Improving Peptide Identification in Proteomics Data Analysis through Repeat-Preserving Decoy and Decoy-Free Retraining

by

Johra Muhammad Moosa

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science

Waterloo, Ontario, Canada, 2023

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner: Kaizhong Zhang
Professor, Dept. of Computer Science,
The University of Western Ontario,
London, Ontario, Canada N6A 5B7

Supervisor(s): Bin Ma
Professor, Dept. of Computer Science, University of Waterloo

Internal Member: Lila Kari
Professor, Dept. of Computer Science, University of Waterloo

Other Member(s): Yang Lu
Assistant Professor, Dept. of Computer Science,
University of Waterloo

Internal-External Member: Mu Zhu
Professor, Dept. of Statistics and Actuarial Science,
University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

This dissertation includes first-authored peer-reviewed materials. Johra Muhammad Moosa was the sole author for Chapters 1, 2, 4, 6 and 7 which were written under the supervision of Prof. Bin Ma. Chapter 3 and 5 are based on co-authored papers. The details of the contributions are as follows:

Research presented in Chapter 3: is published in Johra M Moosa, Shenheng Guan, Michael F Moran and Bin Ma. Repeat-preserving decoy database for false discovery rate estimation in peptide identification. *Journal of proteome research*, 19(3): 1029-1036, 2020. https://doi.org/10.1021/acs.jproteome.9b00555.

Authors' Contribution: Johra Muhammad Moosa is the first author and contributed to the generation of ideas, implementation of algorithms, and testing under the supervision of Prof. Bin Ma. Shenheng Guan, Michael F. Moran, and Bin Ma were co-authors and contributed to the study design and resources. Johra Muhammad Moosa wrote the draft manuscripts, to which all co-authors contributed intellectual input.

Research presented in Chapter 5: is published in Johra M Moosa and Bin Ma. Improving peptide identification rate by machine learning with next-ranked peptide spectrum matches. In *18th Conference on Computational Intelligence Methods for Bioinformatics & Biostatistics, CIBB 2023, Padova, Italy*, 2023.

Authors' Contribution: Johra Muhammad Moosa is the first author and contributed to generation of ideas, implementation of algorithms, analysis, and drafting of the manuscript under the supervision of Prof. Bin Ma. Bin Ma is the co-author of the publication and contributed to the study design, resources, and manuscript finalization.

Abstract

Accurately identifying peptides in proteomics is central to understanding the complexities of biological systems. Despite the advancements in proteomics data analysis, challenges related to False Discovery Rate (FDR) estimation and peptide identification persist. This thesis offers two novel contributions that address these pressing issues.

The first part of the thesis focuses on a critical issue plaguing traditional target-decoy approaches—the inability to preserve repeated peptide structures in decoy databases. Addressing this, we introduce a novel algorithm for decoy database generation that utilizes the de Bruijn graph model. This innovative method effectively conserves the structural repeats found in target protein databases, thereby significantly enhancing the precision of FDR estimations. Comparative evaluations reveal that our de Bruijn graph-based model excels in FDR accuracy and increases the rate of peptide identifications, outperforming existing algorithms.

The second part introduces a machine learning-based retraining strategy for refining Peptide-Spectrum Matches (PSMs). Unlike traditional methods that draw from target and decoy databases for positive and negative training examples, our research presents a novel strategy for calculating *next-best* PSMs. Specifically, our approach employs the *best* and the *next-best* peptides from the same spectrum as the respective positive and negative examples for training. We introduce a tailored solution involving a split database search to address the critical requirement for a sufficient quantity of *next-best* PSMs to estimate the accurate separation between true and false distribution. This innovative decoy-free training paradigm yields notable improvements in peptide identification rates while preserving the integrity of FDR estimations. The effectiveness of this approach has been corroborated through empirical testing, including integration with well-known algorithms like Mokapot and the application of various machine-learning algorithms such as logistic regression, XGBoost, and neural networks.

The thesis also explores the broader implications and possible extensions of the proposed decoy-free re-training method to complement these core contributions. It speculates how the concept of *next-best* PSMs could be adapted for other proteomics applications like FDR estimation on spectral library search. This line of inquiry opens new avenues for future research.

