Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

UKE HAMBURG

Identification of Peptides in Mass Spectrometric Proteomics Data with the PRIDE Cluster Spectral Library and a Neural-net-based Machine-learned Scoring Function

by

Marcus Wurlitzer, M. Sc.

Dissertation

for the acquisition of the academic degree Dr. rer. nat.

at the

University of Hamburg
Faculty of Mathematics, Informatics and Natural Sciences
Department of Chemistry

and the

University Medical Center Hamburg-Eppendorf
Institute for Clinical Chemistry and Laboratory Medicine
Mass Spectrometric Proteomics Group

February 2019

Druckfreigabe: 17.05.2019

1. Gutachter: Prof. Dr. Hartmut Schlüter

2. Gutachter: Prof. Dr. Dr. Christian Betzel

Research for the present thesis was conducted in the Mass Spectrometric Proteomics Group of Prof. Dr. Hartmut Schlüter at the University Medical Center Hamburg-Eppendorf from March 2015 until February 2019.

I Table of contents

II Naming conventions and abbreviations

The following terms and abbreviations are being used in the specified sense throughout this thesis:

m/z = mass-to-charge ratio.

Th = Thomson, the unit of the mass-to-charge ratio (Da/e).

PRIDE spectral library = Library of consensus spectra created from the results of the PRIDE Cluster project. The library was constructed from all 'complete' dataset of the PRIDE repository, which were condensed into a usable spectral library by the PRIDE team. In this work, the human subset of the PRIDE Cluster database is used, which will be referred to as the 'PRIDE spectral library' throughout this work.

Peptide species = A unique combination of peptide sequence, charge state and modifications.

Query peptide = A fragment spectrum of a peptide which is subjected to identification.

High confident peptide = A peptide that has been identified with an FDR threshold of 0.01 by an established search engine.

Very high confident peptide = A peptide that has been identified with an FDR threshold of 0.001 by an established search engine.

Reference or candidate peptide = A peptide from a spectral library with known identity (sequence).

Decoy peptide = A peptide from the spectral library that has been altered in a way to produce a negative hit.

Spectrum-spectrum match (SSM) = a pair of a query and a candidate spectrum which is subjected to a scoring function to measure the similarity between the two spectra.

(True) positive hit = a spectrum-spectrum match where the highest-scoring candidate represents the correct peptide sequence.

Delta score = Difference of the score of the best hit and the next best hit which represent a different peptide. A high delta score implies that the best hit matched the spectrum much better than any other candidate peptide.

False discovery rate (FDR) = The percentage of false-positives among all positives which is tolerated.

1 Zusammenfassung

Neue Möglichkeiten zur Analyse massenspektrometrischer Proteomdaten ergeben sich durch die immer größer werdende Zahl an Datensätzen von Proteomstudien, die durch Online-Datenbanken verfügbar werden. Mit Hilfe umfassender Spektrenbibliotheken, die aus den vielen Datensätzen generiert werden können, und modernen Methoden der Datenanalyse kann die Identifizierung von Peptide anhand von Spektrenbibliotheken (‚Spectral Library-Suche') eine effektive Alternative zur klassischen Sequenzdatenbanksuche werden. In dieser Arbeit soll die Spectral Library-Suche als eine Methode zur schnellen, zuverlässigen und sensitiven Identifizierung von Peptidespektren weiterentwickelt werden.

Die humane ‚PRIDE Cluster'-Spektrenbibliothek umfasst 789,745 Spektren von 189,400 Peptiden und deckt damit 25.5% der tryptischen Peptide im bekannten menschlichen Proteom ab. Sie wurde für die Etablierung der Spectral Library-Suchmaschine genutzt.

Die Suche nach passenden Vorläuferionen (‚Precursor') wurde mit sehr kleiner Toleranz zu den rekalibrierten Masse-zu-Ladungs-Werten der Precursor in der Spektrenbibliothek durchgeführt. Die m/z-Werte der Fragmentspektren wurden zunächst mit einer empirisch ermittelten Rekalibrierungsfunktion rekalibriert und anschließend in Bins von 1 Th Breite vektorisiert. Mehrere Methoden zur Transformation der Intensitäten der Fragmentsignale und zum Scoring der Spektrenpaare wurden mit Hinblick auf deren Fähigkeit zur Unterscheidung von korrekten und falschen Spektrenpaaren getestet. Die Rank-Transformierung der 150 intensivsten Signale in Kombination mit der ‚correlation similarity' Scoring-Funktion erzielte die besten Ergebnisse.

Die Generierung von Decoy-Spektren wurden mit drei verschiedenen Methoden getestet. Die *Precursor-shuffle*-Methode erzeugte die besten Decoy-Spektren. Anders als bei der *intensity shuffle*- und der *m/z randomization*-Methode werden hierbei nicht die Spektren selbst verändert, sondern die Precursor-Masse-zu-Ladungs-Werte modifiziert, so dass effektiv Spektren anderer Peptide mit ähnlichen Precursor-Masse-zu-Ladungs-Werten als Decoy-Spektren verwendet werden. Die so generierten Decoy-Spektren erhielten sehr ähnliche Scores wie die Zufallstreffer, welche sie modellieren sollen. Sie wurden daher bei der anschließenden Validierung zur Abschätzung der Falsch-Positiv-Rate genutzt.

Um das Scoren von Spektrenpaaren und damit die Identifizierungsrate weiter zu verbessern, wurde maschinelles Lernen eingesetzt. Ein neuronales Netzwerk ersetzte dabei die Scoring-Funktion (‚Scoring-Netz'). Ein weiteres neuronales Netzwerk diente als Trainings-Netz für das Scoring-Netz. Letzteres erlernte dabei zwei Vektoren mit Gewichten zur Etablierung einer gewichteten Scoring-Funktion (‚weighted correlation similarity', WCS). Das WCS-Scoring erzielte eine Verbesserung der Scores um 24.3% und der Identifizierungsrate nach Validierung um 6.9% bzw. 14.0% in den beiden HeLa-Datensätzen. Die Vektoren selbst lassen Rückschlüsse auf die Unterscheidungsgewalt von Fragmentsignalen an einzelnen m/z-Positionen zu.

Die WCS-Suchmaschine erreichte Übereinstimmungen von über 98% mit der Sequenzdatenbanksuche für Peptide, die in der Spektrenbibliothek zu finden waren. Nach

Validierung durch konservative globale Abschätzung der Falsch-Positiv-Rate konnten 45% der Identifizierungstreffer bestätigt werden. Weitere 5% wurden nur durch die Spektrenbibliotheksuche gefunden. Die Zahl der validierten Peptid-Treffer war geringer als mit der klassischen Sequenzdatenbanksuche, allerdings konnte die Falsch-Positiv-Rate mit großer Sicherheit auf 1% begrenzt werden, da diese Methode nicht der Problematik des Überanpassens unterliegt. Die hier entwickelte WCS-Suchmaschine produzierte Identifizierungsergebnisse mit hoher Sicherheit an Hand der PRIDE Cluster-Spektrenbibliothek und erreichte höhere Identifizierungsraten als die etablierte Suchmaschine SpectraST.

2 Abstract

Mass spectrometric proteomics data analysis can break new ground through the growing amount of data from proteomics studies that become publicly available in online repositories. By exploitation of the large-scale spectral libraries built from these repositories and application of state-of-the-art computational methods, spectral library searching can become a powerful alternative to conventional sequence database searching. The present work aims to advance spectral library searching as a fast, reliable and sensitive method for the identification of spectra from mass spectrometric proteomics data.

The PRIDE Cluster human spectral library, containing 789,745 spectra of 189,400 peptides, covering 25.5% of the human tryptic peptide sequences, was used to develop a spectral library search engine for the identification of peptides in proteomics datasets.

Precursor matching was performed in a narrow m/z range against the recalibrated precursor mass-to-charge ratios. Fragment spectra were recalibrated with an empirical recalibration function before vectorization into bins of 1 Th. Various methods of intensity transformation and scoring were tested for their ability to discriminate true from false spectrum-spectrum matches. The rank transformation of the top 150 signals combined with the 'correlation similarity' scoring function performed best.

Decoy spectra were generated with three different methods. The *precursor shuffle* was found to produce the best decoys. Unlike the *intensity shuffle* and *m/z randomization* methods, it does not rely on the manipulation of target spectra. Instead, it modifies precursor information in a way that effectively spectra from different peptides with similar precursor m/z values are presented as decoys. The decoy spectra produced by this method achieved very similar scores to the random hits and were used for hit validation and global false discovery rate (FDR) estimation.

A machine learning procedure was established to improve the scoring of spectrum-spectrum matches and hence the identification rate. A neural net was designed to fully replace the spectrum-spectrum scoring function ('scoring net'). Another neural net was implemented to train the scoring net by taking all candidate spectra for a query spectrum as input ('training net'). The scoring net learned two weight vectors that were used to create a 'weighted correlation similarity' (WCS) scoring function. The WCS function improved the spectrum scores by 24.3% and the identification rate after validation by 6.9% and 14% in the two HeLa datasets. The weight vectors themselves gave interesting insight on the discriminative power of signals at every m/z position for spectrum-spectrum matching.

The WCS search engine achieved an overall agreement in identifications with conventional sequence database searching of over 98% for the peptides present in the library. After validation of the hits by a conservative global false discovery estimation, 45% of the sequence database identifications were confirmed, and another 5% of additional peptide identifications were retrieved. While the number of validated peptide hits was lower than for sequence database search, the conservative method of hit validation with global FDR estimation strictly controlled the FDR at 1% without proneness to overfitting. The WCS search engine developed in this work

yielded high quality identification results with the help of the PRIDE Cluster spectral library and achieved higher identification rates than the well-established spectral search engine SpectraST.

3 Introduction

3.1 Mass spectrometry of proteins

Mass spectrometry (MS) is an analytic method which separates and detects charged particles by their mass-to-charge ratio. It has become a widely used tool for bioanalytics and the primary method for protein identification in biological samples. Technological advances in ionization and mass analyzers enabled for high-throughput analyses and established the field of mass spectrometric proteomics, the comprehensive analysis of all proteins from a sample [1].

The mass spectrometric analysis of proteins can be divided into two basic strategies: *top-down* and *bottom-up*. With the *top-down* approach, intact proteins are subjected to mass spectrometric analysis [2]. Proteins are biological macromolecules with molecular weights ranging from 10 kDa to several hundreds of kilodaltons. Some proteins have molecular weights of more than 1 megadalton, like Titin, the largest protein in the human proteome, with 3.6 MDa. These large analytes are much harder to analyze efficiently than smaller molecules, as they produce a large number of different charge states and may come in a large variety of possible isoforms and chemical modifications. Also, mass analyzers may not be able to resolve the signals from large proteins down to the isotopic peaks for their limited resolution and/or mass range, and separation of intact proteins prior to mass spectrometry presents a major challenge [3]. As a result, a mixture of proteins can produce extremely complex spectra that are very hard to interpret. *Top-down* protein analysis is therefore reserved for specific analytic tasks where simpler methods are not sufficient.

The *bottom-up* approach is most widely used in the field of proteomics. *Bottom-up* proteomics involves enzymatic cleavage of the proteins into peptides, which are then subjected to a separation step and to mass spectrometric analysis. Peptides typically have molecular weights between 500 Da and 5 kDa. Many of them are well separable by liquid chromatography and ionize readily into a small number of different charge states upon mass spectrometry [4]. The range of possible modifications is limited to few for the majority of peptides, which facilitates identification of the peptides from the acquired spectra. Once identified, the peptides are mapped to the protein they originated from with the help of a protein database.

The present work focuses on the analysis of *bottom-up* proteomics datasets, specifically on the identification of peptides by a spectral library. The following sections relate to *bottom-up* protein analysis.

3.1.1 Separation by liquid chromatography

Enzymatic cleavage of the proteins in *bottom-up* analyses produces a complex mixture of tens of thousands of peptides. Mass spectrometric acquisition of too many analytes at the same time results in charge suppression, i. e. only those analytes with good desorption and ionization properties are ionized effectively, while others will not be ionized and hence not be detected by the mass analyzer. In addition, many peptides share the same or similar molecular weights. When

one of those peptides is selected for fragmentation, several unrelated peptide ions may pass the mass filter and the resulting fragment spectra will be a mixture of signals from two or more peptides, rendering a reliable identification difficult or even impossible [4].

Peptides in *bottom-up* proteomics are therefore subjected to at least one step of separation. The longer the separation procedure, the higher the possible resolution of separation that can be achieved [5]. Multiple steps of separation can be combined to achieve a higher dimensional separation and even higher resolution, at the expense of increased experimental efforts and longer acquisition times [6, 7]. The last (or only) step is most commonly a reversed-phase liquid chromatography (RP-LC), which separates peptides by hydrophobic interaction with a column matrix. The RP-LC provides good separation efficacy and is suitable for direct coupling with an electrospray ionization source of mass spectrometer (Online-LC-MS) [4].

3.1.2 Mass spectrometric instrumentation

Mass spectrometry analysis of substances involves three basics steps: a) ionization of analytes and transition to the gas phase, b) separation of ionized analytes by their mass-to-charge ratio, and c) detection of the separated ions. Combination of multiple mass analyzers and fragmentation cells allows for an isolation of a specific ion species, its fragmentation, and the analysis of the resulting fragment ions. These ‘tandem mass spectrometers’ are most widely used for proteomics experiments.

3.1.2.1 Ionization

All mass analyzers known to date can only separate charged gas-phase particles. An ion source is therefore necessary to (1) bring the molecules into the gas phase, termed 'desorption', and (2) make them carry a charge, termed 'ionization'. Two ion sources are predominantly used for desorption and ionization of biomolecules: matrix-assisted laser desorption/ionization (MALDI) and electrospray ionization (ESI). Both ionization methods are often termed *soft*, as they deposit low amounts of energy onto the molecules and minimize their in-source fragmentation, which occur otherwise when ionizing biological macromolecules with higher energies [8]. In this work, all datasets were acquired with online-LC-ESI mass spectrometry.

Electrospray ionization

Electrospray ionization (ESI) is the most frequently used soft-ionization method for biological macromolecules [9]. ESI allows for the formation of gas-phase ions from liquid samples. Initial electrospray experiments were conducted by Dole in the 1960s, while ESI was first applied to peptides and proteins by Fenn et al. in the 1980s [10].

By applying a voltage to the solution, charged droplets are created at the tip of the ESI needle. Subsequently, droplets shrink as the solvent evaporates and undergo a Coulomb explosion due to increasing charge density. Once all solvent molecules have evaporated, a single charged molecule is left in the gas-phase.

3.1.2.2 Mass analysis and detection

Once ions have been generated in the ion source, they are guided to the mass analyzer, where they are separated by their mass-to-charge (m/z) ratio. Mass analyzers vary in ion separation and detection principles and perform differently in terms of speed, dynamic range, available m/z range, as well as mass resolution and mass accuracy.

Mass resolution and mass accuracy are important characteristics of a mass analyzer. Mass resolution is defined as (m/z)/(Δm/z) where Δm/z is the minimal difference where a signal can still be distinguished from another signal of the mass-to-charge ratio m/z. Mass accuracy is defined as (Δm/z)/(m/z), where Δm/z is the difference between the m/z observed in the mass spectrum and the m/z value. It is usually given in parts per million (ppm).

The datasets in this work were acquired on two Orbitrap instruments which were equipped with three different types of mass analyzers: quadrupoles, an ion trap and Orbitraps.

Orbitrap

The concept of the Orbitrap mass analyzer dates back to the Kingdon trap published in 1923, but could not be developed into a functioning instrument at that time [11]. The history of the Orbitrap including the work of Kingdon and ensuing scientists was reviewed in 2008 by Perry et al. [12]. Realization of the Orbitrap concept towards a high-performance mass analyzer was facilitated by the work of Makarov [13], including the development of the C-trap as the injection device for the ion packets. Since their commercial introduction in 2005, Orbitrap instruments have become a very popular choice among proteomic facilities for their compact sizes, high resolution, and high speed analytic capabilities [14].

The Orbitrap is a Fourier-transform mass spectrometry (FTMS) mass analyzer, which detects the image current of ions oscillating between the two outer electrodes while rotating around the central electrode. The frequency of the axial oscillation of an ion is proportional to the inverse square root of its mass-to-charge ratio (m/z) [15]:

$$\omega = \sqrt{\frac{e}{(m/z)} \cdot k},$$

w = Axial frequency, e = elementary charge, k = device-specific field curvature constant

Commercial Orbitrap instrument achieve mass resolutions of 1,000,000 at m/z 200, which is the second-highest among all mass analyzers, only surpassed by FTICR instruments [16].

The two mass spectrometers that were used to acquire the experimental data in this study were the ‘Q Exactive’, featuring a regular Orbitrap mass analyzer, and the ‘Fusion’, equipped with a smaller ‘high-field Orbitrap’ with higher resolution [15].

Quadrupole

A quadrupole mass analyzer consists of four rod-shaped electrodes. For each opposing pair of electrodes, a voltage can be applied. A radio frequency signal is used to create a stable flight path for a very small window of m/z values, effectively letting only molecules with a specific mass-to-

charge ratio pass through, while all other molecules will take unstable flight paths and collide with the rods. The quadrupole functions as a mass filter and can be attached with an ion detector or combined with another mass analyzer for tandem mass spectrometry.

3.1.3 Mass spectrometric acquisition of peptides: precursor and fragment spectra

During the mass spectrometric acquisitions of peptides, a spectrum of all analytes eluting from the LC column is acquired (*full scan* or *MS(1) spectrum*). Every signal in the MS1 spectrum represents a molecule of a certain mass-to-charge ratio (m/z). The m/z values can be determined to a certain level of accuracy, which depends on the mass analyzer used. The mass spectrometer then selects a given number of signals (*precursors*) for fragmentation. For each selected precursor ion, a mass filter is adjusted to the precursor's mass-to-charge ratio and filters out all other ions. The selected precursor ion is fragmented by collision-induced dissociation (*CID*) or higher energy C-trap dissociation (*HCD*, only available on Orbitrap instruments) [17] using an inert collision gas, such as helium, argon, or nitrogen, to break the chemical bonds of the precursor. The resulting fragment ions are analyzed by the second mass analyzer to acquire the fragment spectrum (*MS/MS* or *MS2 spectrum*).

Orbitrap Q Exactive and Orbitrap Fusion

The datasets used in this work were acquired on an Orbitrap Q Exactive and an Orbitrap Fusion instrument. Both use a quadrupole as the first mass analyzer for filtering, HCD for fragmentation, and an Orbitrap for MS1 spectrum acquisition. The Q Exactive uses the same mass analyzer for high-resolution MS2 acquisition, while the Fusion uses an ion trap to acquire lower-resolution MS2 spectra. Orbitrap MS1 and ion trap MS2 acquisition can be conducted concurrently on the Orbitrap Fusion, allowing for a higher speed of fragment spectrum acquisition at the expense of lower MS2 resolution.

3.1.4 Peptide identification

A peptide in *bottom-up* datasets is identified from several pieces of information: a) the mass-to-charge ratio of the monoisotopic precursor and its higher isotopes, and b) the mass-to-charge ratios of the fragment ions.

Firstly, the charge (*z*) of the peptide is inferred from the m/z values of the isotopic distribution. The monoisotopic precursor mass (*m*) is then calculated from the monoisotopic m/z and the charge z.

Secondly, when peptides are fragmented by CID or HCD, the molecules usually break along the backbone chain. Fragments are generated in distances of amino acid residues (Figure 1).

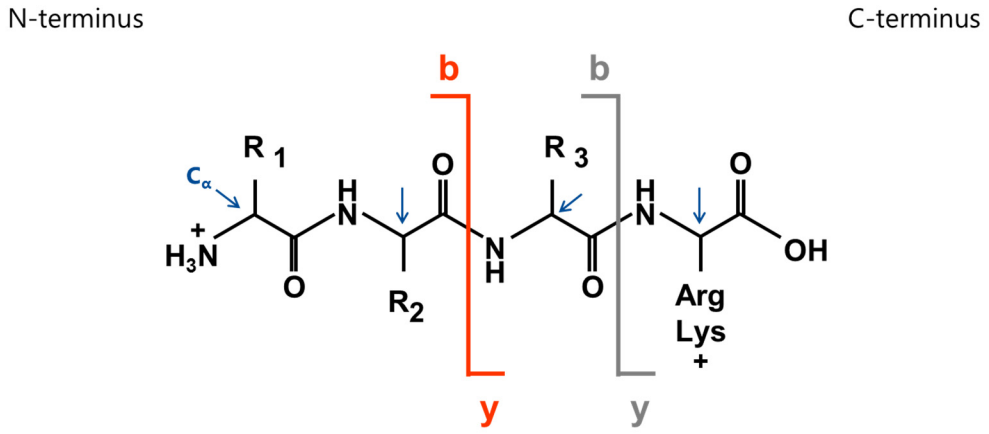

Figure 1: Fragmentation of an example peptide with four amino acids. R_1-R_3 indicate the variable residues of the amino acids, arrows indicate the C_α atoms. Peptides typically break at the peptide bond into an N-terminal part (*b ion*) and a C-terminal part (*y ion*) with CID or HCD fragmentation. The distance between two adjacent b or y ions equals the mass of the amino acid at that position minus water.

Every peptide produces characteristic fragment signals upon fragmentation. The most prevalent fragment ions for tryptic peptides are usually the y ions, since they are likely to carry a charge at their C-terminal amino acid (lysine or arginine for tryptic peptides). Other ions are frequently observed as well, including b ions, immonium ions, the unfragmented precursor ion, or neutral loss variants of the precursor or the side chains of amino acids such as glutamate, aspartate, glutamine and asparagine [18]. The fragment signals can be used to identify the analyzed peptide. Various methods for peptide identification exist, including direct sequencing of the amino acid chain without further reference (*de novo* sequencing), comparison of the fragment signals with theoretical fragments generated from a protein sequence database (sequence database searching), or comparison of the fragment signals with a reference spectra database (spectral library searching).

3.1.4.1 De novo sequencing

The amino acid sequence can be derived from the mass distances between adjacent fragments of the same type (like b ions or y ions). Assignment of an amino acid sequence to a fragment spectrum without any reference sequences is called *de novo sequencing*. *De novo* sequencing is not limited to the pool of existing sequences and may therefore be used where no reference sequences are available, e. g. when identifying peptides from unknown species or to determine the sequence of the variable region of an antibody [19]. However, without reference sequences, the search space is very large since all possible amino acid sequences must be considered. *De novo* sequencing requires fragment spectra of quality (signal-to-noise ratio) and purity levels (little contamination by other peptides) higher than database-driven sequencing in order to correctly identify a peptide. For lower quality spectra, various amino acid sequences might explain the fragment signal pattern equally well, and neither an algorithm nor a trained expert can determine the correct sequence [20].

3.1.4.2 Sequence database searching

Sequence database searching is the method of choice for high-throughput studies, identifying tens of thousands of peptides and thousands of proteins in a single LC-MS/MS run. For each query spectrum, a list of possible (candidate) peptides is extracted from the sequence database by matching of the experimental precursor mass. For every query-candidate pair, the fragment signals of the MS2 query spectrum are compared with the theoretical signals of the candidate sequence. The quality of the match is quantified with a score [20].

Protein sequence databases are assembled from genomic (DNA) sequencing and annotated with protein IDs and names. These databases contain all possible canonical amino acid sequences of the selected species. A commonly used protein database is SwissProt, a manually curated protein database and part of UniProt, which currently holds 559,077 protein entries, including 20,413 human proteins [21].

Popular sequence database search engines include SequestHT, Mascot, X!Tandem, Andromeda, and MS-GF+.

3.1.4.3 Spectral library searching

Spectral library searching is another strategy for peptide identification which has emerged during the past decade as the number of publicly available spectral libraries has been steadily increasing. It is based on the comparison of the query spectra with reference spectra from a library that have been repeatedly identified in previous experiments. Unlike theoretical spectra in database searching, the experimental spectra from the spectral library contain real intensity information and non-canonical fragment ions, two features which are usually not considered by sequence database search engines. As a result, spectral library search engines produce high identification rates and perform well also on lower quality spectra. [20]

An inherent property of spectral library searching is the limited search space. Only those peptides that have been analyzed, correctly identified, as well as submitted to and included in the library can be found by spectral library searching. This is somewhat a strength and a weakness at the same time: Since the search space includes only those peptides that are known to be detectable by mass spectrometry, the number of false hits from non-detectable peptides is reduced. Also, spectral library searching is comparably fast [20]. On the other hand, all peptide species which have not been included in the library will be missed or assigned with a false sequence. It is notable that each combination of peptide sequence, charge, and modifications forms a distinct peptide species, and each combination requires its own library entry in order to be identifiable in spectral library searching.

While the size of the library is the most limiting factor in spectral library searching, this will be mitigated by the growing number of publicly available proteomics data repository. At the time of writing, the PRIDE archive contained data from 11,299 proteomics experiments and continues to grow [22]. Also, open-modification search strategies may help to identify peptides where the exact modification configuration is missing the library [23].

3.2 Identification of peptide fragment spectra with spectral libraries

3.2.1 Spectral libraries built from online proteomics data repositories

Online repositories are a continuously growing source of proteomics data that can be condensed into comprehensive spectral libraries. Many scientific journals require submitting authors to make data from proteomics experiments publicly available through online repositories. Given the resources necessary to keep data in online repository available, the authors, members of the PRIDE team, state that "We must shift our focus from data review to data reuse." [24] The idea of data reuse has been demonstrated as part of the "Draft of the human proteome" by Wilhelm et al. in 2014 [25], where re-analyzed RAW files obtained from public repositories accounted for about 40% the data and hence significantly enhanced the authors' experimental data base.

Re-using online repository data for the generation of reference spectral libraries is not considered a straight-forward endeavor. The submitted data comes in a variety of data formats from different instruments, has been processed with different tools with different parameters, etc. The high heterogeneity is a major hurdle for researchers to re-using data from public repositories [26]. In addition, merging of multiple datasets into a consensus library requires not only sufficient computational resources, but also careful quality control to limit the number of false-positive peptide entries. Specifically, the authors of MassIVE-KB, a spectral library generated from the MassIVE repository, estimated that by simply merging all datasets which were individually filtered at 1% FDR, the overall FDR would grow as high as 28% [27].

The curators of the largest online repositories, PRIDE and MassIVE, both part of the ProteomeXchange consortium [28], created large spectral libraries from all data submitted to their respective repositories [24, 27]. The work involved development and evaluation of algorithms, quality control of the library, and realization through large-scale parallel computing.

The PRIDE Cluster spectral library was constructed from all 'complete' datasets uploaded to the PRIDE repository until April 2015. 256 million spectra (66 million identified, 190 million unidentified) were clustered and condensed into consensus spectra. A cluster was regarded reliably identified when 70% of the spectra, three at least, were identified as the same peptide sequence. The human subset of identified peptides consists of 789,745 consensus entries [29].

3.2.2 Introduction to spectral library searching

In 2006 and 2007, four spectral library search engines were published within a short time frame, including SpectraST [30], X!Hunter [31], BiblioSpec [32] and MSPepSearch [33]. These search engines are similar in that they all perform spectrum-spectrum comparison with the help of the dot product but differ in the details of spectrum processing and score calculation.

To identify a query spectrum with the help of a spectral library, the search engine first selects candidate spectra from the library with precursor masses within an adjustable window of the query spectrum's precursor mass (precursor mass tolerance) and (optionally) with the same

charge state. The query and candidate spectra may be pre-processed with different methods, like noise filters, intensity transformation and normalization. For every pair of query and candidate spectra (spectrum-spectrum match, SSM), the score is calculated as a measure of similarity between the two. After scoring, the SSMs are subjected to a validation step which separates true-positive from false-positive matches at a certain confidence (false-discovery rate, FDR).

3.2.2.1 Peptide modifications

Peptides can possess a variety of chemical modifications, both naturally and artificially. Every modification of a peptide which changes the chemical composition induces a shift of the precursor mass and of some or all fragment masses, depending on the position (the amino acid the modification is chemically bond to) and its fragmentation properties. For peptide identification, the information about present modifications is therefore as vital as the peptide sequence itself. Every combination of modifications of a peptide requires a distinct entry in spectral libraries to be available for identification.

Examples for modifications of peptides are a) carbamidomethylation, an artificially induced modification of cysteine during tryptic digestion as a protection from re-formation of disulfide bonds, b) oxidation as a result of endogenous processes or during sample preparation, c) TMT, iTRAQ or other heavy atom labels when labeled experiments are performed, d) biological post-translational modifications such as phosphorylation, acetylation and glycosylation.

