiently representing the structural change upon mutation. In addition, within this doctorate project, the MicroMiner tool was integrated into the ProteinsPlus web platform [D5]. MicroMiner can be combined on the server with established structure-based drug discovery tools, like binding site detection methods, to bridge the gap between mutation analysis and drug design. The MicroMiner method provides a new way to process existing protein structure collections to extract structure ensembles for a more comprehensive representation of structural changes in atomic details at local sites in protein structures.

### 2.4.1 Motivation and Idea

The idea of the MicroMiner method arose in this doctorate project while investigating the limitations and challenges in mutation effect prediction data sets [133, 186]. Accurately representing the structural changes induced by single mutations on the protein structure is essential for various applications like mutation effect prediction and modeling, structure prediction, and side chain modeling, which affect multiple downstream applications in protein engineering and drug discovery. When working with mutation effect measurements, like energy changes, a protein structure is usually only available for the wild-type and not for the mutant [133]. Current approaches [114, 187, 188] mitigate the problem through 'hypothetical reverse mutations' [189, 190] by modeling the mutant residue in the wild-type structure. However, a reliable representation, especially for more extensive structural changes, induced by a single mutation in atomic detail



remains vital [133, 186]. Intriguingly, even sophisticated structure prediction methods like AlphaFold2 are currently incapable of accurate mutation effect prediction for single mutations [191, 192]. Therefore, practitioners and method developers working in protein engineering and drug discovery need a reliable and directly available representation of the structural changes induced by a single mutation.

In this project [D4], a potential solution to this problem was developed with MicroMiner. Protein structure databases like the PDB were observed to contain many similar proteins, providing examples illustrating structural changes of single mutations at local positions in the protein structure. However, with established tools, extracting these structure pairs of single mutation sites is not straightforward. Dedicated tools that can extract single mutation sites from protein structure databases at scale are lacking. In this doctorate project, the MicroMiner method was developed from the observation that searching for such local protein sites of single mutations has requirements similar to searching for ligand binding sites in protein structure databases. Based on this observation, the MicroMiner method was developed based on the SIENA [97] and ASCONA [193] methods for binding site similarity search, which were available in the NAOMI framework [180–182]. Instead of searching for the 3D protein environments of small molecule ligands, i.e., the binding site, MicroMiner searches for the 3D protein environment of individual residues, called residue 3D micro-environments. The idea behind applying MicroMiner for mutation analysis is to use large quantities of existing protein structure data as a resource to extract local 3D micro-environments representing the structural changes of single mutations. With the structure data in the PDB, structural changes can be represented through experimentally determined structure pairs of the wild-type and the mutant. These depict the structural details in atomic resolution and can readily be used for downstream applications.

### 2.4.2 Approaches for Searching Locally Similar Protein Structures

Since protein similarity search is widely established, various prominent local similarity analysis methods exist. However, it was found in this doctorate project that a tool is lacking to search similar local 3D protein sites in protein structure databases with the requirements for single mutation analysis.

The most well-known protein similarity search methods are local sequence aligners like BLAST [86], MMseqs2 [92], or DIAMOND [94] which are fundamental bioinformatics tools. However, they focus on homology detection, which usually identifies the largest similar sequence part shared between two proteins or domains. An assessment of a particular local 3D site considering its' short sequence fragments, 3D contacts, and

multiple chains is not provided in these tools. In addition, purely sequence-based tools usually have no mechanism to handle experimental artifacts in 3D structures, like unresolved residues. On the other hand, prominent structure-based similarity search tools, like Dali [95, 194], TM-align [96, 168], or Foldseek [59] detect homology through conserved structure, consequently aiming to identify and align structurally similar protein regions, which can be contrary to handling protein flexibility and structural changes upon mutations. In addition, Dali and TM-align perform global protein alignments, and only Foldseek, which was released recently, aligns proteins locally. Therefore, currently, widely employed protein similarity search tools are not directly applicable to the problem of searching and aligning local 3D protein sites of single mutations.

In contrast, it was found in this doctorate project that the perspective of similarity search tools for ligand binding sites seems more suitable for searching similar local 3D protein sites of single mutations. Binding sites are usually modeled as local 3D sites in the protein structure where a ligand resides. Therefore, in this doctorate project, it was hypothesized that binding site similarity search methods build a reasonable basis to model the local 3D site of an amino acid for which a mutation should be found. Numerous tools for binding site similarity assessment exist [195], for example ASCONA [193], PocketShape [196], DeeplyThough [197] and ProCare [198]. Importantly, only some of these tools can be used in a reasonable time for database searches because they only perform pairwise comparisons and do not employ a fast prefiltering step of the search database first. Further, searching for single mutation sites requires an identical target sequence except for a single residue. However, structural deviations should be tolerated to identify structural changes upon mutation. Not all of the mentioned binding site similarity tools can directly consider amino acid identity but employ, for example, more abstract concepts like interaction/pharmacophore points or the protein surface. In addition, many binding site similarity tools are not directly suited to capture structural changes upon mutation because they employ strict geometric matching criteria. This doctorate project hypothesized that the ASCONA [193] and SIENA [97] binding site similarity search tools are likely suited for searching similar 3D sites of single mutations. ASCONA is a fast sequence and geometry-based binding site aligner focusing on aligning binding site conformations while allowing for some mutations. This approach is considered suitable since it can work with protein flexibility and single mutations while being relatively fast due to sequence-based comparisons and only a fuzzy geometric evaluation. SIENA is built on top of ASCONA and adds a fast prefilter for searching protein structure databases using k-mers. Finally, ASCONA and SIENA were a convenient choice since both were developed by Stefan Bietz during his doctorate in the

group of Matthias Rarey and implemented in the NAOMI code base, which was also used during this work.

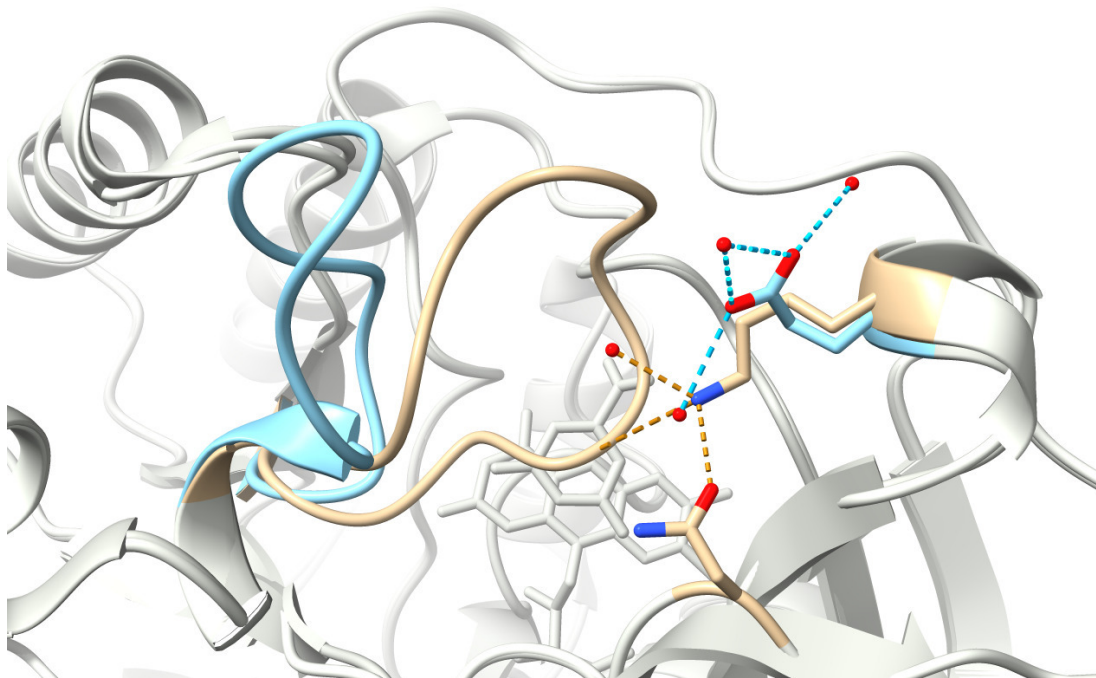
### 2.4.3 Searching Single Mutation Structure Pairs with MicroMiner

MicroMiner proposes a novel perspective on protein similarity search employing residue 3D micro-environment similarity search. The local neighborhood of a single reference residue in the protein structure defines a residue 3D micro-environment. Per default, it is described as all residues within a distance of 6.5 Å to any heavy atom of the reference residue. This micro-environment is used as a query to find protein structures in a database that contain similar local micro-environments. To achieve this, MicroMiner first performs a fast prefiltering of the protein structure database using k-mers and then uses the ASCONA site alignment algorithm to align the query micro-environment to the candidates to identify similar target micro-environments.

By default, MicroMiner simultaneously searches for all residues in the provided query protein structure. However, three preselection modes exist to select specific query micro-environments from the query structure. The 'full\_complex' mode is the default preselection mode and constructs the micro-environments as they are in the file in the asymmetric unit. The 'monomer' mode constructs the micro-environments by only considering residues from the same chain as the reference residue. In contrast, the 'ppi' mode only selects micro-environments with different chains in the environment to search for micro-environments at protein-protein interfaces.

Besides the preselection modes, MicroMiner implements two search modes. The 'standard' search mode implements the same strategy as SIENA, employing a threshold-controlled k-mer filter and the normal ASCONA site alignment. Since this doctorate project focused on single mutation analysis, a dedicated 'single\_mutation' search mode was developed, which exploits the characteristics of single mutations for algorithmic optimizations and faster searches but also reduces the enormous and impractical result sizes obtained with a less restrictive search algorithm. The 'single\_mutation' mode searches and reports only micro-environment hits representing single mutations in which a different amino acid replaces the query reference residue. However, the remaining residues of the micro-environment are identical.

The result of a search is a tabular hit list of hit micro-environments with multiple local and structural similarity measures of the micro-environments, like the root mean square deviation (RMSD) of the local site. The local similarity scores allow, for example, easy filtering and investigation of structural changes induced by single mutations. Optionally,



**Figure 2.11:** MicroMiner can identify and filter for structural changes upon a single mutation with a simple RMSD filter. The mutation Lys213 (2DOR, chain A) to Glu213 (1JQV, chain A) in the dihydroorotate dehydrogenase from *Lactococcus lactis* is illustrated. MicroMiner reports a local C $\alpha$ -RMSD of 4.21 Å for this micro-environment alignment. Norager et al. [199] reported the depicted mutation to be responsible for the open and closed form of the enzyme’s binding site. This figure was taken from [D4].

the hit micro-environments can automatically be superimposed on the query micro-environment to generate structure ensembles for direct visual inspection. Figure 2.11 shows an example single mutation MicroMiner hit for a dihydroorotate dehydrogenase. The hit micro-environment has an RMSD of 4.21 Å indicating considerable structural changes upon mutation.

The MicroMiner tool was integrated into the ProteinsPlus web application [D5] during this doctorate project. Only the ‘single\_mutation’ search mode is currently supported on the web server. This makes MicroMiner openly available at <https://proteins.plus> and part of an interoperable collection of structure-based modeling tools through which MicroMiner builds the connecting link between structural mutation analysis and standard modeling tools in drug discovery.

### 2.4.3.1 Methodological Summary

The workflow of MicroMiner is illustrated with a single query micro-environment in Figure 2.12. First, the query micro-environment is determined from a reference residue

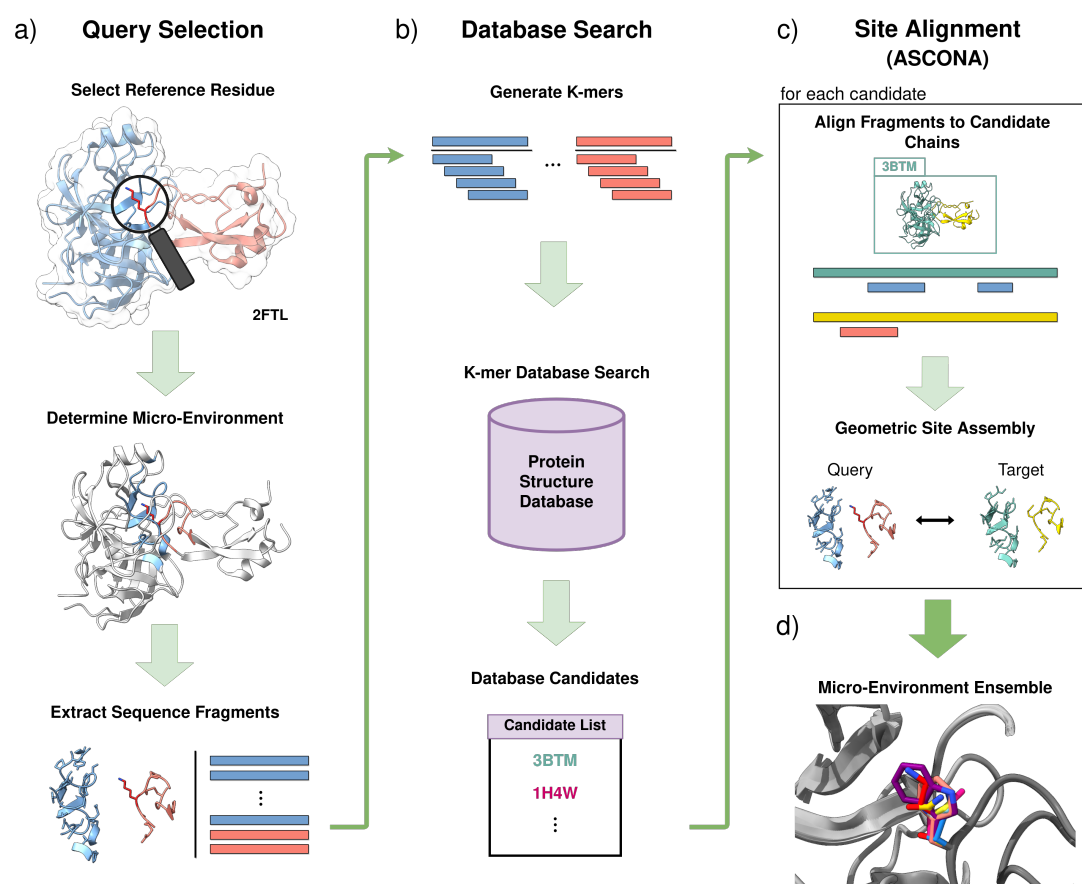
in the query protein structure (see Figure 2.12a). Then, connected sequence fragments are extracted from the query micro-environment for the subsequent database search. The database search uses the k-mers of the query micro-environment and matches these against the k-mers contained in structures from the database (see Figure 2.12b). Database structures fulfilling the k-mer prefiltering criteria are collected as candidates. All candidates from the database search are evaluated by site alignments utilizing the general algorithm of ASCONA [193] (see Figure 2.12c). The algorithm behind ASCONA aligns the query micro-environment sequence fragments to the candidate protein chains. Sufficiently scoring sequence-based matches are subsequently scored geometrically to evaluate whether they can be assembled to form a similar structural arrangement as the query site. Successful site alignments are reported as hits of similar micro-environments. Different similarity scores are calculated and provided for each hit, including the sequence identity of the local environments and the global chains, the RMSD of the C $\alpha$  atoms, and all atoms of the site residues.

Two algorithmic adaptations were developed for the 'single\_mutation' mode. The first adaptation is in the database search step. Instead of using the original k-mers of the query micro-environment, similar k-mers are generated in which the reference residue is substituted by one of the other 19 proteinogenic standard amino acids. This adaptation explicitly enables the search for a mutation at the reference residue position. The second adaptation is in the site alignment step of ASCONA. Instead of generating the sequence alignments with approximate string matching using dynamic programming, a seed and extend approach with linear string matching [200] was developed. In this seed and extend strategy, the reference residue position is again replaced by all 19 other proteinogenic standard amino acids to represent the mutation. Then, exact matches of the query sequence fragments with the candidate sequences are computed. These exact seed matches are then extended without gaps to a minimum size. The subsequent geometric scoring and site assembly are identical to the ASCONA algorithm.

For further methodological details, see [D4].

#### 2.4.3.2 Comparison to SIENA/ASCONA

The similarity search of residue 3D micro-environment poses further challenges than binding site similarity searches. A major difference is the increased search space. Proteins have only a relatively small number of ligand-binding sites. Therefore, a protein with only a handful of relevant binding sites can have hundreds or even thousands of residues, resulting in orders of magnitude larger query and hit sites. In addition, picking



**Figure 2.12:** Illustration of the MicroMiner method workflow on the example of a single query residue 3D micro-environment. a) shows the query selection process of selecting a reference residue, determining its 3D micro-environment, and extracting its sequence fragments. b) demonstrates the database search that generates a candidate list. c) illustrates the detailed evaluation of each candidate with the ASCONA site alignment algorithm. d) shows the resulting structure ensemble of the hit micro-environments with the query micro-environment. This figure was taken from [D4].

query binding sites requires an experimentally determined protein-ligand complex, binding site predictions, or other annotations. In contrast, residue 3D micro-environments in a query protein structure can be specified for all residues directly, which allows the analysis of residue micro-environments on a considerably larger scale than binding sites.

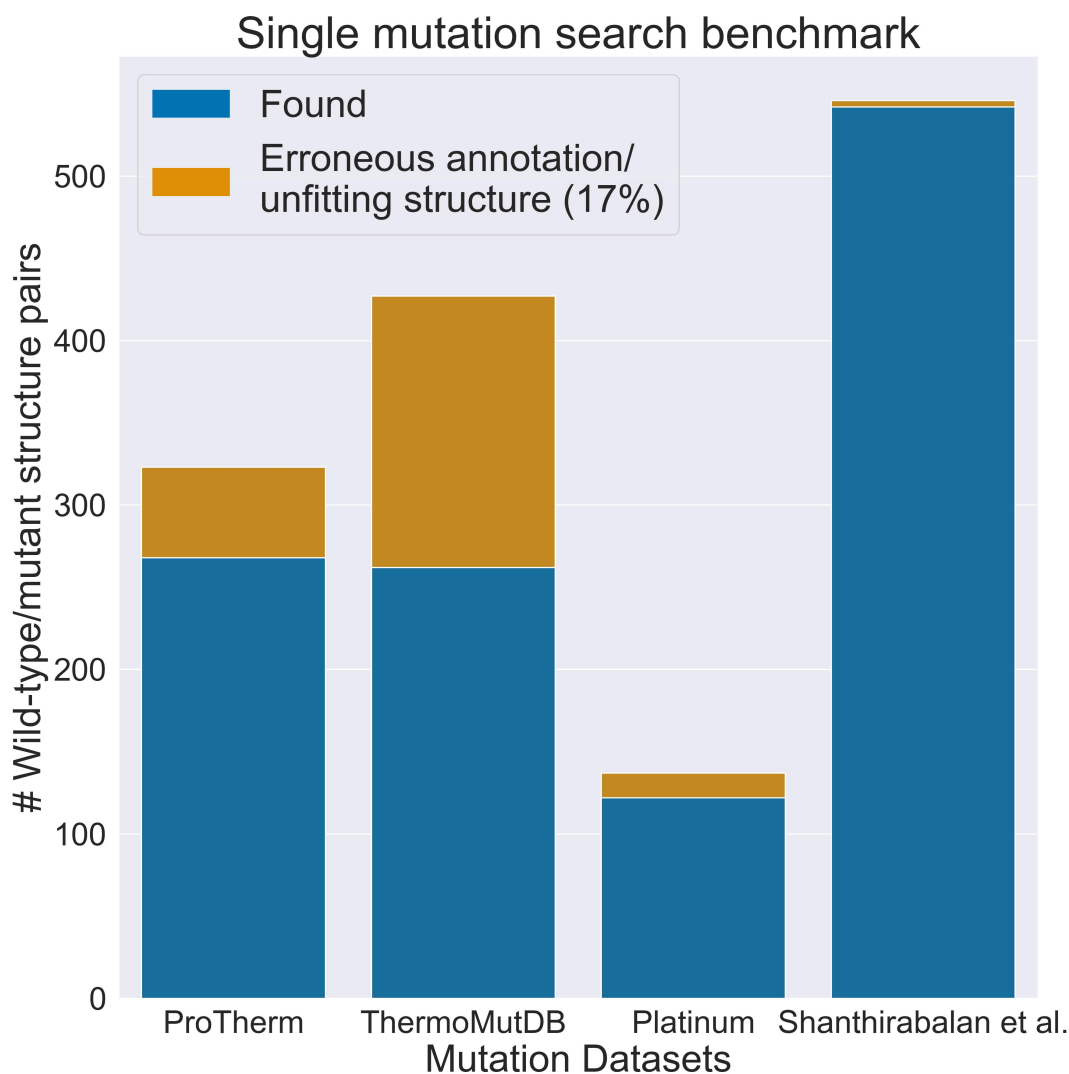
Multiple adaptations compared to the SIENA and ASCONA methodology were developed and implemented in MicroMiner and the NAOMI ChemBio Suite. Specifically, the approach of the search procedure was revised and redesigned to focus not on searching a single query site but on searching all query sites in a query structure at once. This redesign reduced redundant computations in the k-mer prefiltering step and the overall number of k-mer lookups. For comparison, SIENA employs an SQLite database for the k-mer prefiltering, performing separate database queries for each query site, even

if query sites have shared k-mers. Further, in MicroMiner, a faster in-memory k-mer search index was implemented, inspired by the MMseqs2 [92] tool. This search index employs k-mer indexing as a hash function [92]. Also, it stores the sequence start position of each k-mer, allowing additional filtering strategies to reduce false positives. In addition, in binding site similarity search, the number of candidate structures from the prefiltering is small enough to load and process them all at once in the main memory. The result sizes for the residue micro-environment search in MicroMiner are larger than for SIENA, and the candidates must be processed sequentially with ASCONA to stay within the limits of the main memory. Further, the search workflow allows for parallel execution, a feature not implemented in SIENA. In MicroMiner, the independent evaluations of candidate structures with ASCONA can be run in parallel. Furthermore, in MicroMiner, the search database creation was parallelized, reducing the construction time. Finally, the focus on single mutation analysis allowed for algorithmic optimizations in the k-mer and site alignment step (as described in section 2.4.3.1), which helped to reduce the runtime complexity and discard false positive hits earlier.

### 2.4.3.3 Evaluation

Since MicroMiner’s single mutation search is the first of its category, a direct comparison to other methods is not possible. Instead, a single mutation search benchmark was constructed to validate the single mutation search. Known protein structure pairs of wild-type and mutant with a specified position of the single mutation were extracted from four mutation datasets. It was evaluated whether MicroMiner can retrieve the annotated mutant structure from a PDB database search given the wild-type structure as a query. Retrievals were considered successful if the micro-environment alignment aligned the query reference residue to the correct mutated residue in the expected target structure. The evaluation results in Figure 2.13 show that at first, MicroMiner could only retrieve an average of 83% of the mutant structures. Surprisingly, manual investigation of the failed cases showed that many wild-type/mutant structure pairs from the datasets were erroneously annotated. For example, they had multiple mutations in their direct neighborhood despite being labeled a single mutation. After removing these cases, MicroMiner successfully retrieved 100%. This manual investigation demonstrated the necessity of consistently analyzing structure pairs representing single mutation. Current structure annotations in common mutation datasets seem not checked for such issues. With MicroMiner, this is now possible automatically.





**Figure 2.13:** Shows the performance of MicroMiner in retrieving known wild-type/mutant structure pairs from four mutation datasets. The cases labeled 'Found' (83%) were successfully retrieved, while 'Erroneous annotation/unfitting structure' (17%) denotes the cases not retrieved by MicroMiner because of incorrect or unfitting annotations. For example, no mutation was present in the structure, or additional mutations were detected in the micro-environment. This figure was taken from [D4].

#### 2.4.3.4 Applications

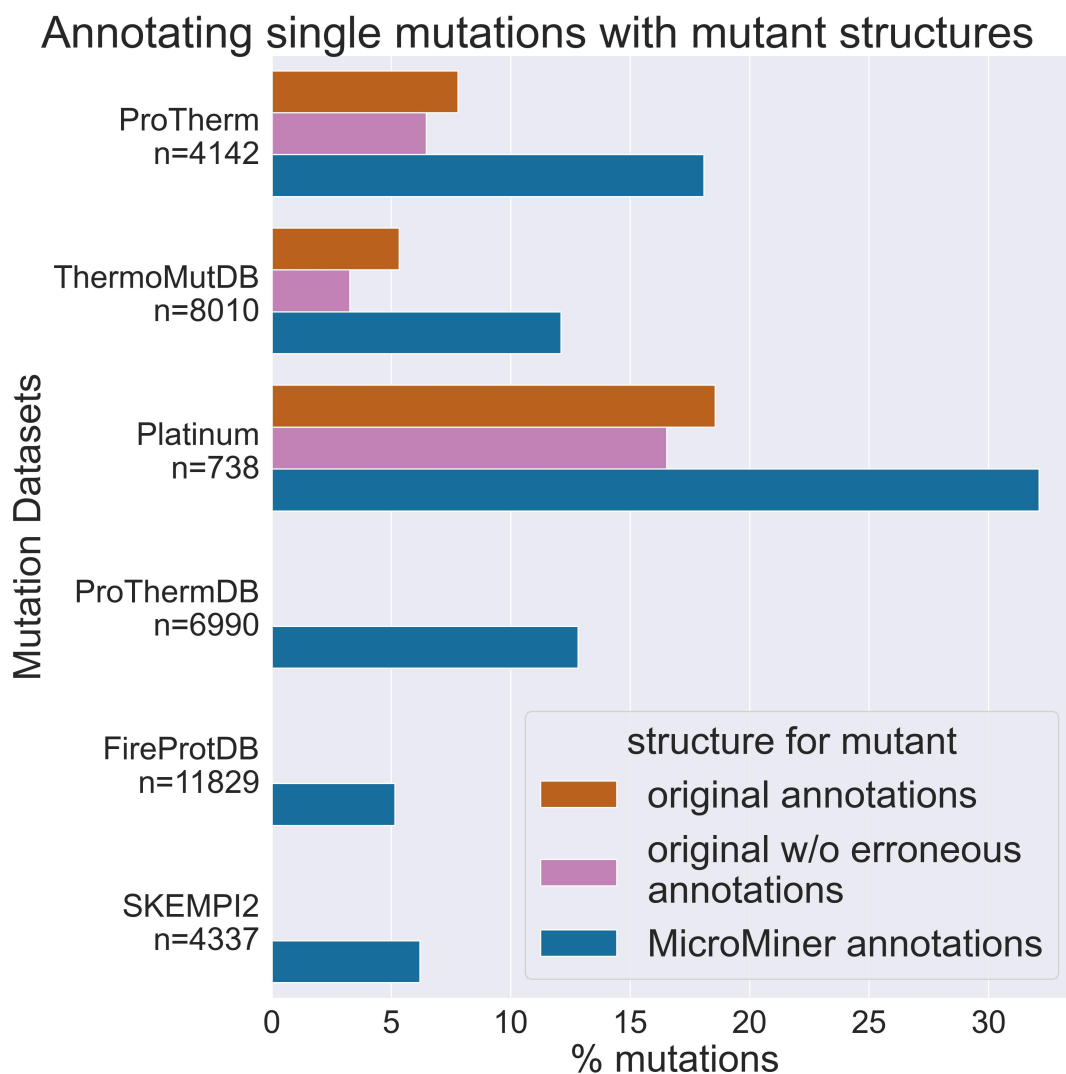
MicroMiner was applied in an all-vs-all experiment on the PDB to extract the structure pairs of all similar micro-environments exemplifying single mutations. More than  $255 \cdot 10^6$  amino acids pairs could be extracted using the 'monomer' mode and more than  $45 \cdot 10^6$  pairs for protein-protein interfaces with the 'ppi' mode. After filtering and redundancy removal, 4 868 764 amino acid pairs for single chains and 799 129 for



protein-protein interfaces remain. These two data sets represent a previously unavailable wealth of data describing the structural changes of single mutations that can now be used in numerous downstream applications.

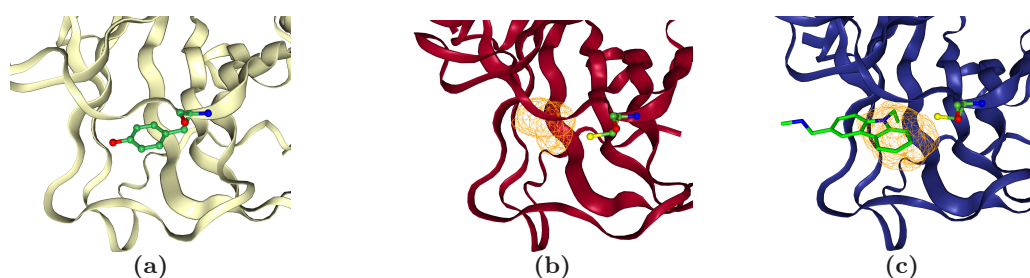
One downstream application is annotating existing data sets of mutation effect measures with protein structures for the mutant, i.e., combining the mutation effect measurement with the structural change upon mutation. MicroMiner considerably improves the experimental structure annotation coverage of the six mutation effect data sets ProTherm [201], ThermoMutDB [202], Platinum [77], ProThermDB [203], FireProtDB [204] and SKEMPI2 [205] (see Figure 2.14). Only three of the six data sets originally had mutant structure annotations. There were 414 unique wild-type/mutant structure pairs over all data sets. Using MicroMiner, this number could be increased 6.4-fold to 2653 structure pairs. Still, many mutations are without a structure for the mutant. However, this is unsurprising since there is probably no experimentally determined protein structure in the PDB for every mutation. Some mutations also do not result in folded proteins or proteins unsolvable experimentally. Regardless, the number of annotated structures could be improved considerably. This experiment also demonstrates how the annotation process of mutation effect data sets can be automated to combine mutation effect data with the structural changes upon mutation in atomic resolution based on experimental data from the PDB.

Another application of MicroMiner is to bridge the gap between mutation analysis and structure-based drug discovery. During this doctorate project, the MicroMiner tool was integrated into the ProteinsPlus web platform [D5]. MicroMiner can be combined on the server with established structure-based modeling tools for drug discovery. For example, users can combine MicroMiner with the DoGSite3 [206] tool to detect binding sites emerging upon mutations directly on the server. The following use case demonstrates this retrospectively [207] on the tumor antigen p53 (p53). Many pathological mutations in p53 lead to thermolabile variants that can not reliably fulfill the tumor suppressor role anymore [207]. However, small molecule stabilizers can target these cancer mutations, making such p53 variants druggable for personalized therapy [208]. A prominent example is the Y220C mutation estimated to be responsible for an annual 100 000 cancer cases [209]. Figure 2.15 demonstrates how MicroMiner can be used in combination with DoGSite3 [206] on the ProteinsPlus server to detect the ligand binding site emerging upon the pathological Y220C mutation originally described by Joerger et al. [207]. The first step to screen p53 for emerging binding sites would be to get a list of disease-related mutations, for example, from UniProt [210] or the TP53 Database [211]. Next, the wild-type p53 structure can be used as a query to



**Figure 2.14:** Improvement of the mutant structure coverage of six mutation effect data sets using MicroMiner. The x-axis shows the percentage of mutations with a structure for the wild-type. The y-axis illustrates six mutation effect data sets. The bars display the percentage of wild-type/mutant structure pairs. Bars labeled with 'original annotations' are structure pairs annotated by the data set curators. The bars with 'original w/o erroneous annotations' labels show the same information but with the erroneous annotations removed. The 'MicroMiner annotations' bars give the percentage of mutations MicroMiner annotated with a mutant structure from the PDB. This figure was taken from [D4].

MicroMiner's single mutation search, and all structures exemplifying single mutations at the positions of disease-related mutations can be investigated based on experimental structures returned from the PDB. Binding pockets can be predicted with DoGSite3 for the wild-type structure and structures retrieved with MicroMiner. In Figure 2.15a



**Figure 2.15:** Illustrates the combination of MicroMiner with DogSite3 on the ProteinsPlus webserver for the detection of druggable ligand binding sites upon disease-related mutations in the tumor suppressor p53. a) shows the p53 wild-type structure (1TUP, chain A) focused on Tyr220. DoGSite3 predicts no pocket. b) depicts the Y220C mutation (6SHZ, chain A). A pocket is predicted by DoGSite3 close to Cys220, where the larger tyrosine side chain has resided in the wild-type. c) also depicts the Y220C mutation (2VUK, chain B) but with ligand P83 bound in the emerged pocket. DoGSite3 also predicts a pocket in this structure. This figure was taken from [D4].

the wild-type p53 at the position of Tyr220 is depicted. At this site, DoGSite3 predicts no pocket. However, a pocket is predicted for the Y220C mutant (see Figure 2.15b). The predicted pocket's location is in accordance with Joerger et al. [207], who described their pocket at the location previously occupied by the larger tyrosine side chain. Finally, Figure 2.15c depicts the Y220C mutation with a bound ligand stabilizing the p53 mutant obtained through virtual screening and rational drug design [208]. This use case illustrates how structure-based drug discovery workflows can benefit from mutant structures provided by MicroMiner and how druggable binding pockets emerging upon mutation could systematically be analyzed.

#### 2.4.4 Outlook

Given the abundance of protein structures, the detailed analysis of large structure data sets will become increasingly important for various applications. With MicroMiner, an unprecedented quantity of structural ensembles of single mutations was extracted in this doctorate project. This data constitutes a reliable representation of the structural change induced by single mutations. The extracted data can readily be used in multiple downstream analyses and new data-driven approaches for mutation effect prediction, mutation modeling, protein structure prediction and more.

Modeling practitioners benefit from MicroMiner and its ProteinsPlus integration because it enables the interactive exploration of the structural landscape of the single mutations of a protein of interest. In this way, MicroMiner can help in rational protein engineering and drug design. For example, modelers could investigate mutations

of functionally important residues, mutations leading to thermo-stabilization or ligand binding sites emerging upon mutation.

The MicroMiner method could be extended and applied to problems in functional site analysis, analyzing motifs, co-evolved 3D contacts, and protein flexibility. One of the most exciting potential research applications is to use residue 3D micro-environment search for knowledge-based mutation modeling. Single mutation micro-environments returned by MicroMiner could be used as a starting template to model and refine single residue substitutions in a given protein structure. Compared to existing methods, a structural micro-environment template could provide a solid basis for modeling mutations inducing larger structural changes in the target protein structure. Another exciting research idea is to use similar micro-environments of identical sequences to refine predicted protein structures on a local and residue-wise basis. The residue-centered perspective of residue 3D micro-environments is similar to the perspective of the pLDDT confidence score of AlphaFold2 [28]. An interesting research direction would be to use a 3D micro-environment similarity search to search for site conformations (micro-environments with identical sequences) in the PDB to provide experimental structures to refine structural details in predicted protein structures based on the local 3D environment of each residue. A final research direction could be to use fast and highly optimized local sequence aligners like MMseqs2 [92] as a prefilter for MicroMiner. Such an approach is necessary when a certain protein similarity and homology are required, and the data set to be analyzed is enormous, like the AlphaFold Protein Structure Database [34] or the ESM Metagenomic Atlas [35].

The next methodological advancement to be incorporated into the MicroMiner tool could be an efficient algorithm to screen for multiple mutations in micro-environments. A promising and systematic approach could be the integration of substitution matrix-based scoring, 3D constraints in the prefilter step, and a comprehensive parameter evaluation and tuning for the ASCONA algorithm. Furthermore, there is potential to improve the search speed of the MicroMiner implementation. For example, candidate structures are currently read as PDB files from the disc as implemented in SIENA. Therefore, much of the overall runtime involves parsing candidate PDB text files, interpreting them, and translating them into the final data structures. Preprocessing the structure files into a serialized format, which can be transferred directly into the relevant data structure, would significantly improve the search time. Moreover, implementations of many modern bioinformatics tools [92, 94] use single instruction, multiple

data (SIMD) parallel processing. SIMD implementations of standard bioinformatics algorithms are openly available [212] for modern architectures and could potentially be integrated into NAOMI and MicroMiner.



## Chapter 3

# Summary

This dissertation presented methods and analysis studies for improving data-driven structure-property relationship modeling. The analyses delivered in-depth insights stressing current boundaries in the research fields. Detailed domain-specific descriptions of data limitations and potential new research directions were obtained using the newly developed interpretability method. The analyses provided significant new insights and raised concerns about the suitability of unbiasing strategies of standard benchmark data sets used in structure-based virtual screening for developing machine learning methods. In addition, a comprehensive and detailed description of the complex picture of protein features correlating with high-pressure environments was derived, pointing to potential future research directions for deciphering protein adaptations.

In the further course of this doctorate, software solutions for large-scale data processing were developed. The evaluations showed good results and various applications opened up during their development, indicating a promising positive impact on the research field. With the new AltLocEnumerator and MicroMiner tools, extensive data sets of structure ensembles can be compiled, which provide a more complete and comprehensive representation of proteins for property prediction and modeling. On the one hand, AltLocEnumerator offers the possibility of using long-ignored AltLoc conformations from structure files simply, consistently, and efficiently. Practical program options help to handle conformation enumeration and focus the display of protein flexibility, for example, on relevant protein sites, such as ligand binding sites. On the other hand, the method behind the MicroMiner tool represents a new and promising perspective for local residue-centered 3D query search in protein structure databases, which can be applied to various possible scientific questions. In this thesis, the focus was on the key application of structural analysis of single mutations. The MicroMiner tool can search similar single mutation sites and filter for structural changes. Using MicroMiner, more

than two hundred million amino acid pairs in protein structures exemplifying structural changes of single mutations could be extracted from the PDB and made accessible in a data set. Simultaneously, experimental measures of mutation effects could be combined with the structural change upon mutation. MicroMiner makes an unprecedented amount of structural single mutation data connected to experimental mutation effect measures available for developing data-driven methods. Furthermore, MicroMiner provides a direct way to combine mutation analysis with structure-based tools, opening up many molecular modeling applications, as demonstrated with the use case of identifying binding sites emerging upon pathological mutation in the tumor suppressor p53.

Overall, achieving effective computational modeling and *in silico* predictions of protein properties is a grand scientific challenge. Its pursuit requires tremendous efforts and solving various subproblems. This thesis addressed problems in the central area of data-driven structure-property relationship modeling. Software solutions were proposed to understand and overcome domain-specific challenges like data set bias, complex protein properties and single rigid protein representations. The analysis and software tools created during this doctorate project are hopefully incremental parts that collectively help solve the grand challenge of property prediction.



# References

- [1] V. Timofeev and V. Samygina. “Protein Crystallography: Achievements and Challenges”. In: *Crystals* 13.1 (2023), p. 71.
- [2] D. M. Fowler and S. Fields. “Deep mutational scanning: a new style of protein science”. In: *Nature methods* 11.8 (2014), pp. 801–807.
- [3] L. A. De Jong, D. R. Uges, J. P. Franke, and R. Bischoff. “Receptor–ligand binding assays: technologies and applications”. In: *Journal of Chromatography B* 829.1-2 (2005), pp. 1–25.
- [4] V. Kairys, L. Baranauskiene, M. Kazlauskiene, D. Matulis, and E. Kazlauskas. “Binding affinity in drug design: experimental and computational techniques”. In: *Expert opinion on drug discovery* 14.8 (2019), pp. 755–768.
- [5] Z. Gaieb, S. Liu, S. Gathiaka, M. Chiu, H. Yang, C. Shao, V. A. Feher, W. P. Walters, B. Kuhn, M. G. Rudolph, et al. “D3R Grand Challenge 2: blind prediction of protein–ligand poses, affinity rankings, and relative binding free energies”. In: *Journal of computer-aided molecular design* 32 (2018), pp. 1–20.
- [6] S. Ackloo, R. Al-Awar, R. E. Amaro, C. H. Arrowsmith, H. Azevedo, R. A. Batey, Y. Bengio, U. A. Betz, C. G. Bologna, J. D. Chodera, et al. “CACHE (Critical Assessment of Computational Hit-finding Experiments): A public–private partnership benchmarking initiative to enable the development of computational methods for hit-finding”. In: *Nature Reviews Chemistry* 6.4 (2022), pp. 287–295.
- [7] A. Kryshtafovych, T. Schwede, M. Topf, K. Fidelis, and J. Moult. “Critical assessment of methods of protein structure prediction (CASP)—Round XIV”. In: *Proteins: Structure, Function, and Bioinformatics* 89.12 (2021), pp. 1607–1617.
- [8] G. Sliwoski, S. Kothiwale, J. Meiler, and E. W. Lowe. “Computational methods in drug discovery”. In: *Pharmacological reviews* 66.1 (2014), pp. 334–395.

- [9] A. V. Sadybekov and V. Katritch. “Computational approaches streamlining drug discovery”. In: *Nature* 616.7958 (2023), pp. 673–685.
- [10] T. S. Wong, D. Zhurina, and U. Schwaneberg. “The diversity challenge in directed protein evolution”. In: *Combinatorial chemistry & high throughput screening* 9.4 (2006), pp. 271–288.
- [11] A. A. Kermani. “A guide to membrane protein X-ray crystallography”. In: *The FEBS journal* 288.20 (2021), pp. 5788–5804.
- [12] I. Samish, C. M. MacDermaid, J. M. Perez-Aguilar, and J. G. Saven. “Theoretical and computational protein design”. In: *Annual review of physical chemistry* 62 (2011), pp. 129–149.
- [13] J. L. Binder, J. Berendzen, A. O. Stevens, Y. He, J. Wang, N. V. Dokholyan, and T. I. Oprea. “AlphaFold illuminates half of the dark human proteins”. In: *Current Opinion in Structural Biology* 74 (2022), p. 102372.
- [14] T. Hegedűs, M. Geisler, G. L. Lukács, and B. Farkas. “Ins and outs of AlphaFold2 transmembrane protein structure predictions”. In: *Cellular and Molecular Life Sciences* 79.1 (2022), p. 73.
- [15] M. Varadi, N. Bordin, C. Orengo, and S. Velankar. “The opportunities and challenges posed by the new generation of deep learning-based protein structure predictors”. In: *Current Opinion in Structural Biology* 79 (2023), p. 102543.
- [16] R. A. Studer, B. H. Dessailly, and C. A. Orengo. “Residue mutations and their impact on protein structure and function: detecting beneficial and pathogenic changes”. In: *Biochemical journal* 449.3 (2013), pp. 581–594.
- [17] I. V. Korendovych and W. F. DeGrado. “De novo protein design, a retrospective”. In: *Quarterly reviews of biophysics* 53 (2020), e3.
- [18] A. G. Murzin, S. E. Brenner, T. Hubbard, and C. Chothia. “SCOP: a structural classification of proteins database for the investigation of sequences and structures”. In: *Journal of molecular biology* 247.4 (1995), pp. 536–540.
- [19] C. A. Orengo, A. D. Michie, S. Jones, D. T. Jones, M. B. Swindells, and J. M. Thornton. “CATH—a hierarchic classification of protein domain structures”. In: *Structure* 5.8 (1997), pp. 1093–1109.
- [20] X.-Y. Meng, H.-X. Zhang, M. Mezei, and M. Cui. “Molecular docking: a powerful approach for structure-based drug discovery”. In: *Current computer-aided drug design* 7.2 (2011), pp. 146–157.

- 
- [21] I. A. Vakser. “Protein-protein docking: From interaction to interactome”. In: *Biophysical journal* 107.8 (2014), pp. 1785–1793.
- [22] I. D. Kuntz. “Structure-based strategies for drug design and discovery”. In: *Science* 257.5073 (1992), pp. 1078–1082.
- [23] E. Capriotti, P. Fariselli, and R. Casadio. “I-Mutant2.0: predicting stability changes upon mutation from the protein sequence or structure”. In: *Nucleic acids research* 33.suppl\_2 (2005), W306–W310.
- [24] A. Volkamer, D. Kuhn, T. Grombacher, F. Rippmann, and M. Rarey. “Combining global and local measures for structure-based druggability predictions”. In: *Journal of chemical information and modeling* 52.2 (2012), pp. 360–372.
- [25] V. Gligorijević, P. D. Renfrew, T. Kosciolk, J. K. Leman, D. Berenberg, T. Vatanen, C. Chandler, B. C. Taylor, I. M. Fisk, H. Vlamakis, et al. “Structure-based protein function prediction using graph convolutional networks”. In: *Nature communications* 12.1 (2021), p. 3168.
- [26] J. Jiménez-Luna, F. Grisoni, N. Weskamp, and G. Schneider. “Artificial intelligence in drug discovery: recent advances and future perspectives”. In: *Expert opinion on drug discovery* 16.9 (2021), pp. 949–959.
- [27] H. Narayanan, F. Dingfelder, A. Butté, N. Lorenzen, M. Sokolov, and P. Arosio. “Machine learning for biologics: opportunities for protein engineering, developability, and formulation”. In: *Trends in pharmacological sciences* 42.3 (2021), pp. 151–165.
- [28] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Židek, A. Potapenko, et al. “Highly accurate protein structure prediction with AlphaFold”. In: *Nature* 596.7873 (2021), pp. 583–589.
- [29] M. Baek, F. DiMaio, I. Anishchenko, J. Dauparas, S. Ovchinnikov, G. R. Lee, J. Wang, Q. Cong, L. N. Kinch, R. D. Schaeffer, et al. “Accurate prediction of protein structures and interactions using a three-track neural network”. In: *Science* 373.6557 (2021), pp. 871–876.
- [30] M. Ragoza, J. Hochuli, E. Idrobo, J. Sunseri, and D. R. Koes. “Protein–ligand scoring with convolutional neural networks”. In: *Journal of chemical information and modeling* 57.4 (2017), pp. 942–957.

- [31] J. S. Smith, B. T. Nebgen, R. Zubatyuk, N. Lubbers, C. Devereux, K. Barros, S. Tretiak, O. Isayev, and A. E. Roitberg. “Approaching coupled cluster accuracy with a general-purpose neural network potential through transfer learning”. In: *Nature communications* 10.1 (2019), p. 2903.
- [32] K. Yang, K. Swanson, W. Jin, C. Coley, P. Eiden, H. Gao, A. Guzman-Perez, T. Hopper, B. Kelley, M. Mathea, et al. “Analyzing learned molecular representations for property prediction”. In: *Journal of chemical information and modeling* 59.8 (2019), pp. 3370–3388.
- [33] wwPDB consortium. “Protein Data Bank: the single global archive for 3D macromolecular structure data”. In: *Nucleic acids research* 47.D1 (2019), pp. D520–D528.
- [34] M. Varadi, S. Anyango, M. Deshpande, S. Nair, C. Natassia, G. Yordanova, D. Yuan, O. Stroe, G. Wood, A. Laydon, et al. “AlphaFold Protein Structure Database: massively expanding the structural coverage of protein-sequence space with high-accuracy models”. In: *Nucleic acids research* 50.D1 (2022), pp. D439–D444.
- [35] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, et al. “Evolutionary-scale prediction of atomic-level protein structure with a language model”. In: *Science* 379.6637 (2023), pp. 1123–1130.
- [36] Y. LeCun, Y. Bengio, and G. Hinton. “Deep learning”. In: *nature* 521.7553 (2015), pp. 436–444.
- [37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).
- [38] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. “An image is worth 16x16 words: Transformers for image recognition at scale”. In: *arXiv preprint arXiv:2010.11929* (2020).
- [39] J. M. Thornton, R. A. Laskowski, and N. Borkakoti. “AlphaFold heralds a data-driven revolution in biology and medicine”. In: *Nature Medicine* 27.10 (2021), pp. 1666–1669.

- 
- [40] J. B. Reece, L. A. Urry, M. L. Cain, W. S. A., P. V. Minorsky, and R. B. Jackson. *Campbell Biologie 2015*. Ed. by J. J. Heinisch and A. Paululat. 10. Hallbergmoos, Germany: Pearson Deutschland GmbH, 2015, p. 104. ISBN: 3868942599.
- [41] G. Klebe. “Protein-Ligand-Wechselwirkungen als Grundlage der Arzneistoffwirkung”. In: *Wirkstoffdesign*. 2. Heidelberg: Spektrum Akademischer Verlag, 2009. Chap. 4, pp. 49–67. ISBN: 9783827420466.
- [42] G. Klebe. “Biopharmaka: Peptide, Proteine, Nucleotide und Makrolide als Wirkstoffe”. In: *Wirkstoffdesign*. 2. Heidelberg: Spektrum Akademischer Verlag, 2009. Chap. 32, pp. 581–605. ISBN: 9783827420466.
- [43] M. Hay, D. W. Thomas, J. L. Craighead, C. Economides, and J. Rosenthal. “Clinical development success rates for investigational drugs”. In: *Nature biotechnology* 32.1 (2014), pp. 40–51.
- [44] H. Kaplon, A. Chenoweth, S. Crescioli, and J. M. Reichert. “Antibodies to watch in 2022”. In: *MAbs*. Vol. 14. 1. Taylor & Francis. 2022, p. 2014296.
- [45] P. K. Robinson. “Enzymes: principles and biotechnological applications”. In: *Essays in biochemistry* 59 (2015), p. 1.
- [46] C. Li, R. Zhang, J. Wang, L. M. Wilson, and Y. Yan. “Protein engineering for improving and diversifying natural product biosynthesis”. In: *Trends in biotechnology* 38.7 (2020), pp. 729–744.
- [47] P. Cozzini, G. E. Kellogg, F. Spyralis, D. J. Abraham, G. Costantino, A. Emerson, F. Fanelli, H. Gohlke, L. A. Kuhn, G. M. Morris, et al. “Target flexibility: an emerging consideration in drug discovery and design”. In: *Journal of medicinal chemistry* 51.20 (2008), pp. 6237–6255.
- [48] L. Pauling, R. B. Corey, and H. R. Branson. “The structure of proteins: two hydrogen-bonded helical configurations of the polypeptide chain”. In: *Proceedings of the National Academy of Sciences* 37.4 (1951), pp. 205–211.
- [49] S. C. Bagley and R. B. Altman. “Characterizing the microenvironment surrounding protein sites”. In: *Protein Science* 4.4 (1995), pp. 622–635.
- [50] A. C. Martin, C. A. Orengo, E. G. Hutchinson, S. Jones, M. Karmirantzou, R. A. Laskowski, J. B. Mitchell, C. Taroni, and J. M. Thornton. “Protein folds and functions”. In: *Structure* 6.7 (1998), pp. 875–884.
- [51] A. S. Pillai, G. K. Hochberg, and J. W. Thornton. “Simple mechanisms for the evolution of protein complexity”. In: *Protein Science* 31.11 (2022), e4449.

- [52] J. C. Kendrew, G. Bodo, H. M. Dintzis, R. Parrish, H. Wyckoff, and D. C. Phillips. “A three-dimensional model of the myoglobin molecule obtained by x-ray analysis”. In: *Nature* 181.4610 (1958), pp. 662–666.
- [53] W. Kühlbrandt. “The resolution revolution”. In: *Science* 343.6178 (2014), pp. 1443–1444.
- [54] T. Nakane, A. Kotecha, A. Sente, G. McMullan, S. Masiulis, P. M. Brown, I. T. Grigoras, L. Malinauskaite, T. Malinauskas, J. Miehl, et al. “Single-particle cryo-EM at atomic resolution”. In: *Nature* 587.7832 (2020), pp. 152–156.
- [55] K. M. Yip, N. Fischer, E. Paknia, A. Chari, and H. Stark. “Atomic-resolution protein structure determination by cryo-EM”. In: *Nature* 587.7832 (2020), pp. 157–161.
- [56] K. He, X. Zhang, S. Ren, and J. Sun. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.
- [57] A. Krizhevsky, I. Sutskever, and G. E. Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems* 25 (2012).
- [58] J. Pereira, A. J. Simpkin, M. D. Hartmann, D. J. Rigden, R. M. Keegan, and A. N. Lupas. “High-accuracy protein structure prediction in CASP14”. In: *Proteins: Structure, Function, and Bioinformatics* 89.12 (2021), pp. 1687–1699.
- [59] M. van Kempen, S. S. Kim, C. Tumescheit, M. Mirdita, J. Lee, C. L. Gilchrist, J. Söding, and M. Steinegger. “Fast and accurate protein structure search with Foldseek”. In: *Nature Biotechnology* (2023), pp. 1–4.
- [60] P. Radivojac, W. T. Clark, T. R. Oron, A. M. Schnoes, T. Wittkop, A. Sokolov, K. Graim, C. Funk, K. Verspoor, A. Ben-Hur, et al. “A large-scale evaluation of computational protein function prediction”. In: *Nature methods* 10.3 (2013), pp. 221–227.
- [61] C. Ehrt, T. Brinkjost, and O. Koch. “Impact of binding site comparisons on medicinal chemistry and rational molecular design”. In: *Journal of medicinal chemistry* 59.9 (2016), pp. 4121–4151.
- [62] V. S. Rao, K. Srinivas, G. Sujini, and G. Kumar. “Protein-protein interaction detection: methods and analysis”. In: *International journal of proteomics* 2014 (2014).

- 
- [63] X. Chen, C. C. Yan, X. Zhang, X. Zhang, F. Dai, J. Yin, and Y. Zhang. “Drug–target interaction prediction: databases, web servers and computational models”. In: *Briefings in bioinformatics* 17.4 (2016), pp. 696–712.
- [64] D. L. Mobley and M. K. Gilson. “Predicting binding free energies: frontiers and benchmarks”. In: *Annual review of biophysics* 46 (2017), pp. 531–558.
- [65] D. J. Jacobs, A. J. Rader, L. A. Kuhn, and M. F. Thorpe. “Protein flexibility predictions using graph theory”. In: *Proteins: Structure, Function, and Bioinformatics* 44.2 (2001), pp. 150–165.
- [66] S. A. Hollingsworth and R. O. Dror. “Molecular dynamics simulation for all”. In: *Neuron* 99.6 (2018), pp. 1129–1143.
- [67] J. H. M. van Gils, D. Gogishvili, J. van Eck, R. Bouwmeester, E. van Dijk, and S. Abeln. “How sticky are our proteins? Quantifying hydrophobicity of the human proteome”. In: *Bioinformatics advances* 2.1 (2022), vbac002.
- [68] F. Pucci and M. Rooman. “Physical and molecular bases of protein thermal stability and cold adaptation”. In: *Current opinion in structural biology* 42 (2017), pp. 117–128.
- [69] S. Iqbal, F. Li, T. Akutsu, D. B. Ascher, G. I. Webb, and J. Song. “Assessing the performance of computational predictors for estimating protein stability changes upon missense mutations”. In: *Briefings in Bioinformatics* 22.6 (2021), bbab184.
- [70] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, et al. “Gene ontology: tool for the unification of biology”. In: *Nature genetics* 25.1 (2000), pp. 25–29.
- [71] O. Emanuelsson, S. Brunak, G. Von Heijne, and H. Nielsen. “Locating proteins in the cell using TargetP, SignalP and related tools”. In: *Nature protocols* 2.4 (2007), pp. 953–971.
- [72] M. Batool, B. Ahmad, and S. Choi. “A structure-based drug discovery paradigm”. In: *International journal of molecular sciences* 20.11 (2019), p. 2783.
- [73] Y. Xu, D. Verma, R. P. Sheridan, A. Liaw, J. Ma, N. M. Marshall, J. McIntosh, E. C. Sherer, V. Svetnik, and J. M. Johnston. “Deep dive into machine learning models for protein engineering”. In: *Journal of chemical information and modeling* 60.6 (2020), pp. 2773–2790.

- [74] R. A. Norman, F. Ambrosetti, A. M. Bonvin, L. J. Colwell, S. Kelm, S. Kumar, and K. Krawczyk. “Computational approaches to therapeutic antibody design: established methods and emerging trends”. In: *Briefings in bioinformatics* 21.5 (2020), pp. 1549–1567.
- [75] W. A. Freed-Pastor and C. Prives. “Mutant p53: one name, many proteins”. In: *Genes & development* 26.12 (2012), pp. 1268–1286.
- [76] D. E. Pires, J. Chen, T. L. Blundell, and D. B. Ascher. “In silico functional dissection of saturation mutagenesis: Interpreting the relationship between phenotypes and changes in protein stability, interactions and activity”. In: *Scientific reports* 6.1 (2016), p. 19848.
- [77] D. E. Pires, T. L. Blundell, and D. B. Ascher. “Platinum: a database of experimentally measured effects of mutations on structurally defined protein–ligand complexes”. In: *Nucleic acids research* 43.D1 (2015), pp. D387–D391.
- [78] O. Hassin and M. Oren. “Drugging p53 in cancer: one protein, many targets”. In: *Nature Reviews Drug Discovery* 22.2 (2023), pp. 127–144.
- [79] X. Du, Y. Li, Y.-L. Xia, S.-M. Ai, J. Liang, P. Sang, X.-L. Ji, and S.-Q. Liu. “Insights into protein–ligand interactions: mechanisms, models, and methods”. In: *International journal of molecular sciences* 17.2 (2016), p. 144.
- [80] A. Goldenzweig and S. J. Fleishman. “Principles of protein stability and their application in computational design”. In: *Annual review of biochemistry* 87 (2018), pp. 105–129.
- [81] D. Stumpfe, Y. Hu, D. Dimova, and J. Bajorath. “Recent progress in understanding activity cliffs and their utility in medicinal chemistry: miniperspective”. In: *Journal of medicinal chemistry* 57.1 (2014), pp. 18–28.
- [82] S. Hait, S. Mallik, S. Basu, and S. Kundu. “Finding the generalized molecular principles of protein thermal stability”. In: *Proteins: Structure, Function, and Bioinformatics* 88.6 (2020), pp. 788–808.
- [83] M. Majewski, S. Ruiz-Carmona, and X. Barril. “An investigation of structural stability in protein–ligand complexes reveals the balance between order and disorder”. In: *Communications Chemistry* 2.1 (2019), p. 110.
- [84] L. Gerasimavicius, B. J. Livesey, and J. A. Marsh. “Loss-of-function, gain-of-function and dominant-negative mutations have profoundly different effects on protein structure”. In: *Nature communications* 13.1 (2022), p. 3895.



- 
- [85] A. Tropsha. “Best practices for QSAR model development, validation, and exploitation”. In: *Molecular informatics* 29.6-7 (2010), pp. 476–488.
- [86] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. “Basic local alignment search tool”. In: *Journal of molecular biology* 215.3 (1990), pp. 403–410.
- [87] W. R. Pearson and D. J. Lipman. “Improved tools for biological sequence comparison.” In: *Proceedings of the National Academy of Sciences* 85.8 (1988), pp. 2444–2448.
- [88] M. Bernhofer, C. Dallago, T. Karl, V. Satagopam, M. Heinzinger, M. Littmann, T. Olenyi, J. Qiu, K. Schütze, G. Yachdav, et al. “PredictProtein—predicting protein structure and function for 29 years”. In: *Nucleic acids research* 49.W1 (2021), W535–W540.
- [89] S. Wang, W. Li, S. Liu, and J. Xu. “RaptorX-Property: a web server for protein structure property prediction”. In: *Nucleic acids research* 44.W1 (2016), W430–W435.
- [90] W. R. Pearson. “An introduction to sequence similarity (“homology”) searching”. In: *Current protocols in bioinformatics* 42.1 (2013), pp. 3–1.
- [91] S. R. Eddy. “Accelerated profile HMM searches”. In: *PLoS computational biology* 7.10 (2011), e1002195.
- [92] M. Steinegger and J. Söding. “MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets”. In: *Nature biotechnology* 35.11 (2017), pp. 1026–1028.
- [93] M. Steinegger, M. Meier, M. Mirdita, H. Vöhringer, S. J. Haunsberger, and J. Söding. “HH-suite3 for fast remote homology detection and deep protein annotation”. In: *BMC bioinformatics* 20.1 (2019), pp. 1–15.
- [94] B. Buchfink, K. Reuter, and H.-G. Drost. “Sensitive protein alignments at tree-of-life scale using DIAMOND”. In: *Nature methods* 18.4 (2021), pp. 366–368.
- [95] L. Holm, S. Kääriäinen, P. Rosenström, and A. Schenkel. “Searching protein structure databases with DaliLite v.3”. In: *Bioinformatics* 24.23 (2008), pp. 2780–2781.
- [96] R. Dong, S. Pan, Z. Peng, Y. Zhang, and J. Yang. “mTM-align: a server for fast protein structure database search and multiple protein structure alignment”. In: *Nucleic acids research* 46.W1 (2018), W380–W386.

- [97] S. Bietz and M. Rarey. “SIENA: efficient compilation of selective protein binding site ensembles”. In: *Journal of Chemical Information and Modeling* 56.1 (2016), pp. 248–259.
- [98] C. M. Bishop. *Pattern recognition and machine learning*. Ed. by M. Jordan, J. Kleinberg, and S. Bernhard. 1. New York, NY: Springer, 2006, pp. 2–3. ISBN: 9780387310732.
- [99] D. E. Pires, D. B. Ascher, and T. L. Blundell. “mCSM: predicting the effects of mutations in proteins using graph-based signatures”. In: *Bioinformatics* 30.3 (2014), pp. 335–342.
- [100] F. Pucci, K. Bernaerts, F. Teheux, D. Gilis, and M. Rومان. “Symmetry principles in optimization problems: an application to protein stability prediction”. In: *IFAC-PapersOnLine* 48.1 (2015), pp. 458–463.
- [101] R. Krivák and D. Hoksza. “P2Rank: machine learning based tool for rapid and accurate prediction of ligand binding sites from protein structure”. In: *Journal of cheminformatics* 10 (2018), pp. 1–12.
- [102] J. Liu and R. Wang. “Classification of current scoring functions”. In: *Journal of chemical information and modeling* 55.3 (2015), pp. 475–482.
- [103] S. Bietz, S. Urbaczek, B. Schulz, and M. Rarey. “Protoss: a holistic approach to predict tautomers and protonation states in protein-ligand complexes”. In: *Journal of cheminformatics* 6 (2014), pp. 1–12.
- [104] W. Kabsch and C. Sander. “Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features”. In: *Biopolymers: Original Research on Biomolecules* 22.12 (1983), pp. 2577–2637.
- [105] Y. Bengio, A. Courville, and P. Vincent. “Representation learning: A review and new perspectives”. In: *IEEE transactions on pattern analysis and machine intelligence* 35.8 (2013), pp. 1798–1828.
- [106] L. Breiman. “Random forests”. In: *Machine learning* 45 (2001), pp. 5–32.
- [107] B. E. Boser, I. M. Guyon, and V. N. Vapnik. “A training algorithm for optimal margin classifiers”. In: *Proceedings of the fifth annual workshop on Computational learning theory*. 1992, pp. 144–152.
- [108] P. Gainza, F. Sverrisson, F. Monti, E. Rodola, D. Boscaini, M. Bronstein, and B. Correia. “Deciphering interaction fingerprints from protein molecular surfaces using geometric deep learning”. In: *Nature Methods* 17.2 (2020), pp. 184–192.

- 
- [109] J. Alvarez and B. Shoichet. “Preface”. In: 1. Boca Raton, USA: Taylor & Francis Group, LLC, 2005, pp. 1–470. ISBN: 9781420028775.
- [110] M. Rarey, B. Kramer, T. Lengauer, and G. Klebe. “A fast flexible docking method using an incremental construction algorithm”. In: *Journal of molecular biology* 261.3 (1996), pp. 470–489.
- [111] R. A. Friesner, J. L. Banks, R. B. Murphy, T. A. Halgren, J. J. Klicic, D. T. Mainz, M. P. Repasky, E. H. Knoll, M. Shelley, J. K. Perry, et al. “Glide: a new approach for rapid, accurate docking and scoring. 1. Method and assessment of docking accuracy”. In: *Journal of medicinal chemistry* 47.7 (2004), pp. 1739–1749.
- [112] J. Eberhardt, D. Santos-Martins, A. F. Tillack, and S. Forli. “AutoDock Vina 1.2.0: New docking methods, expanded force field, and python bindings”. In: *Journal of chemical information and modeling* 61.8 (2021), pp. 3891–3898.
- [113] G. Corso, H. Stärk, B. Jing, R. Barzilay, and T. Jaakkola. “Diffdock: Diffusion steps, twists, and turns for molecular docking”. In: *arXiv preprint arXiv:2210.01776* (2022).
- [114] C. H. Rodrigues, D. E. Pires, and D. B. Ascher. “DynaMut: predicting the impact of mutations on protein conformation, flexibility and stability”. In: *Nucleic acids research* 46.W1 (2018), W350–W355.
- [115] W. Torng and R. B. Altman. “3D deep convolutional neural networks for amino acid environment similarity analysis”. In: *BMC bioinformatics* 18 (2017), pp. 1–23.
- [116] N. Anand, R. Eguchi, I. I. Mathews, C. P. Perez, A. Derry, R. B. Altman, and P.-S. Huang. “Protein sequence design with a learned potential”. In: *Nature communications* 13.1 (2022), p. 746.
- [117] A. Bairoch and R. Apweiler. “The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000”. In: *Nucleic acids research* 28.1 (2000), pp. 45–48.
- [118] A. Bender, N. Schneider, M. Segler, W. Patrick Walters, O. Engkvist, and T. Rodrigues. “Evaluation guidelines for machine learning tools in the chemical sciences”. In: *Nature Reviews Chemistry* 6.6 (2022), pp. 428–442.
- [119] K. Shimizu, W. Cao, G. Saad, M. Shoji, and T. Terada. “Comparative analysis of membrane protein structure databases”. In: *Biochimica et Biophysica Acta (BBA)-Biomembranes* 1860.5 (2018), pp. 1077–1091.

- [120] S. Bittrich, Y. Rose, J. Segura, R. Lowe, J. D. Westbrook, J. M. Duarte, and S. K. Burley. “RCSB Protein Data Bank: improved annotation, search and visualization of membrane protein structures archived in the PDB”. In: *Bioinformatics* 38.5 (2022), pp. 1452–1454.
- [121] J. P. Hughes, S. Rees, S. B. Kalindjian, and K. L. Philpott. “Principles of early drug discovery”. In: *British journal of pharmacology* 162.6 (2011), pp. 1239–1249.
- [122] A. C. Good and T. I. Oprea. “Optimization of CAMD techniques 3. Virtual screening enrichment studies: a help or hindrance in tool selection?” In: *Journal of computer-aided molecular design* 22 (2008), pp. 169–178.
- [123] I. Wallach and A. Heifets. “Most ligand-based classification benchmarks reward memorization rather than generalization”. In: *Journal of chemical information and modeling* 58.5 (2018), pp. 916–932.
- [124] H. He and E. A. Garcia. “Learning from imbalanced data”. In: *IEEE Transactions on knowledge and data engineering* 21.9 (2009), pp. 1263–1284.
- [125] G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue, and G. Bing. “Learning from class-imbalanced data: Review of methods and applications”. In: *Expert systems with applications* 73 (2017), pp. 220–239.
- [126] A. M. Davis, S. J. Teague, and G. J. Kleywegt. “Application and limitations of X-ray crystallographic data in structure-based ligand and drug design”. In: *Angewandte Chemie International Edition* 42.24 (2003), pp. 2718–2736.
- [127] A. Meyder, E. Nittinger, G. Lange, R. Klein, and M. Rarey. “Estimating electron density support for individual atoms and molecular fragments in X-ray structures”. In: *Journal of chemical information and modeling* 57.10 (2017), pp. 2437–2447.
- [128] C. Kramer, T. Kalliokoski, P. Gedeck, and A. Vulpetti. “The experimental uncertainty of heterogeneous public  $K_i$  data”. In: *Journal of medicinal chemistry* 55.11 (2012), pp. 5165–5173.
- [129] L. Montanucci, P. L. Martelli, N. Ben-Tal, and P. Fariselli. “A natural upper bound to the accuracy of predicting protein stability changes upon mutations”. In: *Bioinformatics* 35.9 (2019), pp. 1513–1517.
- [130] D. S. Marks, L. J. Colwell, R. Sheridan, T. A. Hopf, A. Pagnani, R. Zecchina, and C. Sander. “Protein 3D structure computed from evolutionary sequence variation”. In: *PloS one* 6.12 (2011), e28766.

- 
- [131] K. Teilum, J. G. Olsen, and B. B. Kragelund. “Protein stability, flexibility and function”. In: *Biochimica et Biophysica Acta (BBA)-Proteins and Proteomics* 1814.8 (2011), pp. 969–976.
- [132] Z. Sun, Q. Liu, G. Qu, Y. Feng, and M. T. Reetz. “Utility of B-factors in protein science: interpreting rigidity, flexibility, and internal motion and engineering thermostability”. In: *Chemical reviews* 119.3 (2019), pp. 1626–1665.
- [133] J. Fang. “A critical review of five machine learning-based algorithms for predicting protein stability changes upon mutation”. In: *Briefings in bioinformatics* 21.4 (2020), pp. 1285–1292.
- [134] N. Burkart and M. F. Huber. “A survey on the explainability of supervised machine learning”. In: *Journal of Artificial Intelligence Research* 70 (2021), pp. 245–317.
- [135] Z. C. Lipton. “The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery.” In: *Queue* 16.3 (2018), pp. 31–57.
- [136] M. T. Ribeiro, S. Singh, and C. Guestrin. “" Why should i trust you?" Explaining the predictions of any classifier”. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, pp. 1135–1144.
- [137] S. M. Lundberg and S.-I. Lee. “A unified approach to interpreting model predictions”. In: *Advances in neural information processing systems* 30 (2017).
- [138] A. Goldstein, A. Kapelner, J. Bleich, and E. Pitkin. “Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation”. In: *journal of Computational and Graphical Statistics* 24.1 (2015), pp. 44–65.
- [139] **J. Sieg**. *Evaluation of Benchmark Datasets for Virtual Screening with Machine Learning*. Master’s thesis. 2017.
- [140] T. Scior, A. Bender, G. Tresadern, J. L. Medina-Franco, K. Martinez-Mayorga, T. Langer, K. Cuanalo-Contreras, and D. K. Agrafiotis. “Recognizing pitfalls in virtual screening: a critical review”. In: *Journal of chemical information and modeling* 52.4 (2012), pp. 867–881.
- [141] T. B. Kimber, Y. Chen, and A. Volkamer. “Deep learning in virtual screening: recent applications and developments”. In: *International journal of molecular sciences* 22.9 (2021), p. 4435.

- [142] N. Huang, B. K. Shoichet, and J. J. Irwin. “Benchmarking sets for molecular docking”. In: *Journal of medicinal chemistry* 49.23 (2006), pp. 6789–6801.
- [143] S. G. Rohrer and K. Baumann. “Maximum unbiased validation (MUV) data sets for virtual screening based on PubChem bioactivity data”. In: *Journal of chemical information and modeling* 49.2 (2009), pp. 169–184.
- [144] M. M. Mysinger, M. Carchia, J. J. Irwin, and B. K. Shoichet. “Directory of useful decoys, enhanced (DUD-E): better ligands and decoys for better benchmarking”. In: *Journal of medicinal chemistry* 55.14 (2012), pp. 6582–6594.
- [145] S. M. Vogel, M. R. Bauer, and F. M. Boeckler. “DEKOIS: Demanding Evaluation Kits for Objective in Silico Screening — A Versatile Tool for Benchmarking Docking Programs and Scoring Functions”. In: *Journal of chemical information and modeling* 51.10 (2011), pp. 2650–2665.
- [146] J. B. Dunbar Jr, R. D. Smith, C.-Y. Yang, P. M.-U. Ung, K. W. Lexa, N. A. Khazanov, J. A. Stuckey, S. Wang, and H. A. Carlson. “CSAR benchmark exercise of 2010: selection of the protein–ligand complexes”. In: *Journal of chemical information and modeling* 51.9 (2011), pp. 2036–2046.
- [147] Z. Liu, M. Su, L. Han, J. Liu, Q. Yang, Y. Li, and R. Wang. “Forging the basis for developing protein–ligand interaction scoring functions”. In: *Accounts of chemical research* 50.2 (2017), pp. 302–309.
- [148] P. J. Ballester and J. B. Mitchell. “A machine learning approach to predicting protein–ligand binding affinity with applications to molecular docking”. In: *Bioinformatics* 26.9 (2010), pp. 1169–1175.
- [149] M. Wójcikowski, P. J. Ballester, and P. Siedlecki. “Performance of machine-learning scoring functions in structure-based virtual screening”. In: *Scientific Reports* 7.1 (2017), p. 46710.
- [150] I. Wallach, M. Dzamba, and A. Heifets. “AtomNet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery”. In: *arXiv preprint arXiv:1510.02855* (2015).
- [151] J. C. Pereira, E. R. Caffarena, and C. N. Dos Santos. “Boosting docking-based virtual screening with deep learning”. In: *Journal of chemical information and modeling* 56.12 (2016), pp. 2495–2506.

- 
- [152] J. Jiménez, M. Skalic, G. Martinez-Rosell, and G. De Fabritiis. “K deep: protein–ligand absolute binding affinity prediction via 3d-convolutional neural networks”. In: *Journal of chemical information and modeling* 58.2 (2018), pp. 287–296.
- [153] M. L. Verdonk, V. Berdini, M. J. Hartshorn, W. T. Mooij, C. W. Murray, R. D. Taylor, and P. Watson. “Virtual screening using protein- ligand docking: avoiding artificial enrichment”. In: *Journal of chemical information and computer sciences* 44.3 (2004), pp. 793–806.
- [154] J. J. Irwin and B. K. Shoichet. “ZINC- a free database of commercially available compounds for virtual screening”. In: *Journal of chemical information and modeling* 45.1 (2005), pp. 177–182.
- [155] W. D. Ihlenfeldt, Y. Takahashi, H. Abe, and S.-i. Sasaki. “Computation and management of chemical properties in CACTVS: An extensible networked approach toward modularity and compatibility”. In: *Journal of chemical information and computer sciences* 34.1 (1994), pp. 109–116.
- [156] D. L. Wheeler, T. Barrett, D. A. Benson, S. H. Bryant, K. Canese, V. Chetvernin, D. M. Church, M. DiCuccio, R. Edgar, S. Federhen, et al. “Database resources of the national center for biotechnology information”. In: *Nucleic acids research* 35.suppl\_1 (2007), pp. D5–D12.
- [157] I. Guyon and A. Elisseeff. “An introduction to variable and feature selection”. In: *Journal of machine learning research* 3.Mar (2003), pp. 1157–1182.
- [158] L. Chen, A. Cruz, S. Ramsey, C. J. Dickson, J. S. Duca, V. Hornak, D. R. Koes, and T. Kurtzman. “Hidden bias in the DUD-E dataset leads to misleading performance of deep learning in structure-based virtual screening”. In: *PloS one* 14.8 (2019), e0220113.
- [159] V.-K. Tran-Nguyen, C. Jacquemard, and D. Rognan. “LIT-PCBA: an unbiased data set for machine learning and virtual screening”. In: *Journal of chemical information and modeling* 60.9 (2020), pp. 4263–4273.
- [160] V. Sundar and L. Colwell. “The effect of debiasing protein–ligand binding data on generalization”. In: *Journal of Chemical Information and Modeling* 60.1 (2019), pp. 56–62.
- [161] D. Madras, E. Creager, T. Pitassi, and R. Zemel. “Learning adversarially fair and transferable representations”. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 3384–3393.



- [162] N. Ando, B. Barquera, D. H. Bartlett, E. Boyd, A. A. Burnim, A. S. Byer, D. Colman, R. E. Gillilan, M. Gruebele, G. Makhatadze, et al. “The molecular basis for life in extreme environments”. In: *Annual review of biophysics* 50 (2021), pp. 343–372.
- [163] J. Z. Kaye and J. A. Baross. “Synchronous effects of temperature, hydrostatic pressure, and salinity on growth, phospholipid profiles, and protein patterns of four *Halomonas* species isolated from deep-sea hydrothermal-vent and sea surface environments”. In: *Applied and environmental microbiology* 70.10 (2004), pp. 6220–6229.
- [164] M. Salvador-Castell, P. Oger, and J. Peters. “High-pressure adaptation of extremophiles and biotechnological applications”. In: *Physiological and biotechnological aspects of extremophiles*. Elsevier, 2020, pp. 105–122.
- [165] L. M. Peoples, T. S. Kyaw, J. A. Ugalde, K. K. Mullane, R. A. Chastain, A. A. Yayanos, M. Kusube, B. A. Methé, and D. H. Bartlett. “Distinctive gene and protein characteristics of extremely piezophilic *Colwellia*”. In: *BMC genomics* 21 (2020), pp. 1–18.
- [166] C. J. Reed, H. Lewis, E. Trejo, V. Winston, and C. Evilia. “Protein adaptations in archaeal extremophiles”. In: *Archaea* 2013 (2013).
- [167] P. Rice, I. Longden, and A. Bleasby. “EMBOSS: the European molecular biology open software suite”. In: *Trends in genetics* 16.6 (2000), pp. 276–277.
- [168] Y. Zhang and J. Skolnick. “TM-align: a protein structure alignment algorithm based on the TM-score”. In: *Nucleic acids research* 33.7 (2005), pp. 2302–2309.
- [169] L. S. Shapley et al. “A value for n-person games”. In: (1953).
- [170] J. Castro, D. Gómez, and J. Tejada. “Polynomial calculation of the Shapley value based on sampling”. In: *Computers & Operations Research* 36.5 (2009), pp. 1726–1730.
- [171] K. Suhre and J.-M. Claverie. “Genomic correlates of hyperthermostability, an update”. In: *Journal of Biological Chemistry* 278.19 (2003), pp. 17198–17202.
- [172] M. Di Giulio. “The origin of the genetic code in the ocean abysses: new comparisons confirm old observations”. In: *Journal of theoretical biology* 333 (2013), pp. 109–116.
- [173] A. Nath and K. Subbiah. “Insights into the molecular basis of piezophilic adaptation: Extraction of piezophilic signatures”. In: *Journal of Theoretical Biology* 390 (2016), pp. 117–126.



- 
- [174] T. Ichiye. “Enzymes from piezophiles”. In: *Seminars in cell & developmental biology*. Vol. 84. Elsevier. 2018, pp. 138–146.
- [175] D. Alvarez-Garcia and X. Barril. “Relationship between protein flexibility and binding: Lessons for structure-based drug design”. In: *Journal of chemical theory and computation* 10.6 (2014), pp. 2608–2614.
- [176] T. R. Stachowski and M. Fischer. “Large-scale ligand perturbations of the protein conformational landscape reveal state-specific interaction hotspots”. In: *Journal of Medicinal Chemistry* 65.20 (2022), pp. 13692–13704.
- [177] E. F. Pettersen, T. D. Goddard, C. C. Huang, E. C. Meng, G. S. Couch, T. I. Croll, J. H. Morris, and T. E. Ferrin. “UCSF ChimeraX: Structure visualization for researchers, educators, and developers”. In: *Protein Science* 30.1 (2021), pp. 70–82.
- [178] Schrödinger, LLC. “The PyMOL Molecular Graphics System, Version 1.8”. 2015.
- [179] Z. Miao and Y. Cao. “Quantifying side-chain conformational variations in protein structure”. In: *Scientific reports* 6.1 (2016), p. 37024.
- [180] S. Urbaczek, A. Kolodzik, J. R. Fischer, T. Lippert, S. Heuser, I. Groth, T. Schulz-Gasch, and M. Rarey. “NAOMI: on the almost trivial task of reading molecules from different file formats”. In: *Journal of chemical information and modeling* 51.12 (2011), pp. 3199–3207.
- [181] S. Urbaczek, A. Kolodzik, I. Groth, S. Heuser, and M. Rarey. “Reading pdb: perception of molecules from 3d atomic coordinates”. In: *Journal of chemical information and modeling* 53.1 (2013), pp. 76–87.
- [182] S. Urbaczek, A. Kolodzik, and M. Rarey. “The valence state combination model: a generic framework for handling tautomers and protonation states”. In: *Journal of Chemical Information and Modeling* 54.3 (2014), pp. 756–766.
- [183] J. Desaphy, G. Bret, D. Rognan, and E. Kellenberger. “sc-PDB: a 3D-database of ligandable binding sites—10 years on”. In: *Nucleic acids research* 43.D1 (2015), pp. D399–D404.
- [184] F. Flachsenberg, A. Meyder, K. Sommer, P. Penner, and M. Rarey. “A consistent scheme for gradient-based optimization of protein–ligand poses”. In: *Journal of Chemical Information and Modeling* 60.12 (2020), pp. 6502–6522.

- [185] R. E. Amaro, J. Baudry, J. Chodera, Ö. Demir, J. A. McCammon, Y. Miao, and J. C. Smith. “Ensemble docking in drug discovery”. In: *Biophysical journal* 114.10 (2018), pp. 2271–2278.
- [186] F. Pucci, K. V. Bernaerts, J. M. Kwasigroch, and M. Rooman. “Quantification of biases in predictions of protein stability changes upon mutations”. In: *Bioinformatics* 34.21 (2018), pp. 3659–3665.
- [187] A. P. Pandurangan, B. Ochoa-Montano, D. B. Ascher, and T. L. Blundell. “SDM: a server for predicting effects of mutations on protein stability”. In: *Nucleic acids research* 45.W1 (2017), W229–W235.
- [188] B. Li, Y. T. Yang, J. A. Capra, and M. B. Gerstein. “Predicting changes in protein thermodynamic stability upon point mutation with deep 3D convolutional neural networks”. In: *PLoS computational biology* 16.11 (2020), e1008291.
- [189] G. Thiltgen and R. A. Goldstein. “Assessing predictors of changes in protein stability upon mutation using self-consistency”. In: *PloS one* 7.10 (2012), e46084.
- [190] Y. Li, J. Zhang, D. Tai, C. Russell Middaugh, Y. Zhang, and J. Fang. “PROTS: A fragment based protein thermo-stability potential”. In: *Proteins: Structure, Function, and Bioinformatics* 80.1 (2012), pp. 81–92.
- [191] G. R. Buel and K. J. Walters. “Can AlphaFold2 predict the impact of missense mutations on structure?” In: *Nature Structural & Molecular Biology* 29.1 (2022), pp. 1–2.
- [192] M. A. Pak, K. A. Markhieva, M. S. Novikova, D. S. Petrov, I. S. Vorobyev, E. S. Maksimova, F. A. Kondrashov, and D. N. Ivankov. “Using AlphaFold to predict the impact of single mutations on protein stability and function”. In: *Plos one* 18.3 (2023), e0282689.
- [193] S. Bietz and M. Rarey. “ASCONA: rapid detection and alignment of protein binding site conformations”. In: *Journal of Chemical Information and Modeling* 55.8 (2015), pp. 1747–1756.
- [194] L. Holm and C. Sander. “Protein structure comparison by alignment of distance matrices”. In: *Journal of molecular biology* 233.1 (1993), pp. 123–138.
- [195] C. Ehrt, T. Brinkjost, and O. Koch. “A benchmark driven guide to binding site comparison: An exhaustive evaluation using tailor-made data sets (ProSPECCTs)”. In: *PLoS computational biology* 14.11 (2018), e1006483.