In summary, the research encapsulated in this thesis advances the field of bottom-up proteomics by offering solutions for more accurate FDR estimation and enhanced peptide identification. As such, it serves as a foundational framework for future research and presents immediate applications for more reliable and robust proteomics data analysis.

Acknowledgements

I stand at the culmination of an arduous journey, one filled with challenges and triumphs, and I am profoundly grateful for the support and love that has carried me through. This thesis would not have been possible without the incredible people who have touched my life.

First and foremost, I want to express my deepest gratitude to my husband. In the darkest hours, when I felt I could no longer go on, you stood by me, offering strength when I had none left. Your unwavering support sustained me through this journey, and I know it will continue to do so forever. I am forever grateful.

To my cherished family—my parents and my amazing brothers—your unwavering encouragement and belief in me have served as unbreakable pillars of strength. Your love and compassion during the countless hours I spent buried in research are immeasurable. Endless prayers, unending support, and heartfelt well wishes from my parents propelled me forward and ensured that I never felt alone on this journey. My brothers, your unconditional advice was a guiding light during my most perplexing and worst moments. I am confident your counsel will continue to brighten my journey in the years that lie ahead.

To my incredible friends, Fatema Tuz Zohora and all others, your presence and strong friendship have been a source of joy and motivation. Only my friends, you understand my true self, and together we have shared countless laughs and quirky adventures that have brightened this journey. Thank you for seeing and appreciating the real me.

I extend my heartfelt thanks to my lab mates, whose camaraderie enriched both my research and my life. Your support, from shaping my presentations to introducing me to new experiences like scooter riding, bowling, pool, and golf, has been invaluable.

Finally, but most significantly, I extend my boundless gratitude to my supervisor, Bin Ma. Throughout the challenges and obstacles I faced, your constant encouragement and open-hearted guidance illuminated the path through academia's intricate labyrinth. You transformed what seemed impossible into a reality. Your thought-provoking questions and mentorship not only shaped my research, but also ignited and steered my creativity and problem-solving abilities, for which I am deeply indebted.

As I reflect on this journey, I take pride in persevering even when the road was hard and the path uncertain. With the conclusion of this voyage, I commit to further nurturing my sense of pride and happiness. This thesis stands as a testament to my resilience, determination, and the remarkable people who supported me, including myself. My deepest gratitude to all.

Thank you from the bottom of my heart.

Dedication

Dedicated to my angels, not forgotten, in my heart they'll forever stay,
Amidst whispers deep within, ethereal companions find their way,
A fleeting rainbow, untimely flown, and souls too swiftly called away,
Their absence casts a weight profound, through boundless realms they sway.
Close to my heart, their memory clings, eternal lament, a price to pay,
Their essence weaves through my very soul, in this academic journey's bay,
Enigmatic symbols softly etched, their presence in every word, they say.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Proteomics, as a scientific discipline, constitutes a comprehensive exploration of proteins, the fundamental structural elements of all living organisms. To unravel the intricate workings of the proteins and rectify potential functional anomalies, the determination of their precise sequence is imperative. This parallels our need to discern individual letters within a sentence to grasp its meaning. Yet, proteins, given their substantial size and intricacy, elude full-scale decoding of the sequence with current technological constraints. In response to this challenge, scientists employ a strategy similar to that of disassembling a complex puzzle; they fragment proteins into smaller units called peptides. Due to their reduced size, these fragmented peptides become amenable to sequencing. In the realm of bottom-up proteomics, these identified peptides serve as foundational elements from which the structure and composition of proteins are subsequently inferred and computationally derived. This meticulous cataloging and quantification of proteins within a given biological sample provides profound insight into various biological processes. Consequently, identifying protein and peptide sequences emerges as an indisputable cornerstone in proteomics research.

In recent years, high-throughput tandem mass spectrometry (MS/MS) has become a pivotal element in the field of proteomics. The rapid advancement in commercial mass spectrometry technology has led to a substantial amount of high-quality proteomic data at our disposal for analysis. In a bottom-up proteomics study based on mass spectrometry, a proteolytic enzyme first digests the constituent proteins into short peptide fragments. The resulting peptide fragments are then subjected to mass spectrometry analysis, which produces corresponding spectra. Consequently, these spectra are annotated with the amino acids they represent, allowing us to deduce the sequence of the peptides. This process of identifying peptide sequences serves as a foundational step in studying the biological functions encoded within proteins.