3.2.2.2 Sequences and coverage

For the reasons given in the previous sections, 'coverage' in terms of spectral library searching extends beyond presence or absence of a peptide sequence in the library. Spectral libraries must include reference spectra for every desired combination of charge states and modifications in order to produce comprehensive identification for proteomics experiments. The number of reference spectra is therefore several folds larger than the number of distinct peptide sequences.

In the human subset of the PRIDE Cluster library, 789,745 consensus spectra represent 189,400 peptides. The MassIVE-KB spectral library is even larger, containing 2,035,808 spectra which were condensed from 227 proteomics datasets (31 TB of data) that had been uploaded to the MassIVE repository. The library covers 19,610 human proteins (97.4%) and 54% of the known human protein sequences [27]. Notably, the MassIVE-KB library also includes data from PRIDE which was mirrored to the MassIVE repository. They enhanced the coverage of their library by the inclusion of more than 100,000 spectra from synthetic peptides [34].

3.2.3 Precursor matching

Database search engines select candidate peptides from the database by matching their observed (or, optionally, re-calculated) mass to the experimental monoisotopic mass of the precursor ion and its charge state. A tolerance can be defined by the user, which is usually as low as 10 ppm, or even lower, for high-resolution mass analyzers such as the Orbitrap.

The precursor mass tolerance also determines the number of candidate spectra that have to be compared to each query spectrum. Higher precursor mass tolerance leads to a larger search

space and increases the chance of selecting the correct hit for scoring (higher sensitivity), but may potentially lead to more false-positive hits (lower specificity). Conversely, a narrow precursor mass window may result in higher specificity at the expense of lower sensitivity. Generally, the tolerance should be adjusted to the lowest possible value the database and the experimental dataset allows for without compromising sensitivity for specificity.

3.2.4 Fragment spectrum resolution and vectorization

In MS/MS acquisition mode, fragment signals are usually recorded along the m/z axis in a mass range of 100 up to the singly-charged precursor m/z. Some mass analyzers impose limits on the maximum mass range, or the mass range may be narrowed deliberately to save analysis time and/or disk space.

Mass analyzers acquire spectra as vectors of intensities ('profile spectra'). The mass range and the desired resolution determine the length of the vector. During preprocessing of the data for search engines, the profile spectra are subjected to a peak picker which reduces the continuous spectrum to a list of (m/z; intensity) pairs (peak list). Peak picking reduces the amount of data and effectively functions as a noise filter, where only peaks with proper shape and, if appropriate, proper isotopic distribution will pass through. Search engines usually take peak list spectra as input, e.g. in the mascot-generic format (mgf).

To compare a pair of query and candidate spectra in spectral library searching, most search engines transform the peak lists back into a common m/z vector. This allows for the use of vector operations like the dot product to calculate spectrum similarity.

An important decision in vectorization is the bin size, which determines the minimum distance between to signals that can be represented by the spectrum vectors. Small bins of 0.02 Th, for example, retain high fragment mass resolution and accuracy as found in Orbitrap MS2 data for instance, because signals that are as little as 0.02 Th apart will be assigned to different vector bins. However, to correctly compare signals in a query spectrum with the corresponding signals in a candidate spectrum, mass accuracy of both spectra must be equally high. The mass accuracy of query spectra is usually known to the user from the specifications and the quality of calibration of the mass spectrometer. In contrast, the diverse nature of spectra in the spectral library in terms of instrumentation and processing may not allow to assume such high mass accuracies in candidate spectra and require higher fragment mass tolerances. Also, a common m/z vector from 100 to 2000 Th with 0.02 Th bin size would require an enormous number of 95,050 data points per spectrum – most of which represent zeros or noise. This can be mitigated by the use of sparse arrays as memory-efficient storage constructs, but still requires processing of the full-size vectors when the scoring function is applied.

On the contrary, bin sizes of 1.0 Th yield shorter vectors and do not require very high mass accuracy in query and library spectra. But they will reduce the effective fragment mass resolution to 1.0 Th and hence perform the spectrum-spectrum comparison at a comparably high fragment mass tolerance of 1.0 Th. Figure 2 demonstrates how fragment signals are divided into bins of 1.0 Th width.

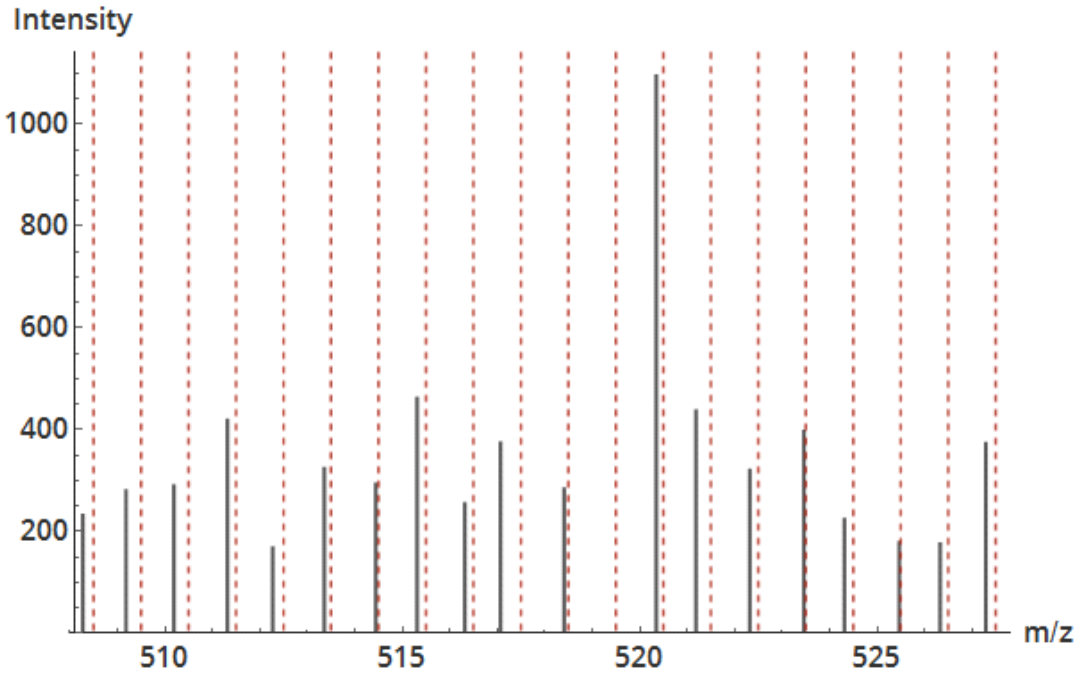

Figure 2: Binning of an experimental fragment spectrum into bins of 1.0 Th. The red dashed lines indicate the binning edges. All signals that fall into the same bin will be merged to one.

An efficient spectrum comparison algorithm needs to find a good trade-off between data reduction, effective signal matching between the spectra and maintaining sufficient mass accuracy. Spectral search engines tackle this challenge in different ways:

The SpectraST method

SpectraST performs binning to 1.0 Th by default, but smaller bins can be specified. To compensate for binning errors, i. e. when peak at the binning edge is assigned a to the wrong bin due to a small mass error, an adjustable fraction of the peak's intensity can be added to the adjacent bins (50% by default) [30].

BiblioSpec

BiblioSpec performs vectorization in 1 Th bins and sums the intensities of all peaks that fall into the same bin. This parameter is non-adjustable in the original version of the program [32].

Pepitome

Pepitome does not perform vectorization by binning of the fragment spectra but applies a pairwise peak-matching between query and candidate spectra to allow for exploitation of high-mass accuracy fragment spectra. Every peak in the query spectrum is matched against the closest peak in the library spectrum within a given mass tolerance. The scores are then calculated from the matched peak pairs. [35]

ANN-SoLo

ANN-SoLo uses 1 Th bins for initial scoring during candidate selection. Once the candidates are found, spectrum comparison is performed on peak pairs with adjustable tolerance, similar to

Pepitome. The authors extended the peak matching to finding mass shifts by modification to a method they termed 'shifted dot product'. [23]

While SpectraST and BiblioSpec use constant bin sizes for spectrum-spectrum match scoring, Pepitome and ANN-SoLo implement a more dynamic approach which allows the user to specify the fragment mass tolerance as a parameter to the search engine.

3.2.4.2 Choosing an optimal bin size for fragment spectrum vectorization

Vectorization of the fragment spectra to 1.0 Th bins offers algorithmic and computational benefits and renders the mass accuracy of query and candidate spectra less critical. It must be evaluated, though, whether the considerable loss of information about the accurate masses of the fragment signals will affect the accuracy of the spectrum identification.

Interesting observations about the effect of fragment mass resolution to identification accuracy can be drawn from experiments with the Orbitrap-linear ion trap hybrid instruments 'Orbitrap Fusion' and 'Orbitrap Elite'. These hybrid mass spectrometers use an Orbitrap mass analyzer for MS1 spectrum acquisition to determine accurate precursor masses and optionally a linear ion trap for MS2 spectrum acquisition. In HeLa runs with the Orbitrap Elite, Michalski et al. used a rather low fragment spectrum resolution with a peak width of 0.47 Th at half maximum with the linear ion trap. Peptide identification was subsequently performed with a fragment mass tolerance of 0.5 Da. The results were compared with HeLa runs where MS1 and MS2 spectra were both acquired on the high-resolution Orbitrap mass analyzer and the search was performed at a fragment mass tolerance of 20 ppm. Interestingly, a higher number of identified peptides (11,543 vs. 10,847) was observed with the low-resolution ion trap data [36]. Here, the higher acquisition speed compensated for the lower mass resolution and accuracy of the ion trap and yielded more identified peptides.

While these findings from instrument performance are not equally applicable to spectral library searching, they demonstrate that good identification performance can be achieved with data featuring high precursor mass accuracy and lower resolution fragment spectra.

3.2.4.3 Recalibration of fragment m/z values

When constant binning is performed on fragment spectra, signals will be assigned to a specific bin based on the fractional parts of their m/z values ('mass defects'). For example, when vectorization to 1.0 Th bin size is applied, a signal at m/z 700.4 will be assigned to the m/z 700 bin, while a signal at m/z 700.6 will be assigned to the m/z 701 bin. This implies that small mass errors at the edge of the bins (around the fractional part of 0.5 in this example) may lead to signals being assigned to an adjacent bin instead of the correct bin.

As mentioned previously, SpectraST used 'peak spreading', where 50% of a bin's intensity is added to the neighboring bins, to compensate for possible binning errors. The peak matching methods as performed by Pepitome and ANN-SoLo do not suffer from this issue.

In this work, a new approach is suggested which provides more accurate constant-bin vectorization of fragment spectra. The method involves a recalibration of the m/z axis of all

fragment spectra prior to vectorization, based on the observation of a typical distance patterns of fragment signals.

This concept relates to the Kendrick mass, where the observed mass is multiplied with the ratio of the nominal (integer) mass and the exact mass of a given base fragment [37]. Characteristic mass defects of certain molecular species can been exploited in mass spectrometric data analysis in various ways [38].

3.2.5 Scoring of spectrum-spectrum matches

For every spectrum-spectrum match, a score is calculated as a measure of similarity between the query and the candidate spectrum. The candidate that yields the highest score is considered the best match and subjected to FDR validation later on.

Many spectral search engines base their scoring scheme on the dot product of the spectrum vectors. Some engines perform additional calculations to derive the final score, like SpectraST, which used a dot bias "to penalize high-scoring matches with massive noise and/or dominant peak" [30] before switching to Rank-transformed dot products in version 5.0 [39].

The absolute intensities of query and library spectra can differ substantially due to the different instruments they were acquired on and different pre-processing methods. The dot product is therefore normalized by the total length of the vectors, which may also be referred to as the cosine similarity (or 1 - CosineDistance) because it reflects the cosine of the angle between the two vectors [40]:

$$\cos(u, v) = u \,.\, v \,/\, (\mathrm{Norm}[u] * \mathrm{Norm}[v])$$

Several measures of similarity were compared by Liu in 2007 [40]. Among these methods were the cosine similarity and the correlation coefficient. The latter was defined as:

$$CE(u, v) = (u - \mathrm{Mean}[u]) \,.\, (v - \mathrm{Mean}[v]) \,/\, (\,\mathrm{StandardDeviation}[u] * \mathrm{StandardDeviation}[v]\,)$$

The correlation coefficient includes shifting of the vectors by their means before dot product calculation and normalization to the standard deviation of the vectors. The authors found that the correlation coefficient yielded slightly better scores than cosine similarity.

In this work, SSM scores are calculated by a related function termed 'correlation similarity'. The correlation similarity is the dot product of spectrum vectors shifted by their means and scaled by their norms – a mixture of cosine similarity and correlation coefficient. The correlation similarity is defined as:

$$CS(u, v) = (u - \mathrm{Mean}[u]) \,.\, (v - \mathrm{Mean}[v]) \,/\, (\,\mathrm{Norm}[u - \mathrm{Mean}[u]] * \mathrm{Norm}[v - \mathrm{Mean}[v]]\,)$$

The correlation similarity yields scores between 0 and 1 for all non-negative valued vectors, with 1 reflecting identical vectors and 0 completely unrelated vectors.

While scaled dot products have proven to be good measures of spectrum similarity, they are prone to certain biases. Dot product scores tend to overweight intense peaks, so that a few intense peaks may dominate the score of a spectrum-spectrum match and average intense peaks

are disregarded, although they may be just as discriminative as the intense ones [39]. Also, spectra with higher number of signals tend to produce higher overall scores [35].

To compensate for the biases, fragment ion intensities are usually transformed before the scoring function is applied. As an alternative, the authors of the spectral search engine *Pepitome* decided to use two probabilistic scoring schemes instead of the dot product, which are not affect by the number of fragment signals [35].

3.2.6 Transformation of fragment ion intensities before scoring

Spectral search engines are provided with experimental signal intensities for all fragment signals both in the library and the query spectra. With virtually all scoring methods, signals of higher intensity have a higher contribution to the score than lower-intensity signals, which is justified, as the intense signals are likely to represent true signals from the peptide and the least intense signals are more likely to be noise. However, as discussed in the previous section, high-intensity signals tend be overweighed, so a reduction of the dynamic range of the intensities is desirable for more accurate SSM scoring. Various methods of intensity transformation have been suggested and implemented (Table 1).

Table 1: Selection of intensity transformation methods and the use in common spectral search engines.

	Transformation function	Search engine examples	References
1	No transformation	(none)	
2	Square root	SpectraST up to 4.0, QuickMod, BiblioSpec	[20, 30]
3	Logarithm	(none)	
4	Rank transformation	SpectraST as of 5.0, QuickMod, Pepitome, ANN-SoLo	[20, 23, 35, 39]
5	Unity intensity	(none)	[40]

The square root (method 2 in Table 1) lowers the impact of high-intensity peaks on the score. When used in conjunction with the dot product as the scoring method, it compensates for the fact that signals of doubled intensity affect the dot product for times as much [39]. Square root transformation and dot product scoring were used together in SpectraST until version 4.0 [20].

Log transformation of the intensities (method 3) reduces the dynamic range of intensity even more, but still preserves relative intensity differences between the signals.

Rank transformation (method 4) does not prevail the intensity values of the peaks at all but only its order. The peak with the lowest intensity is assigned the value 1, the seconds lowest the value 2, and so on. Thus, in a fragment spectrum with 50 peaks, the most intense peak will be assigned the value 50. Rank transformation was introduced in SpectraST with version 5.0 [39].

Setting all intensities to unity (=1) (method 5) will calculate the score based on the presence or absence of peaks with no respect to the peaks' intensities. This has been tested by Liu 2007 as 'counting the number of matching peaks' [40].

In addition to transformation of intensities, spectral search engines may only use a limited number of signal, like the top 50 signals (ANN-Solo), to calculate the score, effectively setting the remaining signal intensities to zero [23]. By limiting the number of fragment signals for scoring, search engines compensate for the fact that denser spectra tend to produce higher scores. Also, *Pepitome* can apply noise peak filtering using adjustable thresholds [35]. In *SpectraST*, the maximum number of peaks used for scoring can be adjusted and is set to 150 by default [41]. X!Hunter only uses the top 20 peaks of each spectrum [20].

3.2.7 Implementation of a weighted scoring function

A possible extension of the scoring of transformed intensity values is the application of a weighted scoring function. In a weighted scoring function, every m/z signal is multiplied with an adjustable weight to account for the potentially different discriminative power of different m/z values. It is conceivable that signals which are universally present in all spectra may be scaled down by the multiplication with lower weights, and signals which are specific to individual peptides may receive higher weights. For instance, the fragment signals at m/z 147 or m/z 175 are frequently observed in tryptic peptides representing the y1 ions of a C-terminal lysine and an arginine. These signals may not contribute useful information to measure the similarity of two spectra, or more so the dissimilarity of two unrelated spectra, because they are present in most spectra, anyway. In contrast, signals that are less ubiquitous among all spectra may have more discriminative power. Ideally, signals of higher discriminative power will receive higher weights and since have more impact on the score, which will render the score more discriminative as well.

The application of a weighted scoring function and its effect on the peptide identification performance will be evaluated in this thesis.

3.2.8 Hit validation

Spectral search engines output a score for every spectrum-spectrum match. While the best scoring candidate spectrum can be assumed to be the closest match to the query spectrum among all spectra in the library, it is not guaranteed to be correct. Particularly, spectral libraries do not cover the entirety of protein sequences, thus it cannot be inferred from the score alone whether the best match is a true-positive hit, or whether it is a false-positive hit, where the real reference spectrum would have scored even better. A validation step therefore follows the scoring to decide whether the best match is likely to be correct or not. Common validation procedures

involve the estimation of the false-discovery rate, which allows to limit the number of false-positive hits among all positive hits.

3.2.8.1 Delta score

The delta score is the difference between the score of the 'best match' and the 'second best match', where the latter is the highest scoring candidate which represents a different peptide than the best match [39]. It can be interpreted as a measure of how much better the best match explains the fragment signals in the query spectrum than the second-best match. A low delta score means that the two matches are very close to each other and that the search engine cannot discriminate well between the two candidates. Conversely, a high delta score indicates that the best candidate matches the query spectrum to a much higher degree than the second best, and thus is more likely to be correct.

The delta score can be used for spectral library searching as well as sequence database searching. SEQUEST uses an adjustable delta score threshold to filter ambiguous hits [42] before FDR-based validation usually follows. SpectraST also calculates the delta score, which can be used for hit validation afterwards [39].

3.2.8.2 False-discovery rate

A false discovery rate (FDR) estimation is the most popular procedure to discriminate between true-positive and false-positive hits at a certain confidence. An FDR of 0.01 (1%) is usually considered acceptable for proteomics studies using modern Q-TOF and orbitrap instruments. The score cutoff is adjusted so that among all identified peptides, 1% should be false-positives while 99% should be true-positives [43].

A non-decoy approach to FDR estimation was developed by Keller et al. with the software PeptideProphet. PeptideProphet models the scores of the true-positive and false-positive hits by fitting two normal distributions into the distribution of all scores [44]. The underlying assumption is that both the scores for the true-positive hits as well as for the false-positives hits can be modelled from the empirical distributions of all hit scores. It was used for hit validation in the original SpectraST publication [30].

In sequence database searching, the use of a decoy library is the most common approach to FDR estimation. Decoy spectra are artificially generated spectra which are inherently 'wrong', so that every match with a decoy spectrum is a false-positive hit. Ideally, the score distribution obtained from search against the decoy database represents the scores of false-positive hits. From the distribution of decoy and target scores, a score cutoff will be calculated so that the number of decoy hits is FDR * the number of target hits at most (global FDR) [45]. Unlike sequence database searching, where decoy libraries can be easily be generated by sequence reversing [46], generation of decoy spectra for spectral library searching is not as simple [47].

3.2.9 Decoy spectrum generation for false discovery rate estimation in spectral library searching

Decoy spectra are generated to model random (false-positive) hits. When generating a library of decoy spectra, it must be ensured that decoy spectrum will match the query spectra equally well as random (false-positive) hits from the spectral library. If decoy spectra produce scores which are lower compared to the false-positives hit scores, the FDR will be underestimated, meaning that the number of false-positive is actually higher than the desired target FDR. Conversely, if the decoy scores are higher, the FDR will be overestimated and the number of total identifications will be below optimum [43].

The quality of a decoy library can be assessed by comparing the scores of the decoy spectra to the scores of known false-positive hits. Cheng et al. searched spectra from human samples against E. coli, yeast, or chicken databases to generate a 'ground truth' for false-positive hits, and compared the performance of their own decoy library generation method [48]. However, Lam et al. argued that cross-species decoy spectra are inferior to decoy spectra created artificially from the target library for its potential different library size and different precursor mass distributions [47].

The generation of decoy spectra has been found to be more complicated than the generation of decoy sequences. In order to match query spectra with similar probability as the false-positive spectra, decoy spectra need to have similar features as the real spectra [47]. Methods that have been proposed for decoy spectrum generation from spectral libraries include the *shuffle-and-reposition* method, the *DeLiberator* method, and the *peak-shift* method. A method which uses original (unaltered) spectra as decoys but changes the precursor mass is the *precursor-swap* method [20].

The *shuffle-and-reposition* method was implemented in SpectraST 4.0 [41]. It creates a decoy spectrum from a real spectrum by shuffling its sequence and repositions all annotated fragment ions accordingly. Non-annotated fragment ions were left unchanged [47].

DeLiberator enhances the shuffle-and-reposition by shuffling the annotated ions incrementally until a similarity score of below 0.5 is achieved and also shuffles non-annotated fragment ions [49].

The *peak-shift* method shifts all fragment ions by a fixed m/z to generate decoy spectra [48].

Precursor-swap does not change the fragment spectra directly but assigns a different precursor mass to them. Among the four methods mentioned here, it is the only method that presents original spectra instead of artificially modified ones – although from different precursor masses – to the search engine.

When applied to their dataset, Cheng et al. observed that all methods tended to produce decoys which would match worse than the real spectra, so they would all underestimate the FDR [48]. However, the authors demonstrated that decoy spectra created with their *precursor-swap* method yielded FDRs that were closest to the true FDRs. SpectraST has implemented the *precursor-swap* method in version 5.0 [41].

In this work, two spectrum manipulation methods, 'intensity shuffle' and 'm/z randomization', and an improved version of the *precursor-swap* method, 'precursor-shuffle', will be implemented and the quality of decoy spectra will be evaluated.

3.3 Machine learning for improvement of SSM scoring

Machine learning is a form of computation where a program learns to improve on a task automatically through experience [50]. Typical applications of machine learning are problems which cannot be efficiently solved by algorithmic programming for their very high complexity and too many degrees of freedom. A self-learning program is being designed and presented with training data with known outcomes. Through many training rounds, the program gets better at predicting the outcome by adjusting its internal parameters. Ideally, it can predict the outcome of unseen (validation) data just as well after training has ended and therefore be used to solve new tasks where the outcome is not yet known.

The large sizes of today's proteomics datasets, possibly enhanced with data available through online repositories, are an excellent training data base for machine learning. Proteomics data processing can be extended by machine learning in many ways, one of which is the frequently used validation tool Percolator. The concept of machine learning is applied to the scoring of spectrum-spectrum matches in this work. An artificial neural network will be built with the aim to automatically learn to produce improved SSM scores based on training with known positive and negative SSMs.

3.3.1 Artificial Neural Networks

The concept of a network of simple primitives which 'fire' at certain input values dates back to a work of McCulloch and Pitts in 1943 [51]. The authors developed a logical model for the behavior of the nervous system in the human brain. At that time, however, computational resources were far away from being able to tackle real-world tasks with neural networks. Other tactics dominated the machine learning fields until the 1990s. With the evolution of computer hardware and the development of the backpropagation method for training of multi-layer networks, interest in neural networks for machine learning regained during the past 25 years.

3.3.1.1 Perceptron

The most basic form of artificial neurons, the building blocks of neural networks, is a perceptron, conceived by Frank Rosenblatt and published in 1958 [52]. The perceptron multiplies an input value with a learnable weight (w) and adds a learnable bias (b) to the product. If the result is > 0, the perceptron returns 1 (it 'fires'), otherwise 0.

3.3.1.2 Neurons

Artificial neurons are a generalization of the perceptron with non-binary outputs. A neuron performs the same calculation w*x + b of input value x, learnable weight w and learnable bias b, but may apply any activation function to determine the output value of this calculation. Common

activation functions are the step function, linear function, the logistic sigmoid or the hyperbolic tangent function, or the ramp or rectifier function [53].

In this sense, perceptrons are a subtype of artificial neurons – linear neurons with the step function as the activation function.

3.3.1.3 Training

The process of adjusting the learnable parameters in a neural net is called training. Training data is provided along with the true output ('ground truth') to a (usually) randomly initialized network. The output of the network is compared to the ground truth by a loss function, which quantifies the difference between the calculated and the true output. The learnable parameters are then adjusted by a method called backpropagation to allow the network to improve on its task. This process is repeated for many rounds.

3.3.1.4 Validation

Training of a large number of learnable parameters with a comparably small number of samples is susceptible to overfitting, i. e. instead of learning a generalized solution to a given problem, the neural net 'memorizes' specific properties of the training data. This results in a neural net that performs much better on training data than on unseen data.

To test for overfitting, a validation set of unseen data can be presented to the neural net and the loss can be calculated. A large difference between training and validation indicates overfitting. In this case, the number of training rounds can be reduced, the training set can be enlarged, regularization can be applied, or a different neural net design can be chosen [54].

3.3.1.5 Usage examples of artificial neural networks

Today, artificial neural networks are state-of-the-art techniques for complex machine learning tasks [55]. Growth of computational power through distributed computing, acceleration by specialized components such as graphics processors (GPUs) or application-specific integrated circuits (ASICs) allowed for the design of complex neural nets with several hidden layers (*deep networks*) and millions of learnable parameters. The introduction of convolutional neural networks, recurrent and long short-term memory networks lead to major improvements in various computational fields in the past decade. Those networks perform tasks which are considered the most complex for any machine learning endeavor, including image and speech recognition, image or text generation [55], re-colorization of grayscale images [56], and chess and go computers [57].

3.3.2 Application of neural nets for optimization of the scoring function

In this work, a shallow neural network, i. e. a neural network with only one layer of learnable parameters, will be used to optimize the scoring function of spectrum-spectrum comparisons.

The underlying premise is that the discriminative power of signals at different m/z values is not equal. To exploit this phenomenon for spectrum similarity scoring, every m/z value is multiplied

with a learnable weight before the scoring function is applied. Separate weight vectors will be established for the query and the database spectra. Optimization of the two vectors is a multi-dimensional optimization problem which will be tackled with a neural net in this work. During training with a subset of experimental data, the neural net will optimize the weights by backpropagation to maximize the difference between the scores between positive spectrum-spectrum matches and negative spectrum-spectrum matches.

3.4 Performance considerations

Processing large amounts of data involves significant computational time and memory. All algorithms were designed to run a single workstation computer in reasonable time and with memory efficiency. Benchmarking of method run times, memory consumption, as well as frequent code optimization has been an integral part in the conduction of this work. The performance evaluations will be addressed at the end of the results section.

4 Aim of the thesis

The large-scale spectral libraries that have become available through the growth of online proteomics data repositories demand for powerful computational methods to be fully exploited for proteomics data analysis. The diversity of reference data in spectral libraries and experimental query datasets due to different instrumentation, sample preparation and acquisition parameters require robust search engines which yield reliable identifications results.

The present work aims to advance spectral library searching as a fast, reliable and sensitive method for the identification of spectra from mass spectrometric proteomics data. A careful analysis of the constitution and coverage of the PRIDE Cluster human spectral library will be conducted. Several steps of spectral library searching will be examined and optimized. The results will be compared to the established spectral library search engine SpectraST. Very high confident identifications from the sequence database search engine SequestHT will serve as a 'ground truth' for all optimization steps.

This work will incorporate machine learning through neural networks to enhance the scoring of spectrum pairs and achieve higher identification rates. While only scratching the surface of what is possible with state-of-the-art neural networks, it will be demonstrated how machine learning can improve the accuracy of a spectral search engine.

Findings from this work shall help to understand the capabilities and limitations of spectral library searching. This work will address coverage and confidence of spectral library searching and discuss strategies to achieve both the high identification rates of sequence database searching and the high specificity of spectral library searching.