- 
- [196] S. Li, C. Cai, J. Gong, X. Liu, and H. Li. “A fast protein binding site comparison algorithm for proteome-wide protein function prediction and drug repurposing”. In: *Proteins: Structure, Function, and Bioinformatics* 89.11 (2021), pp. 1541–1556.
- [197] M. Simonovsky and J. Meyers. “DeeplyTough: learning structural comparison of protein binding sites”. In: *Journal of chemical information and modeling* 60.4 (2020), pp. 2356–2366.
- [198] M. Eguida and D. Rognan. “A computer vision approach to align and compare protein cavities: application to fragment-based drug design”. In: *Journal of Medicinal Chemistry* 63.13 (2020), pp. 7127–7142.
- [199] S. Nørager, S. Arent, O. Björnberg, M. Ottosen, L. L. Leggio, K. F. Jensen, and S. Larsen. “Lactococcus lactis dihydroorotate dehydrogenase A mutants reveal important facets of the enzymatic function”. In: *Journal of Biological Chemistry* 278.31 (2003), pp. 28812–28822.
- [200] E. Ukkonen. “Approximate string-matching with q-grams and maximal matches”. In: *Theoretical computer science* 92.1 (1992), pp. 191–211.
- [201] M. S. Kumar, K. A. Bava, M. M. Gromiha, P. Prabakaran, K. Kitajima, H. Uedaira, and A. Sarai. “ProTherm and ProNIT: thermodynamic databases for proteins and protein–nucleic acid interactions”. In: *Nucleic acids research* 34.suppl\_1 (2006), pp. D204–D206.
- [202] J. S. Xavier, T.-B. Nguyen, M. Karmarkar, S. Portelli, P. M. Rezende, J. P. Velloso, D. B. Ascher, and D. E. Pires. “ThermoMutDB: a thermodynamic database for missense mutations”. In: *Nucleic acids research* 49.D1 (2021), pp. D475–D479.
- [203] R. Nikam, A. Kulandaisamy, K. Harini, D. Sharma, and M. M. Gromiha. “ProThermDB: thermodynamic database for proteins and mutants revisited after 15 years”. In: *Nucleic acids research* 49.D1 (2021), pp. D420–D424.
- [204] J. Stourac, J. Dubrava, M. Musil, J. Horackova, J. Damborsky, S. Mazurenko, and D. Bednar. “FireProtDB: database of manually curated protein stability data”. In: *Nucleic acids research* 49.D1 (2021), pp. D319–D324.
- [205] J. Jančauskaitė, B. Jimenez-Garcia, J. Dapkūnas, J. Fernández-Recio, and I. H. Moal. “SKEMPI 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation”. In: *Bioinformatics* 35.3 (2019), pp. 462–469.

- [206] J. Graef, C. Ehrt, and M. Rarey. “Binding Site Detection Remastered: Enabling Fast, Robust, and Reliable Binding Site Detection and Descriptor Calculation with DoGSite3”. In: *Journal of Chemical Information and Modeling* 63.10 (2023), pp. 3128–3137.
- [207] A. C. Joerger, H. C. Ang, and A. R. Fersht. “Structural basis for understanding oncogenic p53 mutations and designing rescue drugs”. In: *Proceedings of the National Academy of Sciences* 103.41 (2006), pp. 15056–15061.
- [208] F. M. Boeckler, A. C. Joerger, G. Jaggi, T. J. Rutherford, D. B. Veprintsev, and A. R. Fersht. “Targeted rescue of a destabilized mutant of p53 by an in silico screened drug”. In: *Proceedings of the National Academy of Sciences* 105.30 (2008), pp. 10360–10365.
- [209] M. R. Bauer, A. Krämer, G. Settanni, R. N. Jones, X. Ni, R. Khan Tareque, A. R. Fersht, J. Spencer, and A. C. Joerger. “Targeting cavity-creating p53 cancer mutations with small-molecule stabilizers: the Y220X paradigm”. In: *ACS Chemical Biology* 15.3 (2020), pp. 657–668.
- [210] “UniProt: the universal protein knowledgebase in 2023”. In: *Nucleic Acids Research* 51.D1 (2023), pp. D523–D531.
- [211] K. C. de Andrade, E. E. Lee, E. M. Tookmanian, C. A. Kesserwan, J. J. Manfredi, J. N. Hatton, J. K. Loukissas, J. Zavadil, L. Zhou, M. Olivier, et al. “The TP53 database: transition from the International Agency for Research on Cancer to the US National Cancer Institute”. In: *Cell Death & Differentiation* 29.5 (2022), pp. 1071–1073.
- [212] T. Rognes. “Faster Smith-Waterman database searches with inter-sequence SIMD parallelisation”. In: *BMC bioinformatics* 12.1 (2011), pp. 1–11.
- [213] T. Stohn. *Kombination von experimentellen, alternativen Atomkoordinaten zur Generierung konsistenter Proteinstrukturen*. Master’s thesis. 2019.
- [214] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. “Scikit-learn: Machine learning in Python”. In: *the Journal of machine Learning research* 12 (2011), pp. 2825–2830.

# References of the Cumulative Dissertation

- [D1] **J. Sieg**, F. Flachsenberg, and M. Rarey. “In need of bias control: evaluating chemical data for machine learning in structure-based virtual screening”. In: *Journal of chemical information and modeling* 59.3 (2019), pp. 947–961.
- [D2] **J. Sieg**, C. C. Sandmeier, J. Lieske, A. Meents, C. Lemmen, W. R. Streit, and M. Rarey. “Analyzing structural features of proteins from deep-sea organisms”. In: *Proteins: Structure, Function, and Bioinformatics* 90.8 (2022), pp. 1521–1537.
- [D3] T. Gutermuth, **J. Sieg**, T. Stohn, and M. Rarey. “Modeling with Alternate Locations in X-ray Protein Structures”. In: *Journal of Chemical Information and Modeling* 63.8 (2023), pp. 2573–2585.
- [D4] **J. Sieg** and M. Rarey. “Searching similar local 3D micro-environments in protein structure databases with MicroMiner”. In: *Briefings in Bioinformatics* 24.6 (2023), bbad357.
- [D5] K. Schöning-Stierand, K. Diedrich, C. Ehrt, F. Flachsenberg, J. Graef, **J. Sieg**, P. Penner, M. Poppinga, A. Ungethüm, and M. Rarey. “Proteins Plus: a comprehensive collection of web-based molecular modeling tools”. In: *Nucleic Acids Research* 50.W1 (2022), W611–W615.



## Appendix A

# Scientific Contributions

### A.1 Publications

- [D1] **J. Sieg**, F. Flachsenberg, and M. Rarey. “In need of bias control: evaluating chemical data for machine learning in structure-based virtual screening”. In: *Journal of chemical information and modeling* 59.3 (2019), pp. 947–961.

This publication analyzes unbiasing strategies of benchmark data sets in structure-based virtual screening and develops the feature selection-based interpretability method described in this thesis. A subchapter in this publication was preliminary work conducted before this doctorate project during J. Sieg’s Master’s thesis [139]. The content of the Master’s thesis is limited to the subchapter "4. NON-CAUSAL BIAS IN LITERATURE" from page 954 to 957 in the publication, which is summarized in section 2.1.2 of this dissertation. The parts of the paper relevant to this dissertation describe the interpretability method and its application. J. Sieg wrote the manuscript, implemented the necessary software, curated the data, and performed analysis. F. Flachsenberg and M. Rarey contributed to the method development and analysis and provided feedback and supervision. All authors reviewed and approved the final manuscript.

- [D2] **J. Sieg**, C. C. Sandmeier, J. Lieske, A. Meents, C. Lemmen, W. R. Streit, and M. Rarey. “Analyzing structural features of proteins from deep-sea organisms”. In: *Proteins: Structure, Function, and Bioinformatics* 90.8 (2022), pp. 1521–1537.

This publication describes the analysis of high-pressure protein adaptations and the extension of the interpretability method to individual feature attributions. J. Sieg and M. Rarey conceptualized the work. J. Sieg performed the method development, analysis, investigation, validation and writing of the manuscript. J. Sieg, C. C. Sandmeier and J. Lieske conducted the data curation. A. Meents, C. Lemmen, W. R. Streit and M. Rarey provided supervision and feedback. All authors reviewed and approved the final manuscript.

- [D3] T. Gutermuth, **J. Sieg**, T. Stohn, and M. Rarey. “Modeling with Alternate Locations in X-ray Protein Structures”. In: *Journal of Chemical Information and Modeling* 63.8 (2023), pp. 2573–2585.

The publication presents a new method for handling alternate locations (AltLocs) implemented in the AltLocEnumerator tool. T. Stohn, J. Sieg and M. Rarey conceptualized the method, and T. Stohn implemented a prototype during his Master’s thesis [213]. J. Sieg and T. Gutermuth revised the method and implementation and performed the integration in NAOMI. J. Sieg focused on the implementation of the method and T. Gutermuth on the tool. T. Gutermuth, J. Sieg, and M. Rarey conceptualized the PDB analysis. T. Gutermuth and M. Rarey conceptualized the docking analysis. T. Gutermuth conducted the PDB and docking analysis. J. Sieg and T. Gutermuth wrote the paper. The paper is a shared first-author publication. T. Gutermuth and J. Sieg contributed equally. All authors reviewed and approved the final manuscript.

- [D4] **J. Sieg** and M. Rarey. “Searching similar local 3D micro-environments in protein structure databases with MicroMiner”. In: *Briefings in Bioinformatics* 24.6 (2023), bbad357.

This publication presents a new method for searching similar local 3D micro-environments in protein structure databases implemented in the tool MicroMiner. J. Sieg conceptualized the work and performed the method development, data curation, analysis, investigation, validation, and writing of the manuscript. M. Rarey contributed valuable



feedback and helped to drive the project. All authors reviewed and approved the final manuscript.

- [D5] K. Schöning-Stierand, K. Diedrich, C. Ehrt, F. Flachsenberg, J. Graef, **J. Sieg**, P. Penner, M. Poppinga, A. Ungethüm, and M. Rarey. “Proteins Plus: a comprehensive collection of web-based molecular modeling tools”. In: *Nucleic Acids Research* 50.W1 (2022), W611–W615.

This publication presents new features of the modeling server ProteinsPlus, including the interactive web server integration of MicroMiner and JAMDA, new GeoMine features, and AlphaFold structures integration. All authors contributed to the publication and wrote or revised the paper. J. Sieg and C. Ehrt wrote the text about MicroMiner, which is relevant to this dissertation. All authors reviewed and approved the final manuscript.

## A.2 Conference Contributions

### A.2.1 Talks

- [V1] **J. Sieg**, F. Flachsenberg, and M. Rarey. *In the Need of Bias Control: Evaluation of Chemical Data for Machine Learning Methods in Structure-Based Virtual Screening*. 11th International Conference on Chemical Structures (ICCS). Noordwijkerhout, Netherlands, 2018.
- [V2] **J. Sieg**, F. Flachsenberg, and M. Rarey. *Challenges in Protein-Structure-Driven Machine Learning and Applications in Biotechnology*. 2nd Machine learning and AI in (bio)chemical engineering (MABC). Cambridge, United Kingdom, 2019.
- [V3] **J. Sieg** and M. Rarey. *Data-Driven Analysis of Single Point Mutations through Rapid Scan of 3D Micro-Environments*. 29th Intelligent Systems for Molecular Biology (ISMB) / 20th European Conference on Computational Biology (ECCB). Virtual conference, 2021.
- [V4] **J. Sieg** and M. Rarey. *Computational Analysis of Protein Structures from Deep-Sea Organisms*. 13th European Congress of Chemical Engineering (ECCE) / 6th European Congress of Applied Biotechnology (ECAB). Virtual conference, 2021.

- [V5] **J. Sieg** and M. Rarey. *Data-Driven Analysis of Single Point Mutations through Rapid Scan of 3D Micro-Environments*. 3D-BioInfo 2021 Annual Meeting. Virtual conference, 2021.

### A.2.2 Poster

- [V1] **J. Sieg** and M. Rarey. *Data-Driven Analysis of Single Point Mutations through Rapid Scan of 3D Micro-Environments*. 29th Intelligent Systems for Molecular Biology (ISMB) / 20th European Conference on Computational Biology (ECCB). Virtual conference, 2021.
- [V2] **J. Sieg** and M. Rarey. *Data-Driven Analysis of Single Point Mutations through Rapid Scan of 3D Micro-Environments*. 3D-BioInfo 2021 Annual Meeting. Virtual conference, 2021.

## Appendix B

# Software Architecture

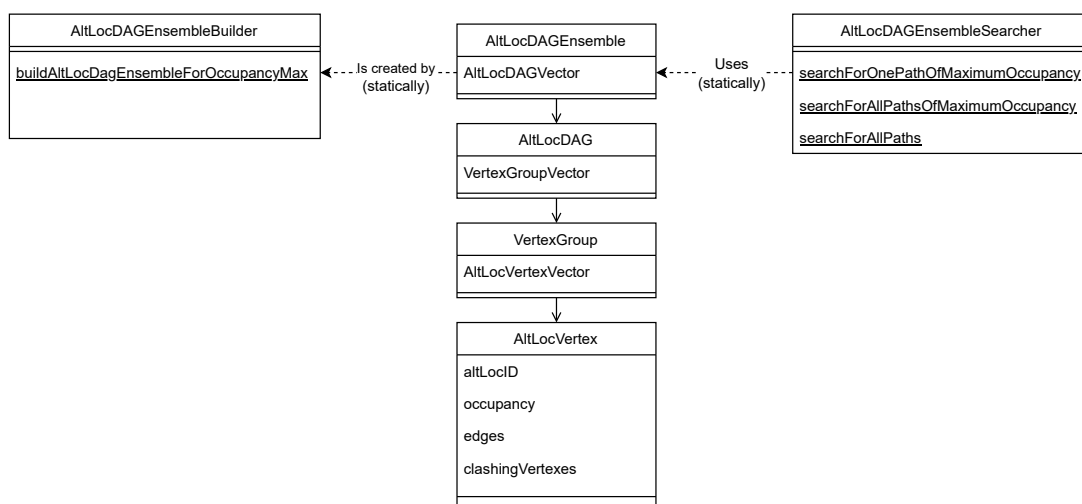
### B.1 NAOMI

NAOMI [180–182] is a software library written in the C++ programming language that provides functionality and tools for various cheminformatics and structural bioinformatics tasks and is used as the basis for most software developed in this doctorate. The NAOMI library already provides a broad spectrum of functionality for cheminformatics and bioinformatics methods and tasks. On the one hand, essential functionality is implemented, like reading common molecule [180] and protein structure [181] formats and transferring them into validated representations stored in data structures. On the other hand, elaborate functions for more specific tasks are available, including predicting non-covalent interactions, predicting binding pockets, aligning binding sites, and more. NAOMI is divided into a core library containing common and reusable functionality for multiple tasks and an application-oriented part providing user-facing command line tools utilizing the core library.

### B.2 AltLocEnumerator

The AltLocEnumerator method is implemented as part of NAOMI and split into tool and library code. The tool code handles the reading of the input structure, additional site specifications, and other user-provided arguments. It calls the library code to execute the AltLoc enumeration algorithm and writes the results to output structure files. On the other hand, the AltLoc enumeration method is implemented in the library code. Figure B.1 shows the main classes of the library. `AltLocDAGEnsemble` is the central data structure. It is built from the `Parser::PDBEntry` representation of the input structure file. The data structure provides necessary information on AltLoc

conformations encoded in an ensemble of annotated directed acyclic graphs (DAGs). The construction of the `AltLocDAGEnsemble` is conducted by a static function of the `AltLocDAGEnsembleBuilder` class. The created `AltLocDAGEnsemble` instance can then be used by one of the static search functions of the `AltLocDAGEnsembleSearcher` class to search for one or more valid `AltLoc` conformations of the overall structure complex. More specifically, an `AltLocDAGEnsemble` is composed of a vector of `AltLocDAG` instances representing individual DAGs. An `AltLocDAG` object contains a vector of `VertexGroup` instances, which organizes the DAG in layers. Lastly, a `VertexGroup` contains multiple `AltLocVertex` instances. An `AltLocVertex` represents a particular `AltLoc` conformation of a residue, storing its `AltLoc` identifier, occupancy, edges to other vertexes, and vertexes of other clashing conformations. Correspondingly, a `VertexGroup` holds all `AltLoc` conformations of a particular residue.



**Figure B.1:** Schematic illustration of important classes in the `AltLocLib` and their interactions. The central data structures and functions of the `AltLoc` enumeration algorithm are depicted. Arrows indicate the dependencies of classes, like composition and usage dependencies. The illustration is inspired by the Unified Modeling Language (UML).

### B.3 MicroMiner

The code of the `MicroMiner` program is separated into a core library called `MicroEnvLib` and tool code providing a command line interface. The `MicroEnvLib` supplies the components to perform residue 3D micro-environment searches. The specific functionality and algorithms are either implemented directly in the `MicroEnvLib` or are used by the `MicroEnvLib` from other core libraries in `NAOMI`. The `MicroEnvLib` uses functionality from the `ComplexLib` for handling protein structures as `Complex` data structures and

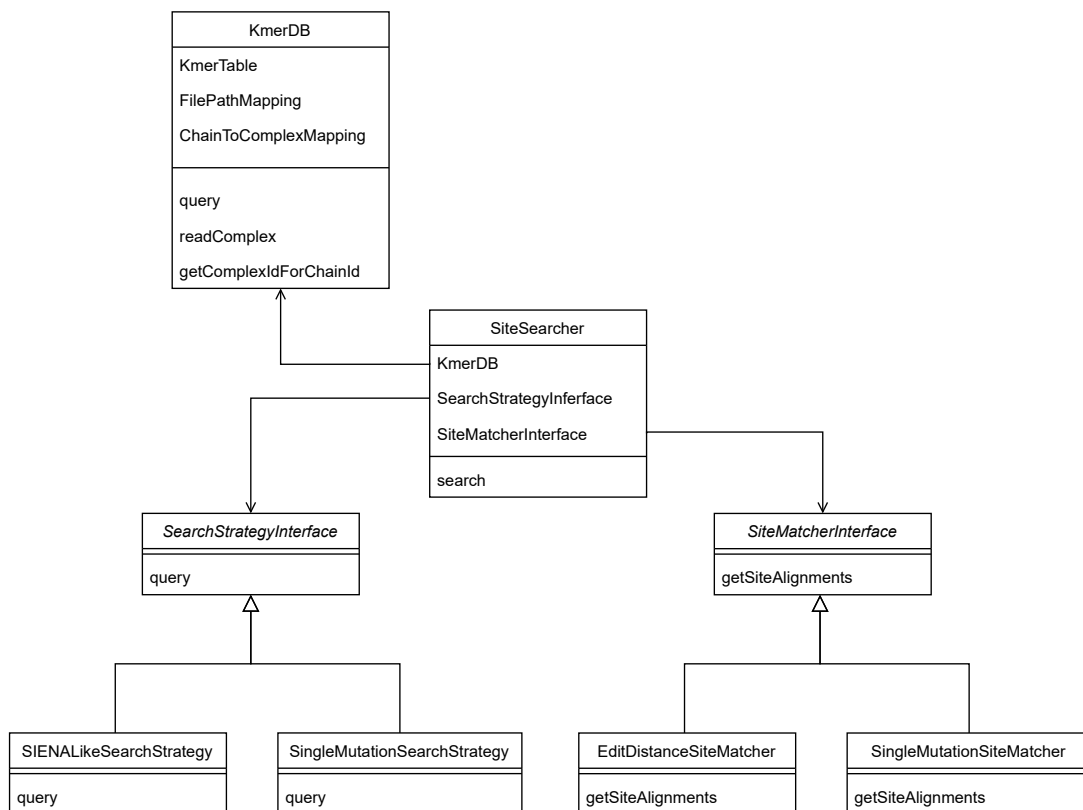
specific 3D sites in protein structures as so-called `ActiveSite` data structures. The `MicroEnvLib` also utilizes multiple alignment algorithms and alignment scores implemented in the `AlignLib` to align sites to candidate protein structures—multiple of which were implemented in this dissertation. Further, functions of the `ProteinFlexibilityLib` are used to superimpose hit structures on query structures.

The core classes of the `MicroEnvLib` are illustrated in Figure B.2. The `SiteSearcher` class builds the center and entry point for executing micro-environment similarity searches. The class executes the search workflow by taking a list of query micro-environments as input, first searching them against the `KmerDB` instance with a certain search strategy, and then matching candidate structures to the query micro-environments with a particular matching strategy. A `KmerDB` used for k-mer look-ups is stored as a member of `SiteSearcher`. The two steps of the search are represented by the two abstract classes `SearchStrategyInterface` and `SiteMatcherInterface` of which the `SiteSearcher` holds an implementation each. Derived classes of the `SearchStrategyInterface` should provide algorithms to retrieve candidate structures from the `KmerDB`. The `SIENALikeSearchStrategy`, as the name suggests, implements the SIENA [97] search strategy while the `SingleMutationSearchStrategy` runs the newly developed k-mer matching strategy for single mutations. On the other hand, implementations of the `SiteMatcherInterface` should provide functionality to match the query sites against the candidate structures. The `EditDistanceSiteMatcher` calls the site alignment algorithm ASCONA [193] uses. In contrast, the `SingleMutationMatcher` implements the newly developed variation of the ASCONA site alignment algorithm optimized for matching single mutations. The class structure was intentionally designed for the components to be exchangeable, which helped during the development to test different site search and matching algorithms by simply swapping instances of different derived classes.

The `MicroMiner` tool code defines and documents the tool’s command line interface. It handles the command line arguments given by a user as input. The arguments are validated and transferred to the internally used data structures and then given to the functions in the `MicroEnvLib` for calculation. The hits and other results generated are written to output files by the tool code. In addition, the tool code handles the input data splitting and chunking for parallel processing of multiple input protein structures.

## B.4 Feature Interpretability Method

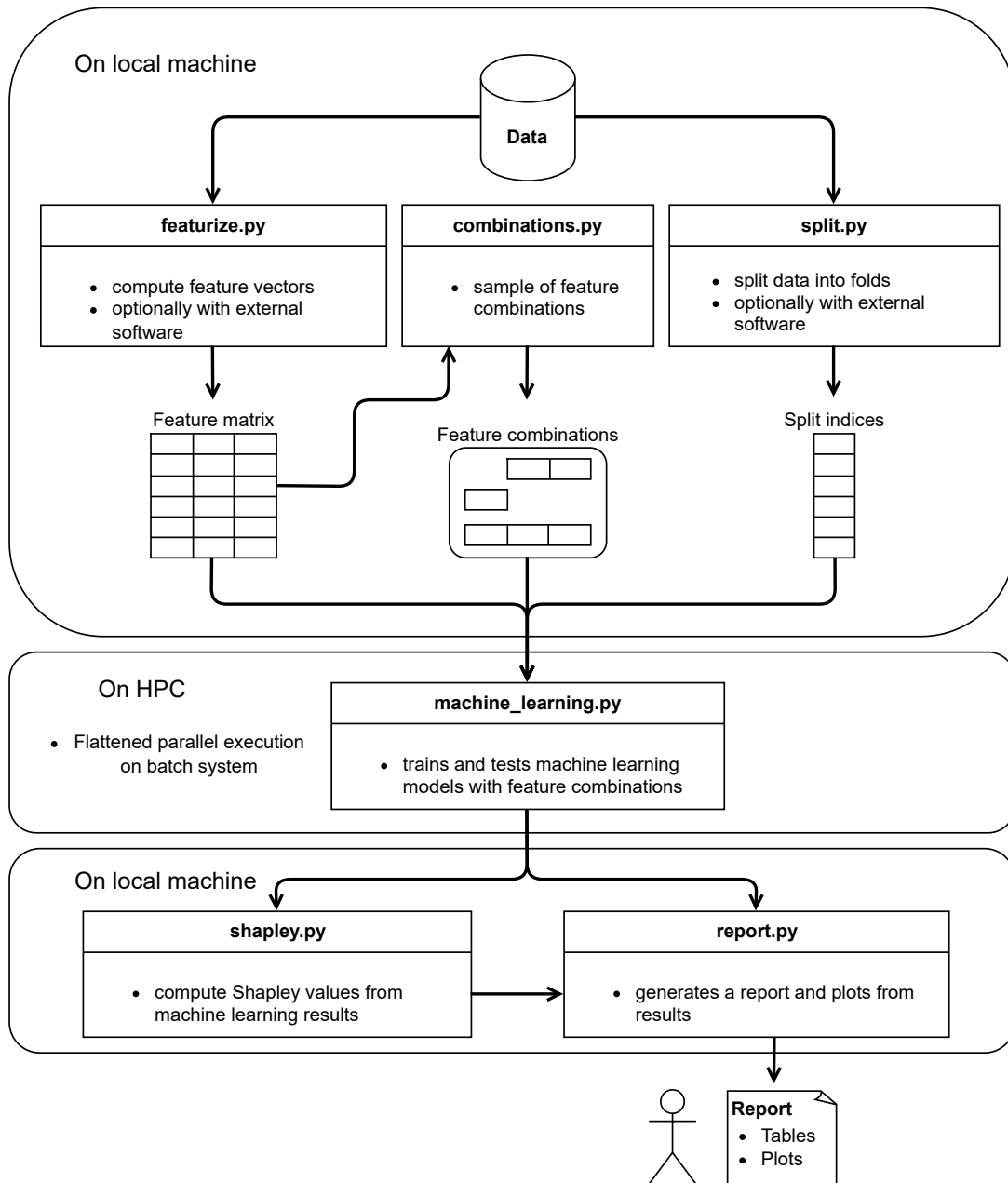
The software implementation of the interpretability method developed and used in the two analysis projects in [D1] and [D2] was realized as a collection of Python scripts



**Figure B.2:** Schematic depiction of the interactions between core classes in the MicroEnvLib. Arrows indicate the dependencies of classes, like composition, usage dependencies, and inheritance. The illustration is inspired by the Unified Modeling Language (UML).

closely coupled to the in-house high-performance cluster (HPC) at the Center for Bioinformatics, Hamburg. In contrast to the other projects, these projects were driven by the analysis and not the development of a dedicated software tool. Figure B.3 illustrates the workflow between the most important Python scripts. The workflow starts with the `featurize.py` script to calculate the feature matrix for a given set of input data and the `split.py` script that calculates a split of the input data into training and test sets. Both scripts may incorporate additional information provided by external software. For example, the `featurize.py` script can incorporate features computed by external tools or libraries, like NAOMI, RDKit, or command line tools. The `split.py` script can use additional information like clusterings of the input data samples to create train and test folds. A set of feature combinations is sampled from the generated feature matrix with the `combinations.py` script. This sampling can be done with different strategies. For example, in the [D2] project, all feature combinations of up to five features were used as the sampled set. The three described scripts are executed locally. In contrast, the

`machine_learning.py` script is executed in different instances over multiple computation nodes using array jobs of the Sun Grid Engine batch system that is operated at the Center for Bioinformatics. One instance of the script processes a batch of the flattened feature selection experiments, which are defined by a certain training set, a test set, and a feature combination. The `machine_learning.py` scripts utilize the scikit-learn library [214] for running various machine learning algorithms. The results from the machine learning computations are collected and analyzed locally. With the `shapley.py` script Shapley values estimates can be computed from the machine learning results. Finally, the machine learning results and the computed Shapley values can be put into a user-centered report format, including plots and tables.



**Figure B.3:** Schematic illustration of the implementation of the interpretability method and the information flow between central components. Arrows indicate the flow of information. The computation is executed by six Python scripts, which are executed on either a local machine or on the HPC. The first step on the local machine prepares all necessary files for an intensive feature selection with wrapper methods experiment. In the second step, machine learning models are trained and evaluated on the prepared input data. In the third step, the machine learning results are used to compute Shapley values, tables, and plots, which are finally reported to the user as a set of tables and plots.



# Appendix C

## Software Usage

### C.1 AltLocEnumerator

#### C.1.1 Command Line Tool

The following shows the interface of the AltLocEnumerator command line tool:

```
$ ./bin/AltLocEnumerator --help
```

```
Tool for enumerating alternative residue conformations (AltLocs) of
a protein structure. The input to AltLocEnumerator is a single PDB
or CIF file. The output is one or more PDB file(s) containing the
alternative conformations
```

```
Usage: AltLocEnumerator -f [protein] -a [Mode]
```

```
The following options are supported:
```

```
--license LicenseKey          Activate a new license
```

```
General Options:
```

```
-h [ --help ]                Show this help and exit.
-f [ --file ] file          Protein input file in format
                             PDB/PDBxmmCIF
-a [ --altLoc ] arg (=default) Enumeration mode. Single complex
                             modes: 'Default', 'BestOcc',
                             'altLocId'. Multiple complex
                             modes: 'AllBest', 'All'.
```

Optional Options:

-l [ --ligand ] arg Input ligand file for site of interest definition  
sdf/mol2/PDB/PDBxmmCIF  
format, only residues in the site of interest will be enumerated

-p [ --pocket ] arg Input pocket by pocket  
PDB/PDMxmmCIF file for site of interest definition, only residues in the site of interest will be enumerated

-e [ --edf ] arg Input pocket by EDF file for site of interest definition, only residues in the site of interest will be enumerated

-o [ --outputfolder ] arg (=./) Output directory

-w [ --writepdb ] arg (=1) write the new pdb files. Can be used to only determine the number of complexes that would be enumerated without writing the PDBs.

-t [ --hetatoms ] arg (=1) Include HETATM entries. If false HETATMs will be removed in preprocessing.

-v [ --verbosity ] arg (=warning) UinfoLevel [Quiet (0), Error (1), Warning (2), Info (3), Steps (4)]

-r [ --activesiteradius ] arg (=6.5) The radius used for active site construction when a ligand file is supplied

-m [ --minRMSD ] arg (=0) Filters conformations by RMSD. Only used for the enumeration modes that export multiple complexes

-n [ --maxNumberComplexes ] arg (=100) Maximum number of complex conformations that will be generated. Only used for the enumeration modes that export multiple complexes. A high number will result in high memory usage (not advised).

--altlocid arg The altLocId to select and use. Only used when altLoc enumeration mode is 'altLocId'

The AltLocEnumerator takes an input structure file with `--file` option and writes structure conformations to files into a directory given with `--outputfolder`.

Five AltLoc enumeration strategies can be selected with the `--altLoc` option. The three single complex modes generating a single output structure file comprise the **Default** strategy, which implements NAOMI's default enumeration strategy before AltLocEnumerator was developed, which is the first-encounter strategy. With the **BestOcc** strategy, one AltLoc conformation with the highest occupancy is selected. Further, the **altLocId** strategy generates the structure conformation using the AltLoc identifier specified with `--altlocid`. In contrast, the two multiple complex modes can generate multiple output structure files. The strategy **AllBest** generates all conformations having the highest occupancy, while the **All** strategy generates output structure files for all valid conformations.

Optionally, AltLocEnumerator can focus on the conformation enumeration of specific sites, like binding sites. With `--ligand`, a small molecule file can be provided to define the site. The size of the ligand-defined site can be controlled with the `--activesiteradius` parameter that defines the distance cutoff to include neighboring residues. Alternatively, residue lists can be given to specify a site, either as a PDB/mmCIF file with the `--pocket` parameter or an EDF with the `--edf` option.

Further options allow controlling the writing of complexes to file (`--writepdb`) or the inclusion of HETATM entries in the enumeration (`--hetatoms`). The logging level can be set with the `--verbosity` parameters. In addition, the generation of multiple complexes can be limited to a fixed number using the `--maxNumberComplexes` option. This can be useful when there are thousands of structure conformations to be enumerated. Furthermore, generated structure conformations can be filtered by RMSD with `--minRMSD` to generate a smaller and more diverse set of conformations. New licenses can be activated with the `--license` option.

The AltLocEnumerator tool can be called via the command line as shown in the following example:

```
./bin/AltLocEnumerator --file protein.pdb --altLoc BestOcc -o out_dir
```

With this command, a single structure conformation with the highest occupancy is generated for `protein.pdb` and written as a PDB file to `out_dir/protein_1.pdb`.

## C.2 MicroMiner

### C.2.1 Command Line Tool

MicroMiner's command line interface offers two modules called `index_builder` and `search`. The usage of the MicroMiner tool and its modules is described in the next sections.

#### C.2.1.1 The `index_builder` module

The `index_builder` module creates a k-mer search index from a directory of protein structure files for fast look-ups of structure complexes containing particular k-mers.

The `index_builder` of MicroMiner has the following command line interface:

```
$ ./MicroMiner index_builder --help
```

```
MicroMiner index_builder ...:
```

```
-h [ --help ]           Print help message
-i [ --input ] arg      Input directories. Will be searched
                        recursively for PDB/mmCIF files. Use
                        with --extension.
-e [ --extension ] arg (=cif.gz) File extension of structure files to
                        read with --input. Supports common
                        formats like .pdb, .ent, .cif, .mcif
                        (note the leading .). For reading
                        compressed files add .gz, like .ent.gz.
-o [ --output ] arg     Output prefix.
-k [ --kmerSize ] arg (=6) k-mer size.
-c [ --cpus ] arg (=1)  Number of threads to use.
```

The `index_builder` module takes one or multiple directories as input with the `--input` option and extracts structure files that have the file extension specified with `--extension`. Different file extensions of standard structure file formats are supported by NAOMI, including gzip-compressed files. The module generates multiple output files representing the k-mer index. A common prefix is given to all generated output files that can be specified with `--output`. The size of k-mers can be set with the `--kmerSize` option, and the number of threads to use for parallel construction of the k-mer index can be selected with `--cpus`.

file name	file size in MB
my_index.db.bin	1,186
my_index.db.paths	11
my_index.db.map	6
my_index.db.log	12

**Table C.1:** Disc space usage of the MicroMiner k-mer search index for the PDB containing 199,534 structure files and using a kmerSize of 5.

An example call to the `index_builder` looks like this:

```
./MicroMiner index_builder -i structure_dir -o my -e .pdb -k 5 --cpus 6
```

This command will generate four files that have the prefix 'my' and different suffixes, which are explained in the following listing:

- `my_index.db.bin`: The serialized k-mer index data structure contains all k-mers and the information in which sequence and sequence position each k-mer appears.
- `my_index.db.paths`: Mapping of structure complex entries to their structure files on the disc.
- `my_index.db.map`: Maps sequence entries to their parent complex entry.
- `my_index.db.log`: A log file containing information on structure files that could be successfully processed or failed and a summary of k-mers in the search index.

The disc space usage of the four files for the PDB version used in the MicroMiner publication is listed in Table C.1.

### C.2.1.2 The search module

The `search` module performs searches against a protein structure database to find similar local 3D micro-environments to a query. The `search` module uses the k-mer search index that first must be created with the `index_builder` module.

MicroMiners `search` module command line interface looks like the following:

```
./bin/MicroMiner_release search --help
```

```
MicroMiner search ...:
```

-h [ --help ]	Print help message
-q [ --complex ] arg	Input query PDB file(s). Search with specific structure files.
--query_dirs arg	Read query structure files from directories. Use with --extension.
--extension arg (=cif.gz)	File extension of query structure files to read with --query_dirs. Supports common formats like .pdb, .ent, .cif, .mcif (note the leading .). For reading compressed files add .gz, like .ent.gz.
-e [ --edf ] arg	Limit query reference residues to residue list in EDF (ensemble data file). Can only be used with the --complex option and with a single input complex.
-s [ --searchdb ] arg	Search k-mer index. Provide the file prefix.
-o [ --output ] arg	Output prefix for files to be generated.
-w [ --write ]	Whether to write PDB files of hits superposed on the query. For large hit numbers this can take multiple GB of disc space.
-c [ --cpus ] arg (=1)	Number of threads to use.
-m [ --mode ] arg (=standard)	Kmer search mode. Can be 'standard' or 'single_mutation'.
-r [ --representation ] arg (=full_complex)	Define how to use the protein structure from the input file. Can be 'full_complex', 'monomer' or 'ppi'.
Algorithm options:	
--site_radius arg (=6.5)	Radius to define the residue 3D micro-environment.
--identity arg (=0.7)	Minimum site identity

---

<code>--fragment_length arg (=7)</code>	threshold [0.3 .. 1.0] Minimum length of a sequence fragment. Shorter fragments will be elongated [3 .. 15]
<code>--fragment_distance arg (=3)</code>	Maximum fragment distance/ mismatches in sequence [0 .. 10]
<code>--flexibility_sensitivity arg (=0.6)</code>	Degree of accepted structural flexibility [0.0 .. 1.0]
<code>--score_threshold arg (=80)</code>	Threshold for substitution scoring. Only used in 'single_mutation' mode for the seed & extend step. The threshold follows the MMseqs2 k-mer scoring scheme.
<code>--kmer_matching_rate arg (=0.9)</code>	Percent of unique k-mers that need to match in hit candidate. Only used in 'standard' mode [0.0 .. 1.0]

For performing a search with the module, MicroMiner needs the k-mer index, which is set with the `--searchdb` option. Query structures can be provided in two different ways. First, query structure files can be provided with `--complex`, which takes one or multiple structure file paths via the command line. Note that the number of query files can be limited. For example, on Linux systems, there is a maximum number of bytes a command line call is allowed to have. For this reason, there is a second way of providing query structures using `--query_dirs` and `--extension` analogously to the `--input` parameter in the `index_builder` module. Directories given with the `--query_dirs` option are searched recursively for query structure files with the specified file extension. Per default, MicroMiner generates query micro-environments for each residue in a query structure. With the `--edf` argument, an Ensemble Data File (EDF) can be provided. This file specifies a list of reference residues for which query micro-environments should be generated. Note that the `--edf` argument can only be used with the `--complex` option and with a single input complex. Three kinds of query micro-environments preselection strategies are provided that can be switched with the `--representation` option. The `full_complex` mode constructs query micro-environments from the query

structures as they are present in the structure file. In contrast, the 'monomer' mode considers only residues of the same chain for the environment construction. Lastly, the 'ppi' mode only uses micro-environments for the search, containing residues from at least two chains. In addition, with the `--mode` option, two different search modes can be selected. The `standard` mode searches with the same strategy as SIENA [97] while the `single_mutation` mode searches only for single mutations.

Multiple algorithmic options can be set for the search. The size of micro-environments can be controlled with the `--site_radius` option. The `--identity` parameter sets a lower bound for sequence identity of the aligned sites (only used in `standard` mode). The `--fragment_length` option specifies the minimal length sequence fragments need to have. With `--fragment_distance`, a limit can be set for the mismatches between aligned sequence fragments. The `--flexibility_sensitivity` controls the tolerance of the structural comparison. These algorithmic options are analogous to the options in SIENAs [97] command line interface. Further, two new algorithmic options are incorporated. The `--score_threshold` controls the extend steps in the sequence matching in the `single_mutation` mode. With the `--kmer_matching_rate` option, the percentage of k-mers that has to match a query site can be set (only used in `standard` mode).

Analogously to the `index_builder`, the output prefix for generated files is specified with `--output` in the `search` module. The default output of a MicroMiner search is a table listing the hits. In addition, using the `--write` flag, PDB structure files of the hits superposed to the query micro-environment can be generated for visual inspection. The processing and aligning of candidate structures can be parallized. The `--cpus` controls the number of threads to use.

The following shows an example call to the `search` module:

```
MicroMiner search -q 2imm.pdb -s my_index.db -o my_query -m single_mutation
```

This call generates a single output file with the name `my_query_resultStatistic.csv` listing the hits. The following illustrates the first three lines of the file, showing the header line and two hits.

```
queryName queryAA queryChain queryPos hitName hitAA hitChain hitPos siteIdentity
siteBackBoneRMSD siteAllAtomRMSD nofSiteResidues alignmentLDDT fullSeqId
2IMM PRO A 43 3W13 SER D 49 0.875 0.431 0.936 8 0.946 0.842
2IMM PRO A 43 43C9 SER A 43 0.875 0.180 1.496 8 1.000 0.772
```



If the above command would, in addition, be run with the `--write` flag, an additional directory would be created with the name `my_query_ensemble`, which contains PDB structure files of superimposed hits. Structure files in the directory are organized by query site (as defined by the reference residue). Hit PDB files are named after both the hit and the hit residue aligned to the query's reference residue. An example file path is `out_ensemble/2IMM_PRO_A_43/3W13_SER_D_49.pdb`.

## C.2.2 Evaluation and Application Scripts

The scripts for evaluating and applying *MicroMiner* were realized as a separate Python project available at [https://github.com/rareylab/microminer\\_utils](https://github.com/rareylab/microminer_utils). The project is called `microminer_utils` and consists of multiple Python and Shell scripts, a Python package, and Jupyter Notebooks with which the experiments in the *MicroMiner* publication were conducted. See the `README.md` file in the repository for a description of how to reproduce the experiments from the paper.

The Python package is called `helper` and bundles common functionality, such as calling command line tools like *MicroMiner* and *TM-align*. It standardizes different mutation data sets to a unified format. In addition, an adapter for the internal high-performance cluster (HPC) was written.

The Python scripts in the repository's top-level directory build the command line interface for executing *MicroMiner* and *TM-align* on a particular data set. The scripts use the functionality in the `helper` Python package to prepare the input and output for *MicroMiner* and *TM-align* and handle the parallel execution on the local machine or the HPC.

Further, the Shell scripts execute specific experiments from the paper and can be used to reproduce the published results. They define the list of data sets to use and the successive calls to different Python scripts for running the experiments. Additionally, system and infrastructure-specific options, like the directories for writing results or whether the experiments should be run on the HPC or locally, are set.

Lastly, the code to generate the plots and other published results can be generated with the Jupyter Notebooks. They receive the results generated with the Shell scripts as input and construct the plots and other details.

The following is an overview of the most noteworthy scripts:

- `create_dataset.py`: Prepares raw data sets for the execution with *MicroMiner* and *TM-align*.
- `search.py`: Runs a search with *MicroMiner*.

- `eval_known_mutations.py`: Compares MicroMiner results to known wild-type/mutant structure pairs.
- `annotate_mutation_datasets.py`: Uses MicroMiner hits to annotate mutant structures to existing mutation data sets, like ProTherm.
- `annotation_statistics.py`: Compares the mutant structure annotations with and without MicroMiner-derived structures.
- `run_mutation_benchmark.sh`: Runs the experiment in the section 'Evaluation of MicroMiner for structural mutation search' in the paper.
- `run_pdb_experiments.sh`: Runs the experiment in the section 'Single mutations in the PDB' in the paper.
- `run_mutation_annotation.sh`: Runs the experiment in the section 'Annotating mutation effect measurements with structures for the mutant' in the paper.

### C.2.3 Web server

The MicroMiner tool is available at the ProteinsPlus server at <https://proteins.plus/>. The ProteinsPlus is a web service developed within the Center for Bioinformatics Hamburg. The web interface and the use of MicroMiner on the server are described in detail in the publication [D5]. A user can use the single mutation search of MicroMiner through the web interface for PDB, AlphaFoldDB, and custom-uploaded structures. MicroMiner searches a given query structure against a weekly updated PDB mirror. MicroMiner can be run for all residues in a query structure. The results will be presented in an interactive and filterable hit table. By clicking on a hit in the table, the hit structure is displayed in the 3D structure viewer, and the side chains of the aligned sites are shown and focused in the viewer for inspection by the user. Further, the hit table can be filtered, for example, by RMSD of the local environments to investigate structural changes upon mutation.

## Appendix D

### Journal Articles

#### D.1 In Need of Bias Control: Evaluating Chemical Data for Machine Learning in Structure-Based Virtual Screening

- [D1] **J. Sieg**, F. Flachsenberg, and M. Rarey. “In need of bias control: evaluating chemical data for machine learning in structure-based virtual screening”. In: *Journal of chemical information and modeling* 59.3 (2019), pp. 947–961.

Available: <https://doi.org/10.1021/acs.jcim.8b00712>. Reprinted with permission from [D1]. Copyright 2019 American Chemical Society.

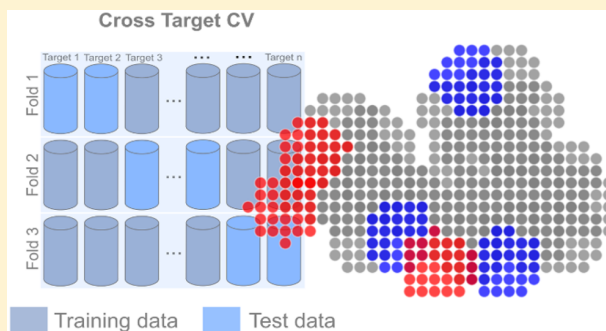
# In Need of Bias Control: Evaluating Chemical Data for Machine Learning in Structure-Based Virtual Screening

Jochen Sieg,<sup>1b</sup> Florian Flachsenberg,<sup>1b</sup> and Matthias Rarey\*<sup>1b</sup>

Universität Hamburg, ZBH - Center for Bioinformatics, Research Group for Computational Molecular Design, Bundesstraße 43, 20146 Hamburg, Germany

**S** Supporting Information

**ABSTRACT:** Reports of successful applications of machine learning (ML) methods in structure-based virtual screening (SBVS) are increasing. ML methods such as convolutional neural networks show promising results and often outperform traditional methods such as empirical scoring functions in retrospective validation. However, trained ML models are often treated as black boxes and are not straightforwardly interpretable. In most cases, it is unknown which features in the data are decisive and whether a model's predictions are right for the right reason. Hence, we re-evaluated three widely used benchmark data sets in the context of ML methods and came to the conclusion that not every benchmark data set is suitable. Moreover, we demonstrate on two examples from current literature that bias is learned implicitly and unnoticed from standard benchmarks. On the basis of these results, we conclude that there is a need for eligible validation experiments and benchmark data sets suited to ML for more bias-controlled validation in ML-based SBVS. Therefore, we provide guidelines for setting up validation experiments and give a perspective on how new data sets could be generated.



## 1. INTRODUCTION

The basic task of virtual screening (VS) is to prioritize large *in silico* molecule libraries by the probability of the molecules to show activity against a particular protein target. A distinction can be made between ligand-based and structure-based VS. In ligand-based VS, new active molecules are predicted based on their similarity to known actives. In contrast, structure-based VS (SBVS) methods model the interactions between small molecules and the protein to predict actives.<sup>1</sup>

New VS methods are typically tested by retrospective validation on benchmark data sets.<sup>2,3</sup> Benchmark data sets contain molecules of known activity and are used for a standardized comparison of different methods to select the method best suited for a task.<sup>4</sup> Frequently used examples are the Directory of Useful Decoys (DUD),<sup>5</sup> the Directory of Useful Decoys - Enhanced (DUD-E),<sup>6</sup> Demanding Evaluation Kits for Objective *in silico* Screening (DEKOIS),<sup>7,8</sup> the Maximum Unbiased Validation Data set (MUV),<sup>4</sup> the Community Structure–Activity Resource (CSAR),<sup>9</sup> PDBbind,<sup>10</sup> and more.<sup>11</sup>

In recent years, machine learning (ML) methods have been trained and evaluated on these data sets.<sup>12–17</sup> The reported results show that ML methods outperform other methodologies such as empirical and knowledge-based scoring functions on these data sets.<sup>12–17</sup> However, the interpretability of many ML methods is not straightforward.<sup>18</sup> On the one hand, it is of great interest to understand the determinants of

decision making of high-performing models to deduce the relationships potentially not captured otherwise. On the other hand, it is not recognizable whether a model's decisions are based on real signals in the data or on bias. We made two conspicuous observations in current literature that suggest that the latter is the case and bias is learned unnoticed from established data sets. We see the reason for this bias in the insufficiency of the current standard of validation experiment design, which is consistent with recent findings in a similar domain.<sup>19</sup>

In the following, an overview of benchmark data sets for VS and their advantages and disadvantages is given. Then it is evaluated if the unbiasing protocols of the examined data sets are suited for ML methods on the examples of the DUD, DUD-E and MUV data sets. Subsequently, our observations from the literature are described and analyzed. The results reveal that small molecule features dominate the predictions across dissimilar proteins when actually a structure-based descriptor is used, leading to biased models. Based on these results and observations, we propose guidelines for validation experiments to avoid bias and finally give an outlook on the generation of new data sets suitable for ML methods.

**Special Issue:** Machine Learning in Drug Discovery

**Received:** October 12, 2018

**Published:** March 5, 2019

**1.1. Examples of Benchmark Data Sets in Virtual Screening.** Benchmark data sets consist of sets of active and inactive molecules, each associated with a specific target. Often the actives are experimentally validated, but the documentation of experimentally validated inactive molecules is scarce. For this reason, assumed inactives, called decoys are frequently used.<sup>20</sup>

The first benchmark-like data sets were published in the early 2000s.<sup>21–23</sup> These data sets include randomly selected molecules as assumed inactives, and first approaches for picking samples while avoiding bias caused by the data set's compositions have been undertaken.<sup>23</sup>

Different biases have been identified by the community over time, which can either artificially increase or decrease prediction performance. Verdonk revealed that differences in distributions of basic physicochemical molecular properties of the active and inactive sets leads to artificial discrimination by those low dimensional features rather than features of higher dimensions.<sup>24</sup> An example is that many scoring functions favor molecules of larger size in docking because the potential number of interactions correlates with size.<sup>24</sup> This bias has been described by the term artificial enrichment and has been a problem when random molecules were selected as inactives. It can be counteracted by selecting inactives such that they are similar to the active molecules in terms of low dimensional properties.<sup>24</sup> In contrast to overestimations, the enrichment can be underestimated when utilizing experimentally non-validated inactives for which the assumption of inactivity turns out to be false.<sup>5</sup> This bias is termed false negative bias.

In 2006, Huang et al. introduced the DUD data set, which in its generation protocol addresses artificial enrichment and false negative bias.<sup>5</sup> DUD focuses on docking methods and comprises 40 different protein targets. The original DUD version contained 2950 actives and 95 326 assumed inactives. To circumvent the problem of the deficiency of experimentally validated inactive molecules, so-called decoys are selected *in silico* from the ZINC database.<sup>25</sup> Artificial enrichment has been addressed by selecting decoys such that they resemble the active molecules in their basic physicochemical properties. Those properties are molecular weight (MW), LogP, number of hydrogen bond acceptors and donors, as well as the number of rotatable bonds (see Table 1). It is worth mentioning that the presence of amine, amide, amidine and carboxylic acid has also been considered but with a lower priority. Simultaneously, to provide a higher confidence that decoys are actually inactive, the selection process ensures that each decoy is dissimilar to any of the active molecules with respect to CACTVS fingerprints<sup>26</sup> and a Tanimoto coefficient threshold of 0.9. The evaluation in a comparative docking study showed less artificial enrichment in DUD than in earlier data sets.<sup>5</sup> In fact, DUD has been considered the gold standard after its release and is still used today.<sup>20</sup>

Two years after the release of DUD, in 2008, analogue bias has been described by Good and Oprea.<sup>27</sup> Analogue bias is based on the observation that artificially improved enrichment can be achieved if a data set contains many analogue actives with the same chemotype. The activity of ligands in a cluster that share the same scaffold is easy to predict as soon as the activity of a single molecule of the cluster can be identified. Consequently, a common scaffold shared by actives but not present in the inactives leads to overestimations. This bias has been found in DUD.<sup>27</sup> A strategy to address this bias is to diversify the ligands by clustering actives by their scaffolds and

**Table 1. List of Unbiased Features of DUD, DUD-E, and MUV**

DUD <sup>5</sup>	DUD-E <sup>6</sup>	MUV <sup>4</sup>
molecular weight	molecular weight	
number of hydrogen bond acceptors	number of hydrogen bond acceptors	number of hydrogen bond acceptors
number of hydrogen bond donors	number of hydrogen bond donors	number of hydrogen bond donors
number of rotatable bonds	number of rotatable bonds	
logP	logP	logP
	net charge	
		number of all atoms
		number of heavy atoms
		number of boron atoms
		number of bromine atoms
		number of carbon atoms
		number of chlorine atoms
		number of fluorine atoms
		number of iodine atoms
		number of nitrogen atoms
		number of oxygen atoms
		number of phosphorus atoms
		number of sulfur atoms
		number of chiral centers
		number of ring systems
5 features	6 features	17 features

selecting representatives.<sup>27</sup> Another limitation of DUD has been the chosen set of matched properties. Multiple groups reported that net charges are a strong discriminative feature in the data set,<sup>28,29</sup> which may lead to artificial enrichment in validation.

In 2012, after DUD had been analyzed in many studies and shortcomings had been identified, the DUD-Enhanced (DUD-E) data set was published.<sup>6</sup> The DUD-E compilation protocol addresses shortcomings of DUD and simultaneously extends the DUD data set to 22 886 actives and 1 411 214 decoys for 102 targets. The additional actives were retrieved from ChEMBL<sup>30</sup> and the inactives from the ZINC<sup>25</sup> database. To address analogue bias, active molecules were clustered by their Bemis-Murcko scaffolds.<sup>31</sup> To further reduce artificial enrichment bias, net charges were added to the matched properties between actives and decoys (see Table 1). Finally, a more stringent topology filter was employed during decoy selection to further reduce the probability of false negative inactives.<sup>6</sup>

Most of the benchmark data sets in VS focus on structure-based methodologies such as docking.<sup>32</sup> A popular example of a benchmark data set specifically designed for ligand-based methods is the maximum unbiased validation (MUV) data set collection.<sup>4</sup> MUV was published in 2009 and it comprises 17 separate data sets each associated with a target protein. Each data set contains 30 active and 15 000 inactive molecules, all retrieved from PubChem.<sup>33</sup> Note that MUV contains experimentally analyzed actives and inactives. Therefore, the probability that the inactives are in fact inactive is high. Samples of MUV were selected by a strategy addressing the data set's representation in a certain descriptor space (termed the data set's topology) with methods from spatial statistics.



The goal of MUV design was to reduce artificial enrichment and analogue bias by selecting samples such that a common spread between actives and other actives as well as actives and inactives is employed in a descriptor space of 17 simple features (see Table 1). The goal is a data set topology in simple descriptor space in which the probability that the nearest neighbor of each active is an active or an inactive is equal.<sup>4</sup> The MUV data sets have been developed with the focus on ligand-based methods, but the authors note its usability in SBVS as well,<sup>4</sup> which has been done in studies.<sup>15</sup>

Table 2 gives an overview of the DUD, DUD-E, and MUV data set.

**Table 2. Overview of Three Benchmark Data Sets DUD, DUD-E, and MUV**

	DUD	DUD-E	MUV
number of targets	40	102	17
targeted methodology	docking	docking	ligand-based similarity search
number of unbiased features	5	6	17
special design decision	2D dissimilarity	2D dissimilarity	experimental inactives
number of citations <sup>34</sup>	782	366	106

**1.2. Bias in Chemical Data.** The term bias has several connotations and is often not used uniformly. In essence, bias describes the distortion from a true underlying relationship. Available chemical data are biased because experiments are conducted with different intentions than sampling the chemical space uniformly.<sup>35–37</sup> Chemical space is infinite, but the pharmacologically relevant space is estimated to comprise about  $10^{60}$  molecules.<sup>38</sup> The diversity of the synthesized subspace is biased due to known molecules and even *de novo* projects focus on molecules near the known active molecules.<sup>36</sup> There are legitimate reasons for excluding certain molecules from drug discovery projects for example costs, synthetic feasibility and availability in a library.<sup>37</sup> These reasons are comprehensible in drug discovery processes, but they prevent a uniform sampling of the chemical space. However, a nonuniform sampling does not mean that methods based on the available chemical data can not be used in practice, but it is important to consider the composition of the data in validation procedures and therefore in any benchmark data set. Otherwise it is not clear whether a method performs better because of a superior methodology or beneficial validation data.

Over time, several tendencies of bias in chemical data have been described. Cleves and Jain<sup>35</sup> presented general biases in chemical data as inductive bias. The authors showed that active ligands that are known today have been synthesized due to decisions of humans based on different assumptions, which may lead to advantageous performance of methods making the same assumptions. Those ligands are often synthesized based on their similarity to known ligands. They demonstrated that historically, known drugs for some targets show a noticeable 2D similarity in dependence of time, which they called 2D bias. Typically, actives for a specific target with high 2D similarity are patented in a narrow time span whereas more 2D dissimilar actives tend to be discovered years later. Consequently, for these ligands 2D methods have an artificial advantage over other methods.<sup>35</sup>

Another bias not specifically addressing the data composition has been described by Jain et al.<sup>39</sup> and is called confirmation bias. This bias is the tendency of a human to try to confirm a hypothesis by purely searching for a correlation with the outcome of the hypothesis. However, this correlation may not be physically founded and this approach can lead to false conclusions. For example, a model can be selected on the basis of correlation with some scoring function, but this scoring function might be based on assumptions that contradict the physical reality. An example from ligand-based VS would be the hypothesis that molecules similar to known active molecules are active as well, while 'activity cliffs' are not considered.<sup>39</sup>

In summary, there are several bias specifications describing certain scenarios of distorted data composition in the literature that contain patterns or signals that should not be learned by a model, because they misrepresent the true underlying distribution.

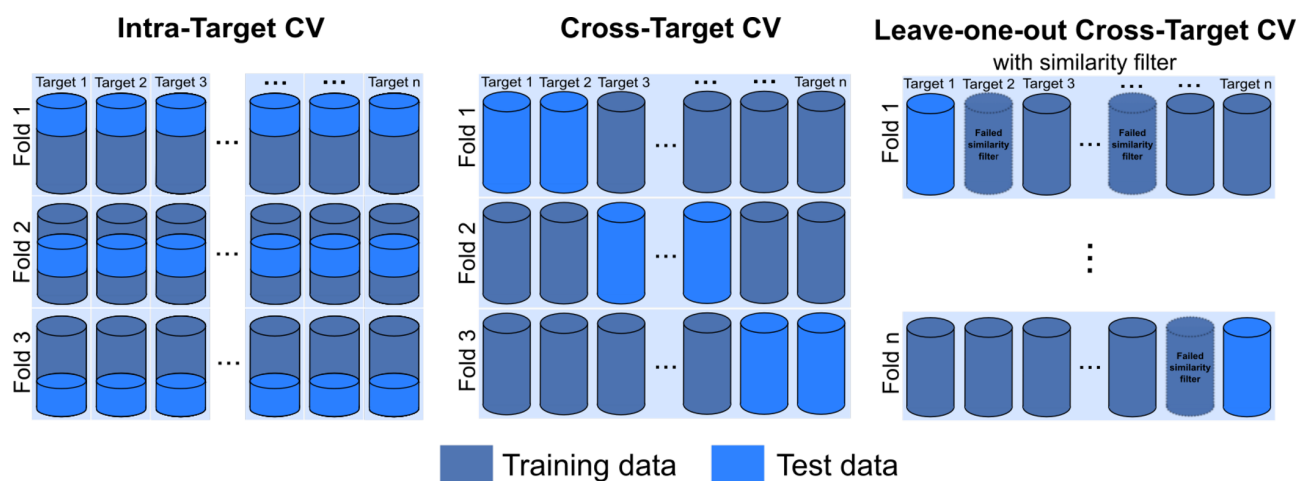
We will introduce another specification related to confirmation bias that in our opinion describes the worst kind of bias in a data collection. In particular, we distinguish domain bias and noncausal bias. Both falsely present prospective predictivity. On the one hand, domain bias distorts a prediction because the distribution of the sampled population resembles easy test cases or less diverse samples. The bias is based on biological mechanisms, such that the model is right for the right reason, but the applicability domain is narrow. An example for domain bias is when the train and test samples are too similar, for example, when a common scaffold is shared by the actives in the training and test set. Therefore, the measured model performance is biased to a certain domain, which is acceptable if modeling this domain is the aim, but is insufficient for generalization.

On the other hand, noncausal bias describes the case in which there is correlation but no causation. In this case, good predictions can be achieved by patterns in the data that do not represent any biological mechanism relevant for binding, but exhibit pure correlation with the labeled outcome. This bias yields a good statistic, but on the basis of fallacious models not based on physical reality. Such models do not work in general but only on data that fits the bias pattern, which makes them unusable for prospective predictions. Interestingly, there are reports of successfully finding leads based on *in silico* predictions, for which it has been experimentally refuted that the molecule binds for the predicted reason.<sup>40</sup>

In Section 4, we will show in detail on ML-based scoring functions from the literature that noncausal bias has been learned implicitly and unnoticed from established benchmark data sets.

**1.3. Review of Benchmark Data Sets in Context of Machine Learning.** ML methods are increasingly used in SBVS,<sup>12–17,41</sup> but to our knowledge there is no data set specifically dedicated to ML. Therefore, we review the purpose of established data sets and their limitations regarding ML methods in this paper.

A general distinction between benchmark data sets for SBVS and ligand-based VS is often made.<sup>20,32</sup> However, this differentiation might be misleading. A more accurate and fine-grained differentiation would be to categorize these data sets according to the methodology they have been designed for. Exemplarily, DUD and DUD-E are generated for docking with conventional scoring functions. Therefore, those data sets have been tailored for the requirements defined by those



**Figure 1.** Three cross validation (CV) scenarios used in the experiments are depicted schematically, namely intra-target CV, cross-target CV, and leave-one-out cross-target CV with a protein similarity filter. In the case of the first two scenarios three folds are exemplarily depicted.

methods and their vulnerabilities. The 1D properties matched between actives and decoys by the DUD and DUD-E protocols are captured by scoring functions, for example the hydrogen bond donor count in hydrogen bonds terms. However, classification should not be driven by simple and unspecific molecular features such as donor and acceptor counts, but rather by the quality of interactions. Simultaneously, the 2D dissimilarity between actives and decoys can be employed, because conventional scoring functions do not capture 2D features like molecular topology. Similarly, the MUV data set is not compiled for general ligand-based VS methods but for nearest neighbor similarity search that starts with a query of actives and does not use the 17 simple features but rather more complex descriptors like MACCS structural keys. Consequently, different methods and descriptors not considered in a data set's compilation protocol might be unsuited for the data set even if they are also by definition ligand-based or structure-based.

An example illustrating this problem are net charges in the first version of the DUD data set, which have not been included in the matched properties between actives and inactives.<sup>6</sup> While 42% of the ligands were charged, only 15% of the decoys had a nonzero charge.<sup>6</sup> Therefore, it has been possible to artificially increase performance by just assigning charged molecules as active. After overoptimistic results were reported due to differing charge distributions, some updated versions of DUD have been released.<sup>6,28,29</sup> Accordingly, for validation it is important to compare the features considered in the compilation protocol with the descriptor to validate. It might be not straightforward to spot whether an improvement in score after the addition of a feature is due to bias.

Another example for a limitation of transferability of DUD and DUD-E is the employed 2D dissimilarity. It is known and stated by the authors<sup>6</sup> of DUD-E that 2D descriptors are inappropriate for use with their data set. Simultaneously, the same is obviously true for DUD. Still there are reports using these descriptors on those data sets. The extent of distinctness by 2D features has been analyzed by Bietz et al.<sup>42</sup> by mining the most discriminative SMARTS-patterns in the DUD data set. It could be shown that, for example, for the AMPC target 80% of all ligands contain a sulfur atom and only 10% of the decoys. There are other examples in which simple patterns

such as the presence of single atoms can discriminate a noticeable portion of actives and inactives.<sup>42</sup> The 2D dissimilarity is expected, but it should be noted that the decoys can be easily distinguished according to very basic substructures, which might be relevant for the validation of novel descriptors.

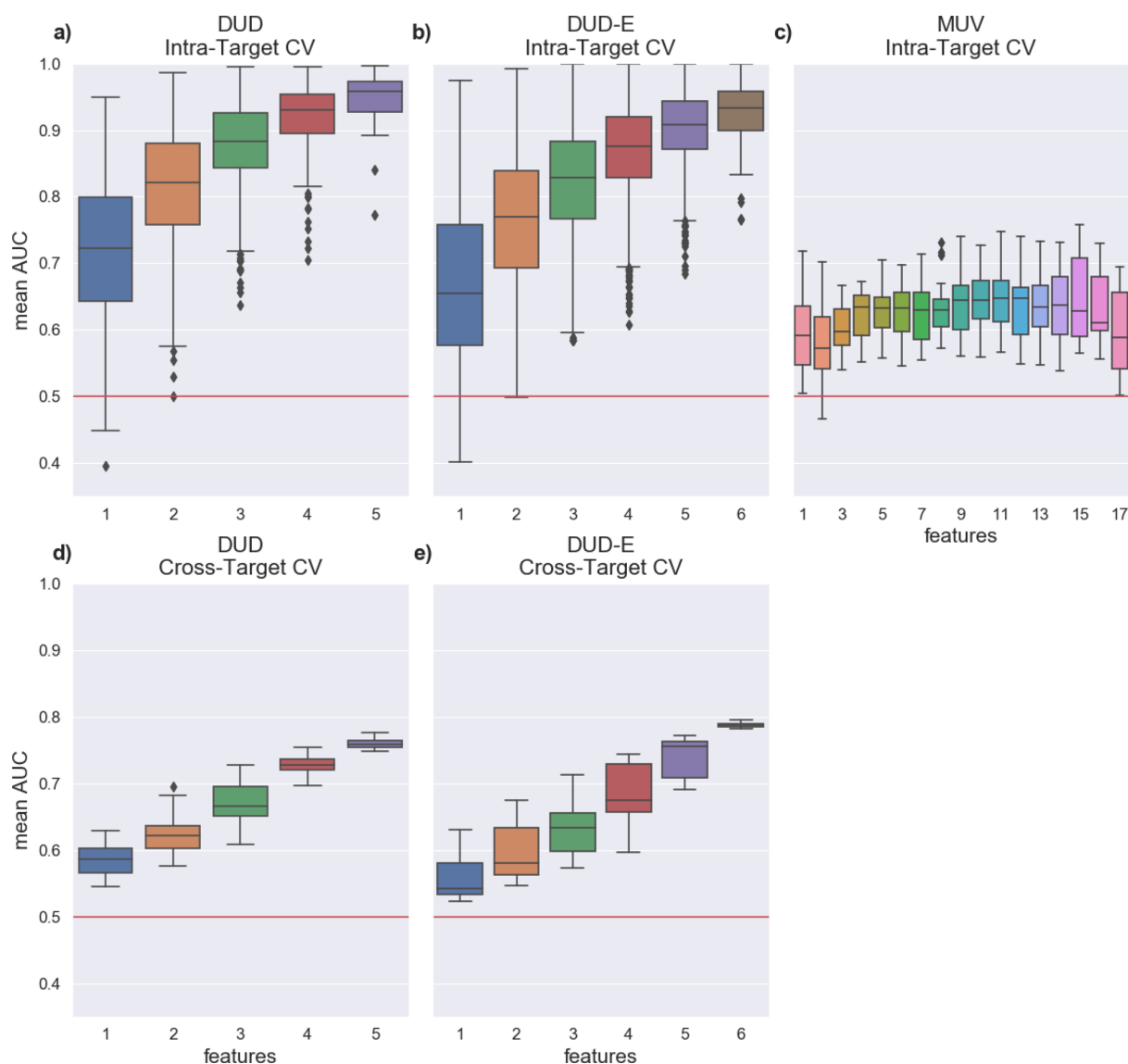
In summary, data sets have a design purpose focusing on a specific methodology. Data set design decisions are made based on the goal of the data set to provide good test cases for the targeted methodology. This might lead to bias when the data set is used with methods or descriptors that are different than the targeted methodology. Special care has to be taken, because there is a repertoire of nonlinear ML algorithms that can be paired with a large variety of chemical descriptors. For this reason we evaluate whether ML methods can be validated on established data sets. Concretely, we will evaluate the unbiasing techniques of DUD, DUD-E, and MUV in Section 3 and analyze noncausal bias learned from a subset of these data sets in Section 4.

## 2. METHODS

Experiments are conducted in Python using RDKit<sup>43</sup> for reading molecules and calculating molecular features. For ML experiments the libraries scikit-learn,<sup>44</sup> Keras,<sup>45</sup> and TensorFlow<sup>46</sup> were utilized. The charge-corrected DUD, DUD-E, and MUV data sets were downloaded from the respective web pages.<sup>47–49</sup>

On the one hand, classification performance of predictors is evaluated with the area under the receiver operating characteristic curve (AUC). This metric provides a value between 0 and 1, where 0.5 indicates a random guess.<sup>50</sup> On the other hand, VS is an early recognition problem, which we assess with the enrichment factor at the top  $x$  percent of the ranked predictions, for example the top one percent (EF1%).<sup>51</sup> Both measures are highly used in the field of VS and therefore should make the results of our experiments comparable to published classifier performances.

In the following experiments, three different cross validation (CV) experiments are performed, which are illustrated in Figure 1. In the first validation scenario, the set of molecules belonging to a single target  $t$  is split into training and validation sets. This validation procedure will be called intra-target CV



**Figure 2.** Results of the evaluation of unbiased features with AUC of DUD, DUD-E, and MUV with RF. The first row shows results of the intra-target CV for (a) DUD, (b) DUD-E, and (c) MUV. In the second row, results of cross-target CV are depicted for (d) DUD and (e) DUD-E.

and represents the task in VS of predicting further molecules that are active against a target based on known active and inactive molecules of this target. The second scenario is called cross-target CV. Here, the set of targets  $T = \{t_1, ..t_n\}$  is split into disjunctive training set  $T_{train} \subset T$  and test set  $T_{test} \subset T$ . With each target in  $T$  a set of molecules is associated. This CV represents a more challenging task for the method because the applicability domain is not restricted to a single target. Instead, the goal of this validation is to assess the predictive power across perhaps unrelated targets. There are different variations of cross-target CV. For example a more sophisticated version is leave-one-out cross-target CV with a similarity filter (see Figure 1). In  $n$  iterations each target  $t \in T$  is once used as the test set. To aggravate predictions, a similarity filter is applied before training to remove targets from the training set, which are similar to the test target. The aim of this CV is to assess predictive power across dissimilar targets.

### 3. EVALUATING UNBIASING TECHNIQUES

The unbiasing techniques applied in the protocols of DUD, DUD-E, and MUV are evaluated for their consistency with machine learning (ML) methods. In all three data sets, the unbiasing comprises the reduction of the discriminative power of simple features (see Table 1 for a list of features) by the compilation protocols. Since protocols address those features, a reasonable assumption would be that those features barely contribute to predictions. To examine this unbiasing with ML, learning models were trained and tested while using only these features for predictions. Since ML methods are effective at capturing patterns across multiple features, it is also interesting to evaluate combinations of features. Accordingly, in this experiment we put to test whether the predictive power of these features is reduced in the data sets when ML is used for prediction.

**3.1. Evaluation Setup.** First, the unbiased features were calculated using RDKit. In the case of DUD all  $\sum_{k=1}^n \binom{n}{k} = 31$



combinations of the 5 unbiased features were calculated. For DUD-E there are 63 feature sets and in the case of MUV 131 071 sets. Since for MUV the number of feature sets is too high, a greedy enumeration strategy was applied. A backward elimination<sup>52</sup> was employed to subsequently remove features from the whole unbiased feature set. Initially, the set of MUV features  $F_f$ , where  $f$  dedicates the number of features in the set, contains  $f = 17$  features. The feature set  $F_{17}$  was used to train and evaluate models with cross validation. Then all possible sets of  $F_{f-1}$  were evaluated in the same way and the single feature contributing the least to the performance in terms of AUC in the cross validation was eliminated from the feature set. The process was iteratively repeated until  $f = 1$  and the highest performing feature was determined. The strategy for feature subset enumeration is analogous to the enumeration used in feature selection tasks with wrapper methods.<sup>52</sup> In feature selection the goal is to reduce the number of features and select the best subset of features relevant to the learning task.<sup>52</sup> In contrast, the aim in this experiment is to identify whether and to what extent unbiased feature subsets of a benchmark data set perform with a given learning algorithm.

For evaluation, intra-target and cross-target CV were utilized (see Figure 1a,b). All CV experiments were conducted with 10 random folds. The folds of intra-target CVs were stratified to preserve the ratio of the samples for both classes in each fold.

As reference ML methods, Random Forest (RF) classifier<sup>53</sup> and logistic regression (LR) from scikit-learn were used with default parameters except for RF, for which the number of estimators was set to 400. LR was selected because it is a simpler linear model. In contrast RF is a nonlinear method and is considered comparatively robust against overfitting and easy to parametrize.<sup>54</sup> Furthermore, RF is a widely used method in the field of VS and drug discovery.

For these experiments, training and test sets were scaled column-wise on the basis of the respective training set by the formula  $z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}$ , where  $i$  denotes the row and  $j$  the column in the feature matrix. Accordingly, from each single feature value  $x_{ij}$ , the mean  $\mu_j$  of the column is subtracted and divided by the standard deviation  $\sigma_j$ .

**3.2. Evaluation Results.** The results of intra-target CV with RF and LR evaluated with the AUC are presented in Figure 2a–c and S1a–c. Plots showing the performance assessed with the EF1% is shown in Figure S2a–c for RF and for LR in Figure S3a–c. Each boxplot shows the results of all targets of the respective data set. Hence, one point in the plot represents the mean performance of a method on a single target in 10-fold CV with a certain set of unbiased features. Consequently, a box in the plot shows the range of performance in AUC or EF1% over all targets when exactly  $x$  features are used for prediction.

The results of cross-target CV with AUC are depicted in Figure 2d and e for RF as well as in Figure S1d,e for LR. Results with EF1% are shown in Figure S2d,e for RF and Figure S3d,e for LR. In this experiment, random cross-target CV has been repeated ten times with different random splits to address empirical stability of the results. One point in these plots represents the mean performance of a method on a 10-fold cross-target CV with a certain set of unbiased features. Therefore, a box in these plots depicts the distribution of mean AUC or EF1% values of differently splitted cross-target CVs when  $x$  features are used.

We did not perform cross-target CV on MUV because of the high number of duplicates in the set of inactives of different targets. Of the 255 510 molecules in the whole MUV data set there are only 95 916 unique PubChem-compound-IDs. Deduplication would leave only 38% of samples and would yield an arbitrary composition not representing the MUV anymore. This redundancy is probably due to the fact that the experimentally analyzed space is generally small, but negative results are even less often reported.

The AUC results of intra-target CV are similar for DUD and DUD-E in Figures 2 and S1, which is comprehensible since both data sets have been generated with a similar strategy. A noticeable observation is that the predictive performance achieved by many of the models is rather high. Comparing RF and LR in these experiments shows higher median and maximum values for the nonlinear RF, which was expected. The highest achieved mean AUC with RF is 1.0 for both DUD and DUD-E, which indicates perfect performance. With LR the maximum mean values are 0.95 and 0.99 for DUD and DUD-E, respectively, which also shows very good classification of molecules. An interesting observation is that AUC values increase when more features are included.

Evaluation with EF1% shows also very good performance on DUD and DUD-E. When AUC results are compared to EF1% a similar correlation of the number of features and performance can be observed for RF (see Figure S2a,b). In contrast, for LR this effect is only partly present on the DUD-E data set (see Figure S3a,b).

The predictive performance on MUV in the intra-target CV is substantially lower as on the other two data set, but still substantial values are achieved. The maximum AUC with RF is 0.76 when using 15 different features, while the best AUC with LR is 0.88 when seven features are used. In the case of MUV, combining features has no observable correlation with the performance.

When considering EF1% on MUV, the results show noticeable enrichment for both RF and LR. It is worth mentioning that for this experiment not all feature combinations were enumerated on MUV, but a backward elimination was employed, which was guided by the AUC metric.

Generally the performance with cross-target CV is lower than with intra-target CV. Still, a noteworthy separation can be achieved in this validation setup. Especially, RF achieves AUC values up to 0.78 and 0.80 on DUD and DUD-E, respectively. In contrast, when LR is utilized the best AUC values are 0.63 for DUD and 0.58 for DUD-E. The results of RF assessed with AUC and EF1% show the same correlation of the number of features with the performance as in the intra-target evaluation. This trend is also observable for most LR results, but not as prominent.

**3.3. Evaluation Discussion. 3.3.1. DUD and DUD-E.** For the interpretation of the results of DUD and DUD-E it is important to consider models trained on a single feature and models trained on combinations of features separately. Since distributions of single features are matched between the classes by the compilation protocol by approximating mean and standard deviation it was expected that they contain not much information for discrimination and single feature-based predictions would be close to a random guess. However, the performance achieved with single features is far from an AUC of 0.5 with RF for most targets and also the LR performs very well. To analyze these results it is useful to examine the distributions of single features from both classes. In the Figures

S4 and S5 the distributions of the matched features for the DUD and DUD-E are depicted. These plots show the distributions over the whole data sets (over all targets). For both data sets, the histograms are mostly overlapping and the properties seem to be well matched, except for the molecular weight in DUD-E for which proportionally more actives are present from a molecular weight of approximately 500 Da. Consequently, it would be expected that molecular weight can be used to discriminate classes on DUD-E.

To explain the high performance with single features, we plotted the feature distribution of the target with the highest AUC in intra-target CV on DUD with RF. It can be seen at the distribution of LogP on the target PNP in Figure S6 (first row) that features are not as well matched as over the whole data set. In this example, a notable subset of actives has lower or higher LogP values than the decoy set. For further examination of results, we plotted the LogP histograms of the training and test sets of a single fold from intra-target CV in the second and third row of Figure S6. The fourth row of the plot shows the distribution of scores of the LR model in an one-dimensional contour plot over a range of LogP values. Test actives are marked by red triangles and test decoys by blue triangles. As can be seen from the blue color of the contour plot all test samples get scores near zero. This could be expected from the histograms since there is no LogP threshold, which could adequately separate the classes. In contrast, RF achieves almost perfect test performance with an AUC of 0.99 on this split. This can be explained with the last row of Figure S6, which shows the contour plot of scores from the RF model. There are many intervals with different scores, which was expected. It can be seen from the triangles marking the test actives that for each test active there exists an interval that corresponds to a higher than zero score. However, very low scores are predicted for almost all decoys of the test set. Therefore test actives get higher scores than the decoys, which is sufficient to classify them correctly. The performance of RF can be explained by the fact that the LogP values are matched in certain intervals only. Since predictions of RF are based on splitting the input space into intervals this unbiasing is ineffective, when the test set matches the intervals learned from the training set. In conclusion, the LogP values of actives and decoys in this example are well enough matched for removing bias for linear models, but not for a nonlinear model as RF. However, this does not explain the still very high AUC values achieved by LR models in Figure S1 when a single feature is used.

The high AUC values of LR model's when only a single feature is used can be explained by looking at Figure S7. In this example, the intra-target CV experiment on the target SAHH of DUD-E with the number of hydrogen bond acceptors as feature is shown. This feature is not well matched between classes. For this target, on average actives contain more acceptors than decoys. This distribution is still present after the random split into training and test set. For this reason, the LR model can learn a well separating threshold as depicted in the contour plot in Figure S7. For this example, LR and RF have the same performance in AUC because most molecules are easy to classify.

As mentioned before, an interesting observation is the correlation of the number of features and the performance, which increases until AUC values close to 1.0 are achieved for most experiments. This is not surprising since ML can capture synergies between subsets of features. However, separability is extremely high. This is probably because in the unbiasing one-

dimensional feature spaces are matched, but not the multi-dimensional spaces. When considering a single target high performance with multiple features is nothing that needs to be avoided at any cost, because activity also depends on nonadditive molecular features.<sup>39</sup> However, when perfect performance is already reached with molecule features alone, a structure-based method might be strongly biased because it uses both the molecule and the protein. Therefore, in our opinion for DUD and DUD-E multidimensional unbiasing is necessary, if they are used for benchmarking of ML methods.

Interestingly, the same correlation can also be observed in the cross-target setup. This CV should be less prone to bias coming from molecular similar actives for example originating from the same molecular series. Correspondingly, the AUC values are lower as in the intra-target CV. However, the resulting AUCs are still reaching values of 0.78 and 0.80 for DUD and DUD-E, respectively, even though all predictions are based on molecular features only. An explanation for this is that we performed a random cross-target CV and the test and training actives of different targets might be similar because the targets could be related. This is further evaluated in Section 4 on the more stringent leave-one-out cross-target CV on DUD.

3.3.2. *MUV*. In contrast to DUD and DUD-E, the AUC values achieved for MUV are substantially lower (see Figure 2c). An explanation for this is that the embedding of actives among inactives in the 17 dimensional feature space with methods from spatial statistic removes more bias from the data set than approximating mean and standard deviation of single features of actives and inactives independently. This also explains that no synergistic effect by combining features is observed. In comparison with DUD and DUD-E, the unbiasing protocol of MUV seems to be more suited for ML methods, but still there are serious limitations. For example, the linear LR reaches higher maximum AUC values than RF. This is probably explained through overfitting of the RF models, which we did not investigate further. Interestingly, Wallach et al. showed for the ECFP that the suitability of MUV for ML is restricted by the MUV bias function, which considers the relation between actives to actives and actives to inactives in the feature space, but not inactives to actives and inactives to other inactives.<sup>37</sup> They proposed their own bias function called AVE<sup>37</sup> to overcome these limitations. However, it is comprehensible that MUV is not a perfect fit for ML because it was designed for similarity search starting from a query active rather than a training set of active and inactive molecules.

3.3.3. *AVE Analysis*. In an additional experiment, we evaluated the AVE-bias score proposed by Wallach et al.<sup>37</sup> on the intra-target CV experiments shown in Figures 2 and S1–S3. AVE is an extension of the MUV scoring function and assess the redundancy between training and test set. The results are shown in Figures S8–S10. AVE values different from zero indicate bias. For DUD, there is a notable correlation of AUC with AVE using RF but much less in LR and EF1% experiments. AVE bias in these experiment is not surprising because there was already substantial MUV bias reported for DUD.<sup>4</sup> For DUD-E, there is a notable correlation between AVE and AUC in RF and LR experiments and moderate correlation with EF1%. The correlation on MUV is weak for RF and moderate for LR experiments. Furthermore, the points in the plots are colored according to the number of features used (corresponding to the number of features on the *x*-axis in Figure 2). For DUD and DUD-E, it is observable that when more features are included also more AVE bias is

exhibited. However, with most single features the performance is very high, but no AVE bias is present. In conclusion, there is a noteworthy correlation in DUD and DUD-E between performance and AVE, but in a substantial part of the experiments AVE can not explain the high performance.