Accurately assigning spectra to their originating peptides is critical in ensuring precise protein identification. When a protein sequence database is available, database searching is the most widely recognized method for peptide identification. To identify the peptides, we search experimentally observed MS/MS spectra against a sequence database to find the best matching database peptide. The observed spectra are then evaluated against the theoretical spectra of candidate peptides, resulting in the determination of peptide spectrum matches (PSMs). In a manner akin to how constituent puzzle pieces seamlessly integrate to unveil a holistic perspective, the identification of the original proteins within the sample is deduced from the identified fragmented peptides. This approach is termed "bottom-up" because it involves reconstructing proteins from individually identified peptide fragments.

Traditionally in bottom-up proteomics, peptide identification involves solving two subproblems: 1) defining a peptide spectrum match (PSM) scoring function and 2) selecting a subset of top-scoring PSMs that are statistically significant [127].

In this chapter, we start by describing the challenges in the field of peptide identification in Section 1.1. Following that, Section 1.2 presents our thesis objectives, which are formulated to address some of these challenges. For those interested in our code and data, Section 1.3 provides relevant links. Lastly, Section 1.4 offers an overview of the organization of the remaining chapters in the thesis.

1.1 Challenges in Peptide Identification

Due to noise and instrumentation errors, some spectra exhibit low quality, making it challenging for the search software to accurately identify the correct peptide. Instead, a high-scoring PSM may be reported due to random matches. Consequently, these low-quality spectra lead to random PSMs, ultimately resulting in false identifications. Thus, the peptide identification search results consist of an aggregation of two trends: true positives and false positives. The PSMs where the spectra matched to their source peptides are correct, whereas those matched with peptides they did not originate from are incorrect. The next step is to separate these trends to distinguish true PSMs from false ones. These incorrect PSMs appear due to poor spectra quality, background noises, or missing peptides in the search database. Nonetheless, we only want to report the highly confident peptides. Therefore, the primary challenge lies in distinguishing between true and random matches to ensure the generation of the most confident PSMs.

Although the ground truth is unknown, controlling the false discovery rate (FDR), the proportion of mismatches among the reported PSMs, warrants our confidence in the output PSMs. The target-decoy search [39] is the widely established strategy to regulate FDR. In this method, a decoy database is incorporated with the original target database to approximate the FDR of the database search results. Nevertheless, existing decoy generation methods often fall short of meeting the criteria of an ideal decoy, presenting a significant challenge in accurately estimating FDR through the target-decoy database search strategy.

1.2 Thesis Objectives

Understanding the functions of peptides and proteins is essential for understanding biological systems and addressing potential malfunctions within them. In this context, our research unfolds in two distinct but interconnected directions.

First, we address the critical aspect of generating decoys that closely approximate the ideal. We have introduced a novel repeat-preserving decoy generation algorithm employing the de Bruijn graph, which addresses the limitations of traditional methods [90]. By ensuring that shared regions in the target database are mirrored in the decoy database, we tackle a key issue: the potential discrepancy between target and decoy databases that can compromise FDR estimation accuracy.

Target-decoy database search has dominated bottom-up proteomics because of its simplicity, yet this widely used method comes with many inherent limitations. It can systematically mislead the scoring functions, leading to underestimated FDRs by increasing target peptide match scores [30]. Some spectra may receive higher scores due to containing more peaks or having precursor masses that result in more candidate peptides [57]. Consequently, target PSMs may be favored in such cases, disrupting the equivalence of random matches in the target and decoy databases. Additionally, generating a perfect decoy database poses a significant challenge.

Therefore, we investigated whether a decoy-free approach could yield results comparable to those of the traditional target-decoy method. Our second objective revolves around harnessing the potential of multiple Peptide Spectrum Matches (PSMs) per spectrum, with a particular emphasis on the lower-ranked PSMs, to enhance the accuracy of identifications. Consequently, within the scope of this research direction, our ultimate objective encompasses two primary aspects: first, to estimate the false discovery rate, and second, to investigate an alternative approach for re-scoring the PSMs by harnessing the additional matches associated with each spectrum.