5 Results and discussion

5.1 The PRIDE Cluster spectral library

The human subset of the PRIDE Cluster spectral library, built in April 2015 by the PRIDE team, was used for all spectral library searches. It consisted of 789,745 spectra from 189,400 peptides. The following sections give insight about basic properties of the spectral library.

5.1.1 Peptide mass-to-charge ratios

The distributions of precursor mass-to-charge ratios and molecular weights are depicted in Figure 3.

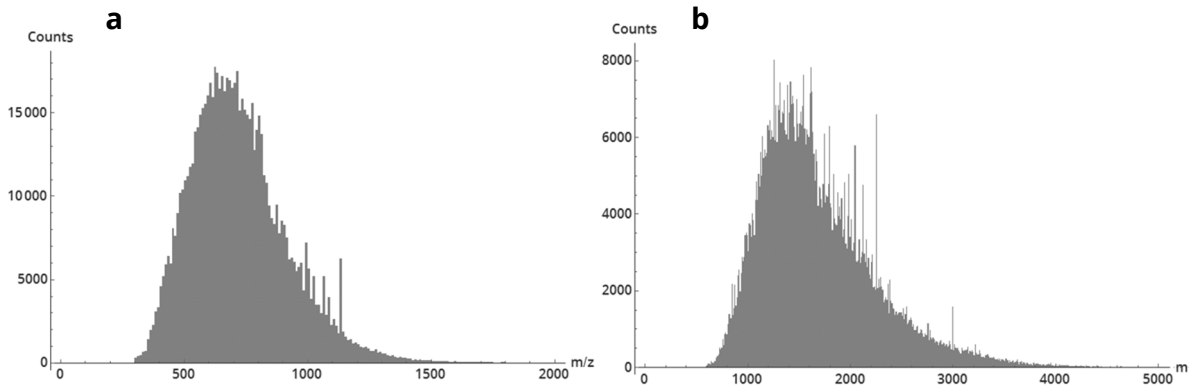

Figure 3: Histograms of precursor mass-to-charge ratios (a) and molecular weights (b) of all peptides in the PRIDE spectral library. The molecular weights were calculated as $m_{MW} = (m/z - 1) * z$. Median values are m/z = 701.8 Th and m = 1,580.1 Da.

The precursor m/z of most peptides was in range of 400 to 1500 Th with the median m/z at 701.8 Th. Molecular weights were in the range of 800 to 4000 Da, median of 1,580.1 Da, suggesting that 2+ is the most dominant charge of the peptides, as the molecular weights are roughly the doubled m/z value. The distribution of molecular weights is mirrored by the distribution of the sequence lengths, which is shown in Figure 4.

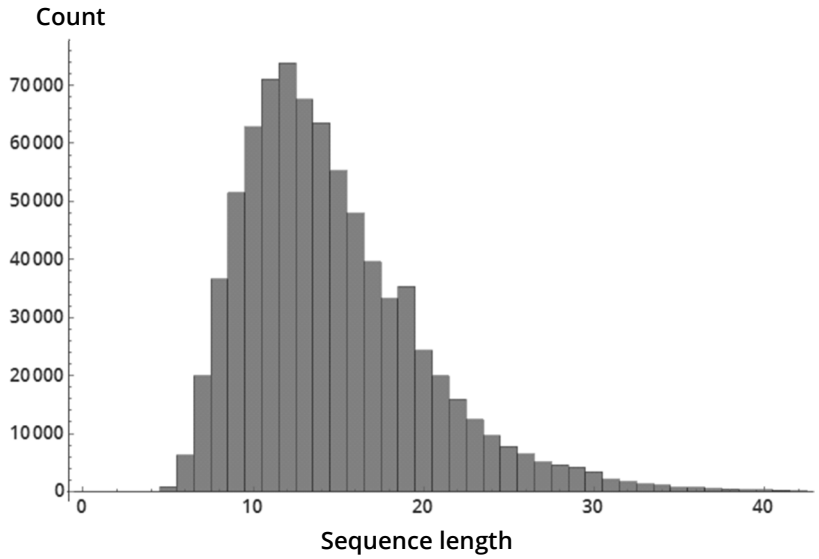

Figure 4: Histogram of sequence lengths of the peptides in the PRIDE spectral library. Min = 5, Median = 16, Max = 82 (outside the depicted range).

The minimum sequence length was 5 amino acids and the longest peptide in the library consisted of 82 amino acids. Median length was 16.

5.1.2 Peptide charge states

Peptide spectra were acquired from precursors of charge states between 1 and 8. Table 2 summarizes the charge distribution in the spectral library.

Table 2: Overview of peptide charge states in the PRIDE spectral library.

Peptide charge	Number of peptides	Percentage
1	10,474	1.33%
2	521,614	66.05%
3	224,745	28.46%
4	30,289	3.84%
5	2,336	0.30%
6	263	0.03%
7	20	0.00%
8	4	0.00%
Total	789,745	100%

Almost two thirds of the peptides were doubly charged, while 28.5% were triply charged. The 2+ and 3+ charge states account for 94% of the peptides.

5.1.3 Peptide modifications

Every distinct combination of modifications of a peptide make for a different peptide species that requires its own library entry to be identifiable in a spectral library search. Table 3 summarizes the number of peptides which contain at least one of the listed modifications.

Table 3: List of modifications with occurrence of >1% in the 789,745 library entries. The number of peptides which contain at least one of the modification is given. One peptide can contain multiple modifications; therefore, the percentages add up to more than 100%.

Modification	Number of entries	Percentage
Unmodified	447,014	56.60%
Carbamidomethyl	95,509	12.09%
Oxidation	64,870	8.21%
TMT6plex	47,499	6.01%
iTRAQ4plex	46,180	5.85%
Phospho	25,534	3.23%
iTRAQ8plex	25,029	3.17%
+144.105919 Th	18,502	2.34%
Label:13C	18,025	2.28%
+31.989829 Th	15,910	2.01%
Formyl	14,650	1.86%
+45.029395 Th	8,384	1.06%
Methylthio	8,308	1.05%

The majority of peptides in the PRIDE Cluster library (56.6%) are unmodified. The most frequent modification is carbamidomethylation, found in 12.1% of all peptides, followed by oxidation (8.2%) and chemical labels (TMT, 6.0%; iTRAQ, 3.2%). Phosphorylation is present in 3.2% of all peptides. Carbamidomethylation is almost universally specified as a fixed modification of cysteine residues in database searches, since it is widely used as the protective group for cysteines to avoid re-formation of disulfide bonds. Peptides with chemical labels (TMT, iTRAQ) accounted for more than 15% of the library. Given that these peptides can only be detected in experiments that specifically incorporate the labels, this indicates that labeled proteomics analyses are somewhat popular among the studies uploaded to the PRIDE repository.

Three unspecified modifications of m/z 144.1059 Th, m/z 31.9898 Th and m/z 45.0294 Th are found in 2.3%, 2.0% and 1.1% of the peptides, respectively. A possible chemical composition for the +31.9898 modification is the addition of two oxygens (double oxidation, m = 31.9988 Da). For +144.1059, a possible sum formula is $C_6H_4D_5N_3O$, which is $C_6H_9N_3O$ with 5 hydrogens exchanged for deuterium (^2H). +45.0294 may correspond to an acetylation with one ^{13}C atom ($^{13}CCH_4O$, m = 45.0296 Da).

5.1.4 Replicate entries

The 789,745 spectra in the PRIDE spectral library represent 189,400 (24,0%) unique peptide sequences. Unique combinations of peptide sequences with modifications sum up to 340,249 (43.1%). When including different charge states of the previous, 412,389 (52.2%) unique peptide species are found in the library.

The remaining entries are replicates of the peptide species which were not condensed into a single cluster by the PRIDE Cluster algorithm. Figure 5 summarizes the number of spectra for each combination of sequence, modifications and charge states.

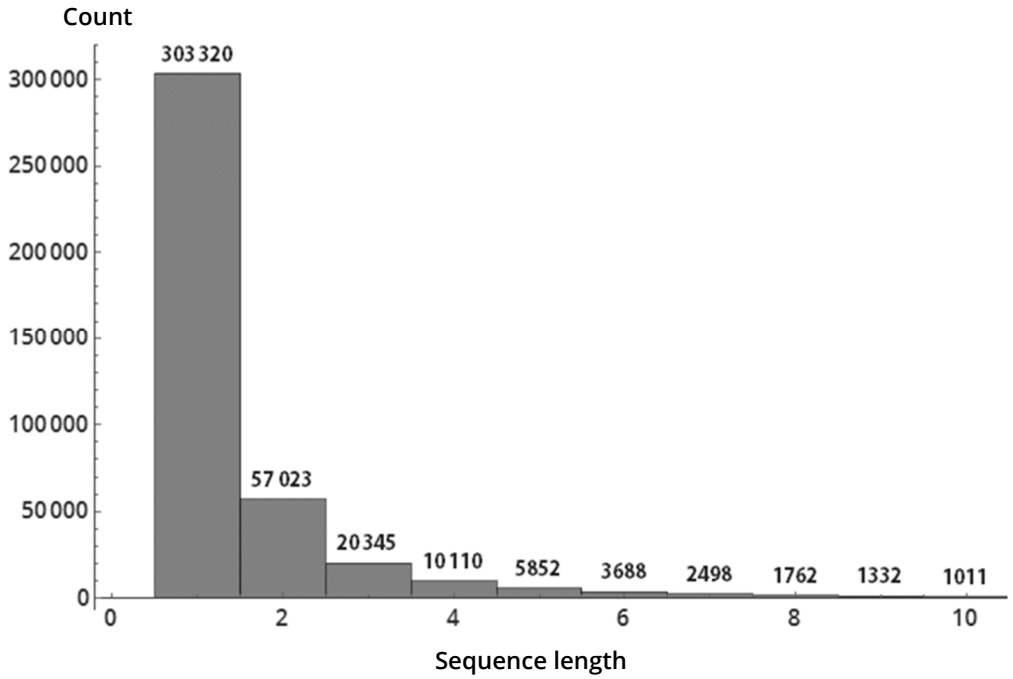

Figure 5: Number of PRIDE clusters per peptide species (unique combinations of sequence-charge-modifications). Most peptide species were represented by a single entry. For some peptide species, higher number of replicates were found. Only the range from 1 to 10 replicates is shown in this plot.

303,320 library entries represented exactly one peptide species in the PRIDE Cluster library. For 57,023 peptide species, two spectra were included in the library, and 20,345 were represented by three spectra. Interestingly, a small number of peptide species was represented by a very high number of spectra in the PRIDE Cluster library, as shown in Figure 6.

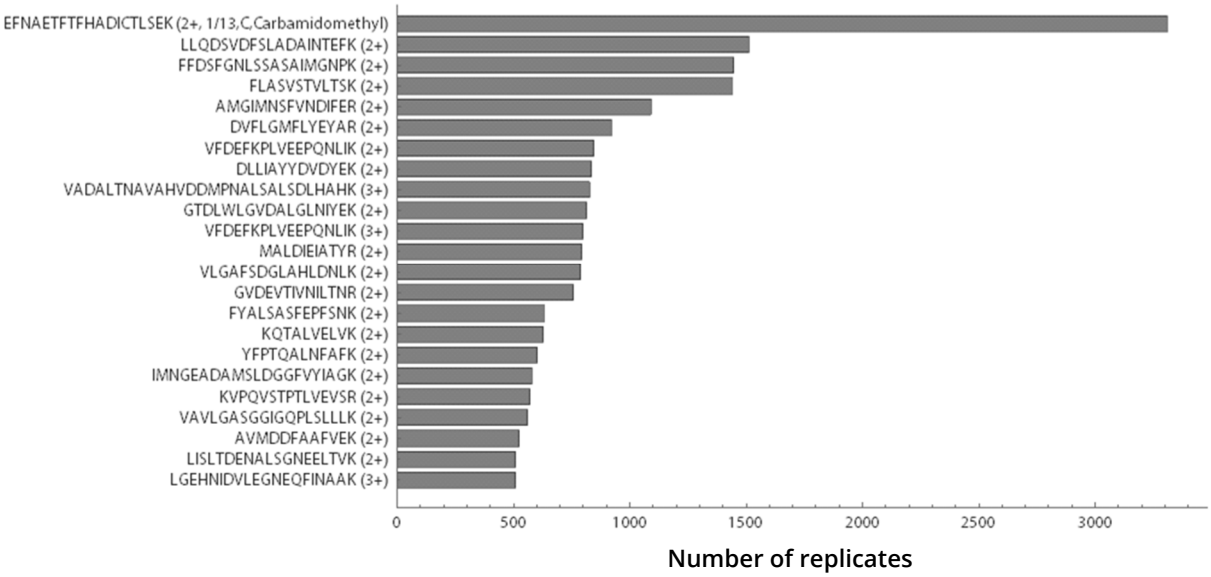

Figure 6: Peptide species in the PRIDE Cluster library with more than 500 spectra. The most frequent peptide species, EFNAETFTFHADICTLSEK with charge state 2+ and a carbamidomethylation at C13, was represented by 3,310 spectra. Other peptide species had up to 1,512 replicates in the library.

When the PRIDE Cluster library was built, experimental spectra from many proteomics datasets were condensed into consensus spectra by clustering for spectral similarity. It seems natural that spectra, which were acquired from a large variety of samples on different instruments and with different parameters, are dissimilar to a certain extent, even when they represent the same peptide species. Multiple spectra of the same peptide species are therefore expected and may be beneficial for the identification rate due to the higher chance of a finding a good match to the query spectrum among the replicates. However, presence of more than 500 replicates for some peptide species seems unreasonably high. It is not known why these replicate spectra were not grouped together into a much lower number of clusters in the PRIDE Cluster library. Further investigations on the properties of the affected spectra and the PRIDE clustering algorithm could reveal possible reasons for this phenomenon but are beyond the scope of this thesis.

5.1.5 Sequence coverage

The 189,400 unique sequences in PRIDE cluster sum up a total length of 2,676,998 amino acids (aa). 89.2% (2,390,019 aa) were found in the human SwissProt database, and a subset of 75.7% (2,043,508 aa) in tryptic peptides of 0 to 2 missed cleavages. The remaining 10.8% of sequences could not be assigned to human proteins in the SwissProt database. Some of them originate from other databases, like the cRAP database with common contaminants (92,323 amino acids, 3.4%), while some could not be assigned to any proteins even by blasting against all non-redundant sequences from all species using *blastp*. It is not known how those sequences were identified in the original experiments that were submitted to the PRIDE repository. Possible explanations are the identification with a custom protein database (e. g. from DNA or RNA sequencing results) or automated or manual *de novo* sequencing.

In total, the peptides in the PRIDE Cluster library cover 25.5% of the tryptic peptide sequences in the human proteome (9,372,879 amino acids).

5.1.6 Fragment spectrum signals

The number of fragment signals per spectrum depends on many parameters, including precursor intensity, precursor isolation purity, fragmentation mechanism and energy, type of mass analyzer, and, finally, spectrum processing with noise filtering and peak picking. Upon building the PRIDE Cluster library, spectra were filtered to keep only the 70 highest peaks before clustering to homogenize the fragment signal patterns [29]. Subsequently, when pre-filtered spectra are clustered, they may add up to higher numbers of fragment signals per consensus spectrum. Figure 7 shows the distribution of fragment signal count per spectrum.

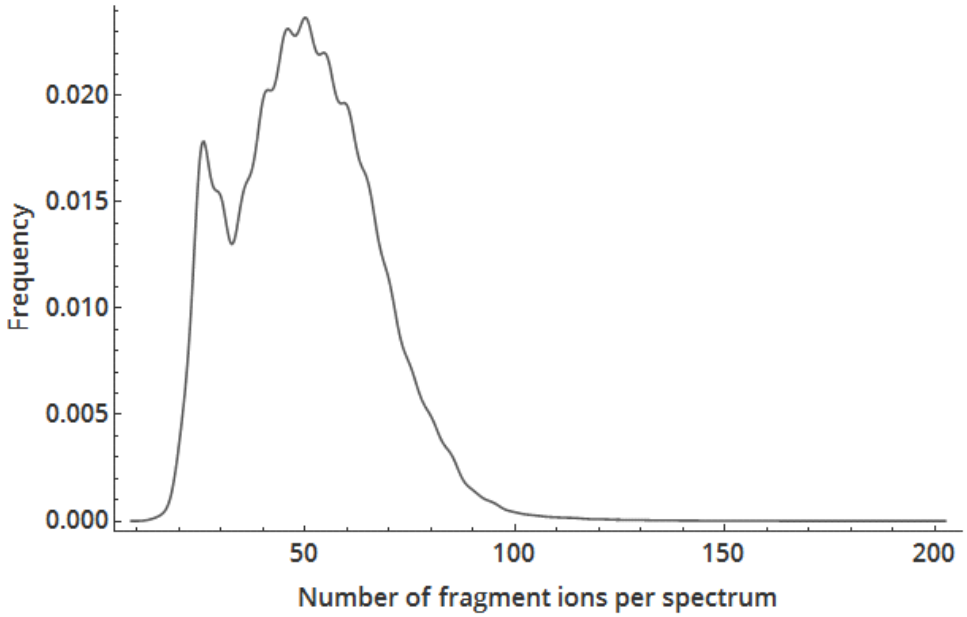

Figure 7: Distribution of fragment signal count per MS2 spectrum in the PRIDE Cluster spectral library. Min = 12, Median = 50, Max = 235.

Most library spectra contained between 20 and 100 signals, the median was 50. The number of fragment signals in the library is a of particular importance for query spectrum filtering. When query and library spectra differ significantly in the number of fragments, scoring may be less discriminative because too many fragment signals match randomly.

Figure 8 depicts how the fragment signals are distributed along the m/z axis.

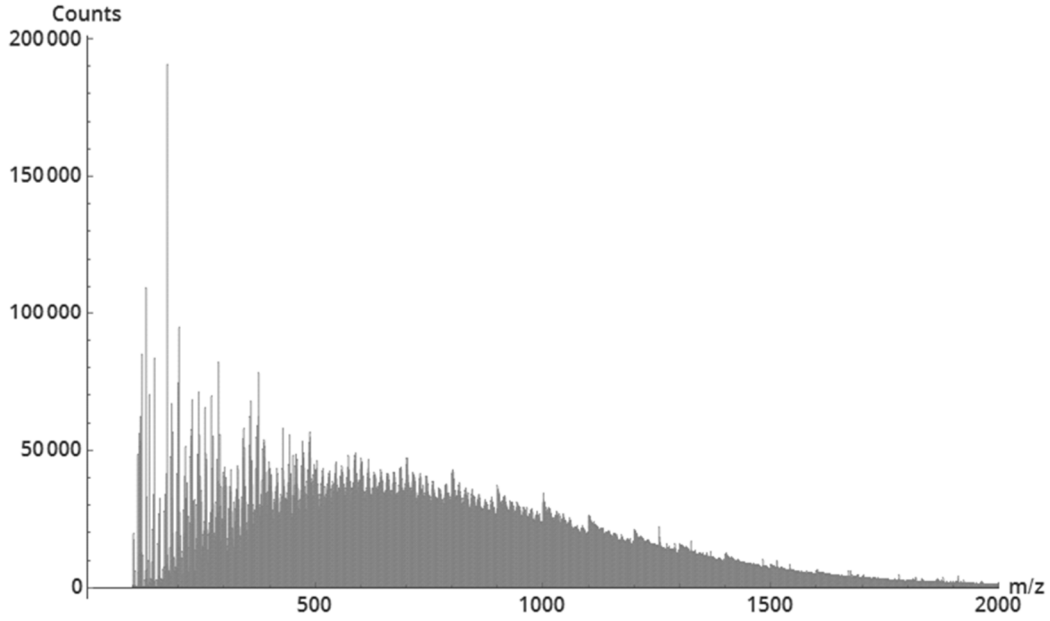

Figure 8: Total number of fragment signals per whole-Th m/z bin in all PRIDE Cluster library spectra. Highest counts were observed for m/z 175, which was present in 190,707 spectra, followed by m/z 147 (83,580 spectra).

The m/z value of the fragment signals follows right-tailed distribution with a center around m/z 600. Outliers are specific fragment signals such as m/z 175 (highest count) and m/z 147 (second-highest count), which commonly represent the y_1 ions of arginine and lysine, respectively. These amino acids are found at the C-termini of tryptic peptides, which account for more than 75% of the peptides in the PRIDE spectral library. Additional outlying signals can be found in the m/z range below 500 Th.

5.2 The HeLa benchmark datasets

Two independent *bottom-up* LC-MS/MS datasets from HeLa lysates were used as benchmark datasets to establish the current spectral identification method and test its performance. The 'QEx-HeLa' dataset was acquired on an Orbitrap Q Exactive and features high resolution, high mass accuracy for both MS1 and MS2 spectra, as well as a large number of MS2 spectra due to the long LC gradient of 120 min. The 'Fus-HeLa' dataset, acquired on an Orbitrap Fusion, features high resolution and mass accuracy for MS1 spectra, but a low mass accuracy on the MS2 level and an overall smaller number of fragment spectra due to the shorter gradient.

Both datasets represent typical real-world experiments in our laboratory. Optimization of the method aims to produce reasonable performance on both datasets. Table 4 summarizes the acquisitions parameters of the datasets.

Table 4: Acquisitions parameters for two HeLa datasets. R = Resolution at m/z 200.

Parameter	Q Exactive HeLa dataset	Fusion HeLa dataset
Liquid Chromatography	Waters nanoAcquity	Thermo Dionex 3000
Mass spectrometer	Thermo Orbitrap Q Exactive	Thermo Orbitrap Fusion
MS1 mass analyzer	Orbitrap (R = 70,000)	Orbitrap (R = 120,000)
MS2 mass analyzer	Orbitrap (R = 17,500)	Ion trap (mode = rapid)
Duration total / gradient	170 min / 120 min	70 min / 45 min

The datasets were processed in Proteome Discoverer to identify the MS2 spectra with SequestHT as a 'ground truth' and to generate *mgf* files containing all MS2 spectra and precursor information. All subsequent processing used the *mgf* files created by Proteome Discoverer. Table 5 prints basic statistics about the datasets.

Table 5: Statistics for HeLa Datasets for Benchmark.

Parameter	Q Exactive HeLa dataset	Fusion HeLa dataset
Number of MS2 spectra	50,176	30,722
Peptides identified by SequestHT (FDR 0.01)	25,424	12,436
Peptides identified by SequestHT (FDR 0.01)	22,352	11,452
Identification rate (FDR 0.01)	50.7%	40.5%

The QEx-HeLa dataset contained more than 50,000 MS2 spectra, approximately half of them were identified at 1% FDR. More than 30,000 MS2 spectra were acquired in the Fus-HeLa dataset, around 40% of which were identified at 1% FDR. Overall, the Fus-HeLa datasets contained roughly half as many identified peptides as the QEx-HeLa dataset, which is justified given the shorter method run-time of 70 min instead of 170 min for the QEx-HeLa dataset. For subsequent method optimization, the peptides identified at 0.1% FDR were used (22,352 from QEx-HeLa, 11,452 from Fus-HeLa) and referred to as 'very high confident peptides' in this work.

To get more insights about the characteristics of the fragment spectra from the two datasets, the number of fragment signals per spectrum was plotted in Figure 9.

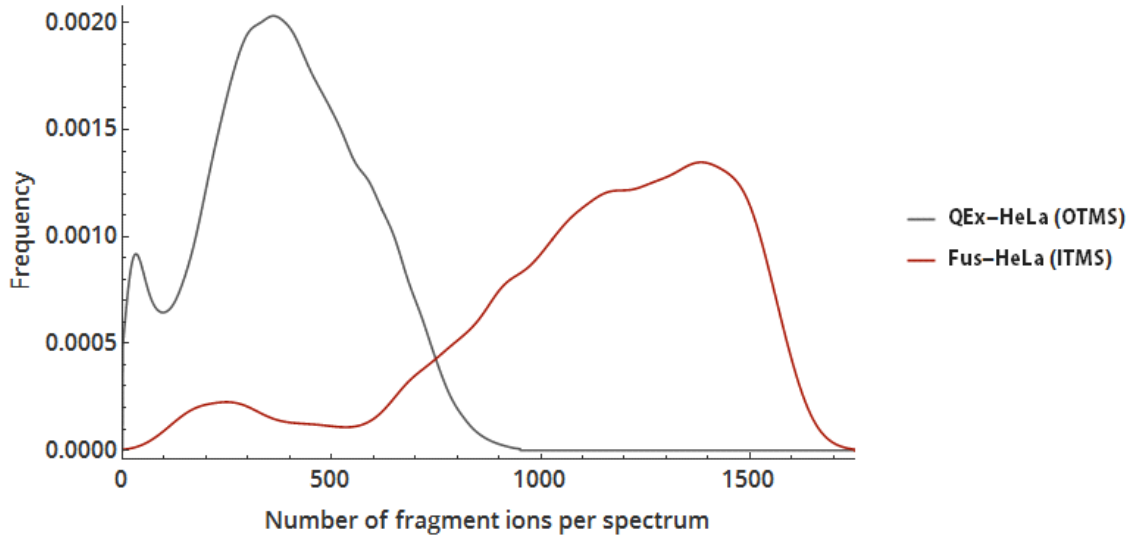

Figure 9: Distributions of fragment signal count per MS2 spectrum in the two HeLa datasets. MS2 spectra were acquired with an Orbitrap mass analyzer (OTMS, QEx-HeLa) or an ion trap mass analyzer (ITMS, Fus-HeLa). QEx: Min = 14, Median = 419, Max = 1044. Fus: Min = 293, Median = 1,179.5, Max = 1,699.

The Fusion dataset contained very dense spectra with a median of 1179 fragment signals per spectrum. In contrast, the Q Exactive spectra contained only 419 fragments per spectrum at the median. This may be a result of the different nature of mass analyzes and/or different data processing, including noise filtering and data reduction upon RAW file storage, and the MS2 peak picking in Proteome Discoverer.

While the Q Exactive spectra were sparser, both datasets still featured considerably more fragment signals than the PRIDE library spectra (median of 50). Since signal density may introduce a bias to similarity scoring, reduction of the number of signals will be evaluated during method optimization.

5.3 Development of a spectral library identification method

5.3.1 Precursor matching

The first step of spectrum identification is the selection of candidates from the spectral library by matching of the precursor m/z. Modern high-mass accuracy instruments allow for small precursor m/z tolerances, but small mass errors may still be present in the library and/or the experimental spectra and account for candidate misses during precursor selection if the precursor window is set too narrow.

A tolerance of 0.02 Th was selected as a typical value for datasets acquired on high-resolution instruments. It was compared to a tolerance of 0.5 Th to estimate the potentially missed peptides due to a higher precursor mass error than 0.02 Th. To estimate the effect on the search space, the

number of candidate spectra that match a given query spectrum's precursor m/z at tolerances 0.02 Th and 0.5 Th is shown in Figure 10.

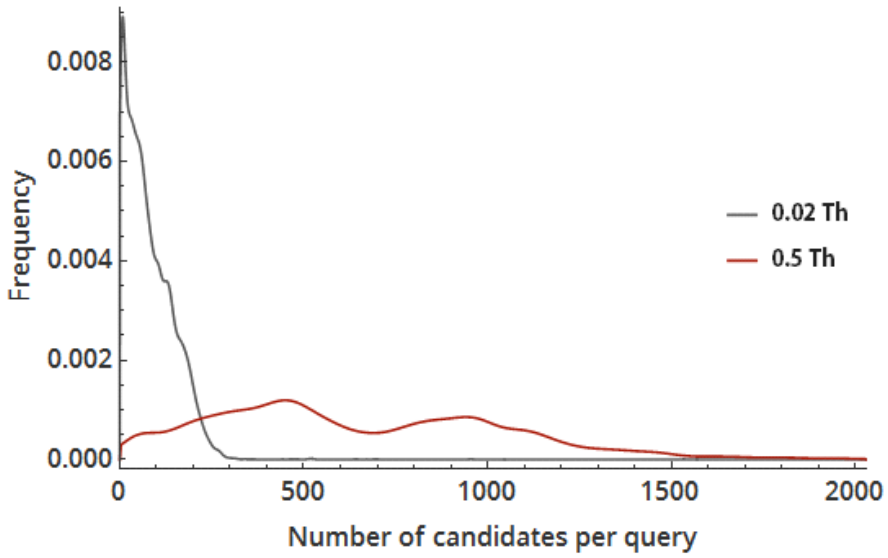

Figure 10: Distribution of numbers of candidates per query at precursor mass tolerances 0.02 Th and 0.5 Th. 0.02 Th: Min = 0, Median = 64, Max = 4220. 0.5 Th: Min = 0, Median = 575, Max = 4335.