**3.3.4. Conclusion.** In summary, DUD and DUD-E have been developed to evaluate conventional scoring functions for docking. With this goal, the unbiasing technique employed tries to approximate the distribution of single low dimensional features between actives and inactives. As our results demonstrate for a nonlinear ML method like RF it is not sufficient to select decoys to match mean and standard deviation of these low dimensional features. RF is able to accurately separate classes even on the basis of a single feature and especially when multiple low dimensional features are combined. Even LR is able to achieve impressive results, because for some targets features are not matched well enough. Moreover, experiments on MUV show also substantial performance when unbiased features are used with RF and LR, which is in accordance with others.<sup>37</sup>

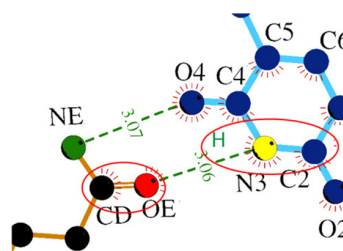
Therefore, we can conclude that when the method is exchanged the unbiasing of the data set might be ineffective and molecular features alone might still play a major role in the validation of structure-based approaches, when ML is used. As a consequence, when using these data sets with ML methods, it should be kept in mind that low dimensional features are sufficient to achieve a notable performance in separating classes in the intra-target CV. Also a noteworthy separation can be achieved in the cross-target CV where predictions for targets dissimilar to the targets in the training set are made. These baseline performances should be considered when more complex descriptors are used which include those features. Finally, the performance should be compared to those baselines instead of random performance.

#### 4. NON-CAUSAL BIAS IN LITERATURE

In the following, two examples of convolutional neural networks (CNN) for the scoring of protein–ligand complexes after docking are presented from literature. We made an observation that indicates that even after elaborate and conscientious validation by the authors bias might be a problem. In the subsequent examples, we first introduce the networks and their validation procedures from literature. Then the experiments we conducted to examine our observations are described.

**4.1. Example 1: DeepVS.** DeepVS is a CNN inspired by natural language processing.<sup>14</sup> The structure-based approach uses a novel descriptor to featurize and vectorize 3D protein–molecule complexes after docking. The aim is to learn general protein–ligand interactions from basic features in the local neighborhood of atoms of the small molecule in the 3D complex, called atom contexts.<sup>14</sup>

The DeepVS descriptor is depicted in Figure 3. For each atom  $a$  in the ligand molecule, the local neighborhood is considered. The neighborhood is described by the distances, atom types, atomic partial charges, and associated protein residues of the  $k_c$  atoms in the ligand molecule nearest to  $a$  and the  $k_p$  atoms in the protein nearest to  $a$ . For example, in Figure 3,  $k_c$  is three and includes N3 (atom  $a$  itself), H, and C2, as indicated by the red circle. In the same example,  $k_p$  is 2 and includes the protein atoms CD and OE. These discrete values are transformed into real-valued vectors, which constitute the first hidden layer of the network. The network consists of the



**Figure 3.** This figure, taken from Pereira et al.,<sup>14</sup> illustrates the DeepVS descriptor on the example of the local atom context of atom N3 (yellow) of thymidine in complex with a thymidine kinase (PDB-ID:1KIM). The parameters  $k_c = 3$  and  $k_p = 2$  are indicated by the two large red circles.

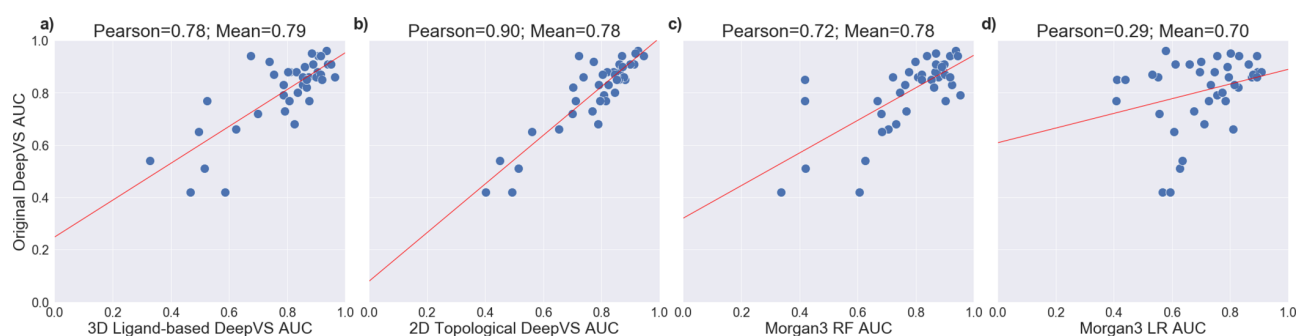
first hidden layer, a convolutional layer (second hidden layer), a third hidden layer and an output layer with a softmax classifier.<sup>14</sup> In the final DeepVS network, the hyperparameters  $k_c$  and  $k_p$  have been set to 6 and 2, respectively. The training was performed on minibatches of size 20 with stochastic gradient descent (SGD), negative log-likelihood as loss function and backpropagation.<sup>14</sup>

DeepVS has been validated on the DUD data set in a leave-one-out cross-target CV (LOO–CV) with a similarity filter as illustrated in Figure 1. In each iteration of the LOO–CV, one of the  $n = 40$  DUD targets is left out as a test set. From the remaining 39 proteins, all that are similar to the selected one are discarded and the rest form the training set. Similarity of proteins has been described as sharing the same protein class or showing a positive cross-target enrichment in the original DUD paper.<sup>5</sup> Each trained model has been used to make predictions for the respective test protein. The validation revealed a mean AUC of 0.81 and an enrichment factor at 2% (EF2%) of 6.62 in the LOO–CV, which outperformed several other scoring functions.<sup>14</sup>

One experiment in the validation of DeepVS raised our attention. It is reported by the authors that setting the parameter  $k_p$  to 0 yields an AUC of 0.80 and EF2% of 6.95 (Table 6 in Pereira et al.<sup>14</sup>).<sup>14</sup> By setting  $k_p = 0$ , all explicit protein information is removed from the descriptor. The remaining descriptor considers only small molecule atoms, but the resulting AUC drops only by 0.01 and the EF2% marginally increases by 0.33, therefore, it seems DeepVS is invariant to the information from protein atoms. The descriptor with  $k_p = 0$  corresponds roughly to a ligand-based approach. With the exception of the molecule conformation which has been generated through docking, no information from the protein is contained in the descriptor. Ligand-based approaches are based on finding active molecules due to their similarity to known ligands. Since targets similar to the test target are removed before training, it is not expected that any ligands of the training targets are structurally similar to the test ligands. Therefore, the predictability based on ligand similarity should be low. However, the achieved prediction performance is almost unchanged, which indicates noncausal bias.

**4.2. Evaluating DeepVS.** To understand these results, we reimplemented the DeepVS network in TensorFlow and performed the same validation experiment with altered input data. Since the input to the original DeepVS was docking output, the small molecules have a binding site specific conformation. To remove this protein-dependent information from the experiment, we generated small molecules con-





**Figure 4.** Correlation plots of AUC values of the structure-based original DeepVS (values taken from ref 14) and the four other approaches. (a) Performance of our 3D ligand-based reimplement. (b) Correlation of the reimplement using the topological distance on the molecular graph instead of 3D distances. Finally, the performance of (c) RF and (d) LR with Morgan3 fingerprints is plotted against the original DeepVS.

formations with RDKit, not considering the individual protein structure. This yielded a purely ligand-based descriptor.

The results of our ligand-based reimplement are plotted against the reported folds of the original DeepVS in Figures 4 and S11. The achieved mean AUC of the reimplement is 0.79, as shown in Figure 4a and the mean EF2% is 5.03 (see Figure S11a). ROC curves are provided in Figure S12. On the basis of these results, it seems that the overall performance of the DeepVS approach is in fact mostly invariant to protein information. Although, for both scores the ligand-based results correlate with the original structure-based DeepVS, it can be observed that the results in AUC are more similar than the achieved EF2% values. For single folds the simple molecule conformations of the ligand-based version suffice to achieve higher early enrichment while for other folds using docking poses and protein information is advantageous. However, the high performance of the ligand-based version in the LOO–CV implies noncausal bias.

Since small molecules only are sufficient to discriminate actives from inactives across dissimilar targets, there must be a discriminative noncausal molecular property across the whole DUD data set. To ensure that decoys are not actually active molecules, a certain minimum topological distance from every decoy to every ligand in the whole data set was required when the data set was designed.<sup>5</sup> For this reason, 2D molecular features should be discriminatory across all DUD targets.

By examining the DeepVS descriptor, it becomes obvious that the descriptor is able to capture 2D topology. In Figure 3, the red circle on the right containing the small molecule atoms shows the local atom context of N3 in 3D space. Indeed, the red circle also marks the local substructure around atom N3, because the nearest atoms in 3D space often correspond to the nearest atoms in the molecular graph.

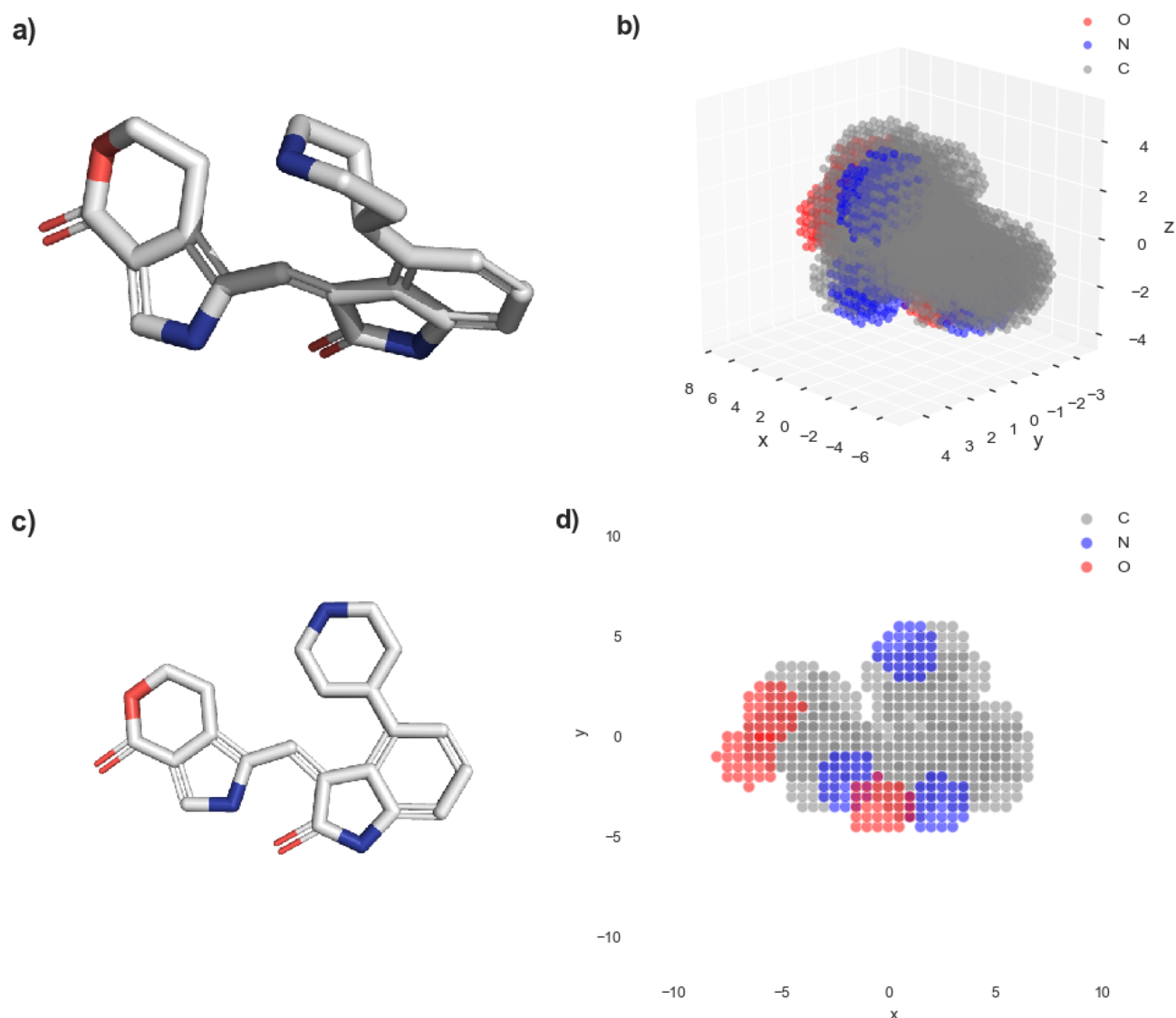
To compare the local neighborhood of an atom in 3D space with the topological neighborhood, we conducted another experiment. Using our ligand-based reimplement, instead of considering the  $k_c$  nearest atoms in 3D space we considered the  $k_c = 6$  nearest atoms in the molecule graph, yielding a topological version of the ligand-based DeepVS. The resulting mean AUC and EF2% over all 40 folds of the topological DeepVS is 0.78 and 5.41, respectively (see Figures 4b and S11b). ROC curves for the 40 folds are shown in Figure S13. A comparison of AUC and EF2% values between our 3D and 2D reimplementations is shown in Figure S14, which indicates a strong correlation in terms of AUC and a good correlation for EF2% values. To further evaluate whether the 3D descriptor captures the 2D information, we examined if the same actives

were enriched by both methods. In the first 5% of the ranked lists, 81% of predicted actives are identical between both methods over all 40-folds with a standard deviation of 18%. A full distribution of the active identity is depicted in Figure S15a. These experiments demonstrate that similar results can be achieved when using exclusively 2D information.

Another open question is to determine to which extent the usage of a CNN contributes to the performance and how standard ML approaches would perform. To examine the baseline performance of a 2D descriptor, we applied Morgan3 fingerprints folded to 4096 bits with RF and LR in the same validation setup as used for DeepVS. The resulting mean AUC is 0.78 for RF and 0.70 for LR (see Figure 4c,d), while the resulting mean EF2% are 5.52 and 5.47 for RF and LR, respectively (see Figure S11c,d). On the one hand, these results show that also other nonlinear ML methods such as RF are sufficient to achieve a comparable performance and using a CNN does not improve the prediction results significantly. On the other hand, when using a simple linear LR model the mean AUC is already quite high with 0.70, which shows that correct classification can be achieved with a linear function for many test cases. As in the previous case, we compare the hit lists to see whether the same actives are enriched by conventional ML methods as with the CNN. At the first 5% of the hit lists on average 46% (28% standard deviation) of actives are identical (see Figure S15b for details). Therefore for most targets (folds) there is a difference between the information captured and used by RF with Morgan3 fingerprints and the 3D DeepVS model but still a comparable performance can be achieved and ligand features suffice to make predictions across dissimilar targets.

**4.3. Example 2: Grid-Based 3D CNN.** For our second example, we wanted to examine a currently frequent approach of ML-based scoring functions using a CNN inspired by image recognition. In these approaches the 3D complex of the protein and ligand is discretized with a grid around the binding site. There are multiple examples, such as AtomNet,<sup>13</sup> an unnamed network by Ragoza et al.<sup>15</sup> and  $K_{\text{DEEP}}$ .<sup>17</sup> As in the case of image recognition, the intention of these networks is to automatically and hierarchically abstract high-level features holding information for binding from low-level features of the complex. Because of the workload required to rebuild and re-evaluate these networks, we decided to examine the CNN by Ragoza et al. only because the authors provided an elaborate validation from different perspectives and shared their code.

Ragoza et al. discretize the protein–ligand complex with a uniform 3D grid of 24 Å centered around the binding site. The



**Figure 5.** Illustration of the two variants of the grid descriptor on the example of compound CHEMBL58224. The first row depicts a (a) 3D structure of the molecule and (b) corresponding representation of the 3D grid descriptor. In the second row, the (c) 2D conformation of the molecule and (d) according 2D grid representation are illustrated. In panels b and d, the coloring of the boolean-valued grid points is overlaid for the illustration of the channels of the grid.

resolution is set to 0.5 Å. A grid point stores information about heavy atom types represented by a continuous density function depending on the distance of an atom to the grid point and the respective van der Waals radius. Each channel of the grid resembles a different atom type (like RGB channels in images). Atom types for protein atoms and ligand atoms are considered separately in different channels. In total 34 distinct atom types are considered. These include simple elements of atoms, such as nitrogen, oxygen or sulfur. In addition aliphatic and aromatic properties of carbons are considered as well as nitrogen and oxygen atoms acting as hydrogen bond acceptors or donors.<sup>15</sup>

The architecture of the CNN consists of three subsequent pooling and convolutional layers and a final output layer. The input to the network is the featurized 3D grid of the binding pocket. The final scores are provided with the softmax function.<sup>15</sup> See Ragoza et al.<sup>15</sup> for a more detailed description of the architecture. Ragoza et al. performed the training of the

network with SGD and backpropagation while minimizing the multinomial logistic loss. Oversampled batches of size 10 have been utilized such that each batch is balanced according to the number of actives and inactives. In addition, training data have been augmented by random rotation and translation.<sup>15</sup>

The described CNN architecture has been evaluated comprehensively for the tasks of pose prediction and VS on CSAR, DUD-E and independent test sets.<sup>15</sup> CSAR and DUD-E have been used in cross-target three-fold cluster cross validations (cCCV). For each of the two data sets, all targets were clustered into three-folds to ensure that targets with sequence identity greater 90% for CSAR and 80% for DUD-E are in the same fold. This should prevent the training targets from being too similar to test targets.<sup>15</sup>

The achieved results for pose prediction and VS differ. Pose prediction with cCCV on CSAR yielded a mean AUC of 0.815 outperforming AutoDock Vina (Vina). In contrast, in an intra-target validation, Vina outperformed the CNN or is almost as

good. For the VS task, the best reported mean AUC with the CNN in the cCCV on DUD-E is 0.86. On the test, Vina achieved an AUC of only 0.68. Interestingly, Ragoza et al. showed that a model trained with DUD-E performed rather poor on CSAR test data and vice versa. The authors evaluated models trained on combinations of DUD-E and CSAR data. The combined model exhibits an AUC of 0.83 on the VS task and 0.79 at the pose prediction task, showing slightly worse performance as the networks trained for a single task. For this reason Ragoza et al. conclude that the results demonstrate that the data sets generated for different tasks prevent models from learning a similar scoring function.<sup>15</sup>

**4.4. Evaluating Grid-Based 3D CNN.** As in the first example from literature, a CNN was trained for the scoring of docking output. The evaluation was performed on the DUD-E data set, which is compiled with a comparable strategy as the DUD data set. Interestingly, Ragoza et al. noticed that learning with data sets for different tasks leads to models with differing scoring functions.<sup>15</sup> A reason for this might be bias learned from the DUD-E data set. Ragoza et al. already suspected overly optimistic predictions in the case of their DUD-E experiments, which should be mitigated through cCCV.<sup>15</sup> The intention in utilizing the cCCV was that the targets used for training are not similar to the test targets. As in the case of DeepVS, the performance of the CNN could be due to small molecule information only.

To examine this possibility, we reimplemented the described CNN in Keras with some adaptations to the descriptor and replicated the cCCV experiments described by Ragoza et al.<sup>15</sup> First, as in the reimplement of DeepVS, protein information was left out completely. No docking was performed. Instead, conformations of the small molecules in DUD-E were generated using RDKit. The molecules are then put into a 3D grid of  $48^3$  grid points with a spacing of 0.5 Å as the original descriptor. The second adaptation was that for the reimplemented descriptor not all 34 atom types were used. For computational efficiency only the elements bromine, carbon, chlorine, fluorine, iodine, nitrogen, oxygen, phosphorus and sulfur were used. This reduced the channels of each grid from 34 to 9. The third adaptation made to the descriptor was that no density function representing the atoms was utilized. Instead, a boolean function was applied, which sets a grid point to 1 if this grid point is in the van der Waals radius of an atom. This adapted descriptor is a simpler version, capturing only a subset of the features of the original descriptor. The descriptor is illustrated in Figure 5. The 3D conformation of the molecule (Figure 5a) is discretized by the grid descriptor. The descriptor for the depicted conformation is shown in Figure 5b, where the different channels (O, N, and C) are overlaid.

With the modified descriptor, the reimplemented CNN was trained and evaluated. The three-folds for the cCCV on DUD-E were generated using the Python script<sup>55</sup> provided by Ragoza et al.<sup>15</sup> The resulting AUC values achieved with the simpler descriptor and the reimplemented CNN are 0.82, 0.84, and 0.85 for the three-folds, which results in a mean of 0.84. Therefore, the difference to the original CNN model trained on DUD-E is 0.02, which is very similar to the performance of the CNN of the original publication using docking. In terms of early enrichment EF2% values of 12.60, 14.87, and 13.98 were achieved for the three folds. ROC curves are shown in Figure S16. The experiment here is ligand-based; therefore, predictions across dissimilar targets should not be possible on the basis of ligand similarity. The still high AUC values

strongly indicate that a similar bias is learned as in the example of DeepVS. As with DUD, it is possible for DUD-E to learn activity prediction across targets with small molecule information only.

As in the case of DUD, molecular topology is a discriminative feature across the DUD-E data set.<sup>6</sup> To examine if this is the reason for the ligand-based results for this CNN, we conducted a 2D version of the original 3D experiment using exclusively a 2D description of the molecules. Instead of generating 3D molecule conformations, we generated only 2D depictions (see Figure 5c) using a constant atom radius of 1.5 Å for all atoms and calculated their respective 2D grid representation as shown in Figure 5d. The 2D grid builds the input to a 2D version of the CNN. We performed the same cCCV and achieved AUC values of 0.82, 0.85, and 0.84 for the three-folds, yielding a mean of 0.84. The achieved EF2% values are 11.70, 15.41, and 11.31. ROC curves are shown in Figure S17. To compare whether the same actives were enriched with the 3D and 2D version of the CNN, we compared the hit lists from both methods regarding the actives. On average, 73% (1% standard deviation) of the actives are identical at the first 5% of the hitlists. The full distribution of identical actives can be found in Figure S18. These results indicate that a substantial part of the performance of the original grid-based CNN is, such as in the case of DeepVS, based on the difference across all DUD-E targets of lower dimensional molecular features included in the molecules' topology.

## 5. TOWARD BIAS-CONTROLLED VALIDATION

A comprehensive and elaborated validation for a VS method is the basis for reliability, acceptance, and usage of this method in the scientific community. Elaborate efforts are made to validate methods, but as we showed in the previous section on two examples from literature, it is possible to achieve comparable results when a ligand-based version of a structure-based descriptor is used. Simultaneously, the performance in those experiments is noncausal bias, because ligand similarity should not suffice to make predictions across dissimilar targets.

A reason this bias remained unnoticed is the non-transparency of the used ML models. In particular, deep learning models such as CNNs are difficult to interpret and are often treated as black boxes. This lack of interpretability increases the effort required for validation as well as the necessity for a comprehensive validation. In the two examples, elaborated validation experiments have been conducted by the authors.<sup>14,15</sup> Still, as we have shown, substantial noncausal bias remained unrecognized.

We demonstrated in detail for DeepVS that molecular topology is a discriminative feature in the used validation setup and is captured implicitly by the DeepVS descriptor. Our results strongly indicate the same reason for the bias influencing the performance measuring of the grid-based 3D CNN. In both cases, discriminative lower dimensional features of the small molecules were contained in a nontransparent structure-based 3D descriptor. This is difficult to spot, and an apparently reasonable model is actually learning noncausal bias, especially, when difficult-to-interpret neural networks are employed. Our experiments with standard ML methods showed that the difference in performance between RF and CNNs is very small, which makes it debatable whether benefits of CNNs outweigh the additional effort and the lack of interpretability that is associated with deep learning models. This also shows the importance of performing baseline

experiments and comparing complex methods with simple and interpretable ones.

That lower dimensional features are biasing higher dimensional descriptors is a recurring problem that has already been addressed in the context of simple properties and conventional scoring functions in DUD and DUD-E as well as for similarity search in MUV. However, the increasing trend of applying ML methods in SBVS comes with a variety of novel descriptors and expressive methods that can abstract higher dimensional features from low dimensional ones. This holds the risk of learning bias, because the currently established data sets focus on different methodologies. When comparing the ML methodology on the basis of DUD and DUD-E it is evident that those data sets have not been designed with ML methods in mind. Conventional scoring functions such as empirical scoring functions are typically weighted sums over physically motivated descriptors.<sup>41</sup> Current ML methods operate differently. They are strongly data-driven, make use of a large number of free variables, and are able to derive nonlinear relationships. While this is an advantage in general, it leads to a higher risk of learning noncausal bias from data sets. In particular, the protocols of DUD and DUD-E assume that simple 1D properties might cause bias. However, as the results of Section 3 show, in contrast to MUV, the combination of 1D features for example by a ML method has not been considered in the unbiasing protocol of DUD and DUD-E and it is arguable if combinations of these features should have a strong influence on structure-based descriptors. In addition, the design decision to employ a 2D dissimilarity of each decoy to any active of the whole data set generates a strongly discriminating feature and could be employed because 2D features are not captured by conventional scoring functions. The authors of DUD-E therefore state that their data set is not suited for topological or 2D methods because "Through its construction, ligands light up against DUD-E decoys using these 2-D similarity methods, which create an artificially favorable enrichment bias for them."<sup>6</sup> This design decision restricts the applicability of DUD and DUD-E for 2D methods, but not in general. However, what should be more emphasized is that the 2D dissimilarity is employed across all targets of the data sets, which can artificially enable the differentiation of actives and inactives on the basis of molecules from biologically unrelated targets.

In conclusion, whether bias occurs in the validation of a method depends on the composition of the data set used as well as the method and descriptor used on the data set (and other factors such as the performance metric and the strategy for splitting into train/test sets).

In our studies, we focused on three well established data sets in VS, but there are other established and also more recent data sets, which should be evaluated for their suitability with ML<sup>8-11,56</sup> as well before similar problems arise from different data sets.

This discloses two current problems in VS with ML. First, method developers need to evaluate the appropriateness of a benchmark data set for their method-descriptor combinations. Second, there is a need for a benchmark data set suited for ML methods in VS. To both, we provide some first guidelines to foster the discussion in the scientific community.

## 6. GUIDELINES FOR VALIDATION EXPERIMENTS

We suggest five guidelines for the setup of validation experiments for ML in VS, which are not necessarily restricted

to these methods. This includes determining if a data set is suitable for validating a particular method and descriptor, as well as identifying implicit bias.

(i) Validation domain of a data set: If a new method is in place, a data set is selected for validation. In general, it is very good to choose established data sets since they were assembled independently from the method development. It is not sufficient, however, to just consider the application problem. The data set must be suited for the method applied. Some authors of data sets already give hints in their publications. For example, DUD and DUD-E are known to be not suitable for methods capturing topological descriptors.

(ii) Method and descriptor design: More detailed evaluations can be conducted if the method and descriptor are modular. Particularly, components of the descriptor and method should be easily controllable in their information content and ideally exchangeable. This is beneficial to evaluate different descriptors and method variations. A positive example for a modular descriptor is the DeepVS descriptor because it allows to control the extent of information from the protein and from the small molecule.

(iii) Data set's unbiasing strategy: As observed from the compilation protocols of DUD, DUD-E, and MUV, each data set applies some unbiasing techniques to reduce the predictive power of certain features. These features are considered biologically irrelevant in that they should have not much influence on the distinction of actives from inactives. Whether an unbiasing technique of a data set compilation protocol works with a certain method needs to be validated. Evaluating these unbiased features and their combinations with a particular method, as in the experiment in Section 3, is important because perceptible predictive power in this experiment indicates that the data set's unbiasing strategy and the employed method are not compatible. If the data set with the unbiased features is still used, it is necessary to compare the method's performance not against random predictions, but to the baseline performance given by the unbiased features. If, however, the unbiased features are not used explicitly or implicitly by the method, the data set can be used.

(iv) Baseline definition: Comparing a method's performance to random predictions is not enough to recognize bias or test cases which are too easy. A novel and complex method should always be compared to standard methods and simple approaches in the same validation scenario. Perhaps simple linear classifiers or nearest neighbor searches are as good as a more complex nonlinear method. The importance of weak-but-realistic baselines was also recognized by others.<sup>37</sup> Beyond that, we showed when a structure-based method identifies most active molecules of a single target in a test set the developer might assume that the performance achieved is due to relationships derived from protein and small molecule features. However, without performing a baseline experiment this conclusion can not be taken for granted. A simple ligand-based similarity search achieving a similar performance on the test set indicates that the test cases might be too simple. Furthermore, it is worthwhile to also perform ligand-based baseline experiments when the validation scenario would be that ligand-based methods have no predictive power. As described in Section 4, unexpected good results strongly indicate implicit and noncausal bias.

(v) Detecting noncausal bias: Our final guideline deals with evaluating a model with multiple sanity tests in the sense of



negative controls to identify noncausal bias. This corresponds to our approach in Section 4, where we evaluated feature subsets of the descriptor where no causal relationship exists and expected to see no performance, for example, the descriptor can be systematically decomposed into features that are expected to be insufficient for activity prediction in the specific validation scenario. This yields a set of negative controls that the model needs to pass to validate against noncausal bias and should finally reveal which feature subsets are decisive. The first step in the validation of a structure-based method would be to remove the protein from the experiment and examine the performance without protein information. Furthermore, one can replace the target protein by a biologically unrelated protein and see whether the performance drops. The hard part in this systematic bias detection is to actually look at the descriptor and validation experiment and come up with a subset of features to evaluate.

Gabel et al.<sup>57</sup> also proposed two guidelines for better validation experiments for ML-based scoring functions for affinity prediction, which are complementary to ours. These guidelines describe testing the sensitivity of the scoring function to the ligand pose and the ability of the scoring function to discriminate actives from inactives in a VS task.<sup>57</sup>

## 7. ON GENERATING NEW DATA SETS

We see a need for data sets appropriate for ML methods in SBVS. A basic concept for designing a data set without ML-specific bias would be to define a bias scoring function to optimize the selection of data points for a new data set. For ligand-based VS, Wallach et al.<sup>37</sup> proposed a variation of the MUV function for ML called AVE. This scoring function assess bias by equating bias with a one-nearest neighbor predictor, a simple learning method. A similar concept was applied during the development of the original MUV data set, for which bias was measured by the performance in a simple descriptor space. This essentially resembles a negative control. When there is good performance in an insufficient experimental setup, then there is bias. This concept could be extended to a wide range of possible biases. For example, to remove the samples introducing the bias in the experiment of DeepVS, the performance in the LOO-CV with similarity filter could be used as a bias scoring function for optimization. Therefore, it is possible to generate specifically tailored data sets for certain methodologies.

The ideal data set would be a data set of uniformly sampled data points from the chemical space without bias for any methodology. Such a data set would enable the comparison of different methodologies without restrictions. However, the explored chemical space is not sampled uniformly. This most likely leads to a trade-off between bias reduction and the comparability of methodologies. Dissimilar methodologies might have different vulnerabilities to bias and tailored data sets for conventional scoring functions might be unsuited to ML methods. For this reason, there is a need for data sets that are suitable for both ML and conventional scoring functions.

Ultimately, an open problem is the quality and quantity of the available data. Some problems could be fixed if more experimentally validated molecules would be published, especially if inactivity will be more frequently reported to the community. However, it is unforeseeable when the level of an adequate number of data points will be reached. There are, however, ongoing efforts to further refine decoy selection protocols.<sup>58</sup> Finally, an important step toward better data sets

would be a more open sharing of negative results, for example, from large industrial screening campaigns.<sup>3,59</sup>

## 8. CONCLUSION

This perspective draws attention to the problem of bias in current ML-based scoring functions for SBVS. We showed that bias, such as artificial enrichment, is still a problem in established data sets when the context of methodology is changed. ML functions are susceptible to unnoticed bias because they tend to be black boxes and in addition are validated on data sets that are not compiled for ML. More complex methods and descriptors can disguise decisive lower dimensional features. The current popularity of difficult-to-interpret deep learning models covers up bias even more, which can lead to models that instead of protein–molecule interactions learn only molecular features from a structure-based descriptor. To recognize this and other biases, we proposed practical guidelines, which should aid the validation process and avoid fallacious models. The guidelines (ii), (iv), and (v) encourage to design novel methods and descriptors modular to compare variations of them to multiple baselines and validate the model against different sanity tests, which enhances the understanding of the model and reveals biased predictors.

Moreover, we see the need for new data sets suited for ML, because established data sets have not been compiled with the nature of ML methods in mind. This makes current benchmark data sets unsuited for differently operating novel methods and descriptors. Therefore, the proposed guidelines (i) and (iii) also include the verification of the validation domain and unbiasing strategy of a data set because not every data set is suited for every method.

To demonstrate the issues raised, we investigated the behavior of two recently published CNN-based scoring functions. Both scoring functions are well-designed and reflect state-of-the-art methodologies. The utilized methods, that is, docking and the CNNs, do not pose a problem, but the validation experiments do pose a problem. The reported performances change little when essential protein information is removed from the descriptors, especially in terms of AUC. Therefore, the validation does not reveal much about the true practical performance on new targets and compound classes. Since experimental prospective studies can be performed only on rather limited scale, they are not appropriate for measuring method performance in general. As a consequence, it is necessary for the method developer to not only validate the model's predictive capabilities in certain applicability domains, as with intra-target validation and cross-target validation, but also to verify that the chosen validation setup is valid for the model under consideration. As part of ML-based research, it is an important challenge for the near future to come up with reasonable validation schemes. Only then will we be able to exploit the full potential of modern machine learning for drug discovery.

## ■ ASSOCIATED CONTENT

### 📄 Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jcim.8b00712.

Boxplots of AUC and EF1% values achieved with logistic regression and random forest in evaluation of unbiasing techniques of DUD, DUD-E, and MUV; distribution of



matched properties of DUD and DUD-E; analysis of model performance when using single unbiased features; AVE analysis of DUD, DUD-E, and MUV; correlation of EF2% of the original DeepVS and our reimplementations as well as random forest and logistic regression; ROC curves of our 2D and 3D reimplementations of DeepVS; AUC and EF2% correlation plots of folds of our 2D and 3D reimplementations of DeepVS; analysis of actives identity between our 2D and 3D reimplementations of DeepVS as well as random forest; ROC curves for 3D and 2D reimplementations of grid-based CNN; analysis of actives identity between 2D and 3D reimplementations of grid-based CNN (PDF)

## AUTHOR INFORMATION

### Corresponding Author

\*E-mail: [rarey@zbh.uni-hamburg.de](mailto:rarey@zbh.uni-hamburg.de).

### ORCID

Jochen Sieg: 0000-0001-5343-7255

Florian Flachsenberg: 0000-0001-7051-8719

Matthias Rarey: 0000-0002-9553-6531

### Notes

The authors declare no competing financial interest.

Software availability: The software containing the reimplemented networks and descriptors as described in section 4.2 and 4.4 are available open source via github from <https://github.com/rareylab/MLValidation>.

## ACKNOWLEDGMENTS

The authors thank D. Koes for providing additional information and scripts related to the 3D CNN approach to us. This work was orally presented at the International Conference on Chemical Structures (ICCS), Noordwijkerhout in June 2018. We are grateful for all the comments and suggestions we received from numerous participants of the conference.

## REFERENCES

- (1) *Virtual Screening in Drug Discovery*; Alvarez, J., Shoichet, B., Eds.; CRC Press, 2005; Vol. 1.
- (2) Special Issue: A snapshot in time: Docking Challenge. *J. Comput.-Aided Mol. Des.* **2012**, *26*, 675–799.
- (3) Carlson, H. A.; Smith, R. D.; Damm-Ganamet, K. L.; Stuckey, J. A.; Ahmed, A.; Convery, M. A.; Somers, D. O.; Kranz, M.; Elkins, P. A.; Cui, G.; Peishoff, C. E.; Lambert, M. H.; Dunbar, J. B. CSAR 2014: A Benchmark Exercise Using Unpublished Data from Pharma. *J. Chem. Inf. Model.* **2016**, *56*, 1063–1077.
- (4) Rohrer, S. G.; Baumann, K. Maximum Unbiased Validation (MUV) Data Sets for Virtual Screening Based on PubChem Bioactivity Data. *J. Chem. Inf. Model.* **2009**, *49*, 169–184.
- (5) Huang, N.; Shoichet, B. K.; Irwin, J. J. Benchmarking Sets for Molecular Docking. *J. Med. Chem.* **2006**, *49*, 6789–6801.
- (6) Mysinger, M. M.; Carchia, M.; Irwin, J. J.; Shoichet, B. K. Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking. *J. Med. Chem.* **2012**, *55*, 6582–6594.
- (7) Vogel, S. M.; Bauer, M. R.; Boeckler, F. M. DEKOIS: Demanding Evaluation Kits for Objective in Silico Screening - A Versatile Tool for Benchmarking Docking Programs and Scoring Functions. *J. Chem. Inf. Model.* **2011**, *51*, 2650–2665.
- (8) Bauer, M. R.; Ibrahim, T. M.; Vogel, S. M.; Boeckler, F. M. Evaluation and Optimization of Virtual Screening Workflows with DEKOIS 2.0 - A Public Library of Challenging Docking Benchmark Sets. *J. Chem. Inf. Model.* **2013**, *53*, 1447–1462.
- (9) Dunbar, J. B.; Smith, R. D.; Yang, C.-Y.; Ung, P. M.-U.; Lexa, K. W.; Khazanov, N. A.; Stuckey, J. A.; Wang, S.; Carlson, H. A. CSAR Benchmark Exercise of 2010: Selection of the Protein-Ligand Complexes. *J. Chem. Inf. Model.* **2011**, *51*, 2036–2046.
- (10) Liu, Z.; Su, M.; Han, L.; Liu, J.; Yang, Q.; Li, Y.; Wang, R. Forging the Basis for Developing Protein-Ligand Interaction Scoring Functions. *Acc. Chem. Res.* **2017**, *50*, 302–309.
- (11) Wallach, I.; Lilien, R. Virtual Decoy Sets for Molecular Docking Benchmarks. *J. Chem. Inf. Model.* **2011**, *51*, 196–202.
- (12) Ballester, P. J.; Mitchell, J. B. O. A machine learning approach to predicting proteinligand binding affinity with applications to molecular docking. *Bioinformatics* **2010**, *26*, 1169–1175.
- (13) Wallach, I.; Dzamba, M.; Heifets, A. AtomNet: A Deep Convolutional Neural Network for Bioactivity Prediction in Structure-based Drug Discovery. *arXiv preprint arXiv:1510.02855*, **2015**.
- (14) Pereira, J. C.; Caffarena, E. R.; dos Santos, C. N. Boosting Docking-Based Virtual Screening with Deep Learning. *J. Chem. Inf. Model.* **2016**, *56*, 2495–2506.
- (15) Ragoza, M.; Hochuli, J.; Idrobo, E.; Sunseri, J.; Koes, D. R. Protein-Ligand Scoring with Convolutional Neural Networks. *J. Chem. Inf. Model.* **2017**, *57*, 942–957.
- (16) Wójcikowski, M.; Ballester, P. J.; Siedlecki, P. Performance of machine-learning scoring functions in structure-based virtual screening. *Sci. Rep.* **2017**, *7*, 46710.
- (17) Jiménez, J.; Skalič, M.; Martínez-Rosell, G.; De Fabritiis, G. KDEEP: Protein-Ligand Absolute Binding Affinity Prediction via 3D-Convolutional Neural Networks. *J. Chem. Inf. Model.* **2018**, *58*, 287–296.
- (18) Polishchuk, P. Interpretation of Quantitative Structure-Activity Relationship Models: Past, Present, and Future. *J. Chem. Inf. Model.* **2017**, *57*, 2618–2639.
- (19) Chuang, K. V.; Keiser, M. J. Comment on “Predicting reaction performance in C-N cross-coupling using machine learning”. *Science* **2018**, *362*, eaat8603.
- (20) Réau, M.; Langenfeld, F.; Zagury, J.-F.; Lagarde, N.; Montes, M. Decoys Selection in Benchmarking Datasets: Overview and Perspectives. *Front. Pharmacol.* **2018**, *9*, 11.
- (21) Bissantz, C.; Folkers, G.; Rognan, D. Protein-Based Virtual Screening of Chemical Databases. 1. Evaluation of Different Docking/Scoring Combinations. *J. Med. Chem.* **2000**, *43*, 4759–4767.
- (22) McGovern, S. L.; Shoichet, B. K. Information Decay in Molecular Docking Screens against Holo, Apo, and Modeled Conformations of Enzymes. *J. Med. Chem.* **2003**, *46*, 2895–2907.
- (23) Diller, D. J.; Li, R. Kinases, Homology Models, and High Throughput Docking. *J. Med. Chem.* **2003**, *46*, 4638–4647.
- (24) Verdonk, M. L.; Berdini, V.; Hartshorn, M. J.; Mooij, W. T. M.; Murray, C. W.; Taylor, R. D.; Watson, P. Virtual Screening Using Protein-Ligand Docking: Avoiding Artificial Enrichment. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 793–806.
- (25) Irwin, J. J.; Shoichet, B. K. ZINC - A Free Database of Commercially Available Compounds for Virtual Screening. *J. Chem. Inf. Model.* **2005**, *45*, 177–182.
- (26) Ihlenfeldt, W. D.; Takahashi, Y.; Abe, H.; Sasaki, S. Computation and management of chemical properties in CACTVS: An extensible networked approach toward modularity and compatibility. *J. Chem. Inf. Model.* **1994**, *34*, 109–116.
- (27) Good, A. C.; Oprea, T. I. Optimization of CAMD techniques 3. Virtual screening enrichment studies: a help or hindrance in tool selection? *J. Comput.-Aided Mol. Des.* **2008**, *22*, 169–178.
- (28) Armstrong, M. S.; Morris, G. M.; Finn, P. W.; Sharma, R.; Moretti, L.; Cooper, R. I.; Richards, W. G. ElectroShape: fast molecular similarity calculations incorporating shape, chirality and electrostatics. *J. Comput.-Aided Mol. Des.* **2010**, *24*, 789–801.
- (29) Mysinger, M. M.; Shoichet, B. K. Rapid Context-Dependent Ligand Desolvation in Molecular Docking. *J. Chem. Inf. Model.* **2010**, *50*, 1561–1573.
- (30) Gaulton, A.; Bellis, L. J.; Bento, A. P.; Chambers, J.; Davies, M.; Hersey, A.; Light, Y.; McGlinchey, S.; Michalovich, D.; Al-Lazikani,

B.; Overington, J. P. ChEMBL: a largescale bioactivity database for drug discovery. *Nucleic Acids Res.* **2012**, *40*, D1100–D1107.

(31) Bemis, G. W.; Murcko, M. A. The Properties of Known Drugs. 1. Molecular Frameworks. *J. Med. Chem.* **1996**, *39*, 2887–2893.

(32) Xia, J.; Tilahun, E. L.; Reid, T.-E.; Zhang, L.; Wang, X. S. Benchmarking methods and data sets for ligand enrichment assessment in virtual screening. *Methods* **2015**, *71*, 146–157.

(33) Wheeler, D. L.; et al. Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res.* **2007**, *36*, D13–21.

(34) *Web of Knowledge*; Clarivate, 2018. [www.webofknowledge.com](http://www.webofknowledge.com) (accessed Sept 7, 2018).

(35) Cleves, A. E.; Jain, A. N. Effects of inductive bias on computational evaluations of ligand-based modeling and on drug discovery. *J. Comput.-Aided Mol. Des.* **2008**, *22*, 147–159.

(36) Ruddigkeit, L.; van Deursen, R.; Blum, L. C.; Reymond, J.-L. Enumeration of 166 Billion Organic Small Molecules in the Chemical Universe Database GDB-17. *J. Chem. Inf. Model.* **2012**, *52*, 2864–2875.

(37) Wallach, I.; Heifets, A. Most Ligand-Based Classification Benchmarks Reward Memorization Rather than Generalization. *J. Chem. Inf. Model.* **2018**, *58*, 916–932.

(38) Bohacek, R. S.; McMartin, C.; Guida, W. C. The art and practice of structure-based drug design: A molecular modeling perspective. *Med. Res. Rev.* **1996**, *16*, 3–50.

(39) Jain, A. N.; Cleves, A. E. Does your model weigh the same as a Duck? *J. Comput.-Aided Mol. Des.* **2012**, *26*, 57–67.

(40) Shoichet, B. K.; Stroud, R. M.; Santi, D. V.; Kuntz, I. D.; Perry, K. M. Structure-based discovery of inhibitors of thymidylate synthase. *Science* **1993**, *259*, 1445–1450.

(41) Liu, J.; Wang, R. Classification of Current Scoring Functions. *J. Chem. Inf. Model.* **2015**, *55*, 475–482.

(42) Bietz, S.; Schomburg, K. T.; Hilbig, M.; Rarey, M. Discriminative Chemical Patterns: Automatic and Interactive Design. *J. Chem. Inf. Model.* **2015**, *55*, 1535–1546.

(43) RDKit: *Open-source cheminformatics*. Version: 2017.09.3.0; RDKit, 2018. <http://www.rdkit.org> (accessed Jan 21, 2018).

(44) Pedregosa, F.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

(45) Chollet, F. Keras. Version: 2.1.5; Keras, 2015. <https://keras.io> (accessed Mar 19, 2018).

(46) Abadi, M. et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Version: 1.7.0; TensorFlow, 2015; <https://www.tensorflow.org/> (accessed Apr 22, 2018).

(47) *Directory of Useful Decoys website - Partial Charges for DUD Molecules recalculated by Inhibox*. AM1; Docking.org, 2017 (accessed May 12, 2017).

(48) *Directory of Useful Decoys Enhanced website*; Docking.org, 2017 (accessed May 21, 2017).

(49) *Maximum Unbiased Validation (MUV) Data Sets website*; Technische Universität Braunschweig, 2017. <https://www.tu-braunschweig.de/pharmchem/forschung/baumann/muv> (accessed Jun 5, 2017).

(50) Fawcett, T. An introduction to ROC analysis. *Pattern Recognit. Lett.* **2006**, *27*, 861–874.

(51) Riniker, S.; Landrum, G. A. Open-source platform to benchmark fingerprints for ligandbased virtual screening. *J. Cheminf.* **2013**, *5*, 26 DOI: 10.1186/1758-2946-5-26.

(52) Guyon, I.; Elisseeff, A. An Introduction to Variable and Feature Selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.

(53) Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32.

(54) Svetnik, V.; Liaw, A.; Tong, C.; Culberson, J. C.; Sheridan, R. P.; Feuston, B. P. Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1947–1958.

(55) *Clustering Script*; GitHub, 2018. <https://github.com/gnina/scripts/blob/9cf8892010873b672f370a122e32aa8bc496a5e1/clustering.py> (accessed May 8, 2018).

(56) Wu, Z.; Ramsundar, B.; Feinberg, E.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: a benchmark for molecular machine learning. *Chem. Sci.* **2018**, *9*, 513–530.

(57) Gabel, J.; Desaphy, J.; Rognan, D. Beware of Machine Learning-Based Scoring Functions-On the Danger of Developing Black Boxes. *J. Chem. Inf. Model.* **2014**, *54*, 2807–2815.

(58) Xia, J.; Reid, T.-E.; Wu, S.; Zhang, L.; Wang, X. S. Maximal Unbiased Benchmarking Data Sets for Human Chemokine Receptors and Comparative Analysis. *J. Chem. Inf. Model.* **2018**, *58*, 1104–1120.

(59) Gaieb, Z.; Liu, S.; Gathiaka, S.; Chiu, M.; Yang, H.; Shao, C.; Feher, V. A.; Walters, W. P.; Kuhn, B.; Rudolph, M. G.; Burley, S. K.; Gilson, M. K.; Amaro, R. E. D3R Grand Challenge 2: blind prediction of protein-ligand poses, affinity rankings, and relative binding free energies. *J. Comput.-Aided Mol. Des.* **2018**, *32*, 1–20.

Supporting Information:

In Need of Bias Control: Evaluating Chemical  
Data for Machine Learning in Structure-Based  
Virtual Screening

Jochen Sieg, Florian Flachsenberg, and Matthias Rarey\*

*Center for Bioinformatics, Research Group for Computational Molecular Design,  
University of Hamburg, Bundesstraße 43, 20146 Hamburg, Germany*

E-mail: [rarey@zbh.uni-hamburg.de](mailto:rarey@zbh.uni-hamburg.de)

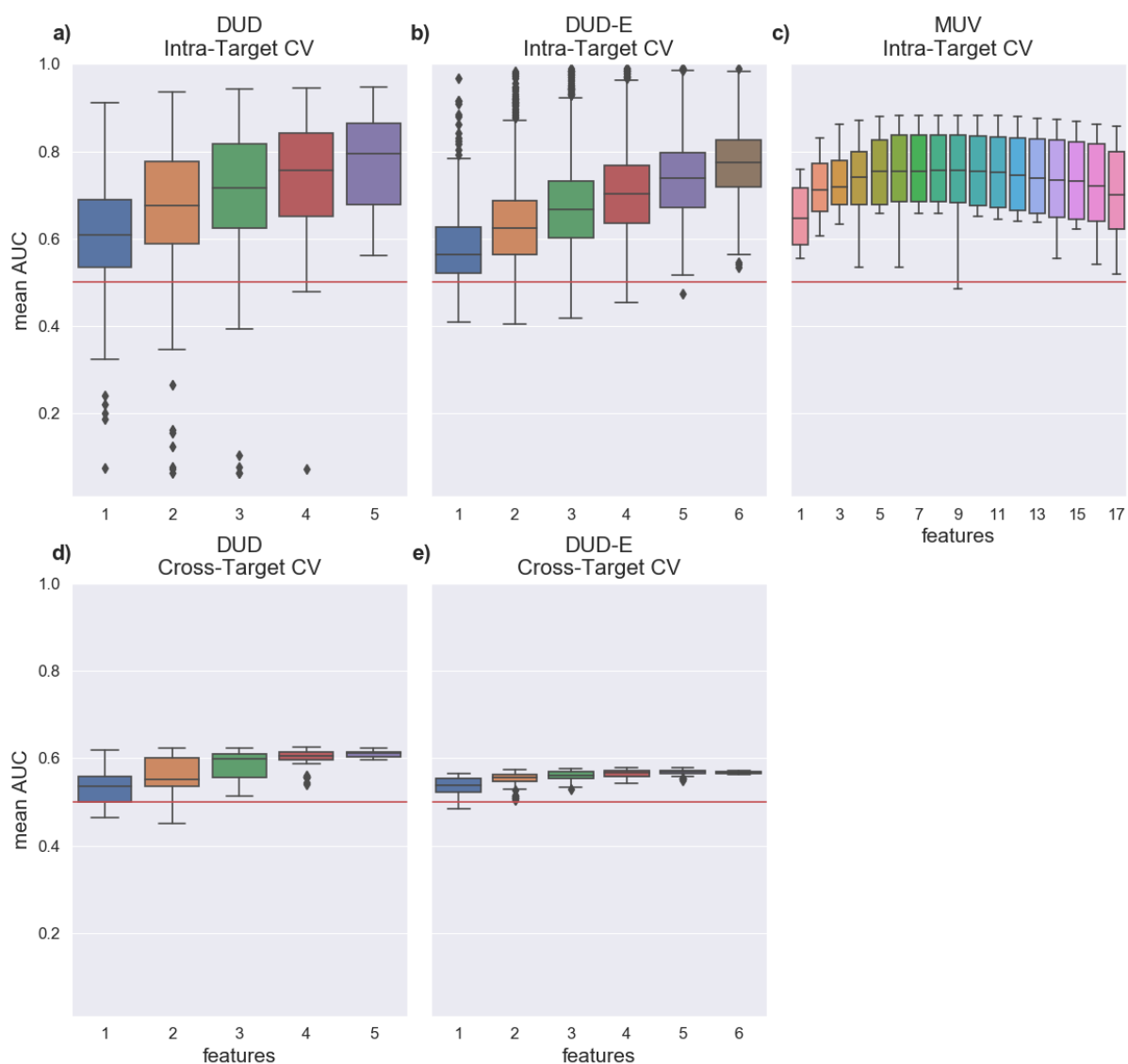


Figure S1: Box plot of the evaluation of unbiased features with logistic regression (LR) of DUD, DUD-E and MUV using AUC. The first row shows results of the intra-target CV for (a) DUD, (b) DUD-E and (c) MUV. In the second row results of cross target CV are depicted for (d) DUD and (e) DUD-E.

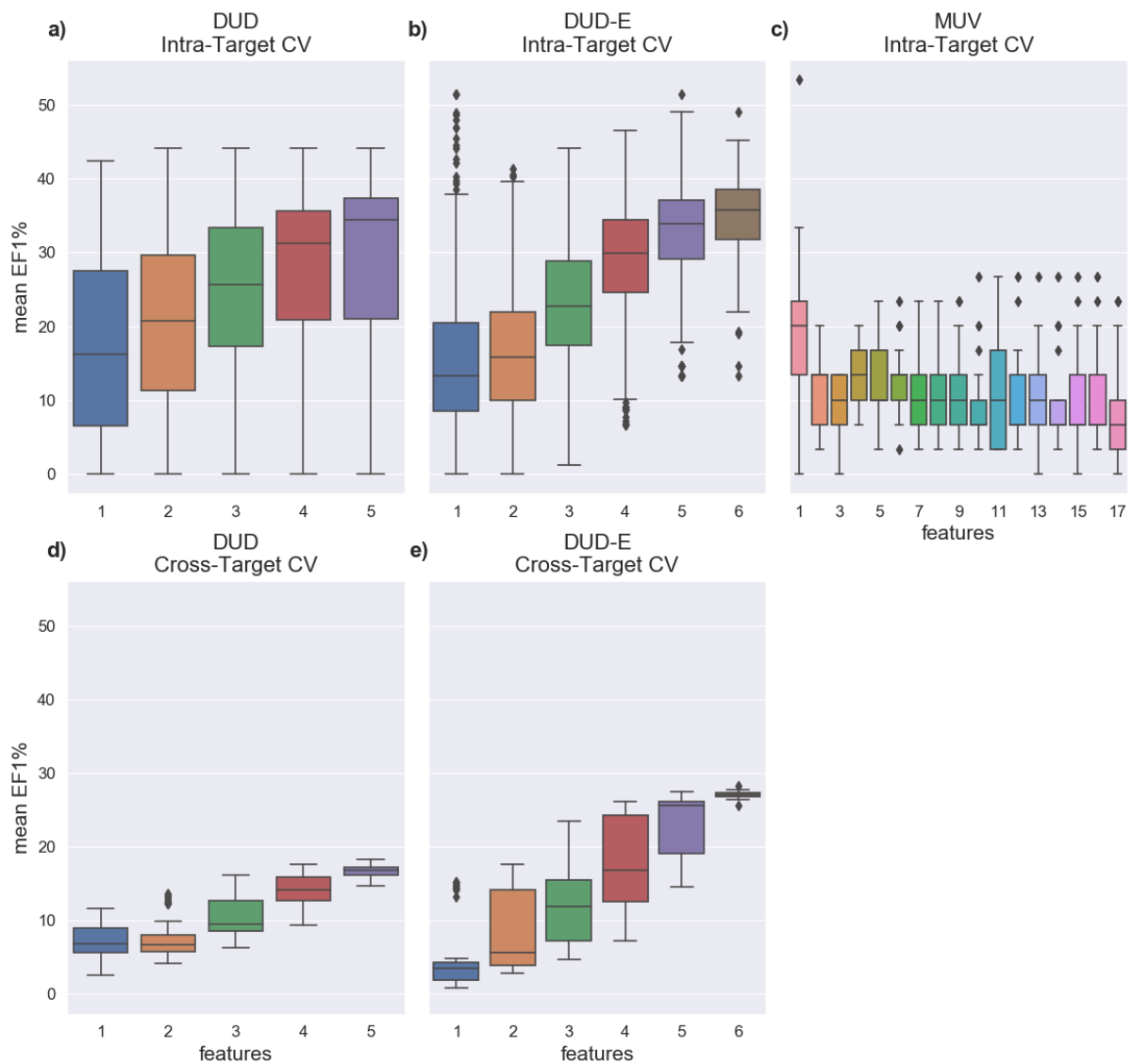


Figure S2: Box plot of the evaluation of unbiased features with random forest (RF) of DUD, DUD-E and MUV. Performance is assessed with the mean enrichment factor on one percent of the test sets (EF1%). The first row shows results of the intra-target CV for (a) DUD, (b) DUD-E and (c) MUV. In the second row results of cross target CV are depicted for (d) DUD and (e) DUD-E.

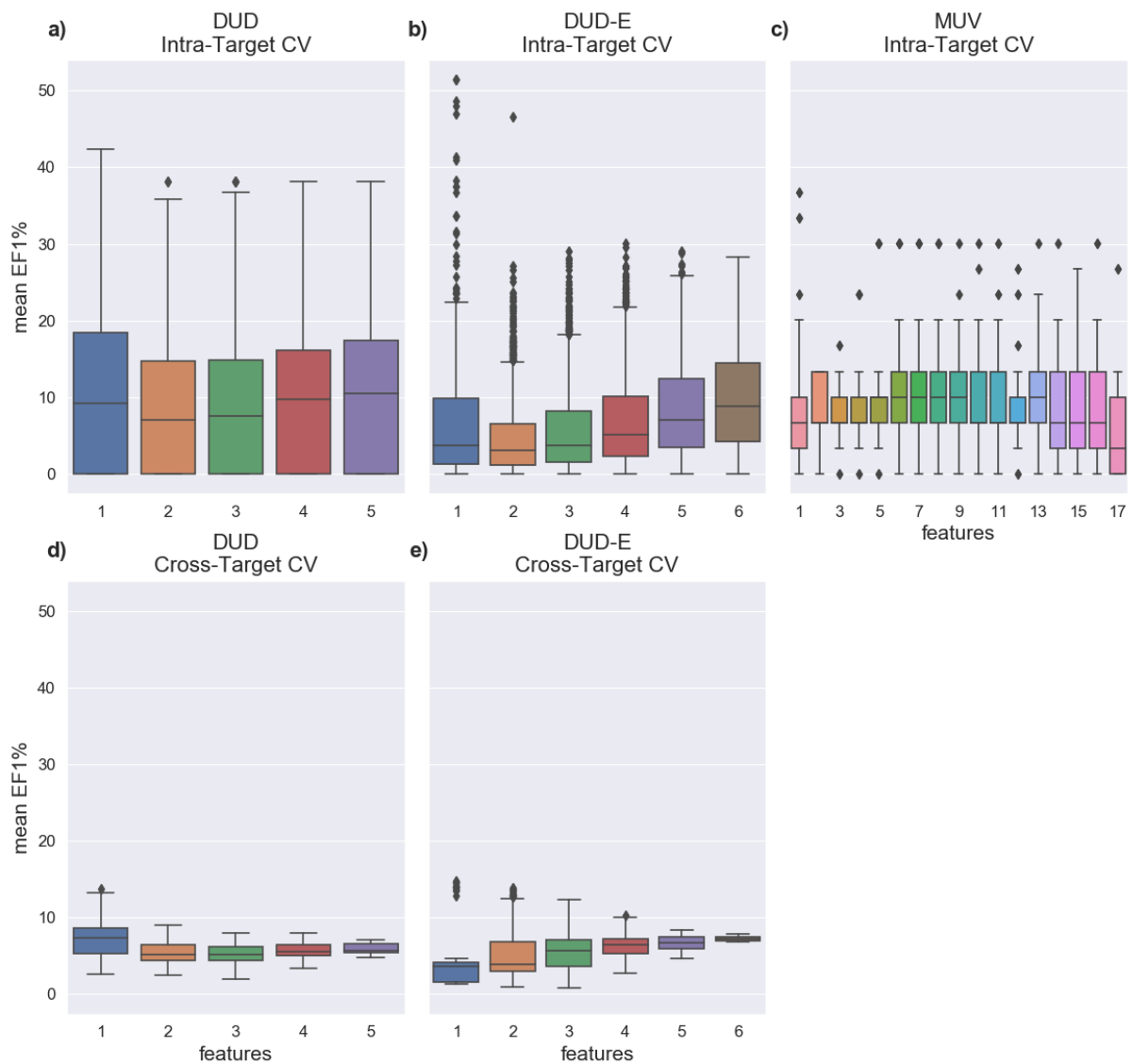


Figure S3: Box plot of the evaluation of unbiased features with logistic regression (LR) of DUD, DUD-E and MUV. Performance is assessed with the mean enrichment factor on one percent of the test sets (EF1%). The first row shows results of the intra-target CV for (a) DUD, (b) DUD-E and (c) MUV. In the second row results of cross target CV are depicted for (d) DUD and (e) DUD-E.

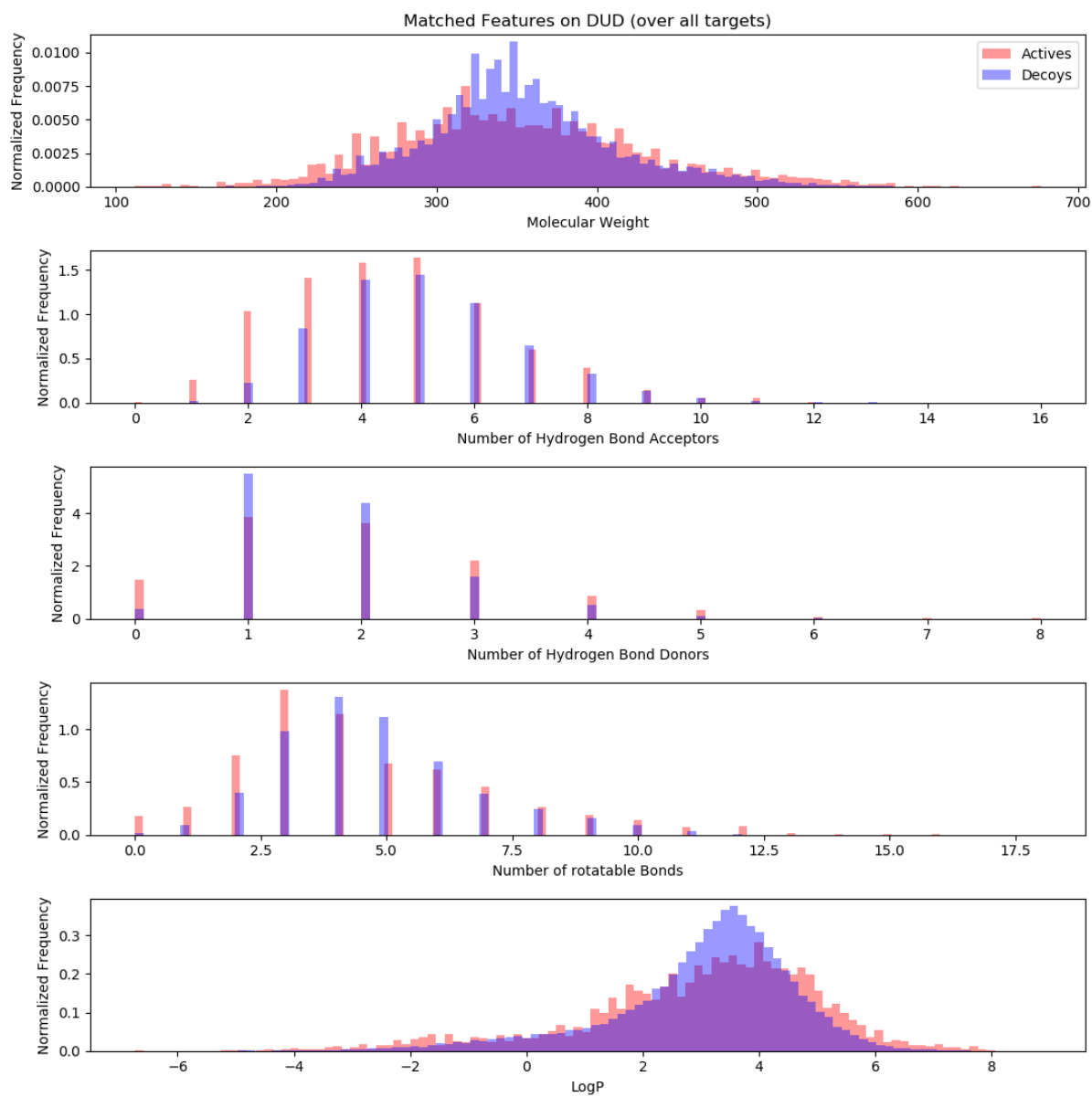


Figure S4: Histograms of all unbiased features of DUD over all targets of the dataset. Actives are marked in red and inactives in blue.

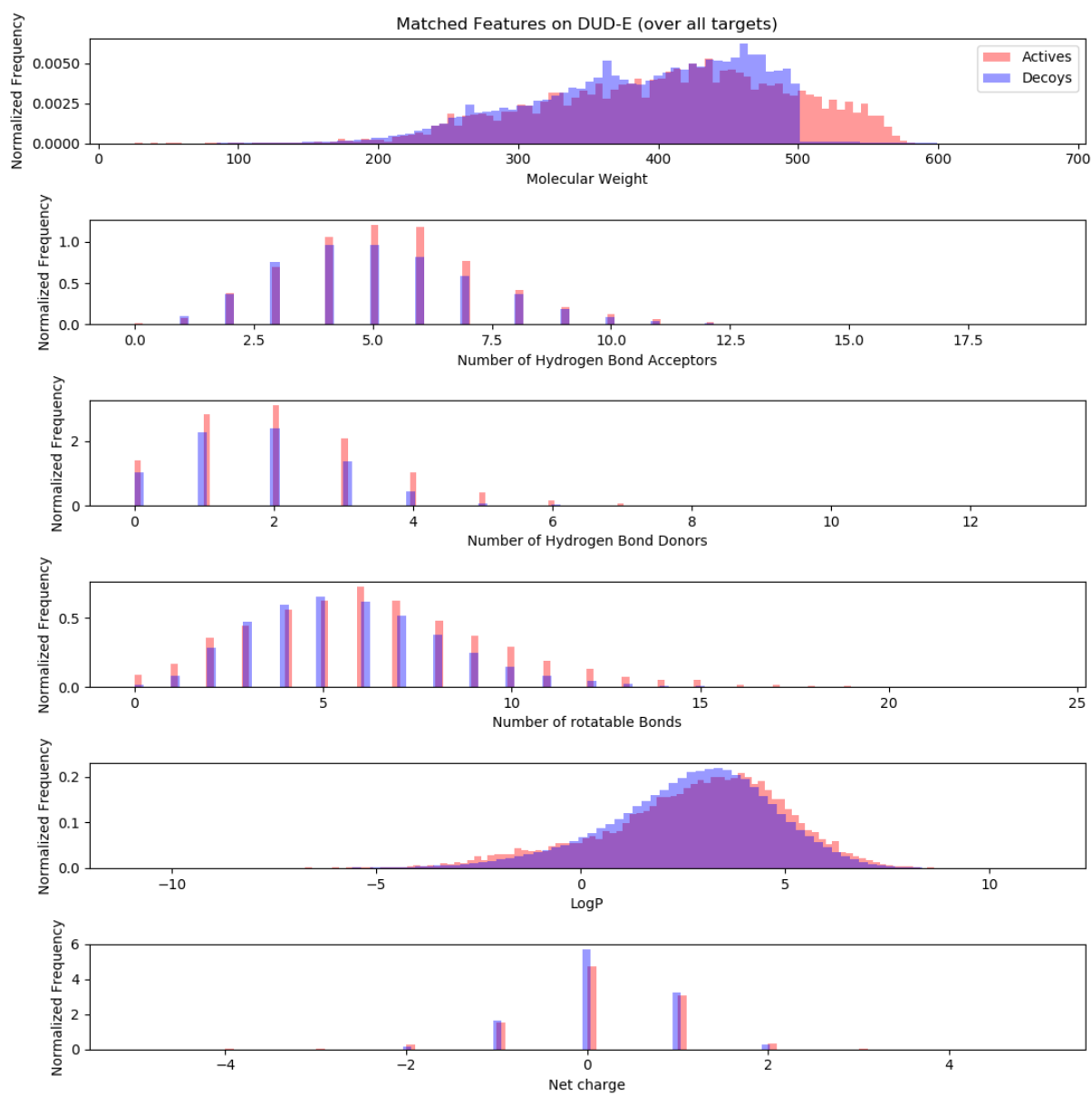


Figure S5: Histograms of all unbiased features of DUD-E over all targets of the dataset. Actives are marked in red and inactives in blue.



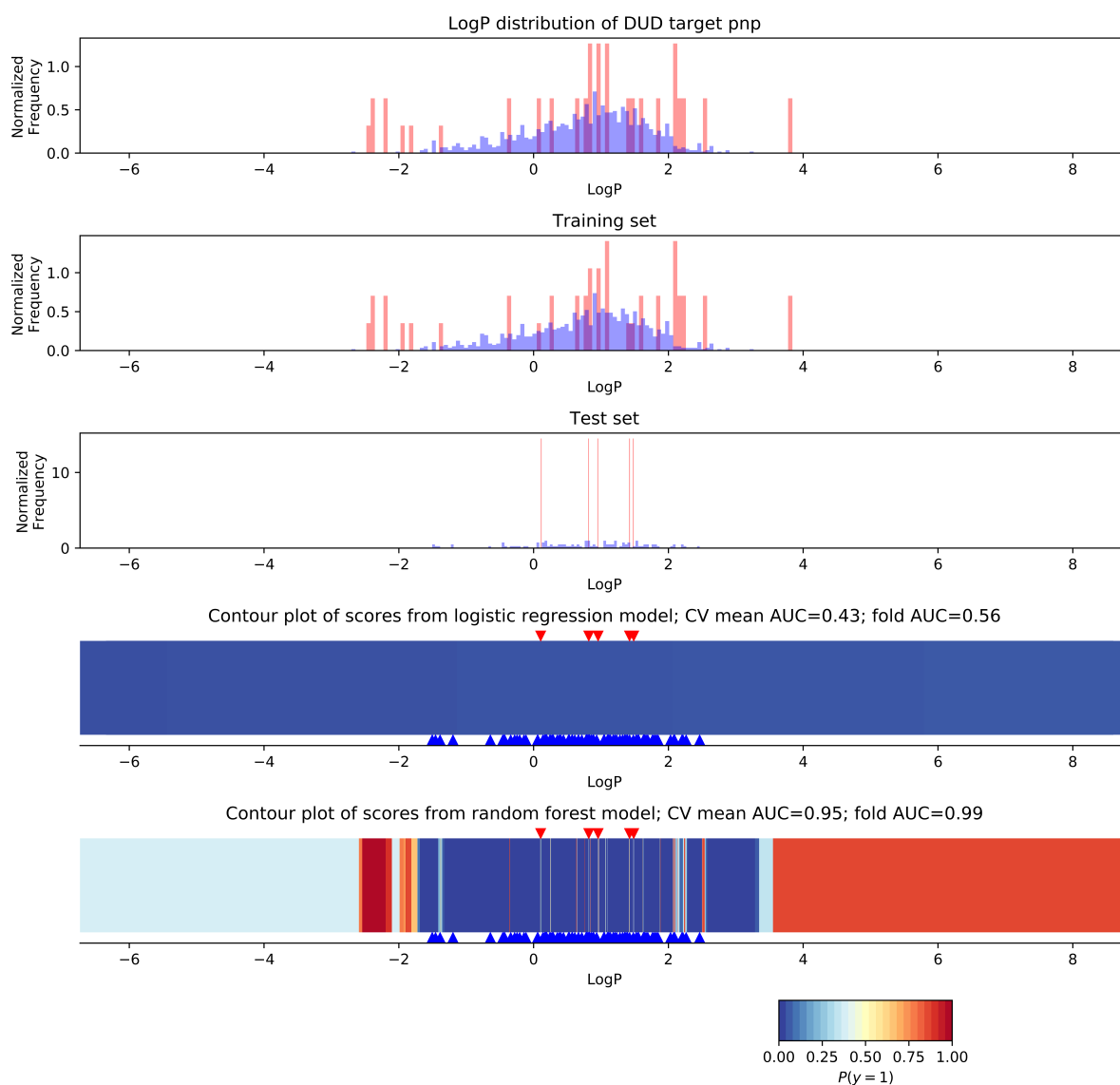


Figure S6: Illustration of the intra-target CV experiment with LogP on the protein target PNP of DUD. In this experiment the highest mean AUC value with random forest was achieved over all DUD targets and single features. Histograms depict the LogP distributions on the target level and of the training and test set of one fold of the CV. Additionally, contour plots of the scores of the random forest model and logistic regression model fitted on the training set is shown. The red triangles on the top of the contour plots mark the LogP value for the actives in the test set, while the blue triangles at the bottom mark the LogP values of the test decoys.

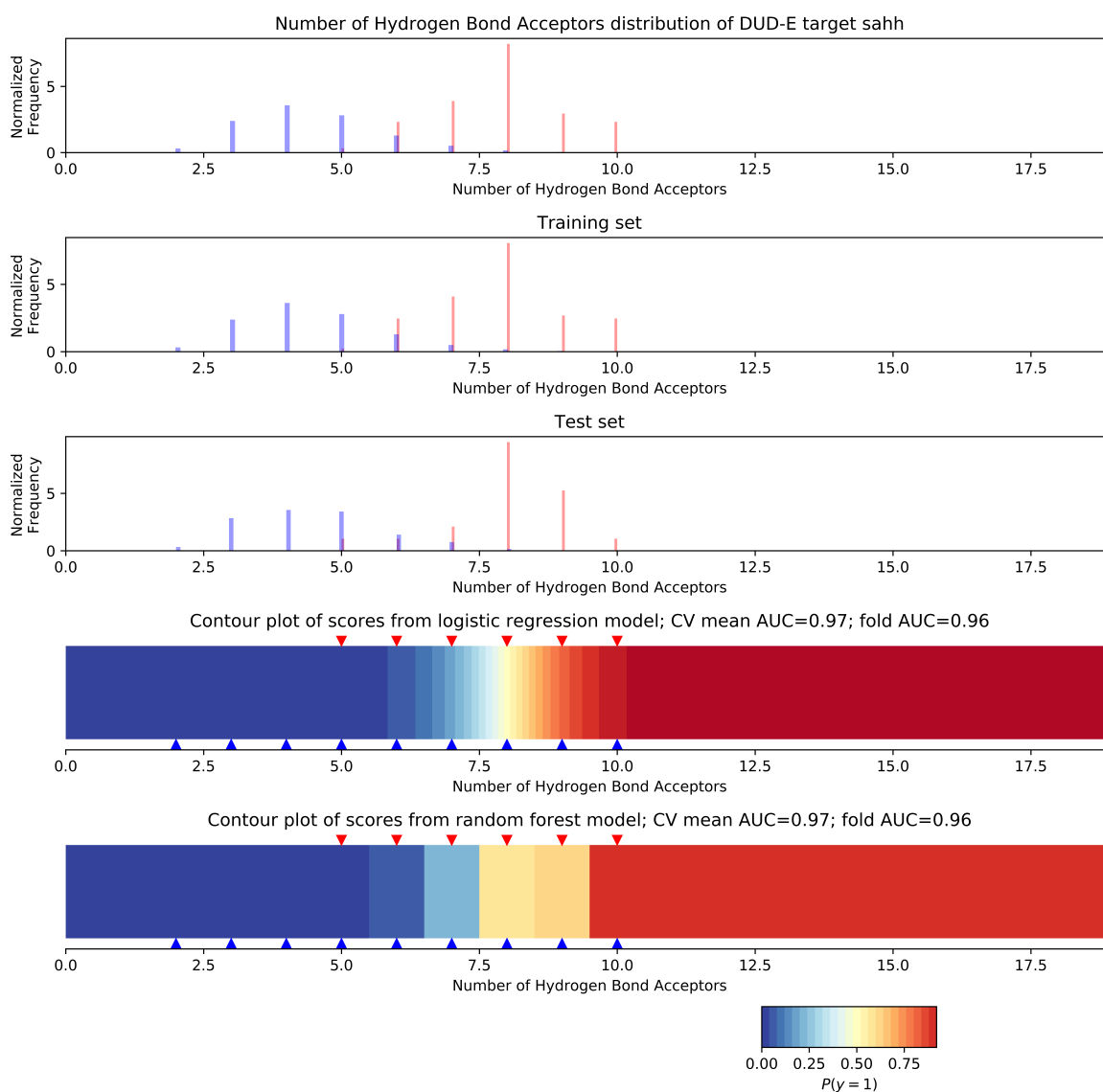


Figure S7: Illustration of the intra-target CV experiment with the number of hydrogen bond acceptors on the protein target SAHH of DUD-E. In this experiment the highest mean AUC value with linear regression was achieved over all DUD-E targets with single features. Histograms depict the acceptor distributions on the target level and of the training and test set of one fold of the CV. Additionally, contour plots of the scores of the random forest model and logistic regression model fitted on the training set is shown. The red triangles on the top of the contour plots mark the number of acceptors for the actives in the test set, while the blue triangles at the bottom mark the number of acceptors of the test decoys.

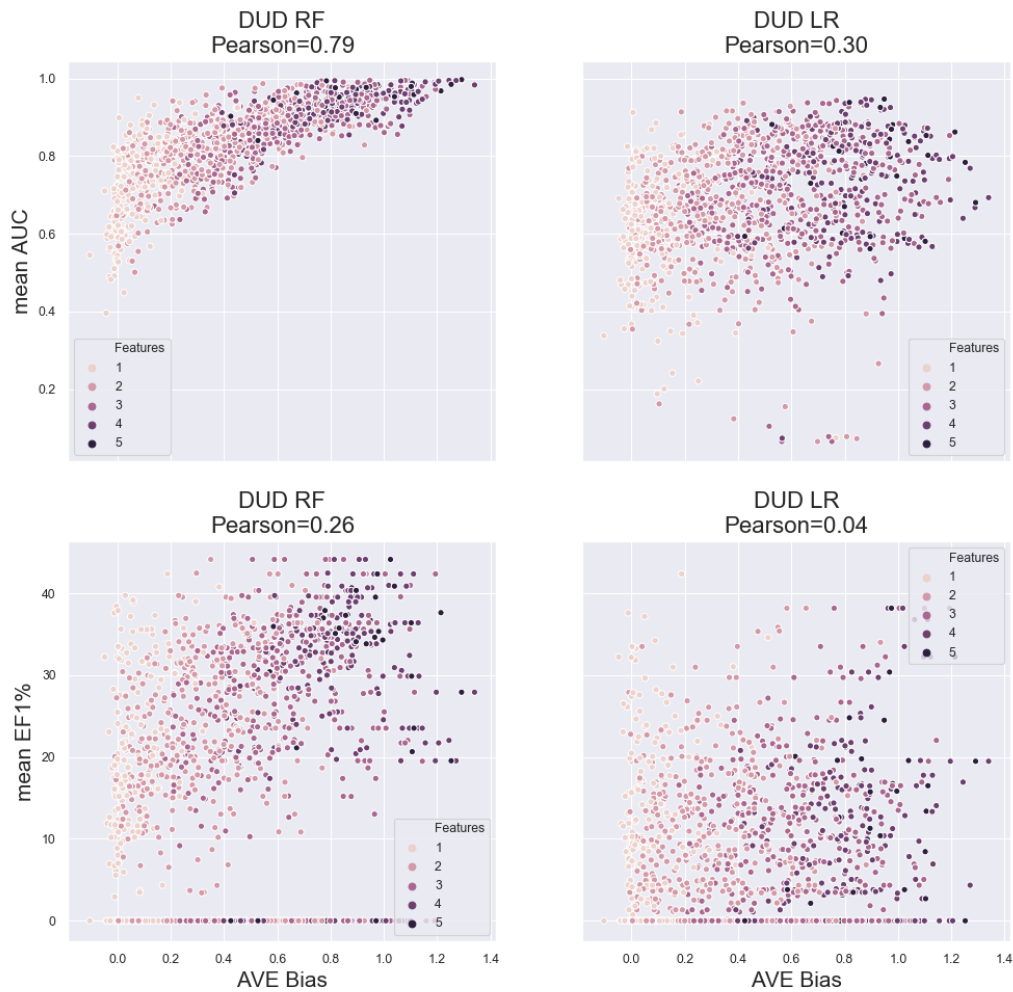


Figure S8: Correlation plots of AVE bias and AUC in the intra-target CV of DUD. One point represents the mean performance in dependence of the AVE bias over a single intra-target CV experiment (this plot sets the results of Figure 2, S1, S2, S3 in relation to AVE bias). For RF evaluated with AUC there is a notable correlation between performance and AVE bias. The coloring of the points illustrates the number of features used in the experiments (light coloring means less features and darker more features). It can be seen that experiments with less features exhibit low or no AVE bias, while experiments with many features have increased AVE bias. Even though there is correlation when using RF with AUC there are also many examples, especially with a single feature, where there is very high performance, but no AVE bias. Therefore, not all bias in DUD is explained by AVE.