To achieve the first objective, we seek to establish an alternative method for computing FDR without the reliance on decoys. Instead, we employ additional PSMs for each spectrum to estimate the FDR. This research focuses on estimating the false distribution with the help of the *next-best* peptide for each spectrum, in addition to the *best* peptide, typically reported by search engines as the identified peptide and used for FDR calculation. The novel *next-best* peptide is derived from additional PSMs associated with a given spectrum. We have additionally developed a modified database search method to strengthen the distribution of these *next-best* peptides. Importantly, all the necessary elements for FDR calculation are already present in the search results, adding to the appeal of this approach.

In the second objective, we explore the integration of a decoy-free approach into the post-processing of database search results. Rather than modifying the search steps to remove decoys, we leverage the *next-best* peptides in post-processing. One straightforward strategy for enhancing peptide identification without altering previous steps involves re-scoring and re-ranking the PSMs to provide a more accurate ordering, thus increasing the expected number of correct PSMs. Machine learning post-processing algorithms like PeptideProphet [74, 18] and Percolator [70, 119] can be employed for this purpose. Typically, these algorithms use decoys as negative examples. However, revealing decoy labels before score function learning poses a risk of biased estimates. By doing so, the probabilities of target-false and decoy identifications may no longer remain equal.

In conclusion, our research encompasses two primary objectives: to improve decoy generation by preserving repeats [90] and to leverage additional PSMs for more accurate FDR estimation and improved identification through re-scoring [91]. This research aims to advance the identification of peptides, addressing critical challenges in bottom-up proteomics research.

1.2.1 Ideal Decoy Generation

The target protein sequence database can contain a large number of repeated peptides. Existing decoy generation algorithms do not preserve the structures of these repeats. As a result, the same peptides in the target represent different peptides in the decoys. Previous studies suggest that such discrepancy between the target and decoy databases may lead to an inaccurate estimation of FDR [126]. An ideal decoy method should generate similar numbers of total and unique decoy peptides to the target database.

To accomplish this, we propose a novel method using the de Bruijn graph to generate a randomized decoy database that preserves the repeats [90]. We have mathematically proved that the method preserves the structure of the repeats in the target database to a great extent, regardless of the enzyme digestion specificity (Section 3.3.4). Therefore, it avoids the FDR overestimation problem that exists in the random decoy method. Meanwhile, the generation of the decoy sequences is considerably random, thus avoiding the problems in the sequence reversal method. The method is also straightforward to implement in a computer program. An example implementation of the algorithm in Java can be found at https://github.com/johramoosa/deBruijn.

1.2.2 Improvement of Peptide Identification Rate by Re-scoring the PSMs

The potential for improvement in peptide identification rate by re-scoring PSMs is a critical aspect that justifies further investigation. The conventional approaches for re-scoring, such as PeptideProphet [74, 18], iProphet [113], and Percolator [70, 119], utilize machine learning methods trained on both target and decoy peptides as positive and negative examples, respectively. While these methods have shown efficacy in enhancing the identification rate, they inadvertently expose the target-decoy information to the scoring function, thereby potentially compromising the integrity of the downstream FDR estimation, which also relies on the target and decoy information for estimating the FDR. While two separate decoy databases can be generated for the re-scoring and FDR, respectively, the target database is always shared between the two steps. In addition, because of the inherent homologies between many proteins in the target database, a cross-validation method such as the one used in Mokapot [45] does not avoid the leak of the target information completely.

To address this limitation, we introduce a novel method for re-scoring PSMs without exposing the target-decoy information. Our approach leverages the top-ranked PSMs as positive examples and the next-ranked PSMs as negative examples for each spectrum during machine learning-based retraining. We refer to the top-ranked PSMs as *best* PSMs and the subsequent ranked PSMs as *next-best* PSMs, as elaborated in Sections 5.2.2 and 4.4.2. This strategy evades the need for utilizing decoy databases for identifying negative examples, thus preserving the integrity of FDR estimations.

Critically, our method not only avoids the introduction of bias into the FDR but also enhances the peptide identification rate. The utilization of *best* PSMs and *next-best* PSMs from the same spectrum for training ensures that the newly trained scoring function remains unbiased with respect to target-decoy information. Consequently, our method offers a robust mechanism for improving peptide identification rates without compromising the FDR estimation.