The increase of the search space at a mass tolerance of 0.5 Th compared to 0.02 Th is remarkable. On average, a query spectrum was compared against 64 spectra at a tolerance of 0.02 Th, but against 575 when 0.5 Th precursor mass tolerance was used. The larger tolerance increases the chance of including correct candidate spectra by the precursor search. At the same time, the larger search space renders random (false-positive) hits more likely, so that stricter score thresholds will have to be used to keep the FDR at the desired level. To test whether higher precursor mass tolerance results in more or fewer overall identifications, two spectral library searches will be performed with either 0.02 Th or 0.5 Th tolerance.

To leverage the high mass accuracy of the experimental data, a search engine (or the spectral library builder) can correct for the mass errors of the identified library peptides. Since the exact peptide m/z can be calculated from the sequence, the charge state and the modifications, precursor matching can be performed with the calculated m/z instead of the experimental m/z. In the PRIDE Cluster library, every identified peptide was annotated with its experimental mass ('PEPMASS' or 'Parent') and the mass deviation from the theoretical mass ('DeltaMass'). Recalculation of the theoretical peptide masses confirmed the correctness of the 'DeltaMass' parameter, so 'Parent' – 'DeltaMass' could be used as the calculated m/z. A third spectral library search will be performed with narrow precursor tolerance (0.02 Th) and the calculated m/z as a reference.

The identification results of a random sample of 2,000 from very high confident query peptides with precursor mass tolerances of 0.5 Th, 0.02 Th, and 0.02 Th to the calculated precursor m/z are summarized in Table 6.

Table 6: Positive hits and peptide misses for spectral library searches of 2,000 very high confident peptides at different precursor m/z tolerances. A query was counted as a 'miss' when no candidate spectrum of correct identity was present in the database within the given precursor m/z tolerance. The Rank-Top150-CorrelationSimilarity method was used for intensity transformation and spectrum scoring, which will be discussed in later sections. When the highest scoring candidate reflected the correct sequence, the match was counted as a 'positive hit'.

Precursor m/z tolerance	Database reference mass	Positive hits	Misses	Median delta score
0.5 Th	experimental m/z	1,722	150	0.193
0.02 Th	experimental m/z	1,745	201	0.233
0.02 Th	calculated m/z	1,784	150	0.292

At a precursor mass tolerance of 0.5 Th, 150 of the query peptides were not found in the database. When the mass tolerance was reduced to 0.02 Th, the number of missed peptides increased to 201 due to their higher mass error. However, the reduction of the search space led to an improvement of positive hits from 1,722 to 1,745 and higher delta scores. Finally, when the calculated m/z was used as the reference, the number of misses was reduced to 150 again, and the number of identifications increased to 1,784 with even higher delta scores.

The present method of spectrum comparison clearly benefits from keeping the search space as small as possible. By using the calculated mass from the library peptides and experimental data from high-accuracy mass analyzers, the precursor m/z tolerance could be set as narrow as 0.02 Th with no loss in sensitivity but enhanced identification rate.

In view of these findings, the default precursor tolerance in SpectraST of 3.0 Th seems to be far from the optimum for the high-accuracy datasets analyzed in this work. However, Hsieh et al. have demonstrated that higher precursor m/z tolerances can be practical when applied during the initial search and filtered later when evaluating the scores [58]. But since no gain in sensitivity was observed for the wide precursor window of 0.5 Th and the identification results were better, both in number and delta scores, for the 0.02 Th tolerance to the recalibrated precursor m/z, the latter method will be used for all future searches throughout the present work.

5.3.2 Vectorization of fragment spectra

In order to compare spectra with vector-based operations, like the dot product, vectors of equal shapes need to be constructed from the query and the library spectra. Vectorization of spectra can be performed by dividing the m/z axis into small intervals, 'bins', and assigning every fragment signal to its closes bin. The bin size determines the effective resolution and therefore the fragment m/z tolerance. A vector can also be constructed from pair-wise peak matching, as performed in Pepitome and ANN-SoLo. This allows for the use of individual peak matching at adjustable tolerance but is computationally more expensive. In the following sections, the effect of fragment spectrum binning will be investigated.

5.3.2.1 Simulating the effect of m/z binning with theoretical fragment spectra

As a first step, the effect of spectrum m/z binning will be simulated for the most frequent canonical ions, the b and y ions. Simulated spectra were created from all peptide sequences stored in the PRIDE spectral library as series of b and y ions. 2,982,569 million unique theoretical b and y ions were created. Fragment ions were unique in that each ion represented exactly one fragment sequence.

All theoretical spectra were binned in a common m/z vector with a bin size of 0.05 to preserve high mass resolution. The histogram of all fragment signals is shown in Figure 11. Three areas were magnified for detailed inspection of the signal m/z values (Figure 12).

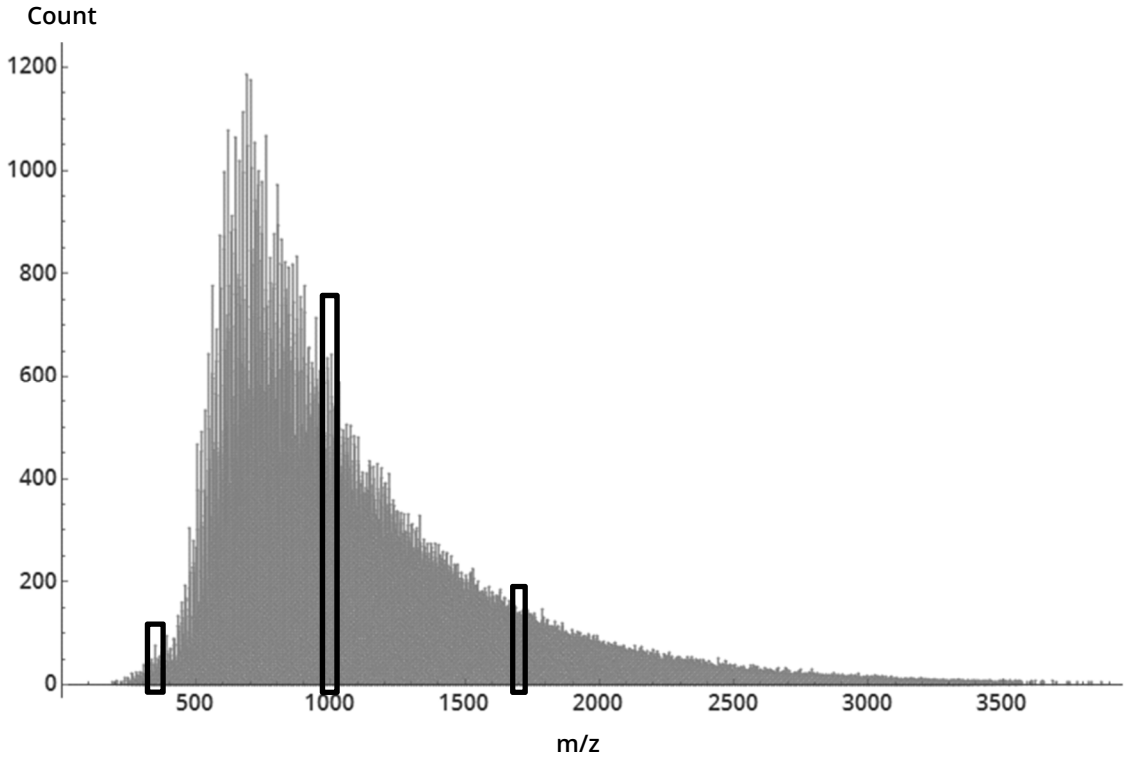

Figure 11: Histogram of m/z values of all simulated y ions at a resolution 0.05 Th. Y ions were created from all PRIDE Cluster peptide sequences. The marked regions are magnified in Figure 12.

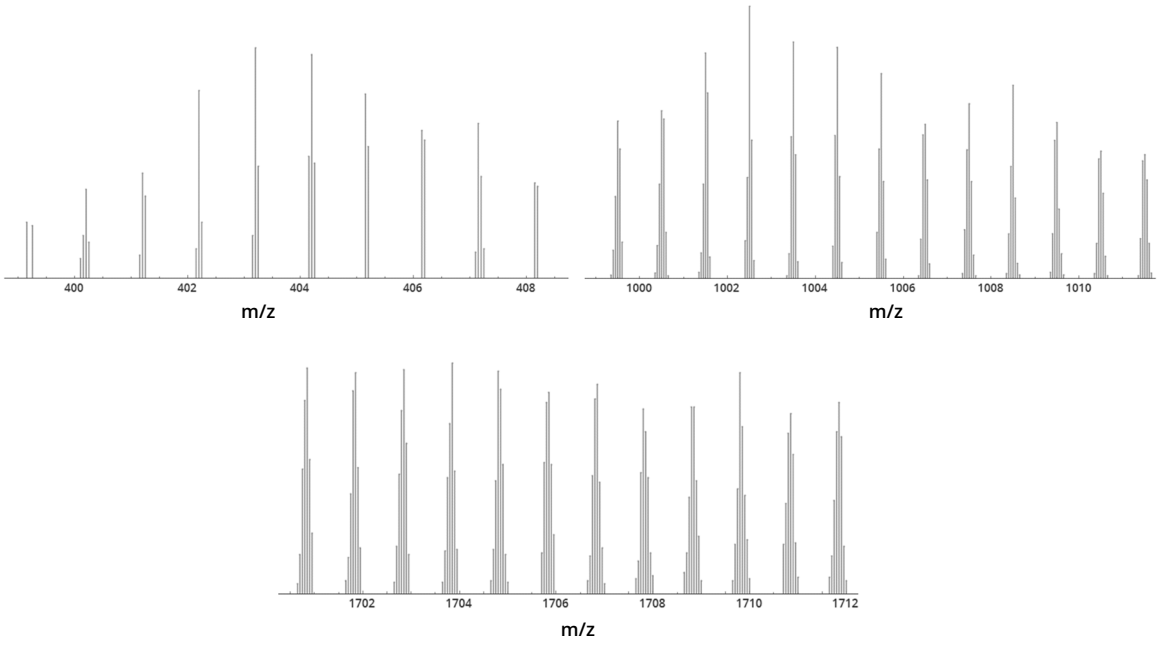

Figure 12: Magnification of three regions of the m/z histogram of all simulated y ions. The histogram depicts the number of m/z signals in bins of 0.05 Th among all simulated fragment spectra.

The overall distribution (Figure 11) yields a similar picture as the previously shown Figure 8, where the frequency of experimental fragment signals per m/z was plotted. Because every signal was only counted once in this analysis, the outlying signals in the < 500 Th m/z range disappeared. The overall shape is similar, however, which confirms that the theoretical b and y ions are a reasonable approximation of most of the experimental fragment signals.

From the magnified regions of the histogram (Figure 12) it is apparent that the distribution of theoretical fragments is strongly discontinuous at the sub-Th level. Signals are grouped in packages of approximately 1 Th steps ('whole-Dalton peaks'). It is important to note that this data is derived from calculated fragment masses and no calibration error is present. The distribution of signals around each whole-Dalton peak is therefore truly a property of the atomic constitution of the peptides.

The centers of these packages, however, are not exactly on the whole-numbered (integer) m/z values, but, for example, at m/z 400.2, m/z 1000.5 and m/z 1700.8. Apparently, signals around the 400 Th mark tend to have fractional parts of about .2, while the fractional parts of signals with m/z 1000 Th are around .5. When the signals around 1000 Th are divided into 1 Th bins, they may be assigned to either the 1000 Th or the 1001 Th bin, depending on small random errors in the experimental data. With the assignment to the wrong bin, the signal in the query spectrum will not be matched against the correct corresponding signal in the candidate spectrum and the calculated score will be less accurate.

5.3.2.2 Evaluation of m/z binning with experimental fragment spectra

To further estimate the effect of vectorization on the fragment spectra, all experimental (consensus) spectra from the PRIDE spectral library were processed. The fragment signals from all spectra were binned in a common m/z vector with a bin size of 0.05 and the distribution of signal frequencies is shown in Figure 13.

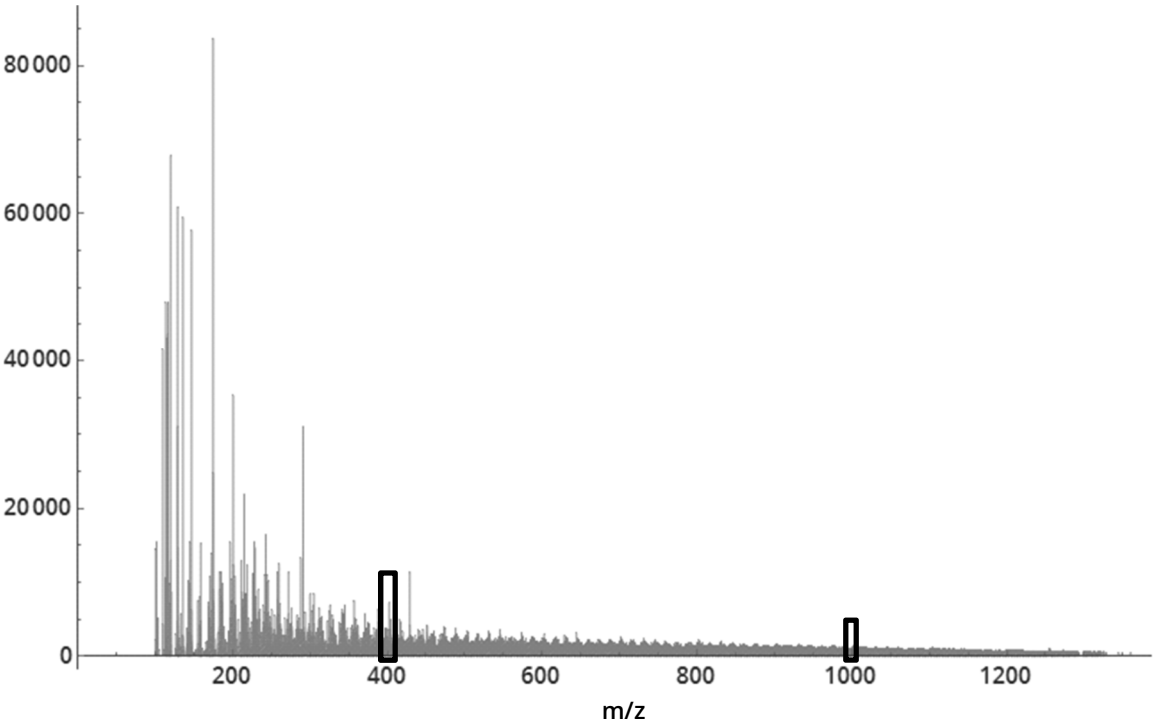

Figure 13: Histogram of m/z values of all experimental fragment ions at a resolution 0.05 Th. All fragment ions from all PRIDE Cluster spectra were included. The m/z range from 0 to 1400 Th is shown. The marked areas are magnified in Figure 14.

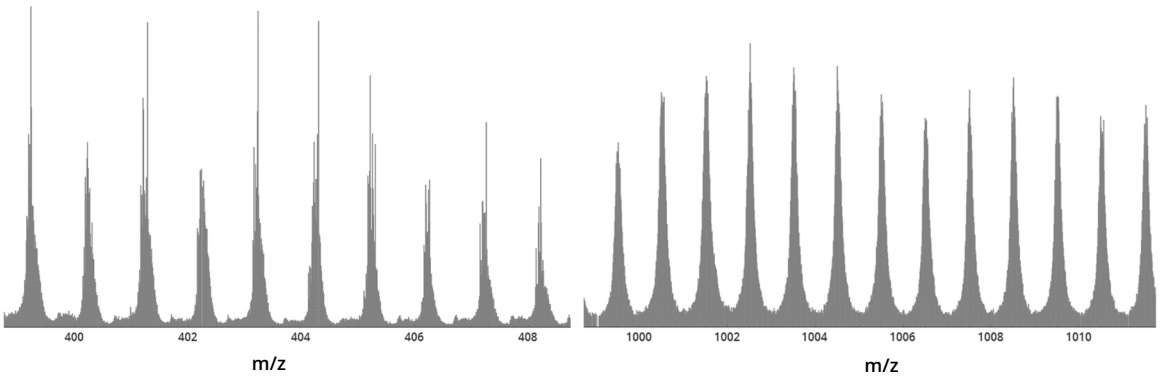

Figure 14: Magnification of two regions of the m/z histogram of all experimental fragment ions. The histogram depicts the number of m/z signals in bins of 0.05 Th among all fragment spectra in the PRIDE Cluster library.

The general signal distribution pattern that has been observed for the theoretical b and y ion series in the previous section can be confirmed in the experimental spectra (compare Figure 12

and Figure 14). Again, the majority of signals are grouped in packages spaced approximately 1 Th apart from each other. In contrast to the theoretical fragments, more signals are observed between those packages in the experimental spectra. Possible sources for the in-between signals may be b or y ions with a higher calibration error in the individual spectra, or other fragment signals that do not represent b or y ions.

5.3.3 Recalibration of the m/z axis

Even though the experimental spectra are more populated between the 1 Th peaks than the simulated spectra, it can be assumed that an exact alignment of the spectrum binning to the signal packages would enhance the binning accuracy considerably. Alignment of the bins to these peaks can be realized by creating a recalibration function which shifts all fragment m/z values to integer m/z values. Subsequent binning can then be performed in 1 Th steps.

A peak picking was performed on the histogram from the previous sections (Figure 13) to determine the positions of the signal packages. The fractional parts of the picked peaks were plotted against the m/z value (Figure 15).

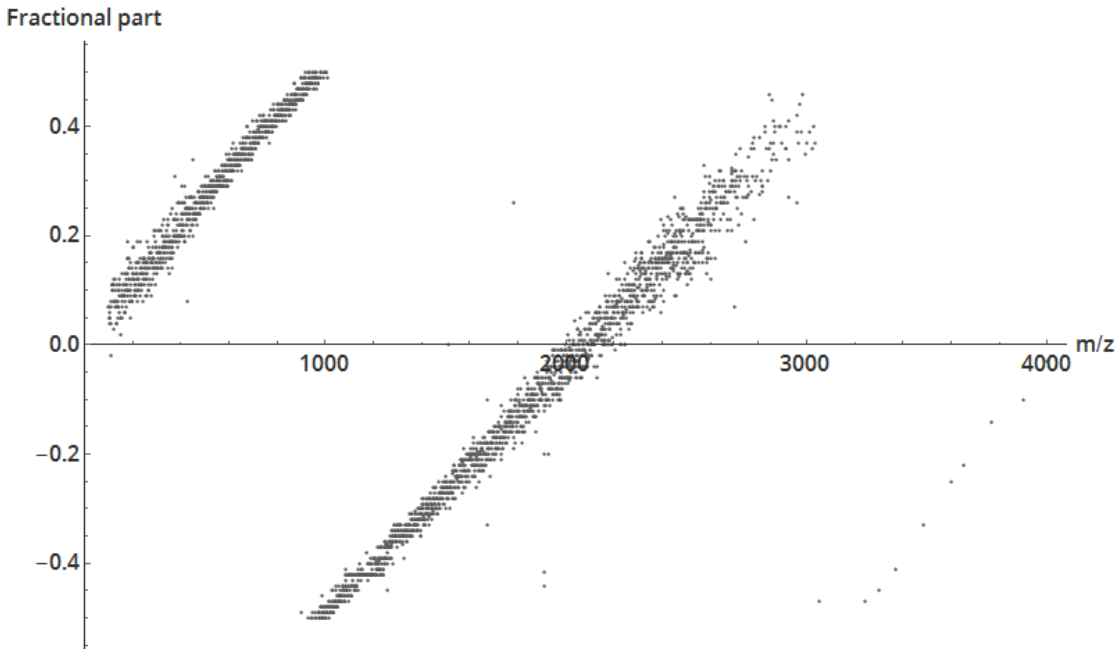

Figure 15: Fractional parts of the peaks picked in the experimental fragment m/z histogram against their m/z value. In principle, the experimental values plotted here represent the Kendrick mass defect of peptide fragments.

The distribution of fractional parts can be approximated by linear function. The 'wrap-around' was corrected for and the linear function was fitted in Figure 16.

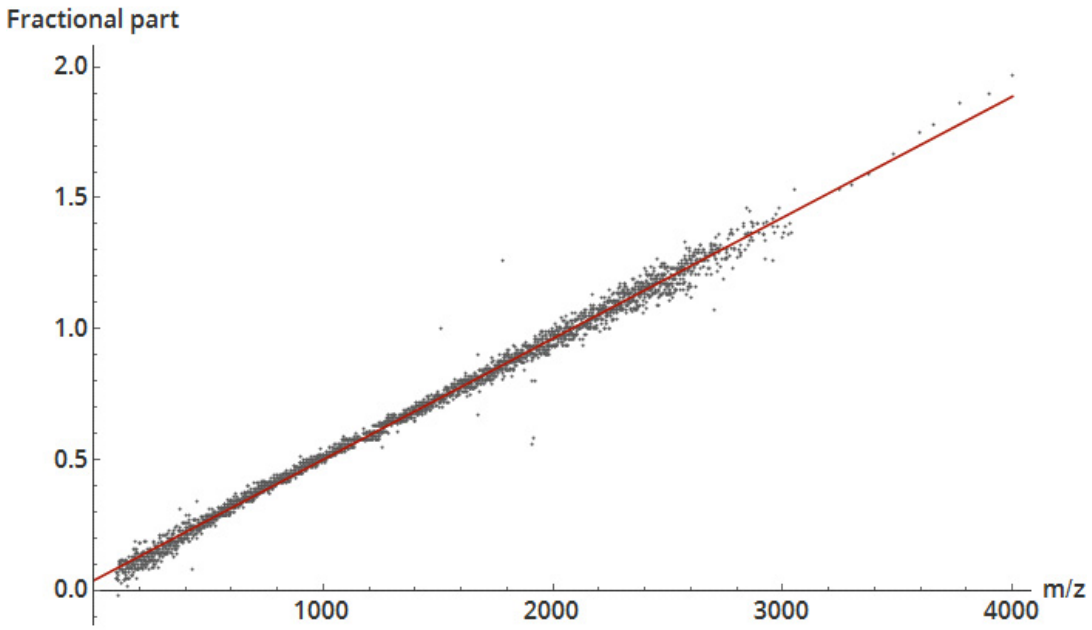

Figure 16: Shift-corrected fractional part of the m/z peaks and the linear fit. The fitted linear function (red line) was used as the common mass recalibration function.

The recalibration function was derived from the linear fit as:

$$m_{recalib}/z = 0.9995\ m_{exp}/z - 0.0388$$

The slope of 0.9995 can be regarded as an approximation of a Kendrick mass correction factor for peptides. It reflects the ratio of the nominal mass to the exact mass of the average atomic composition of peptides. Multiplication of any peptide fragment mass with this factor will approximate the fragment's exact mass to its nominal mass, e.g. 400.2 to 400, 1000.5 to 1000, etc.

5.3.3.1 Application of the recalibration function to theoretical fragment ions

To evaluate the accuracy of the empirical recalibration function, the m/z values of theoretical b and y fragment ions were recalibrated and the distribution of fractional parts ('mass defects') was plotted before and after recalibration (Figure 17).

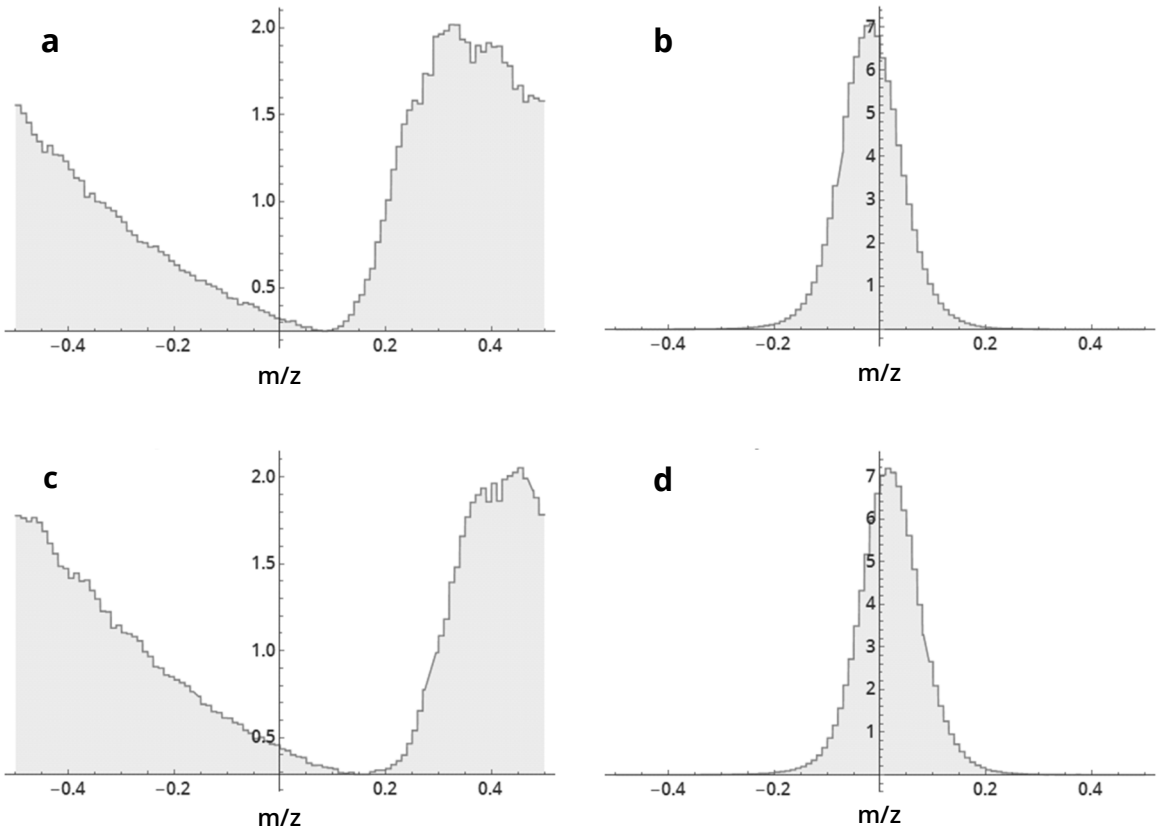

Figure 17: Distribution of fractional parts of the theoretical b ions (a, b) and y ions (c, d) before (a, c) and after (b, d) recalibration.

The fractional parts of the theoretical fragment masses were broadly distributed before recalibration. Specifically, the density was relatively high around +/- 0.5, which is exactly where the edges of the 1.0 Th bins are in vectorization. This may lead to false bin assignments of those signals. After recalibration, the fractional parts of the signals approximate a normal distribution around 0. Only 0.071% of the fragments lie outside the [-0.3; 0.3] interval. Therefore, it can be safely assumed that nearly all fragment signals from b and y ions will be assigned to correct bins after recalibration. The empirical recalibration function produces a very good fit of the theoretical fragment masses.

5.3.3.2 Application of the recalibration function to the experimental fragment ions

The same analysis was performed for all experimental fragment ions in the PRIDE Cluster spectral library. Figure 18 depicts the distribution of fractional parts of the experimental ions.

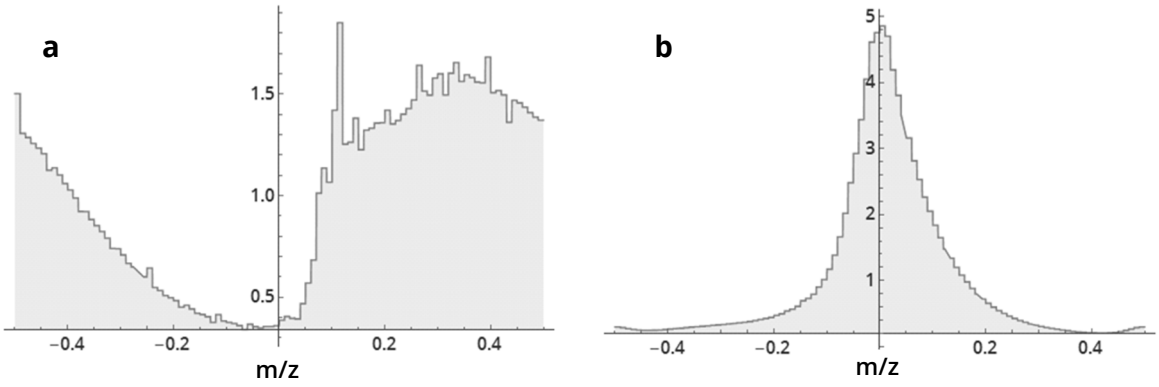

Figure 18: Distribution of fractional parts of the m/z values of the experimental fragment ions before (a) and after (b) recalibration.