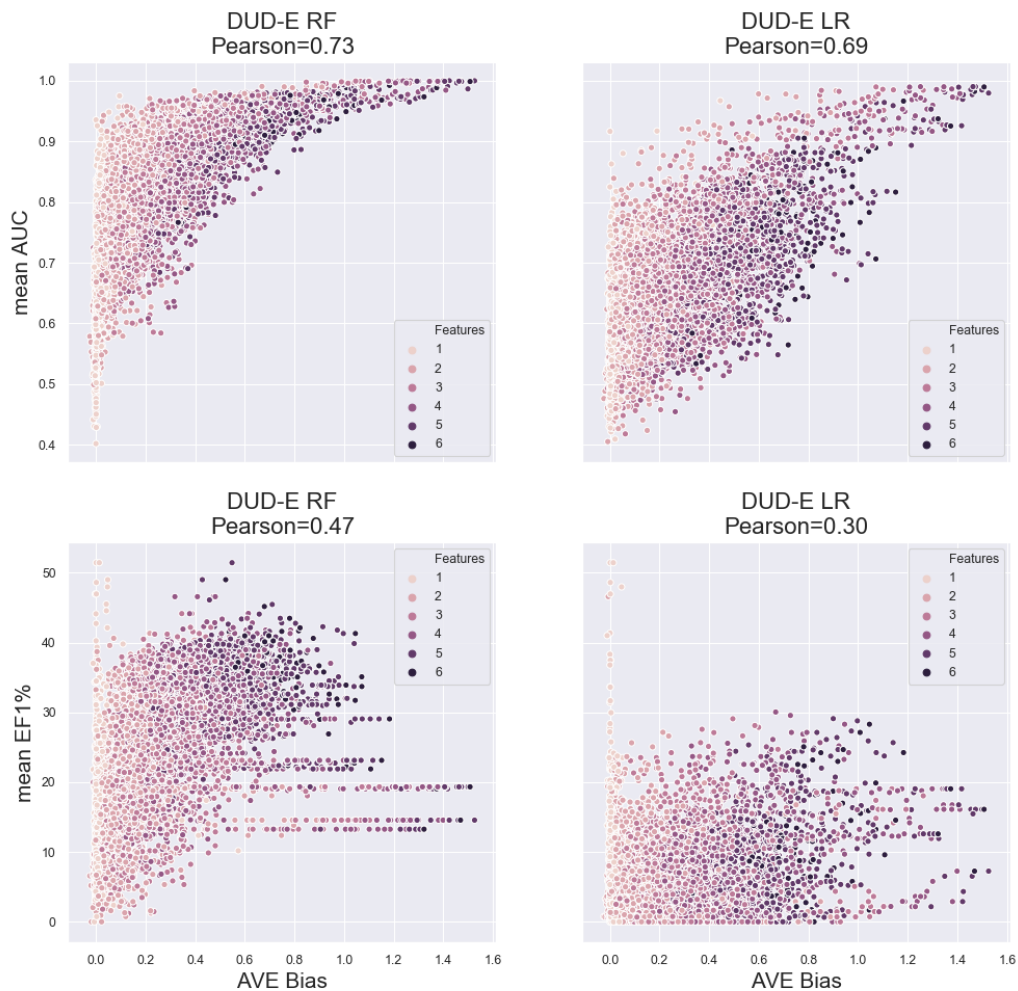


Figure S9: Correlation plots of AVE bias and AUC in the intra-target CV of DUD-E. One point represents the mean performance in dependence of the AVE bias over a single intra-target CV experiment (this plot sets the results of Figure 2, S1, S2, S3 in relation to AVE bias). For RF and LR evaluated with AUC there is a notable correlation between performance and AVE bias. For experiments with EF1% there is a moderate and weak correlation observable. The coloring of the points illustrates the number of features used in the experiments (light coloring means less features and darker more features). It can be seen that experiments with less features exhibit low or no AVE bias, while experiments with many features have increased AVE bias. Even though there is correlation when using RF and LR with AUC there are also many examples, especially with a single feature, where there is very high performance, but no AVE bias. Therefore, not all bias in DUD-E is explained by AVE.

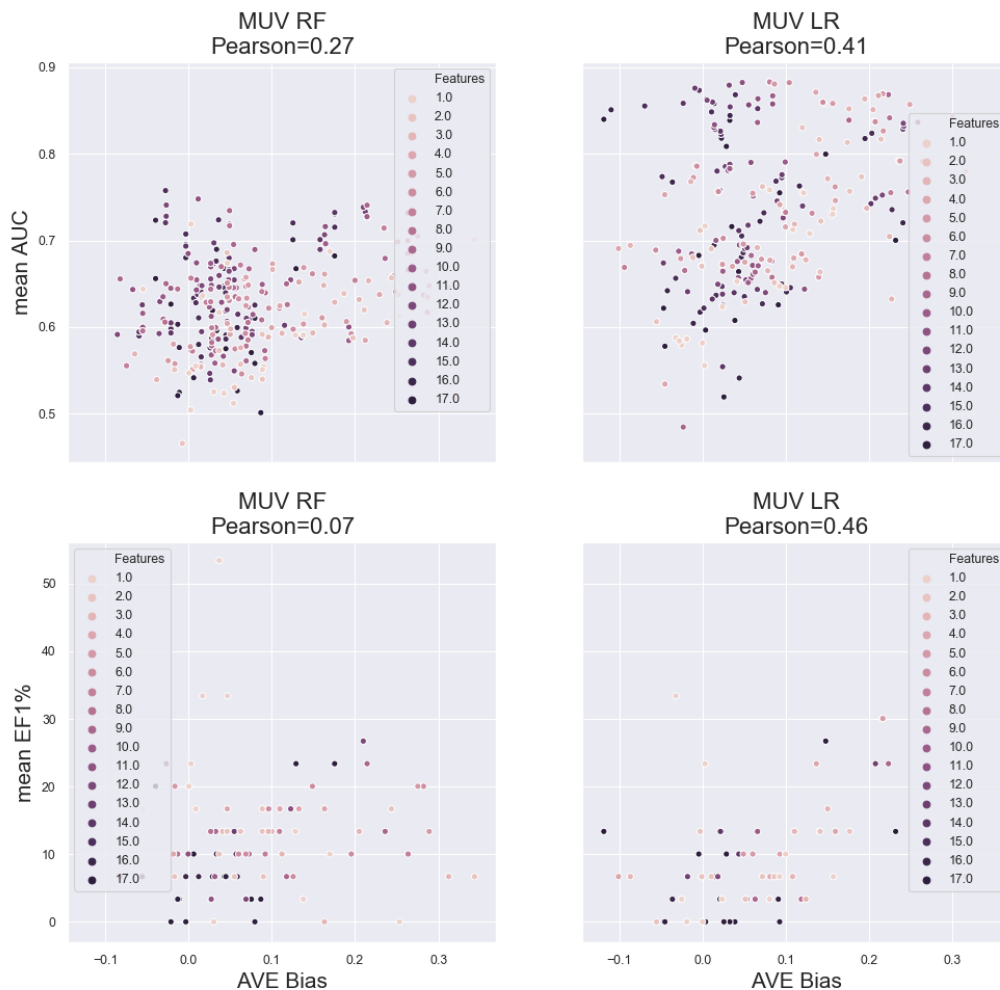


Figure S10: Correlation plots of AVE bias and AUC in the intra-target CV of MUV. One point represents the mean performance in dependence of the AVE bias over a single intra-target CV experiment (this plot sets the results of Figure 2, S1, S2, S3 in relation to AVE bias). There is a weak to moderate correlation between LR results in both AUC and EF1%, while for RF no notable correlation is observed. The coloring of the points illustrates the number of features used in the experiments (light coloring means less features and darker more features). There is no clear trend observable in the coloring.



Figure S11: Correlation plots of enrichment factor at 2% (EF2%) values of the structure-based original DeepVS (values taken from *Pereira et al.* 2016) and the four other approaches. **a)** shows the performance of our 3D ligand-based reimplementaion. The second plot (**b**) shows the correlation of the reimplementaion using the topological distance on the molecular graph instead of 3D distances. Finally, the performance of RF and LR with Morgan3 fingerprints is plotted against the original DeepVS in **c)** and **d)**, respectively.

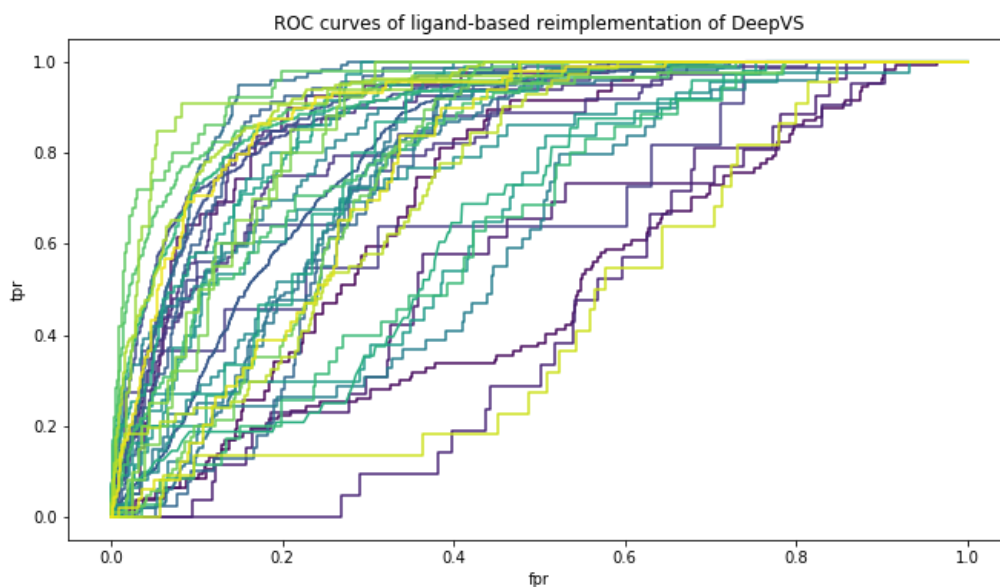


Figure S12: ROC curves of the 40 folds of the LOO-CV with our ligand-based 3D reimplementaion of DeepVS.

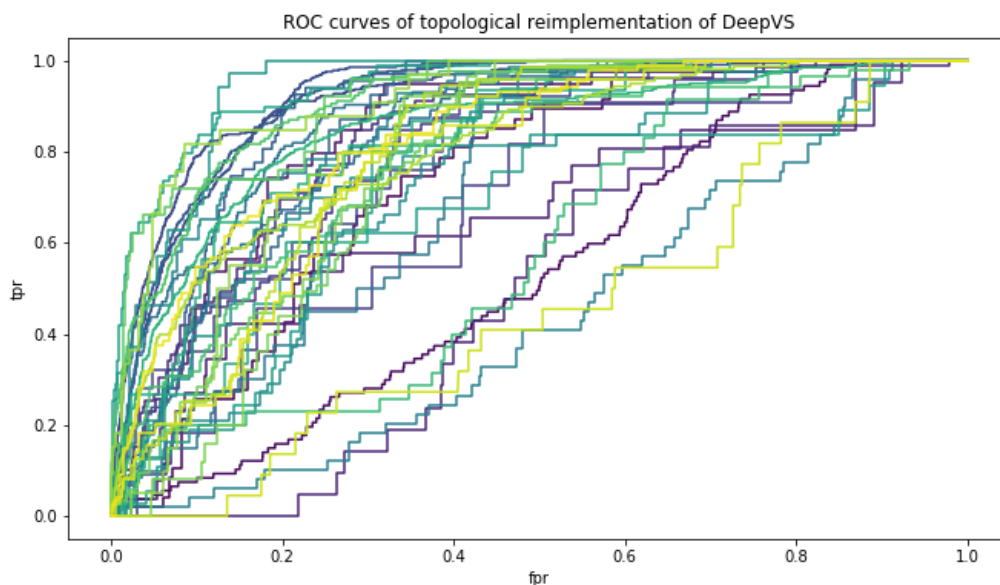


Figure S13: ROC curves of the 40 folds of the LOO-CV with our ligand-based 2D (topological) reimplementations of DeepVS.

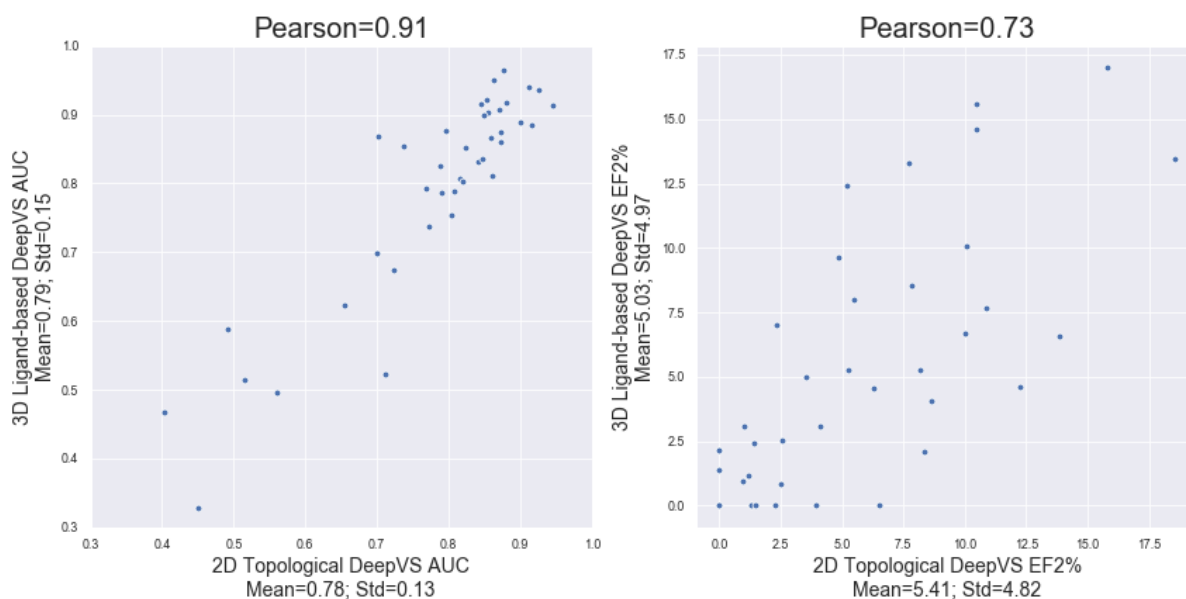


Figure S14: Comparison of AUC and EF2% values of our ligand-based 3D reimplementations and our 2D (topological) reimplementations of DeepVS. Correlation plots over the 40 cross validation folds.

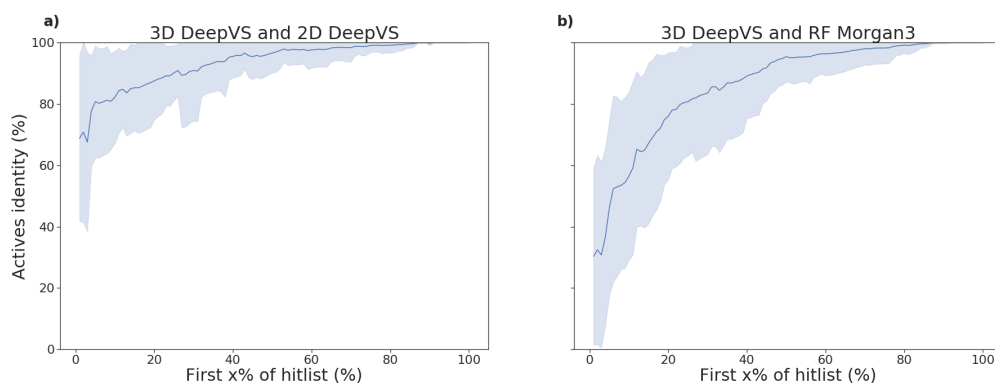


Figure S15: Distribution of overlapping actives between the (a) 3D DeepVS and 2D DeepVS as well as the (b) 3D DeepVS and RF with Morgan3 fingerprints. The mean values over the 40 folds of the LOO-CV on DUD are shown. On the  $y$ -axis the overlap of actives between the hitlists is depicted in dependence of the first  $x\%$  of the ranked molecules, which is shown on the  $x$ -axis. Consequently, one point in the plot describes that in the first  $x\%$  of the hitlists  $y\%$  of the enriched actives are identical between methods. The light colored band shows the standard deviation.

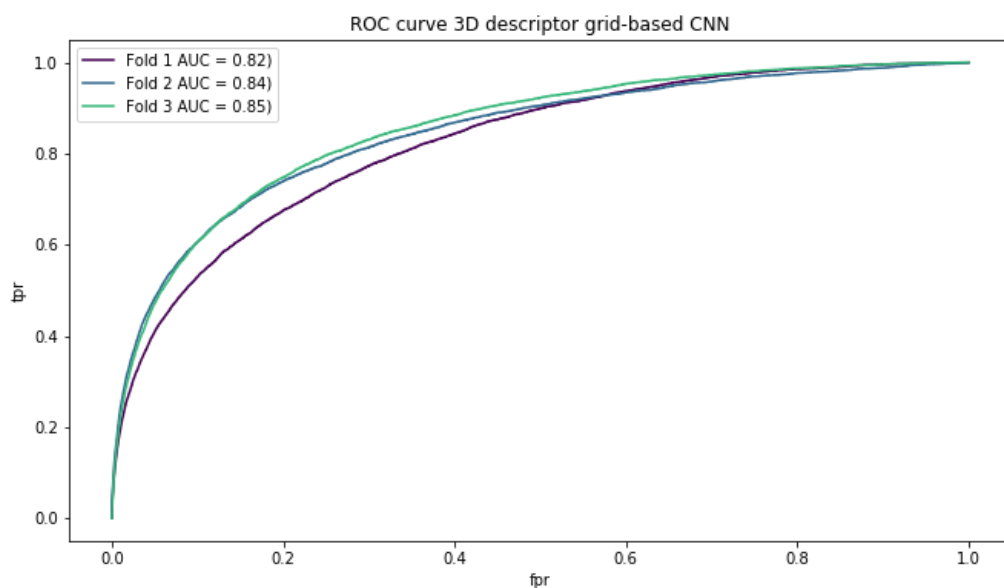


Figure S16: ROC curves for the three folds of the cCCV with our ligand-based 3D reimplementation of the grid-based CNN.



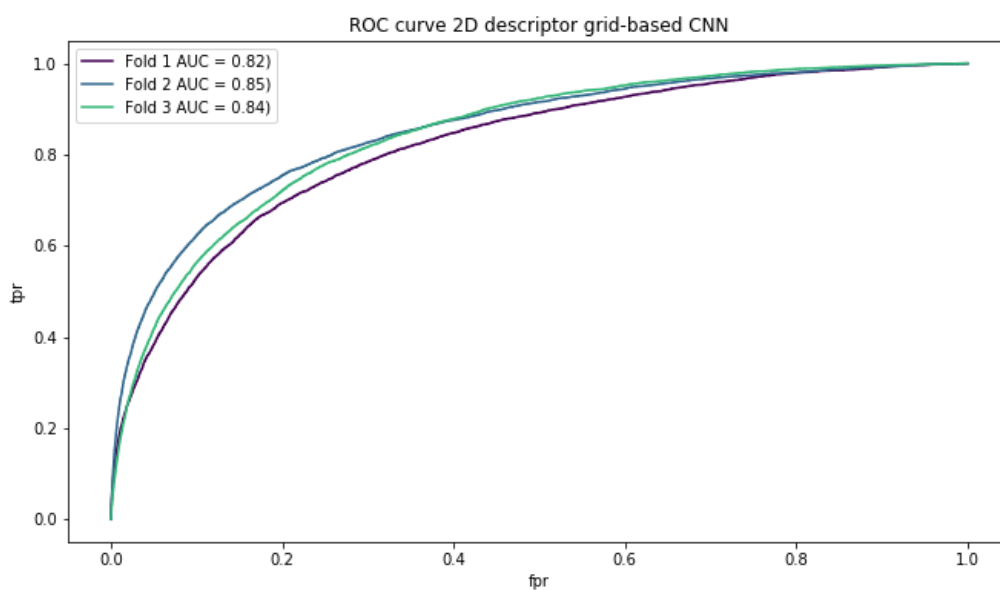


Figure S17: ROC curves for the three folds of the cCCV with our ligand-based 2D reimplementation of the grid-based CNN.

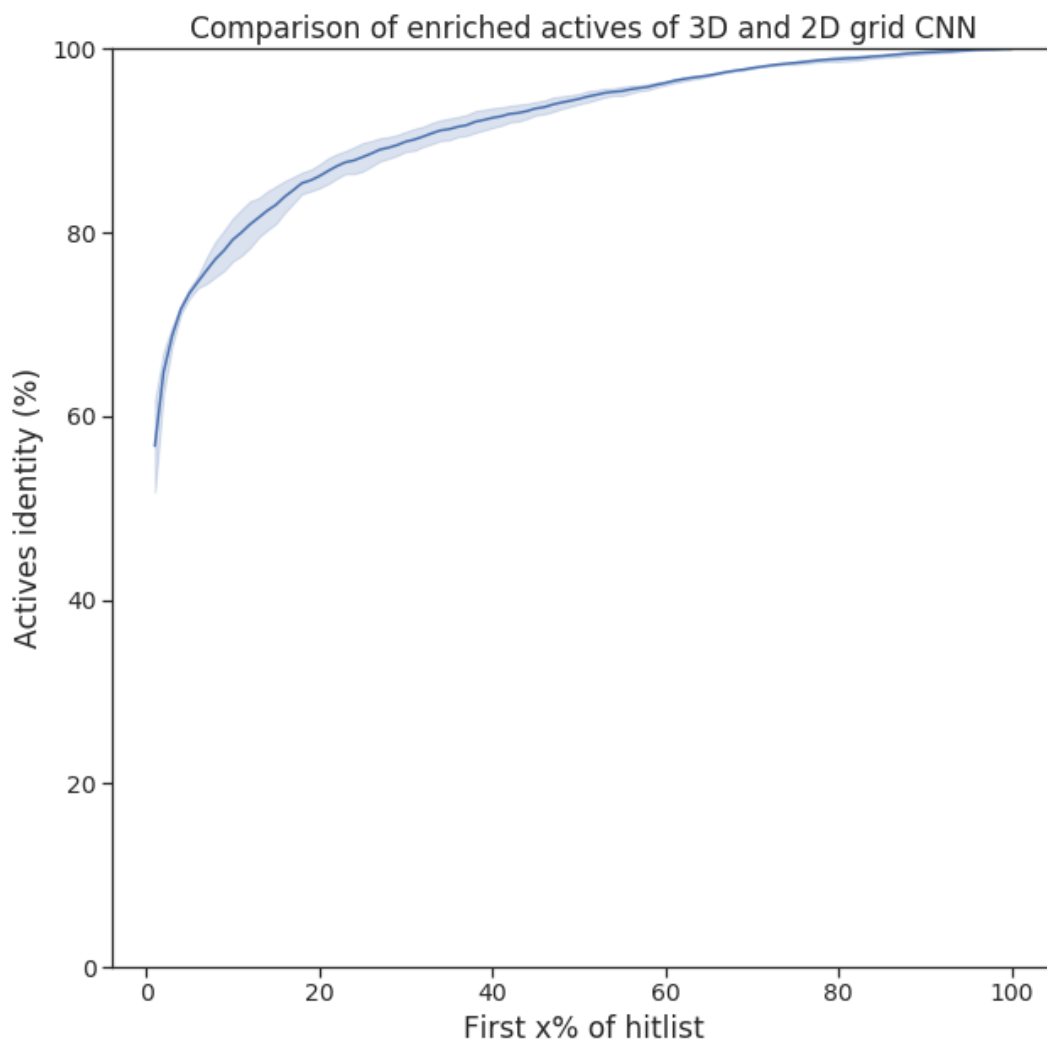


Figure S18: Distribution of overlapping actives between the 3D and 2D version of the grid-based CNN. The mean values over the 3 folds of the cCCV on DUD-E are shown. On the  $y$ -axis the overlap of actives between the hitlists is depicted in dependence of the first  $x\%$  of the ranked molecules, which is shown on the  $x$ -axis. Consequently, one point in the plot describes that in the first  $x\%$  of the hitlists  $y\%$  of the enriched actives are identical between methods. The light colored band shows the standard deviation.

## **D.2 Analyzing structural features of proteins from deep-sea organisms**

- [D2] **J. Sieg**, C. C. Sandmeier, J. Lieske, A. Meents, C. Lemmen, W. R. Streit, and M. Rarey. “Analyzing structural features of proteins from deep-sea organisms”. In: *Proteins: Structure, Function, and Bioinformatics* 90.8 (2022), pp. 1521–1537.

Available: <https://doi.org/10.1002/prot.26337>. Material from [D2].

# Analyzing structural features of proteins from deep-sea organisms

Jochen Sieg<sup>1</sup> | Chris Claudius Sandmeier<sup>1</sup> | Julia Lieske<sup>2</sup> | Alke Meents<sup>2</sup> |  
Christian Lemmen<sup>3</sup> | Wolfgang R. Streit<sup>4</sup> | Matthias Rarey<sup>1</sup>

<sup>1</sup>Universität Hamburg, ZBH - Center for Bioinformatics, Hamburg, Germany

<sup>2</sup>Deutsches Elektronen-Synchrotron DESY, Center for Free-Electron Laser Science, Hamburg, Germany

<sup>3</sup>BioSolveIT GmbH, Sankt Augustin, Germany

<sup>4</sup>Universität Hamburg, Department of Microbiology and Biotechnology, Hamburg, Germany

## Correspondence

Matthias Rarey, Universität Hamburg, ZBH - Center for Bioinformatics, Bundesstraße 43, 20146 Hamburg, Germany.  
Email: [matthias.rarey@uni-hamburg.de](mailto:matthias.rarey@uni-hamburg.de)

## Funding information

Bundesministerium für Bildung und Forschung; German Federal Ministry of Education and Research, Grant/Award Number: 031B0405B

## Abstract

Protein adaptations to extreme environmental conditions are drivers in biotechnological process optimization and essential to unravel the molecular limits of life. Most proteins with such desirable adaptations are found in extremophilic organisms inhabiting extreme environments. The deep sea is such an environment and a promising resource that poses multiple extremes on its inhabitants. Conditions like high hydrostatic pressure and high or low temperature are prevalent and many deep-sea organisms tolerate multiple of these extremes. While molecular adaptations to high temperature are comparatively good described, adaptations to other extremes like high pressure are not well-understood yet. To fully unravel the molecular mechanisms of individual adaptations it is probably necessary to disentangle multifactorial adaptations. In this study, we evaluate differences of protein structures from deep-sea organisms and their respective related proteins from nondeep-sea organisms. We created a data collection of 1281 experimental protein structures from 25 deep-sea organisms and paired them with orthologous proteins. We exhaustively evaluate differences between the protein pairs with machine learning and Shapley values to determine characteristic differences in sequence and structure. The results show a reasonable discrimination of deep-sea and nondeep-sea proteins from which we distinguish correlations previously attributed to thermal stability from other signals potentially describing adaptations to high pressure. While some distinct correlations can be observed the overall picture appears intricate.

## KEYWORDS

deep sea, machine learning, piezophile, protein adaptations, protein pressure adaptations, protein stability, protein structure, protein thermal stability, Shapley values, thermophile

## 1 | INTRODUCTION

Exploiting the properties of proteins from extremophilic microorganisms is a highly active area of research.<sup>1–5</sup> Understanding molecular protein adaptations toward extreme conditions would enable

effective design and engineering of proteins with specific properties, which would have important implications for biotechnological processes in many fields like pharmacology, agriculture, and biofuels production.<sup>1,4–10</sup> While this research objective is around for multiple decades, in recent years, the understanding of extreme environments

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial](https://creativecommons.org/licenses/by-nc/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited and is not used for commercial purposes.

© 2022 The Authors. *Proteins: Structure, Function, and Bioinformatics* published by Wiley Periodicals LLC.

and extremophiles has increased tremendously.<sup>4,11</sup> Increasing efforts in metagenomics for a variety of environments provides a continuously growing number of genomic data from extreme environments,<sup>4,5,12,13</sup> which in turn, yields a rich source of protein data from extremophiles. Not surprisingly, there is a great interest to systematically analyze the data currently available.

Most extreme environments on earth are characterized by multiple extremes.<sup>11</sup> One of the largest and a particular interesting extreme environment is the *deep sea*. It poses multiple extreme conditions on its inhabitants and for this reason, many deep-sea organisms likely exhibit multiple adaptations.<sup>4,9,11,14,15</sup> The extremes in the deep sea are a temperature range from high temperature at hydrothermal vents with up to 120°C to low temperature at the sediment of around 2°C.<sup>8</sup> Especially, elevated hydrostatic pressure is inherent in this biom.<sup>16</sup> The average pressure at the ocean floor is 38 MPa<sup>8</sup> and reaches a maximum of approximately 110 MPa at the Challenger Deep of Mariana Trench.<sup>17</sup> In addition, other stressors like an extreme salt range can be found in the deep-sea sediment and at hydrothermal vents.<sup>18</sup>

Within this study we aim to disentangle the multifactorial aspects of several adaptations in proteins from deep-sea organisms. We are specifically interested to decipher potential protein adaptations to high hydrostatic pressure. The organisms that live under elevated pressure or even need high pressure to grow are called *piezophiles* (or barophiles).<sup>9,15,19,20</sup> Protein adaptations to high pressure are not well-described<sup>9,16,17</sup> and the identification of a molecular signature for high pressure is complicated through other prevailing extremes, for example, the temperature differences of most high pressure environments.<sup>4,15</sup> Different studies also suggest that pressure adaptations might be challenging to detect, because they might be rather subtle and pronounced differently in different protein classes.<sup>4,15</sup> In addition, like for other extremes,<sup>21</sup> protective mechanisms on the cellular level also seem to play a role as an adaptive strategy in some piezophiles,<sup>4,15</sup> which demonstrates that not all pressure adaptations need to be encoded in the protein.

In contrast to pressure, the adaptations to high temperature are by far the most well-described extreme adaptations.<sup>4,9,16</sup> Numerous studies are comparing *thermophiles* and mesophiles.<sup>22–24</sup> Even diverse protein engineering efforts demonstrated the thermal stabilization of proteins.<sup>25</sup> Comparison based studies investigating correlating protein properties between homologous proteins of thermophiles and mesophiles even suggest that a global or “nearly universal” signature of protein thermal adaptations exists.<sup>24</sup> However, while intensively studied a fully precise and global physical picture sufficient to enable large-scale rational protein design is not yet derived.<sup>26</sup> Despite general trends, it seems that an essential bottleneck is that a complex context-dependent combination of multiple factors determines the stability toward extreme temperature.<sup>26</sup>

Equipped with the insights of the last decades on protein adaptations to high temperature we aim to delineate high temperature protein adaptations from potential high pressure adaptations in proteins from deep-sea extremophiles inhabiting both, a high pressure and high temperature environment. By taking this perspective not only

potential pressure adaptations might be deciphered, but even further insights into the still intricate facets of thermal adaptations might be provided.

Currently, not many studies are taking a data-driven perspective to compare characteristics of proteins of piezophiles or deep-sea organisms with their homologs from other environments. Of the existing studies most are comparing amino acid preferences in the proteom based on genome data<sup>16,27–31</sup> while analysis of protein structures on a larger scale are becoming available only recently.<sup>32</sup> Consequently, it becomes interesting to assess the state of experimental protein structures from deep-sea organisms currently available and comprehensively analyze their features regarding adaptations.

In this study, we first establish a dataset of protein structures from deep-sea organisms. We collect names of organisms living in the deep sea from literature and map those names to the Protein Data Bank (PDB).<sup>33</sup> Based on the collected protein structures we assess the current state of available experimental structural protein data of deep-sea organisms. Using the deep-sea protein data we further collect protein structures from organisms not from the deep sea to compile a dataset of orthologous pairs. Protein pairs are selected such that they are related, meaning they are reasonably similar in sequence and structure. This selection aims to enable the isolation of correlating protein features involved in adaptation mechanisms by minimizing evolutionary changes unrelated to extreme adaptations. Subsequently, we analyze a wide range of protein features in a comprehensive top-down machine learning-based feature selection process. The goal is to isolate sequence and structure features that differentiate proteins of deep-sea organisms from proteins of organisms inhabiting different environments. In these experiments we (i) evaluate if there are distinguishing differences between deep-sea proteins and proteins from other environments, like mesophilic and thermophilic organisms and in different protein classes. Then (ii) we determine which features are characteristic and important for differentiation. Finally, (iii) we compare the relevant features derived to already described protein characteristics of thermophiles to assign the observed signals to the individual extremes.

## 2 | MATERIALS AND METHODS

### 2.1 | Collection of deep-sea protein structures

The names of microorganisms found in the deep sea were collected from literature.<sup>17,34–36</sup> The literature for each organism was reviewed manually. The resulting list of organism names was matched to the source organism annotation in the PDB to retrieve protein structures (using the binomial nomenclature and manual review). The list of PDB entries resulting from this protocol can be seen as the currently available experimental deep-sea protein structure data in the PDB. The list of deep-sea organism names collected from literature and the corresponding PDB entries can be found in the Supporting Information (deep\_sea\_species.tsv and deep\_sea\_pdbs.tsv).

## 2.2 | Generation of potential orthologous protein pairs

Based on the collected deep-sea protein structures potential orthologous protein structures in the PDB were searched. Three protein similarity methods are used to identify protein chain pairs that are related in sequence and structure. All sequences from the deep-sea PDB entries are selected based on the `_entity_poly.pdbx_seq_one_letter_code` data field in the CIF files. Exact sequence duplicates from the same PDB entry were removed, for example, from homomeric assemblies to avoid redundant computation. The remaining deep-sea chains are subject to the following protocol to generate orthologous structure pairs from each protein chain.

First, the deep-sea protein chains were used as input to HH-suite<sup>37</sup> (version v3.3.0). HH-suite performs profile–profile alignments of Hidden Markov Models (HMM) to assess the relationship of protein sequences. HH-suite is able to sensitively identify remote homologs with low sequence identity.<sup>38</sup> We used HHblits<sup>37</sup> with UniRef30\_2020\_06<sup>39</sup> to generate HMM profiles for the deep-sea protein chains (using three iterations). The created profiles are then used as input to HHsearch<sup>37</sup> to search PDB70 ([http://wwwuser.gwdg.de/~compbiol/data/hhsuite/databases/hhsuite\\_dbs/](http://wwwuser.gwdg.de/~compbiol/data/hhsuite/databases/hhsuite_dbs/) version 210115) for homologous sequences. From the resulting hits those with a probability  $>50$  and  $E < 10^{-3}$  are kept as potential orthologous partner to a deep-sea chain.

The second phase aims to enrich the protein collection found with HHsearch. Since HH-suite comes with PDB70 a precomputed search database of profiles from a clustered and redundancy reduced version of the PDB a substantial number of sequences from the PDB are removed. However, in this study we are interested in highly similar proteins as long as they are from different organisms. Instead of generating a nonredundant profile database of the whole PDB, which is highly computational intensive, we further enriched our collection of potential protein pairs using the needle tool from EMBOSS suite<sup>40</sup> (version 6.6.0). needle is an implementation of the Needleman–Wunsch algorithm for global sequence alignments. We use needle to compute all pairwise sequence alignments between each deep-sea protein chain and the sequences from the entire PDB. The `gapopen` and `gapextend` parameters were set to 10.0 and 0.5, respectively. From the resulting hits all pairs with a sequence identity  $>25\%$  are kept as potential partner.

In an intermediate step all PDB entries from the potential hits that are present in the deep-sea set are removed from the potential hit list.

Finally, we use TM-align<sup>41</sup> (version 20190822) to compute the final protein pairs by filtering the potential sequence hits from HH-suite and needle by fold similarity of the structure. TM-align computes a three-dimensional (3D) structural protein alignment by minimizing the TM-score. The TM-score is a length dependent measure for global fold similarity of two protein chains. We use a TM-score cut-off of 0.5, which has been reported as a criterion to identify protein chains of the same fold.<sup>42</sup> Any protein chain matched based on sequence which has at least one of the two resulting TM-scores below the threshold is removed.

This protocol is designed to collect protein chain pairs that are at least remotely related in sequence and at the same time share the same overall 3D fold, which finally should provide a high probability that these proteins are evolutionary related and orthologous. The nondeep-sea dataset collected by this protocol will be called decoy dataset in the following.

## 2.3 | Filtering protein structures

PDB entries containing DNA, RNA or chimeric entries or entities, that is, protein chains from different organisms, were removed. Only structures solved with X-ray crystallography, with a resolution better than 3.0 Å are kept. In addition, protein chains with a sequence length  $<50$  are removed. We also remove PDB entries with suspicious source organism names that contain any of the words “synthetic,” “uncultured,” “undefined,” “unidentified,” “artificial,” the symbol “?” or where the organism name is empty.

The list of filtered pairs can be found in the Supporting Information (protein\_pairs.tsv).

For the removal of highly redundant protein chains from our datasets we used the MMseqs2<sup>43</sup> software suite (version 13-45111). The PDB contains many highly similar protein chains. On the one hand, we exploit this redundancy to deduce subtle difference between proteins from different organisms. On the other hand, for proteins from the same organism we want to avoid highly similar chains to avoid skewing the distribution in the subsequent evaluation. For this purpose, MMseqs2 is applied with default parameters (greedy set cover strategy, coverage = 0.8, min\_seq\_id = 0.0) to the sequences of each organism separately. The source organism names were normalized by converting them to lower case and stripping off all words of the name after the first two. For each generated cluster a single representative is selected based on resolution, R-free and largest proportion of resolved residues in the structure. First, the protein chains of the deep-sea dataset were clustered for each source organism separately. Second, the protein chains of the decoy set were clustered also for each source organism separately. After this step we removed identical sequences within the decoy set across all organisms.

## 2.4 | Protein features and structure preparation

In total 25 sequence and 45 structure features were computed and used in the experiments (see Table 1).

For sequence features the relative frequency of the 20 amino acids for each protein is computed as well as the relative frequency of amino acids with the physicochemical properties: polar (SER, THR, TYR, ASN, GLN), hydrophobic (ALA, VAL, LEU, ILE, PRO, TRP, PHE, MET), positively charged (LYS, ARG), negatively charged (ASP, GLU), and aromatic (PHE, TRP, TYR).

Structure features can be grouped into the categories noncovalent molecular interactions, secondary structure features, features of the protein's solvent-accessible surface (SAS) and buried residues, buried

Structure features		Sequence features
Hydrogen bonds	Secondary structure	Amino acid proportions
Hbonds backbone-backbone	Residues in helix	ALA
Hbonds sidechain-sidechain	Residues in strand	ARG
Hbonds backbone-sidechain	Residues in loop	ASN
Acceptors backbone, noninteracting	Solvent-accessible surface (Å <sup>2</sup> )	ASP
Donors backbone, noninteracting	Hydrophobic	CYS
Acceptors sidechain, noninteracting	Polar	GLN
Donors sidechain, noninteracting	Sulfur	GLU
Hbonds surface	Pos. charged	GLY
Acceptors surface, noninteracting	Neg. charged	HIS
Donors surface, noninteracting	Aromatic	ILE
Ionic interactions	Buried residue mass (Da)	LEU
Salt bridges	Hydrophobic	LYS
Anions, noninteracting	Polar	MET
Cations, noninteracting	Sulfur	PHE
Salt bridges surface	Pos. charged	PRO
Anions surface, noninteracting	Neg. charged	SET
Cations surface, noninteracting	Aromatic	THR
Aromatic interactions	Water	TRP
Cation- $\pi$	Buried waters	TYR
Cation- $\pi$ surface	Volume	VAL
Aromatic $\pi$ - $\pi$	Packing density	Hydrophobic residues
Aromatic $\pi$ - $\pi$ surface	Flexibility	Polar residues
Aromatic, noninteracting	Torsional constraints	Pos. charged residues
Aromatic surface, noninteracting	Independent hinge joints	Neg. charged residues
Hydrophobic interactions		Aromatic residues
Hydroph. interactions		
Hydroph. noninteracting		
Hydroph. interactions surface		
Hydroph. interactions surface, noninteracting		

**TABLE 1** List of computed protein structure and sequence features

Note: Counts of these features are computed per protein structure/sequence and used in the machine learning experiments. "Noninteracting" denotes potential interaction sites that are able to form a specific interaction but in the given state of the structure do not participate in an interaction.

waters, volume as well as rigidity/flexibility. Furthermore, features within these categories are combined to create new features.

All structure features except those in the category volume and flexibility were computed using the NAOMI<sup>44</sup> library. The protonation states of each protein chain is determined with Protoss.<sup>45</sup>

Hydrogen bonds and ionic interactions were computed using the definition and scoring within Protoss.<sup>45</sup> Salt bridges were only computed between the residues ASP, GLU with LYS, and ARG.

Cation- $\pi$  and aromatic  $\pi$ - $\pi$  interactions were computed with the NAOMI library. Cation- $\pi$  interactions were considered between LYS

and ARG with PHE, TYR or TRP with a distance threshold of  $<6$  Å between cation and ring center, as well as, a maximal deviation of 2 Å of the cation from the normal defined at the ring center on the ring plane.  $\pi$ - $\pi$  Interactions were calculated between PHE, TYR, and TRP with a maximal distance of 5.5 Å between ring centers.

Hydrophobic/lipophilic atoms were identified using the same definition as the JAMDA scoring function.<sup>46</sup> A hydrophobic contact is predicted between two hydrophobic atoms if for their distance  $d$  applies  $vdW_{\text{sum}} < d < 1.75vdW_{\text{sum}}$ . Hydrophobic contacts were only considered if they are between the side chains of ALA, VAL, LEU, ILE, PRO, TRP, PHE, and MET.

Besides the number of observed interactions we also consider potential interaction sites such as atoms, lone pairs, and  $\pi$  electron systems that are able to form a specific interaction but in the given state of the structure do not participate in an interaction. We term these “noninteracting” in the following. An example are hydrogen bond acceptors or donors which are not involved in a hydrogen bond. We also consider interactions and noninteracting sites at certain locations in the protein structure, like on the protein surface or in sidechain and backbone for hydrogen bonds. All features in the category noncovalent interactions are represented by counts of each feature and are normalized by the number of all residues in the structure.

Secondary structure elements were computed with an implementation of the DSSP algorithm<sup>47</sup> within the NAOMI framework. Residues were assigned to the structure elements helix or strand based on the computation or as loop if neither helix nor strand was predicted. Secondary structure features are also normalized by the number of residues in the whole protein structure.

An SAS representation was calculated using the respective algorithm in HYDE<sup>48</sup> from which features of different atoms on the surface could be derived (e.g., noncovalent interaction features located at the surface). The SAS is computed based on heavy atoms only. From the surface representation we derive the proportion of surface area made up by residues with the physicochemical properties: hydrophobic, polar, positively and negatively charged, aromatic- and sulfur-containing residues (MET, CYS). The definitions of those properties are the same as for the sequence features, except sulfur containing residues. All surface area-based features are normalized by the surface area of the whole protein.

Analogously we compute the physicochemical property distribution of residues buried within the protein (without surface contact). In addition to simply counting we weight the counts by the molecular weight (MW) to capture the size differences of single amino acids. These features are normalized by the MW of the whole protein.

The number of buried waters was used as a descriptor. A water molecule was considered buried if  $< 1/3$  of its oxygen's surface is part of the surface that is defined by the heavy atoms of the protein and waters of the complex. The surface was computed with the respective algorithm in HYDE.<sup>48</sup>

Packing density was computed with ProteinVolume<sup>49</sup> (version 1.3) as the van der Waals volume divided by the total volume of the structure in solution.

Rigidity/flexibility descriptors were used from MSU ProFlex (<https://github.com/psa-lab/ProFlex> version 5.2, formally called FIRST<sup>50</sup>). Specifically, we used the predicted number of torsional constraints and hinge joints as features to describe the global rigidity of the structure. Both features are normalized by the number of residues in the respective protein.

## 2.5 | Machine learning-based feature evaluation

### 2.5.1 | Feature selection scheme

To evaluate the collected protein structure pairs for expressive differences we use machine learning-based feature selection. Supervised

machine learning methods optimize mathematical functions to learn the discrimination of labeled data points. In our case the goal is to learn a model that differentiates between protein structures from deep-sea organisms and protein structures from nondeep-sea organisms. Correspondingly, the labels we use in our experiments are “deep sea” and “decoy” representing the deep-sea and decoy protein dataset, respectively. Machine learning algorithms are effective for capturing correlations not only in single features, but in combinations of features.

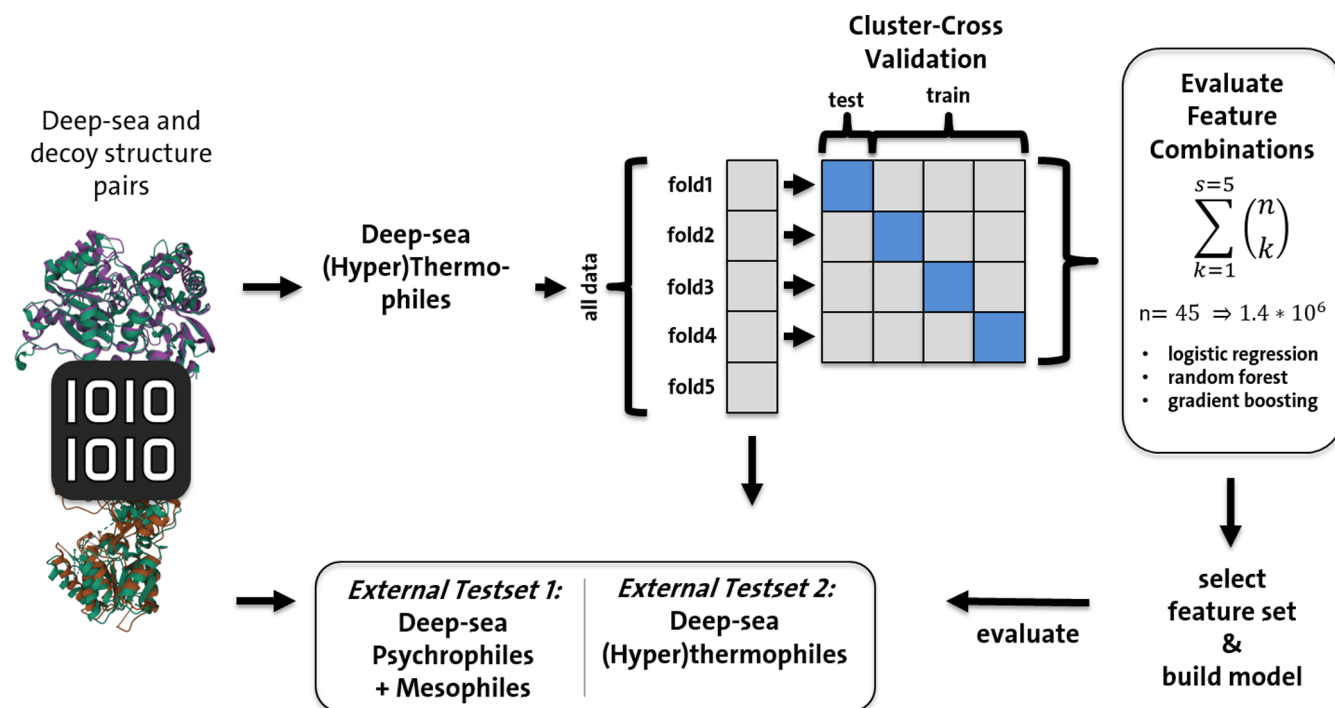
Figure 1 shows the workflow of our feature evaluation experiments. Initially, we split the collected protein pairs based on the optimal growth temperature (OGT) of their deep-sea source organism. The protein pairs with proteins of (hyper)thermophilic deep-sea organisms (called HT-group) will be used for feature selection. In contrast, pairs with proteins of deep-sea psychrophiles and mesophiles (called PM-group) will be used as an external test set.

To be able to measure whether combination of features are predictive across different protein families we employ a cluster cross-validation strategy. For the creation of the folds we cluster the protein sequences of both deep-sea and decoy samples of the HT-group dataset with MMseqs2. We use the connected component clustering mode with a coverage of 0.5 and a min\_seq\_id of 0.3, which is more suited to capture transient graph connections and therefore should assign more remote homologs in the same cluster. Subsequently, we ensure that all orthologous pairs are kept in the same cluster by adding new edges for the orthologous pairs to the graph representing the generated clustering of MMseqs2. On this new graph we compute the connected components again to obtain the final clusters. A fixed number of folds is then generated by assigning the clusters to five folds by trying to keep the size of the folds equal. The folds can be found in the Supporting Information (folds.tsv).

While four of the five cluster-folds are used for feature selection the remaining cluster-fold is used as another external test set. With this we evaluate selected features on a fold of deep-sea proteins sequentially dissimilar to those used in feature selection but which are also from (hyper)thermophiles. The performance on the external test sets is determined by models trained on all four-folds of the cross-validation with the five best performing features from the feature selection. For the PM-group, any decoy chains present in both the PM-group and the four-folds are removed from the PM-group before evaluation.

To further investigate the relevance of the decoys' origin we split the folds based on the decoy source organism. We will call the set of all pairs the DecoyAll set (equivalent to the HT-group). Other decoy sets are specifically selected subsets. The MesoModel set contains pairs with mesophilic model organisms like *Homo sapiens* and *Escherichia coli*. The ThermoAll set contains the structures from thermophilic organisms from literature<sup>24,51</sup> of which the ThermoModel set is a subset that only contains proteins of well studied thermophilic organisms for example *Thermus thermophilus* and *Thermotoga maritima*. The feature selection workflow illustrated in Figure 1 is separately conducted for the four different datasets. A list of the decoys source organisms in each group can be found in the Supporting Information (decoy\_subsets.tsv).





**FIGURE 1** The workflow of the feature selection experiments. Initially, the structure pair data is split based on the deep-sea source organism. The pairs of deep-sea (hyper)thermophiles are used to select important protein features with machine learning. Exemplary, for the  $n = 45$  structure features there are 1.4 million feature combinations which are evaluated. The selected features are evaluated on two external test sets

We perform feature selection with wrapper methods<sup>52</sup> to evaluate different feature combinations and find the optimal feature set for the binary classification task within a large fraction of all possible feature sets. In this feature selection scheme all combinations of a list of single features are enumerated and machine learning models are trained and evaluated with each feature set in the cluster cross-validation. The number of possible feature sets is  $2^n - 1$ , where  $n$  is the number of features. Enumerating all possible feature sets is infeasible. Therefore, we only evaluate feature sets up to a size of  $s = 5$  features (see Figure 1). Correspondingly, for the  $n = 45$  structure features the number of feature sets is  $\sum_{k=1}^{s=5} \binom{45}{k} = 1,385,979$ . The threshold of 5 was chosen as a trade-off between computation time and expected predictive power.

Machine learning algorithms are employed from scikit-learn<sup>53</sup> (version 0.23.2). We use the linear method logistic regression (solver='lbfgs', max\_iter = 10 000) as well as the nonlinear methods random forest classifier (n\_estimators = 200) and the gradient boosting classifier (n\_estimators = 200) which are both based on ensembles of decision trees. The linear method is comparably simple and will be used as baseline method. The two nonlinear ensemble methods are able to capture more complex relationships between features.

The measure of choice to assess the prediction performance of the trained machine learning models is to compute the area under the receiver operating characteristic curve (ROC AUC)<sup>54</sup> on the test datasets. The ROC AUC is a threshold free measure assessing the ability of a model to rank positive instances relative to negative instances.

This metric provides a value between 0 and 1, where 0.5 represents the random baseline. A useful statistical property is that a ROC AUC of a classification model is equivalent to the probability to rank a randomly selected positive sample higher than a randomly selected negative sample.<sup>54</sup> Consequently, in our experiments, a trained model achieving a ROC AUC of 0.70 would correspond to a 70% probability to rank a randomly selected deep-sea protein before a randomly selected decoy protein based on the given test set.

For the experiments training and test sets were normalized columnwise using the respective training set. We used  $z_{ij} = (x_{ij} - \mu_j) / \sigma_j$  to compute the normalized feature value  $z_{ij}$  from each feature value  $x_{ij}$ . Here,  $i$  denotes the row and  $j$  the column in the feature matrix;  $z_{ij}$  is computed by subtracting the mean  $\mu_j$  of the column from  $x_{ij}$  and divide by the column's standard deviation  $\sigma_j$ .

## 2.5.2 | Feature attribution scheme

We follow two approaches to not only select predictive features, but to attribute relevance through prediction performance to single features. With this we want to validate our approach and interpret features in the context of protein adaptations to extreme conditions. The basis for these interpretation approaches is the enumeration and evaluation of feature combinations.

In the first approach we will simply evaluate which feature combinations are sufficient to achieve a notable performance in the validation scenario. Small feature subsets, even single features, achieving a

comparable or better performance relative to larger sets, like the set of all features indicate highly relevant features in the smaller set.

For the second approach we use the framework of *Shapley values*<sup>55</sup> from cooperative game theory. Shapley values provide a concept to attribute contributions single features make in combination with other features to the individual single features. The Shapley value of a feature  $i$  is defined as the weighted sum of the marginal contributions  $i$  makes when  $i$  is included in a feature set  $S$ :

$$Sh_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(n-|S|-1)!}{n!} (v(S \cup \{i\}) - v(S)),$$

where  $N$  is the set of all features,  $S$  is a subset of  $N$ ,  $n$  is the total number of features, and  $v$  is a function that maps a feature set to a real number. In our experiments, we define  $v$  to map a feature set to the ROC AUC value the feature set generates in our experiment. Shapley values can be computed in polynomial time, for example, through sampling.<sup>56</sup> Here, in this study, similar to a sampling approach, we compute the contribution of each feature  $i$  by considering only the marginal contributions from a sample of all possible coalitions (in our case the subset of all feature combinations we enumerate). Specifically, we use the resulting mean ROC AUC values from the cluster cross-validation experiments of all enumerated feature combinations to attribute contributions to each feature  $i$  in terms of ROC AUC. In other words, we will compute the Shapley value for each feature  $i$  using the mean ROC AUC in the cluster cross-validation experiments of all enumerated feature sets. Features with high resulting contribution values would indicate that these features hold valuable information for the classification model.

### 3 | RESULTS AND DISCUSSION

#### 3.1 | Deep-sea protein data in the PDB

The dataset created contains protein structures from 25 deep-sea organisms. In total, 1281 PDB entries could be retrieved. A comprehensive overview of the distribution of organisms and number of proteins is shown in Table 2.

The organisms listed in Table 2 were collected in depths greater 1000 m or an elevated optimal growth pressure was reported. Based on the collection depth and the linearly increasing pressure through the water column the approximate pressure range the retrieved organisms inhabit is 10 MPa to roughly 110 MPa (starting from 1000m depth). There are 14 Bacteria, 10 Archaea, and 1 Eukarya in the dataset. The reported OGTs (or preferred temperature range) of the organisms are between 2°C and 98°C. This illustrates both extremes of hyperthermophilic and psychrophilic organisms that inhabit the deep sea. Following the definition of Hait et al.<sup>24</sup> for hyperthermophilic (HT) ( $T \geq 75^\circ$ ), thermophilic (T) ( $50^\circ \leq T < 75^\circ$ ), mesophilic (M) ( $24^\circ \leq T < 50^\circ$ ), and psychrophilic (P) ( $T < 24^\circ$ ) there are 10 hyperthermophilic, 5 thermophilic, 3 mesophilic, and 6 psychrophilic organisms in the dataset.

The distribution of protein structures collected from the PDB is imbalanced between the organisms. This reflects the imbalanced organism distributions in the PDB itself and is not surprising since research interest, accessibility and cultivation conditions are also different for different organisms. Most PDB entries that have been retrieved are from *Pyrococcus horikoshii* with 562 proteins structures (44%) and *Methanocaldococcus jannaschii* with 359 structures (28%). Besides the proportions of proteins of the dataset that come from individual organisms it is interesting to look at proportions of groups of organisms. The 10 hyperthermophilic Archaea, for example, make up the majority of proteins in the dataset (1078 PDB entries, 84%). In addition, 139 protein entries are from thermophilic organisms which means that 95% of proteins are from organisms living under elevated temperature. In contrast, only 38 proteins (3%) are from psychrophilic organisms and 25 (2%) from mesophilic organisms.

These results show that the current state of available protein data of deep-sea organisms in the PDB is skewed toward individual organisms and toward hyperthermophilic Archaea. Therefore, it is unlikely that the currently available experimental protein structure data on deep-sea proteins is representative for the whole population of proteins from the deep-sea habitat. However, the data available provides a reasonable basis to compare the proteins of deep-sea (hyper)thermophiles to those of organisms from other environments.

#### 3.2 | Protein pair generation

Protein chains of the retrieved deep-sea proteins were used to identify related protein chains from nondeep-sea organisms from the PDB named decoys in the following (see Section 2).

For 1243 deep-sea protein chains (1204 PDB entries) at least one decoy chain could be identified. In total, 19 173 decoy chains were found in the PDB by the protocol (see protein\_pairs.tsv in the Supporting Information). The matching of deep-sea and decoy chains in this step can be represented by a bipartite graph. In this set of pairs a single deep-sea chain can be paired with multiple decoy chains and a decoy chain can be paired with multiple deep-sea chains.

Highly redundant protein chains were removed with MMseqs2 as described in the methods section. The final dataset contains 501 deep-sea chains and 8200 decoy chains that come from 20 different deep-sea and 1379 decoy organisms and form 17 148 chain pairs. According to the applied clustering criteria 60% of the deep-sea chains were highly similar and therefore redundant. This dataset was then grouped into connected component clusters for cluster cross-validation (see Section 2) and is the basis for the machine learning experiments in the following sections.

In Figure 2A the distribution of sequence and structure similarity of the pairs is depicted. We calculated the mean TM-score (mTM-score) as the mean of the two resulting TM-scores from each alignment. The distribution of this structural measure of similarity shows an expected value of 0.69 for the average chain pair indicating considerable structural similarity.<sup>42</sup> In contrast, the mean sequence identity as calculated by TM-align is 0.19, which alone would not suffice to

**TABLE 2** Deep-sea organisms from literature with protein structures in the PDB

Species name	Depth (m)	T (°C)	P (MPa)	Domain	T-phile	PDB entries
<i>Pyrococcus horikoshii</i> <sup>57</sup>	1395	98	30 <sup>19</sup>	Archaea	HT	562
<i>Methanocaldococcus jannaschii</i> <sup>58</sup>	2600	85	75 <sup>59</sup>	Archaea	HT	359
<i>Geobacillus</i> sp. HTA-462 <sup>60</sup>	10 897	55–75		Bacteria	T	113
<i>Pyrococcus abyssi</i> <sup>61</sup>	2000	96	20–40	Archaea	HT	91
<i>Methanopyrus kandleri</i> <sup>62</sup>	2000	98	20 <sup>17</sup>	Archaea	HT	39
<i>Shewanella loihica</i> <sup>63,64</sup>	1325	18		Bacteria	P	20
<i>Methanothermococcus thermolithotrophicus</i> <sup>35,65</sup>	0.5	65	50	Archaea	T	17
<i>Thermococcus thioreducens</i> <sup>66</sup>	2300	83–85		Archaea	HT	16
<i>Oceanobacillus iheyensis</i> <sup>67</sup>	1050	30	30 (max)	Bacteria	M	14
<i>Persephonella marina</i> <sup>68</sup>	2507	73		Bacteria	T	7
<i>Photobacterium profundum</i> <sup>69–71</sup>	2551 <sup>69,71</sup> /5110 <sup>70</sup>	15 <sup>69,71</sup> /8–12 <sup>70</sup>	28 <sup>69,71</sup> /10 <sup>70</sup>	Bacteria	P	6
<i>Idiomarina loihiensis</i> <sup>72</sup>	1296	4–46		Bacteria	M	6
<i>Marinactinospora thermotolerans</i> <sup>73</sup>	3865	28		Bacteria	M	5
<i>Shewanella benthica</i> <sup>74</sup>	10 898	cold	70	Bacteria	P	4
<i>Pyrococcus yayanosii</i> <sup>75</sup>	4100	98	52	Archaea	HT	4
<i>Moritella profunda</i> <sup>76</sup>	2815	2	22	Bacteria	P	4
<i>Shewanella violacea</i> <sup>77</sup>	5110	8	30	Bacteria	P	3
<i>Thermovibrio ammonificans</i> <sup>78</sup>	2500	75		Bacteria	HT	3
<i>Thermococcus chitonophagus</i> <sup>79</sup>	2600	85	23	Archaea	HT	2
<i>Caldithrix abyssi</i> <sup>80</sup>	3000	60		Bacteria	T	1
<i>Thermosipho melanesiensis</i> <sup>81</sup>	1832–1887	70		Bacteria	T	1
<i>Cryptococcus liquefaciens</i> N6 <sup>82,83</sup>	6500			Eukarya		1
<i>Shewanella piezotolerans</i> WP3 <sup>84</sup>	1914	15–20	20	Bacteria	P	1
<i>Palaeococcus ferrophilus</i> <sup>85</sup>	1338	83	30	Archaea	HT	1
<i>Methanocaldococcus vulcanius</i> <sup>86</sup>	2600	80		Archaea	HT	1
						1281

Note: The number of PDB entries corresponds to the number after filtering and with redundancy. The depth column shows the sample depth in the sea. The T column shows the optimal growth temperature (OGT) or the preferred temperature range if not indicated differently. The P column shows the optimal growth pressure or preferred pressure range if not indicated differently. The T-phile column indicates the classification in hyperthermophile (HT) ( $T \geq 75^\circ$ ), thermophile (T) ( $50^\circ \leq T < 75^\circ$ ), mesophile (M) ( $24^\circ \leq T < 50^\circ$ ), and psychrophile (P) ( $T < 24^\circ$ ). Abbreviation: PDB, Protein Data Bank.

indicate an evolutionary relation. The ranking in Figure 2B lists the most frequent source organisms in the decoy dataset based on the number of protein chains. Figure 3 illustrates the 3D protein structures of two examples of deep-sea protein chains and their paired structures.

### 3.3 | Machine learning-based feature evaluation

#### 3.3.1 | Data Preparation

The compiled protein pair dataset was processed for feature selection (see Figure 1). The data was first split based on the deep-sea organisms in the HT-group and PM-group. The HT-group was clustered and grouped into cross-validation folds. From the folds subsets were generated based on the source organisms of each decoy in the protein

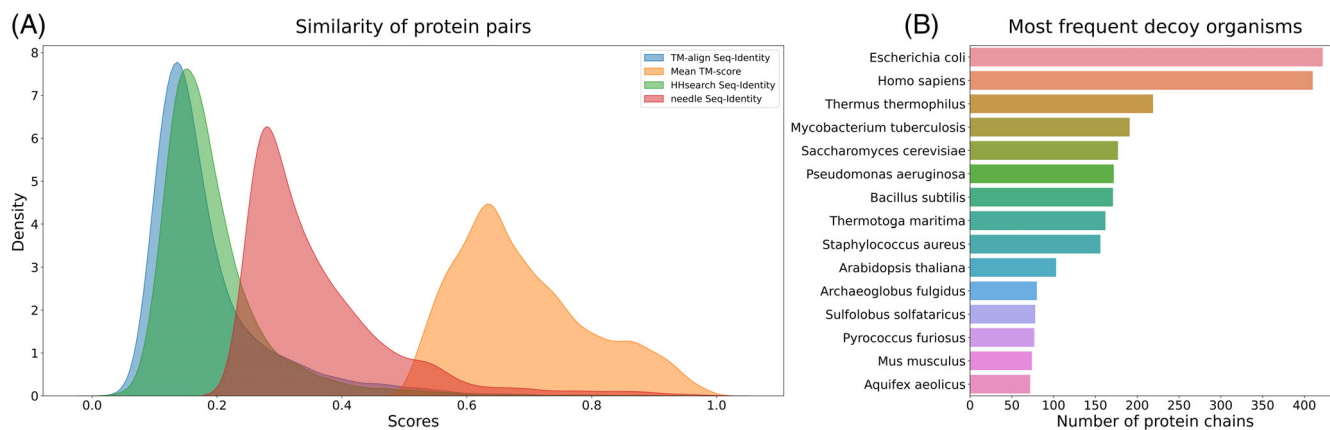
pairs. The composition of the resulting four data subsets of the HT-group are listed in Table 3. Each of these datasets is evaluated separately with the feature selection workflow in the following.

#### 3.3.2 | Can deep-sea proteins be distinguished?

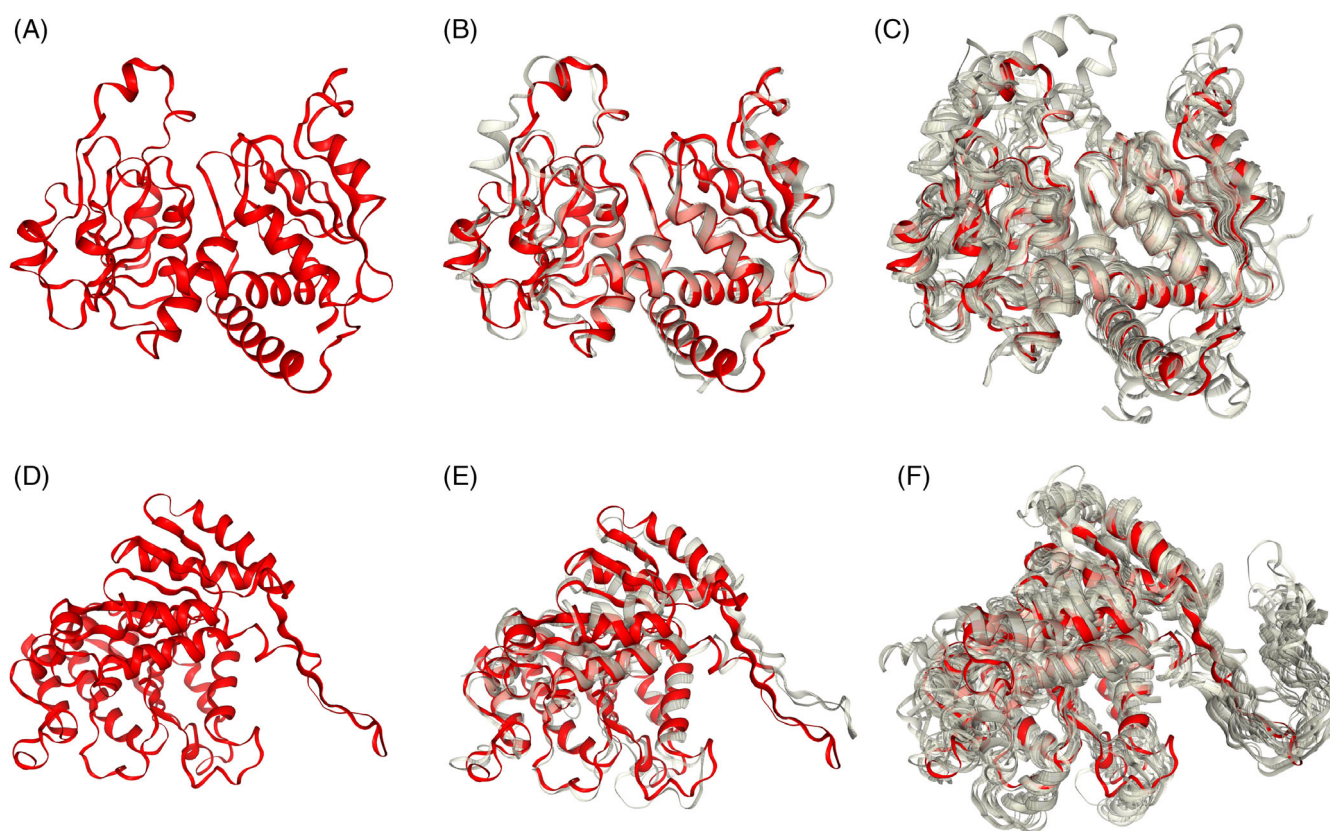
As a first analysis we investigate the extent deep-sea proteins can be predicted and distinguished from orthologs, but not which specific features are distinguishing them. We will look into the specific features in the next sections.

The prediction performance of feature sets in the cross-validation experiment and on the external test sets are depicted in the first and second row in Figure 4, respectively.

The cluster cross-validation results in Figure 4 show the distribution of obtained mean ROC AUC over all enumerated feature



**FIGURE 2** Distributions of the protein pair dataset. (A) The distributions of similarity scores between the protein chain pairs of the deep-sea and decoys dataset. The mean TM-score (orange) is calculated by taking the mean of the two resulting TM-scores for each protein structure alignment. (B) Lists the 15 most frequent source organisms in the decoy dataset based on the number of protein chains



**FIGURE 3** Exemplary structures of protein pairs. Structures from deep-sea organisms are colored in red and decoys in gray. The first row shows pairs generated for the aspartate carbamoyltransferase (1MI4 chain A) from the deep-sea organisms *Pyrococcus abyssi* in red (A). (B) The structure paired with an ornithine carbamoyltransferase (1PVV chain A) from *Pyrococcus furiosus*. (C) structure ensemble with 10 different paired protein chains. The second row shows pairs collected for the 3-isopropylmalate dehydrogenase (3VMK chain A) from the deep-sea organism *Shewanella benthica* in red (D). (E) The pair with the 3-isopropylmalate dehydrogenase (1CM7 chain A) from *Escherichia coli*. (F) Structure ensemble with 10 different paired protein chains. Structure alignments have been computed with TM-align and are visualized with NGL.<sup>87</sup> Opacity of decoy structures has been set to 0.6 for visualization purposes

combinations in the four-fold cluster cross-validation. Feature sets are plotted by their size, meaning at each  $x$  position the distribution of mean ROC AUC of all feature sets containing  $x$  features is shown. For

example, a data point in the column  $x = 1$  shows the mean ROC AUC achieved by one of the three used machine learning algorithms in the cross-validation by using only a single feature, for example, the



proportion of ALA in the protein. Correspondingly, in the  $x = 2$  column performances of feature sets containing two features, for example the proportion of ALA and VAL is depicted. Figure 4 illustrates the performance of all three used machine learning algorithms at once. The performances per algorithm are comparable and can be found in Figures S1–S3 in the Supporting Information.

Three general trends of sequence and structure features can be observed from the cross-validation results. First, the best and average performance to distinguish deep-sea proteins from their orthologs increases by including more features in almost all cases. Second, a small number of  $\leq 5$  features already yield results similar in comparison to using all features. Even single features yield considerable prediction

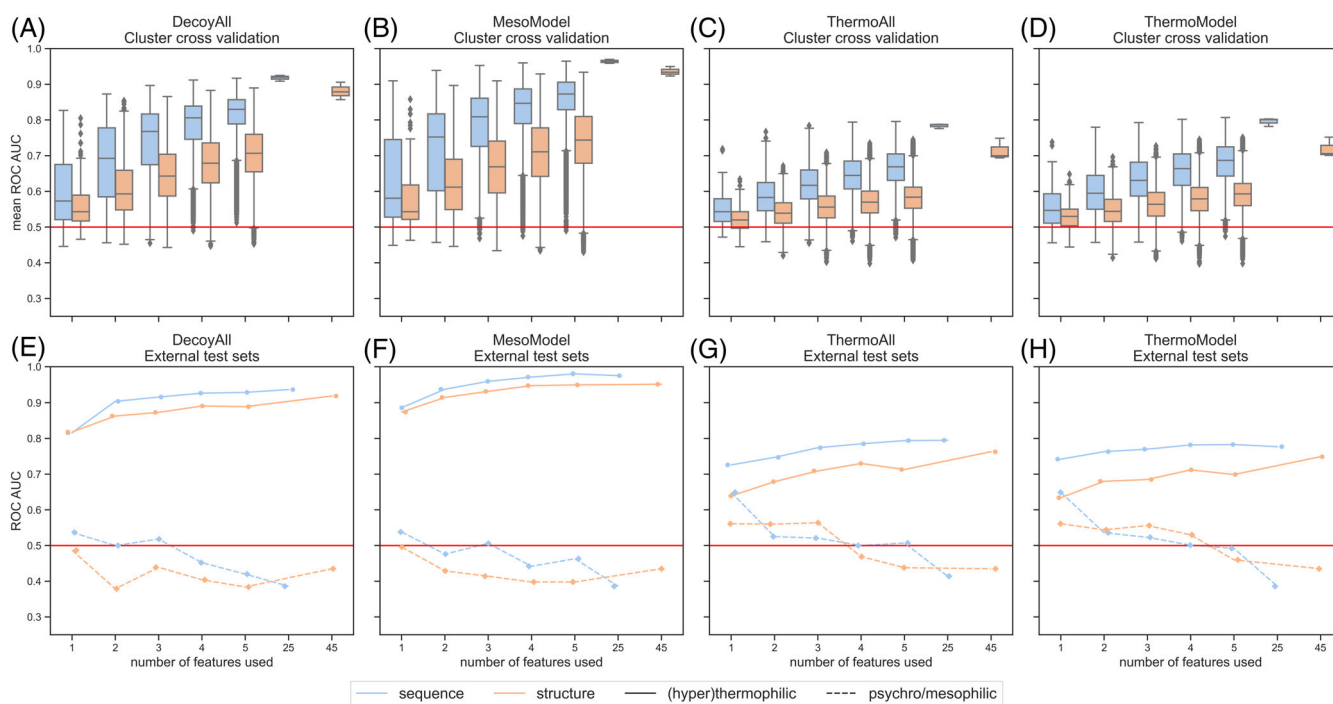
performance in certain cases. Finally, the prediction performance is observed to be higher when using sequence features instead of structure features. In contrast, to these general trends the prediction performance between decoy datasets differs. For the DecoyAll (A) and MesoModel (B) decoy sets the best mean ROC AUC performance is  $>0.90$  in both sequence and structure. This is an almost perfect class separation. Substantially lower, but also reasonable predictive are the results on the ThermoAll (C) and ThermoModel (D) set with a best mean ROC AUC of 0.81 for sequence and 0.75 for structure features.

The second row of Figure 4 shows the results on the two hold-out external test sets. Models were generated for the five best performing feature sets from the feature selection for each feature set

**TABLE 3** Overview of the protein pair datasets used for feature evaluation

Decoy Set	Deep-sea species	Decoy species	Deep-sea proteins	Decoy proteins
All decoy organisms (DecoyAll)	14	1343	474	7699
Mesophilic model organisms (MesoModel)	12	7	361	1215
All thermophilic organisms (ThermoAll)	11	60	398	931
Thermophilic model organisms (ThermoModel)	9	8	370	684

*Note:* The DecoyAll set contains all deep-sea/decoy protein pairs (equals the HT-group). The other rows are protein pair subsets selected on the decoys source organisms. The MesoModel datasets contains protein pairs with mesophilic model organisms like *Homo sapiens* and *Escherichia coli*. The ThermoAll dataset contains pairs with decoy proteins of thermophiles from literature and the ThermoModel dataset contains pairs with decoy proteins of model thermophiles like *Thermus thermophilus* and *Thermotoga maritima*.



**FIGURE 4** Prediction performance of protein feature combinations on the different protein pair datasets. The first row shows the distribution of mean ROC AUC in the cluster cross-validation. Performance is depicted for all used machine learning methods over all enumerated feature combinations for the four different protein pair datasets (A–D). The x-axis shows the number of used features in the feature sets. The performance achieved with protein sequence and structure features is depicted separately. The two rightmost entries on the x-axis show the performance with all sequence and structure features, respectively. The second row shows the single best obtained prediction performance on the external test sets from the five best features from feature selection for each of the four protein pair dataset (E–H). The red horizontal line illustrates the random performance baseline of 0.5

size and for each algorithm, respectively. Only the best performance over all machine learning algorithms and feature sets are shown. Results for all machine learning algorithms and feature sets can be found in Figures S4–S9 in the Supporting Information.

The results on the hold-out cluster-fold from (hyper)thermophiles, of the HT-group, show a prediction performance that is comparable to the top performance achieved in the cross-validation in all four experiments. In contrast the performance on the deep-sea proteins from psychro- and mesophiles, the PM-group, is considerably lower and in most cases not better than a random prediction.

The results of both the cluster cross-validation and the external test sets show that deep-sea proteins can be successfully separated from orthologous proteins of different environments. However, the extent of this separation depends strongly on the specific source environment of deep-sea organisms and decoy organisms. Noteworthy, on all datasets but the hold-out PM-group data good to perfect prediction performance could be achieved with both sequence and structure features. Consequently, there are systematic differences across the dissimilar protein clusters of (hyper)thermophilic deep-sea organisms. For the DecoyAll and MesoModel dataset these differences are global and very easy to capture, they are even encoded in single features. In contrast, systematic differences in the ThermoAll and ThermoModel sets are less obvious and not global. Furthermore, the poor results on the hold-out PM-group suggest that the most relevant features to recognize deep-sea proteins from (hyper)thermophiles are not necessarily relevant to predict proteins from deep-sea psychrophiles and mesophiles. Different adaptation strategies might exist between these groups. However, with only 27 structures the

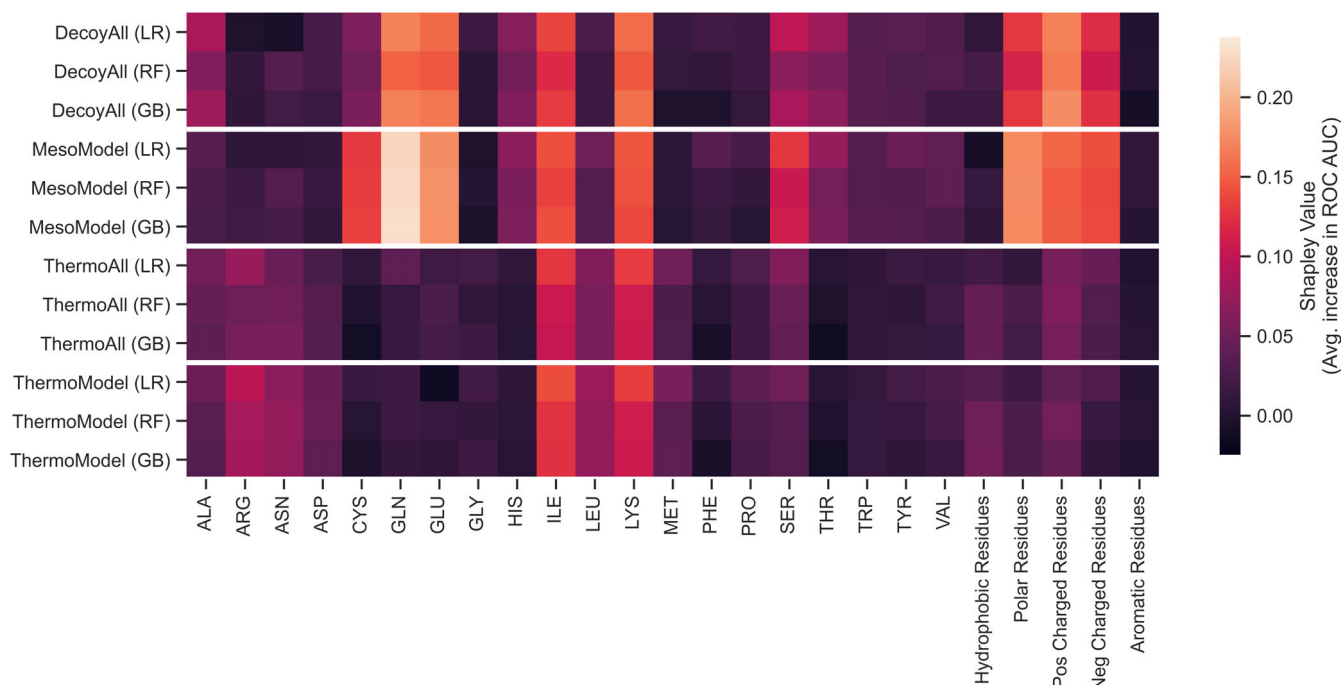
external test dataset on proteins from deep-sea psychro/mesophiles is probably not comprehensive enough for conclusions.

### 3.3.3 | Which features are important?

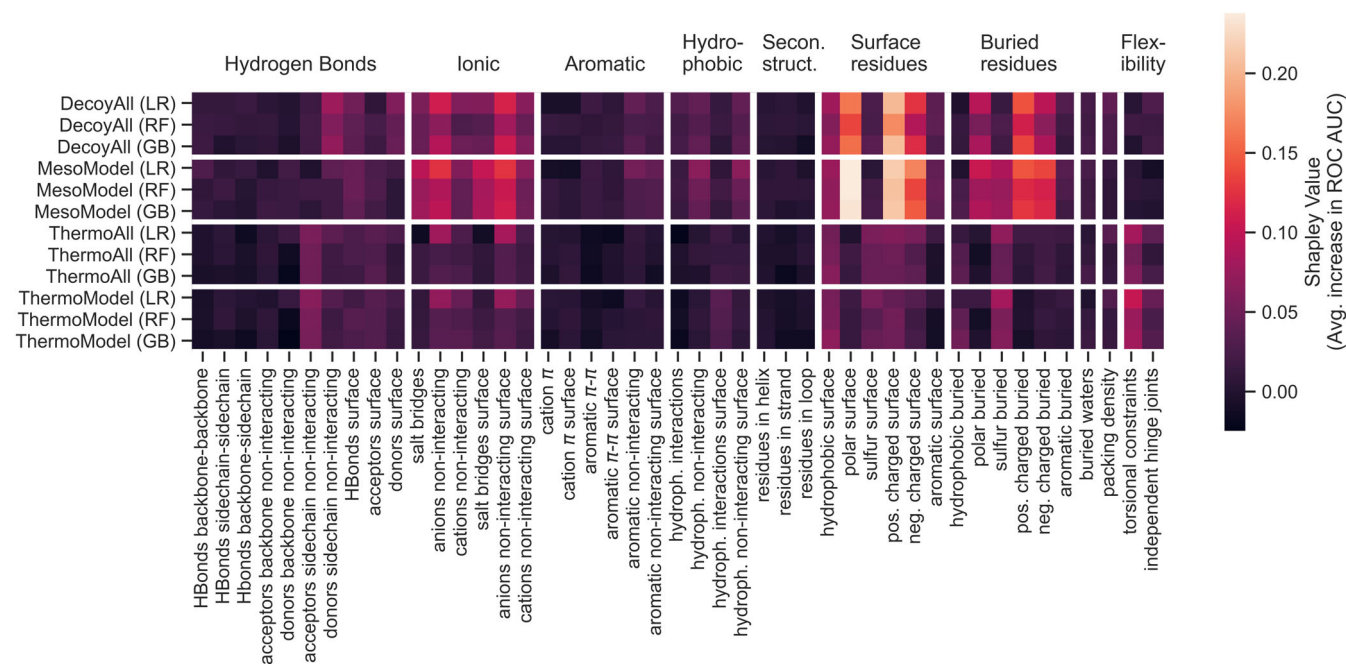
To determine the features important for predictions we use the attributing schemes described in the methods section, mainly the Shapley values analysis. The Shapley values of all features in the different experiments are depicted in Figures 5 and 6. We also provide the standard deviations of the marginals in Figures S10 and S11. In addition, the distribution of each individual feature in all four datasets can be found in Figures S12–S25 as well as a list of the best performing feature sets from Figure S4 (best\_features files).