Our empirical validation through integration with established methods like Mokapot and the application of diverse machine learning techniques, ranging from logistic regression to XGBoost and neural network models, affirms the efficacy and robustness of our approach.

While related work such as Nokoi [52] has touched upon similar concepts, it falls short in several key areas. Unlike Nokoi, which does not consider peptide similarity for selecting negative samples and relies on a pre-trained classification model, our method offers the flexibility of training a new scoring function tailored to each dataset. This adaptability ensures that our method remains applicable across diverse datasets, species, and instrumentation setups.

This work aims to underscore the novel aspects and advantages of our re-scoring method, thereby establishing its utility in improving peptide identification rates while maintaining accurate FDR estimations.

1.2.3 Estimation of False Distribution Without The Decoys

Several methods have been proposed in the literature that use decoys differently than the target-decoy database search approach or even without using any decoys. Apart from the inherent challenges of the target-decoy method, we also need to provision for an ideal decoy. Constructing an ideal decoy is highly challenging, including some paradoxical conundrums (Section 4.1.4). The complex nature of generating ideal decoys inspired us to devise a method without the decoys.

Our aim is to construct the false distribution based on the distribution of second-ranked (referred to as "*next-best*") peptides (Section 4.4). Initial analysis of the proposed algorithm yielded promising outcomes. Nonetheless, further refinements are imperative to enhance its efficacy.

1.3 Data and Code Availability

All the datasets utilized in this study are accessible through their respective ProteomeXchange partner repositories [124], which can be found at https://www.ebi.ac.uk/pride/archive/ or https://massive.ucsd.edu. Target databases were obtained from the Uniprot protein database [22], accessible at https://www.uniprot.org/.

We have deposited our in-house Human HeLa mass spectrometry proteomics data to ProteomeXchange via MassIVE (ProteomeXchange project ID: PXD015028). This dataset can also be accessed via FTP at ftp://MSV000084207@massive.ucsd.edu.

In our experiments, we have additionally utilized several publicly available datasets to assess and compare the performance, generalizability, and robustness of our proposed method. The additional data and the database employed in Chapter 3 to validate the algorithm are described below.

- The Yeast dataset was obtained from a Yeast cell lysate, accessible via ProteomeXchange project ID PXD009740 [7].
- The human and yeast proteome sequence databases were procured from UniProt, with proteome IDs UP000005640 and UP000002311, respectively. Additionally, the cRAP (common Repository of Adventitious Proteins) sequence database was sourced from https://www.thegpm.org/crap/, representing a compilation of common protein contaminants frequently encountered in proteomics laboratories.

In Chapters 4, 5, and 6, we employed multiple datasets from various species and cell lines to assess generalizability and guard against overfitting. Throughout our experiments, we utilized four distinct datasets, including the HeLa dataset (ProteomeXchange project ID: PXD015028) discussed in Chapter 3. The three additional datasets are as follows.

- Mouse Muscle Spindle dataset (ProteomeXchange project ID: PXD035552) [10].
- Human Pulmonary Microvascular Endothelial Cell dataset (ProteomeXchange project ID: PXD036260) [78].
- Human HeLa dataset (ProteomeXchange project ID: PXD005280) [9].

In all our experiments, the decoy databases for target-decoy database searches were created using the repeat-preserving decoy algorithm [90] presented in Chapter 3, unless specifically stated otherwise.

Finally, an example implementation of the de Bruijn decoy generation algorithm, as detailed in Chapter 3, written in Java, is available at `https://github.com/johramoosa/deBruijn`.