As with the theoretical ions, the broad distribution of fractional parts could be condensed into a narrow, roughly normal distribution. 89,6% of all fragments lie in the [-0.3; 0.3] interval. It can be assumed that most fragment ions will be assigned to the correct whole-Dalton bin upon vectorization.

5.3.3.3 Recalibration of amino acid masses

The empirical recalibration procedure presented in the previous section relates to the concept of the Kendrick mass, where the exact masses of molecules from a specific class are approximated to an integer mass by setting the mass of the class's building blocks to an integer value.

For peptides, the building blocks that determine the masses of the precursor and their fragments are a) the amino acids, b) a water molecule, c) adducts like protons or sodium/potassium ions, d) fragmentation losses, and e) chemical modifications. Thus, when the recalibration function is applied to the individual masses of these components, the fractional parts of the recalibrated masses should be zero on average. Figure 19 shows the fractional part before and after recalibration of amino acids, common modifications, and protonated water.

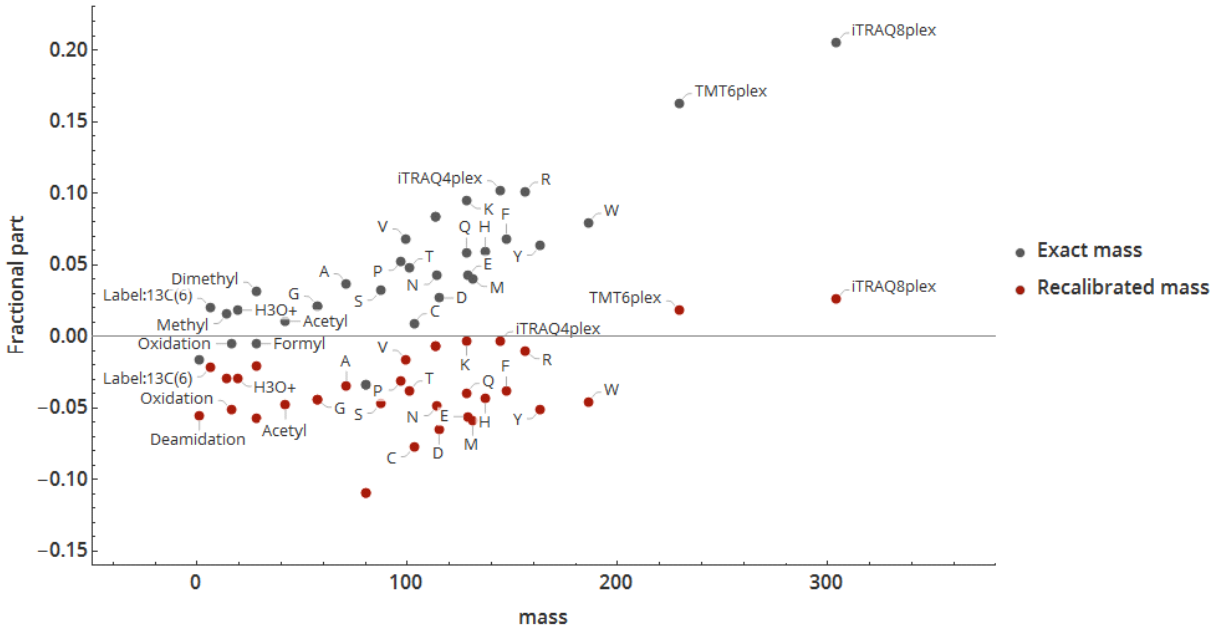

Figure 19: Fractional parts of amino acid and common modification monoisotopic masses before (gray) and after (red) recalibration. Masses are given for the amino acid residues as they occur in peptide bonds. Dots are labeled with the single letter code for amino acids and modification names as used in the PRIDE Cluster library. Unlabeled dots represent Phosphorylation (m = 79.966) and Deamidation (m = 0.984).

Before recalibration, the masses of all amino acids (gray dots) were above the nominal mass. For example, the alanine residue has an exact mass of 71.037 Da, so it is 0.037 above its nominal mass of 71. The fractional part increases as the mass of the amino acids increases. This phenomenon is a direct result of the atomic composition of the amino acids: hydrogen and nitrogen have positive fractional parts (1.008 and 14.003 resp.), carbon has zero (12.000, by definition), and oxygen has a negative fractional part (15.995).

After recalibration, the fractional parts of all amino acid masses are roughly equal and negative (red dots). Application of the recalibration function has successfully corrected for the slope of the fractional parts. But surprisingly, the amino acid masses were also shifted to negative values. This is unexpected since it seems to lead to a negative drift of fractional parts when recalibrating the spectra. An explanation might lie in the difference of fractional parts between b and y ions. From Figure 17 b and d it is evident that fractional parts of the b ions were a little lower than 0, while the fractional part of the y ions were slightly above 0. Because y ions are usually more prevalent in the spectra than b ions, the empirical recalibration function inherently corrects for the additional shift in y ions and therefore offsets the whole spectra by a negative value.

Another remarkable feature of the recalibrated masses is the offset of the TMT6plex and iTRAQ8plex modifications. TMT6plex (m = 221.163) has an unusual high fractional part with 0.163, which is reduced to 0.018 after recalibration. iTRAQ8plex is recalibrated from 304.205 to 304.026. 6% of the peptides in the PRIDE Cluster repository were modified with TMT6plex and 3.2% with iTRAQ8plex. While these peptides are not the majority, they can be considered outliers of the

recalibration fit. If so, it might be desirable to introduce additional recalibration functions which are specifically adjusted for TMT- or iTRAQ8plex-modified peptides.

5.3.3.4 Special role of isotopically labeled peptides

Separate recalibration functions were created for TMT6plex-modified and unmodified spectra (Figure 20).

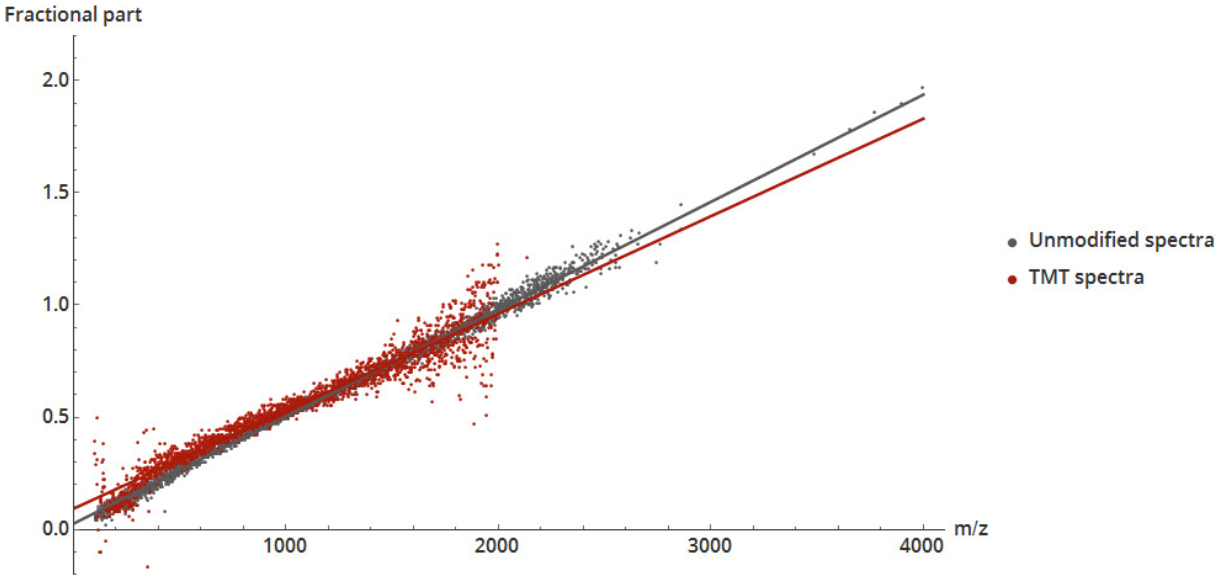

Figure 20: Fractional parts of fragments from TMT-modified (red) and unmodified (gray) spectra and the corresponding linear fits.

The linear fit of the two distributions lead to recalibration functions with slightly different slopes and offsets (Table 7). Also, there is a diversification of fractional parts towards the upper end of the fragment masses, between m/z 1400 and 2000. This is probably due to a higher noise level in the histogram data as the spectra get sparser.

Table 7: Slope and offset of the recalibration function derived from the linear fit of fractional parts.

Dataset	Slope	Offset
All spectra	0.99954	-0.03876
Spectra from unmodified peptides	0.99952	-0.02583
Spectra from TMT-modified peptides	0.99957	-0.09243

The recalibration function of the spectra from unmodified peptides did not change much compared to the previously determined recalibration function of all spectra. When only the spectra from TMT-modified peptides were used, the slope was a little higher and offset changed a little more (5th decimal place in the slope, 2nd decimal place in the offset). A similar effect can be expected for the spectra of peptides with the iTRAQ8plex modification.

Another possible source of shifting of mass defects may be introduced by higher isotopes of fragment signals in general. The most frequent heavy atoms, however, have only a minor effect (third decimal place) on the mass offset (Table 8). This is about 10-fold less than what has been observed for the amino acids, which affected the mass defect at the second decimal place.

Table 8: Difference between exact masses of atoms H, C, and N and their respective higher isotopes.

Isotopes	Mass difference
D vs. ^{1}H	1.0063 u
^{13}C vs. ^{12}C	1.0034 u
^{15}N vs. ^{14}N	0.9970 u

In summary, calculating the recalibration function specifically for TMT-modified peptides results in a different function that fits the fractional parts of the fragment masses slightly better. A similar effect can be expected for iTRAQ8plex-modified peptides. While these modification-specific recalibration functions will be a better approximation to the Kendrick mass of the modified peptides than the general-purpose recalibration function, it remains to be determined whether the small increase in accuracy has any effect on the identification rate in spectral library searching. This question will surely be interesting when TMT- or iTRAQ-modified peptides are of specific interest in a work on spectral library search engines, but it is beyond the scope of this thesis. The general recalibration function has shown to be a good approximation to all fragment signals regardless of the peptide modifications and will therefore be used for all spectra in subsequent analyses.

5.3.3.5 Identification performance of recalibrated spectra

The effect of recalibrating the spectra before binning has been simulated, but it remains to be determined how recalibration affects the performance of the spectral library search engine. The 2,000-peptide benchmark subset from the QEx-HeLa dataset was used to evaluate this effect. The search was performed with library and query spectra with original fragment masses vs. library and query spectra with all fragment masses recalibrated. The result is shown in Figure 21.

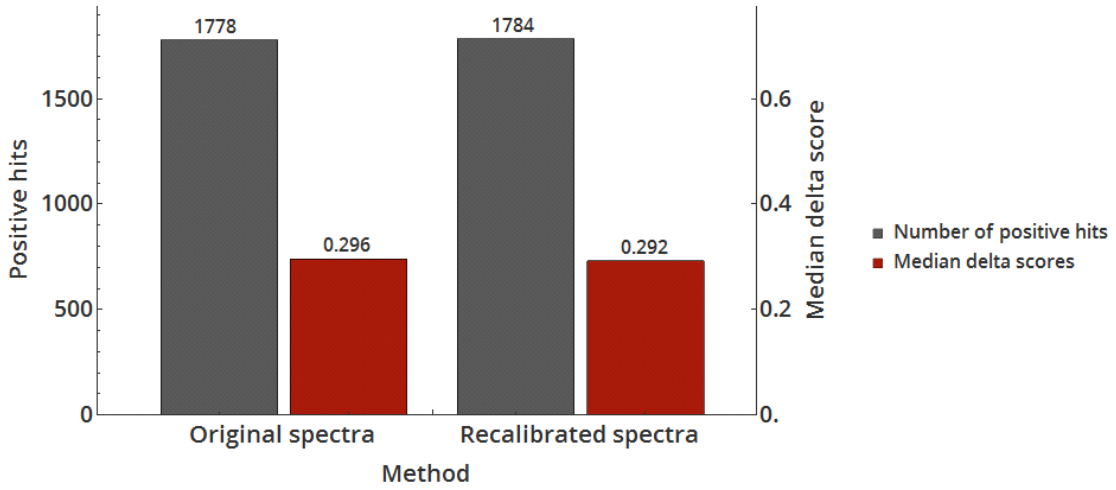

Figure 21: Positive hits and median delta scores from the spectral library search of 2,000 previously identified query spectra (by SequestHT) with original or recalibrated spectra. The Rank-Top150-CorrelationSimilarity method was used for intensity transformation and spectrum scoring, which will be discussed in later sections.

By using the recalibrated spectra, a small increase in positive hits (six additional hits) was observed. Median delta scores were a lower, but the change in score was only minor. While the effect is not dramatic, recalibration of spectra can be interpreted as one step towards an optimized spectral library identification method. All future searches will be performed on recalibrated spectra.

5.3.4 Additional parameters for the vectorization of fragment spectra

In addition to spectrum recalibration, two other methods which have been proposed to enhance the accuracy of vectorized spectra will be evaluated, the 'peak spreading' feature of SpectraST and the usage of bins smaller than 1 Th.

5.3.4.1 Peak spreading

SpectraST uses vectorization in 1 Th bins without recalibration and implements a feature named 'peak spreading', where an adjustable fraction (50% by default) of a bin's intensity is added to the adjacent bins. This approach compensates for binning errors but lowers the effective accuracy of peak matching. To compare the performance of both methods with the present data, SpectraST searches were performed with all very high confident peptides from the HeLa datasets against the PRIDE spectral library with *peak spreading* enabled or disabled. The results are summarized in Table 9.

Table 9: Number of positive hits in SpectraST searches with peak spreading enabled or disabled. All very high confident peptides from the two HeLa datasets were used. Precursor m/z tolerance was set to 0.02 Th.

Peak spreading	Query spectra	Positive hits	Positive hits %
0.5 (default)	22,352 from QEx-HeLa	18,625	83.3%
0.0 (disabled)	22,352 from QEx-HeLa	19,093	85.4%
0.5 (default)	11,452 from Fus-HeLa	10,506	91.7%
0.0 (disabled)	11,452 from Fus-HeLa	10,561	92.2%

By disabling the *peak spreading* feature, the number of correctly identified peptides in the QEx-HeLa dataset increased by 2.1% from 18,625 to 19,093. For the Fus-HeLa dataset, a 0.5% increase was observed. It is safe to assume that *peak spreading* had been added to SpectraST as a feature to enhance spectrum identification, and it is enabled by default. Its inferior identification performance in the two HeLa datasets may therefore seem unintuitive.

A possible explanation might lie in the higher mass accuracy of both the HeLa datasets and the spectral library compared to the data available in 2007 when SpectraST was originally published. For example, peptide identification from the QEx-HeLa data, which had been acquired on an Orbitrap Q Exactive – a high-resolution, high-mass accuracy mass analyzer –, is usually performed with a fragment mass tolerance of 10 ppm. A mass accuracy in this scale may render the addition of peak intensities to adjacent bins unnecessary and counter-productive for the resulting lower peak-matching accuracy. This is supported by the observation that the effect of *peak spreading* was less pronounced for the Fus-HeLa dataset, where the MS2 spectra were acquired on an ion trap mass analyzer. Ion trap data is usually searched with 0.6 Da fragment mass tolerances in our lab to account for its significantly lower mass accuracy. Disabling of the *peak spreading* feature still resulted in a slightly better identification rate for the HeLa-Fus data.

In view of these results, *peak spreading* was not added as a feature to spectral library searching in this work. Moreover, the recalibration procedure established in the previous section reduces the likelihood of binning errors in the first place, so spreading of peak intensities will be even less necessary.

5.3.4.2 Bin width

Bins of 1 Th width have shown to be a good approximation of the distribution of fragment signals. Still, the choice of smaller bin sizes would allow for spectrum-spectrum comparisons at a higher resolution and may be beneficial for the scoring of SSMs and hence the number of correct identifications. To test smaller bins but still having a 'center' bin in steps on 1 Th, a width of 1/3 Th was chosen. This divides each 1 Th window in three parts, a 'center' bin, where the majority of fragment signals still fall into, and the left and right side to the center bin (compare Figure 18 b).

The 2,000-peptide benchmark dataset was searched at 1 Th and 1/3 Th bin size. Results are summarized in Figure 22.

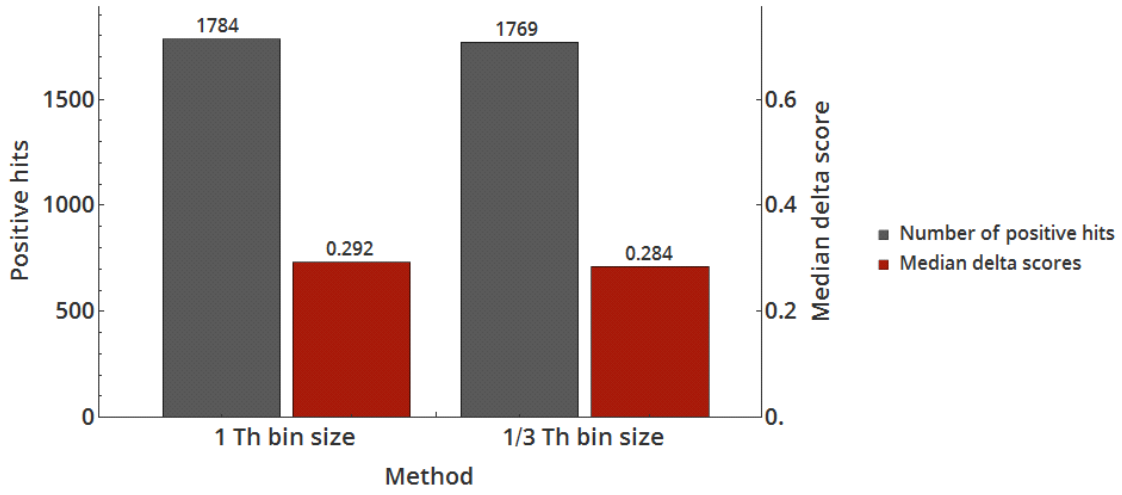

Figure 22: Positive hits and median delta scores from the spectral library search of 2,000 previously identified query spectra (by SequestHT) with vectorization bin sizes of 1.0 Th and 1/3 Th. The Rank-Top150-CorrelationSimilarity method was used for intensity transformation and spectrum scoring.

Although smaller bins of 1/3 Th lead to a 3-fold resolution increase for spectrum-spectrum matching, the search result was inferior to the search performed with 1 Th bins. Both the number of correct matches and the median scores were lower. The bin size of 1.0 Th is therefore regarded a good trade-off for matching the peaks with sufficient accuracy while tolerating small mass differences between library and query spectra, which naturally occur due to the diversity and acquisition parameters of the experimental library spectra. This work will use 1 Th bins for subsequent analyses.

5.3.5 Spectrum-spectrum match score calculation

Calculating a score for a spectrum-spectrum match as a measure of similarity of the two can involve intensity transformation and the actual scoring function. Two scoring functions and multiple transformations were evaluated with the benchmark dataset.

5.3.5.1 Scoring function

The most common scoring function is the dot product of the normalized vectors. Since it equals the cosine angle between the two vectors, it is termed 'cosine similarity', derived from the 'CosineDistance' function in Mathematica and calculated as 1 - CosineDistance. An extension of this function is the 'correlation similarity' (calculated as 1 - CorrelationDistance), which shifts the two vectors by their means before calculating the CosineDistance. The performance of the two score functions with respect to the number of identifications was tested with the benchmark dataset (Figure 23).

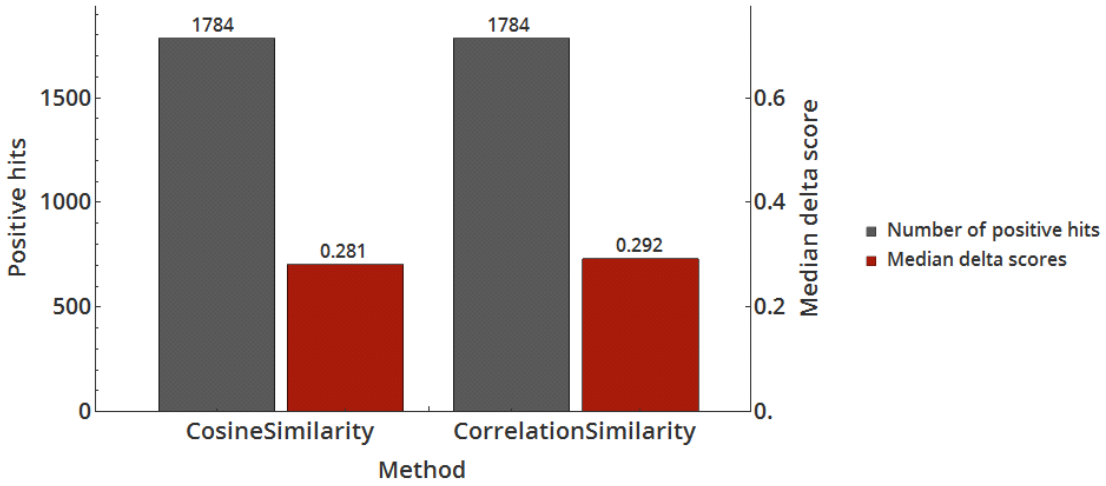

Figure 23: Number of positive hits and median delta scores for the spectral library search of 2,000 benchmark peptides. The Rank-Top150 intensity transformation method was used, which will be discussed in the following section. When the highest scoring candidate had the correct sequence, the match was counted as a 'true positive'.

The two scoring functions yielded the same number of peptide hits, which is not surprising since they both apply the dot product to the spectrum pairs in a similar way. Higher delta scores were achieved for the 'correlation similarity' method, indicating better separation of positive and negative hits, so this function will be used for scoring in subsequent analyses.

5.3.5.2 Transformation of fragment ion intensities

Various methods of intensity transformation were suggested and used for spectral library searching, with the most common being the square root and rank transformation. The 2,000 benchmark peptides were identified against the PRIDE Cluster spectral library with different methods of intensity transformation before scoring. The results of various methods are compared in Figure 24.

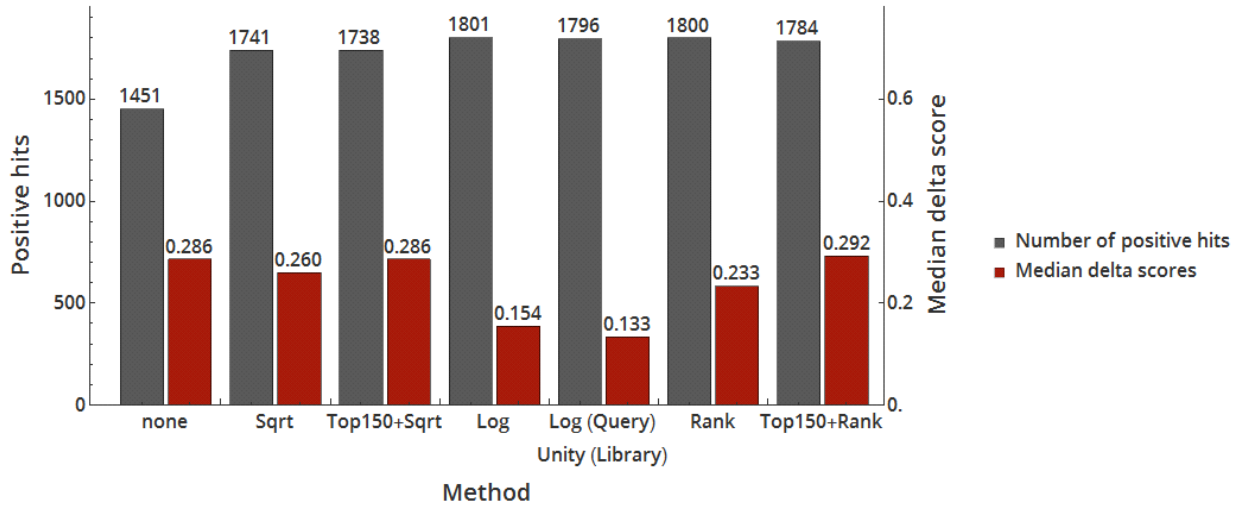

Figure 24: Identification results for various intensity transformation functions which were applied to both the library and the query peptides. The number of positive hits and the median delta score are plotted. The correlation similarity was used as the scoring function. Transformation functions were: Sqrt = Square root, Log = Logarithm and normalization to the median peak, Rank = Rank transformation, Unity = Intensity set to 1 for all peaks, 0 for baseline. Top-150 = Remove all but the top 150 peaks.

For 150 out of 2,000 query peptides (7.5%), no corresponding spectrum was found in the library. From the remaining 1,850 peptides, 1,451 were assigned to the correct peptide sequence when no intensity transformation was applied. Square root transformation increased the number of positive hits to 1,741, and log and rank transformation achieved the highest numbers of identifications with 1,801 and 1,800, respectively. Filtering of the spectra for the top 150 fragment signals decreased the numbers a little but improved the delta scores for both Sqrt and Rank transformation. Higher delta scores allow for better separation of true and false positive hits upon hit validation. The Top150+Rank method was therefore selected for subsequent searches, since it yielded the highest delta scores and only slightly fewer identifications than the top-scoring methods.

An interesting result was observed when library spectra were unitized, i. e. signal intensities of all peaks were replaced with 1 and all baseline intensities were kept at 0. Despite the simplification of the library spectra, 1,796 peptides were still correctly identified, albeit with lower delta scores. This phenomenon may be interpreted with the help of the 'number of unexplained intense peaks' hypothesis. When the candidate spectrum is transformed to a vector of 0 and 1 and the dot product with the query spectrum is calculated, only those peaks in the query spectrum are added to the score that have a 1 at the corresponding m/z value in the candidate spectrum. Conversely, query spectrum signals that have no corresponding signal in the candidate spectrum will not contribute to the dot product. Since the final dot product is normalized to the total intensity, higher numbers of unexplained query peaks will result in a lower score. It is notable that the concept of unitized intensities has also been used in sequence database search engines. The original version of SEQUEST assigned a constant intensity of 50 to all predicted b and y ions [42]. These observations suggest that the presence or absence of signals at specific m/z values are of

much more importance than the actual intensities, provided that the spectra have been sufficiently noise-filtered.

5.3.6 Search of all very high confident HeLa peptides with the correlation similarity method and SpectraST

In the previous sections, a spectral library search method has been established which involves spectrum recalibration, filtering for the top 150 signals, rank transformation of the signal intensities, and correlation similarity scoring of the spectrum-spectrum matches.

Another possible processing step prior to the identification pipeline is the condensation of replicate entries in a spectral library into consensus spectra. This procedure has been suggested, among others, by the authors of SpectraST [30] and is a prerequisite for SpectraST to construct decoy libraries, which will be addressed in later sections. As pointed out previously, the PRIDE Cluster spectral library was created by clustering MS2 spectra by their spectral similarity. A consensus spectrum was generated for every cluster of similar spectra, where at least 70% of the identifications agreed. Thus, for any sequence-charge-modification combination, multiple spectra can exist in the spectral library. It needs to be tested whether another iteration of consensus reduction is beneficial or detrimental for peptide identifications.

All very high confident peptides from the two HeLa datasets (22,352 for QEx-HeLa, 11,452 for Fus-HeLa) were searched against the complete PRIDE spectral library by SpectraST and the correlation similarity method established by this work. Another SpectraST search was performed against a consensus version of the library. Table 10 summarizes the results.

Table 10: Search all very high confident peptides from the two HeLa datasets against the complete or consensus PRIDE spectral library with SpectraST and the correlation similarity method. Correct identifications ('correct IDs') were counted as IDs that agree with the 0.1% FDR peptides identified by SequestHT. Number of misses was determined from the output of the correlation similarity (CS) search engine. A query was counted as a 'miss' when no candidate spectrum of correct identity was present in the database within the given precursor m/z tolerance. The identification rate was calculated with the number query spectra that were represented by at least on corresponding library spectrum (the 'non-misses') as the 100% reference.