The Shapley value plots in Figures 5 and 6 illustrate the average ROC AUC contributions each individual feature makes for sequence and structure features, respectively. More precisely, a cell in the plot shows the average ROC AUC contribution a specific single feature makes on a specific dataset for a certain machine learning algorithm in the cluster cross-validation.

Distinct contributions of certain features can be observed from the results. Notably, these distinct features are in accordance with the most predictive feature sets in the cross-validation in the respective experiments (see best\_features.csv). While the contributions within each experiment are relatively consistent for all three machine learning algorithms (minimal Pearson's correlation coefficient of 0.93 for sequence; 0.78 for structure) the important features differ between the four experiments. The feature contributions in the DecoyAll and



**FIGURE 5** Average ROC AUC contributions of each individual sequence feature over all enumerated and evaluated feature sets in the cluster cross-validation. Contributions are computed as the mean of the marginals based on Shapley values. Features are depicted on the x-axis and datasets with the machine learning methods logistic regression (LR), random forest (RF), and gradient boosting (GB) on the y-axis



**FIGURE 6** Average ROC AUC contributions of each individual structure feature over all enumerated and evaluated feature sets in the cluster cross-validation. Contributions are computed as the mean of the marginals based on Shapley values. Features are depicted on the x-axis and datasets with the machine learning methods logistic regression (LR), random forest (RF), and gradient boosting (GB) on the y-axis

MesoModel experiment are similar (minimal Pearson's correlation coefficient 0.91 for sequence; 0.90 for structure). In addition, contributions in ThermoAll and ThermoModel are similar (minimal Pearson's correlation coefficient of 0.88 for sequence; 0.77 for structure). However, the contributions between ThermoAll, ThermoModel and DecoyAll, MesoModel are rather dissimilar (maximal Pearson's correlation coefficient of 0.30; 0.31 for structure). Given that the same or similar features are important in the two respective datasets we analyze their results separately.

### 3.3.4 | Deep-sea proteins versus proteins from all decoys and mesophilic model organisms

In the DecoyAll and MesoModel experiments the most contributing sequence features are the proportion of GLN, GLU, ILE, LYS, SER, positively and negatively charged residues, polar residues, as well as CYS for the MesoModel experiment. There are also slighter contributions from the proportion of ALA, HIS, and THR for the DecoyAll experiment and HIS and THR for the MesoModel experiment. Using only the single most contributing features for classification leads already to a mean ROC AUC of 0.91 (MesoModel with GLN) and 0.83 (DecoyAll with pos. charged residues) in the cross-validation (see best\_features.tsv). This illustrates that the distribution of these residue features alone are highly descriptive. Unsurprisingly, the distribution plots of these features show clear differences between deep-sea proteins and corresponding decoy proteins (see Figures S13 and S16). Specifically, on average deep-sea proteins have less GLN and more positively charged residues than their orthologs from other environments.

For structure features, the most contributing features are the proportion of polar surface, positively and negatively charged surface as well as the buried polar residues, buried positively and negatively charged residues and the number of noninteracting anions in the whole protein and on the surface. In addition, salt bridges seem to play a role at the surface and in the whole protein. From these polar and charged surface features are by far the most contributing. Using for example positively and negatively charged surface as feature set yields a mean ROC AUC of 0.85 and 0.90 on the DecoyAll and MesoModel dataset with logistic regression in the cross-validation. The difference in these features can also be well observed in the distribution plots in Figure S23. On average deep-sea proteins show an increased fraction of charged surface and an decreased fraction of polar surface.

While both sequence and structure features are effective predictors, sequence features are more predictive. It is known that the amino acid composition of an organism's proteom correlates with the organism and an organism's environment.<sup>88,89</sup> However, the trends in important sequence features correspond well to the important structure features considering their physicochemical properties. In both the distribution of charged and polar residues is highly important. The differences in these features even seem to be sufficient to almost perfectly distinguish the here used deep-sea and decoy proteins.

To interpret which biological mechanism these correlations describe it is reasonable to consider the organisms from which the used data was derived. We are comparing deep-sea proteins of mostly hyperthermophilic Archaea. Therefore, it is interesting to determine which features are already attributed to thermal stability in the literature. A recent comparison study by Hait et al.<sup>24</sup> aimed to identify

generalized molecular principles of thermal adaptation and extracted “nearly universal” signatures from a larger set of prokaryotes with known OGT. The signatures were identified over a diverse set of orthologous protein pairs from (hyper)thermophiles and mesophiles similar to our approach. Hait et al. reported that in 94% of the experiments hyperthermophiles preferred charged amino acids, a pattern that is also very prominent in our results. In addition, it is reported that the small amino acids GLY and ALA (88%) as well as amid amino acids (96%) are disfavored. While we only observe a moderate contribution from ALA in the DecoyAll (and none in the MesoModel experiment) and none from GLY we can observe a very strong contribution from the proportion of the amid amino acid GLN, but not ASN. On the structure level Hait et al. reported an increased hydrophobic core (73%), higher exposure of charged/polar surface area (79%) and abundant salt-bridges (83%) as well as a higher number of cation- $\pi$  interactions (74%). In our result the hydrophobic core and the number cation- $\pi$  interactions seems to have no noteworthy contribution. However, we also observe high contributions of charged and polar surface area as well as ionic interactions (salt bridges). Explicitly, the polar surface is reduced on average in our deep-sea proteins and charged surface and salt bridges are increased.

In conclusion, the comparison of the deep-sea proteins from mainly hyperthermophilic Archaea shows a very strong and simple to capture pattern of protein properties to differentiate them from mesophilic proteins and orthologous proteins in general (as measured on the baseline DecoyAll set). Intriguingly, it seems that the patterns observed are very similar to the protein properties typically attributed to protein adaptations for thermal stability, which are specifically a reduced number of polar and an increased number of charged residues.<sup>14,24,26,90</sup> This is not surprising since most available experimental protein structure data from deep-sea organisms comes from hyperthermophilic Archaea. As a consequence, it is complicated to assign these correlations unambiguously to extreme adaptations, like to temperature, pressure, or both.

### 3.3.5 | Deep-sea proteins versus proteins from thermophiles

In the experiments involving only decoy structures from thermophiles (ThermoAll, ThermoModel) the most prominent sequence features are the proportion of ILE and LYS which by far show the highest contributions (see Figure 5). The third most relevant feature is ARG and we can also observe smaller contributions from ASN and LEU. Using LYS or ILE individually as feature sets results in mean ROC AUC values between 0.71 and 0.74 in the cross-validation and a similar performance on the external cluster-fold (see Figure 4, best\_feature files). Deep-sea proteins contain more LYS and more ILE on average than their respective decoys (see Figures S13 and S14). In contrast, ARG is slightly decreased on average in deep-sea proteins (see Figure S12). Noteworthy, the feature set of both ILE and LYS is only marginally better than the two features alone, suggesting that both are correlated.

The most contributing structure features (contribution  $>0.05$ ) are the number of torsional constraints, buried sulfur residues and number of noninteracting anions at the surface and in the whole protein, as well as, the number of noninteracting acceptors in side chains, surface area of sulfur residues and hydrophobic surface, the positively and negatively charged surface area and buried hydrophobic residues are contributing. These most contributing features correspond well to the best performing single feature sets in the cross-validation experiment (see best\_features.tsv). Notably, on the external test cluster-fold especially the number of noninteracting anions at the surface and in the whole protein show prediction performance consistent with the cross-validation when used as single features (ROC AUC of approx. 0.63).

In the literature, there are only a handful of studies exploring differences between deep-sea proteins and thermophiles which focus mainly on sequence composition of proteins from deep-sea piezophiles. Nath et al.<sup>31</sup> determined relevant amino acids to differentiate the protein sequences of piezophilic-thermophilic and thermophilic-nonpiezophilic of *Pyrococcus yayanosii* and *Pyrococcus furiosus* as well as *Thermococcus barophilus* and *Thermococcus kodakarensis* KOD1. They ranked ARG, LYS, ASN, and ILE for the first pair and ILE, LYS and ARG for the second pair as the most important features. These results are in agreement with the results we obtained. In another sequence of studies, Di Giulio<sup>27,28</sup> also found that especially the frequency of LYS, ILE, and ARG are correlated with piezophilic organisms. The author described the hydrostatic pressure asymmetry index for the protein sequences of three pairs of piezophilic-thermophilic and thermophilic-nonpiezophilic organisms, namely, *Pyrococcus abyssi* with *P. furiosus*, *P. yayanosii* with *P. furiosus*, and *T. barophilus* with *T. kodakarensis*. Interestingly, depending on the organism pairs the correlation was either positive or negative.<sup>28</sup> The author reasoned that because both LYS and ARG have similar physicochemical properties at some point in evolution the organisms committed to one or the other. In our experiment we see an increased use of LYS and ILE but an reduced use in ARG meaning that the proteins from organisms we investigate show a positive correlation with LYS and ILE and a negative correlation with ARG.

Again, sequence features are more predictive than structure features. Interestingly, no clear correspondence between the relatively well correlating amino acids LYS and ILE and the properties of important structure features is apparent. A reason for this might be, on the one hand, structural adaptations induced by the sequence adaptations might be simply not well described by our chosen structure features or cannot be sufficiently captured from the accuracy or static state of the crystal structure. On the other hand, this discrepancy might be because the amino acid preference is not expressed in structural differences and is therefore potentially unrelated to protein extreme adaptations.

In conclusion, both ILE and LYS and also ARG to a lesser extent are reasonably important in sequence and were also found to be important by others. In contrast, no individual structural feature is contributing very distinctively, except perhaps noninteracting anions. The predictive power of structure features observed in Figure 4 is therefore rather due to combinations of multiple features, instead of a clear preference in one of the single structure feature. The results



suggest that this combination is related to noninteracting anions, sulfur containing residues and the flexibility of the protein.

### 3.3.6 | Important deep-sea protein features

When we compare the prediction performance and important features from all four experiments, we observe that deep-sea protein structures are harder to distinguish from structures of thermophiles (ThermoAll, ThermoModel) than from structures of mesophiles and all decoys (MesoModel, DecoyAll). This is not surprising, considering that most (hyper)thermophilic deep-sea organisms are likely evolutionary more similar to the thermophiles. While the important features in the MesoModel experiments are clear, it is not possible to assign single correlations to individual (or multiple) extreme conditions on these results alone. For this reason, we compared the deep-sea proteins to proteins of thermophiles to further isolate potential pressure adaptations in proteins from (hyper)thermophilic deep-sea organisms.

An interesting result is that the features most important in the ThermoAll and ThermoModel experiments seem to be relevant also in the MesoModel and DecoyAll experiments (see Figures 5 and 6). Over all four datasets deep-sea proteins contain more LYS and more ILE on average than their respective decoys (see Figures S13 and S14). The only structure features that are reasonably important over all four datasets are the distribution of noninteracting anions in the whole protein and at the surface. Both are increased on average in deep-sea proteins (see Figure S19). These results suggests that the distribution of these features is a rather unique trait of deep-sea proteins.

While our results provide clues about the features characterizing (hyper)thermophilic deep-sea proteins, a clear pattern or mechanism for high pressure adaptations is not apparent. However, effective prediction of deep-sea proteins is possible in all four experiments. These results demonstrate that predictive structural patterns between different deep-sea protein clusters exists. Our most predictive features and feature sets indicate which kind of protein property this hidden pattern might be related to.

## 4 | CONCLUSION

### 4.1 | Molecular adaptations to the deep-sea environment

The result that proteins of deep-sea (hyper)thermophiles are nearly perfectly separable from proteins from mesophiles likely illustrates the obvious differences between the proteins from thermophiles and mesophiles which have been explored heavily in the past.<sup>14,24,26,90</sup> However, these obvious differences alone are not sufficient to fully enable engineering of proteins toward high temperature<sup>26</sup> and probably other extreme conditions. Our results on the ThermoAll and ThermoModel datasets show that in addition to the general trends already analyzed in detail, there are other, more complicated patterns in protein sequence and structure correlating with the deep-sea source

environment. However, these correlations are not global for the whole population of deep-sea proteins. On the one hand, these non-global correlations are in accordance with current beliefs on pressure adaptations,<sup>4,15</sup> which are stating that pressure adaptations are only present in a subset of deep-sea proteins. Yet, some of the relevant features, most importantly LYS, ILE and charged atoms (especially anions), are shared across protein clusters and different decoy sets, which indicate the same adaptations in different protein classes. On the other hand, it seems that the features characteristic for proteins of deep-sea (hyper)thermophiles differ from those of deep-sea psychrophiles. However, the available structure data on deep-sea psychrophiles is scarce, which make these results not conclusive.

Consequently, the next interesting question to address would be in which deep-sea proteins and protein classes do we see molecular adaptations? One approach would be to investigate the proteins for which the determined important features are relevant. It would also be interesting to analyze individual protein classes that are more likely to hold adaptations, like enzymes involved in the energy metabolism.<sup>4</sup> In addition, further experiments with different subpopulations of deep-sea organisms are necessary, for example based on evolutionary relations of organisms or the similarity of their source environments, like the prevailing extreme conditions or the metabolism. Besides that, with our experiments we could provide a picture of the importance of a wide range of different features. However, to pin point single highly important features, further features need to be evaluated. An interesting example would be the proteins energetics and dynamics, which are not directly captured by our current descriptors or with the static protein structures. In conclusion, there are still multiple directions little explored yet and which are likely to provide valuable clues to disentangle the multiple protein adaptations to extremes.

### 4.2 | The current status of protein structures from deep-sea organisms

The currently available experimental protein structure data from deep-sea organisms in the PDB is scarce. In this work we could retrieve 1281 experimental protein structures (501 nonredundant) from 25 deep-sea organisms (see Table 2). While this constitutes a first data basis to analyze protein structures from deep-sea organisms and the absolute number of structures is probably sufficient for many analyses, the diversity of the retrieved organisms is limited. Most structures are from hyperthermophilic Archaea and 95% of the proteins are from organisms living under elevated temperature while only 5% are from psychrophilic and mesophilic deep-sea organisms. While the protein structure is more informative, the sequence data that is available in more variety and quantity would foster our understanding given the more tangible signals in sequence features.

In contrast, the available structure data for generating orthologous protein pairs with proteins of organisms from other environments from the PDB seems to be plenty. While we generated structure pairs having the same fold and at least remote homology is detectable in sequence, a more stringent sequence similarity likely provides an even less noisy

picture of protein differences. However, this would reduce the number of pairs and leave more deep-sea proteins unpaired. Probably the most important bottleneck in the pair generation process is the annotation of the source organisms environments, OGT and pressure or . This, however, remains a grand and largely multidisciplinary challenge.<sup>4</sup> Furthermore, while we currently using deep-sea proteins as a proxy for pressure stability (or other extreme adaptations), it would be extremely beneficial to compare protein pairs with experimentally determined low and high pressure stability.

In the future, the ever increasing efforts in environmental metagenomics<sup>4,12</sup> will provide more genome data from extreme environments. Carefully curated metadata annotation of these genomes with the conditions of their natural environment would provide an invaluable resource to comprehend the relationship between protein structure and environmental conditions. At the same time recent advancement made in protein structure prediction from sequence<sup>91,92</sup> provides an incomparable amount of structural protein information which is detached from what is experimentally solvable. Although, we still need to find out whether these methods model the subtleties in protein structures that we are looking for when we search for protein adaptations. Nevertheless, with more data available more comprehensive evaluation on single protein classes could be conducted. In addition, more expressive methodologies could be applied which allow to explore not only handcrafted features but also derive features from the data itself, which might be a fitting approach given the subtlety and context-dependence molecular adaptations are believed to have. An example of these are deep neural networks which we intentionally set aside in this study because of the limited data. Therefore, the future promises to further advance our understanding of the molecular limits of life and to exploit the full potential of enzymes from extremophiles.

Finally, we hope the compiled dataset and our feature evaluation will be useful to the community and a helpful starting point for other studies.

## ACKNOWLEDGMENT

This work was supported by the German Federal Ministry of Education and Research as part of protP.S.I. (031B0405B). Open access funding enabled and organized by Projekt DEAL.

## PEER REVIEW

The peer review history for this article is available at <https://publons.com/publon/10.1002/prot.26337>.

## DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available in Protein Data Bank at <https://www.rcsb.org/>. These data were derived from the following resources available in the public domain: - Protein Data Bank, <https://www.rcsb.org/>.

## REFERENCES

- Vieille C, Zeikus GJ. Hyperthermophilic enzymes: sources, uses, and molecular mechanisms for thermostability. *Microbiol Mol Biol Rev.* 2001;65:1-43.
- Feller G. Psychrophilic enzymes: from folding to function and biotechnology. *Scientifica.* 2013;2013:512840.
- Acinas SG, Sánchez P, Salazar G, et al. Deep ocean metagenomes provide insight into the metabolic architecture of bathypelagic microbial communities. *Commun Biol.* 2021;4:604.
- Ando N, Barquera B, Bartlett DH, et al. The molecular basis for life in extreme environments. *Annu Rev Biophys.* 2021;50:343-372.
- Aevarsson A, Kaczorowska A-K, Adalsteinsson BT, et al. Going to extremes—a metagenomic journey into the dark matter of life. *FEMS Microbiol Lett.* 2021;368:fnab067.
- Ravindra R, Winter R. On the Temperature-Pressure Free-Energy Landscape of Proteins. *ChemPhysChem.* 2003;4(4):359-365.
- Balny C. What lies in the future of high-pressure bioscience? *Biochim Biophys Acta.* 2006;1764:632-639.
- Winter R, Lopes D, Grudzielanek S, Vogtt K. Towards an understanding of the temperature/pressure configurational and free-energy landscape of biomolecules. *J Non-Equil Thermodyn.* 2007;32:41-97.
- Chakravorty D, Khan MF, Patra S. Multifactorial level of extremostability of proteins: can they be exploited for protein engineering? *Extremophiles.* 2017;21:419-444.
- Hikida Y, Kimoto M, Hirao I, Yokoyama S. Crystal structure of deep vent DNA polymerase. *Biochem Biophys Res Commun.* 2017;483:52-57.
- Harrison JP, Gheeraert N, Tsigelnitskiy D, Cockell CS. The limits for life under multiple extremes. *Trends Microbiol.* 2013;21:204-212.
- Konstantinidis KT, Braff J, Karl DM, DeLong EF. Comparative metagenomic analysis of a microbial community residing at a depth of 4,000 meters at station ALOHA in the North Pacific subtropical gyre. *Appl Environ Microbiol.* 2009;75:5345-5355.
- Bar-On YM, Phillips R, Milo R. The biomass distribution on earth. *Proc Natl Acad Sci U S A.* 2018;115:6506-6511.
- Reed CJ, Lewis H, Trejo E, Winston V, Evilia C. Protein adaptations in archaeal extremophiles. *Archaea.* 2013;2013:373275.
- Ichiye T. Enzymes from piezophiles. *Semin Cell Dev Biol.* 2018;84:138-146.
- Peoples LM, Kyaw TS, Ugalde JA, et al. Distinctive gene and protein characteristics of extremely piezophilic *Colwellia*. *BMC Genomics.* 2020;21:692.
- Salvador-Castell M, Oger P, Peters J. High-pressure adaptation of extremophiles and biotechnological applications. In: Salwan R, Sharma V, eds. *Physiological and Biotechnological Aspects of Extremophiles*. Academic Press; 2020:105-122.
- Kaye JZ, Baross JA. Synchronous effects of temperature, hydrostatic pressure, and salinity on growth, phospholipid profiles, and protein patterns of four *Halomonas* species isolated from deep-sea hydrothermal-vent and sea surface. *Environments.* 2004;70:6220-6229.
- Horikoshi K. Barophiles: deep-sea microorganisms adapted to an extreme environment. *Curr Opin Microbiol.* 1998;1:291-295.
- Abe F, Horikoshi K. The biotechnological potential of piezophiles. *Trends Biotechnol.* 2001;19:102-108.
- Jarab A, Kurzawa N, Hopf T, et al. Meltome atlas—thermal proteome stability across the tree of life. *Nat Methods.* 2020;17:495-503.
- Kumar S, Tsai C-J, Nussinov R. Factors enhancing protein thermostability. *Protein Eng Des Sel.* 2000;13:179-191.
- Razvi A, Scholtz JM. Lessons in stability from thermophilic proteins. *Protein Sci.* 2006;15:1569-1578.
- Hait S, Mallik S, Basu S, Kundu S. Finding the generalized molecular principles of protein thermal stability. *Proteins.* 2020;88:788-808.
- Eijsink VG, Bjørk A, Gåseidnes S, et al. Rational engineering of enzyme stability. *J Biotechnol.* 2004;113:105-120.
- Pucci F, Rooman M. Physical and molecular bases of protein thermal stability and cold adaptation. *Curr Opin Struct Biol.* 2017;42:117-128.

27. Di Giulio M. A comparison of proteins from *Pyrococcus furiosus* and *Pyrococcus abyssi*: barophily in the physicochemical properties of amino acids and in the genetic code. *Gene*. 2005;346:1-6.
28. Di Giulio M. The origin of the genetic code in the ocean abysses: new comparisons confirm old observations. *J Theor Biol*. 2013;333:109-116.
29. Yafremava LS, Di Giulio M, Caetano-Anollés G. Comparative analysis of barophily-related amino acid content in protein domains of *Pyrococcus abyssi* and *Pyrococcus furiosus*. *Archaea*. 2013;2013:680436.
30. Pradel N, Ji B, Gimenez G, et al. The first genomic and proteomic characterization of a deep-sea sulfate reducer: insights into the Piezophilic lifestyle of *Desulfovibrio piezophilus*. *PLoS One*. 2013;8:e55130.
31. Nath A, Subbiah K. Insights into the molecular basis of piezophilic adaptation: extraction of piezophilic signatures. *J Theor Biol*. 2016;390:117-126.
32. Avagyan S, Vasilchuk D, Makhatadze GI. Protein adaptation to high hydrostatic pressure: computational analysis of the structural proteome. *Proteins*. 2020;88:584-592.
33. Berman HM, Westbrook J, Feng Z, et al. The Protein Data Bank. *Nucleic Acids Res*. 2000;28:235-242.
34. Fang J, Zhang L, Bazylinski DA. Deep-sea piezosphere and piezophiles: geomicrobiology and biogeochemistry. *Trends Microbiol*. 2010;18(9):413-422.
35. Jebbar M, Franzetti B, Girard E, Oger P. Microbial diversity and adaptation to high hydrostatic pressure in deep-sea hydrothermal vents prokaryotes. *Extremophiles*. 2015;19:721-740.
36. Zhang Y, Li X, Xiao X, Bartlett DH. Current developments in marine microbiology: high-pressure biotechnology and the genetic engineering of piezophiles. *Curr Opin Biotechnol*. 2015;33:157-164.
37. Steinegger M, Meier M, Mirdita M, Vöhringer H, Haunsberger SJ, Söding J. HH-suite3 for fast remote homology detection and deep protein annotation. *BMC Bioinformatics*. 2019;20:473.
38. Söding J. Protein homology detection by HMM-HMM comparison. *Bioinformatics*. 2005;21:951-960.
39. Mirdita M, Von Den Driesch L, Galiez C, Martin MJ, Soding J, Steinegger M. Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic Acids Res*. 2017;45:D170-D176.
40. Rice P, Longden L, Bleasby A. EMBOSS: the European Molecular Biology Open Software Suite. *Trends Genet*. 2000;16:276-277.
41. Zhang Y, Skolnick J. TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res*. 2005;33:2302-2309.
42. Xu J, Zhang Y. How significant is a protein structure similarity with TM-score = 0.5? *Bioinformatics*. 2010;26:889-895.
43. Steinegger M, Söding J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat Biotechnol*. 2017;35:1026-1028.
44. Urbaczek S, Kolodzik A, Fischer JR, et al. NAOMI: on the almost trivial task of reading molecules from different file formats. *J Chem Inf Model*. 2011;51(12):3199-3207.
45. Bietz S, Urbaczek S, Schulz B, Rarey M. Protoss: a holistic approach to predict tautomers and protonation states in protein-ligand complexes. *J Cheminform*. 2014;6:12.
46. Flachsenberg F, Meyder A, Sommer K, Penner P, Rarey M. A consistent scheme for gradient-based optimization of protein-ligand poses. *J Chem Inf Model*. 2020;60:6502-6522.
47. Kabsch W, Sander C. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Bio-polymers*. 1983;22:2577-2637.
48. Schneider N, Lange G, Hindle S, Klein R, Rarey M. A consistent description of HYdrogen bond and DEhydration energies in protein-ligand complexes: methods behind the HYDE scoring function. *J Comput Aided Mol Des*. 2013;27:15-29.
49. Chen CR, Makhatadze GI. ProteinVolume: calculating molecular van der Waals and void volumes in proteins. *BMC Bioinformatics*. 2015;16:101.
50. Jacobs DJ, Rader AJ, Kuhn LA, Thorpe MF. Protein flexibility predictions using graph theory. *Proteins*. 2001;44:150-165.
51. Leigh JA, Albers SV, Atomi H, Allers T. Model organisms for genetics in the domain Archaea: methanogens, halophiles, Thermococcales and Sulfolobales. *FEMS Microbiol Rev*. 2011;35:577-608.
52. Guyon I, Elisseeff A. An introduction to variable and feature selection. *J Mach Learn Res*. 2003;3:1157-1182.
53. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: machine learning in python. *J Mach Learn Res*. 2011;12:2825-2830.
54. Fawcett T. An introduction to ROC analysis. *Pattern Recognit Lett*. 2006;27:861-874.
55. Shapley LS. A value for  $n$ -Person Games. In: Kuhn H, Tucker A, eds. *Classics in Game Theory*. 2nd ed. Princeton University Press; 2020:69-79.
56. Castro J, Gómez D, Tejada J. Polynomial calculation of the Shapley value based on sampling. *Comput Oper Res*. 2009;36:1726-1730.
57. González JM, Masuchi Y, Robb FT, et al. *Pyrococcus horikoshii* sp. nov., a hyperthermophilic archaeon isolated from a hydrothermal vent at the Okinawa Trough. *Extremophiles*. 1998;2:123-130.
58. Jones WJ, Leigh JA, Mayer F, Woese CR, Wolfe RS. *Methanococcus jannaschii* sp. nov., an extremely thermophilic methanogen from a submarine hydrothermal vent. *Arch Microbiol*. 1983;136:254-261.
59. Miller JF, Shah NN, Nelson CM, Ludlow JM, Clark DS. Pressure and temperature effects on growth and methane production of the extreme thermophile *Methanococcus jannaschii*. *Appl Environ Microbiol*. 1988;54:3039-3042.
60. Takami H, Inoue A, Fuji F, Horikoshi K. Microbial flora in the deepest sea mud of the Mariana Trench. *FEMS Microbiol Lett*. 1997;152:279-285.
61. Godfroy A, Lesongeur F, Raguénès G, et al. *Thermococcus hydrothermalis* sp. nov., a new hyperthermophilic archaeon isolated from a deep-sea hydrothermal vent. *Int J Syst Bacteriol*. 1997;47:622-626.
62. Kurr M, Huber R, König H, et al. *Methanopyrus kandleri*, gen. and sp. nov. represents a novel group of hyperthermophilic methanogens, growing at 110°C. *Arch Microbiol*. 1991;156:239-247.
63. Gao H, Obratzova A, Stewart N, et al. *Shewanella loihica* sp. nov., isolated from iron-rich microbial mats in the Pacific Ocean. *Int J Syst Evol Microbiol*. 2006;56:1911-1916.
64. Masanari M, Wakai S, Ishida M, Kato C, Sambongi Y. Correlation between the optimal growth pressures of four *Shewanella* species and the stabilities of their cytochromes  $c_5$ . *Extremophiles*. 2014;18:617-627.
65. Bernhardt G, Jaenicke R, Lüdemann H-D, König H, Stetter KO. High pressure enhances the growth rate of the thermophilic archaeobacterium *Methanococcus thermolithotrophicus* without extending its temperature range. *Appl Environ Microbiol*. 1988;54:1258-1261.
66. Pikuta EV, Marsic D, Itoh T, et al. *Thermococcus thio-reducens* sp. nov., a novel hyperthermophilic, obligately sulfur-reducing archaeon from a deep-sea hydrothermal vent. *Int J Syst Evol Microbiol*. 2007;57:1612-1618.
67. Lu J, Nogi Y, Takami H. *Oceanobacillus iheyensis* gen. nov., sp. nov., a deep-sea extremely halotolerant and alkaliphilic species isolated from a depth of 1050 m on the Iheya Ridge. *FEMS Microbiol Lett*. 2001;205:291-297.
68. Götz D, Banta A, Beveridge TJ, Rushdi AI, Simoneit BRT, Reysenbach AL. *Persephonella marina* gen. nov., sp. nov. and *Persephonella guaymasensis* sp. nov., two novel, thermophilic, hydrogen-oxidizing microaerophiles from deep-sea hydrothermal. *Int J Syst Evol Microbiol*. 2002;52:1349-1359.
69. DeLong EF, Franks DG, Yayanos AA. Evolutionary relationships of cultivated psychrophilic and barophilic deep-sea bacteria. *Appl Environ Microbiol*. 1997;63:2105-2108.

70. Nogi Y, Masui N, Kato C. *Photobacterium profundum* sp. nov., a new, moderately barophilic bacterial species isolated from a deep-sea sediment. *Extremophiles*. 1998;2:1-8.
71. Allen EE, Facciotti D, Bartlett DH. Monounsaturated but not polyunsaturated fatty acids are required for growth of the deep-sea bacterium *Photobacterium profundum* SS9 at high pressure and low temperature. *Appl Environ Microbiol*. 1999;65:1710-1720.
72. Donachie SP, Hou S, Gregory TS, Malahoff A, Alam M. *Idiomarina loihiensis* sp. nov., a halophilic  $\gamma$ -Proteobacterium from the Lō'ihi submarine volcano, Hawai'i. *Int J Syst Evol Microbiol*. 2003;53:1873-1879.
73. Tian XP, Tang SK, Dong JD, et al. *Marinactinospora thermotolerans* gen. nov., sp. nov., a marine actinomycete isolated from a sediment in the northern South China Sea. *Int J Syst Evol Microbiol*. 2009;59:948-952.
74. Kato C, Li L, Nogi Y, Nakamura Y, Tamaoka J, Horikoshi K. Extremely barophilic bacteria isolated from the Mariana Trench, Challenger Deep, at a depth of 11,000 meters. *Appl Environ Microbiol*. 1998;64:1510-1513.
75. Birrien JL, Zeng X, Jebbar M, et al. *Pyrococcus yayanosii* sp. nov., an obligate piezophilic hyperthermophilic archaeon isolated from a deep-sea hydrothermal vent. *Int J Syst Evol Microbiol*. 2011;61:2827-2831.
76. Xu Y, Nogi Y, Kato C, et al. *Moritella profunda* sp. nov. and *Moritella abyssi* sp. nov., two psychropiezophilic organisms isolated from deep Atlantic sediments. *Int J Syst Evol Microbiol*. 2003;53:533-538.
77. Nogi Y, Kato C, Horikoshi K. Taxonomic studies of deep-sea barophilic *Shewanella* strains and description of *Shewanella violacea* sp. nov. *Arch Microbiol*. 1998;170:331-338.
78. Vetricani C, Speck MD, Ellor SV, Lutz RA, Starovoytor V. *Thermovibrio ammonificans* sp. nov., a thermophilic, chemolithotrophic, nitrate-ammonifying bacterium from deep-sea hydrothermal vents. *Int J Syst Evol Microbiol*. 2004;54:175-181.
79. Huber R, Stöhr J, Hohenhaus S, et al. *Thermococcus chitonophagus* sp. nov., a novel, chitin-degrading, hyperthermophilic archaeum from a deep-sea hydrothermal vent environment. *Arch Microbiol*. 1995;164:255-264.
80. Miroshnichenko ML, Kostrikin NA, Chernyh NA, et al. *Caldithrix abyssi* gen. nov., sp. nov., a nitrate-reducing, thermophilic, anaerobic bacterium isolated from a Mid-Atlantic Ridge hydrothermal vent, represents a novel bacterial lineage. *Int J Syst Evol Microbiol*. 2003;53:323-329.
81. Antoine E, Cilia V, Meunier JR, Guezennec J, Lesongeur F, Barbier G. *Thermosipho melanesiensis* sp. nov., a new thermophilic anaerobic bacterium belonging to the order Thermotogales, isolated from deep-sea hydrothermal vents in the southwestern Pacific Ocean. *Int J Syst Bacteriol*. 1997;47:1118-1123.
82. Miura T, Abe F, Inoue A, Usami R, Horikoshi K. Purification and characterization of novel extracellular endopolygalacturonases from a deep-sea yeast, *Cryptococcus* sp. N6, isolated from the Japan Trench. *Biotechnol Lett*. 2001;23:1735-1739.
83. Abe F, Minegishi H, Miura T, Nagahama T, Usami R, Horikoshi K. Characterization of cold- and high-pressure-active polygalacturonases from a deep-sea yeast, *Cryptococcus liquefaciens* strain N6. *Biosci Biotechnol Biochem*. 2006;70:296-299.
84. Xiao X, Wang P, Zeng X, Bartlett DH, Wang F. *Shewanella psychrophila* sp. nov. and *Shewanella piezotolerans* sp. nov., isolated from West Pacific deep-sea sediment. *Int J Syst Evol Microbiol*. 2007;57:60-65.
85. Takai K, Sugai A, Itoh T, Horikoshi K. *Palaeococcus ferrophilus* gen. nov., sp. nov., a barophilic, hyperthermophilic archaeon from a deep-sea hydrothermal vent chimney. *Int J Syst Evol Microbiol*. 2000;50:489-500.
86. Jeanthon C, L'Haridon S, Reysenbach AL, et al. *Methanococcus vulcanius* sp. nov., a novel hyperthermophilic methanogen isolated from East Pacific Rise, and identification of *Methanococcus* sp. DSM 4213<sup>T</sup> as *Methanococcus fervens* sp. nov. *Int J Syst Bacteriol*. 1999;49:583-589.
87. Rose AS, Hildebrand PW. NGL viewer: a web application for molecular visualization. *Nucleic Acids Res*. 2015;43:W576-W579.
88. Moura A, Savageau MA, Alves R. Relative amino acid composition signatures of organisms and environments. *PLoS One*. 2013;8:e77319.
89. Hormoz S. Amino acid composition of proteins reduces deleterious impact of mutations. *Sci Rep*. 2013;3:2919.
90. Suhre K, Claverie JM. Genomic correlates of hyperthermostability, an update. *J Biol Chem*. 2003;278:17198-17202.
91. Baek M, DiMaio F, Anishchenko I, et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*. 2021;373:871-876.
92. Jumper J, Evans R, Pritzel A, et al. Highly accurate protein structure prediction with AlphaFold. *Nature*. 2021;596:583-589.

## SUPPORTING INFORMATION

Additional supporting information may be found in the online version of the article at the publisher's website.

**How to cite this article:** Sieg J, Sandmeier CC, Lieske J, et al. Analyzing structural features of proteins from deep-sea organisms. *Proteins*. 2022;90(8):1521-1537. doi:[10.1002/prot.26337](https://doi.org/10.1002/prot.26337)

# Supporting Information:

## Analyzing Structural Features of Proteins from Deep-Sea Organisms

Jochen Sieg,<sup>†</sup> Chris Claudius Sandmeier,<sup>†</sup> Julia Lieske,<sup>‡</sup> Alke Meents,<sup>‡</sup> Christian Lemmen,<sup>¶</sup> Wolfgang R. Streit,<sup>§</sup> and Matthias Rarey<sup>\*,†</sup>

<sup>†</sup>*Universität Hamburg, ZBH - Center for Bioinformatics, Bundesstraße 43, 20146  
Hamburg, Germany*

<sup>‡</sup>*Center for Free-Electron Laser Science, Deutsches Elektronen-Synchrotron DESY,  
Notkestraße 85, 22607 Hamburg, Germany*

<sup>¶</sup>*BioSolveIT GmbH, An der Ziegelei 79, 53757 Sankt Augustin, Germany*

<sup>§</sup>*Universität Hamburg, Department of Microbiology and Biotechnology, Ohnhorststraße 18,  
22609 Hamburg, Germany*

E-mail: [matthias.rarey@uni-hamburg.de](mailto:matthias.rarey@uni-hamburg.de)

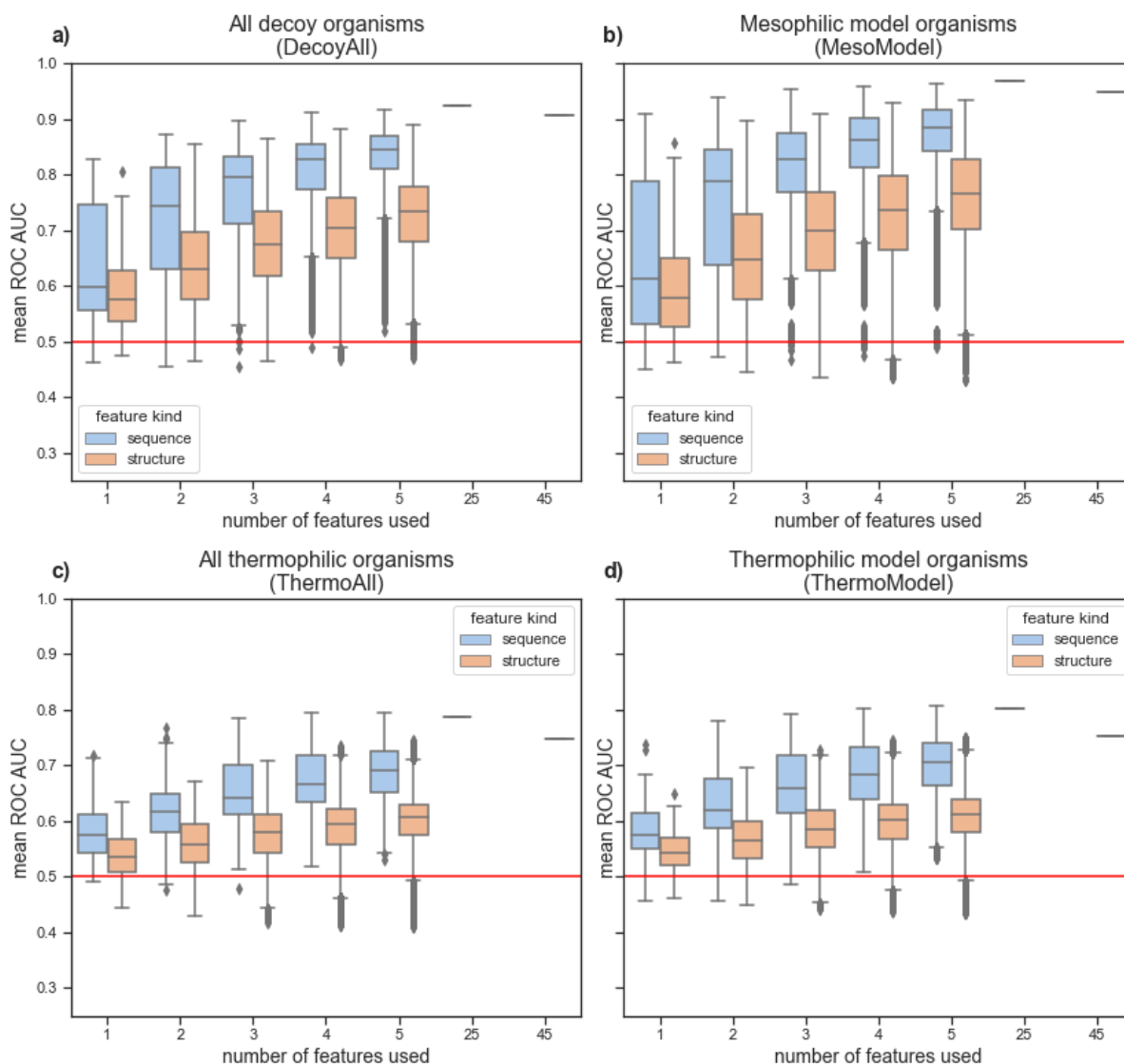


Figure S1: Logistic regression feature selection results of the cluster cross validation for all 5 decoy set experiments. Distribution of mean ROC AUC values of all built models in the cross validation is shown for the number of features used by the models. Performance achieved with protein sequence and structure features is depicted separately. The two rightmost entries on the *x*-axis show the performance with all sequence and structure features, respectively.



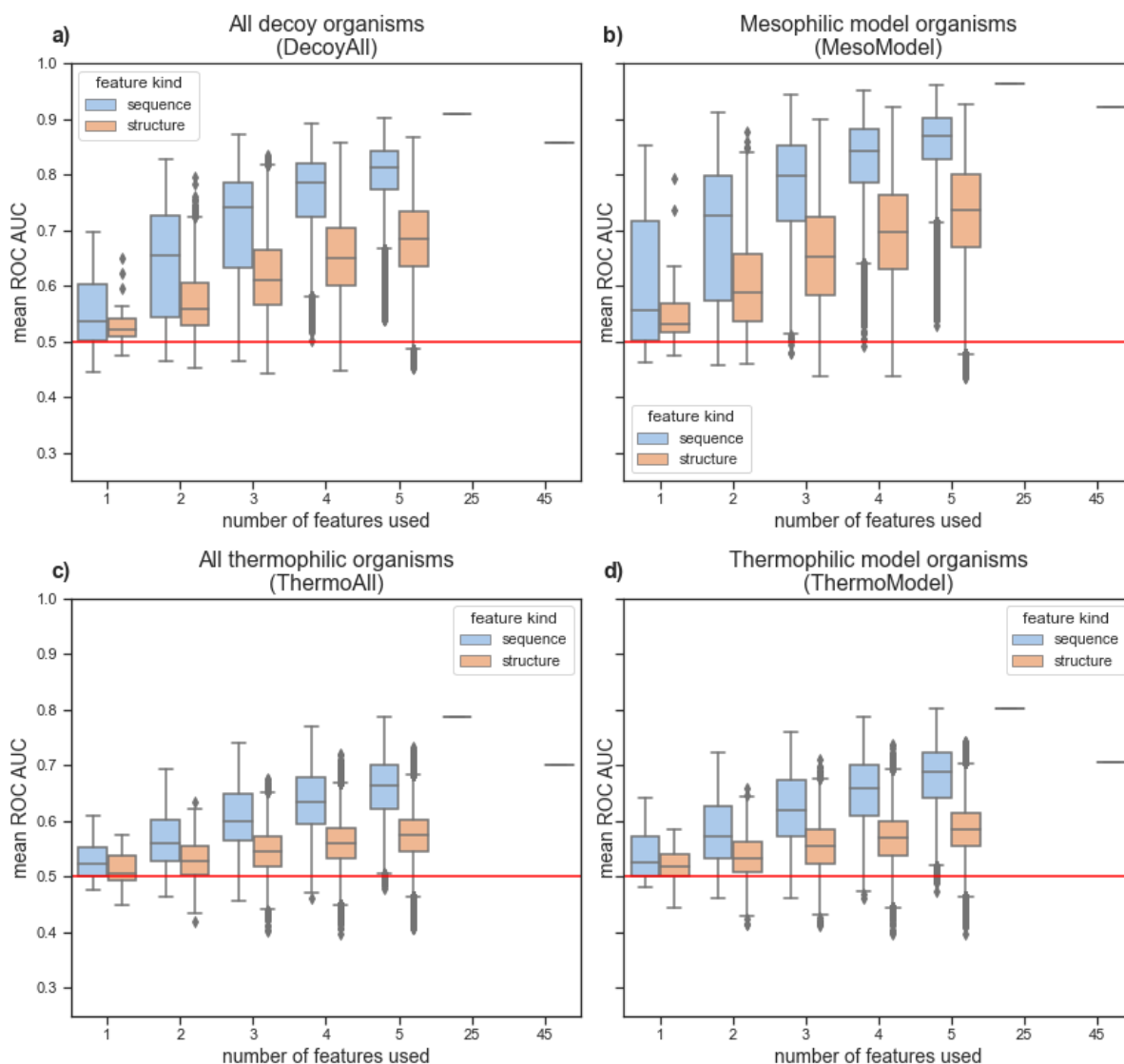


Figure S2: Random forest feature selection results of the cluster cross validation for all 5 decoy set experiments. Distribution of mean ROC AUC values of all built models in the cross validation is shown for the number of features used by the models. Performance achieved with protein sequence and structure features is depicted separately. The two rightmost entries on the  $x$ -axis show the performance with all sequence and structure features, respectively.

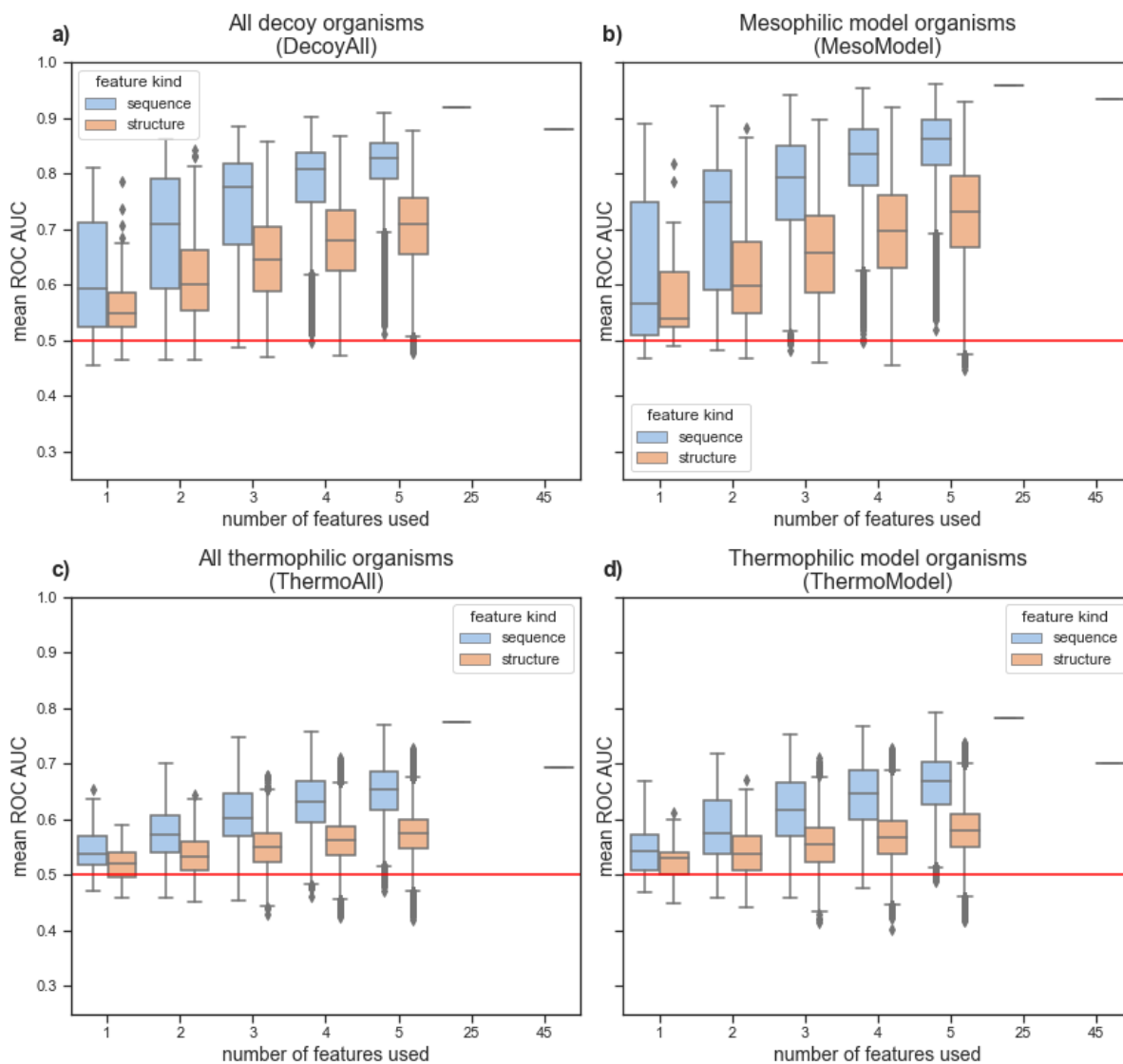


Figure S3: Gradient boosting classifier feature selection results of the cluster cross validation for all 5 decoy set experiments. Distribution of mean ROC AUC values of all built models in the cross validation is shown for the number of features used by the models. Performance achieved with protein sequence and structure features is depicted separately. The two right-most entries on the  $x$ -axis show the performance with all sequence and structure features, respectively.



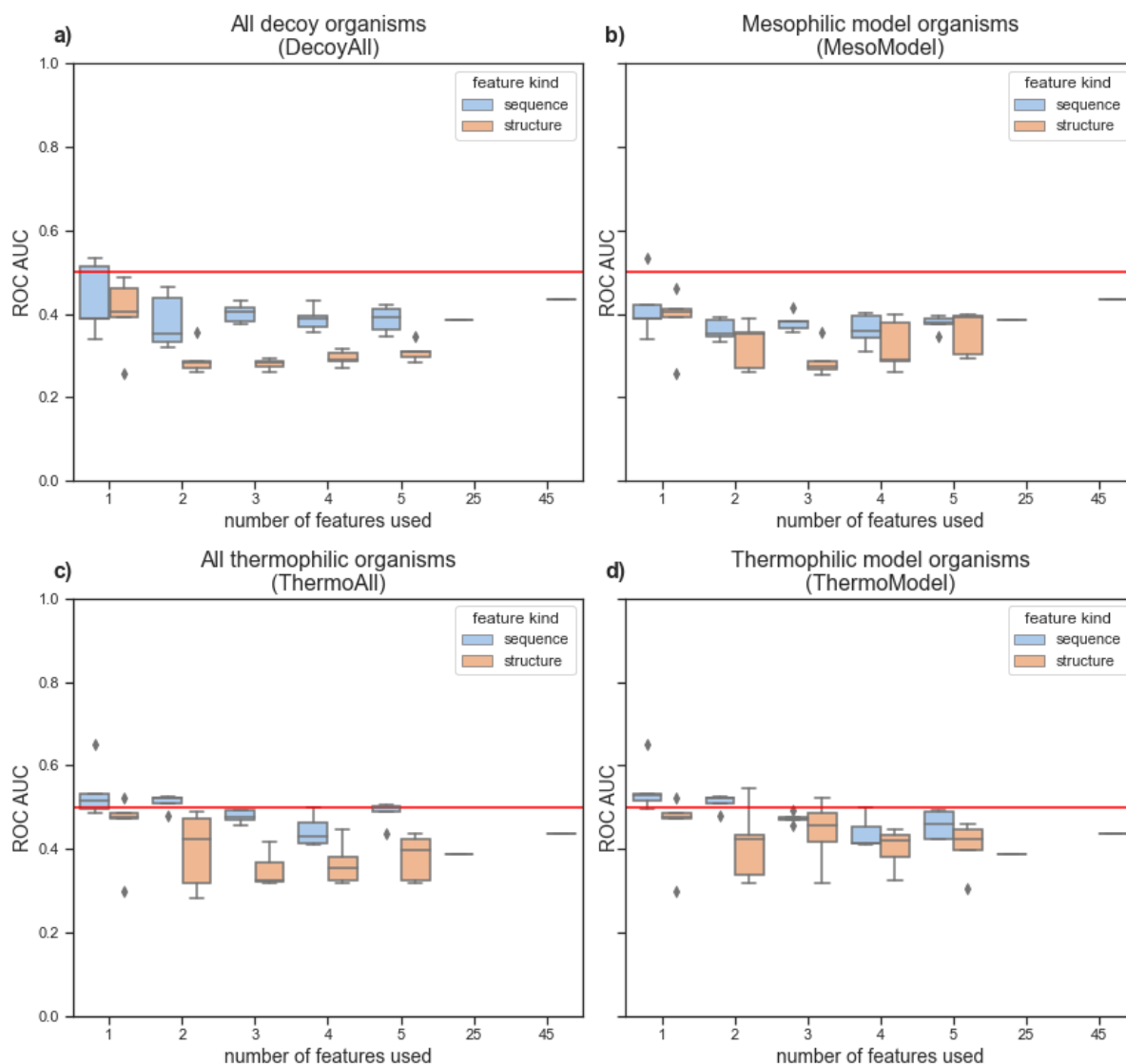


Figure S4: Logistic regression results on external test set number 1 of a hold-out set of protein pairs from psychrophilic and mesophilic deep-sea organisms. Each box shows the ROC AUC distribution of the five best feature sets from feature selection for the respective machine learning algorithm and feature set size. The two rightmost entries on the  $x$ -axis show the performance with all sequence and structure features, respectively.

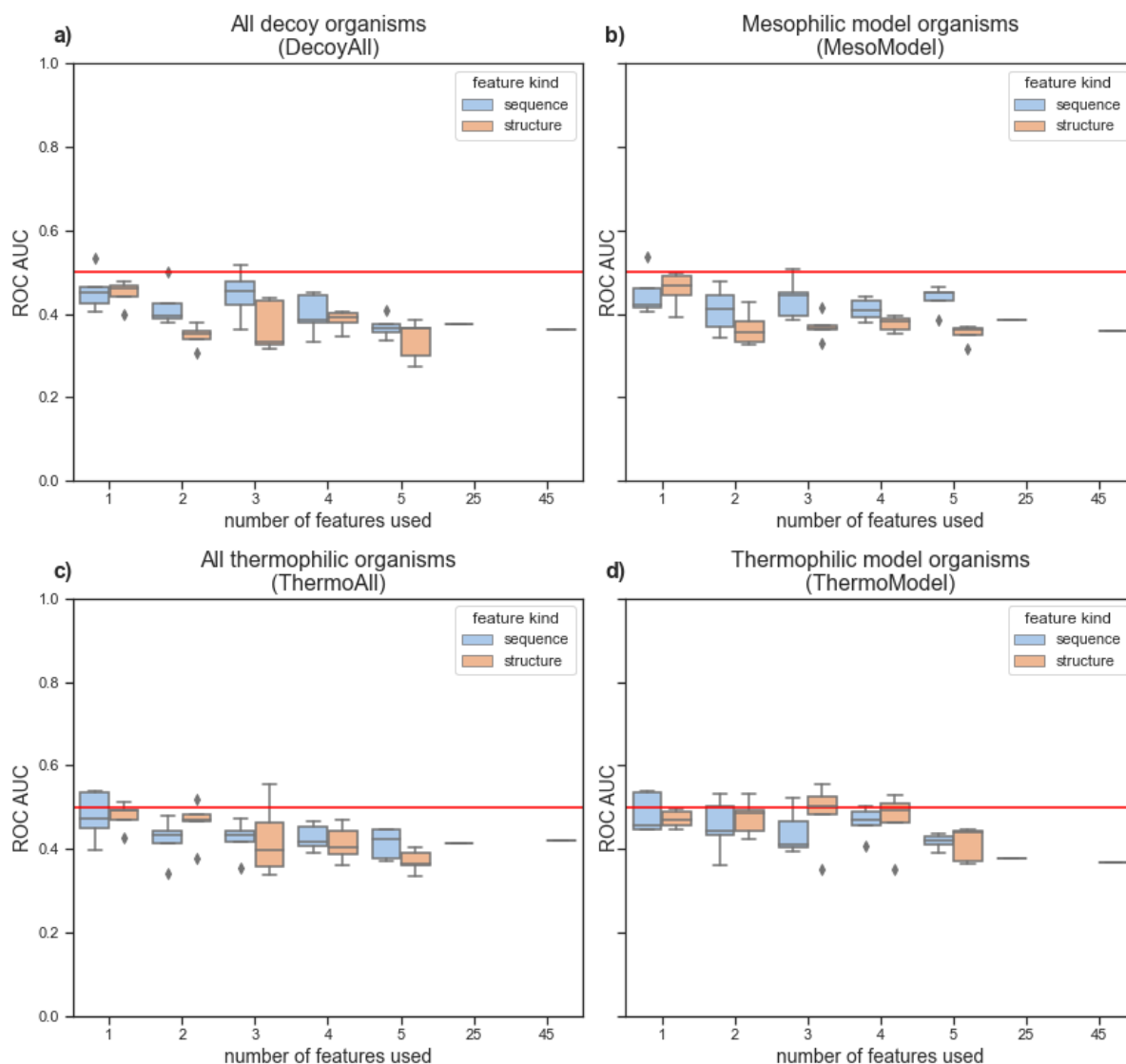


Figure S5: Random forest results on external test set number 1 of a hold-out set of protein pairs from psychrophilic and mesophilic deep-sea organisms. Each box shows the ROC AUC distribution of the five best feature sets from feature selection for the respective machine learning algorithm and feature set size. The two rightmost entries on the  $x$ -axis show the performance with all sequence and structure features, respectively.

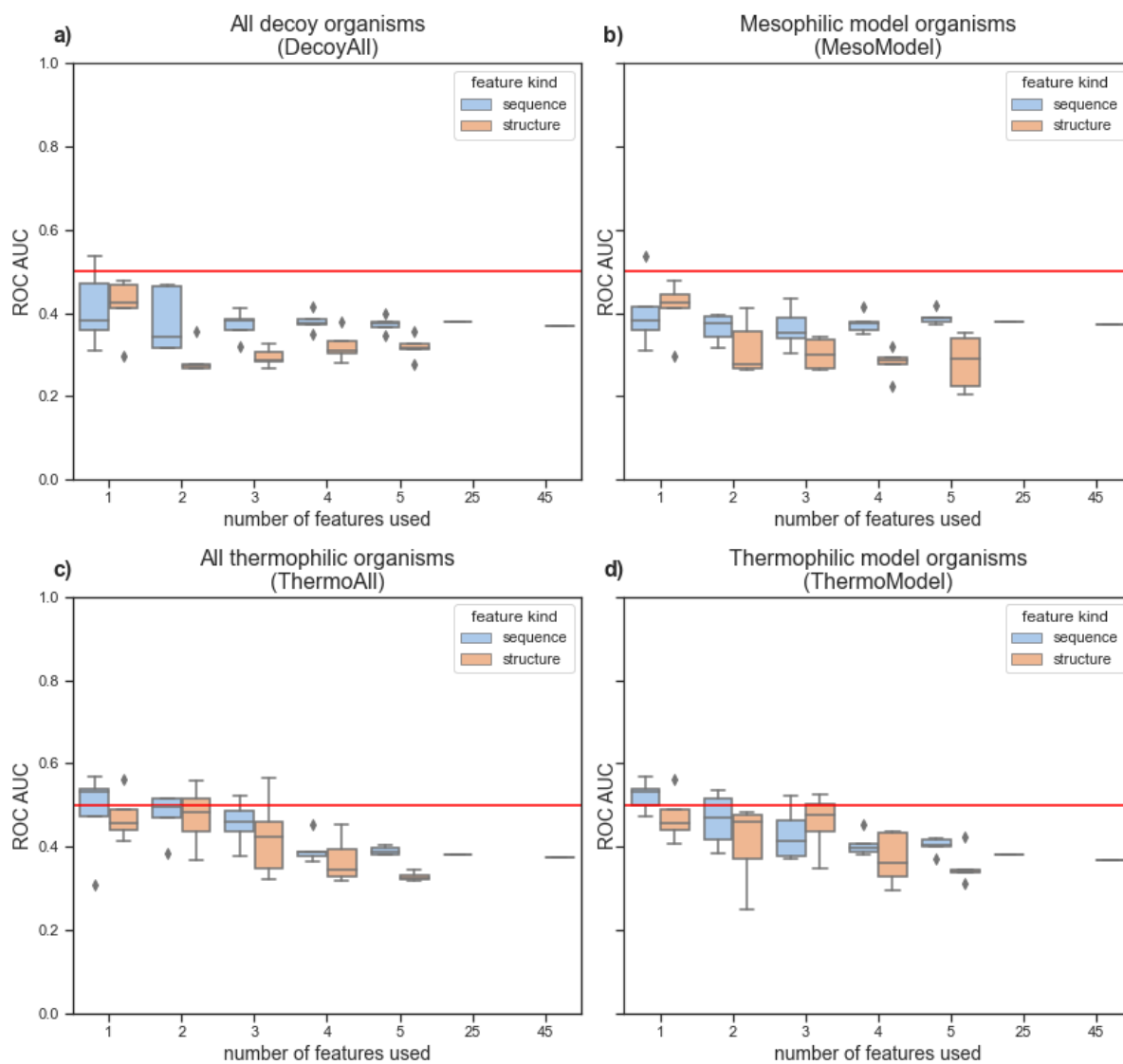


Figure S6: Gradient boosting results on external test set number 1 of a hold-out set of protein pairs from psychrophilic and mesophilic deep-sea organisms. Each box shows the ROC AUC distribution of the five best feature sets from feature selection for the respective machine learning algorithm and feature set size. The two rightmost entries on the  $x$ -axis show the performance with all sequence and structure features, respectively.

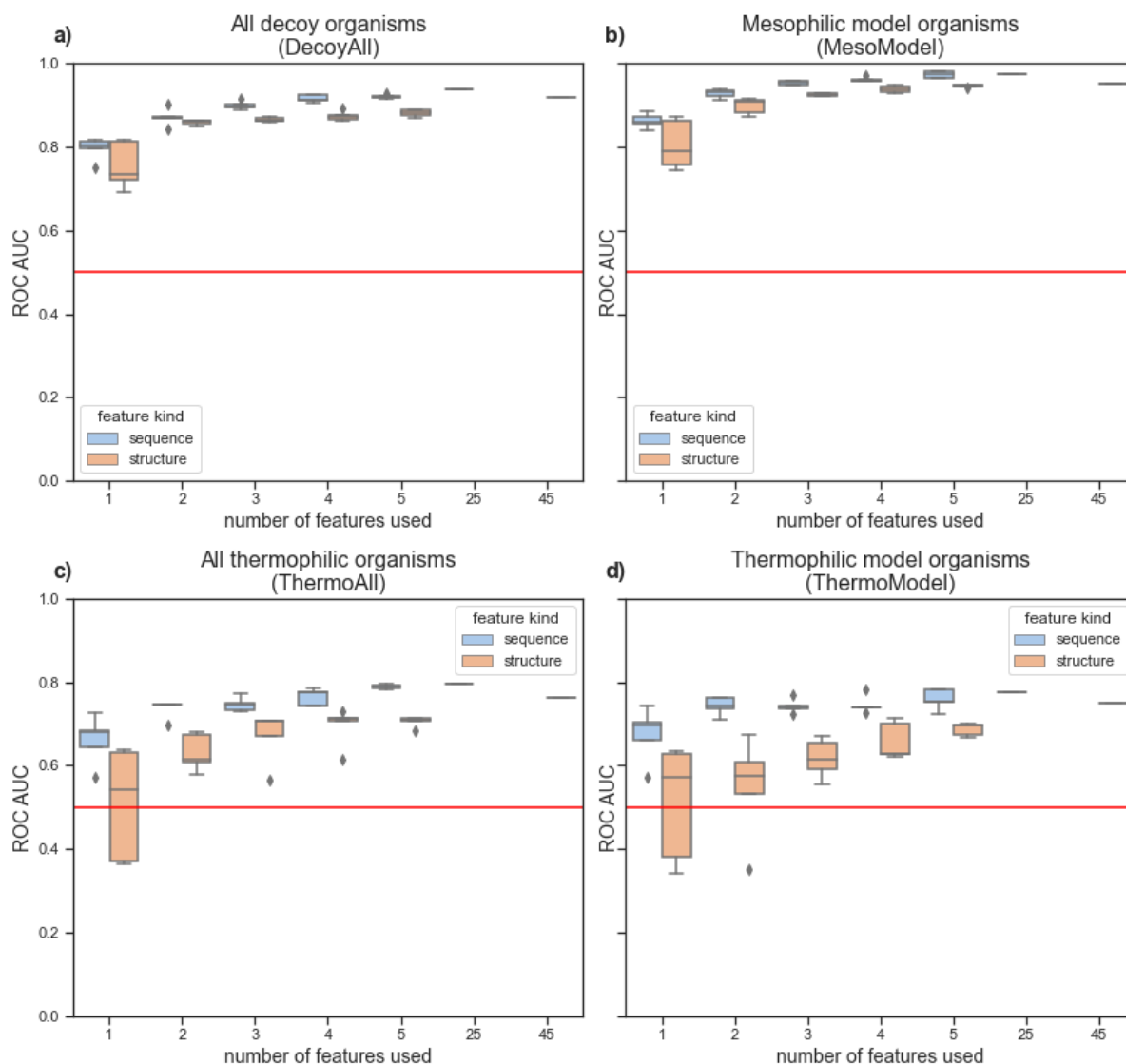


Figure S7: Logistic regression results on external test set number 2 of a hold-out cluster-cross-validation fold. Each box shows the ROC AUC distribution of the five best feature sets from feature selection for the respective machine learning algorithm and feature set size. The two rightmost entries on the  $x$ -axis show the performance with all sequence and structure features, respectively.

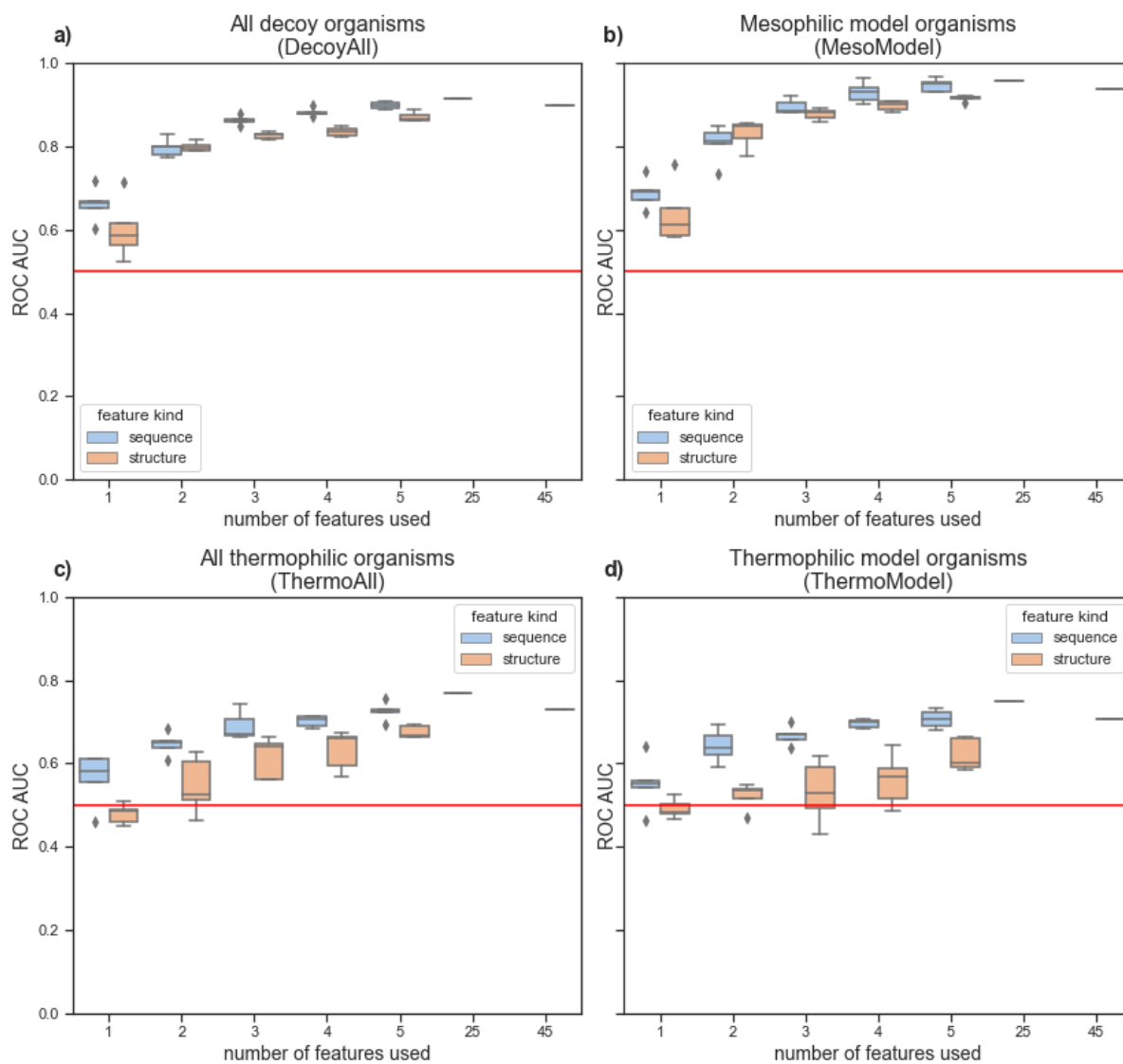


Figure S8: Random forest results on external test set number 2 of a hold-out cluster-cross-validation fold. Each box shows the ROC AUC distribution of the five best feature sets from feature selection for the respective machine learning algorithm and feature set size. The two rightmost entries on the  $x$ -axis show the performance with all sequence and structure features, respectively.

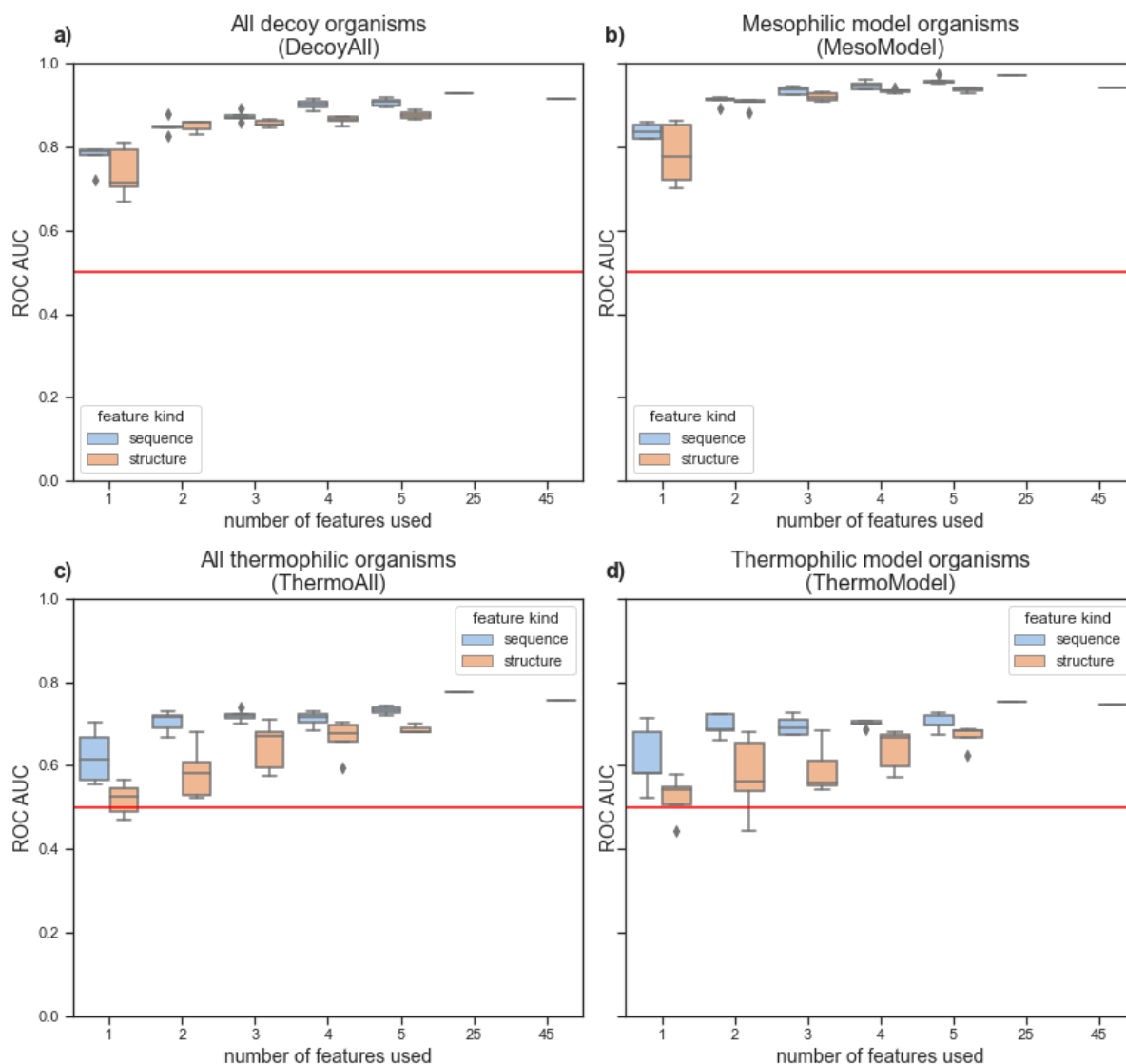


Figure S9: Gradient boosting results on external test set number 2 of a hold-out cluster-cross-validation fold. Each box shows the ROC AUC distribution of the five best feature sets from feature selection for the respective machine learning algorithm and feature set size. The two rightmost entries on the  $x$ -axis show the performance with all sequence and structure features, respectively.

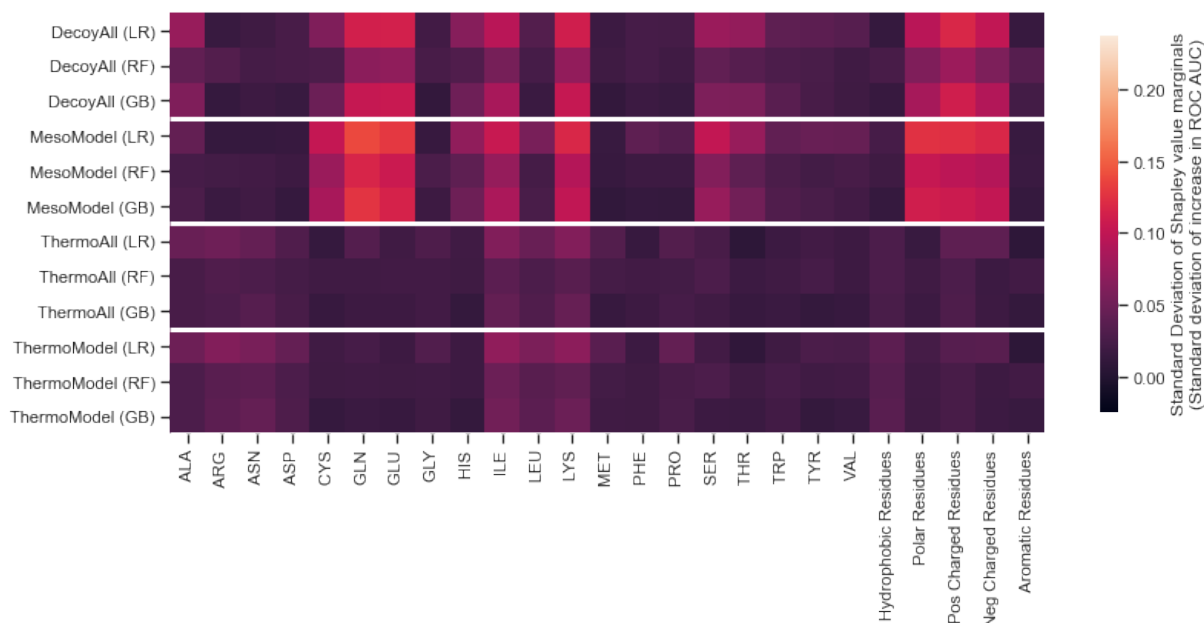


Figure S10: Standard deviation of Shapley values of the single sequence features. This corresponds to the standard deviation of ROC AUC contributions of each individual sequence feature over all enumerated and evaluated feature sets in the cluster cross validation. In other words, the standard deviation of the Shapley values is computed as the standard deviation of the marginals over all coalitions of features. Features are depicted on the  $x$ -axis and data sets with the machine learning methods logistic regression (LR), random forest (RF) and gradient boosting (GB) on the  $y$ -axis. The same color range as in the Shapley values plots is used. The standard deviation of the marginals expresses the relation of individual features to other features and the target variable. For example the standard deviation can be high when features are correlated or a single feature is only predictive in combination with certain other features.

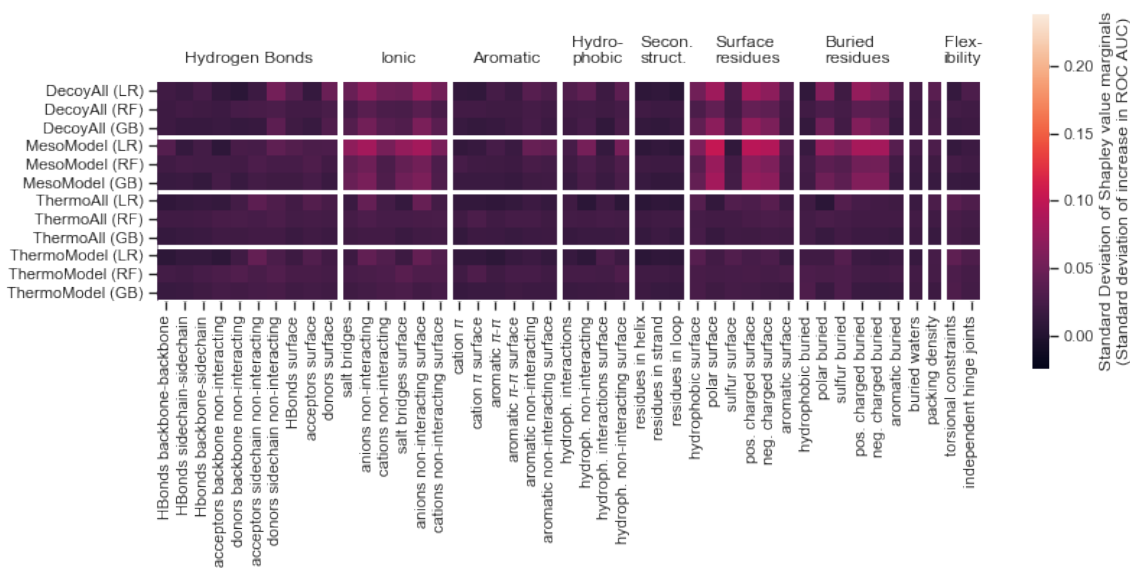


Figure S11: Standard deviation of Shapley values of the single structure features. This corresponds to the standard deviation of ROC AUC contributions of each individual structure feature over all enumerated and evaluated feature sets in the cluster cross validation. In other words, the standard deviation of the Shapley values is computed as the standard deviation of the marginals over all coalitions of features. Features are depicted on the  $x$ -axis and data sets with the machine learning methods logistic regression (LR), random forest (RF) and gradient boosting (GB) on the  $y$ -axis. The same color range as in the Shapley values plots is used. The standard deviation of the marginals expresses the relation of individual features to other features and the target variable. For example the standard deviation can be high when features are correlated or a single feature is only predictive in combination with certain other features.