1.4 Thesis Organization

The remainder of the thesis is structured into several chapters, each dedicated to a specific aspect of our research. Chapter 2, titled “Peptide Identification & Validation”, establishes the fundamental concepts of peptide identification and validation in proteomics research. In Chapter 3, “Repeat-Preserving Decoy Database for False Discovery Rate Estimation in Peptide Identification”, we dive into the development and assessment of a novel decoy generation method. This chapter is derived from the published manuscript [90]. Chapter 4, “FDR Estimation: Training with Next-Best PSMs”, explores false discovery rate estimation techniques using machine learning, with a focus on leveraging the *next-best* peptide spectrum matches and the concept of utilizing multiple PSMs from a single spectrum. Moving on to Chapter 5, “Improving Peptide Identification Rate by Machine Learning with Next-Ranked Peptide Spectrum Matches”, we discuss the application of machine learning to enhance peptide identification rates, emphasizing the use of *next-best* PSMs as negative examples and *best* PSMs as positive examples for training. This chapter is derived from the published manuscript [91]. Chapter 6, “Advanced Machine Learning to Retrain and Re-score the PSMs”, proposes advanced machine learning methods for retraining and re-scoring peptide spectrum matches, utilizing *best* and *next-best* PSMs. Finally, Chapter 7, “Conclusion”, provides a summary of our findings, discusses their implications, and outlines potential avenues for future research. Collectively, these chapters form a comprehensive exploration of our research aimed at advancing the field of proteomics, particularly in the context of peptide identification and validation.

Chapter 2

Peptide Identification & Validation

Scientific inquiry, driven by the quest to understand biological functions and address disruptions within natural systems, hinges on the identification of proteins. These biomolecules, with their multifaceted roles, are central to the intricate workings of life. Peptide sequencing, a fundamental step in protein identification, involves the analysis of peptides derived from protein digestion. Tandem mass spectrometry analysis, followed by a database search, is the predominant technique for peptide sequencing. This method compares experimentally observed spectra with theoretical spectra derived from sequences in a reference protein database. However, this approach produces a mixture of true and false PSMs, which presents the challenge of reporting PSMs with statistical significance. Therefore, control of false discovery rates becomes paramount in the absence of ground truth to ensure confidence in the results.

Peptide and protein identification form the foundational pillars of proteomics research. High-throughput mass spectrometry, a powerful tool, generates sets of MS/MS spectra, facilitating the identification of peptides and, eventually, proteins within a given sample. The primary approach involves searching these spectra against sequence databases.

Yet, the path to precise identification is not without its challenges. The presence of noise and the potential for instrumentation errors during mass spectrometry analysis can lead to the generation of poor-quality spectra. In light of this, the validation of identified peptides is crucial for establishing confidence in reported results. Rigorous and meticulous validation procedures are indispensable prerequisites for deriving meaningful conclusions in proteomics research.

Throughout this chapter, we lay the fundamental groundwork for peptide identification and validation, encompassing peptide spectrum matches (PSMs), Post-translational

Modifications (PTMs), database search engines, and false discovery rate (FDR) estimation. We will then delve into state-of-the-art methodologies that underpin the validation of the identified peptides, facilitating a seamless exploration of our research in the upcoming chapters.

2.1 Peptide Identification in Bottom-up Proteomics

Although mass spectrometers can measure the mass of intact proteins, peptides are sequenced in bottom-up proteomics studies [117]. The fundamental reason is that it is much less complex to determine the smaller peptide sequence from the fragment ion peaks than to determine a larger protein sequence of an intact protein. As a result, proteins are fragmented into smaller peptides using a proteolytic enzyme (typically trypsin) for ease of analysis. Two other frequently employed proteolytic enzymes in proteomics studies are pepsin and chymotrypsin.

Although under-reported, the occurrence of missed cleavages in tryptic peptide bonds is a common phenomenon, as noted in a study by Siepen et al. [115]. The task of protein identification becomes more complicated when enzymes partially break down proteins, leading to peptides that have missed cleavage sites within them. Missed cleavage in proteomics refers to the situation where a protease enzyme, typically employed for protein digestion, fails to cleave a peptide bond within a protein sequence as anticipated. These missed cleavages can lead to the generation of longer-than-expected peptide sequences, subsequently impacting the precision of peptide identification and the characterization of proteins in mass spectrometry-based proteomics investigations.

To identify the proteins, we first need to sequence the fragmented peptides. First, MS/MS spectra containing the peaks list are generated with a mass spectrometer to identify the peptides contained in the sample. When a reference protein sequence database is available, database searching is the most widely used method for peptide identification. Many sequence database search software tools have been developed [99, 85, 29, 49, 42, 34, 75]. A reference peptide sequence database is constructed from reference protein sequences by in-silico digesting them into theoretical peptides following protease specificity rules as the sample preparation. The experimentally observed spectra are matched with the theoretical spectra of peptides in the reference database to identify the peptides. Finally, from these identified peptides, the original proteins present in the sample are deduced [74].