Method and library	Query dataset	Found in the library	Misses %	Correct IDs	ID rate %
SpectraST against consensus	QEx-HeLa (22,352 identified spectra)	20,305	9.2%	10,131	49.9%
SpectraST against all				19,093	94.0%
CS against all				19,415	95.6%
SpectraST against consensus	Fus-HeLa (11,451 identified spectra)	10,797	5.7%	5,460	50.6%
SpectraST against all				10,561	97.8%
CS against all				10,646	98.6%

Searching against the consensus version of the PRIDE Cluster library yielded much lower identification rates with the SpectraST search than the complete library. The second iteration of consensus spectra building (with the first being the original PRIDE Cluster method itself) did not preserve a sufficient number of representative spectra for each peptide to achieve good sensitivity. Replicate spectra in the library exist because they did not merge with the other spectra of the same peptide upon clustering for spectral similarity. Possible reasons include the use of different mass analyzers or presence of chimeric spectra, which contain fragment signals from co-eluting peptides with a close precursor m/z. These results suggest that higher identification rates can be achieved by searching against a spectral library that contains multiple (dissimilar) spectra which represent the same peptide.

Searching the HeLa datasets against the original PRIDE spectral library yielded good identification rates of 94.0% for the Q Exactive dataset and 97.8% for the Fusion dataset among the peptides that were included in the spectral library. The CorrelationSimilarity (CS) scoring method achieved minor improvements of 1.6% and 0.8% over SpectraST (+322 peptides for QEx-HeLa, +85 peptides for Fus-HeLa). Despite optimization of various processing steps, a small percentage of the very high confident peptides in the two HeLa datasets remain unidentified by the present search engine (4.4% for the QEx dataset, 1.4% for the Fus dataset). Further enhancement of the scoring method will be performed in the machine learning section of this work.

It is notable that peptides which were missing in the library accounted for most of the false identifications of the search engines. For the QEx-HeLa dataset, 890 spectra (4.0%) were assigned to wrong peptides by the CS scoring method although the correct peptide was part of the library, but 2,047 (9.2%) could not be found in the library at all. Likewise, wrong assignments accounted for 151 (1.3%) of false hits in the Fus-Hela datasets, but complete misses for 655 (5.7%).

These results give insight about the performance of the search engines on the very high confident peptides that had been previously identified with SequestHT. But spectral search engines are of particular interest for the identification of the lower quality spectra which cannot be readily identified with sequence database search engines. To compare the overall performance of the search engines without prior ID knowledge, a validation step, which controls the false discovery rate, has to follow the search. The decoy spectrum generation and hit validation will be performed in later sections.

5.3.7 Peptides not in the library

The significant number of peptides which were not in the library limited the identification rate of the spectral search engines. Out of the 22,352 very high confident peptides from the QEx-HeLa that were subjected to spectral library searching, 2,047 (9.2%) were not found in the library and therefore could not be identified in the first place. Table 11 classifies the causes of peptides misses, which could be a) the sequence was not found at all, b) the sequence was found but with the wrong charge states, c) the sequence was found but with wrong modifications.

Table 11: Number and percentage of very high confident peptides from the QEx-HeLa dataset that were not found in the PRIDE spectral library for either reason. Numbers add up to more than 2,047 due to peptides where both charge state and modification configuration did not match.

Cause of miss	Number	Percentage
Sequence not found	691	33.8%
Charge not found	485	23.7%
Modification not found	1,066	52.1%

About 1/3 of the very high confident query peptides did not have a library spectrum with the correct sequence. For 23.7% of the queries, a peptide with correct sequence and modifications was present, but the charge state did not match. Likewise, more than half of the queries had a modification configuration which was not present in the library, although entries with the correct sequence and charge were present.

The fact that 9.2% of the query peptides could not be identified with spectral library searching stresses the importance of the completeness of the spectral library. Besides the 'natural growth' of spectral libraries built from online repositories as more experimental data becomes available and incorporated, several approaches have been suggested to enhance the coverage of spectral

library searching. The *ANN-SoLo* search engine implements an 'open modification search' for spectral library identification [23]. First, a large m/z tolerance is used for precursor selection. Secondly, a specialized scoring function, which accounts for peak shifts caused by modifications, has been designed. The authors demonstrated an increase in identified spectra from 4,141 to 6,019 due to the open modification search strategy. The most prevalent modifications were oxidation, amidation and substitution of glutamine for pyro glutamic acid, followed by ammonium adduction and aminoethylbenzenesulfonylation. Many hits also resulted from fragmentation of higher isotopic peaks instead of the monoisotopic peak of a precursor, which would be have been missed if a narrow precursor m/z tolerance was used. Originally developed for sequence database search engines, the open modification search has been demonstrated to be an effective strategy for spectral library search. It can enhance the coverage of identifications and may be of particular interest when unusual modifications, which are rarely included in public spectral libraries, are searched for.

However, for the 691 peptides that were not represented in the library spectrum by any spectrum with the correct sequence, identification would not have been possible even when this strategy had been applied.

5.4 Decoy spectrum generation

Decoy spectra are the most popular method for the simulation of random hits to estimate the false discovery rate. For an accurate estimation of the FDR, the generated decoy spectra should produce truly random matches, i. e. they should match a query spectrum with the same likelihood as the true-negative target candidates.

Three different methods were tested for decoy spectrum generation and evaluated with the QEx-HeLa dataset. The 2,000 benchmark peptides were run against the target library (original spectra) and the decoy library (artificial spectra). The 'target-decoy delta score', the difference between the best negative target score and the best decoy score, was calculated for each query spectrum to test the quality of the decoy spectra. Ideally, the target-decoy delta score should be evenly distributed around 0 and yield positive and negative values in a 1:1 ratio, i. e. exactly 50% of all hits should be decoys.

5.4.1 Intensity shuffle method

The *intensity shuffle* method randomizes the intensities of all fragment signals within a spectrum. The m/z values remain unchanged. Figure 25 plots the distribution of decoy delta scores.

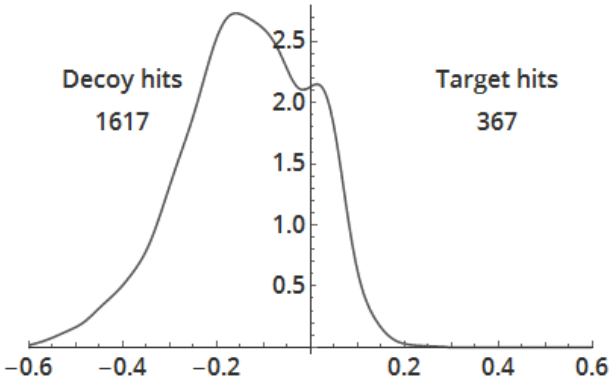

Figure 25: Distribution of the 'target-decoy delta score' of 2,000 QEx-HeLa benchmark peptides with the intensity shuffle method.

The *intensity shuffle* method produced more decoy than target hits (81.5%). Even after shuffling the intensities around, the decoy spectra still too closely resembled the original spectra and therefore produced better scores on average than the random target hits. Shuffling of the fragment signal intensities is therefore not sufficient to generate decoy spectra from experimental library spectra, which is supported by the previous observation that the intensity information of the fragment ions is of secondary importance for spectrum matching compared to the m/z values.

5.4.2 m/z randomization method

The m/z randomization method replaces an adjustable fraction of signals to a random position in the spectrum. Unlike the *shuffle and reposition* method used in SpectraST, *m/z randomization* does not respect fragment ion annotation but simply repositions an adjustable fraction of signals randomly. Resulting 'target-decoy delta scores' are plotted in Figure 26.

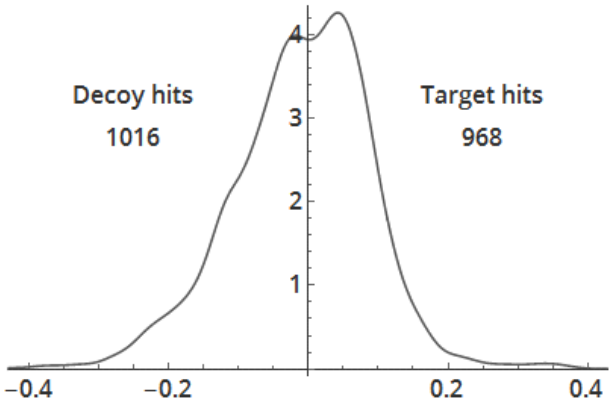

Figure 26: Distribution of the 'target-decoy delta score' of 2,000 QEx-HeLa benchmark peptides with the m/z randomization method at a randomization rate of 60%.

Repositioning 60% of the fragment signals to a random m/z value was found to produce decoys that match the query spectra with similar scores as the true negative targets on average. While the fraction of decoy matches (51.2%) suggest a good quality, the score distribution is skewed to one side and the variance is higher than desirable. This indicates unevenness among the decoy spectra – while they produce roughly equal numbers of matches on average, some decoys will

match the query spectra much better than random hits, some much worse. Higher score variance requires stricter thresholds upon validation to keep the FDR at the desired level, so the overall number of validated hits will be reduced.

5.4.3 Precursor shuffle method

The *precursor shuffle* method randomly assigns a precursor m/z from all other precursors within a predefined range. The present implementation assigns precursors within a window of +/- 1.0 Th but excludes the +/- 0.1 Th window to avoid matching a library entry to a replicate of itself. Figure 27 shows the distribution of the 'target-decoy delta scores' for the *precursor shuffle* method.

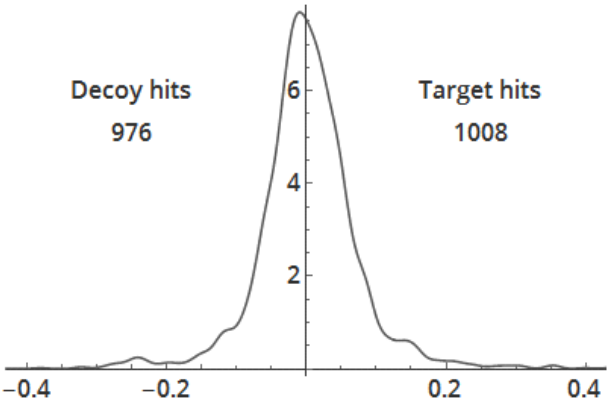

Figure 27: Distribution of the 'target-decoy delta score' of 2,000 QEx-HeLa benchmark peptides with the *precursor shuffle* method.

Decoy spectra performed almost randomly by matching 49.2% of the queries. Also, the distribution of the decoy delta scores is confined and symmetric. This suggests that the decoys generated by the *precursor shuffle* method provide a good estimation of the random hits.

The original implementation of the *precursor swap* method by Cheng 2013 produced decoy match rates between 45 and 49% [48], depending on the dataset. The *precursor shuffle* method can be regarded a modification to the *precursor swap* method by assigning different precursor randomly from near entries instead of swapping them around. This avoids the need to find an even number of spectra to form pairs and allowed for a narrower m/z inclusion window of +/- 1 Th, with an exclusion window of +/- 0.1 Th, compared to the exclusion window of +/- 8 Th with the *precursor swap* method. A narrower range for re-assigning precursor m/z values may yield spectra which are more similar, yet represent different peptides, and therefore increase the quality of decoy library. Unlike with swapping, the *precursor shuffle* method does not guarantee that every precursor will have exactly one counterpart in the decoy library. However, no adverse effect has been observed on the benchmark dataset of 2,000 peptides. For large libraries with many precursor entries and a dense distribution of precursor masses, such as the PRIDE library, this is probably not an issue.

The *precursor shuffle* method will be used to generate the decoy spectra for FDR estimation in this work.

5.4.4 Decoy spectrum generation with SpectraST

SpectraST 5.0 provides the *shuffle and reposition* and the *precursor swap* method for decoy spectrum generation. With the *precursor swap* method, however, the SpectraST would crash every time halfway in the process, so the method could not be used. It would have been interesting to compare the *precursor swap'ped* decoy spectra with the *precursor shuffle* method from this work, as the latter is an extension to the first.

A main limitation in SpectraST is that decoy spectrum generation only works with unique (consensus) libraries. Attempting to generate decoys with non-unique libraries will produce an error message. It is not clear why the authors chose to impose this limitation, as decoy spectra can be generated from non-unique libraries as well (as demonstrated in this work). It might reflect the authors' general philosophy of working only with unique libraries when doing spectral searching [59]. Notably, the authors of the PRIDE Cluster study experienced this issue as well when working with SpectraST [29] (supplementary note 5).

The SpectraST search with decoy generation and validation could therefore only be performed with the consensus PRIDE library and the *shuffle and reposition* method. Importantly, the decoys generated from the consensus library may only be used for FDR validation when the same consensus library was used for the main search. They cannot be used as decoys for the validation of searches with the complete library, since the search space is much larger, and the FDR would be underestimated otherwise.

5.5 Machine learning for advanced method optimization

Spectrum-spectrum scoring functions generally calculate the scores from the fragment signals of a spectrum pair with respect to the signals' intensities. More intense signals have a higher contribution to the score and vice-versa, and the magnitude of the contribution is altered by the intensity transformation function used.

In this work, a scoring function that contains adjustable weights for all m/z positions will be implemented. The underlying premise is that the discriminative power of signals at different m/z values is not equal. The 'weighted correlation similarity' scoring function is an extension to the previously used 'correlation similarity' which multiplies the intensity vectors of the query spectrum and the candidate spectrum each with a learnable weight vector before calculating the correlation similarity.

5.5.1 Implementation of a 'weighted correlation similarity' scoring function

Two neural nets were constructed to learn optimal weight vectors for query spectra and the candidate spectra: a scoring net and a training net.

The scoring net serves as a replacement of the scoring function. It takes two vectors as input, a query and a candidate spectrum. The query spectrum is multiplied with the query weight vector and the candidate spectrum with the candidate weight vector, and the correlation similarity is calculated between the two.

The training net wraps around the scoring net to train it. For each query spectrum, it takes all positive query-candidate pairs and all negative query-candidate pairs, invokes the scoring net for each pair, and finally calculates the loss as the difference between the highest-scoring positive pair (as the difference to 1) and the highest-scoring negative pair. This is effectively the contrastive loss calculated from the delta score.

5.5.1.1 Neural net training

The neural net was trained on a training set of 4,000 query spectra from the very high confident peptides in the QEx-HeLa dataset. The training net was designed to take the all candidates with known outcome (true or false match) for a query as input, present the spectrum pairs to the scoring net, and calculate the contrastive loss between the best positive SSM and the best negative SSM. In 20 min of training, the loss was reduced from the initial value of 0.67 to 0.60 after 5 rounds with 1013 batches x 32 inputs. Here, every input was a package of all candidate spectra for one query.

5.5.1.2 Neural net scoring in training and validation datasets

To test the performance of the neural net after training, the scores calculated by the trained neural net (weighted correlation similarity) were compared to the (unweighted) correlation similarity scores for the training and the validation dataset. Higher scores are favorable for positive matches, lower scores for negative matches. Reduction of tailing leads to higher overall identifications because the score threshold can be lowered while staying below the target FDR. Score distributions are depicted in Figure 28 for the training dataset and in Figure 29 for the validation dataset.

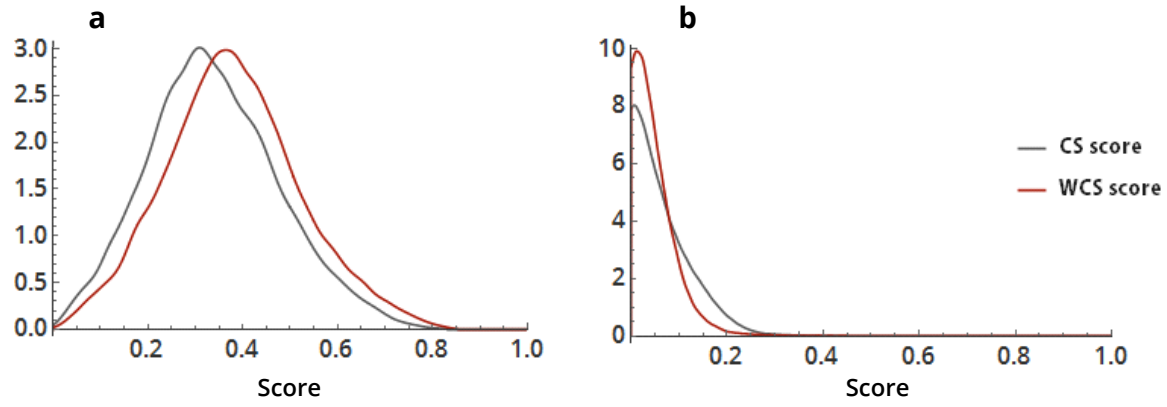

Figure 28: Score distributions in the training dataset of correlation similarity (gray) and the weighted correlation similarity (red) for positive (a) and negative (b) spectrum-spectrum matches.

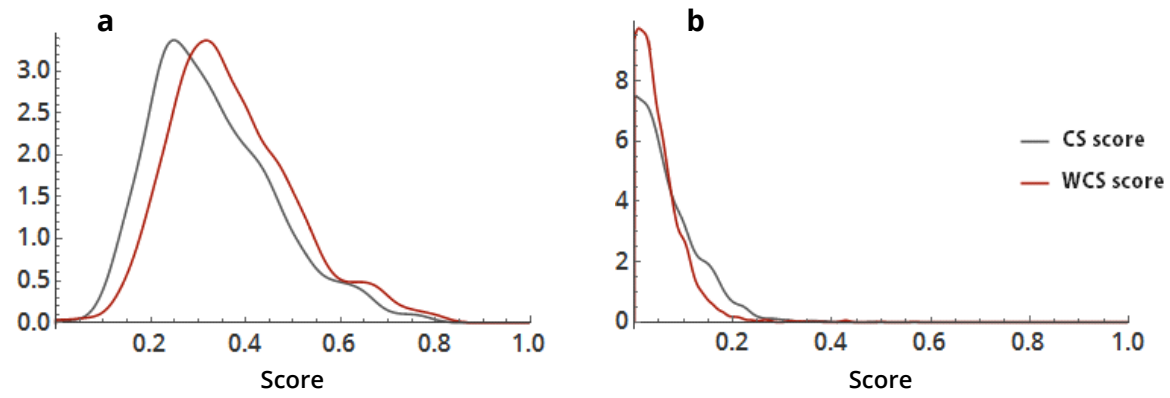

Figure 29: Score distributions in the validation dataset of correlation similarity (gray) and the weighted correlation similarity (red) for positive (a) and negative (b) spectrum-spectrum matches.

The weighted scoring function produced better scores than the unweighted CS scoring function in training and validation datasets. Positives scores were shifted to the right, and the tailing of negative scores towards higher values has been reduced. Also, the scores from the validation dataset are nearly as good as in the training dataset, indicating that the WCS scoring function also performs well on unseen spectra.

5.5.1.3 Weight vectors learned by the neural net

The weight vectors can be plotted to visualize how the neural net has learned to produce better scores after the training. Figure 30 depicts the weights for all m/z values for the query and candidate spectra.

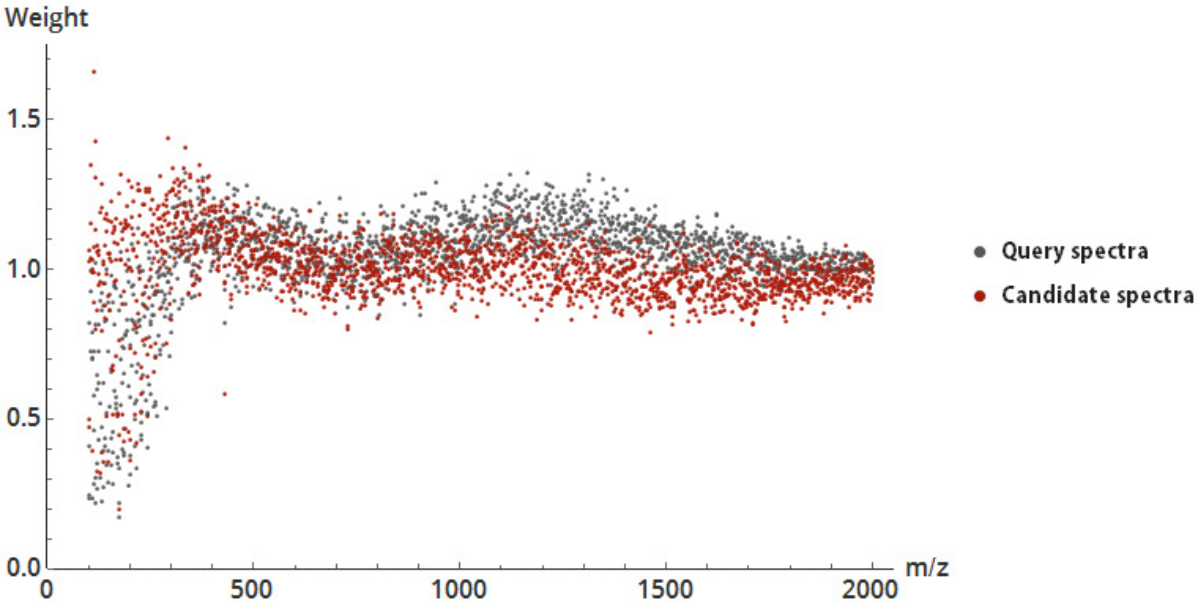

Figure 30: Weights for all m/z values in the query and the candidate weight vector, as learned by the neural net. Query spectrum weights: Min = 0.173 at m/z 175, Median = 1.066, Max = 1.322 at m/z 333. Candidate spectrum weights: Min = 0.198 at m/z 175, Median = 0.998, Max = 1.663 at m/z 114.

In Figure 31, weights were averaged to better visualize the overall trend of each vector.

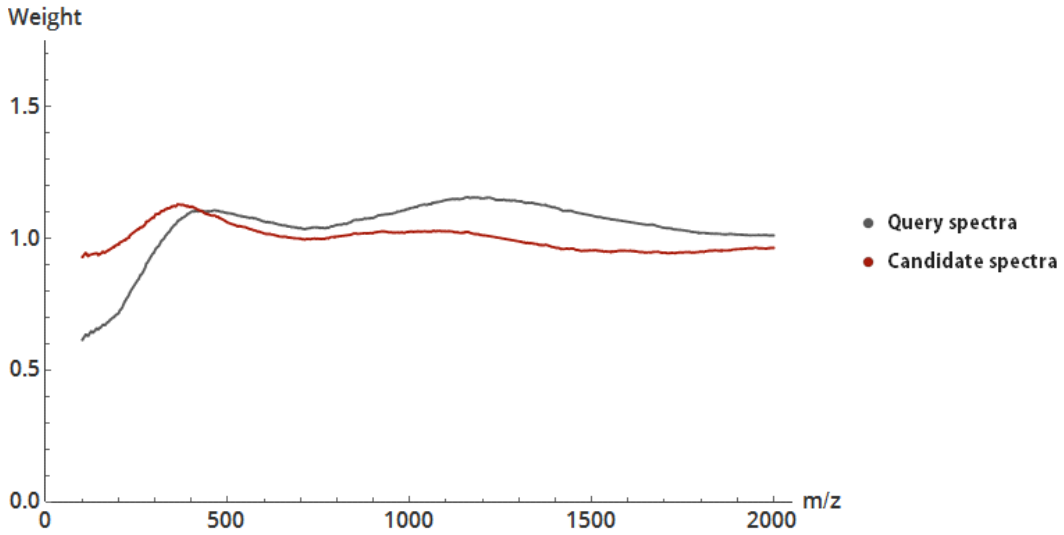

Figure 31: Weights for all m/z values in the query and the candidate weight vector, averaged with a mean filter of 100 Th width.

For the candidate spectra, the weights average around 1.0 throughout the whole m/z range, but individual weights are as low as 0.198. For the query spectra, significant weight adjustments can be found in the range of 100 to 500 Th. The weights tend to be smaller than average, meaning that fragment ion intensities in this m/z range will be downscaled before scoring.

The weights learned by the neural net can be interpreted as a measure of the discriminative power of fragment signals at every m/z position. The descent towards the lower m/z range coincides with previous observation that the lower end of the m/z range is highly populated with fragment signals throughout all spectra, both in the library and the query spectra. These frequent and intense signals tend to dominate the score although they may not help to distinguish a true from a false spectrum-spectrum match. Specifically, the almost ubiquitous y1 fragment ions of tryptic peptides at m/z 175 and m/z 147 received weights of 0.173 and 0.289 in the query spectrum weight vector, respectively. As a result, those signals will only contribute 17.3% (resp. 28.9%) of their original intensity to the score. Other signals with higher discriminative power will therefore have a higher contribution to the score, rendering the score more discriminative as well.

5.5.2 Performance of the 'weighted correlation similarity' scoring in HeLa datasets

The weighted correlation similarity (WCS) scoring function was constructed from the two weight vectors learned by the training of the neural nets as follows:

WCS(q, c) = CorrelationSimilarity[QueryWeightVector * q, CandidateWeightVector * c]

This function is equivalent to the calculation of the scoring net but runs a lot faster since it avoids the overhead of the neural net framework. The two HeLa datasets were re-searched with the WCS scoring function.

5.5.2.1 Search of all very high confident HeLa peptides with the WCS method

The previous search of all very high confident HeLa peptides was repeated with the WCS scoring function and compared to the CS scoring function and SpectraST. Identification results are summarized in Table 12.

Table 12: Identification results from SpectraST, the CS and the WCS scoring method. 'Correct identifications' were counted as IDs that agree with the 0.001% FDR peptides identified by SequestHT. Number of misses was determined from the output of the CS search engine. The identification rate was calculated with the number query spectra that were included in the library as the 100% reference.

Method	Query dataset	Found in the library	Misses %	Correct IDs	ID rate %
SpectraST	QEx-HeLa (22,352 identified spectra)	20,305	9.2%	19,093	94.0%
CS				19,415	95.6%
WCS				19,918	98.1%
SpectraST	Fus-HeLa (11.452 identified spectra)	10,797	5.7%	10,561	97.8%
CS				10,646	98.6%
WCS				10,671	98.8%

The WCS scoring function led to improved identification rates for both HeLa datasets to a very high level. Notably, the discrepancy of identification rates between the QEx-HeLa and the Fus-HeLa data almost disappeared. High identification rates have been achieved for both datasets (98.1% and 98.8%).

These results included all correctly identified query spectra (= the best-scoring hit was correct) before validation. However, in non-training datasets the identity of spectra is not known beforehand, so the best-scoring hits need to be validated in order to control the number false-positive hits.

5.5.2.2 Decoy spectrum performance with the WCS method

Before target and decoy results are used to validate the hits, it is evaluated whether the decoy spectra generated previously still perform well when the new scoring function is used. The decoy delta scores were re-calculated for both HeLa datasets. Ideally, the decoy spectra would match exactly 50% of the query spectra. Figure 32 and Figure 33 show the distribution of the target-decoy delta scores in the QEx and the Fus HeLa datasets with the original and the weighted scoring function.

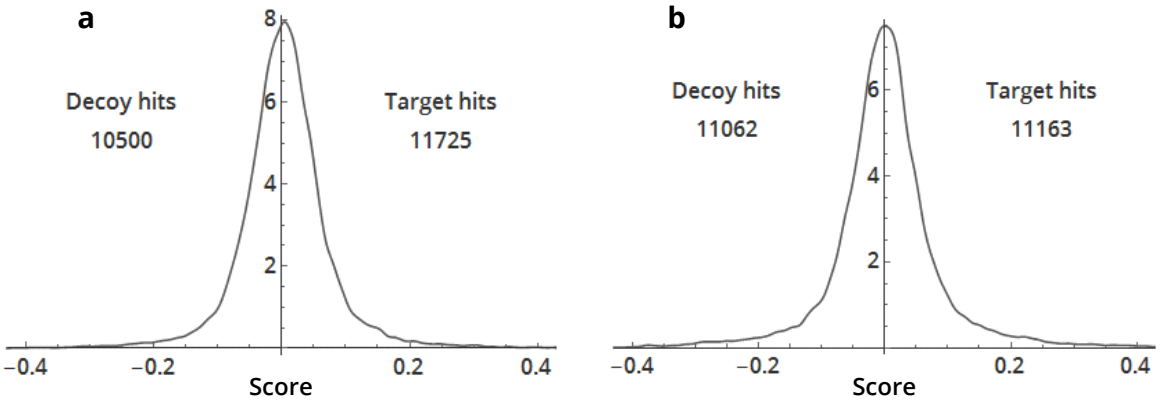

Figure 32: Decoy delta scores for the QEx-HeLa dataset with the original CS (a) and the weighted correlation similarity, WCS, (b) scoring function. The decoy delta scores were calculated for the very high confident query peptides as the best true-negative score minus the best decoy score. CS: Targets = 11,725, Decoys = 10,500 (47.2%). WCS: Targets = 11,163, decoys = 11,062 (49.8%).