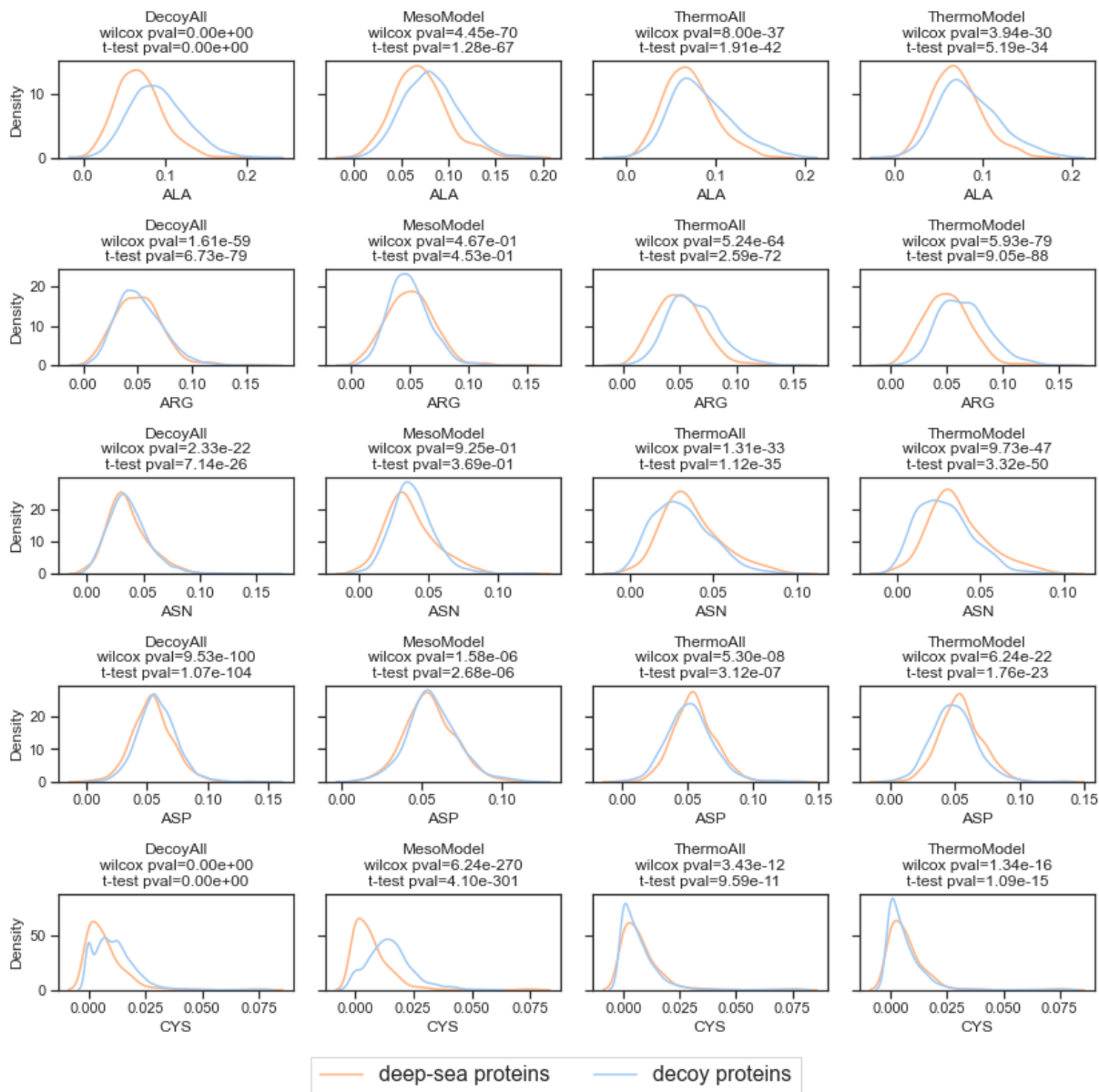


Figure S12: First distribution plots of single sequence features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

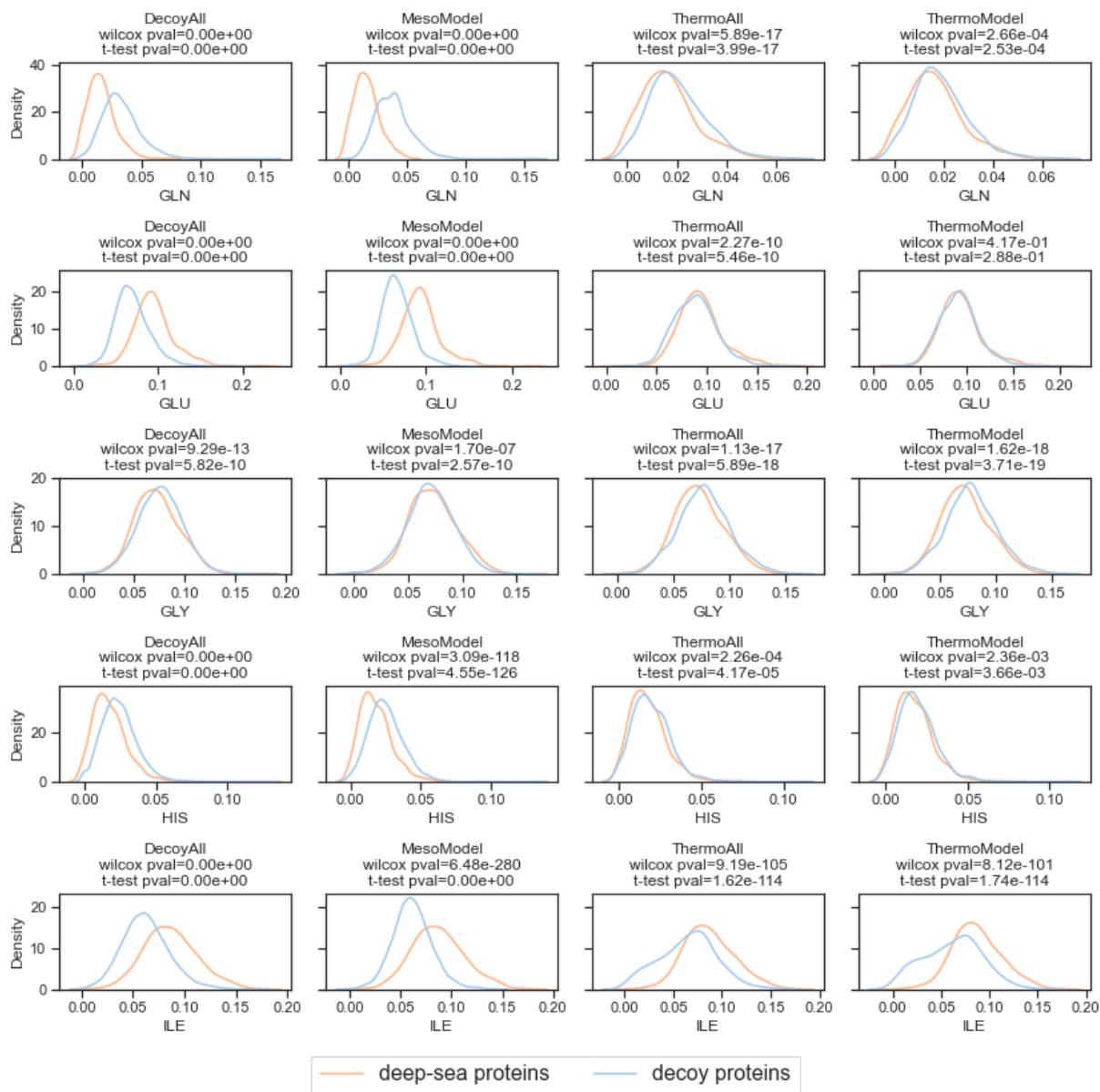


Figure S13: Second distribution plots of single sequence features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

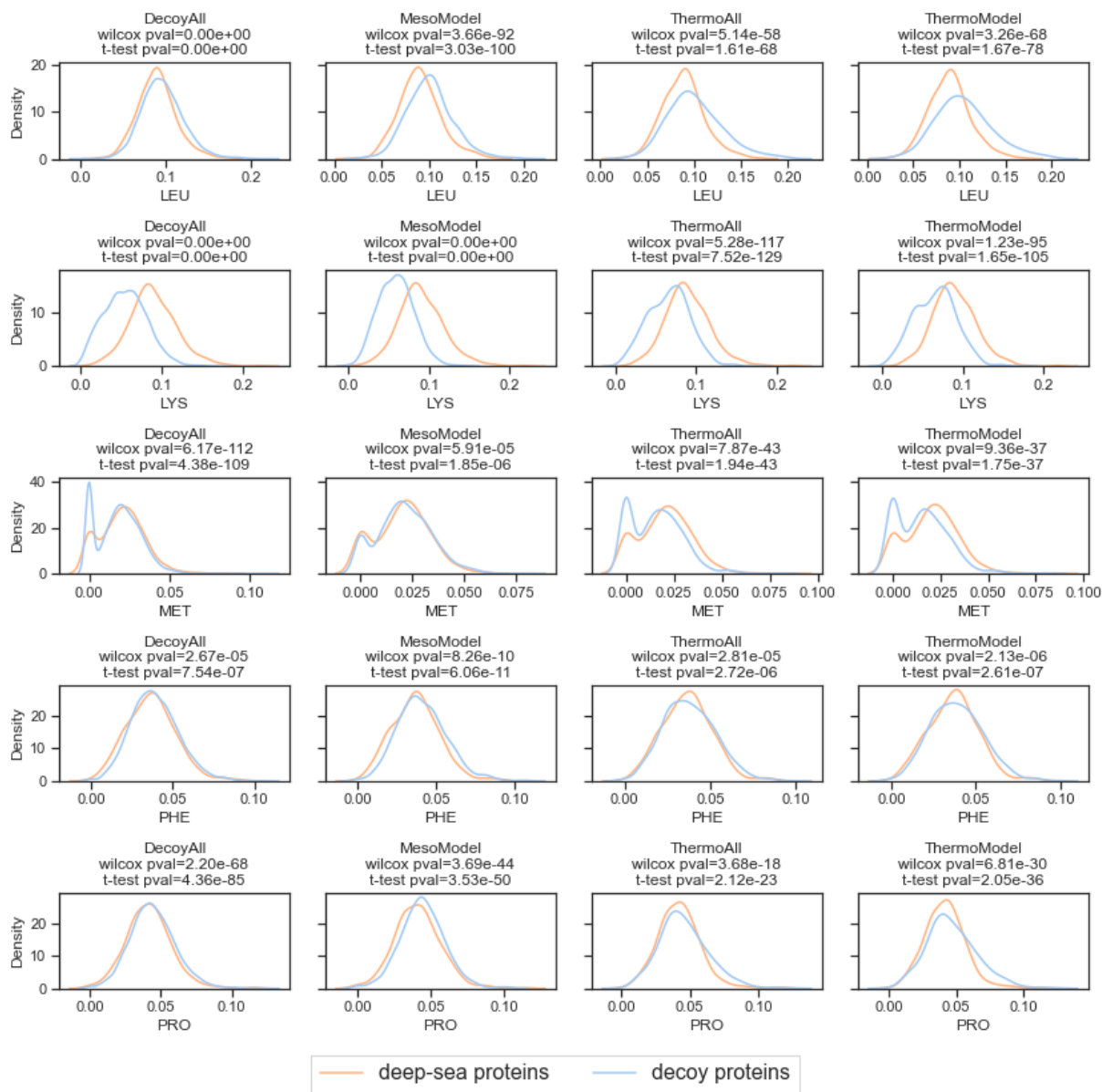


Figure S14: Third distribution plots of single sequence features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

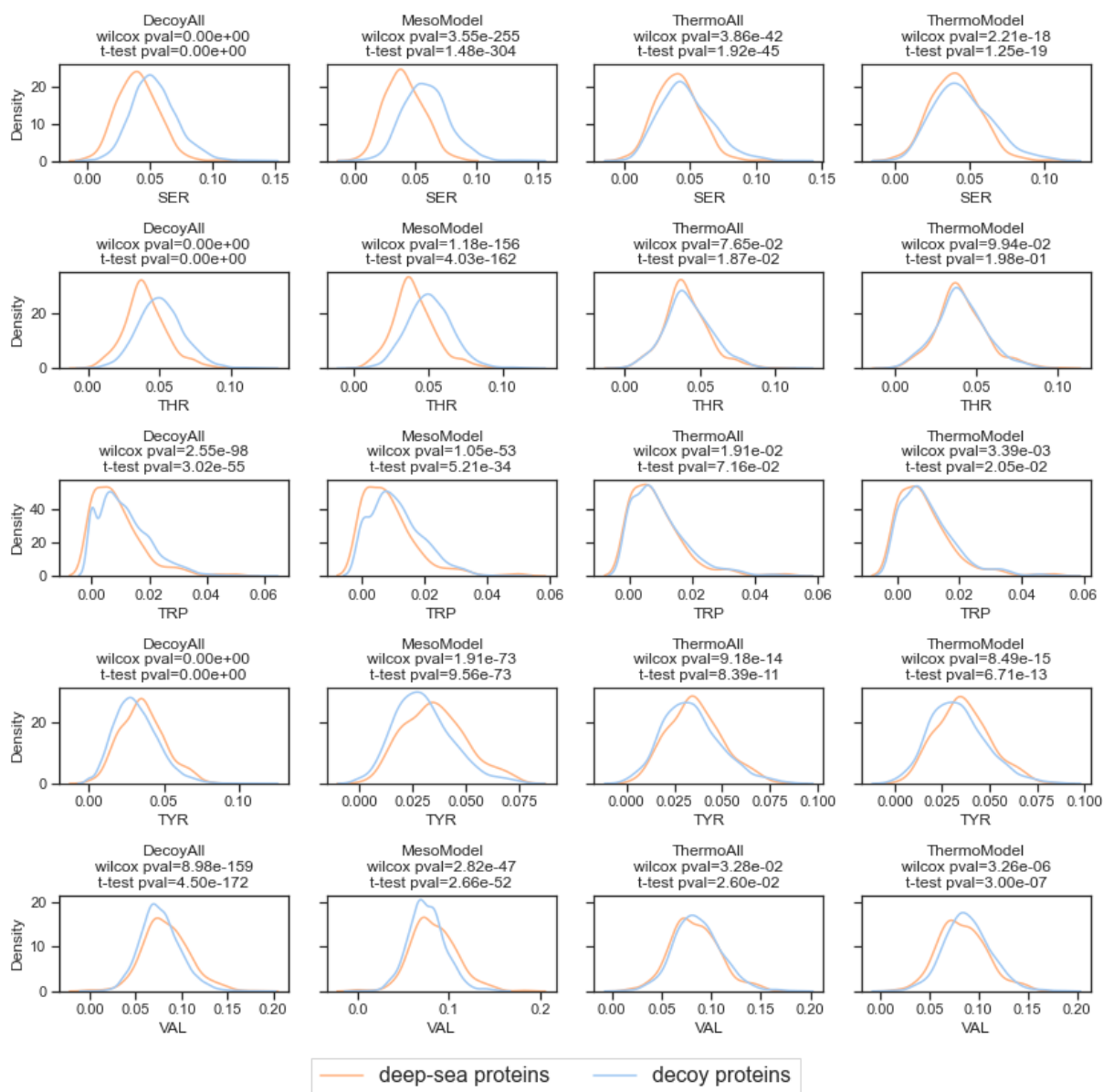


Figure S15: Fourth distribution plots of single sequence features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

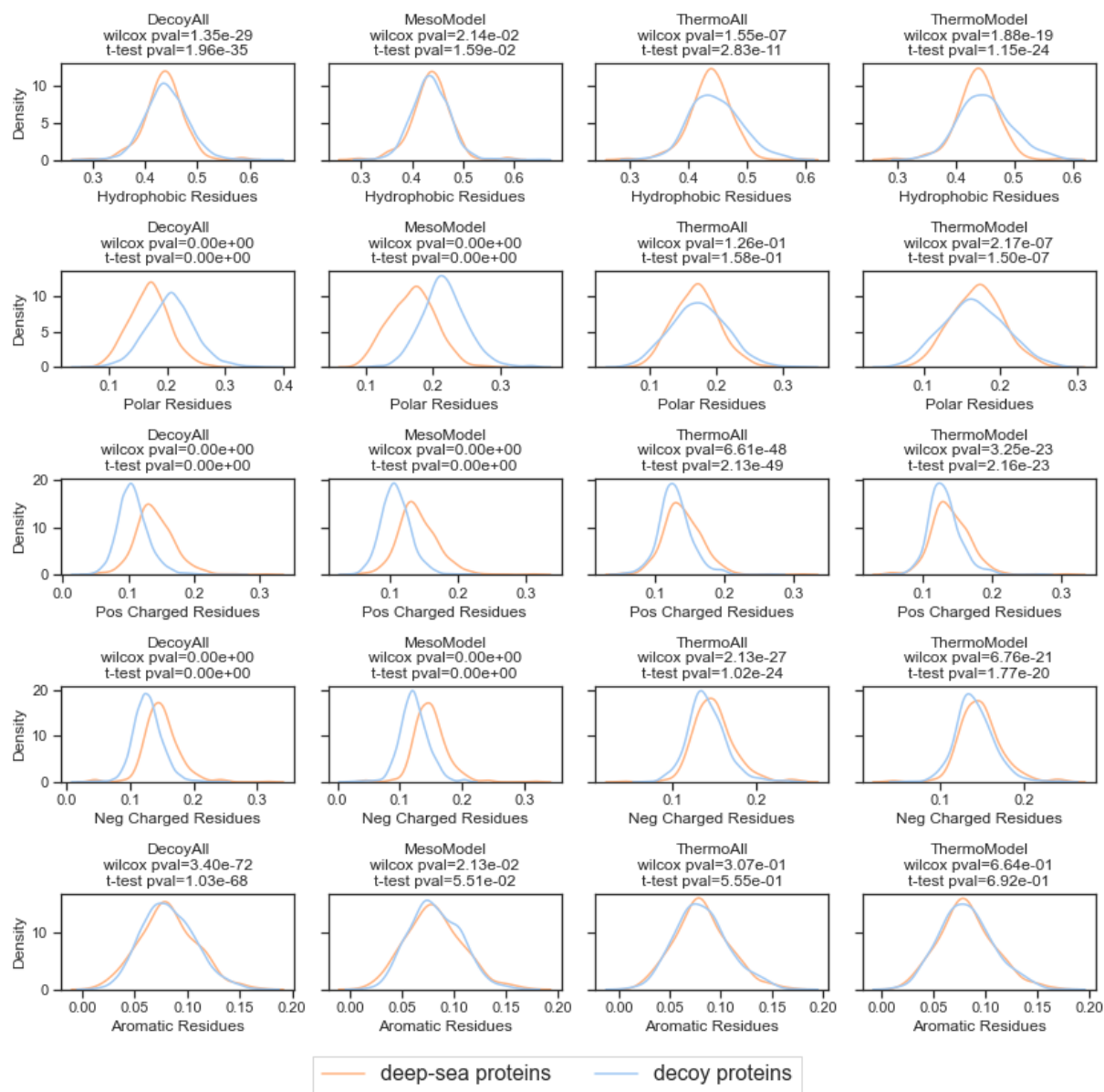


Figure S16: Fifth distribution plots of single sequence features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

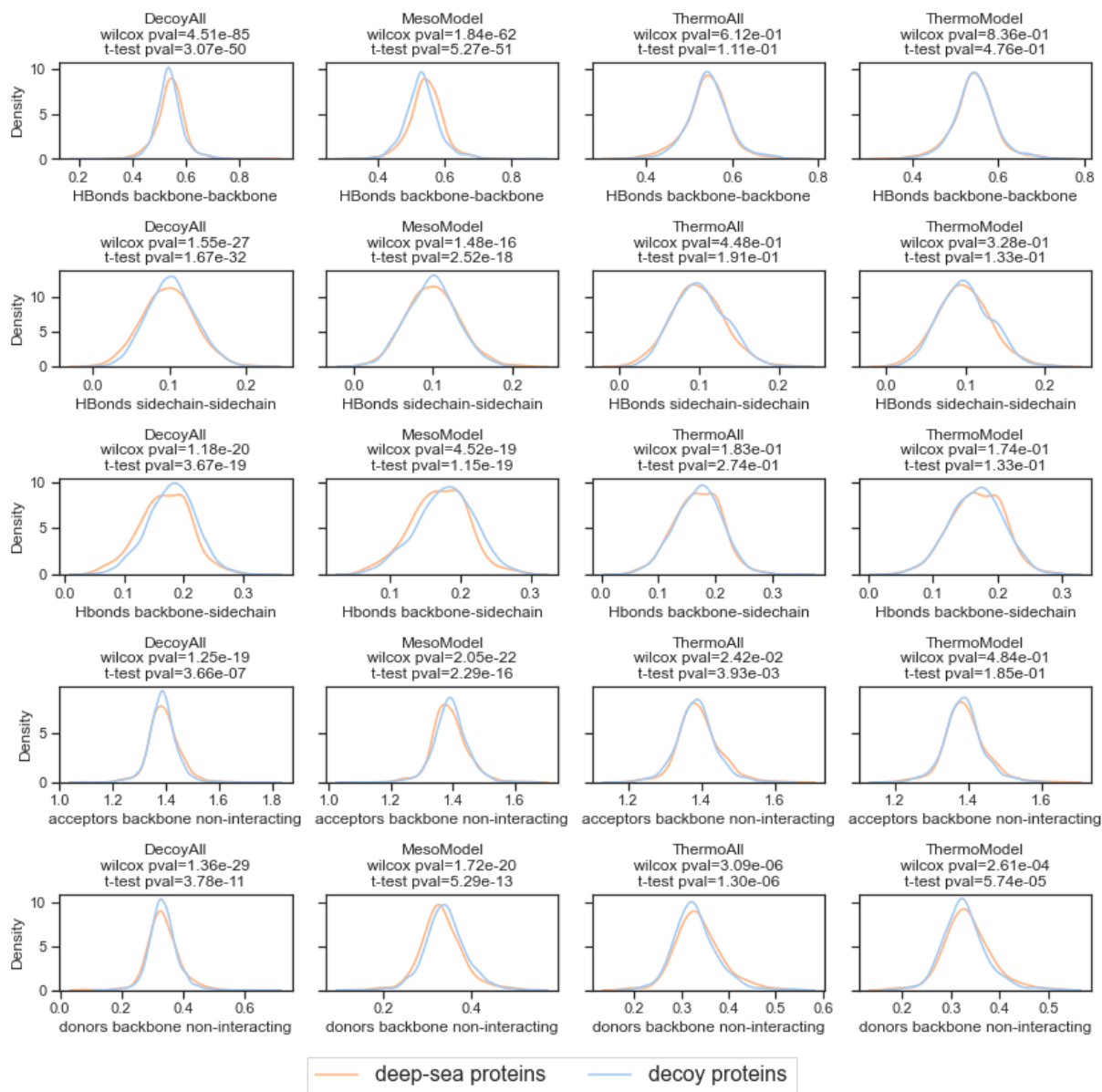


Figure S17: First distribution plots of single structure features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

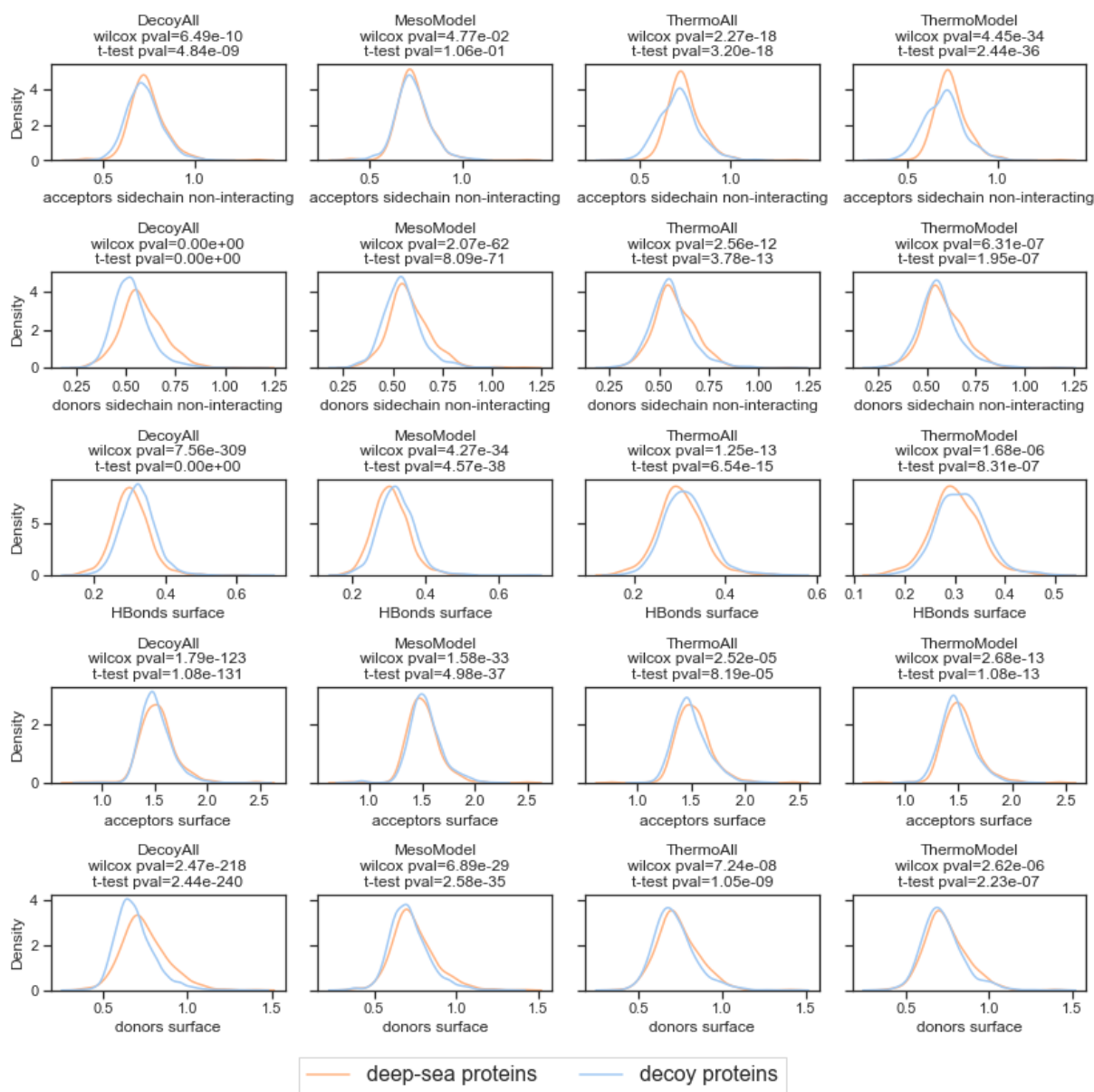


Figure S18: Second distribution plots of single structure features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.



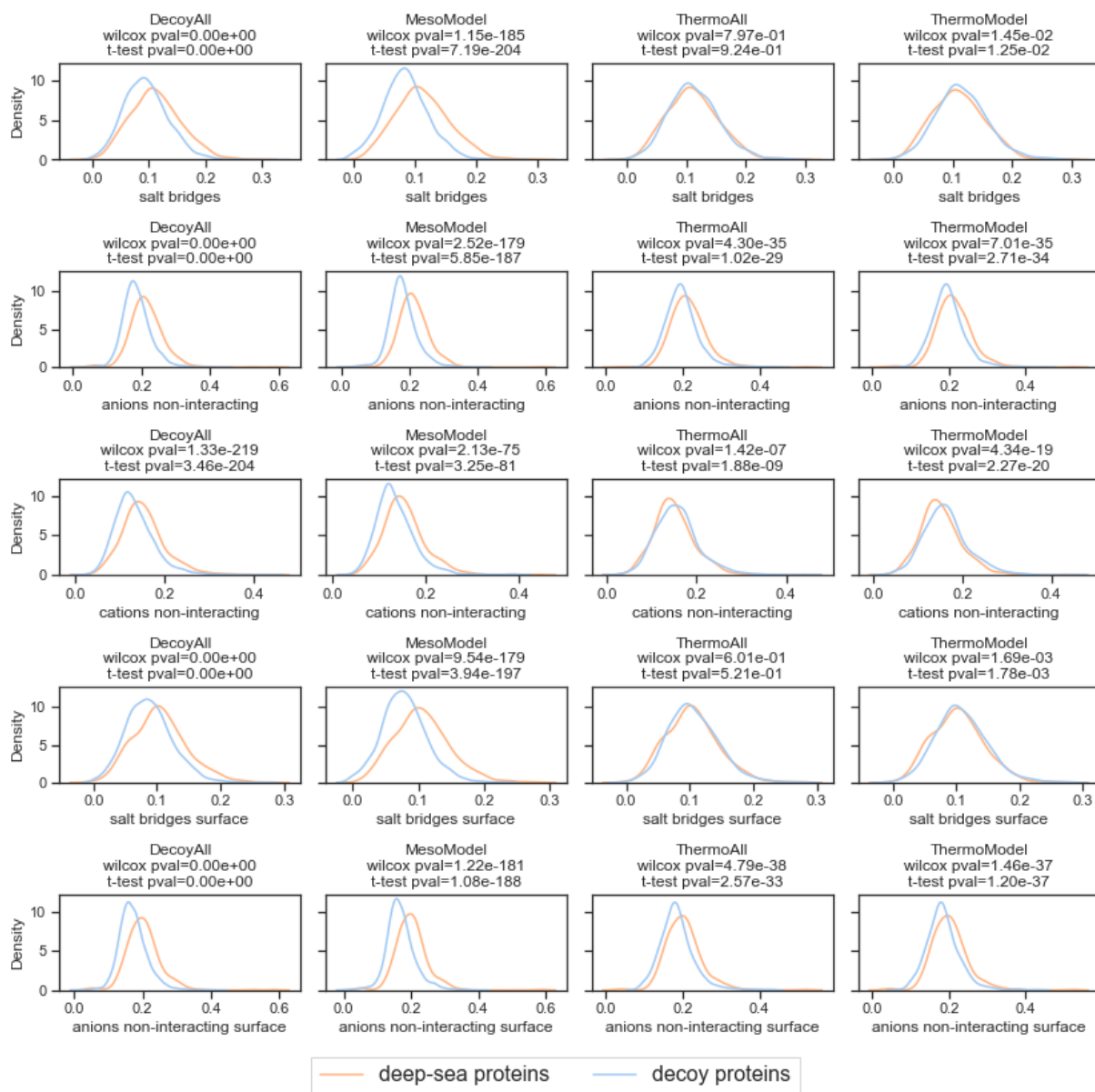


Figure S19: Third distribution plots of single structure features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.



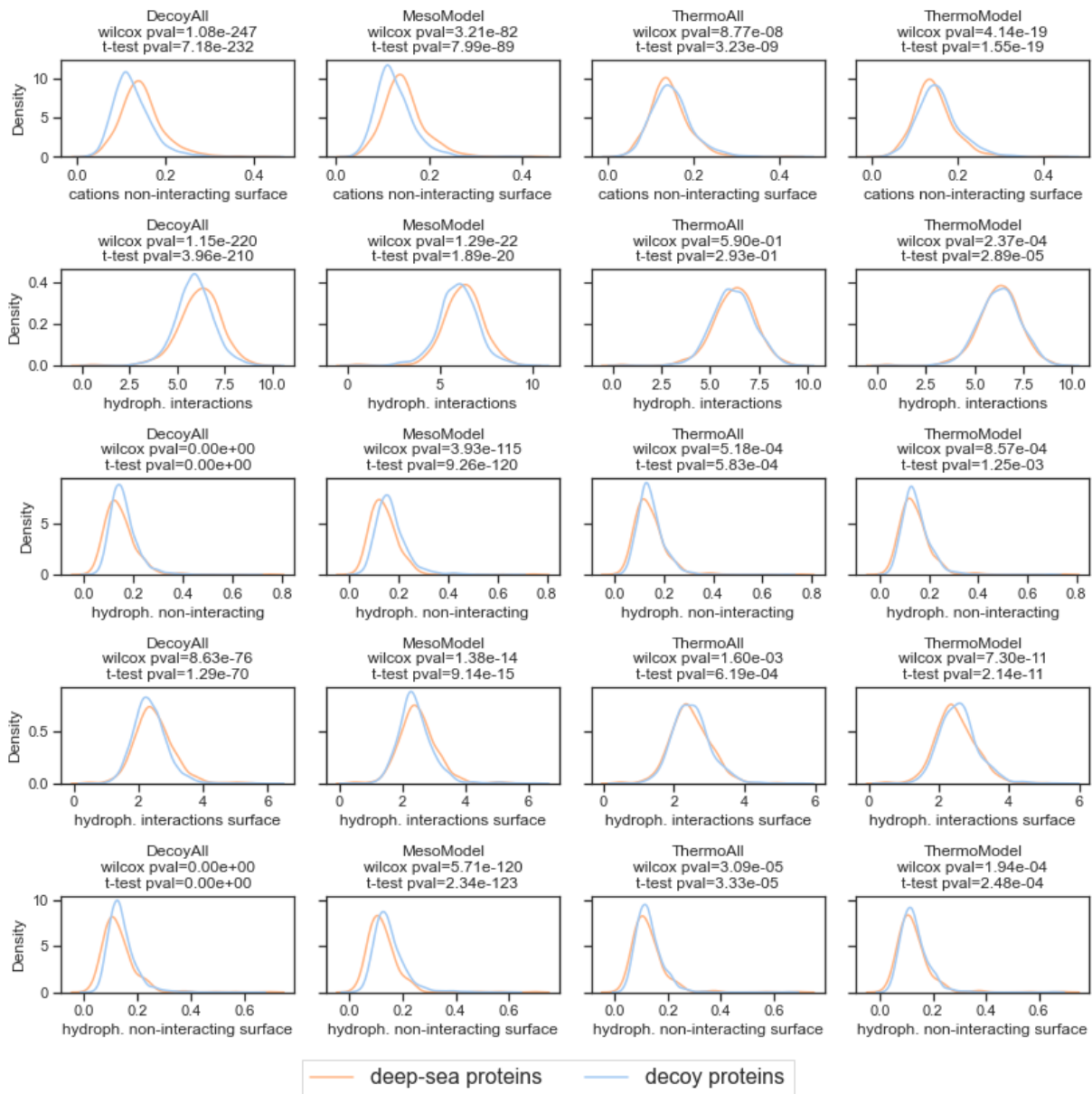


Figure S20: Fourth distribution plots of single structure features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

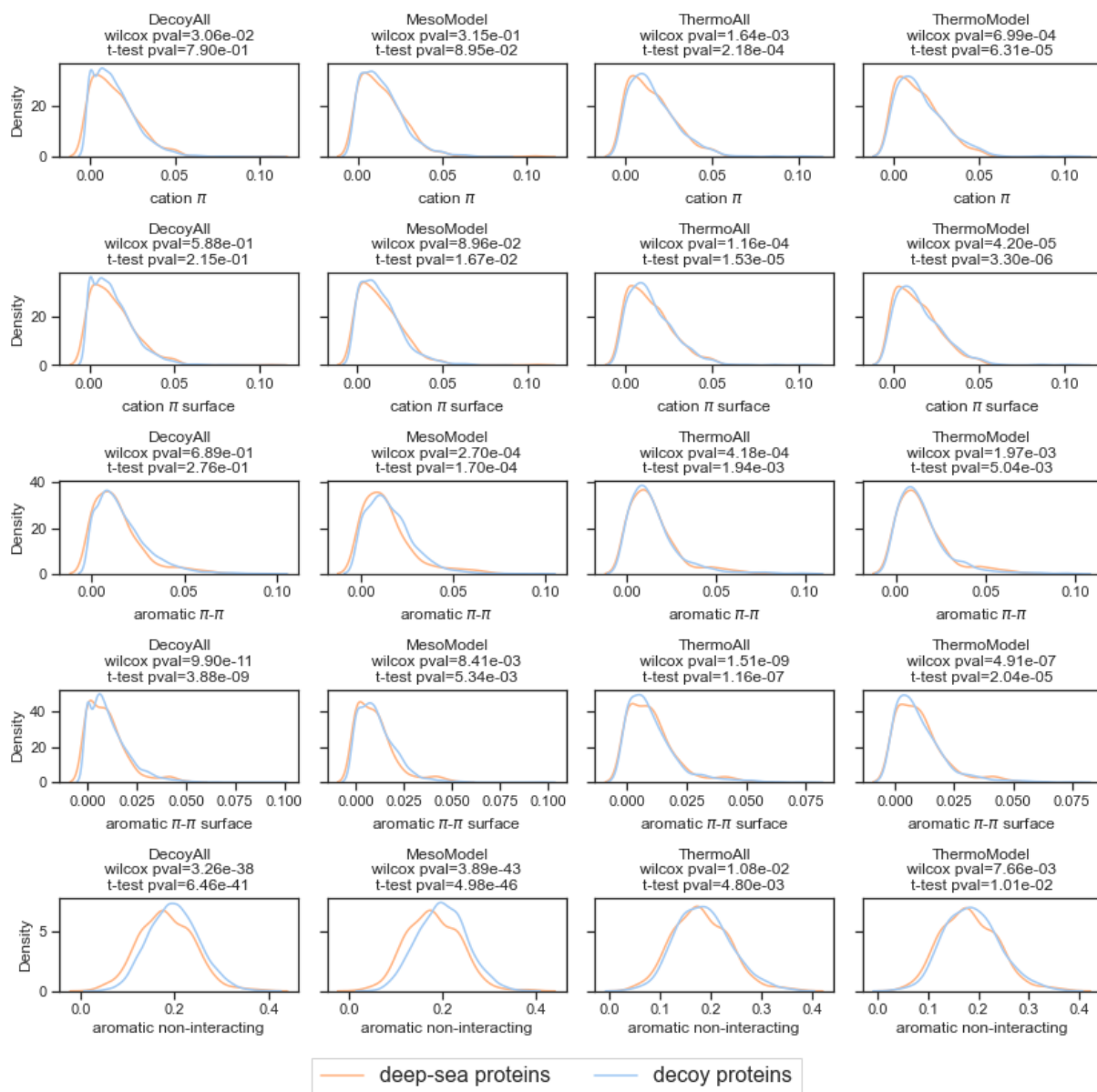


Figure S21: Fifth distribution plots of single structure features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

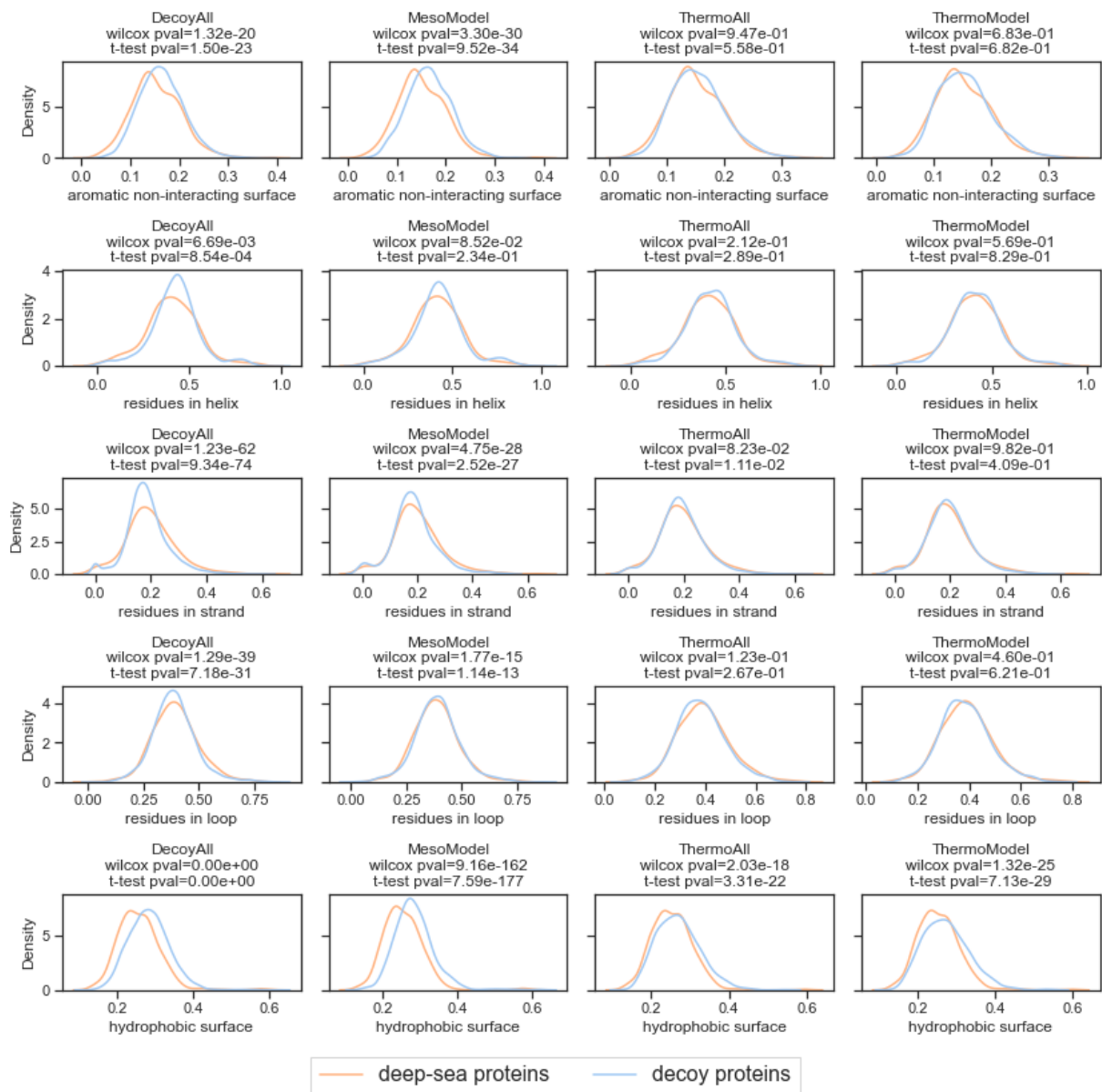


Figure S22: Sixth distribution plots of single structure features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

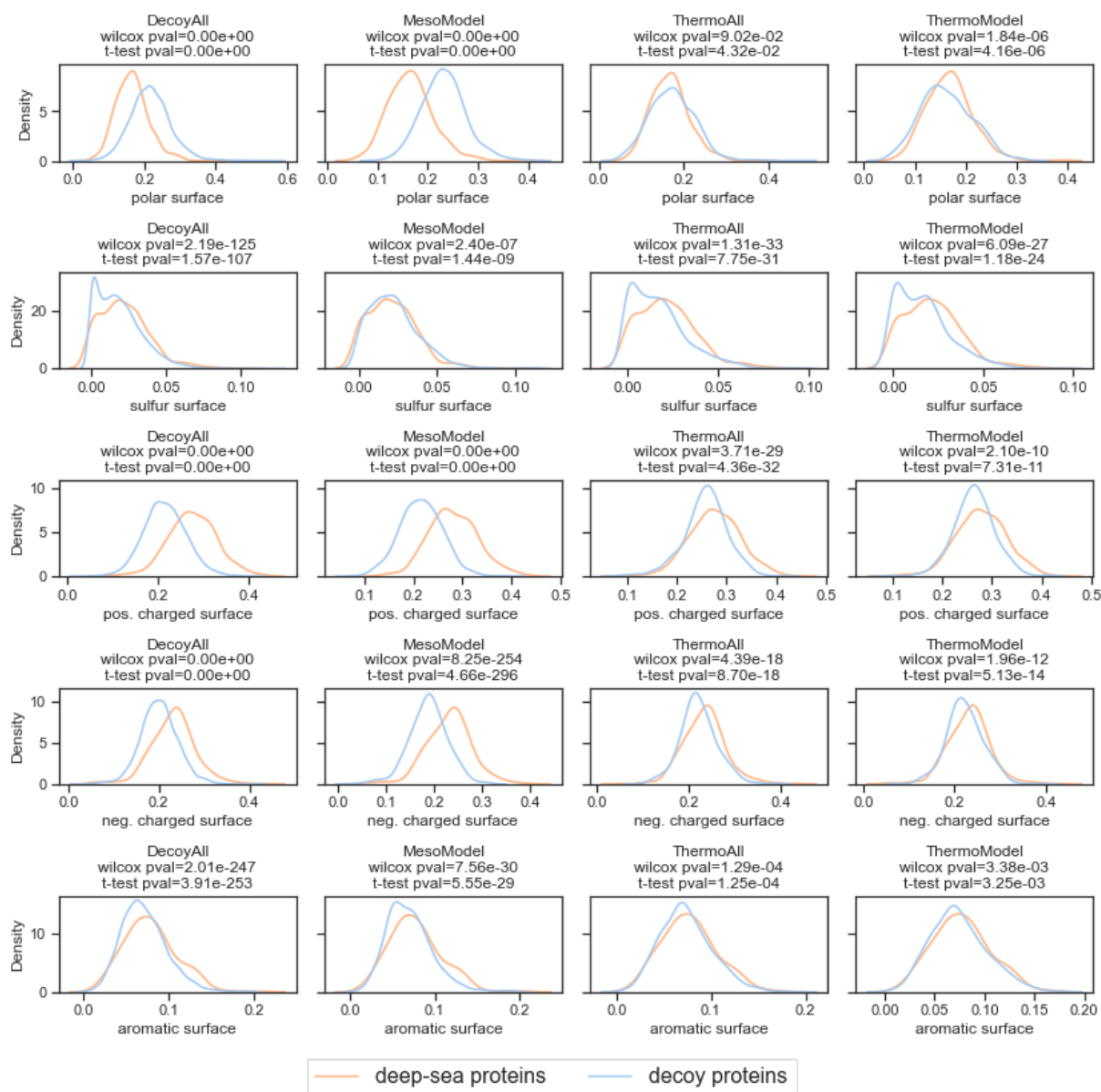


Figure S23: Seventh distribution plots of single structure features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

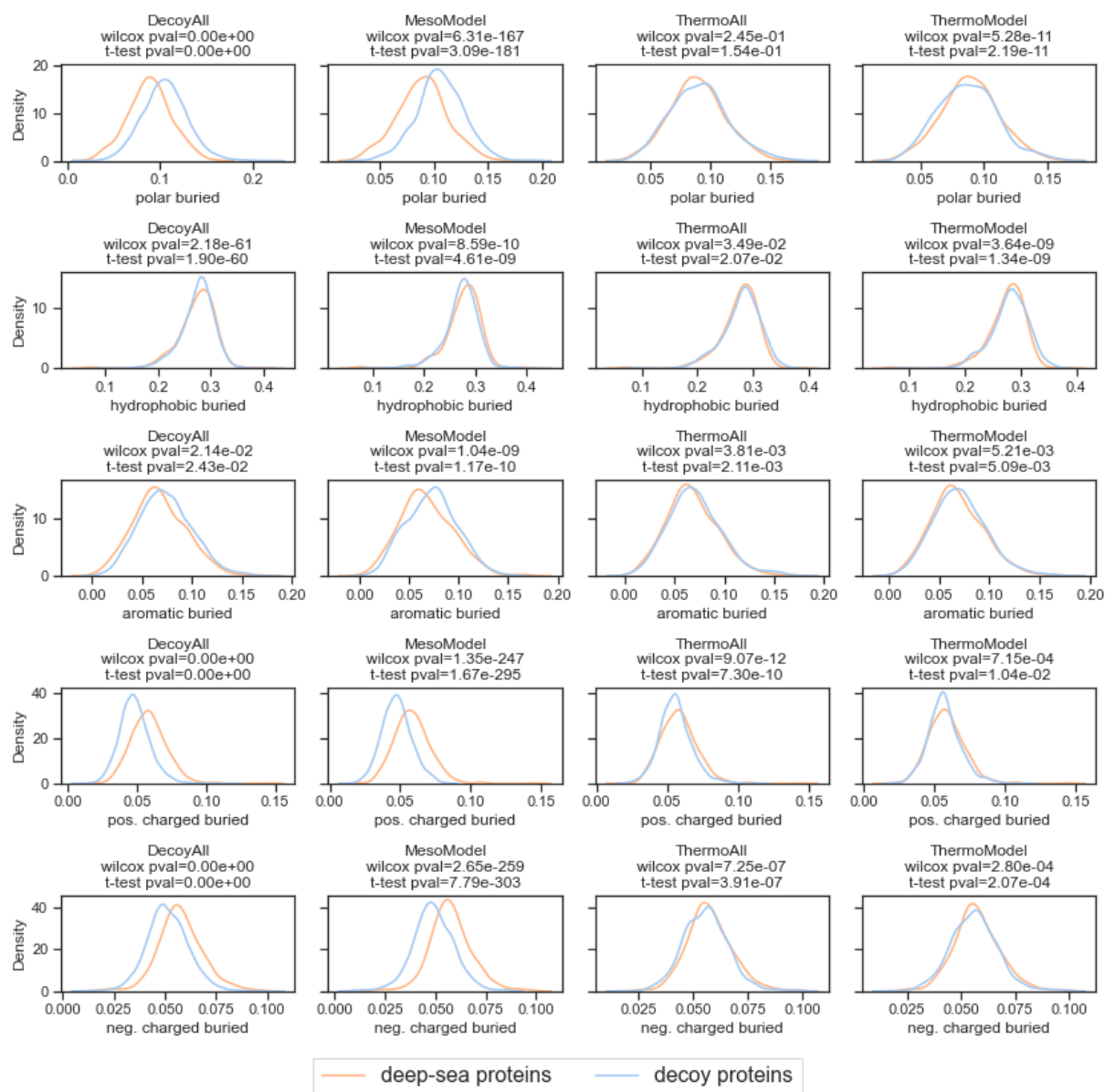


Figure S24: Eighth distribution plots of single structure features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

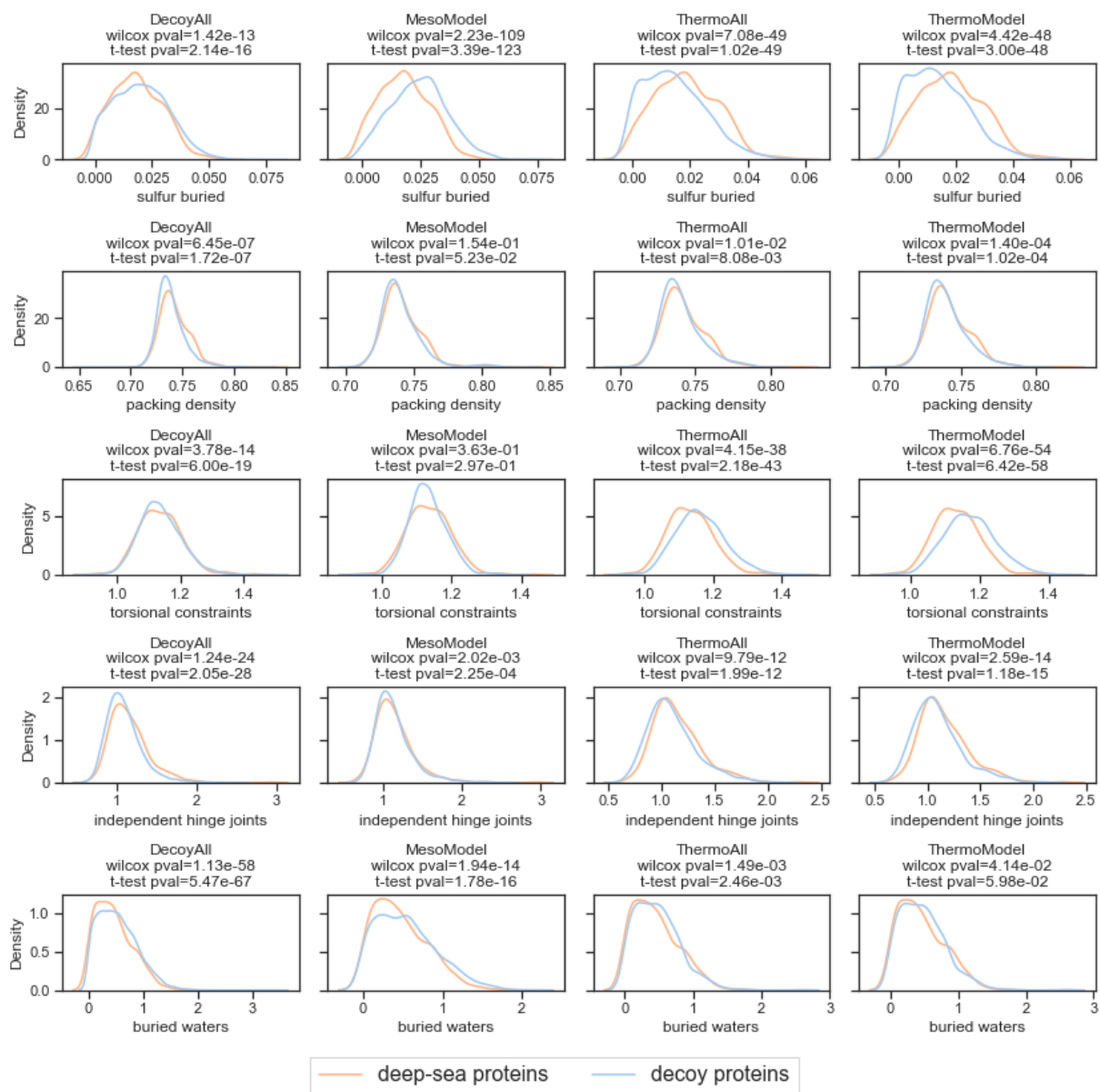


Figure S25: Ninth distribution plots of single structure features for deep-sea and decoy proteins. Distributions show the data of all protein pairs in the 5 cross validation folds of the HT-group respectively for the DecoyAll, MesoModel, ThermoAll and ThermoModel data sets. p-values for paired two-sided t-test and wilcoxon signed-rank test are shown in the title.

## **D.3 Modeling with Alternate Locations in X-ray Protein Structures**

- [D3] T. Gutermuth, **J. Sieg**, T. Stohn, and M. Rarey. “Modeling with Alternate Locations in X-ray Protein Structures”. In: *Journal of Chemical Information and Modeling* 63.8 (2023), pp. 2573–2585.

Available: <https://doi.org/10.1021/acs.jcim.3c00100>. Reprinted with permission from [D3]. Copyright 2023 American Chemical Society.



# Modeling with Alternate Locations in X-ray Protein Structures

Torben Gutermuth,<sup>#</sup> Jochen Sieg,<sup>#</sup> Tim Stohn, and Matthias Rarey\*



Cite This: *J. Chem. Inf. Model.* 2023, 63, 2573–2585



Read Online

ACCESS |



Metrics & More

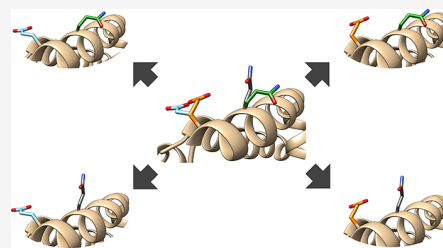


Article Recommendations



Supporting Information

**ABSTRACT:** In many molecular modeling applications, the standard procedure is still to handle proteins as single, rigid structures. While the importance of conformational flexibility is widely known, handling it remains challenging. Even the crystal structure of a protein usually contains variability exemplified in alternate side chain orientations or backbone segments. This conformational variability is encoded in PDB structure files by so-called alternate locations (AltLocs). Most modeling approaches either ignore AltLocs or resolve them with simple heuristics early on during structure import. We analyzed the occurrence and usage of AltLocs in the PDB and developed an algorithm to automatically handle AltLocs in PDB files enabling all structure-based methods using rigid structures to take the alternative protein conformations described by AltLocs into consideration. A respective software tool named AltLocEnumerator can be used as a structure preprocessor to easily exploit AltLocs. While the amount of data makes it difficult to show impact on a statistical level, handling AltLocs has a substantial impact on a case-by-case basis. We believe that the inspection and consideration of AltLocs is a very valuable approach in many modeling scenarios.



## INTRODUCTION

The Protein Data Bank (PDB)<sup>1</sup> provides a wealth of protein and DNA/RNA structures. One significant approximation of many computational methods using structures from the PDB is assuming the structure as rigid. Even one of the essential breakthroughs in recent years, the prediction of protein structures by AlphaFold2,<sup>2</sup> focuses on the prediction of rigid structures. While this achievement and many other method improvements are based on rigid structures, the inherent flexibility still poses a challenge even for the most sophisticated methods.<sup>3</sup> Abandoning this approximation and treating the structure as at least partially flexible is considered one significant opportunity to improve existing methods in drug discovery and design<sup>4–6</sup> and would ultimately provide a more precise description of the true nature of proteins. In a first step, this can be done by exploiting existing experimental data, for example, by building an ensemble of multiple structures. Alternatively, computational methods like molecular dynamics simulations can be used to generate ensembles as well, which can require significant computational effort.<sup>7–11</sup>

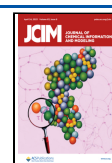
One description of structural flexibility prevalent in the PDB, with almost 42% of structures using it according to our analysis, is alternative locations (AltLocs). AltLocs are utilized by crystallographers to describe parts of the structure that could reasonably be modeled in multiple locations. There exist multiple tools to automatically detect and model AltLocs for side chains as well as backbone atoms<sup>12,13</sup> trying to describe conformational polymorphisms present in the experimental data. However, AltLocs are barely used by tools using PDB structures even though structural flexibility is deemed critical, and their importance in the process of ligand binding is widely

accepted.<sup>14</sup> Using AltLocs is a starting point toward fully incorporating computationally demanding protein flexibility with the added advantage that AltLocs have experimental evidence. While AltLocs have been used rather rarely in studies working with PDB structures, there are examples where the consideration of AltLocs was essential to the success<sup>15</sup> or their negligence an inevitable problem<sup>16</sup> to the study at hand. One reason for the rare use of AltLocs in the literature could be the few tools that work with AltLocs automatically. While there are some options to decide on specific AltLocs manually (using ChimeraX,<sup>17,18</sup> Pymol,<sup>19</sup> or a text editor), which allows deletion of unwanted conformations, this can be tedious or even result in invalid structures. Despite being a well-known problem in the modeling community, it is often neglected, and automatic selection of AltLocs by software libraries or tools is barely described.<sup>20,21</sup>

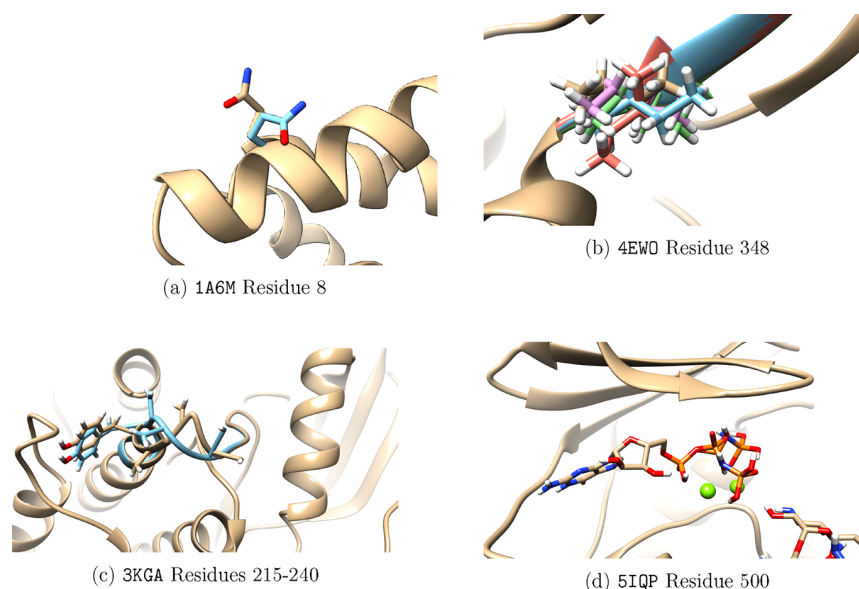
To solve this problem, we developed an algorithm to automatically enumerate different conformations described by AltLocs while considering the occupancy and correctness of the resulting model. This algorithm can be used through a new tool named AltLocEnumerator, allowing users to easily enumerate different conformations of their structures and decide which ones to use for their structure-based task.

Received: January 20, 2023

Published: April 5, 2023







**Figure 1.** Examples for the usage of AltLocs in the PDB. (a) Example of a residue with two AltLocs that differ solely in the side chain position. PDB 1A6M with atoms of residue 8 shown. Multiple AltLocs are colored differently: AltLoc A in tan, AltLoc B in blue. (b) Example of one residue with five different AltLocs that differ in both the side chain as well as backbone atom placement. PDB 4EWO with atoms of residue 348 shown. Multiple AltLocs are colored differently. (c) Example of multiple connected residues with dependent AltLocs. In this case, the AltLocs describe a differently built loop between residues 215–240 in PDB 3KGA. Multiple AltLocs are colored differently. (d) Example of a HETATM entry with two different AltLocs. PDB 5IQB and GNP with atoms of residue number 500 shown.

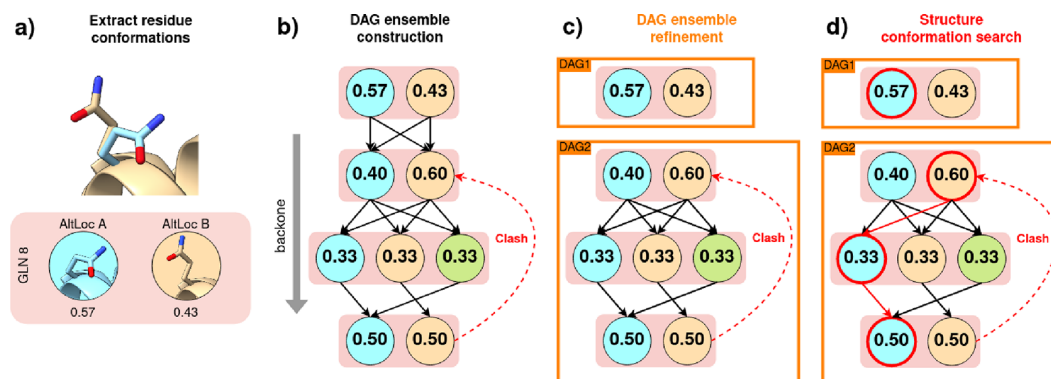
Crystallographers use AltLocs to express alternative atom coordinates when the electron density indicates multiple conformations in parts of the structure. The PDB provides a description of how to annotate AltLocs inside the file (PDB file format specification version 3.30).<sup>22</sup> Interestingly, a general and comprehensive description of when and how to annotate AltLocs does not seem to be available.

For this reason, AltLocs are not annotated entirely consistent through the PDB, which makes the automation of selecting AltLoc conformations more challenging. AltLocs are annotated in the PDB file as multiple ATOM/HETATM entries with different AltLoc identifiers with most structures using the alphabet characters starting with A. Although AltLocs can be used to model multiple alternative, discrete conformations, most describe only two states of a single residue. An occupancy value is assigned to each different conformation, with the more likely conformation having a higher occupancy. All occupancy values of an atom with AltLocs should add up to 1. In many cases, it is difficult for crystallographers to extract substantial differences in occupancy resulting in equal values for all conformations. An example of a simple AltLoc of a single residue with two AltLocs can be seen in Figure 1a. A more complex example of a residue with five different conformations described with AltLocs is displayed in Figure 1b. While AltLocs mostly describe independent side chain conformations and can be trivially combined, some AltLocs of different residues can depend on each other. This dependence can, for example, happen when the AltLocs describe changes in the backbone that only allow the combination of certain AltLocs, or some AltLoc conformations would clash if used in conjunction. An example of the first behavior in which a complete loop is modeled in two different conformations is shown in Figure 1c. Just as ATOM entries in the PDB can have AltLocs, so can HETATM entries, and therefore, small molecules, cofactors, or even water

molecules can be described with multiple conformations or locations simultaneously. An example in which a part of a small molecule in a binding site has multiple locations is shown in Figure 1d.

AltLocs can have a huge influence on any method using structures, for example, by enabling hydrogen bonds that are not possible in another conformation, enabling apolar interactions or opening up subpockets. In the case of Fischer et al., modeling an entire loop in three different conformations that are present in the apo structure enabled them to find novel ligands with new chemotypes that would not have been found in a standard rigid docking procedure.<sup>15</sup> However, this work was done on a single protein with multiple crystal structures, introducing the question how AltLocs are used in the complete PDB. To the best of our knowledge there is only one analysis of the occurrence and variety of AltLocs in the PDB, and it is focused on the effect of AltLocs in side chain conformation prediction.<sup>23</sup> Miao and Cao<sup>23</sup> use a manually curated data set of 3590 protein chains without nucleic acids and describe in great detail the solvent accessibility as well as conformational variety of all AltLocs within this data set. They find 56% of the structures in the data set have AltLocs and highlight many interesting trends, namely, that the presence of AltLocs in structures is correlated with the resolution.

In this work, we analyze the current state of AltLocs in the PDB including their frequency in different residues and dependence on the resolution as well as submission date of the structure, and we show examples of special cases recurring throughout the PDB. Furthermore, we present our new algorithm implemented in the AltLocEnumerator tool for the automatic handling of AltLocs designed to enable any software using rigid structures to automatically use AltLocs. We use AltLocEnumerator and our recently developed docking tool JAMDA<sup>24</sup> to test if using additional complexes described by



**Figure 2.** Algorithm for generating valid structure conformations. (a) Extraction of AltLoc conformations with their occupancies for a residue (PDB 1A6M). (b) Construction of a DAG ensemble. Nodes represent residue AltLoc conformations which are weighted by their occupancy score. Compatible residue conformations consecutive in the backbone are connected with a directed edge. Clashes are tracked as indicated by the dashed red arrow. (c) DAG ensemble is refined. Fully connected layers are split into separate DAGs, but DAGs connected through a red edge (clash) are merged. (d) Search for a valid conformation of the entire structure. In this example, residue conformations are selected to maximize the occupancy score (as indicated by the red path) while avoiding clashes and chain breaks.

**Table 1. Different AltLoc Generation Strategies<sup>a</sup>**

	Output conformations	Name	Description
(i)	single	best occupancy score	Generation of a single structure conformation with the maximal occupancy score
(ii)	multiple	all best occupancy score	Generation of all structure conformations with the same maximum occupancy score
(iii)	multiple	enumerate all	Enumeration of all possible valid structure conformations
(iv)	single	first encounter	Selection of structure conformation based on the first encountered AltLoc identifier while reading the file
(v)	single	specific AltLoc-ID	Selection of structure conformation with a user-specified AltLoc identifier, for example, all AltLoc conformations with identifier "B"

<sup>a</sup>The first three strategies use the algorithm to generate valid structures, while the last two simply select AltLoc conformations based on a single AltLoc identifier.

AltLocs yields better results and showcase some examples in which AltLocs were especially important.

## METHODS

We present a fast branch-and-bound algorithm to generate valid alternative protein structure conformations described through AltLoc annotations. The algorithm searches for compatible residue conformations maximizing the conformational states' probabilities by scoring the AltLoc occupancy values.

In general, there is an exponential number of ways to combine single residue conformations. For this reason, enumerating all combinations becomes infeasible as soon as structures contain a higher number of residues with AltLoc conformations. The aim is, therefore, to efficiently rule out invalid AltLoc conformations early. To solve this problem, we developed a constrained conformational search which avoids combinations of residue conformations that lead to atom clashes or introduce backbone chain breaks. The algorithmic workflow is illustrated in Figure 2 supporting five different search strategies listed in Table 1.

The first three search strategies ((i), (ii), (iii)) use the algorithm depicted in Figure 2 and therefore systematically check for chain breaks and clashes of AltLocs. In contrast, the second two ((iv), (v)) are simpler strategies that select residue conformations based on a single AltLoc identifier. However, all strategies check overall structural validity using NAOMI.<sup>25,26</sup> In case no valid structure conformation is detected, no structure is returned. Strategy (iv) was our default strategy

before and is our baseline method in the experiments of this work.

**DAG-Based Search Space Construction.** The alternative residue conformations need to be enumerated systematically to find valid overall conformations of the structural complexes of protein, nucleic acid, and other molecules. The central data structure, named AltLoc-DAG, is an ensemble of directed acyclic graphs (DAGs) to represent a reduced search space of only the compatible residue conformations (see Figure 2a–c). Conformations are grouped into the same DAG if they cannot be modeled independently.

To construct the AltLoc-DAG for a given input structure, we first extract all atoms with at least two alternate location identifiers. Although theoretically feasible, we ignore cases where AltLocs are used to model different molecules, for example, different amino acids. The resulting list of all residues with AltLocs is then annotated with occupancies by taking the mean of the occupancy values of the atoms with AltLoc annotations.

Atom clashes are calculated between all pairs of residue conformations using the AltLoc annotated atoms. After visualization of the first results, we set the clash threshold at more than 35% overlap of the van der Waals radii of two heavy atoms, i.e.,

$$\frac{vdW_{sum} - d}{vdW_{sum}} > 0.35$$

The AltLoc-DAGs nodes represent the individual residue conformations. Node weights are set with the occupancy of the residues' conformations. Then, edges are added organizing the

AltLoc-DAG in layers, where each layer contains the set of alternate conformations for a single residue. An edge is added between nodes whose residues are adjacent in the macromolecular sequence, and their conformations can be connected by a backbone peptide bond or phosphodiester bond for nucleic acids. In this way, we do not add edges when combinations of conformations would discontinue the backbone and introduce chain breaks. We use the NAOMI library<sup>25,26</sup> to determine the linear backbone connections. In addition, we use NAOMI to identify disulfide bridges between residue conformations. Conformation pairs forming these bonds are considered as not clashing.

The initial DAGs are then split between two consecutive layers exactly if two conditions hold (see also Figure 2, steps b and c): First, the two subsequent layers are fully connected; i.e., all alternate locations are pairwise compatible. Second, there is no red edge indicating a clash connecting any layer below to any layer above it.

**Determining Valid Structure Conformations.** We implemented three different strategies to search the DAG ensemble for valid overall structure conformations (see Table 1 (i), (ii), and (iii)). All strategies search for one or more valid paths in each DAG from the top to the bottom layer. A path  $p = v_1, \dots, v_n$  is valid if  $n$  equals the maximal depth of the DAG, and there is no node pair in the path with a clash. If there is not at least one valid path for all DAGs in the ensemble, then there exists no valid conformation of the overall structure.

In the first two search strategies (i) and (ii), we employ a scoring function to select one or multiple valid and likely overall conformations. This scoring is guided by the structure's occupancy annotations. Occupancy values estimate the amount of each AltLoc conformation observed in the crystal experimentally. Therefore, occupancy values of residue conformations can be interpreted as probabilities of the independent events that the residue is observed in a certain conformation. Our score is inspired from a maximum-likelihood estimation based on these probabilities. However, in practice, conformations of different residues might be dependent; e.g., they clash or introduce chain breaks. Therefore, the goal is to select a valid overall conformation by maximizing the occupancy values under these dependencies. We achieve this by searching for valid paths in each DAG while maximizing the following scoring function

$$s_{occ} = \sum_{i=1}^n \log(occ(v_i))$$

where  $v_i$  is the  $i$ th node in  $p$ , and  $occ$  is a function that retrieves the occupancy of  $v_i$ .

The first two search strategies (i) and (ii) generate a single conformation of the maximal occupancy score and all conformations of the maximal occupancy score, respectively. The difference between the two search strategies is that the first strategy retrieves only the first occurring path with a maximal score, while the second additionally considers and stores equally scoring paths. This property is useful because occupancy values are often uniformly distributed between residue conformations yielding not a single best overall conformation but multiple.

A DAG is traversed starting from the top layer nodes with a depth-first search-like procedure to obtain an initial solution. We apply a greedy search approach to provide a good scoring first solution by visiting adjacent nodes in decreasing order of

their occupancy. Before visiting a new node, it is checked if the residue conformation representing the new node clashes with any conformation in the current path indicated by a red edge from the new node to one of its predecessors. If there is a clash, the node will not be visited. When a target node in the last layer is found, we save the path and its occupancy score as a valid solution if it is the currently best solution according to the occupancy score (strategy (i)) or not worse than the best solution found so far (strategy (ii)). In addition, to each node during the recursive backtracking, we annotate the best score achievable from this node (ignoring clashes). Once a valid solution is found, we search in the remaining graph for better solutions while pruning subgraphs we already visited when they can provably not lead to a better scoring path.

The third search strategy (iii) simply enumerates all paths and, therefore, all possible valid overall structure conformations. Here no occupancy scoring is used, but just the DAGs are traversed while considering clashes.

**AltLocEnumerator.** The algorithm described above forms the central component of a software tool named AltLocEnumerator. We implemented AltLocEnumerator within the NAOMI framework.<sup>25,26</sup> AltLocEnumerator takes a structure file in PDB or mmCIF format as input and generates one or more conformations of the contained structure that can be exported as PDB files. In addition to the described algorithm, AltLocEnumerator contains the following options and features:

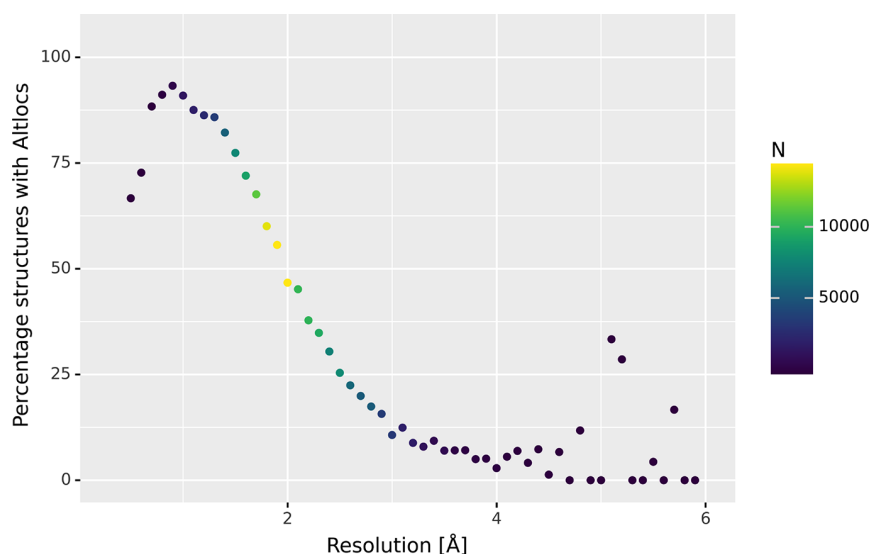
**Standard AltLoc Strategies.** In addition to the more sophisticated strategies (i)–(iii), AltLocEnumerator also implements the more simple strategies (iv) and (v), which represent often employed standard strategies. In these strategies, resulting conformations are validated with NAOMI, but no additional clash or chain break detection is employed. For example, strategy (iv) is somewhat arbitrarily selecting some AltLoc identifier based on the ordering in the file (which will mostly be identifier “A”). This was our previous standard strategy. In contrast, strategy (v) allows the user to specifically select all residue conformations in the structure of a particular AltLoc identifier. This option can, for example, be used if the authors of the structure file intended to represent functional or structural meaning to the conformation with a specific identifier.

**Limit Number of Enumerated Structures.** For the search strategies yielding multiple overall structure conformations (strategies (ii) and (iii)), the user can provide a maximum number of structures to return. The resulting structure conformations will be sorted by occupancy, according to the greedy depth-first search-like traversal. This option can be useful when there is a combinatorial explosion of residue conformations leading to millions of valid structure conformations.

**Specify Local Sites.** The algorithm can focus on a local site or subset of residues in the structure. With this option, only AltLoc conformations of a user-specified subset of residues is considered for conformation generation. This option helps to reduce the number of generated structures. For example, for most protein–ligand docking tools, only the residues belonging to the active site are of interest. A list of residues can be specified as a PDB or mmCIF file or in a text-based configuration file. Alternatively, a ligand file can be provided, in which case all residues in a user-defined distance to any ligand atom are used to define a surrounding binding site.

**Remove HETATMs.** Before running the algorithm, all HETATMs can be removed from the input. This option





**Figure 3.** Percentage of structures resolved with X-ray diffraction containing at least one AltLoc on the y-axis and the resolution of the structures on the x-axis. The number of structures present at each point is color coded.

enables building some protein and DNA/RNA structures by stripping off HETATMs that would clash otherwise.

**Filter by RMSD.** An RMSD (root-mean-square deviation) threshold can be provided to filter generated valid structures by their structural similarity. All structure conformations below the user-given RMSD difference threshold to a previously generated conformation will be omitted, resulting in smaller and more diverse conformation sets.

**Data Sets.** The Protein Data Bank<sup>1</sup> is the largest and continually expanding public repository for protein and nucleic acid structures and contains X-ray, NMR, and cryo-EM structures. We used the state of the 21.03.2022 for the analysis in this paper, containing over 185,000 structures. A full list of all PDB codes can be extracted from the [Supporting Information](#).

The sc-PDB<sup>27</sup> is a data set extracted from the PDB describing a large collection of drug-like protein–ligand complexes. It contains 17,594 binding pockets of 16,359 different PDB structures enabling the simulation of early phase drug discovery scenarios. Since all proteins within the sc-PDB are in mol2 format and NAOMI<sup>25,26</sup> can only read proteins in the PDB/CIF formats, the respective PDB files from the current PDB release were used as input. The reference ligands of all binding pockets in the sc-PDB were utilized to only enumerate relevant AltLocs within 6.5 Å to the reference ligand.

**Extracting AltLocs Statistics from the PDB.** To extract data about AltLocs from the PDB, bash scripts were used that extract all residues together with their ID, residue name, chain, and number of conformations. This data can then be used to calculate other information like how many AltLocs are present per PDB structure or the distribution of the number of conformations per AltLoc. This analysis is done in separate steps for all ATM/HETATM entries in the PDB to differentiate between the residue types. The scripts used to extract the information and the resulting data can be found in the [Supporting Information](#).

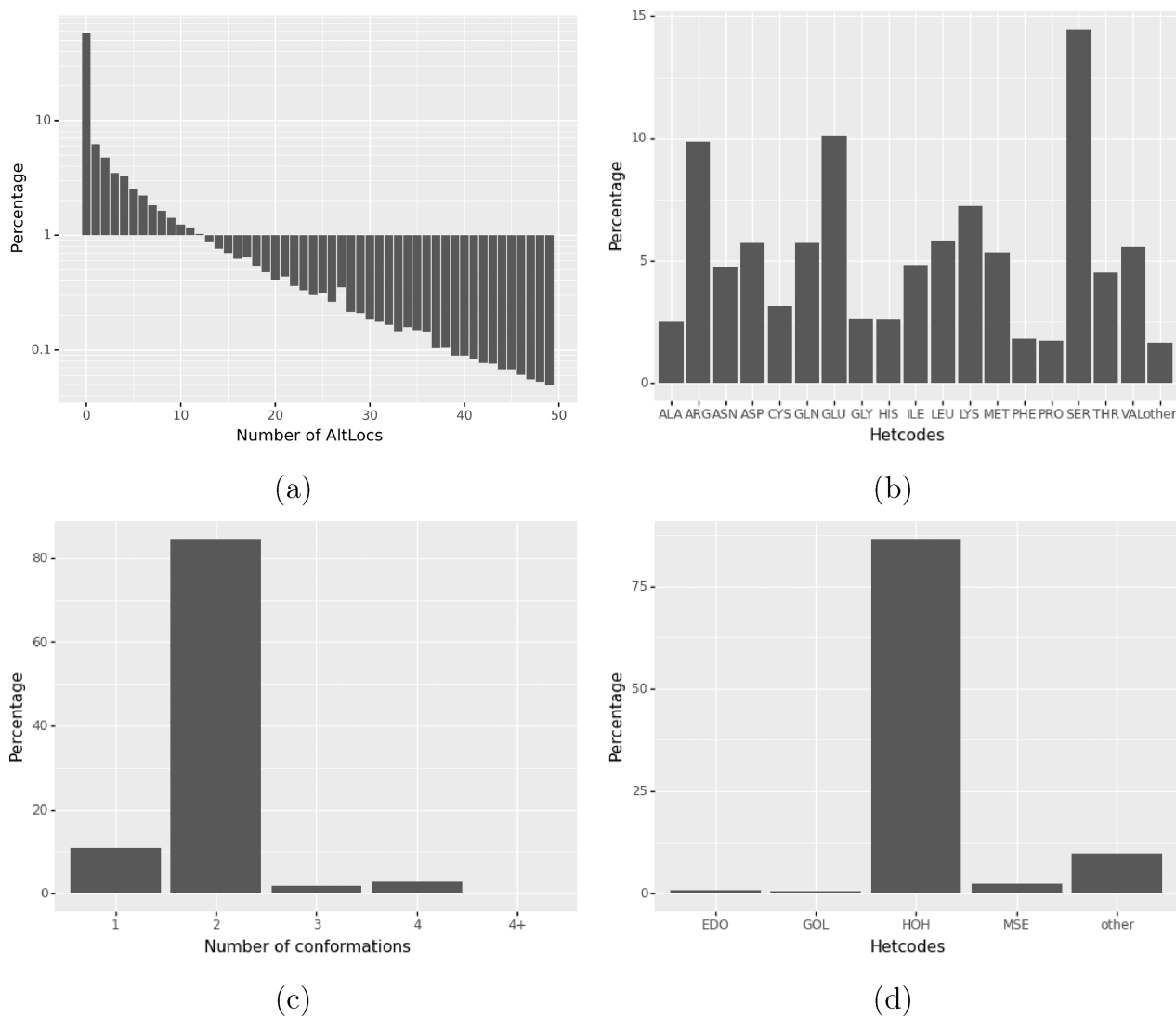
**Runtime Evaluation.** We evaluate the runtime of AltLocEnumerator, its enumeration strategies (i) to (iv), and its running modes on the subset of 2700 structures of the sc-

PDB that contains AltLocs. In addition to the four strategies, we evaluate the runtime when we focus on the AltLocs of the whole structure in comparison to only the AltLocs of a single binding site of interest. We define the local site as the ligand binding site using all residues at a distance of 6.5 Å from any ligand heavy atom. We restrict the maximum number of structures enumerated to 513 in the strategies creating ensembles. Experiments were conducted on a standard desktop machine (Intel(R) Core(TM) i5-9500 CPU 3.00 GHz, 16GB DDR4 RAM, NVMe TOSHIBA 512GB).

**Preparation and Docking of the sc-PDB.** The PDB files were prepared by removing all ligands, only keeping relevant water molecules (two or more hydrogen bonds to the protein and reference ligand) and choosing standard protonation states of all amino acids. The initial placement of the molecule to be docked is restricted to within 6.5 Å of the reference ligand. All molecules were redocked into the prepared receptors with JAMDA<sup>24</sup> using the ligand file provided in the sc-PDB, and the RMSD to the crystal pose was calculated. An arbitrary number of poses can be investigated in the redocking, and we chose the best scored pose, best two, best three, best 10, and best 32. In this way, unsuccessful redockings can be differentiated to dockings in which good docking poses are scored low.

## ■ RESULTS AND DISCUSSION

**AltLocs in the PDB.** Following up the development of the AltLocEnumerator, we wanted to investigate how AltLocs are used in the PDB. Doing this analysis there are a plethora of reasons for any given trend, and many are based on the tools and preferences of crystallographers rather than the data that were collected. As these statistics on the complete PDB are not published anywhere else, we report them here to inform the reader, but complete interpretation of the data exceeds the scope of this publication. According to our analysis, 41.97% of the structures in the PDB (state 21.03.2022) contain at least one AltLoc. Interestingly, the data set by Miao and Cao<sup>23</sup> is enriched in AltLocs compared to the complete PDB and contains 56% of structures with AltLocs but otherwise follows many similar trends as seen in our analysis of the complete PDB. One commonality of both, the number of AltLocs in a



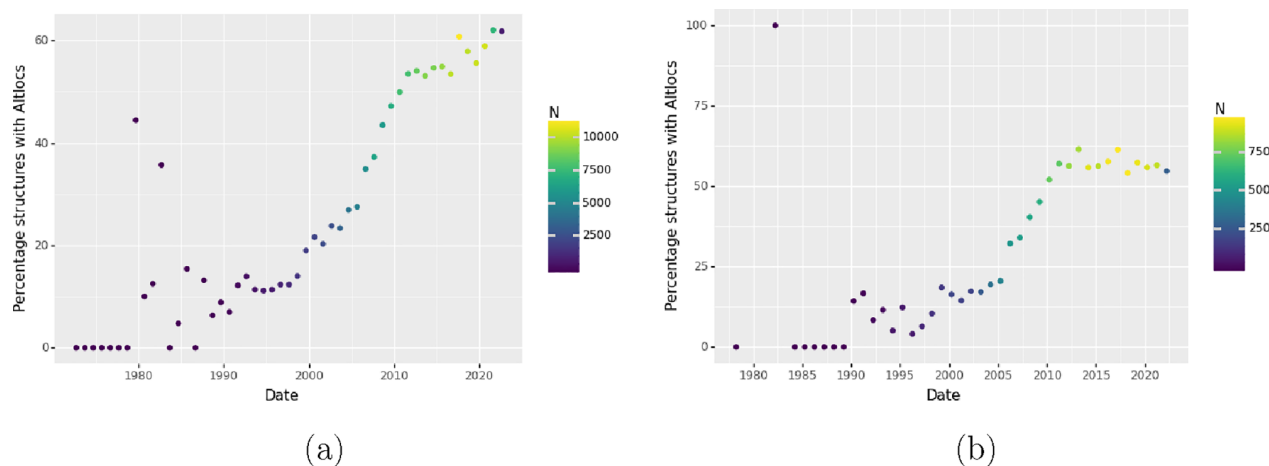
**Figure 4.** AltLoc statistics for the complete PDB (state 21.03.2022). (a) Percentage of structures containing  $x$  residues with an AltLoc. (b) Percentage of structural residues with an AltLoc by residue type. Residues are all structural subunits modeled with ATOM entries. (c) Distribution of the number of conformations per AltLoc. (d) Percentage of hetero groups with AltLoc by molecule type. Hetero groups are all structural subunits modeled with HETATM entries.