Although advances in tandem mass spectrometry technology have significantly increased the number of acquired spectra, the confidence in the identification of peptides and proteins remains at approximately 60% [34, 35]. In particular, instrumental advances in resolution and detection have outpaced complementary improvements in database search. Consequently, the efficacy of protein identification is heavily dependent on the choice of the peptide identification algorithm.

2.2 Peptide Spectrum Match (PSM)

In bottom-up proteomics, database search engines play a pivotal role by aligning the theoretical spectra of individual peptides with their corresponding experimental spectra, ultimately yielding a list of potential peptides. These identified assignments are formally referred to as peptide spectrum matches (PSMs). Typically, the quality of these PSMs is evaluated using a designated scoring function. This scoring function assigns a numerical value to each peptide-spectrum pairing (P, S), indicating the probability that the fragmentation of a peptide (P) is reflected in the experimental mass spectrum (S) [46, 75]. In an ideal scenario, this scoring function should attribute higher scores to all correct PSMs as opposed to their incorrect counterparts [46], thus allowing accurate distinction between true and false identifications. Traditionally, for each spectrum, the PSM with the highest-scoring peptide is reported for further investigation.

2.3 Post-translational Modification (PTM)

Biochemical modifications of proteins known as post-translational modifications (PTMs) play a crucial role in functional proteomics. PTMs can occur to one or more amino acids on the protein at any stage of the protein's existence as well as during the analysis in the mass spectrometer. Compared to the unmodified primary sequence, it appears either as a mass increase or a mass loss. When searching for peptides in the database, we also need to consider these modified peptides. A variable modification search technique is applied during the database search to find the PTMs. In this method, the particular PTM (e.g., phosphorylation) is allowed to occur on any instances of selected amino acid residues (e.g., threonine, serine, or tyrosine) for all of the theoretical peptides digested in-silico from the entire search database. As a result, it significantly increases the search space. Some PTMs may result in a similar mass as other amino acid residues. For example, the following PTMs and the amino acid residues result in almost identical masses:

1. Deamidation of Asparagine: $N + 0.984$ and D

2. Deamidation of Glutamine: $Q + 0.984$ and E

3. Oxidation of Methionine: $M + 15.995$ and F

In such circumstances, if everything else is similar, m/z (mass-to-charge ratio) will be the same or approximately the same for both candidate peptides. We consider such two peptides interchangeable in our *next-best* calculation (to be discussed in Section 4.4.2 and 5.2.2), i.e., considered the same rank.

2.4 Database Search Engines

Database search engines identify proteins and peptides via mass spectrometry data from primary sequence databases. Bottom-up proteomics entails the enzymatic digestion of proteins prior to their mass spectrometry analysis. The terminology "bottom-up" signifies that the identification of constituent proteins is achieved by reconstructing the proteins from individually identified fragment peptides.

In bottom-up proteomics, a variety of database search engines are available, each with its own unique strengths. In our research, we primarily utilized the search results obtained from MS-GF+ [75], Comet [42], and MS Amanda [34].

2.4.1 MSGFPlus: Database Search Engine

MSGFPlus, also known as MS-GF+, identifies peptides by scoring MS/MS spectra against the in-silico peptides derived from a reference protein sequence database. An MS/MS spectrum-based database search approach generally compares each spectrum against all theoretical peptides. On the other hand, MS-GF+ first computes a suffix array to compare each peptide against all spectra containing the same precursor mass, allowing it to compute rigorous E-values [75]. The program is freely available at http://proteomics.ucsd.edu. MS-GF+ is also incorporated into many other proteomics software or pipelines such as Trans-Proteomics Pipeline [32], SearchGUI [4], Skyline [86], Percolator [119], PeptideShaker [122], etc.

2.4.2 Comet

Comet [42] represents a database search engine dedicated to peptide sequencing through analysis of tandem mass spectrometry data. Originally emerging as an open-source initiative in late 2012, the search engine had its roots in the academic version of the SEQUEST database search tool at the University of Washington [41]. Written in C++, Comet's development extends across both Linux and Windows operating systems. Researchers can readily access Comet by visiting http://comet-ms.sourceforge.net, or alternatively, it is seamlessly integrated into numerous broader software projects.