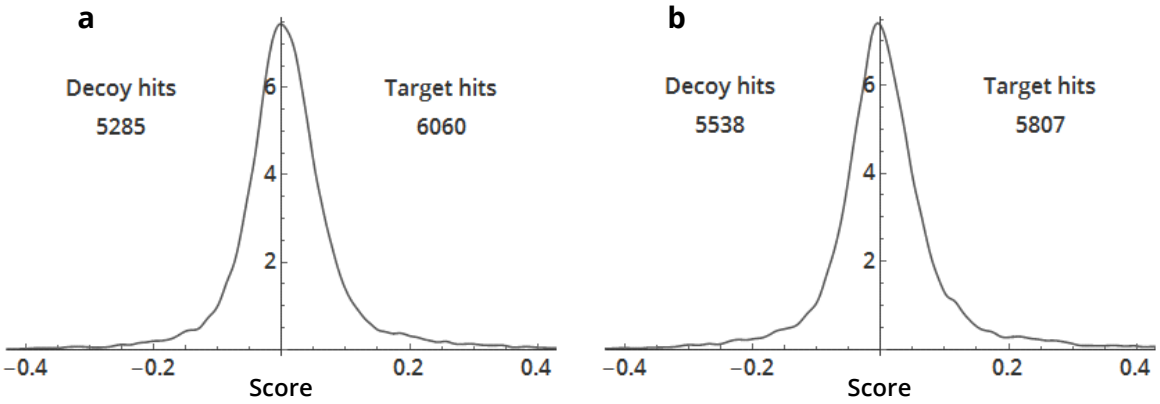

Figure 33: Decoy delta scores for the Fus-HeLa dataset with the original CS (a) and the weighted correlation similarity, WCS, (b) scoring function. CS: Targets = 6,060, decoys = 5,285 (46.6%). WCS: Targets = 5,807, decoys = 5,538 (48.8%).

The decoy matching rate improved from 47.2% with the CS scoring to 49.8% with WCS scoring in the QEx-HeLa dataset, and from 46.6% to 48.8% in the Fus-HeLa dataset.

The decoy match rate of 48.8% for the Fus-HeLa dataset is in line with the best results reported by [48] for their *precursor swap* method and better than the *shuffle and reposition* method by [47].

For the QEx-HeLa dataset, a nearly perfect match rate of 49.8% has been achieved with the *precursor shuffled* decoy spectra and the WCS scoring method. The high match rates and the high symmetry of decoy delta scores suggest that the decoy spectra generated by this method provide a very good estimation of the score distribution of random hits, which leads to a very accurate estimation of the false discovery rate.

5.6 FDR validation of spectral library identifications

The target hits were validated with a global FDR estimation from the decoy hit scores. The delta score (the score difference between the best and the second-best hit) was included as a discriminative parameter to better separate the target from the decoy hits. Figure 34 and Figure 35 show the number of accepted identifications vs. the delta score at 1% FDR for the QEx-HeLa and the Fus-HeLa dataset, respectively.

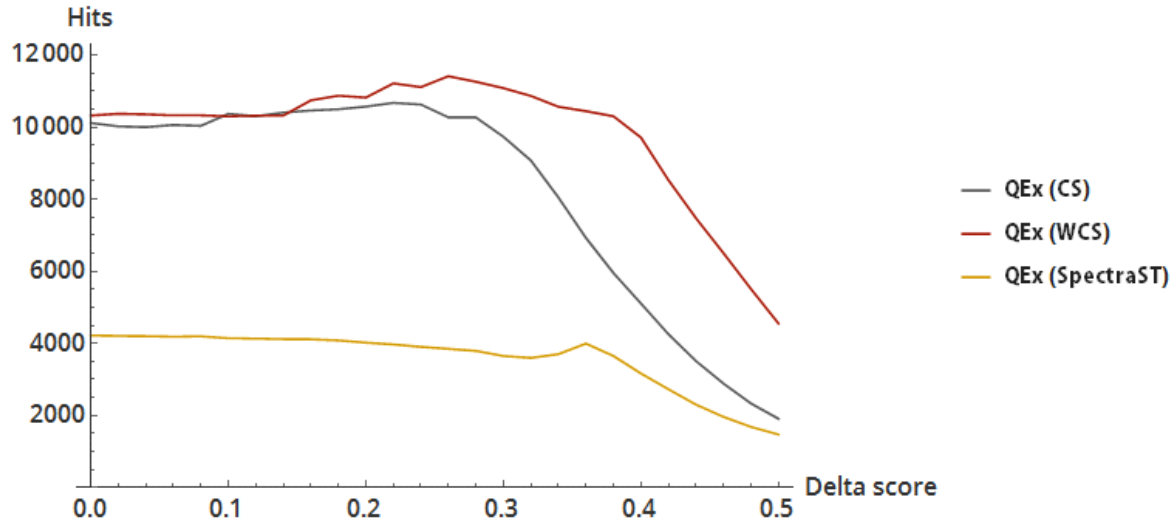

Figure 34: Accepted peptide identifications at 1% global FDR in the QEx-HeLa dataset with the correlation similarity (CS) scoring, the weighted correlation similarity (WCS) scoring, and SpectraST (against the consensus library).

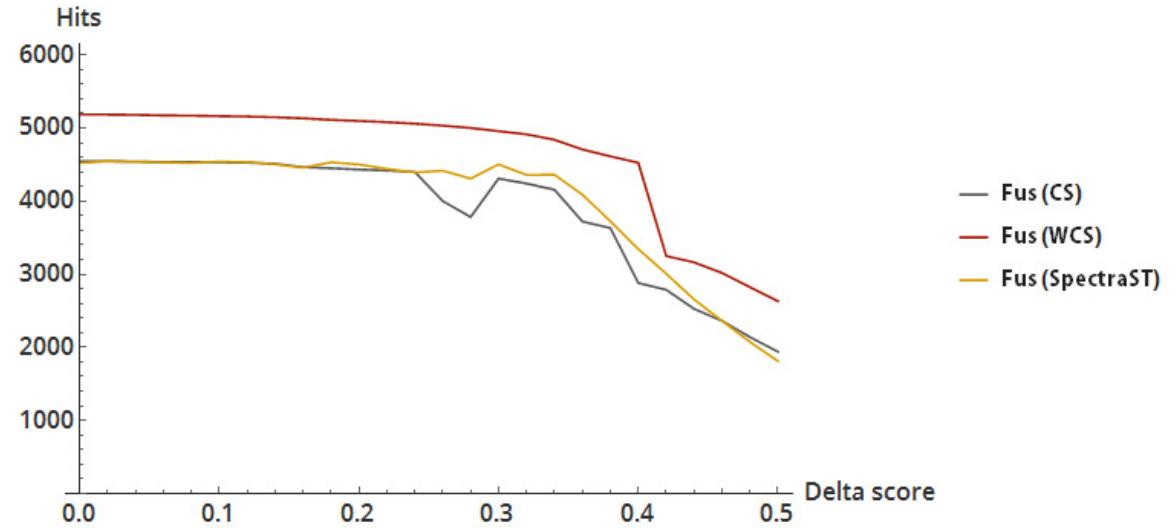

Figure 35: Accepted peptide identifications at 1% global FDR in the Fus-HeLa dataset with the correlation similarity (CS) scoring, the weighted correlation (WCS) similarity scoring, and SpectraST (against the consensus library).

The CorrelationSimilarity method and SpectraST yielded very similar identification numbers at all delta scores for the Fus-HeLa dataset. Significant improvement was achieved by the WCS scoring function. The highest identifications rates were achieved at a delta score threshold of 0.

Effectively, the delta score did not help to improve the separation of the target hits from the decoy hits.

For the QEx-HeLa dataset, a higher delta score threshold improved the number of validated hits for the CS and the WCS function. Detailed identification statistics are summarized in Table 13.

Table 13: Number of peptide identifications at 1% global FDR in both HeLa datasets.

Search engine	Dataset	Spectral library	Library entries	Decoy entries	Validated hits (FDR = 0.01)
SpectraST	QEx-HeLa (50,176 spectra)	PRIDE Cluster (human consensus)	187,709	187,709	4,222
CS		PRIDE Cluster (human)	789,745	788,760	10,673
WCS		PRIDE Cluster (human)	789,745	788,760	11,411
SpectraST	Fus-HeLa (30,722 spectra)	PRIDE Cluster (human consensus)	187,709	187,709	4,546
CS		PRIDE Cluster (human)	789,745	788,760	4,548
WCS		PRIDE Cluster (human)	789,745	788,760	5,186

The WCS function achieved the highest identification rates in both datasets and an improvement of 6.9% and 14.0% over the CS function in the QEx-HeLa and Fus-HeLa dataset, respectively.

The SpectraST target-decoy search could only be performed with a consensus library. Reduction of the PRIDE library to consensus library has shown inferior performance in the previous analysis of the very high confident peptides. The results presented here may therefore improve when the original spectral library is being used, but due to the limitation of SpectraST for decoy library generation, this was not possible.

The overlap of search results between WCS searching and SequestHT is depicted in Figure 36.

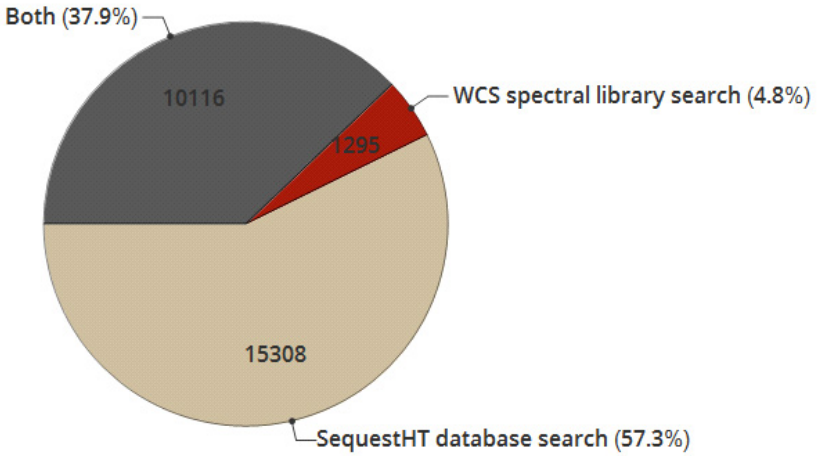

Figure 36: Percentage of peptide hits at 1% FDR identified with SequestHT, the WCS search engine, and both. The 100% mark was set to the union of validated identifications of both search engines.

The SequestHT sequence database search accounted for most of the validated peptide hits among all (57.3 + 37.9% = 95.2%). Many of the peptides identified by the spectral library search engine were also found by SequestHT (37.9% out of 42.7% = 88.7%). However, 4.8% (1,295) of all validated hits were exclusively identified with the WCS spectral search. Table 14 lists modifications among the 1,295 peptides that were only found by the WCS spectral library search.

Table 14: Modifications of peptides identified exclusively by the spectral library search engine at 1% FDR. Since one peptide can contain multiple modifications, the numbers add up to more than 100% resp. 1,295.

Modification	Occurrence in validated peptides	Occurrence (%)
Unmodified	704	54.4%
Modified	591	45.6%
Carbamidomethyl	221	17.1%
Oxidation	179	13.8%
Acetyl	172	13.3%
Pyro-glu	154	11.9%
Phospho	36	2.8%
Deamidation	4	0.3%
Formyl	1	0.1%

Carbamidomethylation and oxidation were used as fixed resp. variable modifications in the sequence database search as well, so these peptides could have been found by SequestHT. There are, however, peptides with other modifications, including acetylation, exchange of glutamine for pyro-glutamate, and phosphorylation, that could not be identified by SequestHT because these modifications were not allowed in the search. Adding them to the list of variable modifications extends the search space significantly and may lead to overall fewer identifications because of more false-positive hits. This observation underlines the strength of spectral library searching, that specific modifications can be detected without explicitly being searched for. Also, inclusion of these modifications does not increase the search space nearly as much as with sequence database searching. Spectral library searching therefore has an advantage in detecting rare modifications, provided that the peptides are represented in the library.

For all methods investigated here, there is a discrepancy between number of peptides that were known to be correctly identified (by comparison to the SequestHT results) and the number of hits which could be validated. Specifically, at least 19,918 peptides were correctly identified by the WCS method (as known from the very high confident peptide IDs from SequestHT), but only 11,411 could be validated after decoy library search and score thresholding to 1% FDR. The overall identification rate at 1% FDR seems low compared to SequestHT (11,411 vs. 25,424) even when accounting for the 2,047 peptides which were not in the spectral library. However, the high agreement between the very high confident peptides from SequestHT and the of identifications this search engine promote the idea that this is not a consequence of lower search engine accuracy.

Instead, it may result from the global FDR validation method used, which would only use the main score and the delta score discriminative parameters to separate all decoy from target hits. The advantage of this naïve validation method is that overfitting is very unlikely to happen, thus the given FDR is likely to be true. Overfitting in this context means that an algorithm learns to distinguish between target and decoy spectra by exploiting specific properties of the decoy spectra that would not naturally occur in random hits. For instance, if the precursor mass deviation had been included as a parameter for validation of target vs. precursor-shuffled decoys hits in this work, a validation method could learn to discriminate targets from decoys to 100% accuracy without ever inspecting a single spectrum or the scores, since the precursor mass shift is exactly the parameter the decoy spectra had been generated by. Likewise, a validation procedure for reversed sequences in sequence database searching would be able to distinguish decoy from target hits by testing whether the C-terminal amino acid is a lysine or an arginine. If not, the sequence was obviously shuffled (semi- or non-tryptic peptides aside) and belong to the group of decoys. Such exploitation of specific artifacts of decoy generation will produce invalid FDR estimations because the decoys no longer resemble the random hits.

Search engines validate their results by taking multiple discriminative parameters into account. For instance, individual (subgroup) FDRs can be calculated for every charge state of the peptides. These strategies should be carefully evaluated, though, in order not to underestimate the FDR and accept more false-positive hits than expected. The authors of ANN-SoLo stated about the ‘subgroup FDR’ approach they used for their open modification search: “Caution has to be

observed, however, because the actual FDR might be underestimated when too small groups are used." [23]. Specifically, for the PRIDE spectral library used in this work, the charge states 2 and 3 dominate in the library, and FDR estimation for the comparably underrepresented charge states 1 und 4-8 may be inaccurate due to small sample sizes.

Percolator, the method used to validate the SequestHT hits, optimizes thresholding by a combination of 37 parameters, including, for example, the main score (XCorr), delta scores, precursor mass and mass deviation, and fraction of matched fragment ions [60]. Optimization of large numbers of features may lead to overfitting and therefore underestimation of the FDR, an issue which has been specifically addressed by the authors.

The conservative global FDR method of hit validation yielded only 44.9% of the identifications compared to SequestHT, along with 4.8% of additional identified peptides, although agreement between the identified spectra of both methods was very high (>98%). A more sophisticated hit validation method may lead to improvement of validated peptide IDs. The decoys generated by this method have shown very good properties, with score distributions and match likelihoods that accurately modeled the random hits. For the results presented here, it can be safely assumed that the FDRs are truly at the 1% mark and not underestimated due to overfitting or inferior decoy spectrum quality.

5.7 Performance considerations

The development of the present spectral library search method involved processing of millions of spectra with various methods and parameters. Time- and memory-efficient code design has been an important part throughout the creation of the code base, which consisted of an estimated 3,000 lines of Mathematica code. Optimization techniques include parallelization, caching of intermediate results, data indexing through the NearestFunction and custom B trees, fast object-like loading and unloading of data. The following section gives details about run-time and memory optimization.

Code design decisions had to be made on the trade-off between run-time and memory consumption frequently, as in when to cache precomputed results or when to generate them on-the-fly. The choice of data structure was equally important, which range from low-level but very efficient lists and tables to more flexible high-level data structures such as Mathematica's Associations and Datasets. The general trend was to use low-level data structures for large lists, e. g. the PRIDE spectral library, and higher-level structures for smaller lists, including the spectrum-spectrum matches produced by searching. The first allowed for fast spectral library access while the latter increased the flexibility when handling SSMs. For example, it allowed for a very easy implementation of additional named scoring schemes that were evaluated by the spectrum scoring function.

Another important point was when to parallelize or when to stay with serial processing, since parallel processing requires additional coding effort and produces significant overhead in Mathematica. Frequent benchmarks were performed to make optimal decisions for each of these questions. Generally, parallelization was universally applied whenever possible, although the speedup was as low as 1.5x for certain tasks even when all 12 CPU cores were used due to the parallelization overhead. Many tasks, however, could be effectively parallelized with speedup of almost 12x, including data import and export, and spectrum recalibration and vectorization.

Memory management needed to be addressed as well. While the loaded library and datasets required 3-4 GB of memory, peak memory consumption was observed for the neural net training, where in-memory storage of the spectral library, both HeLa datasets, the generated training data and the neural net framework consumed a total amount of 25 GB of memory. By storing related data and definitions as DownValues of a singular symbol similar to object-oriented programming, symbols could be saved to disk by DumpSave at any time and re-loaded later in a manner of seconds. This enabled fast loading and unloading of data into/from memory during the design phase of this work.

5.7.1.1 Evaluation times

All operations were performed on a 12-core Intel Xeon E1650 v2 with 40 GB of memory. The spectra were held in memory during all calculations. Most operations could be parallelized to all 12 cores. Table 15 summarizes the execution times for several core functions.

Table 15: Execution times for several core functions of the search engine developed in this work. SpectraST execution times were included for comparison.

Operation	No. of spectra/items	Parallelized	Time
Spectral Library Import	789,745	Yes	274 s
Import of QEx-HeLa MS/MS data	50,176	Yes	175 s
Import of Fus-HeLa MS/MS data	30,722	Yes	305 s
Recalibration and vectorization of the spectral library	789,745	Yes	111 s
Benchmark search with a subsample for evaluation of the scoring function (single-threaded)	2,000 against 789,745	No	88 s
Benchmark search with a subsample (multi-threaded)	2,000 against 789,745	Yes, partly	69 s
Decoy generation	788,630 from 789,745	Yes	28 s
Neural net training	4,000	Yes	1200 s
Target Search QEx-HeLa	50,176 against 789,745	Yes, partly	1210 s
Decoy Search QEx-HeLa	50,176 against 788,630	Yes, partly	1200 s
Target Search Fus-HeLa	30,722 against 789,745	Yes, partly	642 s
Decoy Search Fus-HeLa	30,722 against 788,630	Yes, partly	638 s
Total time for import and searching of a spectral library and an experimental dataset with decoys	**50,176 + 789,745**	**Yes, partly**	**2998 s**
SpectraST library import	789,745	No	1,399 s
SpectraST search of the QEx-HeLa dataset	50.176	No	10,928 s
Total time for import and searching of a spectral library and an experimental dataset with SpectraST	**50,176 + 789,745**	**No**	**14,927 s**

The search engine implemented in this work required 50.0 min for a complete processing of the PRIDE spectral library and the QEx-HeLa dataset. Through parallelization of many tasks, a significant speedup could be realized. The main search was still the most time-consuming step which could only partly be parallelized due to specific limitations in Mathematica.

Import of the Fus-HeLa data took longer than QEx-HeLa, although the number of spectra was smaller (30 k vs. 50 k). However, the number of fragment ions for each spectrum was considerably higher in the Fusion dataset, which required more time to read the data from the text file.

SpectraST library import and the target search alone required 4.1 h in total. The decoy generation and search were not included here because they could not be performed with the complete spectral library. SpectraST did not use parallelization.

6 Outlook

Spectral library searching has been continuously improved in the past decade – a process this work aims to be part of – but is inherently limited by the availability of reference spectra for the experimental targets. In a joint paper by the participants of the 2017 'Dagstuhl Seminar on Computational Proteomics', the authors concluded that "it seems logical to couple spectral library searching with sequence database searching, where the former assigns those peptide ions that have been previously identified, and the latter identifies peptide species that are not in the library merging the results of the two approaches into a single output for the user" [61]. A unified search engine including both reference spectra from spectral libraries and sequence databases may provide proteomics researches with the best of both worlds and achieve higher sensitivity and specificity than either strategy on its own.

The detailed statistic inspection of the PRIDE Cluster spectral library revealed some very interesting insights, including high numbers of replicate spectra for individual peptide species. While replicate spectra were found to be advantageous for peptide identifications in general, high number of replicates from a single species may have negative influences on spectral library search performance. The PRIDE Cluster method could be examined in view of these observations and may have room for improvement to achieve even higher clustering accuracy.

The authors of the 'Dagstuhl Seminar' also stressed the importance of making new computational methods available to end-users by implementing ready-to-use tools and standard file formats [20]. This is fair criticism to the present work as well. This work has been implemented in Mathematica 11.3 as a software prototype. Processing steps have to be entered as function calls, error checking must be done manually, and the code only be executed in the Mathematica environment. Since Mathematica is commercial software, license availability may limit the adoption by other users in addition to technical obstacles. In order to release this method to the community, the code that has been prototyped in Mathematica would need to be re-written in other programming languages, such as R or Python.

The machine learning procedure applied in this method demonstrated how spectrum scoring can be improved by automated learning. However, it only scratches the surface of the capabilities of state-of-the-art neural networks, which solve complex tasks like image or speech recognition with high accuracy. It is conceivable that deep neural networks can be trained to recognize the sequence from a peptide spectrum just from the spectrum itself, with no direct comparison to a reference spectrum, and outperform current methods of spectrum-spectrum matching.

Finally, the idea of "[shifting] our focus from data review to data reuse" [24] can become reality through the increasing use of spectral library searching. By using the resources of large-scale online repositories of proteomics data, analysis of proteomics experiments can become more comprehensive and more accurate, especially when combined with conventional sequence database search into a unified search engine and the use of modern machine learning techniques.

7 Methods

7.1 Mass spectrometry

HeLa lysates (Thermo Fisher Scientific) were acquired with the established standard LC-ESI-MS/MS methods in our laboratory on two nano-liquid chromatography and Orbitrap instruments. Two datasets, 'Q Exactive HeLa' and 'Fusion HeLa', were randomly selected as benchmark data from the set of standards that is run on the instruments on a daily basis. These datasets were not specifically acquired for the development of this method.

7.1.1 Q Exactive HeLa dataset

The *Q Exactive HeLa* ('QEx-HeLa') dataset was acquired on an Orbitrap Q Exactive mass spectrometer (Thermo Scientific). Separation of the peptides was performed on a nanoAcquity nano-LC (Waters) with a C18 trapping column (nanoAcquity UPLC Symmetry C18 trap column, 180 µm × 20 mm, 5 µm, 100 Å), a C18 separation column (nanoAcquity UPLC column, BEH 130 C18, Waters; 75 µm × 250 mm, 1.7 µm, 100 Å), a gradient from 2.0% A to 30% B in 120 min, and a total runtime of 170 min. Mass spectrometric acquisition was done in data-dependent acquisition mode (DDA). MS1 spectra were recorded in a mass range of m/z 375 to 4,000 at a resolution of R = 70,000, AGC target = 3e6. The top 10 signals were selected for fragmentation with a dynamic exclusion of 20 s. HCD collision energy was set to 27, isolation width to m/z 4.0. MS2 spectra were recorded at R = 17,500 from m/z 100 up to the precursor mass, AGC target = 1e5.

7.1.2 Fusion HeLa dataset

The *Fusion HeLa* ('Fus-HeLa') dataset was acquired on an Orbitrap Fusion mass spectrometer (Thermo Scientific). Separation of the peptides was performed on a Dionex UltiMate 3000 nano-LC (Thermo) with a C18 trapping column (Acclaim PepMap µ-precolumn, C18, 300 µm× 5 mm, 5 µm, 100 Å, Thermo Scientific), a C18 separation column (Acclaim PepMap 100, C18, 75 µm × 250 mm, 2 µm, 100 Å, Thermo Scientific), a gradient of 45 min and a total runtime of 70 min.

Mass spectrometric acquisition was done in data-dependent acquisition mode (DDA). MS1 spectra were recorded every 3 s (cycle time) in a mass range of m/z 400 to 1,300 at R = 120,000, AGC target = 2e5. As many signals as possible within the cycle time were selected for fragmentation with a dynamic exclusion of 30 s. HCD collision energy was set to 30, isolation width to m/z 1.6. MS2 spectra were recorded with the ion trap in 'rapid' resolution mode from m/z 120 to 2000, AGC target = 1e4.

7.2 Peptide identification by sequence database search

The two LC-MS datasets were analyzed with Proteome Discoverer 2.0 (Thermo). MS2 spectra were identified with SequestHT against the SwissProt Human database (04/2018) containing 20,260

protein entries. Precursor mass tolerance was set to 0.02 Da, fragment mass tolerance to 0.02 Da (Q Exactive) and 0.6 Da (Fusion). Fully tryptic peptides with a maximum of two missed cleavages and a minimum length of five amino acids were included. Carbamidomethylation of cysteines was defined as a fixed modification, and oxidation of methionine as a dynamic modification, allowing three at most.

The MS2 spectra of both datasets were exported as *mgf* (Mascot generic format) files from Proteome Discoverer 2.0 for subsequent search with spectral libraries. The list of peptide spectrum matches (PSMs) was exported from the SequestHT results as reference identifications.

7.3 Development of a spectral library search engine for the PRIDE Cluster spectral library

All development was done in Wolfram Mathematica 11.3 (Wolfram Research). An estimated total of 3,000 lines of code were written to implement the method, evaluation, calculation of the statistics etc.

7.3.1 Chemical element, amino acid and modifications data

Monoisotopic masses of elements, amino acids and common peptide modifications were imported into Mathematica from different sources. The exact monoisotopic masses of the chemical element H, C, N, O and P were from Wolfram Research's IsotopeData included in Mathematica [62]. Amino acid data was obtained from ExPASy with precision of five decimal places [63]. Chemical modifications of peptides including Carbamidomethyl, Oxidation, Phosphorylation, Phospho, Methyl, Dimethyl, Formyl, Acetyl, Deamidation, Label:13C(6), TMT6plex, iTRAQ4plex, iTRAQ8plex were obtained from unimod.org [64].

7.3.2 Spectrum and library import and data storage

Spectra were imported from *msp* (PRIDE Cluster spectral library) or *mgf* formats (Proteome Discoverer). Both are human-readable text files which can be easily interpreted. All spectra were loaded and kept in memory for all processing steps. Spectrum import was distributed to multiple threads for faster processing by splitting the input file into chunks of spectra of approximately equal length. Data were stored in a table-like predefined structure to allow for fast random access of the entries.

Next, the entire datasets were indexed with Mathematica's NearestFunction, which allows for very fast lookups in constant runtime. Separate indices were built for the precursor m/z ('PEPMASS' property in mgf files, 'Parent' in msp files), the calculated precursor m/z ('Parent' minus 'DeltaMass'), and the identified peptide sequence, if applicable ('sequence' property).

For the HeLa datasets that had been processed by Proteome Discoverer, the SequestHT identification results were imported from the 'TargetPeptideSpectrumMatch.txt' tab-separated table files. Spectrum data and identification results were joined by matching the scan number.

The in-memory structures were dumped to disk using DumpSave to enable subsequent loading of the data in a few seconds, instead of several minutes of re-importing msp/mgf files and re-calculating the indices and other caches.

7.3.3 Basic statistics

After import and preparation of the datasets, basic statistical analyses were performed to ensure complete and error-free import of the data and to learn about the constitution of the data. Histograms were created with Mathematica's Histogram or SmoothHistogram function. The latter effectively plots the probability density function (PDF) of an empirical distribution derived from the data. Tabular statistics were created with the Tally function.

7.3.3.1 Sequence coverage of the PRIDE spectral library

The human SwissProt database was loaded using Mathematica's FASTA Importer. The sequences were split into tryptic peptides, allowing peptide lengths of 5 to 80 amino acids and up to 2 missed cleavages. The intersection of the PRIDE peptide sequences and the digested SwissProt sequences was calculated. Then, the total number of amino acids of all intersecting peptides was divided by the total number of amino acids in the SwissProt database to calculate the sequence coverage.