PDB structure resolved using X-ray diffraction heavily correlates with the structure's resolution: Well resolved structures contain more AltLocs than ones with lower resolution (see Figure 3). Structures resolved using electron microscopy show low usage of AltLocs (see Figure S1).

The reason for this might be that only high resolution structures contain enough data to model further parameters to the model by adding AltLocs to it.<sup>28</sup> Unsurprisingly, the higher the number of residues with AltLocs gets, the fewer structures can be found with this number of AltLocs with an exponential decline, as seen in Figure 4a. Most structures in the PDB contain no AltLocs at all, but more than 21% also contain six or more residues with AltLocs. Although rare, some structures can contain more than a thousand residues with an AltLoc (PDB 5TQP), and every AltLoc is not limited to two different conformations. The number of conformations per AltLoc can be seen in Figure 4c. Over 84% of AltLocs describe two conformations, while a surprising amount of  $\sim 10\%$  contains only one conformation. This behavior is counterintuitive because using an AltLoc with only one conformation defeats

the purpose of the AltLoc describing multiple possible conformations simultaneously. One reason is that sometimes crystallographers build a model with a part of the structure that is only present in one AltLoc and not present in another, yielding parts of the structure with a single AltLoc. This can be seen in the carbonic anhydrase II complex (PDB 6ROB). Another reason is that some structures, especially ones published at the beginning of the 2000s, use AltLocs to distinguish differently built models. This can be seen in the thymidylate synthase complex (PDB 2AAZ) which describes all residues in Model 1 as AltLoc A and all in Model 2 as B, which is contrary to how AltLocs should be used in our opinion. Three and four conformations per AltLoc are still reasonably frequent, while a higher number of conformations is scarce with a maximum of seven conformations per AltLoc (e.g., PDB 3B2C) and a single structure (PDB 2V93) with up to 13 conformations. The type of residue for which an AltLoc is used is not equally distributed, as seen in Figure 4b.

Almost 15% of all AltLocs in ATOM entries occur in serine residues. Following in frequency are glutamic acid, arginine,



**Figure 5.** Percentage of structures with AltLocs over time (state 21.03.2022). The number of structures present at each point is color coded. (a) Percentage of structures with AltLocs over time. (b) Percentage of structures with AltLocs over time with a resolution between 2.0 and 2.2 Å.

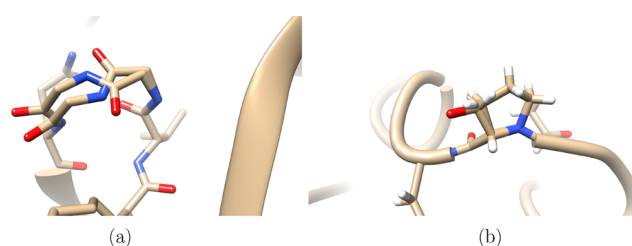
and lysine, which are more flexible than serine and can also form strong Coloumbic or hydrogen bond interactions. Aspartic acid is only half as frequently modeled with AltLocs as its longer counterpart glutamic acid, which might be due to the increased flexibility of the latter. Due to the prevalence of proteins inside the PDB, most AltLocs are inside amino acids, but RNA/DNA structures also contain AltLocs. Apart from serine, the general trend seems to be that more flexible amino acids tend to be modeled with AltLocs more frequently. This frequency of residues is broadly in line with the previously mentioned analysis of Miao and Cao.<sup>23</sup> The residue names of all HETATM AltLoc entries are dominated by water (>84%), as seen in Figure 4d. Besides water, AltLocs of HETATM entries are diverse and span over many chemically different molecules. At the same time, crystallization additives (e.g., glycerol, 1,2-ethandiole), simple metals/ions (e.g., sulfate, chloride ions), and cofactors are more common probably due to their respective higher frequency in the PDB.

Another interesting trend is the usage of AltLocs in the history of the PDB, as they become more popular over time. While between 1990 and 2000 only about 10% of structures contained residues with AltLocs, in the past 10 years, around 60% of all structures contained AltLocs, as seen in Figure 5a. This trend may be correlated with improved programs for structure modeling as well as methods and, therefore, improved published structures' resolutions. However, the trend is still present even when restricting the resolution to small ranges like, for example, from 2.0 to 2.2 Å (Figure 5b), showing that AltLocs are still getting more frequent even when corrected by the structures' resolutions. The complete statistics and the scripts to generate them can be found in the Supporting Information.

**Validation of the AltLocEnumerator.** To validate that AltLocEnumerator works as intended, we compared the AltLocs found and the number of structures generated from AltLocEnumerator to the data created when parsing all PDBs for the sc-PDB subset. From the 17,594 binding pockets in the sc-PDB, 2700 (15.04%) contained AltLocs. For each binding pocket, between 1 (AltLocs present, but only one clash-free conformation version could be built using NAOMI) to 513 (maximum number of conformations) different structure conformations are built. The number of structures generated for the data-based approach was calculated as the product of

residue conformations with an AltLoc in the PDB, which ignores compatibility issues or dependencies between AltLocs. The residues that were different between generated structures and the number of structures generated in total were recorded. AltLocEnumerator and our PDB analysis do not differ in the residues that have AltLocs but in the number of structures generated. For the complete sc-PDB subset, 2546 of the 16,359 (15.56%) differ in the number of structures generated. The vast majority of this difference is due to the maximum number of structures that will be enumerated using AltLocEnumerator. Removing all PDB entries in which AltLocEnumerator enumerates the maximum number of structures, only 370 of 14,183 (2.61%) differ. Since AltLocEnumerator considers the compatibility of AltLocs from different residues, we can expect a lower number of conformations compared to full enumeration.

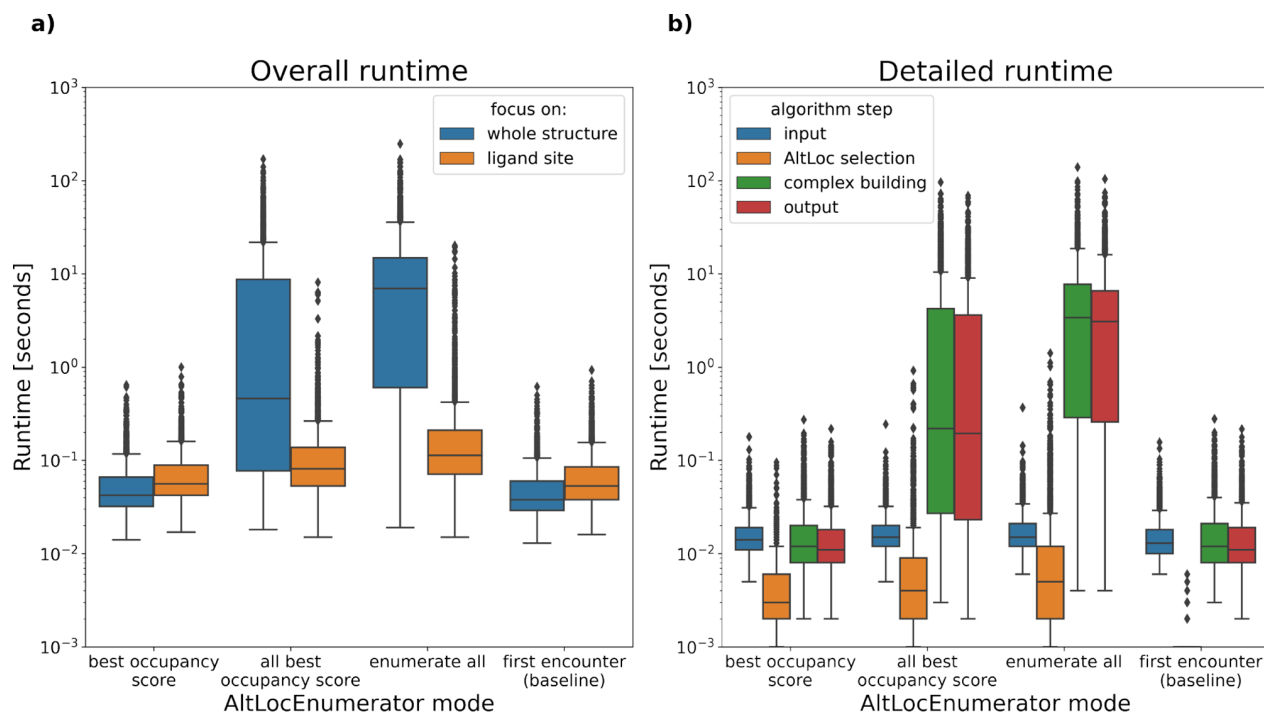
An example is the HIV protease complex (PDB 4J54) in which the residues 16, 17, and 47 each have two alternative conformations modeled as AltLocs (see Figure 6a). From the



**Figure 6.** Highlights of special cases of AltLocs that introduce differences in the validation. (a) HIV protease complex (PDB 4J54) is shown in which the AltLocs of residues 16 and 17 are dependent on each other, as enumerating them would introduce backbone breaks. (b) Crambin complex (PDB 1JXU) is shown in which the same residue is described as two different amino acids (serine, proline) using AltLocs.

eight possible complexes, only four remain due to incompatibility of backbone atoms. Another reason for conformer reduction are clashes between specific AltLocs and other parts of the structure like HETATM entries.

We also noticed that in some rare cases AltLocs are used to model alternative amino acids or molecules rather than conformations. The Crambin mixed sequence complex (PDB



**Figure 7.** AltLocEnumerator runtime analysis on sc-PDB structures with AltLocs ( $n = 2700$ ). (a) Overall runtime of AltLocEnumerator for four different enumeration strategies. Showing runtime when focusing on the whole structure and only on the local ligand binding site. (b) Runtime of each step of the tool for four different enumeration strategies using the whole structure. “input” denotes the time needed to process the input data. “AltLoc selection” shows the time for running the core algorithm for selecting residue conformations. “complex building” represents the time for NAOMI to build and chemically validate the overall structure. “output” refers to the step for writing the built structure as a PDB file. Time values below  $10^{-3}$  were set to  $10^{-3}$  for illustration purposes.

IJXU) is an example for this with residue 22 being modeled as either serine or proline (see Figure 6b). Since this is, in our opinion, not the intended use of AltLocs, we decided not to model the change of molecular entities.

Some differences can be related to our codebase NAOMI.<sup>25,26</sup> For example, NAOMI focuses on the first model inside any structure and ignores subsequent models, yielding fewer conformations than a simple enumeration of all AltLocs in a PDB file. Another reason is AltLocs with a single conformation. As AltLocEnumerator completely neglects any single conformation AltLocs and the data analysis does enumerate one structure, this can lead to a difference in the analysis if there are no other multi-conformation AltLocs present. One example is the thymidylate synthase complex (PDB 2AAZ), in which all atoms in Model 1 are written in AltLoc notation as AltLoc A, and all atoms in Model 2 are written as AltLoc B, and this trend continues as the model numbers increase. As we discard any information on subsequent models using NAOMI, connecting AltLocs described in this way is not possible. This usage of AltLocs is contrary to how we believe they should be used and is not modeled in our algorithm, as all residues in different models are treated as independent. One last interesting case is the PTP1B complex (PDB SQEL), which is part of a group deposition created with the PanDDa<sup>29,30</sup> method. In this case, the complete protein is described using AltLocs, but almost all residues are entirely identical in all described AltLocs. The intention of this is unclear to us, but we suspect that AltLoc identifiers are given consistently across the group deposition. As our algorithm does not check if AltLocs are identical before trying to enumerate them, this leads to a numerical explosion

that inhibits correct handling of AltLocs resulting in different models. However, if the intent of the AltLoc labeling is clear, such cases could be handled by a specific AltLoc-ID with strategy (v).

**Runtime.** The results of the runtime experiments on a standard desktop machine are illustrated in Figure 7. The evaluation shows that generating a single valid structure conformation with the highest occupancy score is approximately as fast as the much simpler baseline strategy (compare “best occupancy score” and “baseline” in Figure 7a). Therefore, AltLocEnumerator allows selecting the most probable valid conformation without noteworthy loss in computation efficiency.

For most AltLoc selection strategies, AltLocEnumerator takes less than one second on average. Only the “enumerate all” strategy on the whole structure takes more time (median = 6.96 s, mean = 11.88 s), which is not surprising since it is the strategy generating most conformations.

Focusing the AltLoc enumeration on a site of interest, e.g., a ligand binding site, harms the runtime when generating only a single structure conformation but improves the average runtime considerably when enumerating multiple conformations. The processing time of the local 3D site is probably only profitable after a certain number of conformations, for which enumeration can be avoided by focusing on the local site.

Figure 7b shows the runtime of each intermediate step of the algorithm. On average, the “AltLoc selection” step (representing the algorithm in Figure 2) is the least time-consuming step. Especially in the strategies enumerating multiple conformations, the complex building time and the time to write the output PDB files dominate the runtime.



**Significance of AltLocs for Docking.** Most existing docking tools use structures from the PDB on an as-is basis and ignore AltLocs. This introduces the question of whether considering AltLocs is beneficial for structure-based design methods such as docking, as shown in a single case by Fischer et al.<sup>15</sup> To partially answer this question, we investigate the task of redocking an existing ligand in its corresponding structure and measure the success as root-mean-square deviation (RMSD) to the experimentally determined binding pose. To analyze a redocking, the most widely used measure for success is the prediction of a pose with an RMSD of lower than a threshold (mostly 2 Å) within the top scored ones. Here, we investigate the top one, two, three, 10, and 32 poses. Using a changing number of top poses allows us to differentiate between cases in which the docking tool is unable to create a correct binding pose and those in which it is able to do so but ranks it worse than poses with high RMSD deviations.

We use all structures present in the sc-PDB that contain AltLocs in the binding site ( $n = 2700$ ) and dock into the baseline model as well as all enumerated conformations using our recently developed docking tool JAMDA.<sup>24</sup> Using improved structures from the pdb-redo might be beneficial, but since most users work with standard PDB files, they were used here as well.<sup>31</sup> The maximum number of conformations was set to 100 to remain in the tractable region with our available compute resources. When enumerating all possible complexes using the AltLocEnumerator, the 2700 binding pockets containing AltLocs are enumerated to 20,545 complexes. The exact number of binding pockets enumerated to a given number of complexes is shown in Table S1. We compare the baseline conformation that was described by the crystallographers to the ones that were added with AltLocs. This poses an additional challenge, as the crystallographers will have chosen the best-suited conformation for the ligand in many cases.

In a first experiment, we evaluate whether taking both the baseline and nonbaseline AltLoc complexes into consideration does yield better poses. This solely shows if better solutions are possible using nonbaseline AltLoc complexes, and it is important to keep in mind that more complexes and therefore more poses are investigated when taking nonbaseline complexes into consideration.

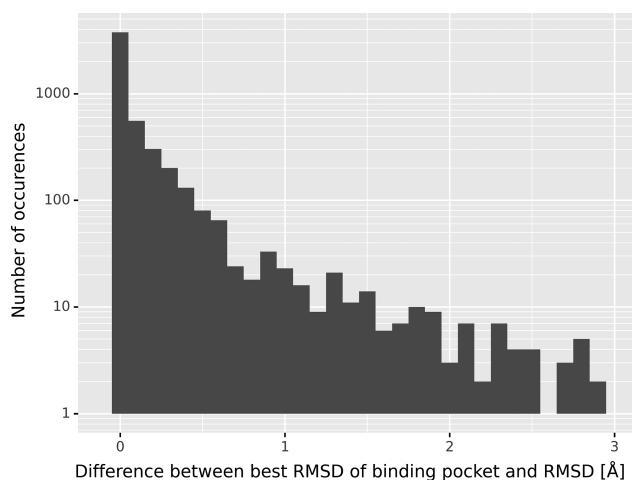
Conducting this experiment, we can see in Table 2 that in over 50% of the cases the RMSD to the reference ligand can be improved using a nonbaseline AltLoc conformation. However,

**Table 2. Results of the Analysis if at Least One Complex Using the AltLoc Ensemble Achieves a Better RMSD in Redocking Using the Top X Poses than the Complex Using the Baseline Conformation<sup>a</sup>**

Top X	Percentage improved	Percentage improvement $\geq 0.5$ Å
1	52.9	15.0
2	53.7	13.4
3	53.7	12.9
10	54.9	10.5
32	53.0	8.6

<sup>a</sup>Results are based on the percentage of the 2700 binding pockets containing AltLocs with improved redocking RMSD. The third column is only counting an improvement once the redocking RMSD improves by more than or equal to 0.5 Å.

this experiment drastically overrates this performance increase because most improvements are minor and irrelevant from the modeler's perspective. To highlight this, the differences in the best-achieved redocking RMSD of the baseline AltLoc conformation and the best-achieved redocking RMSD of nonbaseline conformations are displayed in Figure 8.



**Figure 8.** Histogram of the difference between best RMSD and RMSD of complex with baseline AltLocs for best score for the top 10 and top 32 poses. The  $x$ -axis displays the difference between the best-achieved RMSD of the binding pocket and the RMSD of one conformation and is binned with a binwidth of 0.1. The  $y$ -axis displays the count of observations in the respective bin.

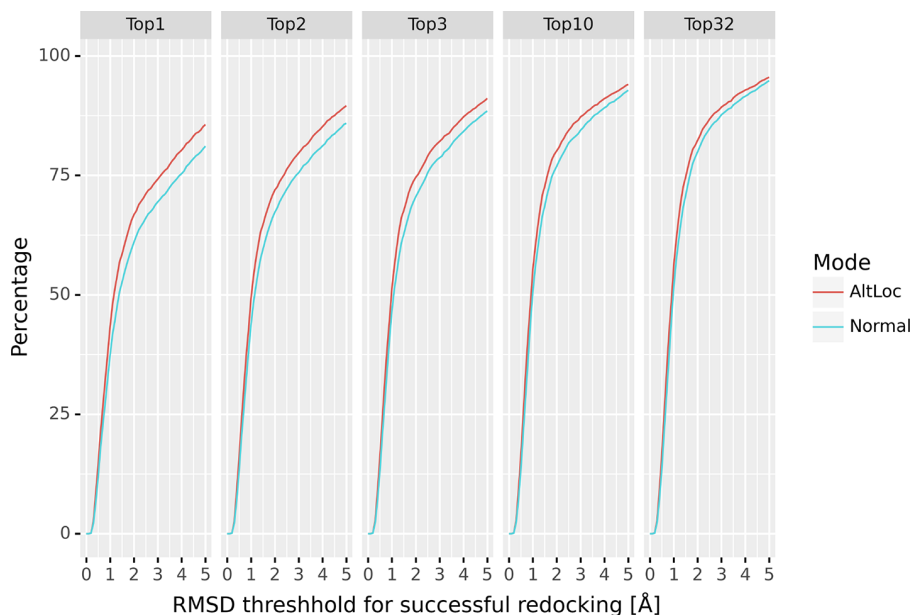
While some complexes show a relevant gain in redocking RMSD of, in exceptional cases up to 8 Å, the differences are tiny for most, introducing the question if minor changes constitute a better pose.

If we define a significant gain in redocking RMSD as a minimum difference of 0.5 Å, between 15% and 8.6% of the docking experiments fit this criterion when considering the AltLoc ensemble (Table 2). In addition, the percentage of binding pockets that can be significantly improved shows a negative trend the more poses we investigate. This behavior could be explained by the fact that once an excellent binding pose is generated in the baseline model it is unlikely that a binding pose in the AltLoc ensemble significantly improves it.

Another estimate of pose quality is for how many binding pockets the improved pose results in a successful redocking that was not successful using only the baseline model. A successful redocking is defined as a prediction in which the lowest RMSD of the top  $x$  poses is lower than a user-given threshold, mostly lower than 2 Å. As the success depends on the chosen RMSD cutoff, and to most accurately describe the data, we decided to display the success percentage in correlation to this cutoff for each top  $x$  value in Figure 9.

There is a visible gain in success percentage if we only investigate the best scored pose for each complex. This gain diminishes the more poses we consider for each enumerated structure, which is an identical effect to the one seen prior in Table 2. In addition, the performance difference is threshold dependent and only starts to become visible at a threshold of about 1–1.5 Å. The effect is diminished compared to the effect seen in Table 2, because only the subset of RMSD improvements are counted that improve the redocking above the success threshold.





**Figure 9.** Percentage of binding pockets with successful redocking on the  $y$ -axis and RMSD threshold used to define a successful redocking on the  $x$ -axis. Using only the baseline AltLoc complex (blue) and using all enumerated AltLoc complexes (red). The number of poses investigated is indicated above the panel.

While these different analyses showcase that using AltLocs can yield better poses, this advantage is only relevant if ranking them correctly is possible. Because when using AltLocs we investigate both the baseline and all nonbaseline complexes, the number of complexes and therefore poses considering the AltLoc ensemble always outnumbers the one using just the baseline conformation. This fact causes a numeric advantage for the AltLoc mode. To solve this advantage, we tried multiple methods to evaluate an identical number of poses for both approaches, namely, using the best scored poses in all complexes, all poses of the complex with the best scored top pose, and the top poses of docking to the ensemble of all receptors. However, for all these approaches, we could not yield a significantly better (or worse) performance than just using the baseline model (see Figures S2–S4).

A detailed discussion of these different approaches can be found in the Supporting Information. We came to the conclusion that harvesting the entire value of considering AltLocs during docking is only possible with scoring functions taking protein flexibility explicitly into account, for example, by scoring the energetic differences between alternative side chain contacts.

A part of the problem might be that for the binding pockets containing AltLocs many conformations described do not interact directly with the ligand or only describe minor differences. These conformations will, in many cases, not alter the redocking experiment (even though they may alter other structure-based experiments). Nevertheless, because we found cases in which the consideration of AltLocs significantly improved the outcome of the experiment, any well-prepared redocking should consider AltLocs. This will be most likely even more true for cross-docking experiments in which alternative binding site conformations are critical for correct predictions for ligands varying in shape and size.

**Specific AltLocs can be Crucial for Structure-Based Tasks.** To explain why and how AltLocs can influence

structural tasks like redocking, we highlight some cases in which AltLocs were critical to the success of a redocking.

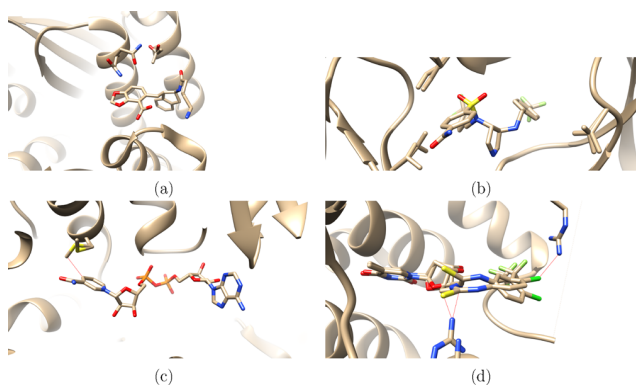
For the HIV integrase complex (PDB 4CJ3, Figure 10a), there is a possible hydrogen bond in the baseline AltLoc complex with a threonine that is not possible in the nonbaseline complex. Counterintuitively, because there is no hydrogen bond between this threonine in the PDB, using the baseline complex does yield worse results. The worse results are probably because many poses form a hydrogen bond with the threonine, leading to a suboptimal binding pose.

In the HIV protease complex (PDB 3QRM), the baseline AltLocs of multiple residues perform worse than their nonbaseline counterparts (Figure 10b). In this case, the reason is not obvious because there are no direct clashes or interactions formed or broken between the baseline and nonbaseline AltLoc complex, but the effect is still present. This highlights that in some cases the reason for a structure being better suited can be difficult to find.

In the human malate dehydrogenase complex (PDB 2DFD, Figure 10c), the baseline AltLoc complex clashes with the ligand. Therefore, the correct binding mode cannot be reproduced in the baseline complex. While this is probably an oversight by the crystallographers, errors like this happen and are not always corrected when publishing structures.

Another example that is probably due to an error by the submitting crystallographers is the *Plasmodium falciparum* thymidylate kinase complex (PDB 2YOF, Figure 10d). In this example, both ligand and residues have AltLocs, but the same identifier versions of the ligand and residue AltLocs clash. Due to this (and the RMSD calculation focused on the first appearing AltLoc of the ligand), the nonbaseline AltLoc complex significantly outperforms the baseline complex.

As these single cases show, there are many reasons for why a nonbaseline AltLoc complex can be better suited for the given task than its baseline counterpart. Reasons for this include but are not limited to human error, to interactions that are only present (or absent) in one specific conformation described by

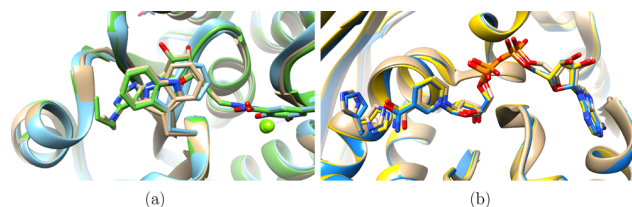


**Figure 10.** Crystal structures of examples of nonbaseline AltLoc complexes with significantly improved redocking RMSD compared to baseline complexes. All examples are taken from the top 10/top 32 poses to use cases in which JAMDA does not achieve the correct binding pose in the baseline AltLoc. All examples allow for a successful redocking in a nonbaseline AltLoc complex that was unsuccessful in the baseline one. The clash criteria used are the baseline criteria implemented in Chimera. (a) AltLocs in the binding pocket of PDB 4CJ3. The baseline AltLoc of the threonine allows forming a hydrogen bond that is not present in the complex. In addition, the nonbaseline AltLoc in glutamine improves the redocking to a lesser extent. In redocking, the nonbaseline complex outperforms the baseline one by  $\sim 2.7$  Å using the top 32 poses, making the redocking successful. (b) AltLocs in the binding pocket of PDB 3QRM. The baseline AltLoc of residues 50–51 and 82 in chain A and 84 in chain B perform worse than the nonbaseline counterparts. In redocking, the nonbaseline complex outperforms the baseline one by  $\sim 5.4$  Å using the top 10 poses (1.14 Å in top 32), making the redocking successful. (c) AltLocs in the binding pocket of PDB 2DFD. The baseline AltLoc clashes with the ligand. In redocking, the nonbaseline complex outperforms the baseline one by  $\sim 8.0$  Å using the top 10 poses, making the redocking successful. (d) AltLocs in the binding pocket of PDB 2YOF. The baseline AltLoc clashes with the ligand. In redocking, the nonbaseline complex outperforms the baseline one by  $\sim 7.2$  Å using the top 32 poses, making the redocking successful.

AltLocs, or by the binding pocket just being better suited for the task at hand for nonobvious reasons.

**AltLocs can Describe Protein Flexibility Present in Multistructure Ensembles.** Another reason to consider AltLocs is that they describe protein flexibility that would otherwise only be described using an ensemble of multiple PDB structures or chains. This makes AltLocs very valuable information especially for proteins for which only a few structures exist in the PDB. Using SIENA<sup>32</sup> to construct an ensemble of PDB structures with very similar binding pockets, we can find protein conformations in other PDBs that are described by AltLocs in an original PDB. As an example for structures of Catechol O-methyltransferase (PDB 3HVI) and WlbA (PDB 3O9Z), we found the side chain conformations described by their AltLocs in other structures/chains (see Figure 11).

This highlights that the protein flexibility described by AltLoc conformations is consistent with structure ensembles of the same or a similar protein in the PDB. AltLocs have the advantage that they must not be retrieved with a database search since they are contained in the same file and in addition are curated by the structure's author. When ignoring AltLocs, experimentally validated conformations might be missed that are not available otherwise.



**Figure 11.** Showcase of AltLocs that describe side chain conformations that otherwise would only be possible to model when considering multiple PDBs/chains. The overlays of the binding pockets were created using SIENA.<sup>32</sup> (a) Overlay of PDBs 3HVI (tan), 1H1D (blue), and 3HVJ (green). The AltLoc in 3HVI does describe the side chain conformations of TRP186 of both 1H1D and 3HVJ. (b) Overlay of PDBs 3O9Z (tan), 3OAO chain A (blue), and 3OAO chain B (yellow). The AltLoc in 3O9Z does describe the side chain conformations of HIS185 of both chains A and B in 3OAO.

## CONCLUSION

In this publication, we report the development of a novel algorithm for the automatic treatment of AltLocs. A software tool using this algorithm allows users to handle AltLocs in PDB structures by creating single, consistent structures or structure ensembles.

AltLocEnumerator was developed to fit into existing workflows, only building valid structures that confer the quality criteria of typical modeling environments like NAOMI, allowing users to focus on any part of the structure.

First, we thoroughly investigated how AltLocs occur in the PDB and how they are used by crystallographers. This analysis revealed general statistical trends on AltLocs in the complete PDB and that AltLocs became more popular over time, with around 60% of all structures released in the last 10 years containing AltLocs, emphasizing the importance of accounting for them.

Our validation showed that AltLocEnumerator's algorithm creates reasonable structures from AltLocs combinations. In addition, we measured the required computation times on the entire sc-PDB subset and highlighted some cases in which AltLocEnumerator produced unexpected results.

For the example of redocking, we have shown that different structural conformations described by AltLocs improve pose quality. The ranking of docking poses with varying AltLocs was challenging with a standard scoring function not considering intraprotein interactions. This analysis also emphasized that the problem of automatically describing protein flexibility for docking remains challenging and requires further research. Respective studies for cross-docking and virtual screening experiments are the next logical steps to fully comprehend how AltLocs impact docking. However, due to computational constraints, they are not part of this publication.

The examples in which AltLocs significantly impact redocking highlight why it is essential to investigate AltLocs before employing structures in modeling work. Our analysis of AltLocs and their conformations in other PDB structures using SIENA<sup>32</sup> also highlights that AltLocs can be applied to create decent conformational flexibility models from a single structure.

To this day AltLocs are mostly ignored entirely by choosing the first or highest occupancy option for the complete structure. Handling AltLocs can prevent using faulty combinations of structural elements and provides a way to standardize and automate receptor preparation. With the introduction of AltLocEnumerator, we hope to offer a tool for

any structure-based task that allows users to integrate an informed decisions on which structural conformations to use.

## ■ ASSOCIATED CONTENT

### Data Availability Statement

AltLocEnumerator is available for Linux, Mac, and Windows as part of the NAOMI ChemBio Suite at <https://uhh.de/naomi> and is free for academic use and evaluation purposes. The data on AltLoc occurrence as well as the resulting scores of the docking evaluation are available in the Supporting Information. The sc-PDB is available at <http://bioinfo-pharma.u-strasbg.fr/scPDB/>.<sup>27</sup>

### SI Supporting Information

The Supporting Information is available free of charge at <https://pubs.acs.org/doi/10.1021/acs.jcim.3c00100>.

Supporting data (ZIP)

Discussion about reranking of docking results, additional statistics on AltLocs, additional statistics on docking calculations and performance, additional statistics about docking performance and RMSD of AltLocs used, table of the number of enumerated structures from the original structures in the sc-PDB (PDF)

## ■ AUTHOR INFORMATION

### Corresponding Author

Matthias Rarey – Universität Hamburg, ZBH - Center for Bioinformatics, 20146 Hamburg, Germany; [orcid.org/0000-0002-9553-6531](https://orcid.org/0000-0002-9553-6531); Email: [matthias.rarey@uni-hamburg.de](mailto:matthias.rarey@uni-hamburg.de)

### Authors

Torben Gutermuth – Universität Hamburg, ZBH - Center for Bioinformatics, 20146 Hamburg, Germany; [orcid.org/0000-0002-9304-8251](https://orcid.org/0000-0002-9304-8251)

Jochen Sieg – Universität Hamburg, ZBH - Center for Bioinformatics, 20146 Hamburg, Germany; [orcid.org/0000-0001-5343-7255](https://orcid.org/0000-0001-5343-7255)

Tim Stohn – Universität Hamburg, ZBH - Center for Bioinformatics, 20146 Hamburg, Germany; Present Address: Tim Stohn: Computer Science Department, Center for Integrative Bioinformatics (IBIVU), Vrije Universiteit Amsterdam, De Boelelaan 1111, Amsterdam 1081 HV, The Netherlands; Division of Molecular Carcinogenesis, The Oncode Institute, The Netherlands Cancer Institute, Plesmanlaan 121, Amsterdam 1066 CX, The Netherlands

Complete contact information is available at: <https://pubs.acs.org/doi/10.1021/acs.jcim.3c00100>

### Author Contributions

<sup>#</sup>Torben Gutermuth and Jochen Sieg contributed equally.

### Notes

The authors declare no competing financial interest.

## ■ ACKNOWLEDGMENTS

This work was supported by the German Federal Ministry of Education and Research as part of de.NBI (031L0105) and protP.S.I. (031B0405B).

## ■ REFERENCES

- (1) Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shindyalov, I. N.; Bourne, P. E. The Protein Data Bank. *Nucleic Acids Res.* **2000**, *28*, 235–242.
- (2) Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Zidek, A.; Potapenko, A.; Bridgland, A.; Meyer, C.; Kohl, S. A. A.; Ballard, A. J.; Cowie, A.; Romera-Paredes, B.; Nikolov, S.; Jain, R.; Adler, J.; Back, T.; Petersen, S.; Reiman, D.; Clancy, E.; Zielinski, M.; Steinegger, M.; Pacholska, M.; Berghammer, T.; Bodenstein, S.; Silver, D.; Vinyals, O.; Senior, A. W.; Kavukcuoglu, K.; Kohli, P.; Hassabis, D. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583–589.
- (3) Saldano, T.; Escobedo, N.; Marchetti, J.; Zea, D. J.; Mac Donagh, J.; Velez Rueda, A. J.; Gonik, E.; Garcia Melani, A.; Novomisky Nechcoff, J.; Salas, M. N.; Peters, T.; Demitroff, N.; Fernandez Alberti, S.; Palopoli, N.; Fornasari, M. S.; Parisi, G. Impact of protein conformational diversity on AlphaFold predictions. *Bioinformatics* **2022**, *38*, 2742–2748.
- (4) Alvarez-Garcia, D.; Barril, X. Relationship between Protein Flexibility and Binding: Lessons for Structure-Based Drug Design. *J. Chem. Theory Comput.* **2014**, *10*, 2608–2614.
- (5) Wong, C. F. Flexible receptor docking for drug discovery. *Expert Opin. Drug Discovery* **2015**, *10*, 1189–1200.
- (6) Cozzini, P.; Kellogg, G. E.; Spyarakis, F.; Abraham, D. J.; Costantino, G.; Emerson, A.; Fanelli, F.; Gohlke, H.; Kuhn, L. A.; Morris, G. M.; Orozco, M.; Pertinhez, T. A.; Rizzi, M.; Sotriffer, C. A. Target Flexibility: An Emerging Consideration in Drug Discovery and Design. *J. Med. Chem.* **2008**, *51*, 6237–6255.
- (7) Antunes, D. A.; Devaurs, D.; Kavraki, L. E. Understanding the challenges of protein flexibility in drug design. *Expert Opin. Drug Discovery* **2015**, *10*, 1301–1313.
- (8) Stachowski, T. R.; Fischer, M. Large-Scale Ligand Perturbations of the Protein Conformational Landscape Reveal State-Specific Interaction Hotspots. *J. Med. Chem.* **2022**, *65*, 13692.
- (9) Kamenik, A. S.; Singh, I.; Lak, P.; Balias, T. E.; Liedl, K. R.; Shoichet, B. K. Energy penalties enhance flexible receptor docking in a model cavity. *Proc. Natl. Acad. Sci. U. S. A.* **2021**, *118*, e2106195118.
- (10) Bottegoni, G.; Rocchia, W.; Rueda, M.; Abagyan, R.; Cavalli, A. Systematic Exploitation of Multiple Receptor Conformations for Virtual Ligand Screening. *PLoS One* **2011**, *6*, No. e18845.
- (11) Barril, X.; Morley, S. D. Unveiling the Full Potential of Flexible Receptor Docking Using Multiple Crystallographic Structures. *J. Med. Chem.* **2005**, *48*, 4432–4443.
- (12) Keedy, D. A.; Fraser, J. S.; van den Bedem, H. Exposing Hidden Alternative Backbone Conformations in X-ray Crystallography Using qFit. *PLoS Comput. Biol.* **2015**, *11*, No. e1004507.
- (13) Lang, P. T.; Ng, H.-L.; Fraser, J. S.; Corn, J. E.; Echols, N.; Sales, M.; Holton, J. M.; Alber, T. Automated electron-density sampling reveals widespread conformational polymorphism in proteins. *Protein Sci.* **2010**, *19*, 1420–1431.
- (14) Wankowicz, S. A.; de Oliveira, S. H.; Hogan, D. W.; van den Bedem, H.; Fraser, J. S. Ligand binding remodels protein side chain conformational heterogeneity. *eLife* **2022**, *11*, No. e74114.
- (15) Fischer, M.; Coleman, R. G.; Fraser, J. S.; Shoichet, B. K. Incorporation of protein flexibility and conformational energy penalties in docking screens to improve ligand discovery. *Nat. Chem.* **2014**, *6*, 575–583.
- (16) Qi, H. W.; Kulik, H. J. Evaluating unexpectedly short non-covalent distances in x-ray crystal structures of proteins with electronic structure analysis. *J. Chem. Inf. Model.* **2019**, *59*, 2199–2211.
- (17) Pettersen, E. F.; Goddard, T. D.; Huang, C. C.; Meng, E. C.; Couch, G. S.; Croll, T. I.; Morris, J. H.; Ferrin, T. E. UCSF ChimeraX: Structure visualization for researchers, educators, and developers. *Protein Sci.* **2021**, *30*, 70–82.
- (18) Goddard, T. D.; Huang, C. C.; Meng, E. C.; Pettersen, E. F.; Couch, G. S.; Morris, J. H.; Ferrin, T. E. UCSF ChimeraX: Meeting



modern challenges in visualization and analysis. *Protein Sci.* **2018**, *27*, 14–25.

(19) The PyMOL Molecular Graphics System, Version 1.8.; Schrödinger, LLC, 2015.

(20) Cock, P. J. A.; Antao, T.; Chang, J. T.; Chapman, B. A.; Cox, C. J.; Dalke, A.; Friedberg, I.; Hamelryck, T.; Kauff, F.; Wilczynski, B.; de Hoon, M. J. L. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* **2009**, *25*, 1422–1423.

(21) Spruce Toolkit, version 2022.2. *Open Eye Scientific Software*. <https://docs.eyesopen.com/toolkits/python/sprucetk/OESpruceConstants/OEAlternateLocationOption.html> (accessed March 8, 2023).

(22) Berman, H.; Henrick, K.; Nakamura, H. Announcing the worldwide Protein Data Bank. *Nat. Struct. Mol. Biol.* **2003**, *10*, 980–980.

(23) Miao, Z.; Cao, Y. Quantifying side chain conformational variations in protein structure. *Sci. Rep.* **2016**, *6*, 37024.

(24) Flachsenberg, F.; Meyder, A.; Sommer, K.; Penner, P.; Rarey, M. A Consistent Scheme for Gradient-Based Optimization of Protein–Ligand Poses. *J. Chem. Inf. Model.* **2020**, *60*, 6502–6522.

(25) Urbaczek, S.; Kolodzik, A.; Fischer, J. R.; Lippert, T.; Heuser, S.; Groth, I.; Schulz-Gasch, T.; Rarey, M. NAOMI: On the Almost Trivial Task of Reading Molecules from Different File formats. *J. Chem. Inf. Model.* **2011**, *51*, 3199–3207.

(26) Urbaczek, S.; Kolodzik, A.; Groth, I.; Heuser, S.; Rarey, M. Reading PDB: Perception of Molecules from 3D Atomic Coordinates. *J. Chem. Inf. Model.* **2013**, *53*, 76–87.

(27) Kellenberger, E.; Müller, P.; Schalon, C.; Bret, G.; Foata, N.; Rognan, D. sc-PDB: an Annotated Database of Druggable Binding Sites from the Protein Data Bank. *J. Chem. Inf. Model.* **2006**, *46*, 717–727.

(28) Davis, A. M.; Teague, S. J.; Kleywegt, G. J. Application and limitations of X-ray crystallographic data in structure-based ligand and drug design. *Angew. Chem., Int. Ed.* **2003**, *42*, 2718–2736.

(29) Pearce, N. M.; Krojer, T.; Bradley, A. R.; Collins, P.; Nowak, R. P.; Talon, R.; Marsden, B. D.; Kelm, S.; Shi, J.; Deane, C. M.; von Delft, F. A multi-crystal method for extracting obscured crystallographic states from conventionally uninterpretable electron density. *Nat. Commun.* **2017**, *8*, 15123.

(30) Pearce, N. M.; Krojer, T.; von Delft, F. Proper modelling of ligand binding requires an ensemble of bound and unbound states. *Acta Crystallogr., Sect. D: Struct. Biol.* **2017**, *73*, 256–266.

(31) van Beusekom, B.; Touw, W. G.; Tatineni, M.; Somani, S.; Rajagopal, G.; Luo, J.; Gilliland, G. L.; Perrakis, A.; Joosten, R. P. Homology-based hydrogen bond information improves crystallographic structures in the PDB. *Protein Sci.* **2018**, *27*, 798–808.

(32) Bietz, S.; Rarey, M. SIENA: efficient compilation of selective protein binding site ensembles. *J. Chem. Inf. Model.* **2016**, *56*, 248–259.

## Recommended by ACS

### Solvent Accessibility Promotes Rotamer Errors during Protein Modeling with Major Side-Chain Prediction Programs

Tareq Hameduh, Yazan Haddad, *et al.*

JULY 06, 2023

JOURNAL OF CHEMICAL INFORMATION AND MODELING

READ 

### Hydrophobicity—A Single Parameter for the Accurate Prediction of Disordered Regions in Proteins

Nitin Kumar Singh, Mithun Radhakrishna, *et al.*

AUGUST 15, 2023

JOURNAL OF CHEMICAL INFORMATION AND MODELING

READ 

### AlphaFold2-RAVE: From Sequence to Boltzmann Ranking

Bodhi P. Vani, Pratyush Tiwary, *et al.*

MAY 12, 2023

JOURNAL OF CHEMICAL THEORY AND COMPUTATION

READ 

### Best Practices of Using AI-Based Models in Crystallography and Their Impact in Structural Biology

Marc Graille, Antoine Taly, *et al.*

JUNE 12, 2023

JOURNAL OF CHEMICAL INFORMATION AND MODELING

READ 

Get More Suggestions >

# Supporting Information:

## Modeling with Alternate Locations in X-ray Protein Structures

Torben Gutermuth,<sup>†,‡</sup> Jochen Sieg,<sup>†,‡</sup> Tim Stohn,<sup>†,¶</sup> and Matthias Rarey<sup>\*,†</sup>

<sup>†</sup> *Universität Hamburg, ZBH - Center for Bioinformatics, Bundesstraße 43, 20146  
Hamburg, Germany*

<sup>‡</sup> *These authors contributed equally*

<sup>¶</sup> *Currently working: Computer Science Department, Center for Integrative Bioinformatics  
(IBIVU), Vrije Universiteit Amsterdam, De Boelelaan 1111, Amsterdam 1081 HV, The  
Netherlands; Division of Molecular Carcinogenesis, The Oncode Institute, The Netherlands  
Cancer Institute, Plesmanlaan 121, Amsterdam 1066 CX, The Netherlands*

E-mail: matthias.rarey@uni-hamburg.de

### Supporting discussion

As elaborated in the main text, one problem persisting with the analysis is that all approaches considering AltLocs evaluate more poses than the approach just considering the standard complex. Due to this difference, there is a numeric advantage for the approaches that utilise all enumerated complexes. To fix this advantage, we tried multiple rescoring schemes so that the approach that utilises non-standard AltLoc complexes evaluates an equal number of poses as its standard counterpart. One approach is to use the top scored  $x$  poses when merging the results of all complexes compared to the standard complex. The results of this approach

can be seen in Figure S2. Conducting this rescoring scheme, the standard approach shows identical performance or even outperforms the AltLoc approach. This behaviour might be because through merging results, very similar poses are evaluated using the AltLoc approach. However, similar poses are clustered in the standard approach, and another different pose is evaluated instead. Another approach to alleviating this problem is to use only the docking to a single structure. This could be done by always choosing the structure with the best score of the top scored pose, which might indicate that this structure is best suited for redocking the ligand. As shown in Figure S3, this approach yields an almost identical performance to just using the traditional complex. Another option is to use an ensemble approach and eliminate poses that are too similar to each other while using all structures. However, as in the previous examples, this approach performs identical to the traditional method, as seen in Figure S4. As we already knew that the performance difference is heavily dependent on the type of AltLoc and its interactions with the ligand, another hypothesis was that the AltLocs we investigate might constitute primarily minute changes to the protein, which do not alter the docking significantly. However, as seen in Figure S5, when investigating only structural conformations that correspond to bigger changes, the exhibited effect diminishes, contrary to our stated hypothesis. Mostly the AltLocs that constitute minor changes to the structure create significant differences in the RMSD of the redocking. With that analysis and the effects already shown in the main text, we can conclude that considering AltLocs for redocking does improve the results. We are unable to yield these results in an automated way. This is because while there are structural conformations described by AltLocs that are beneficial for the redocking, we cannot discriminate between them and those that are not beneficial automatically.

## Supporting Figures and Tables

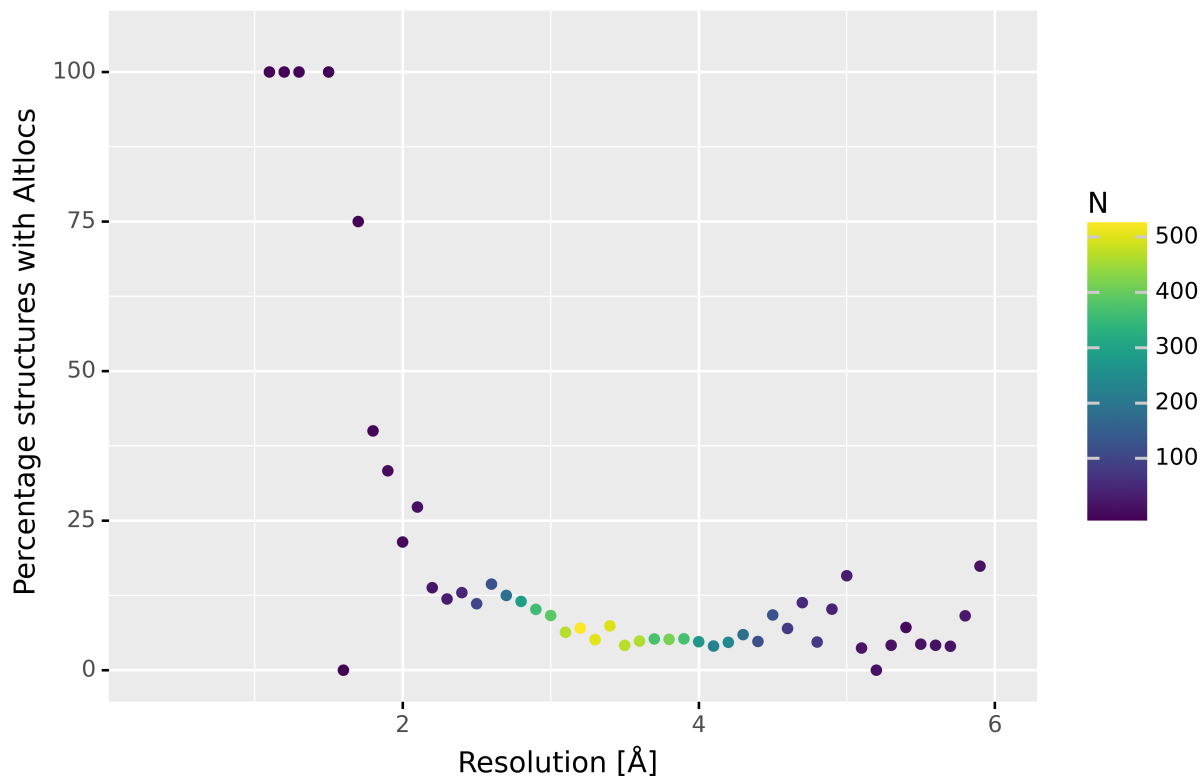


Figure S1: Percentage of structures resolved with electron microscopy containing at least one AltLoc on the y-axis and the resolution of the structures on the x-axis. The number of structures present at each point is colour-coded.

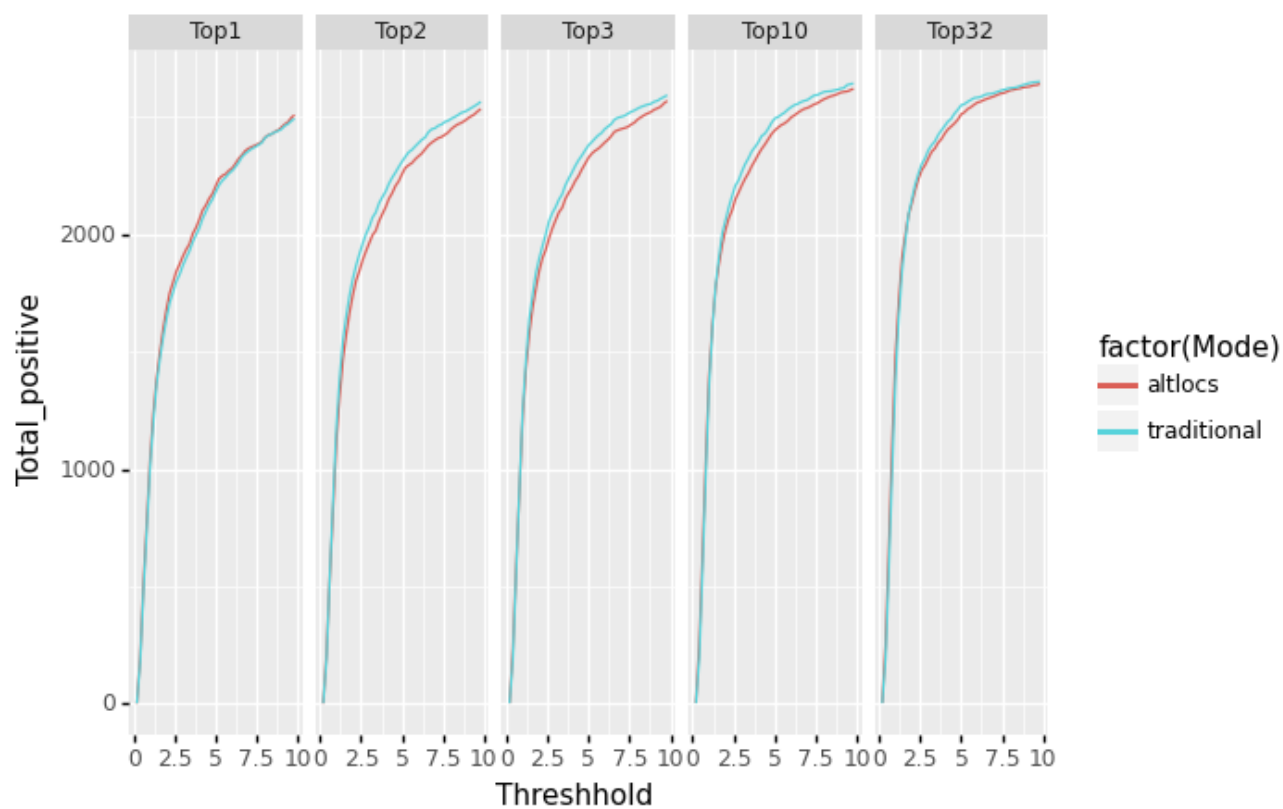


Figure S2: Redocking performance for standard AltLoc complexes compared the Top x best-scored poses of all enumerated complexes with success thresholds as x variable for multiple Top x poses.



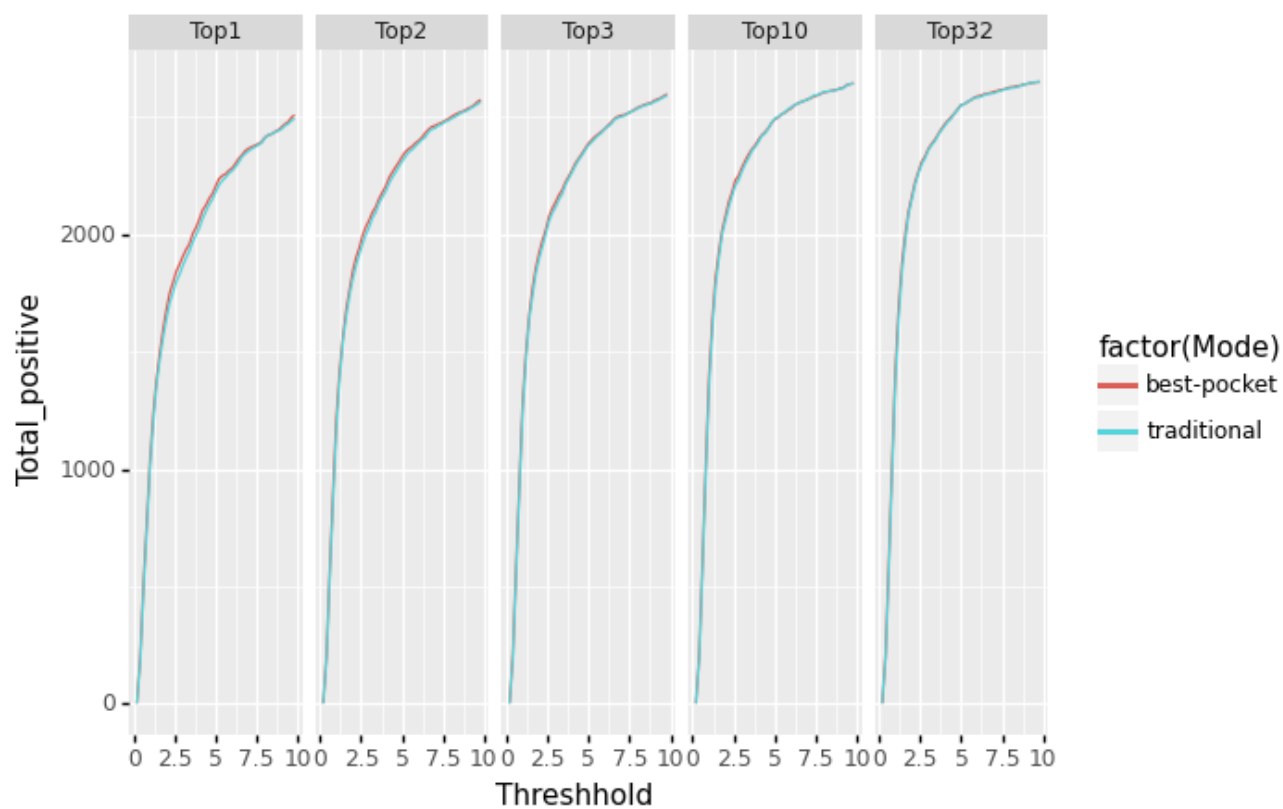


Figure S3: Redocking performance for standard AltLoc complexes compared the Top x of the structure with the best scored top pose of all enumerated complexes with success thresholds as x variable for multiple Top x poses.

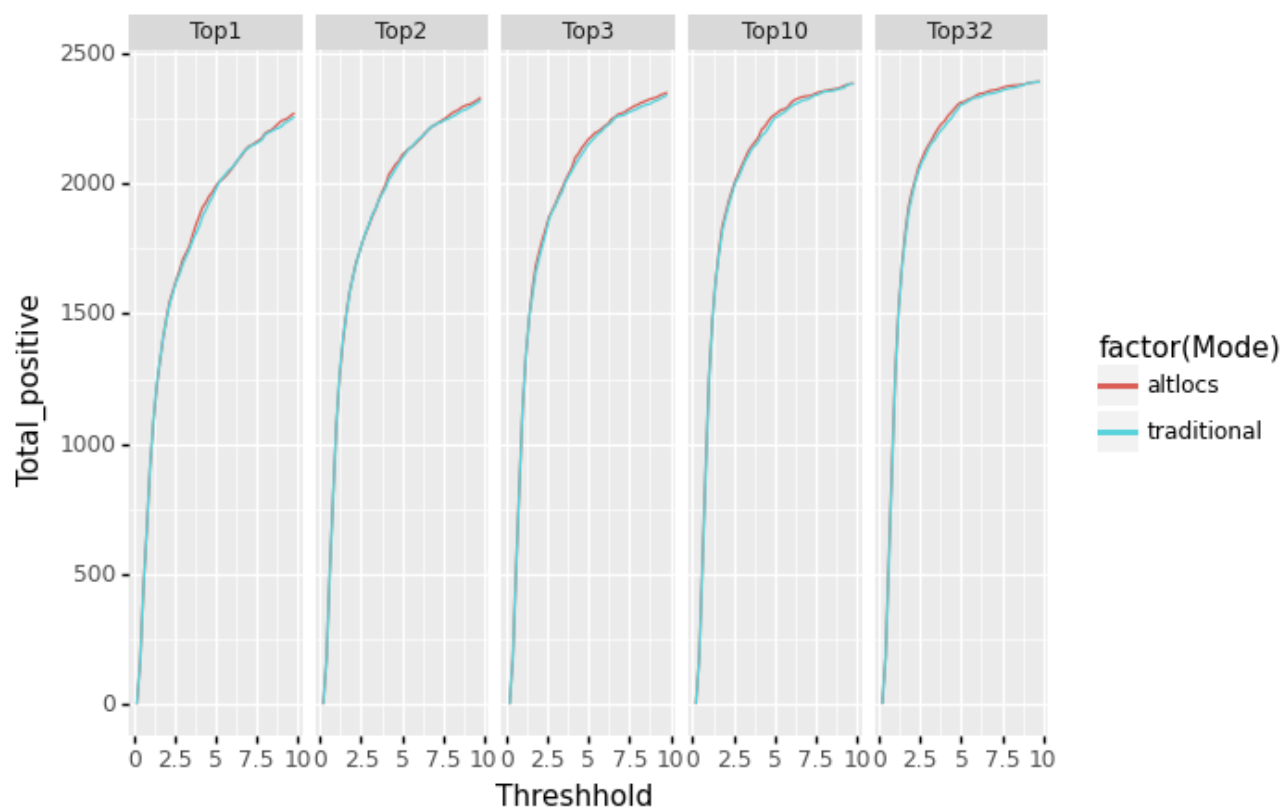


Figure S4: Redocking performance for standard AltLoc complexes compared the Top x of the ensemble of all structures with success thresholds as x variable for multiple Top x poses.

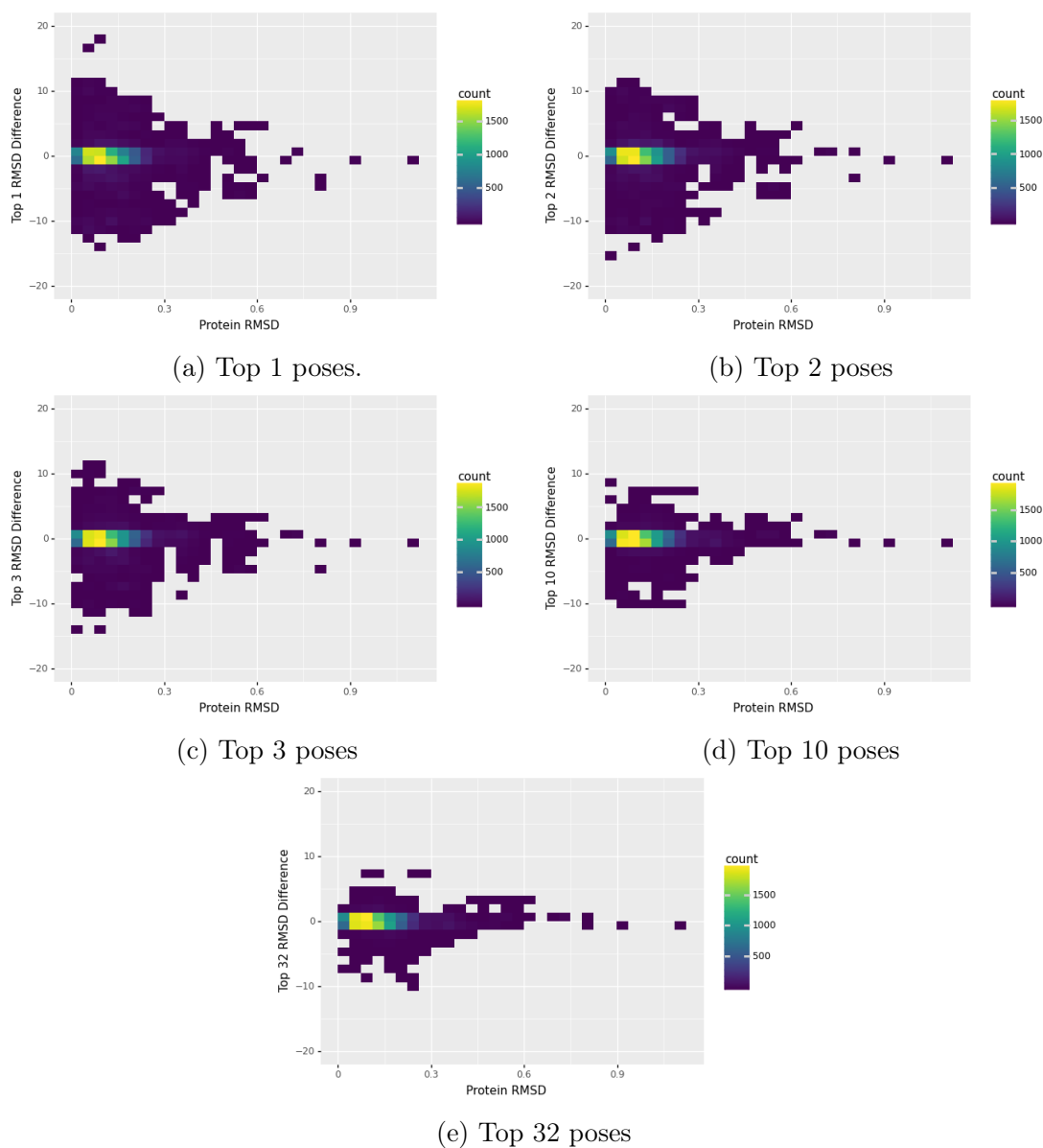


Figure S5: Two-dimensional histogram of protein RMSD difference between traditional Al-tLoc and one enumerated version on the x-axis with the respective difference in the best-achieved redocking RMSD in the top 32 poses on the y-axis. The number of observations found in each voxel is colour-coded.

Table S1: Statistic of the number of complexes generated for the 2700 binding pockets in the sc-PDB containing AltLocs. The first column shows the number of complexes generated using the AltLocEnumerator, and the second column the number of binding pockets used as input to generate this number of complexes. The maximum number of complexes generated is 100.



Number of complexes enumerated using AltLocEnumerator	Number of binding pockets
1	6
2	1555
3	48
4	565
5	1
6	31
8	207
9	2
10	4
12	19
16	93
20	2
24	15
28	1
32	37
36	1
48	16
64	16
72	3
80	1
88	1
96	5
100	71

## **D.4 Searching similar local 3D micro-environments in protein structure databases with MicroMiner**

- [D4] **J. Sieg** and M. Rarey. “Searching similar local 3D micro-environments in protein structure databases with MicroMiner”. In: *Briefings in Bioinformatics* 24.6 (2023), bbad357.

Available: <https://doi.org/10.1093/bib/bbad357>. Material from [D4].

# Searching similar local 3D micro-environments in protein structure databases with MicroMiner

Jochen Sieg  and Matthias Rarey 

Corresponding author. Matthias Rarey, Universität Hamburg, ZBH - Center for Bioinformatics, Bundesstraße 43, 20146 Hamburg, Germany. Tel.: +49 40 42838-7351; Fax: +49 40 42838-7352; E-mail: matthias.rarey@uni-hamburg.de

## Abstract

The available protein structure data are rapidly increasing. Within these structures, numerous local structural sites depict the details characterizing structure and function. However, searching and analyzing these sites extensively and at scale poses a challenge. We present a new method to search local sites in protein structure databases using residue-defined local 3D micro-environments. We implemented the method in a new tool called MicroMiner and demonstrate the capabilities of residue micro-environment search on the example of structural mutation analysis. Usually, experimental structures for both the wild-type and the mutant are unavailable for comparison. With MicroMiner, we extracted  $> 255 \times 10^6$  amino acid pairs in protein structures from the PDB, exemplifying single mutations' local structural changes for single chains and  $> 45 \times 10^6$  pairs for protein–protein interfaces. We further annotate existing data sets of experimentally measured mutation effects, like  $\Delta\Delta G$  measurements, with the extracted structure pairs to combine the mutation effect measurement with the structural change upon mutation. In addition, we show how MicroMiner can bridge the gap between mutation analysis and structure-based drug design tools. MicroMiner is available as a command line tool and interactively on the <https://proteins.plus/> webservice.

**Keywords:** mutation modeling; mutation effect prediction; protein structure; protein site; micro-environment

## INTRODUCTION

The quality and quantity of protein structures have increased tremendously in recent years. Even large multimeric structures can be resolved at atomic resolution with cryo-EM [1–3]. More recently, breakthroughs in structure prediction [4, 5] provide an unprecedented manifold of predicted high-quality protein structure data [6]. Therefore, many new possibilities for large-scale structural bioinformatics arose, and an avalanche of data is available for detailed exploration.

The standard methods for searching protein structure databases focus on homology detection and can be grouped into sequence [7–11] and structure-based [12–14] tools. They are central for annotating functional, structural and sequential features from experimentally characterized to unknown proteins based on evolutionary relationships. Usually, the similarity is assessed by identifying the largest similar part shared between two proteins or domains. Although these methods are foundational for bioinformatics, their focus is not ideal for searching for similarities between local 3D sites in protein structures.

Local 3D sites of the protein structure are of particular interest in many applications. Their local chemical and structural environment, often characterized by sequentially distant 3D contacts, can be crucial for the structure–function relationship since they facilitate binding, catalysis, structural support or dynamics and other functions [15–22]. For example, binding sites provide a local protein micro-environment that facilitates binding ligands, proteins or nucleic acids. Knowledge about the similarity of binding sites can be used to deal with selectivity issues and off-target effects in drug design, help predict unknown binding sites and

uncharacterized protein functions [23]. However, while methods for calculating the similarity of binding sites, especially ligand binding sites, are well established [23], methods for comparing and searching other similar sites in protein structures are less widespread.

In this work, we developed a method that considers the similarity of local amino acid residue sites in the protein structure. We use the term residue 3D micro-environment to describe the neighboring protein residues of a particular residue in 3D space [16]. This local neighborhood represents the local structural and chemical environment of the residue [16, 17]. In the past, residue micro-environments have been an important resource for mutation effect prediction [24–26], protein design [27] and functional site prediction [16, 17, 28].

The method, implemented in the tool MicroMiner, automatically extracts the surrounding residues of a query residue making up the query micro-environment and uses them to find similar local environments in other protein structures. A focus on the 3D site allows the identification of local similarities. Global similarities or differences are not necessary to detect them.

With MicroMiner local sites with identical sequences but structural deviations can be searched to explore local protein conformations and flexibility. Besides that, local structural changes due to mutations can be explored by searching for similar environments with deviating sequence. In addition, multimeric protein complexes can be considered, which enables the search for local environments in protein–protein interfaces (PPIs). The method and the implementation of MicroMiner are fast enough to search for similar micro-environments of all residues in an input protein

Jochen Sieg is a PhD candidate at the Universität Hamburg working in structural bioinformatics and cheminformatics.

Matthias Rarey is a full professor at Universität Hamburg. His research interests focus on novel computational methods for cheminformatics, structure-based molecular design, structural bioinformatics and visualizing molecular data.

Received: June 9, 2023. Revised: August 28, 2023. Accepted: September 18, 2023

© The Author(s) 2023. Published by Oxford University Press. All rights reserved. For Permissions, please email: journals.permissions@oup.com

structure against the entire PDB [29] with almost 200 000 structures in just a few seconds.

An important difference between the perspective of residue 3D micro-environments and classical sequence-based homology detection is the detailed residue-wise assessment of local 3D protein sites considering sequence, 3D contacts and multiple chains, instead of the sequence similarity on the level of two protein chains or domains. Even though standard sequence homology detection methods can already find many protein sequence pairs sharing similar 3D micro-environments, it is not directly possible to extract the 3D micro-environments and estimate their similarity simply from these tools' output. We believe that a dedicated method to query similar 3D micro-environments will be important for many applications and a valuable complement to classical homology detection tools.

As a key application, we demonstrate how our method can be used for single mutation search. Accurate structures for the wild-type and the mutant are crucial for mutation modeling and prediction in various applications. Despite breakthroughs in structure prediction, methods like AlphaFold2 are insufficient to predict mutation effects [30, 31]. Various community efforts are dedicated to connecting protein structures with mutation data [32, 33]. However, data set biases and limited data are well-known obstacles [34–39]. A major limitation for the analysis of mutation data is the lack of structures for both the wild-type and the mutant. Usually, only the wild-type structure is available. Thus a precise representation of the mutation-induced structural changes is missing [36, 37], even though a reliable structure is vital for method development and molecular modeling in drug discovery and protein engineering.

Single mutations can be represented as similar 3D micro-environments with identical sequences except for the position of the reference residue, where there should be a mutation. MicroMiner can exploit the redundancy in the PDB to provide a wealth of experimental wild-type/mutant structure pairs elucidating local structural changes of single mutations. The PDB contains roughly 200 000 experimental structures but represents only around 100 000 unique protein structures [5], meaning many variants of the same or similar proteins exist. Furthermore, we can use the detected structure pairs to annotate existing mutation data sets of experimental thermodynamic measurements, like  $\Delta\Delta G$  stability changes, protein–protein and protein–ligand affinity changes upon mutation with experimental structures for both the wild-type and the mutant. Finally, for the use case of the tumor suppressor p53, we illustrate how MicroMiner can bridge the gap between mutation analysis and structure-based drug discovery.

## METHODS

### Overview

We developed a new method for searching similar local 3D sites in protein structure databases. The workflow to search with a single query residue 3D micro-environment is illustrated in Figure 1. The query micro-environment is extracted from the query structure (Figure 1A) and used to find similar local environments in a protein structure database (Figure 1B–C). The workflow output is an ensemble of the hit structures superposed to the query micro-environment (Figure 1D) and multiple similarity measures.

Our new method originates from SIENA [40] and ASCONA [41], which are ligand binding site comparison, search, and filter methods. Instead of describing the local protein environment of

a small molecule ligand, i.e. the ligand binding site, we describe the local protein environment of individual residues. Since there are far more residues in a protein than ligand binding sites this constitutes a significant increase in the search space. Furthermore, our new algorithm was designed to search for all residue micro-environments in the query structure simultaneously. To achieve this, we incorporated a faster in-memory search index for prefiltering, parallel processing and an algorithm optimized for detecting single mutations.

The method and the MicroMiner tool are implemented as part of the NAOMI ChemBio Suite [42, 43].

### Query construction

Protein structures in PDB/mmCIF format are supported as input (see Figure 1A). A distance threshold can be set to control the size of the local environments. Per default, all residues within 6.5 Å from each heavy atom of the reference residue are selected as the environment. Individual reference residues can be selected manually using a text-based configuration file. Additionally, we provide three preselection options: (i) search with residue environments as they are present in the input structure file ('full\_complex'); (ii) use each chain in the input structure separately for micro-environment construction by only including residues from the same chain as the reference residue, effectively searching with monomeric structures ('monomer'); and (iii) searching exclusively with the residue environments located at protein–protein interfaces, specifically, environments comprising residues from multiple chains ('ppi').

Subsequently, the selected query residue micro-environments are processed and query sequence fragments are extracted (see bottom of Figure 1A). If the query fragments are shorter than a minimum length (default 7 residues), they are elongated in both N- and C-terminal directions until they fulfill the length requirement. Fragments are ignored if they cannot be extended to the required length. Then, the sequence fragments are used for candidate selection.

### Candidate database construction and selection

A fast k-mer prefiltering is applied to the protein structure database to select candidate structures likely to contain similar local environments (see Figure 1B), which is a similar approach to well-established tools [7–9, 11] and is also used by SIENA [40].

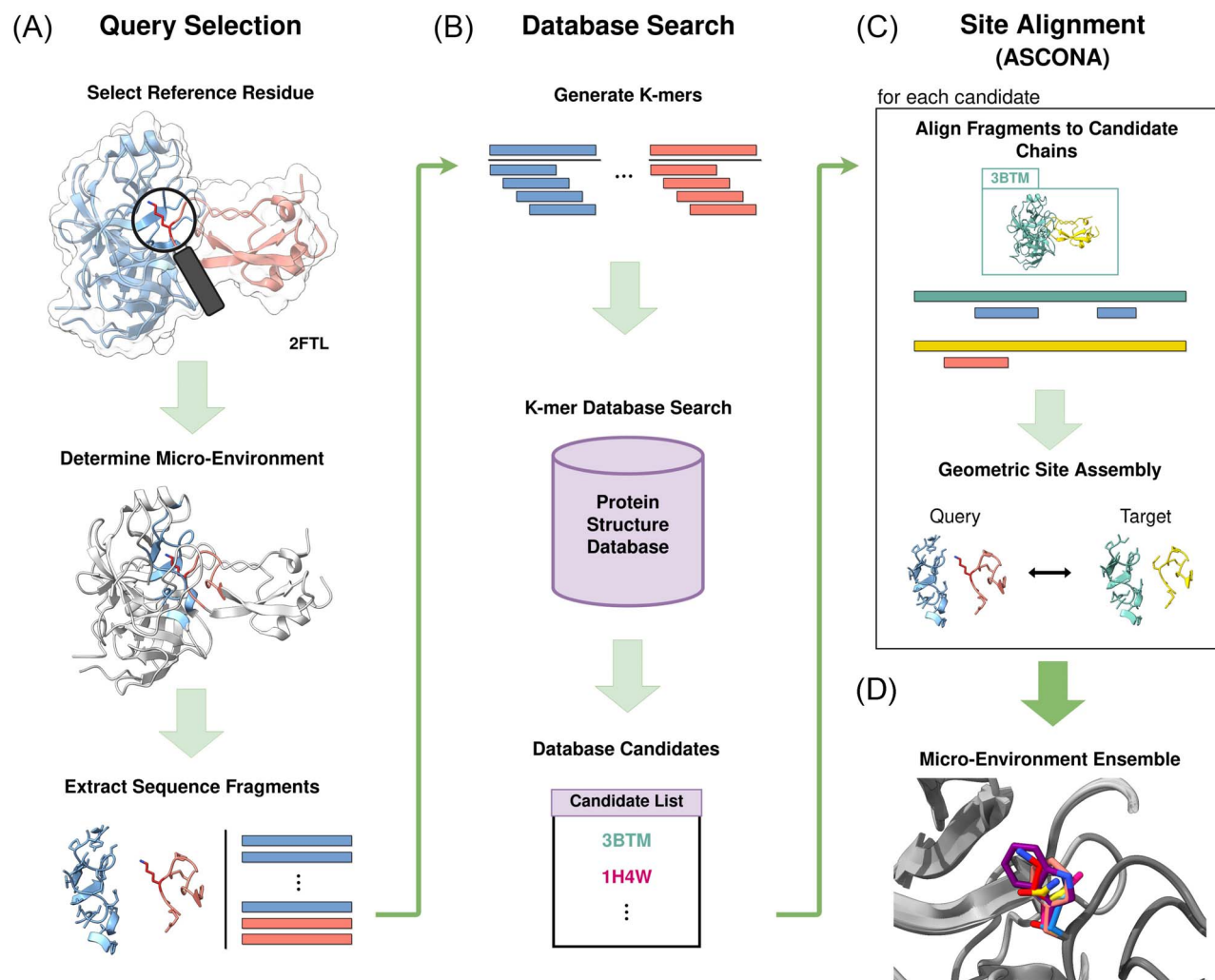
The k-mer search database is implemented as in-memory arrays inspired by MMseqs2 [9] (see section 1 in the Supporting Information for details). To construct the database, MicroMiner takes a directory of PDB/mmCIF files as input and maps k-mers to the chains of structure complexes. The peptide chains from the ATOM records of the structure files are extracted with NAOMI [42, 43]. We map posttranslationally modified residues to the 20 standard proteinogenic amino acids (see Table S1) and ignore k-mers containing other residues. By default, we use 5-mers.

The prefiltering for candidate structures starts by generating k-mers from the query sequence fragments of the query residue micro-environment (see Figure 1B).

In the standard search mode, the candidate selection strategy of SIENA is executed, which reports a candidate if the structure contains a threshold-controlled percentage of the query micro-environments k-mers [40].

We developed a single\_mutation mode, in which the algorithm searches for mutations at the position of the reference residue. The desired micro-environment hits should be identical in sequence except for the position of the reference residue.





**Figure 1.** MicroMiner workflow. The workflow for searching with a single residue's 3D micro-environment is illustrated. **(A)** Query selection: selection of a reference residue (Lys15 of chain I of BPTI complex with trypsin, 2FTL) and determination of its structural neighborhood (micro-environment) from which connected sequence fragments are extracted. **(B)** Database search: Generation of k-mers from the fragments for a database search to obtain a list of candidate structures. **(C)** Local site alignment with ASCONA: The candidates are aligned to the query micro-environment. **(D)** The structure ensemble of similar local micro-environments is reported.

Therefore, instead of using the original k-mers for the reference fragment, we generate sets of similar k-mers in which the reference residue is substituted by one of the other 19 standard amino acids, respectively. With these 19 k-mer sets, we look for structures containing the entire mutated reference fragment by considering only pairs of k-mer hits with the same distance on the sequence as the query k-mers. In addition, if there is more than one fragment in the query micro-environment, we expect at least a single further k-mer match on the additional fragments to avoid random hits.

### Structural site alignment

The ASCONA algorithm aligns each candidate structure to the query site by identifying local protein sites in the candidate structure with a sequence and geometric arrangement similar to the query residue micro-environment. The ASCONA algorithm is described in detail by Bietz et al. [41]. Firstly, the query site is represented through peptide fragments elongated to the minimal size (by default 7 residues). Next, local sequence alignments between the fragments and all chains of the target structure are computed. Then, the resulting sequence-based matches are

assembled geometrically to match the query site using a fuzzy geometric scoring of the matched fragments' backbone atoms. ASCONA scores the distance deviations and Euclidean distances of rotation quaternions of the query fragments and a potential assembled target site to identify a set of target fragments with similar relative orientation and distances.

To align single mutation sites, we designed a seed and extend strategy instead of using dynamic programming. First, we search for exact seeds in the target sequence using linear string matching [44] without elongating the query's fragments beforehand. Again, we substitute the reference residue by all 19 amino acids to mimic a point mutation at that position. Then, short seeds are extended to the minimum length without gaps. Extended seeds will be accepted if they contain, at most, a maximal number of mismatches (by default 2). The subsequent geometrical fragment scoring by ASCONA is unchanged.

The result is one or more structurally validated residue-wise alignments of the query site with sites in the target structure, which may be within a single chain or different chains of multimeric proteins. Additionally, there may be multiple hits in the target structure, such as in homomeric structures.

**Table 1:** Used mutation data sets and their experimental measurements upon mutation.

Data set name	Mutation location	Experimental measurement	Has structures for mutant
ProTherm [45]	Protein mutations	Thermodynamic changes	Yes
FireProtDB [46]	Protein mutations	Thermodynamic changes	No
ThermoMutDB [47]	Protein mutations	Thermodynamic changes	Yes
ProThermDB [48]	Protein mutations	Thermodynamic changes	No
SKEMPI2 [49]	Protein-protein interface	Binding free energy changes	No
Platinum [50]	Protein-ligand interface	Binding affinity changes	Yes
Shanthirabalan et al. [51]	Protein mutations	Structural deviations	Yes

## Output

MicroMiner generates a residue-wise hit list for each query residue micro-environment as TSV file. Optionally, PDB files of the structure ensembles superposed on the environments' C $\alpha$  atoms can be exported. Multiple similarity measures of the aligned micro-environments are calculated and reported, including sequence identity, root mean square deviation (RMSD) of the matched C $\alpha$  atoms and all atoms, and the mean local distance difference test (LDDT) score of the environments' C $\alpha$  atoms. Global similarity measures are also provided, such as the global sequence identity of the reference residue's chain to the aligned chain. The output allows for easy hit filtering by local and global similarity measures and the visualization of identified micro-environments.

## Data sets preparation

Table 1 shows the mutation data sets used.

We downloaded FireProtDB ("EXPORT CSV" option, 15 November 2022), ThermoMutDB (JSON file, 15 November 2022), SKEMPI2 (version 08.06.2018) and Platinum (16 June 2021) from their respective websites. ProTherm was obtained from <http://togodb.biosciencedbc.jp/db/protherm> (21 February 2019). ProThermDB (version 29\_march\_2021) and the data set of Shanthirabalan et al. were provided through their authors after personal communication.

All data sets were processed into a uniform format and single mutations with a protein structure for the wild-type (and mutant) were extracted. In the case of the Shanthirabalan et al. data set, we had to apply additional preprocessing to determine the position of the mutation (see section 2 in the Supporting Information). For SKEMPI2 and Platinum, the custom PDB files were downloaded. The PDB files from PDB version 20230331 were used for all other data sets.

## Evaluation of 3D mutation search

We used 123 unique wild-type PDB structures paired with known mutant structures to validate our method for searching single mutations. Only ProTherm, ThermoMutDB, Platinum and the Shanthirabalan et al. data set contain such structure pairs. Single mutation structure pairs are extracted as the tuple of PDB-ID and residue type of wild-type and mutant, respectively, as well as the sequence position of the residue in the wild-type. We also use the wild-type chain identifier except for ProTherm, where it is mostly unavailable.

Then we use MicroMiner with the structure file of the wild-type as the query to search for single mutations in the PDB. Successful retrieval is evaluated by checking if there was a hit with the expected mutant PDB-ID and the residue type in MicroMiner's output for the corresponding wild-type residue position. We used

'full\_complex' mode for Platinum since it comes with custom PDB files. For all other data sets, we used 'monomer' mode.

## MMseqs2

We used MMseqs2 [9] (version ad6dfc66d7bbc4fd626fc19adf10ba587bc137c4) with the `pdb_seqres.txt` (version 18 August 2023) downloaded from the PDB FTP server. We use the search module with default parameters, except for `-max-seqs` that we set to one million for an exhaustive search against the PDB. Chain hits are determined by checking if there was a hit for the query tuple of PDB-ID and chain identifier with the expected PDB-ID of the target.

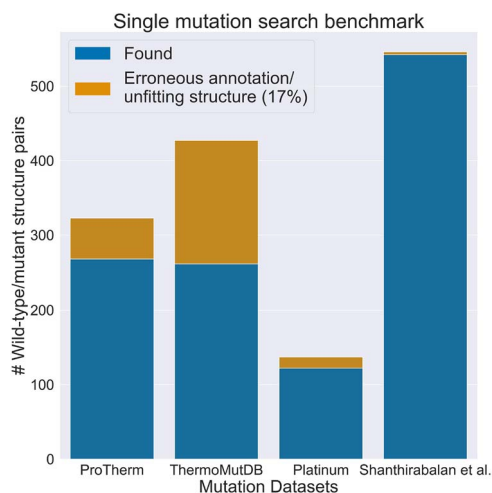
## Annotating experimentally measured effects upon mutations with protein structures for the mutant

We use MicroMiner and the PDB to annotate existing data sets of experimentally measured effects of single mutations with experimentally solved protein structures for the mutant. All single mutations with a protein structure for the wild-type were extracted from ProTherm, FireProtDB, ThermoMutDB, ProThermDB, SKEMPI2 and Platinum. Single mutations are extracted as a tuple of wild-type PDB-ID, residue type, residue position and chain identifier (except for ProTherm), and the residue type of the mutant. Then, we query the PDB with the wild-type protein structures for single mutations using MicroMiner. From the results, we extracted all hits matching the single mutation tuples. For analysis, hits with a global sequence identity < 40% are discarded. Since they come with custom PDB files, we run experiments on Platinum and SKEMPI2 with 'full\_complex' mode. On all other data sets 'monomer' mode was used.

## RESULTS AND DISCUSSIONS

### Evaluation of MicroMiner for structural mutation search

The single mutation search benchmark results are illustrated in Figure 2. Initially, in this retrospective experiment, MicroMiner could only retrieve an average of 83% of mutant structures from the PDB. Surprisingly, after investigating the pairs not found by MicroMiner, we found that many wild-type/mutant structure pairs from the evaluation data sets were erroneously annotated or, for example, had additional mutations in the direct neighborhood. This last finding was surprising as the data sets were reported only to include single and not multiple mutations. After excluding these cases, we found that MicroMiner retrieved 100% of the point mutant structures, even for the Shanthirabalan et al. data set designed to capture high structural changes upon mutations. The list of problematic mutations can be found in



**Figure 2.** Performance of MicroMiner for retrieval of known structure pairs with a single mutation. The number of correct mutant structures MicroMiner retrieved from the PDB, given the wild-type structure is shown for four data sets. 'Found' shows the number of successful retrievals. 'Erroneous annotation/unfitting structure' shows the number of cases in which the expected mutant structure could not be retrieved because of erroneous data annotation (e.g. the same amino acid was present; no mutation) or structural matching criteria were not met (e.g. the mutant structure contained additional mutations at the local 3D micro-environment).

the file 'problematic\_single\_mutations.tsv' in the Supporting Information.

We analyzed the failed cases manually by inspecting the mutation positions of the superposed structure pairs in the Mol\* 3D viewer [52]. In total, we found 189 problematic structure pairs in the data sets. 140 mutations were incorrectly annotated for multiple reasons. In 116 cases, there was no mutation. For six mutations, the position in the structure was unresolved. In addition, six mutations had another adjacent mutation in the sequence, despite being annotated as point mutation variants. Lastly, 12 structure pairs were unrelated proteins we could not reliably align with TM-Align [13]. The remaining 49 cases were not found because they did not meet the matching criteria of MicroMiner. There were either additional mutations or unresolved residues in the local 3D site in the mutant structure. We consider the local environments different in these cases because the query micro-environment cannot be matched completely. For example, for nine PPI mutations from Platinum, the second chain of the PPI was missing in the target structure from the PDB, or there were additional nearby mutations in the other chain (see Table S2). However, these nine mutations are successfully retrieved with the 'monomer' mode. These cases demonstrate the local details in the structure MicroMiner can check and evaluate.

## Comparison of micro-environment search and homology detection

We ran the single mutation benchmark with MMseqs2 [9] to compare conventional homology detection and 3D micro-environment search. MMseqs2 is a widely utilized standard tool for homology detection employing local sequence alignments. Local sequence aligners provide an alignment of the protein chains and do not directly report single mutations. Therefore, a direct comparison of MMseqs2 results to the mutated residues in the benchmark is not straightforwardly possible. Instead, we simply searched with MMseqs2 for the target sequence of the mutant given the query sequence of the wild-type. From the

1041 unique chain pairs in the benchmark, MMseqs2 successfully reported 99.42%. As expected, it is easy for MMseqs2 to reliably retrieve the target sequences for the benchmark test. The missed pairs contained only the structure pairs identified as actually different proteins in the previous section. Interestingly, one sequence pair flagged by us as a different protein was found by MMseqs2, which was due to many unresolved residues in the atom sequence which are present in the seqres sequence used by MMseqs2.

MMseqs2 also reports the 94 sequence pairs corresponding to 178 problematic single mutations described in the previous section. These are 9.10% of the sequence pairs MMseqs2 reports from the benchmark chain pairs and corresponds to 15.67% of all single mutations in the benchmark.

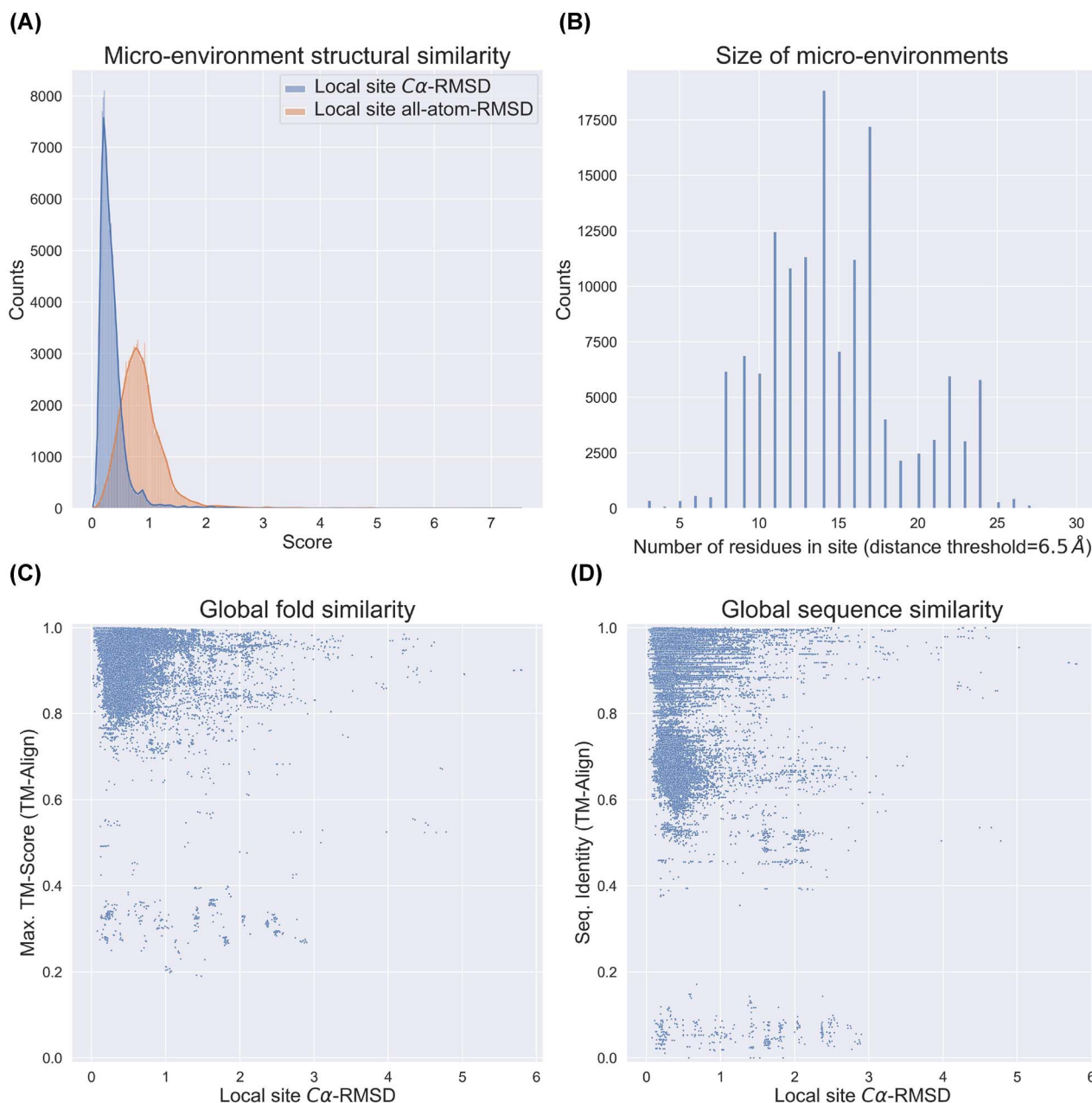
In general, even though similar chains can be reliably reported, extracting specific residues constituting single mutations from MMseqs2's local sequence alignment output is not trivial. While it is possible to select only the alignments with a single mismatch, it would reduce the number of results. Identifying single mutations when there are multiple mismatches and gaps present is more complicated. For example, even when only a single mismatch has no other mismatches in its sequence neighborhood does not mean that there are no close contacts to other mutated residues in 3D space, either in the same chain or in different protein chains at PPIs. Furthermore, experimental artifacts, like unresolved residues, need to be handled. By representing single mutations as residue 3D micro-environments MicroMiner can differentiate between these cases and additionally compute structural similarity measures of the mutation site.

Nevertheless, fast and sensitive local sequence aligners like MMseqs2 could potentially be used as a prefiltering for 3D micro-environment analysis depending on the protein similarity (and homology) constraints one wants to impose on the search. We wanted to focus on the local 3D micro-environments for the prefiltering which consist of multiple smaller sequence fragments and frequently coming from different protein chains. However, if suitable for the application at hand, it is possible to combine MicroMiner with existing local sequence aligners, for example, for the analysis of the structural details contained in massive protein structure data sets.

## Local flexibility and global similarity analysis

The analysis results of the hits for the 123 unique wild-type proteins from the evaluation mutation data sets are shown in Figure 3. 137 325 similar environment hits for 3344 query micro-environments are reported by MicroMiner.

The distribution of the local  $C\alpha$ -RMSD of the environment hits is shown Figure 3A and has a mean of 0.36 Å (median=0.29 Å, standard deviation=0.27 Å) and the mean local all-atom-RMSD is 0.87 Å (median=0.81 Å, standard deviation=0.44 Å). The low RMSD values demonstrate that MicroMiner retrieves environments with, on average, high structural similarity. The size of the local environments is illustrated in Figure 3B. The mean number of residues in environments is 14.77 (median=14.00, standard deviation=4.37). Intuitively, residues buried in the structure's core will have more residues in their 3D micro-environment than residues at the surface and in loops. The relationship between similar local environments of single mutations and the proteins' global structure and sequence similarity are depicted in Figure 3C–D. The mean maximal TM-Score of the chains with the mutation is 0.97 (median=0.98, standard deviation=0.05) and 0.91 (median=0.94 standard deviation=0.10) for sequence identity.



**Figure 3.** Analysis of single mutation hits. **(A)** Histograms of local C $\alpha$ -RMSD (blue) and local all-atom RMSD (orange) values of the hits to the query. **(B)** Histogram of micro-environment query sizes described by the number of residues in the sites. Micro-environments are defined with a distance threshold of 6.5 Å from the reference residue. **(C)** Comparison of local C $\alpha$ -RMSD on the x-axis to global fold similarity on the y-axis. Max. TM-Score is the maximum of the two TM-Scores reported by TM-Align. **(D)** Comparison of the environment's local C $\alpha$ -RMSD on the x-axis to the global protein sequence similarity on the y-axis.

The results show that, on average, the wild-type structures and the mutant's reported structure are related and share the same fold. Accordingly, MicroMiner can provide plenty of structures for investigating the local structural changes of mutations through available experimental structures from the PDB.

We investigated the 0.33% of hits with a max-TM-Score  $\leq 0.5$  for false positives. These hits are in query sites with fewer residues (see bottom left in Figure S1). Manual inspection of these hits shows that they appear at structural variable regions and regions of missing data, i.e., mainly at the termini (see Figure S2). These query sites are small and solvent-exposed with no residues in the 3D neighborhood, leaving structurally and sequentially unspecific single chain fragments. However, such false positive hits can

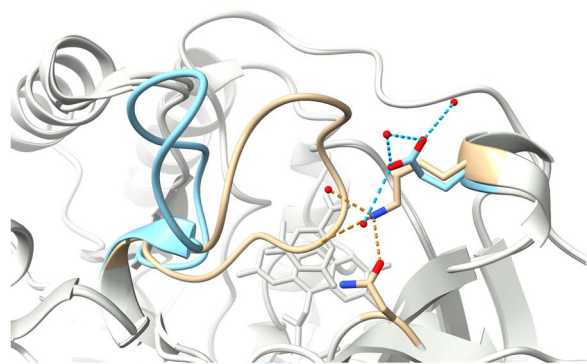
be avoided by ignoring small sites at unspecific regions or only considering structures with high global similarity.

While many of the hits with max-TM-Score  $\leq 0.5$  are false positive random hits, some of them arguably might be interesting cases where globally unrelated proteins share locally well-conserved sites, potentially a structural or functional motif (see Figure S3).

### Local structure deviations

While single mutations leading to a folded protein usually do not have a substantial effect on the global fold [51], MicroMiner reports multiple environments with higher local RMSD (see Figure 3C). Therefore, hits with high local RMSD (and reasonable





**Figure 4.** Example hit with a local  $C\alpha$ -RMSD of 4.21 Å. Shows a mutation of dihydroorotate dehydrogenase from *Lactococcus lactis* from Lys213 (2DOR, chain A) to Glu213 (1JQV, chain A). The mutation is responsible for the open and closed form of the binding site [53].

global similarity) might be interesting cases where considerable structural changes upon single mutations are exemplified. However, fully automatic isolation of structural changes upon mutation is challenging since different crystallization conditions, bound ligands and other protein modifications might play a role [51]. Still, hits with higher local RMSD are potentially interesting cases for visual inspection that can give valuable insights. For example, Figure 4 shows a hit with high local RMSD due to a nearby loop's movement. In this case, it was reported [53] that the mutation also found by MicroMiner is responsible for an active site's open and closed loop conformation. Another example of larger global re-arrangements is shown in Figure S4.

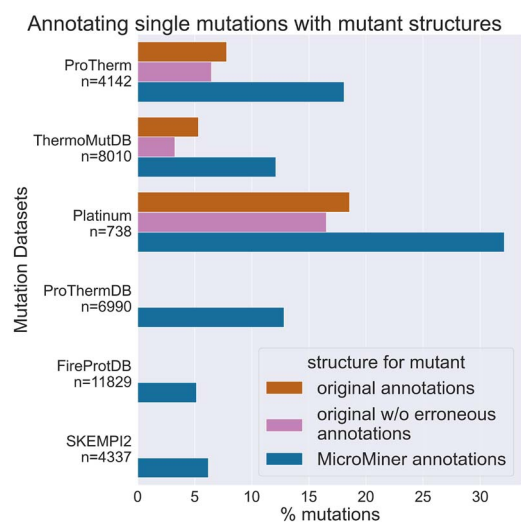
### Single mutations in the PDB

We used MicroMiner to extract the available structure pairs exemplifying single mutations from the PDB for single chains ('monomer' mode) and protein-protein interfaces ('ppi' mode), respectively. All-vs-all experiments were performed by querying each PDB structure against the PDB database using their asymmetric units. For single chains, we found 255 853 766 structure hits that exemplify the local structural changes of single mutations. After filtering bidirectional hits and selecting a single mutation tuple (query PDB-ID, chain ID, residue type, residue position and hit residue type) by choosing the hit with the highest global sequence identity and filtering hits with a global sequence identity < 40% and mutations to non-standard amino acid, we get 4 868 764 single mutations structure pairs. For PPI mutations, we found 45 752 144 single mutation hits. After filtering, 799 129 hits that exemplify the local structural changes of single mutations at PPIs remain. We also provide mutations to non-standard residues in separate files. The data sets can be found at <http://doi.org/10.25592/uhhfdm.13411>.

### Annotating mutation effect measurements with structures for the mutant

We used MicroMiner to annotate the mutations of ProTherm, FireProtDB, ThermoMutDB, ProThermDB, SKEMPI2 and Platinum with experimental protein structures from the PDB for the mutant. Figure 5 shows the improvement in annotation.

For all data sets, the number of mutant structures could be increased. Even half of the data sets did not previously contain mutant structure annotations. In summary, before, all six data sets had 414 unique mutations with a wild-type/mutant structure pair annotated (596 uncorrected). With



**Figure 5.** Improvement of experimental structure annotation coverage for mutation effect data. The x-axis shows the percentage of mutations derived from the data set with a known wild-type structure. On the y-axis, the mutation data sets are listed, including the absolute number of single mutations with a structure for the wild-type. The 'original annotations' bars illustrate the portion of mutations annotated with a structure for the mutant by the data sets curators. Bars labeled with 'original w/o erroneous annotations' show the percent of mutations with erroneous data annotation removed. The label 'MicroMiner annotations' gives the percentage of mutations that could be annotated with structures for the mutant from the PDB using MicroMiner.

the structures retrieved with MicroMiner there are now 2653 unique mutations with wild-type/mutant structure pairs—a 6.4-fold increase. Considering structure hits for the same mutation, 16 313 pairs can be used to describe mutations as structure ensembles.

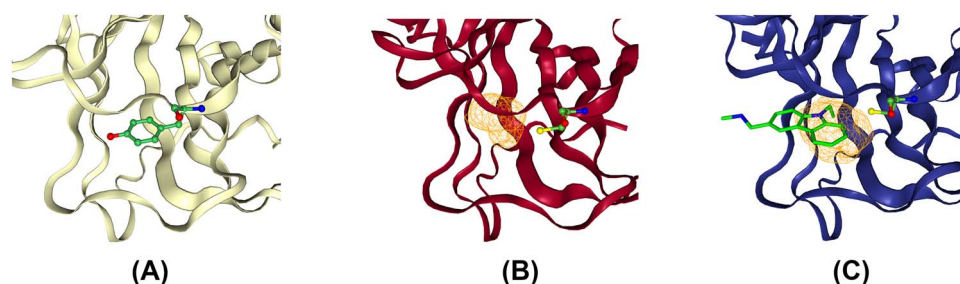
Having protein structures for both wild-type and mutant can be valuable for training and validating methods for predicting effects upon mutation. However, most mutation effect measurements are still without a mutant structure. This is unsurprising, since there is probably no experimental protein structure for every mutation in the PDB. Furthermore, even many of the mutations measured might not lead to folded proteins or are observable in structure determination experiments.

The results illustrate that manual annotation of mutation data with structures is a cumbersome and error-prone task that can now be automated. The generated structure annotations can be found in the supporting information.

### Case study p53: detecting ligand binding sites emerging upon mutation

In a case study on the cellular tumor antigen p53 (p53), we will show retrospectively how MicroMiner can bridge the gap between mutation analysis and structure-based drug design tools. We combine MicroMiner with the binding site prediction method DoGSite3 [54] for the ligand-based rescue of p53. This experiment was solely performed on the <https://proteins.plus/> webserver.

p53 is a prime target in cancer therapy as it is responsible for multiple deleterious effects on cancer cells. Missense mutations through which p53 loses its tumor suppressor activities are the most common single gene alterations in human cancers [55, 56]. Prevalent mutations can be grouped into mutations hindering the protein-DNA interaction and structural mutations leading to thermolabile variants and, therefore, a loss of function [56, 57]. It is estimated that one-third of the pathological mutations are



**Figure 6.** Identifying druggable binding sites upon mutation in p53. (A) Wild-type p53 (1TUP, chain A). No pocket is predicted for wild-type Y220. (B) Y220C p53 mutant (6SHZ, chain A). A single pocket is predicted at the position where the tyrosine side chain resided. (C) Y220C p53 mutant with ligand P83 (2VUK, chain B).

structural mutations that potentially can be targeted with small molecule chaperons to regain tumor suppressor activity [57]. This rescue strategy is a promising paradigm that renders specific p53 mutants druggable for ‘personalized’ cancer therapy. An example is the mutation Y220C, the ninth most frequent cancer-related mutation in p53 and responsible for an estimated number of 100 000 cancer cases per year [57, 58]. Currently, a phase I/II candidate in the clinic is targeting Y220C p53 mutants [56].

We can apply MicroMiner retrospectively [58] to show how to identify binding sites emerging upon mutations. First, a list of diseases-related p53 mutations can be obtained from the TP53 Database [59], UniProt [60] or the literature. We then can query the PDB for single residue mutations in the p53 wild-type using MicroMiner and investigate diseases-related mutations on the example of experimentally solved structures. Figure 6A highlights Tyr220 in the wild-type p53. DoGSite3 predicts no pockets in the proximity of Tyr220. In Figure 6B, tyrosine is mutated to a smaller cysteine residue (the disease-related Y220C mutation). In the retrieved structure, a pocket is predicted in proximity to Cys220 where the tyrosine side chain resided in the wild-type. This result corresponds to the conclusion of Joerger et al. [58] in their pioneering study where the binding pocket of the Y220C mutant was originally described. A series of structure-based studies followed [57, 61, 62], identifying and optimizing small molecules to stabilize mutant p53. Figure 6C shows a Y220C mutant structure retrieved by MicroMiner containing the ligand found by Boeckler et al. [61].

We hope that MicroMiner will be helpful for mutation analysis in structure-based drug discovery by finding druggable pockets and binding small molecules or functional groups. In this way, MicroMiner can support the development of selective drugs.

## CONCLUSION

Having structures of similar protein sites available enables many applications, analyses, and method development across different disciplines in structural bioinformatics. We here presented a new method to efficiently search for similar local residue 3D micro-environments in protein structure databases implemented in the tool MicroMiner.

Despite the great success in structure prediction, a method like AlphaFold2 can not be used directly to predict the effects of single mutations [31]. With MicroMiner, we extracted hundreds of millions of experimental wild-type/mutant structure pairs that exemplify the local structural changes of single mutations from the PDB. Many of these pairs elucidate considerable structural changes upon mutation. We believe that having accurate structures for the mutant will be crucial for improving future methods

for mutation effect prediction, mutation modeling, protein structure prediction and side chain modeling.

The web interface provides an interactive way to explore a protein’s structural mutation landscape through experimental PDB structures. This can help model mutations and side chain conformations and investigate mutation effects that lead to thermo-stabilization, better crystallization or even detecting novel binding sites upon mutations.

Given the enormous increase of available protein structures, we believe methods to search local protein sites for the focused analysis of structural details will be an important addition to existing tools. For example, MicroMiner could analyze functional sites, motifs, co-evolved contacts and local protein flexibility. Such applications and further methodological improvements are subject to future work.

With MicroMiner, we present a novel method to analyze the structural changes of mutations. MicroMiner can structurally annotate relevant unexplored mutations in protein structures, bridge the gap between mutation analysis and structure-based drug discovery and foster new methods and modeling approaches.

### Key Points

- We propose a new method implemented in the tool MicroMiner to extract similar local residue micro-environments at scale from protein structure databases to explore the details characterizing structure and function.
- With MicroMiner, we extracted  $> 255 \times 10^6$  amino acid pairs in protein structures from the PDB, exemplifying single mutations’ local structural changes for single chains and  $> 45 \times 10^6$  pairs for protein-protein interfaces. We believe these large data sets will enable the development of future methods and modeling approaches for mutation analysis and structure prediction.
- We provide MicroMiner as a stand-alone tool and in a web interface to interactively explore a protein’s mutational landscape and connect mutation analysis with structure-based drug discovery tools.

## ACKNOWLEDGMENTS

The authors thank Annika Jochheim and Jonathan Pletzer-Zelgert for discussions about implementation details. Christiane Eht and Patrick Penner for proofreading the manuscript, discussions about the case study and implementation details of the webserver.

This work was orally presented at the conference on Intelligent Systems for Molecular Biology (ISMB), July 2021. We are grateful for all the comments and suggestions from numerous conference participants.

## FUNDING

This work was supported by the German Federal Ministry of Education and Research as part of de.NBI [grant number 031L0105] and protP.S.I. [grant number 031B0405B].

## DATA AND SOFTWARE AVAILABILITY

MicroMiner is available as part of the NAOMI ChemBio Suite and is free for academic use and evaluation purposes at <https://software.zbh.uni-hamburg.de/>. MicroMiner can be used interactively within our webserver <https://proteins.plus/>. Currently, the webserver only supports the “single\_mutation” search mode of MicroMiner. However, the stand-alone version of MicroMiner provides the “single\_mutation” search mode and the “standard” search mode which can search for sequence identical and mutated micro-environments. The generated data sets of single mutation structure pairs are available at <http://doi.org/10.25592/uhhfdm.13411>. The code for performing the experiments is available at [https://github.com/rareylab/microminer\\_utils](https://github.com/rareylab/microminer_utils).

## REFERENCES

- Stella M, Hurtley. Continuing the resolution revolution. *Science* 2018;**360**(6386):280.11–282.
- Nakane T, Kotecha A, Sente A, et al. Single-particle cryo-EM at atomic resolution. *Nature* 2020;**587**(7832):152–6.
- Yip KM, Fischer N, Paknia E, et al. Atomic-resolution protein structure determination by cryo-EM. *Nature* 2020;**587**(7832):157–61.
- Baek M, DiMaio F, Anishchenko I, et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science* 2021;**373**(6557):871–6.
- Jumper J, Evans R, Pritzel A, et al. Highly accurate protein structure prediction with AlphaFold. *Nature* 2021;**596**(7873):583–9.
- Varadi M, Anyango S, Deshpande M, et al. AlphaFold protein structure database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Res* 2022;**50**(D1):D439–44.
- Pearson WR, Lipman DJ. Improved tools for biological sequence comparison. *Proc Natl Acad Sci U S A* 1988;**85**(8):2444–8.
- Altschul SF, Gish W, Miller W, et al. Basic local alignment search tool. *J Mol Biol* 1990;**215**(3):403–10.
- Steinegger M, Söding J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat Biotechnol* 2017;**35**(11):1026–8.
- Steinegger M, Meier M, Mirdita M, et al. HH-suite3 for fast remote homology detection and deep protein annotation. *BMC Bioinform* 2019;**20**(1):1–15.
- Buchfink B, Reuter K, Drost HG. Sensitive protein alignments at tree-of-life scale using DIAMOND. *Nat Methods* 2021;**18**(4):366–8.
- Holm L, Sander C. Protein structure comparison by alignment of distance matrices. *J Mol Biol* 1993;**233**(1):123–38.
- Zhang Y, Skolnick J. TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res* 2005;**33**(7):2302–9.
- van Kempen M, Kim SS, Tumescheit C, et al. Foldseek: fast and accurate protein structure search. *Nat Biotechnol* 2023;1546–696.
- Overington J, Donnelly D, Johnson MS, et al. Environment-specific amino acid substitution tables: tertiary templates and prediction of protein folds. *Protein Sci* 1992;**1**(2):216–26.
- Bagley SC, Russ B, Altman. Characterizing the microenvironment surrounding protein sites. *Protein Sci* 1995;**4**(4):622–35.
- Yoon S, Ebert JC, Chung EY, et al. Clustering protein environments for function prediction: finding PROSITE motifs in 3D. *BMC Bioinform* 2007;**8**(SUPPL. 4):1–12.
- Das S, Lee D, Sillitoe I, et al. Functional classification of CATH superfamilies: a domain-based approach for protein function annotation. *Bioinformatics* 2015;**31**(21):3460–7.
- Bhatnagar A, Apostol MI, Bandyopadhyay D. Amino acid function relates to its embedded protein microenvironment: a study on disulfide-bridged cysteine. *Proteins* 2016;**84**(11):1576–89.
- Mazmanian K, Sargsyan K, Lim C. How the local environment of functional sites regulates protein function. *J Am Chem Soc* 2020;**142**(22):9861–71.
- Blum M, Chang HY, Chuguransky S, et al. The InterPro protein families and domains database: 20 years on. *Nucleic Acids Res* 2021;**49**(D1):D344–54.
- Wang J, Lisanza S, Juergens D, et al. Scaffolding protein functional sites using deep learning. *Science* 2022;**377**(6604):387–94.
- Ehrt C, Brinkjost T, Koch O. Impact of binding site comparisons on medicinal chemistry and rational molecular design. *J Med Chem* 2016;**59**(9):4121–51.
- Dehouck Y, Grosfils A, Folch B, et al. Fast and accurate predictions of protein stability changes upon mutations using statistical potentials and neural networks: PoPMuSiC-2.0. *Bioinformatics* 2009;**25**(19):2537–43.
- Pires DEV, Ascher DB, Blundell TL. MCSM: predicting the effects of mutations in proteins using graph-based signatures. *Bioinformatics* 2014;**30**(3):335–42.
- Torg W, Altman RB. 3D deep convolutional neural networks for amino acid environment similarity analysis. *BMC Bioinform* 2017;**18**(302):302.
- Anand N, Eguchi R, Mathews II, et al. Protein sequence design with a learned potential. *Nat Commun* 2022;**13**(1):1–11.
- Torg W, Altman RB. High precision protein functional site detection using 3D convolutional neural networks. *Bioinformatics* 2019;**35**(9):1503–12.
- Burley SK, Berman HM, et al. Protein data Bank: the single global archive for 3D macromolecular structure data. *Nucleic Acids Res* 2019;**47**(D1):D520–8.
- Buel GR, Walters KJ. Can AlphaFold2 predict the impact of missense mutations on structure? *Nat Struct Mol Biol* 2022;**29**(1):1–2.
- Pak MA, Markhieva KA, Novikova MS, et al. Using AlphaFold to predict the impact of single mutations on protein stability and function. *PLoS One* 2023;**18**(3):e0282689.
- Wang CY, Chang PM, Ary ML, et al. ProtaBank: a repository for protein design and engineering data. *Protein Sci* 2018;**27**(6):1113–24.
- Kooistra AJ, Mordalski S, Pándy-Szekeres G, et al. GPCRdb in 2021: integrating GPCR sequence, structure and function. *Nucleic Acids Res* 2021;**49**(D1):D335–43.
- Pucci F, Bernaerts K, Teheux F, et al. Symmetry principles in optimization problems: an application to protein stability prediction. *IFAC-PapersOnLine* 2015;**48**:458–63.
- Usmanova DR, Bogatyreva NS, Bernad JA, et al. Self-consistency test reveals systematic bias in programs for prediction change of stability upon mutation. *Bioinformatics* 2018;**34**(21):3653–8.

36. Pucci F, Bernaerts KV, Kwasigroch JM, Rooman M. Quantification of biases in predictions of protein stability changes upon mutations. *Bioinformatics* 2018;**34**(21):3659–65.
37. Fang J. A critical review of five machine learning-based algorithms for predicting protein stability changes upon mutation. *Brief Bioinform* 2019;**21**(4):1285–92.
38. Sanavia T, Birolo G, Montanucci L, et al. Limitations and challenges in protein stability prediction upon genome variations: towards future applications in precision medicine. *Comput Struct Biotechnol J* 2020;**18**:1968–79.
39. Caldararu O, Mehra R, Blundell TL, Kepp KP. Systematic investigation of the data set dependency of protein stability predictors. *J Chem Inf Model* 2020;**60**(10):4772–84.
40. Bietz S, Rarey M. SIENA: efficient compilation of selective protein binding site ensembles. *J Chem Inf Model* 2016;**56**(1):248–59.
41. Bietz S, Rarey M. ASCONA: rapid detection and alignment of protein binding site conformations. *J Chem Inf Model* 2015;**55**(8):1747–56.
42. Urbaczek S, Adrian Kolodzik J, Fischer R, et al. NAOMI: on the almost trivial task of reading molecules from different file formats. *J Chem Inf Model* 2011;**51**(12):3199–207.
43. Urbaczek S, Kolodzik A, Groth I, et al. Reading PDB: perception of molecules from 3D atomic coordinates. *J Chem Inf Model* 2013;**53**(1):76–87.
44. Ukkonen E. Approximate string-matching with q-grams and maximal matches. *Theor Comput Sci* 1992;**92**(1):191–211.
45. Shaji Kumar MD, Abdulla Bava K, Michael Gromiha M, et al. ProTherm and ProNIT: thermodynamic databases for proteins and protein-nucleic acid interactions. *Nucleic Acids Res* 2006;**34**(Database issue):D204–6.
46. Stourac J, Dubrava J, Musil M, et al. FireProtDB: database of manually curated protein stability data. *Nucleic Acids Res* 2021;**49**(D1):D319–24.
47. Xavier JS, Nguyen TB, Karmarkar M, et al. ThermoMutDB: a thermodynamic database for missense mutations. *Nucleic Acids Res* 2021;**49**(D1):D475–9.
48. Rahul Nikam A, Kulandaisamy KH, Sharma D, Michael Gromiha M. ProThermDB: thermodynamic database for proteins and mutants revisited after 15 years. *Nucleic Acids Res* 2021;**49**(D1):D420–4.
49. Jankauskaite J, Jiménez-García B, Dapkunas J, et al. SKEMPI 2.0: an updated benchmark of changes in protein-protein binding energy, kinetics and thermodynamics upon mutation. *Bioinformatics* 2019;**35**(3):462–9.
50. Pires DEV, Blundell TL, Ascher DB. Platinum: a database of experimentally measured effects of mutations on structurally defined protein-ligand complexes. *Nucleic Acids Res* 2015;**43**(D1):D387–91.
51. Shanthirabalan S, Chomilier J, Carpentier M. Structural effects of point mutations in proteins. *Proteins* 2018;**86**(8):853–67.
52. Sehnal D, Bittrich S, Deshpande M, et al. Mol\* viewer: modern web app for 3D visualization and analysis of large biomolecular structures. *Nucleic Acids Res* 2021;**49**(W1):W431–7.
53. Nørager S, Arent S, Björnberg O, et al. Lactococcus lactis dihydroorotate dehydrogenase mutants reveal important facets of the enzymatic function. *J Biol Chem* 2003;**278**(31):28812–22.
54. Graef J, Ehrt C, Rarey M. Binding site detection remastered: enabling fast, robust, and reliable binding site detection and descriptor calculation with DoGSite3. *J Chem Inf Model* 2023;**63**(10):3128–37.
55. Joerger AC, Fersht AR. The p53 pathway: origins, inactivation in cancer, and emerging therapeutic approaches. *Annu Rev Biochem* 2016;**85**:375–404.
56. Hassin O, Oren M. Drugging p53 in cancer: one protein, many targets. *Nat Rev Drug Discov* 2023;**22**(2):127–44.
57. Bauer MR, Krämer A, Settanni G, et al. Targeting cavity-creating p53 cancer mutations with small-molecule stabilizers: the Y220X paradigm. *ACS Chem Biol* 2020;**15**(3):657–68.
58. Joerger AC, Ang HC, Fersht AR. Structural basis for understanding oncogenic p53 mutations and designing rescue drugs. *Proc Natl Acad Sci U S A* 2006;**103**(41):15056–61.
59. César K, de Andrade EE, Lee EM, et al. The TP53 database: transition from the International Agency for Research on Cancer to the US National Cancer Institute. *Cell Death Differ* 2022;**29**(5):1071–3.
60. The Uniprot Consortium. UniProt: the universal protein knowledgebase in 2023. *Nucleic Acids Res* 2023;**51**(D1):D523–31.
61. Boeckler FM, Joerger AC, Jaggi G, et al. Targeted rescue of a destabilized mutant of p53 by an in silico screened drug. *Proc Natl Acad Sci U S A* 2008;**105**(30):10360–5.
62. Bauer MR, Jones RN, Tareque RK, et al. A structure-guided molecular Chaperone approach for restoring the transcriptional activity of the p53 cancer mutant Y220C. *Future Med Chem* 2019;**11**(19):2491–504.



# Supporting Information:

## Searching Similar Local 3D Micro-Environments in Protein Structure Databases with MicroMiner

Jochen Sieg and Matthias Rarey\*

*Universität Hamburg, ZBH - Center for Bioinformatics, Bundesstraße 43, 20146 Hamburg,  
Germany*

E-mail: matthias.rarey@uni-hamburg.de

### 1. K-mer Indexing and Look-up Table Implementation

Our prefiltering of protein structure databases uses k-mers and k-mer integer indices for efficient lookups (also called q-grams<sup>S1</sup>). A k-mer  $x_k$  is a substring of length  $k \in \mathbb{N}$  of a sequence  $s$ , where  $s$  is from an ordered integer-based alphabet  $A = a_1, \dots, a_n \quad \forall 0 \leq a_1 < \dots < a_n$ . Let a start position of a k-mer in  $s$  be denoted with  $i$ ; then a k-mer is a substring  $x_k = s_i \dots s_{i+k-1}$ . We denote the set of the  $|A|^k$  possible k-mers of length  $k$  as  $A^k \mid x_k \in A^k$ . The function

$$f: A^k \longrightarrow [0, |A|^k - 1]$$

maps each k-mer to a unique unsigned integer representation. We refer to the integer of a particular k-mer as its k-mer-index. The k-mer-index is computed with:

$$f(x_k) = \sum_{j=1}^k x_k[j] \cdot |A|^{k-j}$$

We implemented a k-mer lookup table with two in-memory arrays. The two arrays map k-mer integer indices to k-mers present in particular sequences. The first array holds all sorted k-mer entries extracted from the protein data, and the second is an offset array of size  $20^k + 1$ , mapping the k-mer integer indices to the ranges of k-mer entries in the first array. A k-mer entry contains the k-mers sequence identifier and the starting position of this k-mer in its sequence. We additionally store another array to map sequence identifiers to complex identifiers to handle multimeric structures.

## 2. Preprocessing of Shanthirabalan et al. Data Set

Shanthirabalan et al.<sup>S2</sup> compiled a data set of protein chains representing single mutations. We kindly received a list of the PDB entry pairs and their chain identifier of the wild-type/mutants through personal communication with the authors. To obtain the residue positions of every single mutation, we aligned the respective chains with TMalign.<sup>S3</sup> From the alignment, we selected the mutation with maximal distance to the termini. We removed structure pairs with TM-Scores  $\leq 0.5$ , a sequence identity  $\leq 80\%$  and structure pairs with indels that are not within a distance of 20 residues from each terminus. The code can be found in the `preprocess_shanthirabalan_dataset.py` file at [https://github.com/rareylab/microminer\\_utils](https://github.com/rareylab/microminer_utils).

Table S1: Modified to standard residues conversion table.

Modified Residue	Standard Residue
MSE	MET
SEP	SER
TPO	THR
CSO	CYS
PTR	TYR
KCX	LYS
LLP	LYS
CME	CYS
CSD	CYS
MLY	LYS
TYS	TYR
OCS	CYS
ALY	LYS
FME	MET
CAS	CYS
M3L	LYS
HYP	PRO
CSX	CYS
HIC	HIS
CSS	CYS
YCM	CYS
MLZ	LYS
KPI	LYS
SAC	SER
MEN	ASN
CXM	MET
CGU	GLU
TPQ	TYR
NEP	HIS
OTD	ASP
DAL	ALA
B3K	LYS
B3D	ASP
B3E	GLU
B3A	ALA
B3Y	TYR

Table S2: The 9 mutations from the Platinum data set not found because of the differences at the PPI in another chain of the structural assembly when run in 'full\_complex' mode. The mutations are found when 'monomer' mode is used. Note that the wild-type structures are custom-prepared PDB files by Platinum.

Wild PDB-ID	Mutant PDB-ID	Wild chain	Wild residue	Mutant residue	Wild position	reason
2Z4O	2QD6	A	I	V	50	near other mutation I 150 V in chain B
3NU3	3NU5	A	I	V	50	near other mutation I 150 V in chain B
3OXC	3CYX	A	I	V	50	near other mutation I 150 V in chain B
2JBZ	2WDY	A	D	A	111	Chain C missing in 2WDY
2JBZ	2WDS	A	H	A	110	Chain C missing in 2WDS
1CNQ	1YXI	A	A	L	54	Chain C missing in 1YXI
1AMK	1QDS	A	E	Q	65	Chain B missing in 1QDS
2TDM	1TSV	A	R	A	179	Chain B missing in 1TSV
2TDM	1TSY	A	R	K	179	Chain B missing in 1TSY

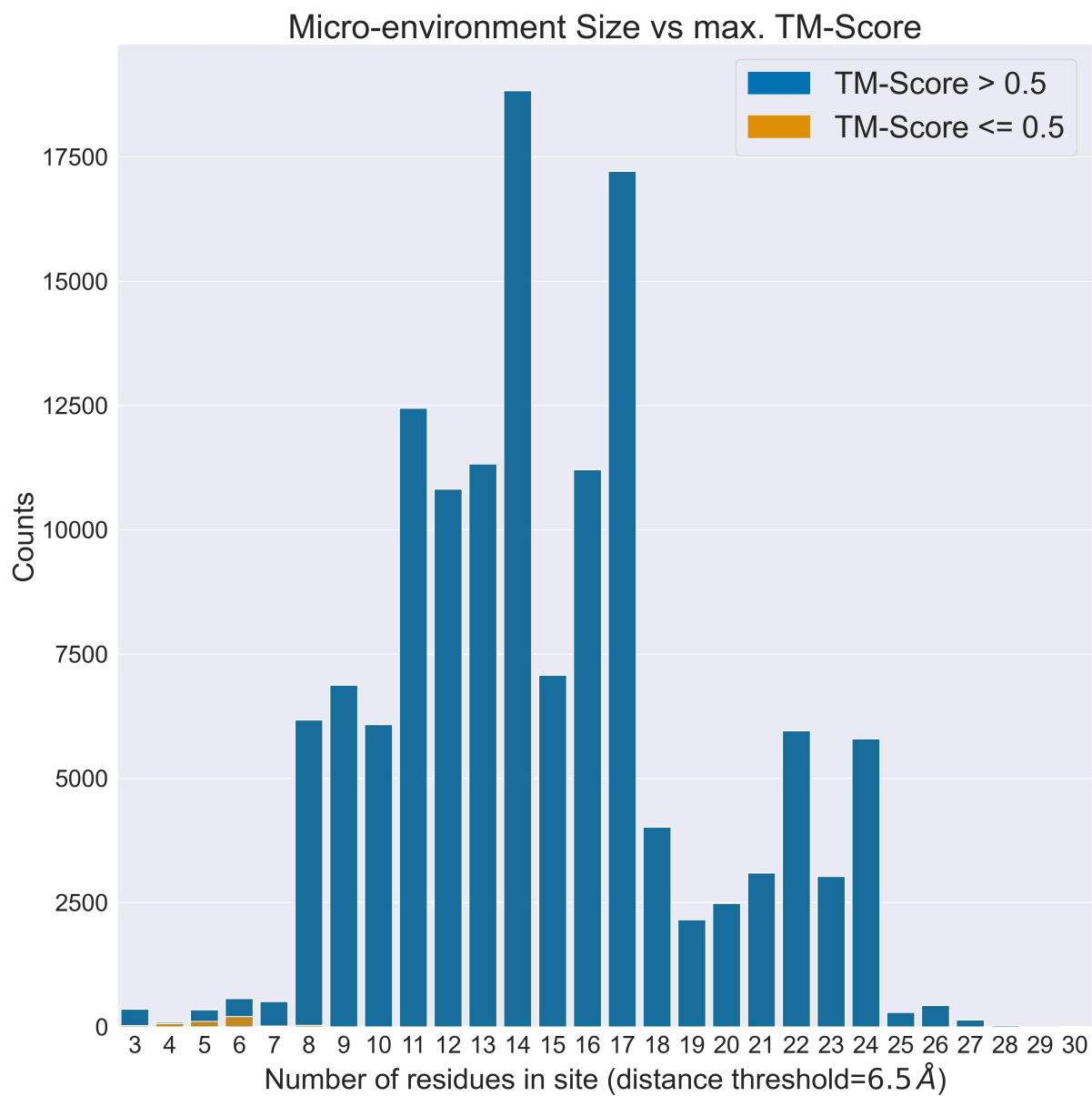
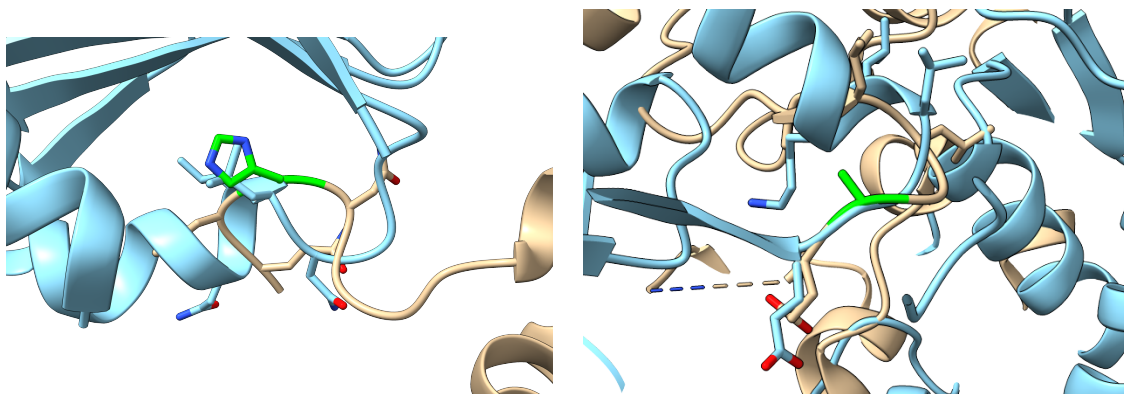


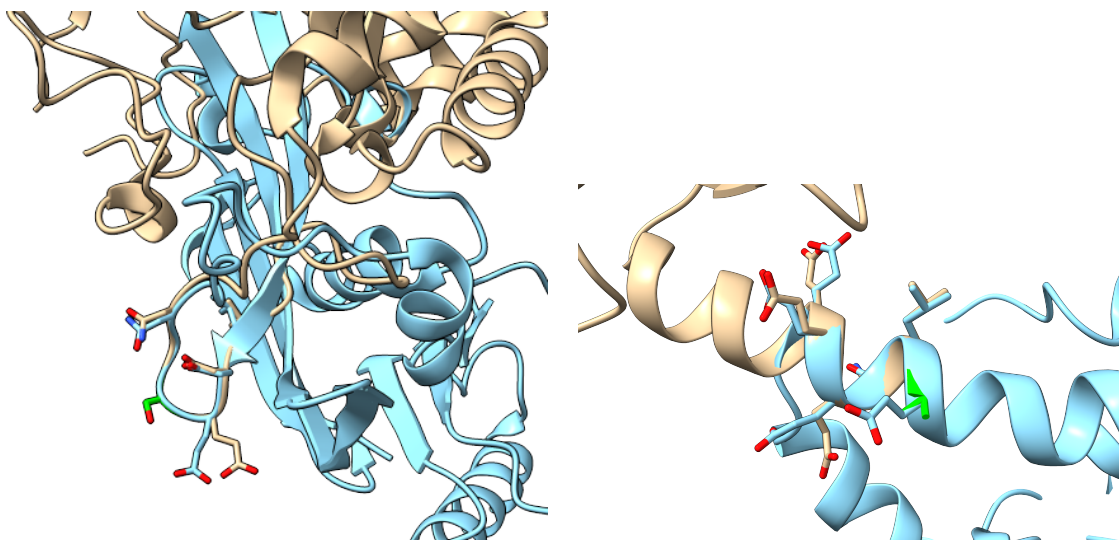
Figure S1: Comparison of query site size and global fold similarity (max. TM-Score) on the single mutation benchmark data set.



(a) Example of a false positive hit for a small query site at a terminus. The query is a rat cytosolic PEPCK (3DT4, tan), reference residue of the micro-environment is His6 in chain A (marked in green). The hit structure is an aminopeptidase from *E. coli* (1A16, blue) where Leu121 (chain A) is matched to the queries reference residue. Max. TM-Score=0.31, local  $C\alpha$  RMSD=2.35.

(b) Example of a false positive hit for a small query site at a chain break. The query is a human EGFR kinase domain (2ITY, tan), reference residue of the micro-environment is Ala864 in chain A (marked in green). The hit structure is an alcohol dehydrogenase from *Pyrobaculum aerophilum* (4JBG, blue) where Gly189 (chain A) matches the queries reference residue. Max. TM-Score=0.27, local  $C\alpha$  RMSD=1.79.

Figure S2: Examples of false positive random hits.



(a) Example of a locally similar but globally dissimilar hit. The query is a human Aldose Reductase (1PWM, tan), reference residue of the micro-environment is Ser127 in chain A (marked in green). The hit structure is an OLD nuclease from *Thermus Scotoductus* (6P74, blue) where Gly111 (chain A) matches the queries reference residue. Max. TM-Score=0.31, local C $\alpha$  RMSD=0.38. The structures share a locally similar loop.

(b) Example of a locally similar but globally dissimilar hit. Query is a human DNA-binding domain of human p53 (2XWR, tan), reference residue of micro-environment is Arg290 in chain B (marked in green, residue is resolved incompletely). The hit structure is a putative enoyl CoA hydratase/isomerase (crotonase) from *Legionella pneumophila* (3I47, blue) where Glu83 (chain A) is matched to the queries reference residue. Max. TM-Score=0.28, local C $\alpha$  RMSD=0.11. The hit structure shows a helix that extends the terminal helix in the query.

Figure S3: Examples of locally similar but globally dissimilar hits potentially representing structural or functional motifs.



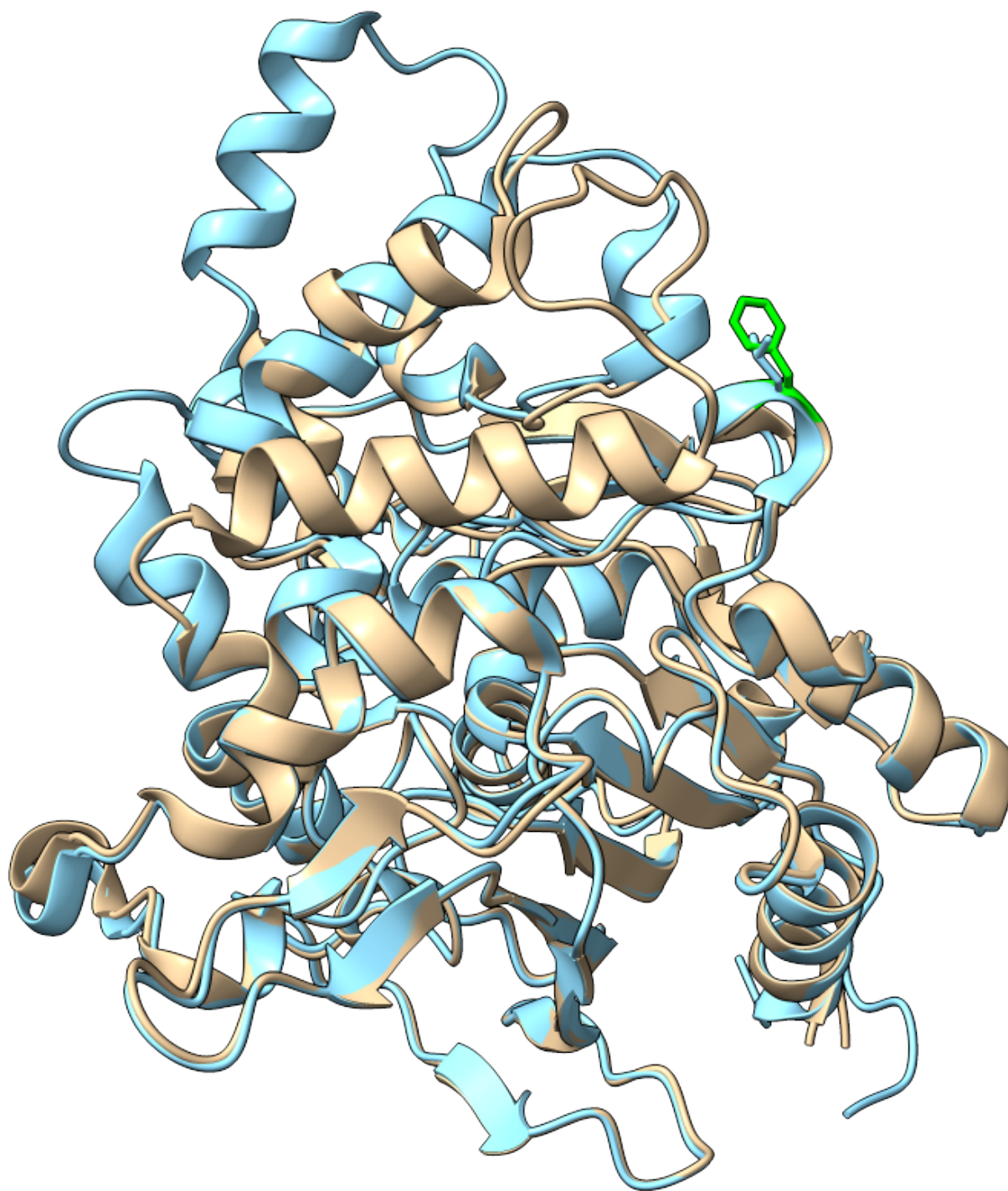


Figure S4: Example of a hit with high local RMSD due to larger structural rearrangements of the whole structure. The example shows a lipase from a *Geobacillus* strain (4FMP, tan), the reference residue of the micro-environment is Phe25 in chain A (marked in green). The hit structure (5CE5, blue) shows a mutation to Leu26 (chain A). The max. TM-Score is 0.90 but the local C $\alpha$  RMSD is 5.8. Superposition was calculated with ChimeraX.<sup>S4</sup>

## References

- (S1) Ukkonen, E. Approximate string-matching with q-grams and maximal matches. *Theoretical Computer Science* **1992**, *92*, 191–211.
- (S2) Shanthirabalan, S.; Chomilier, J.; Carpentier, M. Structural effects of point mutations in proteins. *Proteins: Structure, Function and Bioinformatics* **2018**, *86*, 853–867.
- (S3) Zhang, Y.; Skolnick, J. TM-align: A protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research* **2005**, *33*, 2302–2309.
- (S4) Pettersen, E. F.; Goddard, T. D.; Huang, C. C.; Meng, E. C.; Couch, G. S.; Croll, T. I.; Morris, J. H.; Ferrin, T. E. UCSF ChimeraX: Structure visualization for researchers, educators, and developers. *Protein Science* **2021**, *30*, 70–82.

## **D.5 ProteinsPlus: a comprehensive collection of web-based molecular modeling tools**

- [D5] K. Schöning-Stierand, K. Diedrich, C. Ehrt, F. Flachsenberg, J. Graef, **J. Sieg**, P. Penner, M. Poppinga, A. Ungethüm, and M. Rarey. “Proteins Plus: a comprehensive collection of web-based molecular modeling tools”. In: *Nucleic Acids Research* 50.W1 (2022), W611–W615.

Available: <https://doi.org/10.1093/nar/gkac305>. Material from [D5].

# ProteinsPlus: a comprehensive collection of web-based molecular modeling tools

Katrin Schöning-Stierand<sup>1,†</sup>, Konrad Diedrich<sup>1,†</sup>, Christiane Ehrh<sup>1,†</sup>,  
Florian Flachsenberg<sup>1,†</sup>, Joel Graef<sup>1,†</sup>, Jochen Sieg<sup>1,†</sup>, Patrick Penner<sup>1</sup>,  
Martin Poppinga<sup>1,2</sup>, Annett Ungethüm<sup>3</sup> and Matthias Rarey<sup>1,\*</sup>

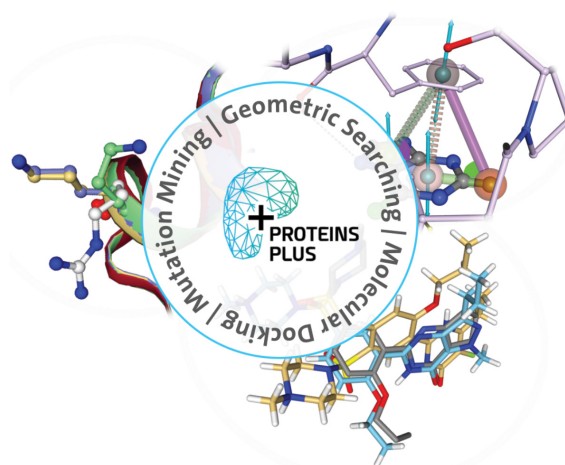
<sup>1</sup>Universität Hamburg, ZBH - Center for Bioinformatics, Bundesstraße 43, 20146 Hamburg, Germany, <sup>2</sup>Universität Hamburg, Department of Informatics, Vogt-Kölln-Straße 30, 22527 Hamburg, Germany and <sup>3</sup>Universität Hamburg, Center for Data and Computing in Natural Sciences (CDCS), Notkestraße 11, 22607 Hamburg, Germany

Received February 26, 2022; Revised April 05, 2022; Editorial Decision April 10, 2022; Accepted April 19, 2022

## ABSTRACT

Upon the ever-increasing number of publicly available experimentally determined and predicted protein and nucleic acid structures, the demand for easy-to-use tools to investigate these structural models is higher than ever before. The ProteinsPlus web server (<https://proteins.plus>) comprises a growing collection of molecular modeling tools focusing on protein–ligand interactions. It enables quick access to structural investigations ranging from structure analytics and search methods to molecular docking. It is by now well-established in the community and constantly extended. The server gives easy access not only to experts but also to students and occasional users from the field of life sciences. Here, we describe its recently added new features and tools, beyond them a novel method for on-the-fly molecular docking and a search method for single-residue substitutions in local regions of a protein structure throughout the whole Protein Data Bank. Finally, we provide a glimpse into new avenues for the annotation of AlphaFold structures which are directly accessible via a RESTful service on the ProteinsPlus web server.

## GRAPHICAL ABSTRACT



## INTRODUCTION

The ProteinsPlus (1,2) web server, openly available at <https://proteins.plus>, offers molecular modeling support for all protein structures that are publicly available as PDB files in the Protein Data Bank (PDB) (3). Usually, workflows for structure-based design necessitate a comprehensive user knowledge of different molecular modeling tools. For example, predicting potential binding sites, finding similar binding sites for ensemble docking, and molecular docking of small molecules of interest into a binding site requires access to and knowledge of a high number of tools with a multitude of parameters. Furthermore, researchers must rely on their computational resources. With the ProteinsPlus server, these shortcomings are overcome by enabling users to perform all these steps via one unique and easily accessible interface. The server is under constant development including

\*To whom correspondence should be addressed. Tel: +49 40 428387350; Fax: +49 40 428387352; Email: [matthias.rarey@uni-hamburg.de](mailto:matthias.rarey@uni-hamburg.de)

<sup>†</sup>The authors wish it to be known that, in their opinion, the first six authors should be regarded as Joint First Authors.

Present address: Florian Flachsenberg, BioSolveIT GmbH, An der Ziegelei 79, 53757 St. Augustin, Germany.

© The Author(s) 2022. Published by Oxford University Press on behalf of Nucleic Acids Research.

This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial License

(<http://creativecommons.org/licenses/by-nc/4.0/>), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited. For commercial re-use, please contact [journals.permissions@oup.com](mailto:journals.permissions@oup.com)

fine-tuning, feature extensions, and the integration of additional modeling tools.

Here, we offer insights into feature extensions for the structural multi-purpose comparison tool GeoMine, the newly integrated molecular docking tool JAMDA and MicroMiner - a method that can be used to screen for single-residue substitutions in local protein environments in the whole PDB.

Finally, the artificial intelligence-based protein structure predictions by AlphaFold (currently predicted by AlphaFold Monomer v2.0) enable unprecedented access to high-quality models of proteins of yet unknown structure (4). These models are now readily accessible via the AlphaFold Protein Structure Database (<https://alphafold.ebi.ac.uk/>) and can be directly imported via the provided REST API.

## MATERIAL AND METHODS: EXTENSIONS AND NOVEL TOOLS

### GeoMine

From the analysis of binding sites to investigations of geometric preferences for interactions, the ever-increasing number of molecular structures in the PDB offers a multitude of possibilities for in-depth studies of binding sites, their properties and their similarities. This requires comprehensive search capabilities. With GeoMine (5,6), we have developed a search engine that allows for the generation of and the search for atom-based geometric query patterns and an extensive textual and numerical filtering of the PDB. The query atoms can be described manually or automatically with varying degrees of detail, from major properties like the corresponding molecule type, i.e. nucleic acid, protein, ligand, water, or metal, to more restrictive ones, e.g. the molecular surface contribution of a protein or nucleic acid atom. Further feature points like aromatic ring centers can be added to the query and described equally. Distance ranges or hydrogen bond, pi-pi, pi stacking, pi-cation, metal and ionic interactions between atoms and feature points can be introduced into the query, and angle ranges between those can be specified. With the combination of all these features, almost any 3D pattern can be designed and searched in the entire PDB.

In the *ProteinsPlus* user interface, the query can be created in a 3D viewer from scratch by the placement of new atoms and feature points or by selecting those in a visualized binding site of a PDB/AlphaFold structure or any uploaded structure file. For this structure, GeoMine predicts binding pockets with interactions and hydrogen atoms using the tools DoGSiteScorer (7) and Protoss (8,9), respectively. If a ligand is present but no pocket has been calculated, a pocket is defined using a radius of 6.5 Å of any ligand atom. The computing times for the iterative search of over one million preprocessed bindings sites depends on the specificity of the query. Most requests can be processed in the range of minutes. For each detected hit, the root-mean-square deviation (RMSD) between the query and the part of the site matching the query is calculated enabling a ranking of the results by geometric fit. The 150 best results are listed in a table and can be visually inspected superimposed to the query in the NGL viewer. Different visualization options are available,

for example, choice of residues (complete pocket or only of the residues that match the query). The 150 best-matching pockets can be downloaded in PDB format together with a report containing the statistical overview of all results. The statistics report lists the PDB IDs and ligand names of all found pockets, the distributions of the RMSD values, and the properties of all matched atoms, feature points, distances, interactions, and angles of the query, e.g. the functional group distribution for a matched ligand atom. The user interface with a query history allows a continuous refinement of the results providing an interactive workflow of query modification and subsequent searching in the results. With this tool, protein function or ligand off-targets can be discovered by searching similar binding site properties in 3D space. GeoMine has recently been applied for a detailed analysis of structural features in protein kinase structures (5).

### JAMDA

Protein-ligand docking is one of the core tasks in structure-based drug design. With JAMDA, we aimed for the implementation of a fully-automated docking workflow in the *ProteinsPlus* server that does not only provide the actual docking algorithm but also encompasses all necessary preprocessing steps, including protonation state assignment and calculation of hydrogen coordinates for the protein (8), prediction of protonation and tautomeric states of the molecules to be docked (10), as well as the generation of 3D coordinates/conformations (11). While a certain degree of manual intervention is possible, our goal was to provide a fully automated workflow with optimized default parameters. This enables even less experienced users to derive potential binding modes of small molecules in the binding site of interest. From the analysis of structure-activity relationships to the test of new binding hypotheses, the established pipeline offers unlimited access to predicted binding modes.

JAMDA docking combines the TriX docking algorithm (12,13) for initial pose generation with the JAMDA scoring function (14), and our novel LSL-BFGS optimization algorithm (14,15) for scoring and pose optimization. Initially, conformers for the molecule to be docked are generated with the Conformer (11). The raw poses are subjected to a scoring and optimization cascade using the JAMDA scoring function to refine and rank the docking poses.

On *ProteinsPlus*, JAMDA allows for a fully automated docking: Only the protein, the binding site, and the molecules to be docked must be provided by the user. The binding site can be defined based on a known ligand or selected from the pocket definitions in *ProteinsPlus* (1) (e.g. predicted by DoGSiteScorer (16)). To enable the user to manually adjust the binding sites, all ligand-based and predicted binding sites which do not originate from GeoMine are editable by the user in the pockets tab by clicking on the pencil symbol of the pocket of interest in the upper right corner. Neither the protein nor the molecules to be docked must be manually prepared by the user because this is an integral part of the JAMDA docking workflow: The protein is prepared by assigning likely protonation states using Protoss (8). Furthermore, only structurally relevant water molecules and small molecules that are common cofac-



tors are kept. The molecules to be docked can be provided by picking a ligand from the NGL viewer for redocking studies or by uploading molecules in any common molecular file format (including SMILES without coordinates). Their predominant protonation and tautomeric states are predicted with UNICON (10) prior to docking. Most of these preprocessing steps can optionally be customized by the user.

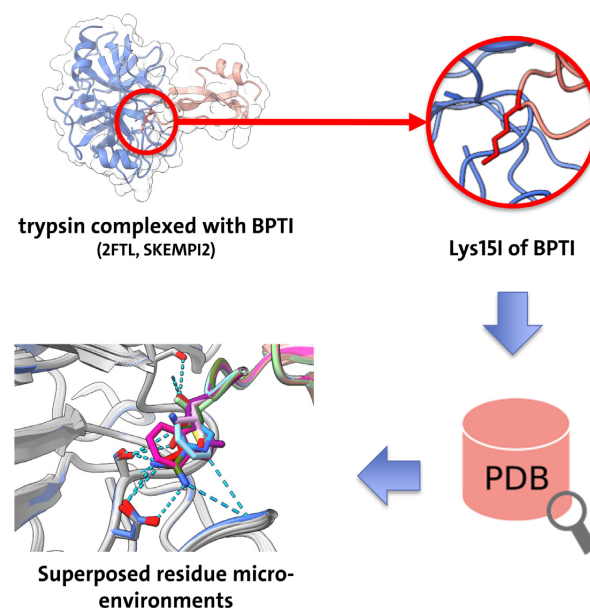
The preprocessing and docking are performed on the server and, currently, up to five molecules may be docked simultaneously. In the *ProteinsPlus* web interface, the resulting docking poses are shown in a table (with JAMDA score and the RMSD if a redocking was performed) and visualized in the NGL viewer panel for interactive analyses. They can also be downloaded for alternative visualizations and further processing. In consequence, JAMDA offers a pipeline for molecular docking that provides reliable results even in the absence of substantial knowledge regarding molecular modeling tools.

### MicroMiner

MicroMiner searches for mutations in protein structure databases. On *ProteinsPlus*, it screens for single-residue substitutions in the experimental structures of the entire PDB. Retrieved mutant structures can be easily analyzed and compared to the wildtype through automatically generated superpositions in the NGL viewer. The tool focuses on the local 3D micro-environment of single residues in a query protein. It searches the protein structure database for similar local environments with a mutated central residue. For reasonably large wildtype protein structures it is feasible to search for substitutions of all residues in the query at once. In this way, a user can comprehensively explore the wealth of experimental protein structures that exemplify the local effects of mutations through the interactive web interface.

MicroMiner originates from the ASCONA (17) and SIENA (18) technology for binding site similarity search and ensemble compilation. However, instead of focusing on the protein environment of ligands, MicroMiner uses the local 3D micro-environment of any individual residue as the query to search for residues embedded in similar local arrangements. A database search starts by selecting a query residue from which the local 3D protein neighborhood within a distance cutoff (default 6.5 Å) represents the query micro-environment. The connected sequence fragments of this environment are used to identify candidate protein structures with similar sequence fragments in the database. Second, all potential matches are identified by residue-wise sequence alignments. A subsequent fuzzy geometric filter based on the C $\alpha$  atom orientation and distances of the matching sequence fragments ensures a reasonably similar structural arrangement while tolerating structural changes upon mutation. Thus, we identify local micro-environments with a high sequence and structural similarity. Figure 1 shows the MicroMiner workflow.

Within the *ProteinsPlus* server, the user can select single residues of interest or all residues in the input structure to be searched against the PDB. Searching for all residues is feasible within one minute or less on average, depending on the size of the input protein and the number of similar



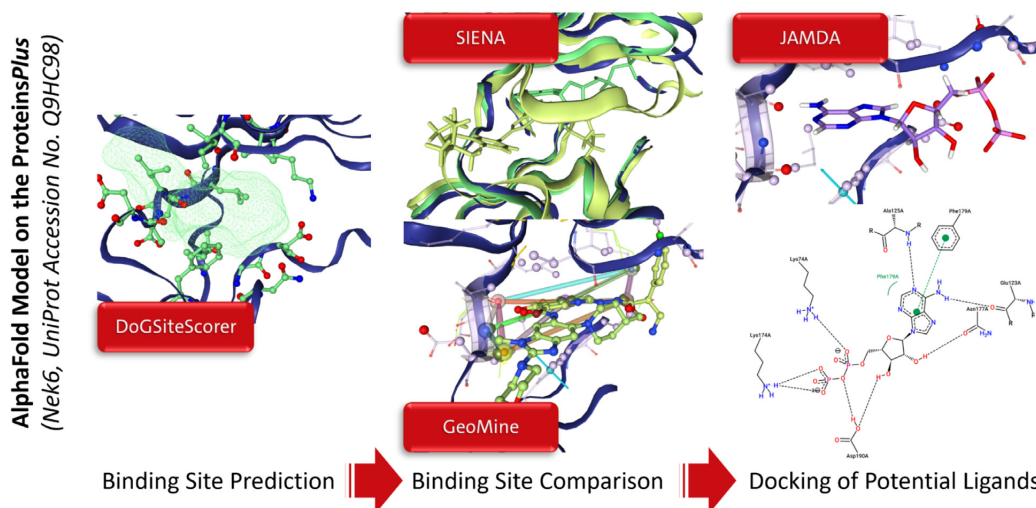
**Figure 1.** MicroMiner workflow. With the local 3D micro-environment of a selected query residue, the PDB is searched. Structures from the database containing a similar micro-environment identical in sequence except for the query residue position are retrieved and superposed for analysis. In this way, MicroMiner yields structure ensembles exemplifying the local effects of mutations.

micro-environments in the PDB. The protein structures of retrieved micro-environments can be explored interactively as a structure ensemble in the 3D viewer and sorted by properties of interest, for example, the RMSD of the local environments to investigate the structural effects of mutations. Further applications are the search for highly conserved regions in protein structures, comparisons of the impact of conservative and radical substitutions, or the investigation of structural effects upon substitution for evaluating the reliability and accuracy of computationally generated models of single-residue substitutions.

### Integration of AlphaFold structures

The inclusion of AlphaFold protein structure models (4) (<https://alphafold.ebi.ac.uk/>) in the *ProteinsPlus* web server enables easy access to machine learning-based predictions of previously unknown structures. The models are accessible on our web server by entering the UniProt Accession Number on the landing page or uploading a preprocessed structure. The user can analyze these structures in the same way publicly available PDB structures can be analyzed by making use of all applicable capabilities of the *ProteinsPlus* tools.

Besides the structural uncertainty of AlphaFold structures (19), the missing ligand annotations are a major drawback. This led to the development of the database AlphaFill (20) which annotates the 3D models with cofactors and metal ions and transfers them into the structure assisting in the functional annotation of the models. However, this annotation procedure was only followed for structures that show an identity of at least 35% to known 3D structures



**Figure 2.** This workflow shows exemplary results for structural investigations of the AlphaFold model for the Nek6 (UniProt Accession Number Q9HC98). First, the user can detect druggable binding sites with DoGSiteScorer. Pocket ‘P\_2’ which was predicted as druggable is depicted in green on the right. Next, the pocket can be used for a SIENA search for similar binding sites. Shown are two matches from this analysis with Nek7 structures: 2WQN with ADP and 6S73 in complex with the ligand with the ID F9N in the PDB. GeoMine can be applied for more specific user-defined searches in the binding sites of the PDB. Using a geometric query annotating solvent-exposed potentially interacting atoms and their distances, we found 116 pockets with a similar geometry in the PDB (e.g. cAMP-dependent protein kinase A with the PDB ID 7BAQ, PDB ligand ID T82 or interleukin-1 receptor-associated kinase 4 with the PDB ID 6O94, PDB ligand ID LRS). The corresponding query can be found in the Supplementary Data for upload to the GeoMine tool on the ProteinsPlus for this structure. Interesting small molecules from the identified similar sites can be downloaded and subsequently be used for molecular docking with JAMDA. The figures on the right show the second highest-scoring predicted binding mode for ADP in the binding site of Nek6 and its 2D interaction visualization with PoseView (21).

stored in the PDB and restricted to common cofactors and ions with potentially functional roles. For researchers interested in the structural annotation of structures that have no known homologs in the PDB, the ProteinsPlus web service comes in handy. It enables on-the-fly prediction of binding sites with DoGSiteScorer, retrieval of similar binding sites with SIENA, the identification of further potentially interesting ligands by user-defined GeoMine queries, and the molecular docking of these ligands into the AlphaFold model with JAMDA, see Figure 2.

### Ligand annotation for AlphaFold models

Given a protein of interest, e.g. the human protein kinase NIMA-related kinase 6 (Nek6), we can start our ProteinsPlus investigations by providing its UniProt Accession Number Q9HC98 and entering the structural analysis mode of the web service. Next, we can predict potential binding sites using DoGSiteScorer. These predicted sites can be used to search for potential ligands with SIENA. By selecting, for example, the pocket named ‘P\_2’ and performing a SIENA search for this predicted binding site, we can retrieve similar sites in complex with various ligands. Besides ADP (the annotation which was also found by AlphaFill), we find similar kinase binding sites in complex with further ligands, in this case, the inhibitor with the PDB ligand ID F9N in complex with Nek2 and Nek7. The active site sequence identity is 94%. The retrieved aligned complexes can be downloaded, together with the corresponding ligand SDF files. The results also enable the exploration of structural flexibility of similar binding sites that can be used, e.g. for the generation of other conformational states that are not covered

in the AlphaFold database by homology modeling based on the identified structures.

The ligands retrieved from the SIENA run can either be transferred into the binding site based on the resulting alignment or using the on-the-fly docking tool JAMDA. It can be applied to find whether the found ligands from similar sites can be accommodated in the model’s binding site. However, care should be taken regarding the model quality of the binding site residues as this can have a huge impact on the docking performance. Some preprocessing steps of the original AlphaFold structure might be necessary to obtain reliable ligand binding modes (22).

The search for similar binding sites using the ProteinsPlus, however, is not restricted to binding sites with a high sequence identity. GeoMine can be applied to generate user-defined queries that search for geometric patterns of interacting binding site residues in nearly one million binding sites (predicted or ligand-annotated) in the PDB. For our example protein kinase, additional GeoMine queries result in the identification of further protein kinases in complex with inhibitors which can be used as idea generators for *in silico* drug design.

### SUMMARY AND OUTLOOK

The ProteinsPlus web server offers a unique access point to protein structure and protein–ligand complex data processing on the worldwide web. Current developments with only conservative extensions of the user interface enable even broader access to molecular modeling tools which usually require comprehensive user knowledge. Furthermore, steady improvements and feature extensions based



on suggestions of users render it a lively and well-kept platform. To support users in getting started with the web server, we offer comprehensive documentation of the provided services (<https://proteins.plus/help/index>) and hands-on tutorials (<https://proteins.plus/help/tutorial>). As with all computational modeling approaches, the tools behind ProteinsPlus have their limitations. All users are asked to consult the corresponding methods' publication for more details on the respective restrictions and application domains.

Besides the introduction of new features for GeoMine and the integration of the novel methods JAMDA and MicroMiner, we are in a constant process of elaborating the web server, its tool base, and its potential use cases. The first inclusion of AlphaFold structures in the web server opens new avenues for structural explorations that have not yet been fully explored. With numerous extensions in mind, including 2D and automated query generation in GeoMine or multiple mutations search in MicroMiner, we hope to create a steadily growing, easy-to-use modeling infrastructure for the life science community.

## DATA AVAILABILITY

ProteinsPlus is a publicly available web-based protein structure analysis service, available at <https://proteins.plus>.

## SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

## FUNDING

Development of ProteinsPlus was supported by de.NBI (in part); German Federal Ministry of Education and Research (BMBF) [031L0105 to K.S. and J.S.]; Development of GeoMine was supported by the German Federal Ministry of Education and Research as part of CompLS and de.NBI [031L0172, 031L0105 to K.D. and J.G.]; Development of MicroMiner was supported by the German Federal Ministry of Education and Research (BMBF) as part of protPSI [031B0405B to J.S.]; DASHH: Data Science in Hamburg - Helmholtz Graduate School for the Structure of Matter, funded by the Helmholtz Association [HIDSS-0002 to C.E.]; Center for Data and Computing in Natural Sciences (CDCS), funded by Authority for Science, Research and Equality of the Free and Hanseatic City of Hamburg (BWFG) [LFF-HHX-03 to A.U.]. Funding for open access charge: Internal university funds.

*Conflict of interest statement.* ProteinsPlus uses some methods that are jointly owned and/or licensed to BioSolveIT GmbH, Germany, M.R. is a shareholder of BioSolveIT GmbH.

## REFERENCES

1. Schönig-Stierand, K., Diedrich, K., Fährrolfes, R., Flachsenberg, F., Meyder, A., Nittinger, E., Steinegger, R. and Rarey, M. (2020) ProteinsPlus: interactive analysis of protein–ligand binding interfaces. *Nucleic Acids Res.*, **48**, W48–W53.

2. Fährrolfes, R., Bietz, S., Flachsenberg, F., Meyder, A., Nittinger, E., Otto, T., Volkamer, A. and Rarey, M. (2017) Proteins plus: a web portal for structure analysis of macromolecules. *Nucleic Acids Res.*, **45**, W337–W343.
3. Bertram, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H. and Shindyalov, I.N. (2000) The protein data bank ([www.rcsb.org](http://www.rcsb.org)). *Nucleic Acids Res.*, **28**, 235–242.
4. Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Židek, A., Potapenko, A. *et al.* (2021) Highly accurate protein structure prediction with alphafold. *Nature*, **596**, 583–589.
5. Graef, J., Ehrt, C., Diedrich, K., Poppinga, M., Ritter, N. and Rarey, M. (2022) Searching geometric patterns in protein binding sites and their application to data mining in protein kinase structures. *J. Med. Chem.*, **65**, 1384–1395.
6. Diedrich, K., Graef, J., Schönig-Stierand, K. and Rarey, M. (2021) GeoMine: interactive pattern mining of protein–ligand interfaces in the protein data bank. *Bioinformatics*, **37**, 424–425.
7. Volkamer, A., Kuhn, D., Rippmann, F. and Rarey, M. (2012) DoGSiteScorer: a web server for automatic binding site prediction, analysis and druggability assessment. *Bioinformatics*, **28**, 2074–2075.
8. Bietz, S., Urbaczek, S., Schulz, B. and Rarey, M. (2014) Protoss: a holistic approach to predict tautomers and protonation states in protein–ligand complexes. *J. Cheminform.*, **6**, 12.
9. Lippert, T. and Rarey, M. (2009) Fast automated placement of polar hydrogen atoms in protein–ligand complexes. *J. Cheminform.*, **1**, 13.
10. Sommer, K., Friedrich, N.-O., Bietz, S., Hilbig, M., Inhester, T. and Rarey, M. (2016) UNICON: a powerful and Easy-to-Use compound library converter. *J. Chem. Inf. Model.*, **56**, 1105–1111.
11. Friedrich, N.-O., Flachsenberg, F., Meyder, A., Sommer, K., Kirchmair, J. and Rarey, M. (2019) Conformer: a novel method for the generation of conformer ensembles. *J. Chem. Inf. Model.*, **59**, 731–742.
12. Schlosser, J. and Rarey, M. (2009) Beyond the virtual screening paradigm: structure-based searching for new lead compounds. *J. Chem. Inf. Model.*, **49**, 800–809.
13. Henzler, A.M., Urbaczek, S., Hilbig, M. and Rarey, M. (2014) An integrated approach to knowledge-driven structure-based virtual screening. *J. Comput. Aided. Mol. Des.*, **28**, 927–939.
14. Flachsenberg, F., Meyder, A., Sommer, K., Penner, P. and Rarey, M. (2020) A consistent scheme for gradient-based optimization of protein–ligand poses. *J. Chem. Inf. Model.*, **60**, 6502–6522.
15. Flachsenberg, F. and Rarey, M. (2021) LSOpt: an open-source implementation of the step-length controlled LSL-BFGS algorithm. *J. Comput. Chem.*, **42**, 1095–1100.
16. Volkamer, A., Griewel, A., Grombacher, T. and Rarey, M. (2010) Analyzing the topology of active sites: on the prediction of pockets and subpockets. *J. Chem. Inf. Model.*, **50**, 2041–2052.
17. Bietz, S. and Rarey, M. (2015) ASCONA: rapid detection and alignment of protein binding site conformations. *J. Chem. Inf. Model.*, **55**, 1747–1756.
18. Bietz, S. and Rarey, M. (2016) SIENA: efficient compilation of selective protein binding site ensembles. *J. Chem. Inf. Model.*, **56**, 248–259.
19. Perrakis, A. and Sixma, T.K. (2021) AI revolutions in biology. *EMBO Rep.*, **22**, e54046.
20. Hekkelman, M.L., de Vries, I., Joosten, R.P. and Perrakis, A. (2021) AlphaFill: enriching the alphafold models with ligands and co-factors. bioRxiv doi: <https://doi.org/10.1101/2021.11.26.470110>, 27 November 2021, preprint: not peer reviewed.
21. Stierand, K., Maass, P.C. and Rarey, M. (2006) Molecular complexes at a glance: automated generation of two-dimensional complex diagrams. *Bioinformatics*, **22**, 1710–1716.
22. Skolnick, J., Gao, M., Zhou, H. and Singh, S. (2021) AlphaFold 2: why it works and its implications for understanding the relationships of protein sequence, structure, and function. *J. Chem. Inf. Model.*, **61**, 4827–4831.

# Eidesstattliche Versicherung

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Dissertationsschrift selbst verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Hamburg, den December 14, 2023



Jochen Sieg