To extend the sequence matching to non-tryptic peptides, all PRIDE sequences (189,400) were search in the entire human SwissProt database (11.4 million amino acids). To conduct this search, the PRIDE sequences were indexed with a custom-built B tree of depth 5, so that the first five amino acids of a sequence determine its position in the B tree. Then, all protein sequences from the SwissProt database were searched for matches in the B tree (lookup). When a match was found, i. e. when the first 5 characters were the same, all matching peptide sequences were extended and compared to the remaining FASTA protein sequence to check if the PRIDE peptide is fully contained in the FASTA protein.

The B tree index was implemented as a large SparseArray of PackedArrays. The efficient implementation of the sequence index and parallelization of the lookup and string extension enabled searching of all 189,400 unique PRIDE sequences in the entire FASTA database (11.4 million amino acids) in only 2 minutes.

7.3.3.2 Generation of theoretical fragment spectra

Theoretical fragment spectra were generated as series of b and y ions. B ion series were generated from summing up the monoisotopic amino acid masses. The corresponding y ion series was generated by adding the mass of water and subtracting the b ions from the calculated singly-charged peptide mass.

7.3.3.3 Creation of a virtual sum spectrum

The m/z values of fragment ions were binned in steps of 0.05 and summed up to a single 'virtual sum spectrum'. This step was performed for both the theoretical spectra and the real PRIDE spectral library.

7.3.3.4 Peak picking in the virtual sum spectrum

It was apparent that the m/z values of fragment ions were not evenly distributed along the axis but occurred in groups of approx. 1 Th distance. To pick those small clusters specifically, a peak picker was applied using Mathematica's FindPeaks (height threshold of 5, Gaussian blurring of 25 (equals 25 * 0.05 Th = 1.25 Th)). The fractional parts of the m/z value of all signals were plotted in a ListPlot.

7.3.3.5 Recalibration of fragment ion m/z values

The PRIDE library spectra were used to calculate the recalibration function. First, a rough estimation of the slope and offset was applied manually to correct for the 'wrap-around' of the fractional parts at 0.5. Then, a linear fit was applied to the data using LinearModelFit. Separate fits were created for the subsets of unmodified and TMT-modified peptides.

The linear function derived from the fit of all spectra was turned into a CompiledFunction and used as the m/z recalibration function throughout this work.

7.3.3.6 Fractional parts of fragment ions

The fractional parts of the fragment ions (theoretical or experimental) was calculated as the distance from the nearest whole number, ranging from -0.5 to 0.5. All fractional parts were quantified by creating a HistogramDistribution from the observed data at a resolution of 0.01 Th. The probability density function (PDF) of the HistogramDistribution was plotted to visualize the distribution of fractional parts of the m/z values. The percentage of fractional parts in a given interval [a; b] was calculated with the cumulative density function (CDF).

7.3.4 Vectorization of fragment spectra

Vectorization transforms the list of (m/z, intensity) pairs into a list of intensities with predefined m/z spacing (bins). The common m/z vector was set to range from 100 Th to 2000 Th with a bin size of 1 Th. When multiple ions fall into the same bin, only the highest intensity is taken.

The total length of m/z vectors is 1901. The PRIDE library spectra had an average of 50 ions per spectrum, so most values in the vectors are going to be zero. The vectors were therefore stored as SparseArrays instead of conventional PackedArrays. Both are built-in Mathematica structures. SparseArrays are a very efficient way to handle matrices with many zeros ('sparse matrices') and stored the data in 3-4x less memory than conventional arrays.

7.3.5 Dynamic spectrum processing: The filtering pipeline

The present work involves processing of spectral data with a variety of different methods and parameters. During development and optimization of this method, the various steps of data processing have to be adjusted frequently and efficiently. A filtering pipeline has been implemented to facilitate this process. A filter is a function that may apply any operation to a spectrum object as long as it returns another valid spectrum object. Filters can be given at the time of executing any function using the spectral library or the query datasets and can be

combined in any order. Every access to spectral data, including searching, visualizing and exporting, can be passed dynamically through the filters.

Various core functions have been implemented with the filtering scheme, including recalibration and vectorization, intensity transformations (for subsequent spectrum matching and scoring), and decoy spectrum generation. Most filters are small methods with only a few lines of code, like the IntensityTransformationFilters. Others, such as the AddCalibratedVectorIonListFilter, calls multiple subroutines to calculate its result. A list of filters is given in Table 16.

Table 16: Overview of core functions implemented through filters.

Vectorization filters	
AddVectorIonListFilter	Adds a vectorized version of the ion list (m/z-intensity pairs) to the spectrum record.
AddCalibratedVectorIonListFilter	Same, but recalibrates all m/z values before vectorization
Fragment ion intensity transformation filters	
IntensityTransformationFilters["Sqrt"]	takes the square root of all intensities
IntensityTransformationFilters["Rank"]	Rank-transforms the intensities
IntensityTransformationFilters["LogMedian Normalize"]	Log10-transforms intensities and normalizes them to the median
IntensityTransformationFilters["Top150"]	Filters the top 150 ions, replacing all other ions with zeros
IntensityTransformationFilters["Top150 Subtract"]	Same as above, but also subtracts the 151th intensity from the top 150 ion intensities
IntensityTransformationFilters["Constant-1"]	Sets all intensities > 0 to 1 (unitizes the data)
Decoy spectrum generation filters	
DecoyFilter["IntensityShuffle"]	Shuffles the intensities of a fraction of x of the fragment ions. m/z values are left unchanged. (default x = 1.0)
DecoyFilter["MZRandomize"]	Repositions a fraction of x of the fragment ions along the m/z axis. (default x = 0.6)
Generic filters	
GenericFilter[“Confidence”, qvalue]	Filters spectrum identification by SequestHT confidence (Percolator q-value). Removes ID information for spectra with q-value worse than ‘qvalue’.

Because filters operate on each spectrum individually, the *precursor shuffle* decoy method was not implemented as a filter since it requires precursor information other library entries.

For the PRIDE spectral library, the result of AddCalibratedVectorIonListFilter – the recalibrated, vectorized fragment spectra – was stored in a cache to avoid recalculation for every library query. For the HeLa query spectra, the vectorized fragment spectra were generated on-the-fly by the filter.

The generic confidence filter was used to filter the very high confident peptides from the SequestHT identifications (FDR 0.001).

7.3.6 Decoy spectrum generation

Three methods of decoy spectrum generation were implemented, including a) shuffling the intensities of a certain percentage of fragment signals (IntensityShuffle), b) randomizing the m/z values of a certain percentage of fragment masses (MZRandomize), and c) shuffling the precursor masses around.

7.3.6.1 Intensity shuffle

The intensity shuffle method replaces an adjustable fraction of intensities with a random intensity from the same spectrum. m/z values remain unchanged.

7.3.6.2 m/z randomize

The m/z randomize method assigns a random m/z to an adjustable fraction of signals.

7.3.6.3 Precursor shuffle

The *precursor shuffle* method replaces the precursor m/z of a fragment signal with a randomly selected precursor m/z within a given range. The minimum distance and the maximum distance from the original precursor can be adjusted. The two distances were adjusted so that the number of precursors shuffled to a replicate spectrum of itself was low, and at the same time the shuffled precursor m/z was as close as possible to the original m/z. From all precursors of the same charge state that were present within the specified m/z range, one was randomly selected and assigned as the new 'decoy precursor mass'. Precursor masses that shuffled to a replicate of itself were removed.

The precursor shuffling function was implemented to operate on a sorted list of precursor masses with linear run-time and parallelization support. Masses were sorted before shuffling, if necessary, and the original order was restored afterwards. A NearestFunction ('DataShuffledMassLookup') was generated from the shuffled recalibrated precursors and used as the precursor mass lookup function for decoy spectrum search.

7.3.7 Decoy spectrum evaluation

The quality of decoy spectra was evaluated by searching the very high confident peptides from a dataset against the target and the decoy library. Since the positive target hits are very likely to be true (hence very high confident peptides), the negative hits are assumed to be all wrong. They serve as a 'ground truth' for the random hits which the decoys should model as accurately as possible. The 'target-decoy delta score' is calculated by subtracting the score of the best negative

hit from the score of the best decoy hit to compare the decoy scores with the negative (random) hit scores.

7.3.8 False discovery rate estimation and hit validation

False discovery rate (FDR) was estimated as described previously and regularly performed in identification through both database and spectral library searching of LC-MS/MS datasets [43]. The distribution of decoy scores was used as the null distribution. The FDR was controlled to the desired level by calculating the percentage of accepted decoy hits among all hits and adjusting the score threshold accordingly ('global FDR' estimation).

7.3.9 SpectraST identification

SpectraST version 5.0 (TPP v5.1.0 Syzygy, Build 201711031215-7670 (Windows_NT-x86_64)) was used for spectral library searching [30]. The PRIDE spectral library (in *msp* format) was imported to SpectraST (*splib* format). Two SpectraST libraries were created, a library containing all spectra from PRIDE, and a consensus library, where duplicates, i. e. spectra which represent the same sequence-charge-modification peptide, were removed by merging them into consensus spectra. Decoy spectra were generated from the consensus library with the *shuffle-and-reposition* method into a separate library.

The HeLa datasets were exported from Mathematica to *msp* files (MSPExporter) to have the spectra annotated with their SequestHT identification result, and then used in SpectraST.

SpectraST parameters were set to the default, except the following: precursorMzTolerance was set to 0.02, and peakBinningFractionToNeighbor was set to either 0.5 or 0.0.

7.4 Benchmarking and method optimization

Optimizations were performed with a randomly sampled subset of 2,000 spectra which had been identified with very high confidence (FDR 0.1%) by SequestHT for efficient computation. They very high confident peptides served as a 'ground truth' this method was optimized towards. All optimizations were conducted with the filtering pipeline by including or excluding the corresponding filters or adjusting their parameters.

The number of true positive matches and the delta score were used as measures of method performance. The delta score was calculated as the score of the best hit minus the score of the next best hit which represents a different peptide.

7.4.1 Scoring schemes

The cosine similarity and the correlation similarity were used for the calculation of the score of a spectrum-spectrum match.

The cosine similarity is the dot product of two vectors divided by their norms. It was calculated as:

$$\cos(u, v) = u \cdot v / (\text{Norm}[u] * \text{Norm}[v])$$

The correlation similarity is the dot product of two vectors divided by their norms. It was calculated as:

$$cs(u, v) = (u - Mean[u]) \,.\, (v - Mean[v]) \,/\, (Norm[u - Mean[u]] * Norm[v - Mean[v]])$$

7.5 Machine Learning for advanced method optimization

7.5.1 Training and validation data

4,000 random spectra (out of 22,352 identified peptides with very high confidence (FDR=0.001)) were selected to generate training data for the neural net training. For each of the 4,000 spectra, candidates were selected within a m/z tolerance of 0.02 Th, as before. Fragment signals were rank-transformed and reduced to the top-150 signals. All positive-matching query-candidate pairs were included in the training data. The number of negative-matching pairs was limited to 10x the number of positives and picked randomly. Otherwise, the negative-matching pairs would have greatly outnumbered the positive-matching pairs and the neural net would bias towards the negatives during the learning process. A total of 225,151 query-candidate pairs (30,423 positive pairs, 194,728 negative pairs) were used as training data.

Another set of 2,000 different spectra was prepared as a validation dataset and processed in the same manner as the training dataset. The validation dataset was built with a negative-to-positive-ratio of 2:1 and contained 43,441 query-candidate pairs (15,599 positive pairs, 27,842 negative pairs).

7.5.2 Learning of a weighted scoring function

7.5.2.1 Neural net construction

The aim of neural net training was the implementation of a modified version of the correlation similarity scoring function where every m/z value in the query and the candidate spectrum is multiplied with an adjustable weight. The two weight vectors are independent. The neural net should adjust the weight vectors by finding values that would minimize the similarity scoring score for false matches and maximize the score for true matches.

First, a scoring neural net was set-up to replace the correlation similarity scoring function. The neural net takes two intensity vectors as input, the first from the query spectrum, the second from the candidate spectrum. The vectors are fed into two ConstantTimesLayer ('QueryScaling' and 'CandidateScaling'), which multiply the vectors element-wise with the learnable weights. The result is passed to the 'Scoring' subnet, which is a neural net implementation of the correlation similarity function. The output of the neural net is the similarity score for the spectrum pair (Figure 37).

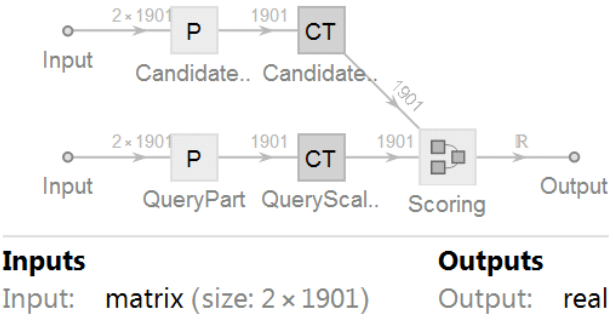

Figure 37: The scoring neural net used for training of the two weight vectors. 'QueryPart' and 'CandidatePart' select the query and candidate spectrum intensity vector. 'QueryScaling' and 'CandidateScaling' are the two layers with learnable weights that multiply the spectrum vectors. 'Scoring' is a sub-net which implements the correlation similarity scoring function.

The scoring sub-net is a neural-net implementation of the correlation similarity (Figure 38).

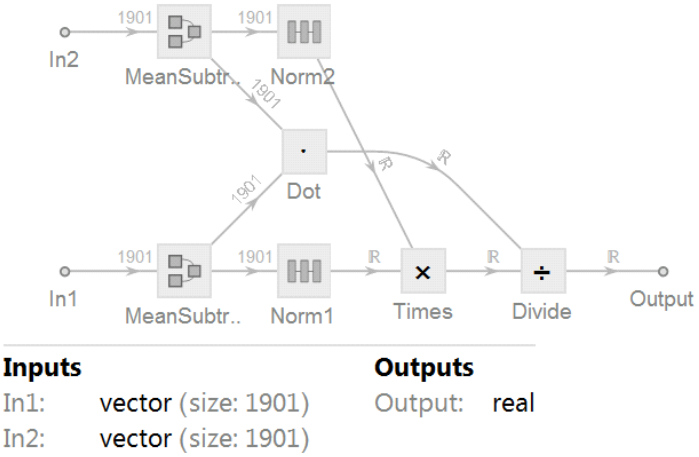

Figure 38: The correlation similarity neural net. The net takes two vectors as inputs (In1 and In2) and calculates the correlation similarity between the two.

7.5.2.2 Neural net training

Training of the neural net was carried out with the help of another neural net which 'wraps around' the scoring the. Implementation of the training net allowed for a presentation of all positive and negative spectra for a single peptide in one run, so that the loss could be calculated per-peptide (Figure 39).

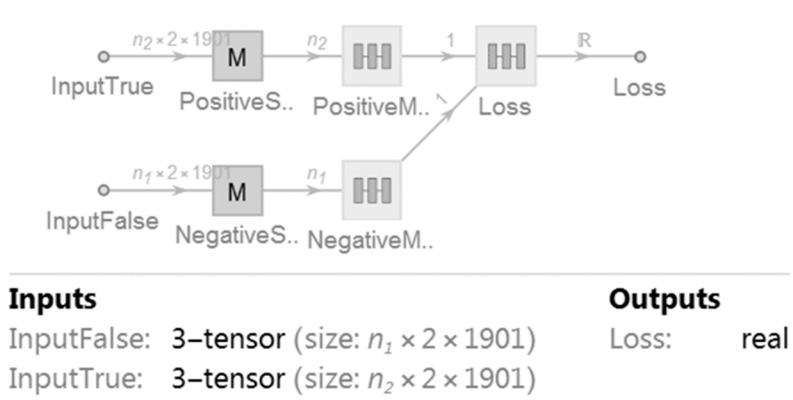

Figure 39: The neural net used for training of the scoring net. The training nets accepts all positive pairs and negative pairs for a query peptide as separate inputs, then applies the scoring net to all positives ('PositiveScoring') and negatives ('NegativeScoring'). 'PositiveScoring' and 'NegativeScoring' are identical copies of the scoring net with shared weights. Next, the loss is calculated as (1 - best positive score) - (best negative score), which is effectively the contrastive loss calculated from the delta score.

Training was performed with the ADAM optimizer for 20 min.

7.5.2.3 Use of the weight vectors learned by the neural net

While the neural net can be used as the scoring function itself, the two weight vectors that were learning during the training can be extracted and applied directly to the spectra inside the correlation similarity scoring function when doing the main search. This avoids the overhead produced by the neural net architecture and achieves higher performance when calculating the scores.

The weight vectors learned by the neural net were extracted ('QueryWeightVector', 'CandidateWeightVector') and a weighted scoring method was implemented as

WCS(q, c) = CorrelationSimilarity[QueryWeightVector * q, CandidateWeightVector * c].

8 References

1. Aebersold, R. and M. Mann, *Mass spectrometry-based proteomics.* Nature, 2003. **422**(6928): p. 198-207.
2. Kelleher, N.L., et al., *Top Down versus Bottom Up Protein Characterization by Tandem High-Resolution Mass Spectrometry.* Journal of the American Chemical Society, 1999. **121**(4): p. 806-812.
3. Toby, T.K., L. Fornelli, and N.L. Kelleher, *Progress in Top-Down Proteomics and the Analysis of Proteoforms.* Annual review of analytical chemistry (Palo Alto, Calif.), 2016. **9**(1): p. 499-519.
4. Aebersold, R. and M. Mann, *Mass-spectrometric exploration of proteome structure and function.* Nature, 2016. **537**: p. 347.
5. Hebert, A.S., et al., *The one hour yeast proteome.* Mol Cell Proteomics, 2014. **13**(1): p. 339-47.
6. Wiśniewski, J.R., A. Zougman, and M. Mann, *Combination of FASP and StageTip-Based Fractionation Allows In-Depth Analysis of the Hippocampal Membrane Proteome.* Journal of Proteome Research, 2009. **8**(12): p. 5674-5678.
7. Bensaddek, D., A. Nicolas, and A.I. Lamond, *Evaluating the use of HILIC in large-scale, multi dimensional proteomics: Horses for courses?* International journal of mass spectrometry, 2015. **391**: p. 105-114.
8. Aebersold, R. and D.R. Goodlett, *Mass spectrometry in proteomics.* Chem Rev, 2001. **101**(2): p. 269-95.
9. Fenn, J.B., et al., *Electrospray ionization for mass spectrometry of large biomolecules.* Science, 1989. **246**(4926): p. 64-71.
10. Kebarle, P. and U.H. Verkerk, *Electrospray: from ions in solution to ions in the gas phase, what we know now.* Mass Spectrom Rev, 2009. **28**(6): p. 898-917.
11. Kingdon, K.H., *A Method for the Neutralization of Electron Space Charge by Positive Ionization at Very Low Gas Pressures.* Physical Review, 1923. **21**(4): p. 408-418.
12. Perry, R.H., R.G. Cooks, and R.J. Noll, *Orbitrap mass spectrometry: instrumentation, ion motion and applications.* Mass Spectrom Rev, 2008. **27**(6): p. 661-99.
13. Makarov, A., *Electrostatic Axially Harmonic Orbital Trapping: A High-Performance Technique of Mass Analysis.* Analytical Chemistry, 2000. **72**(6): p. 1156-1162.
14. Zubarev, R.A. and A. Makarov, *Orbitrap mass spectrometry.* Analytical chemistry, 2013. **85**(11): p. 5288-96.
15. Makarov, A., E. Denisov, and O. Lange, *Performance Evaluation of a High-field Orbitrap Mass Analyzer.* Journal of the American Society for Mass Spectrometry, 2009. **20**(8): p. 1391-1396.
16. Nikolaev, E.N., et al., *Initial Experimental Characterization of a New Ultra-High Resolution FTICR Cell with Dynamic Harmonization.* Journal of The American Society for Mass Spectrometry, 2011. **22**(7): p. 1125-1133.
17. Olsen, J.V., et al., *Higher-energy C-trap dissociation for peptide modification analysis.* Nat Methods, 2007. **4**(9): p. 709-12.
18. Tyanova, S., T. Temu, and J. Cox, *The MaxQuant computational platform for mass spectrometry-based shotgun proteomics.* Nature Protocols, 2016. **11**: p. 2301.
19. Tran, N.H., et al., *Complete De Novo Assembly of Monoclonal Antibody Sequences.* Scientific Reports, 2016. **6**: p. 31730.
20. Griss, J., *Spectral library searching in proteomics.* Proteomics, 2016. **16**(5): p. 729-40.
21. UniProt.org. https://www.uniprot.org/ (accessed Jan 2019)
22. PRIDE Archive - proteomics data repository. https://www.ebi.ac.uk/pride/archive/ (accessed Feb 2019)
23. Bittremieux, W., et al., *Fast Open Modification Spectral Library Searching through Approximate Nearest Neighbor Indexing.* J Proteome Res, 2018. **17**(10): p. 3463-3474.
24. Griss, J., et al., *Identifying novel biomarkers through data mining-a realistic scenario?* Proteomics Clin Appl, 2015. **9**(3-4): p. 437-43.

25. Wilhelm, M., et al., *Mass-spectrometry-based draft of the human proteome.* Nature, 2014. **509**(7502): p. 582-7.
26. Griss, J., et al., *PRIDE Cluster: building a consensus of proteomics data.* Nat Methods, 2013. **10**(2): p. 95-6.
27. Wang, M., et al., *Assembling the Community-Scale Discoverable Human Proteome.* Cell Syst, 2018. **7**(4): p. 412-421 e5.
28. The ProteomeXchange Consortium. http://www.proteomexchange.org (accessed 2019)
29. Griss, J., et al., *Recognizing millions of consistently unidentified spectra across hundreds of shotgun proteomics datasets.* Nature Methods, 2016.
30. Lam, H., et al., *Development and validation of a spectral library searching method for peptide identification from MS/MS.* Proteomics, 2007. **7**(5): p. 655-67.
31. Craig, R., et al., *Using annotated peptide mass spectrum libraries for protein identification.* J Proteome Res, 2006. **5**(8): p. 1843-9.
32. Frewen, B.E., et al., *Analysis of peptide MS/MS spectra from large-scale proteomics experiments using spectrum libraries.* Anal Chem, 2006. **78**(16): p. 5678-84.
33. NIST Libraries of Peptide Tandem Mass Spectra. http://peptide.nist.gov (accessed Feb 2019)
34. MassIVE-KB Peptide Spectral Libraries. http://massive.ucsd.edu/ProteoSAFe/static/massive-kb-libraries.jsp (accessed Feb 2019)
35. Dasari, S., et al., *Pepitome: evaluating improved spectral library search for identification complementarity and quality assessment.* J Proteome Res, 2012. **11**(3): p. 1686-95.
36. Michalski, A., et al., *Ultra high resolution linear ion trap Orbitrap mass spectrometer (Orbitrap Elite) facilitates top down LC MS/MS and versatile peptide fragmentation modes.* Molecular & cellular proteomics : MCP, 2012. **11**(3): p. O111 013698.
37. Kendrick, E., *A Mass Scale Based on CH2 = 14.0000 for High Resolution Mass Spectrometry of Organic Compounds.* Analytical Chemistry, 1963. **35**(13): p. 2146-2154.
38. Sleno, L., *The use of mass defect in modern mass spectrometry.* J Mass Spectrom, 2012. **47**(2): p. 226-36.
39. Shao, W., K. Zhu, and H. Lam, *Refining similarity scoring to enable decoy-free validation in spectral library searching.* Proteomics, 2013. **13**(22): p. 3273-83.
40. Liu, J., et al., *Methods for peptide identification by spectral comparison.* Proteome Sci, 2007. **5**: p. 3.
41. SpectraST. http://tools.proteomecenter.org/wiki/index.php?title=Software:SpectraST (accessed Feb 2019)
42. Eng, J.K., A.L. McCormack, and J.R. Yates, *An approach to correlate tandem mass spectral data of peptides with amino acid sequences in a protein database.* J Am Soc Mass Spectrom, 1994. **5**(11): p. 976-89.
43. Kall, L., et al., *Assigning significance to peptides identified by tandem mass spectrometry using decoy databases.* J Proteome Res, 2008. **7**(1): p. 29-34.
44. Keller, A., et al., *Empirical statistical model to estimate the accuracy of peptide identifications made by MS/MS and database search.* Anal Chem, 2002. **74**(20): p. 5383-92.
45. Deutsch, E.W., et al., *Human Proteome Project Mass Spectrometry Data Interpretation Guidelines 2.1.* Journal of proteome research, 2016. **15**(11): p. 3961-3970.
46. Moore, R.E., M.K. Young, and T.D. Lee, *Qscore: an algorithm for evaluating SEQUEST database search results.* J Am Soc Mass Spectrom, 2002. **13**(4): p. 378-86.
47. Lam, H., E.W. Deutsch, and R. Aebersold, *Artificial decoy spectral libraries for false discovery rate estimation in spectral library searching in proteomics.* J Proteome Res, 2010. **9**(1): p. 605-10.
48. Cheng, C.Y., et al., *Spectrum-based method to generate good decoy libraries for spectral library searching in peptide identifications.* J Proteome Res, 2013. **12**(5): p. 2305-10.
49. Ahrne, E., et al., *An improved method for the construction of decoy peptide MS/MS spectra suitable for the accurate estimation of false discovery rates.* Proteomics, 2011. **11**(20): p. 4085-95.
50. Jordan, M.I. and T.M. Mitchell, *Machine learning: Trends, perspectives, and prospects.* Science, 2015. **349**(6245): p. 255-60.

51. McCulloch, W.S. and W. Pitts, *A logical calculus of the ideas immanent in nervous activity. 1943.* Bull Math Biol, 1990. **52**(1-2): p. 99-115; discussion 73-97.
52. Rosenblatt, F., *The Perceptron - a Probabilistic Model for Information-Storage and Organization in the Brain.* Psychological Review, 1958. **65**(6): p. 386-408.
53. Glorot, X., A. Bordes, and Y. Bengio, *Deep Sparse Rectifier Neural Networks*, in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, G. Geoffrey, D. David, and D. Miroslav, Editors. 2011, PMLR: Proceedings of Machine Learning Research. p. 315--323.
54. Srivastava, N., et al., *Dropout: a simple way to prevent neural networks from overfitting.* Journal of Machine Learning Research, 2014. **15**: p. 1929-1958.
55. LeCun, Y., Y. Bengio, and G. Hinton, *Deep learning.* Nature, 2015. **521**(7553): p. 436-44.
56. Iizuka, S., E. Simo-Serra, and H. Ishikawa, *Let there be color!* ACM Transactions on Graphics, 2016. **35**(4): p. 1-11.
57. Silver, D., et al., *Mastering the game of Go with deep neural networks and tree search.* Nature, 2016. **529**(7587): p. 484-9.
58. Hsieh, E.J., et al., *Comparison of database search strategies for high precursor mass accuracy MS/MS data.* J Proteome Res, 2010. **9**(2): p. 1138-43.
59. Lam, H., et al., *Building consensus spectral libraries for peptide identification in proteomics.* Nature Methods, 2008. **5**: p. 873.
60. Spivak, M., et al., *Improvements to the percolator algorithm for Peptide identification from shotgun proteomics data sets.* Journal of proteome research, 2009. **8**(7): p. 3737-3745.
61. Deutsch, E.W., et al., *Expanding the use of spectral libraries in proteomics.* J Proteome Res, 2018.
62. Wolfram Research's IsotopeData. https://reference.wolfram.com/language/note/IsotopeDataSourceInformation.html (accessed 2018)
63. ExPASy Amino Acid Masses. http://education.expasy.org/student_projects/isotopident/htdocs/aa-list.html (accessed 2018)
64. Unimod.org. http://www.unimod.org (accessed 2018)

9 Appendix

List of hazardous chemicals

No hazardous chemicals were used in this work.

10 Eidesstattliche Versicherung

Hiermit versichere ich an Eides statt, die vorliegende Dissertation selbst verfasst und keine anderen als die angegebenen Hilfsmittel benutzt zu haben. Die eingereichte schriftliche Fassung entspricht der auf dem elektronischen Speichermedium. Ich versichere, dass diese Dissertation nicht in einem früheren Promotionsverfahren eingereicht wurde.

_____

Hamburg, den 27.02.2019