2.4.3 MS Amanda

MS Amanda represents a specialized peptide identification algorithm meticulously tailored to excel with high-accuracy and high-resolution mass spectrometry data. This software stands out for its exceptional accuracy, as evidenced by the substantial concordance in identified spectra compared to gold-standard algorithms like SEQUEST [43] and Mascot [99]. MS Amanda has also undergone significant enhancements with the introduction of MS Amanda 2.0 [35], notably in terms of its remarkable speed in peptide identification. MS Amanda is available as a freely accessible and stand-alone tool. The software can be downloaded from https://ms.imp.ac.at/?goto=msamanda.

MS Amanda demonstrates remarkable versatility, now extending to enable a secondary search, facilitating the identification of peptides within chimeric tandem mass spectra. Moreover, it possesses several other vital features, including enhancing score readability (where higher scores indicate better matches) and calculating the likelihood of a match occurring coincidentally.

2.5 Validation of Identified Peptides

The database searching method is widely used in proteomics for peptide sequencing. In this method, the experimental spectra are annotated with the best-scoring peptides from a list of peptides generated by digesting the database. Two types of errors can occur in peptide identification, (1) the spectra itself can be an 'unmatchable' [39] spectra, i.e., not resulted from a peptide present in the sample, but occurred because of some random noises, (2) the spectra is generated because of an existing peptide in the sample, but an incorrect peptide is assigned.

Given the presence of noise and potential instrumentation errors, certain spectra may exhibit low quality, leading to the generation of random peptide spectrum matches. Consequently, the PSMs obtained from a database search encompass a blend of true and random peptides. To extract the most reliable PSMs, we need to distinguish between true and false matches. Despite the unknown ground truth, regulating the false discovery rate substantiates our confidence in the resulting PSMs.

2.5.1 False Discovery Rate (FDR) in Proteomics

In proteomics, the false discovery rate (FDR) is an interpretation of the number of random matches present in the output of a database search. In other words, FDR provides the expected proportion of false PSMs. Therefore, controlling FDR allows us to determine our confidence in the search result. In bottom-up proteomics, the target-decoy search strategy is the most adopted method to regulate FDR. This method integrates a decoy database with the original target database to approximate the FDR of the database search results. Following the FDR computation, we can then infer the anticipated distribution of the expected correct PSMs.

2.5.2 Estimation of False Distribution

Estimation of FDR is a crucial aspect of shotgun proteomics. Output PSMs from a database search consist of true and false matches. We need to distinguish between true and random matches up to a remarkable degree to make use of the results confidently. Estimation of the false distribution allows us to decide on a cutoff score, which allows us to only report the peptides above the score threshold with a specific significance. Incorporating a decoy database during the database search is one of the most common methods to estimate the false distribution. In bottom-up proteomics, it is standard to compute FDR with the following equation:

$$FDR = \frac{\#False\ Positve}{\#True\ Positive + \#False\ Positive} \tag{2.1}$$

2.5.3 Target-decoy Database Search [90]

A robust method is essential to evaluate the false discovery rate during identification. The most commonly adopted method for estimating FDR in database search is the target-decoy approach (TDA), as established by Elias et al. and Kislinger et al. [39, 77]. In this approach, MS/MS spectra are matched against a combined sequence database that includes actual target protein sequences and synthetic decoy (incorrect) sequences. The decoy sequence database consists of non-existent biological sequences designed to mimic the typical statistical characteristics of the real target sequences. Ideally, the assumption is that the random peptide spectrum matches will match both the decoy and the target sequences with the same frequency. Therefore, the number of decoy matches becomes an estimation of the number of false matches in the target database. Among the set of reported target matches above a score threshold, the ratio between the number of decoy matches and the number of reported target matches above the same score threshold delivers an estimate of the FDR [90]. Eventually, the FDR provides us with an estimate of the number of validated random targets, also referred to as false positives.

2.5.4 Decoy Generation Methods

Various techniques for decoy preparation have emerged since the inception of the target-decoy validation method [89, 40]. Typically, a decoy sequence database is generated by either reversing or shuffling the sequences of target proteins. During the shuffling process, the enzymatic cleavage sites may be optionally preserved. Alternatively, decoy sequences can be created by complete randomization of each protein sequence. An ensemble of decoy strategies is also feasible. However, the optimal way of decoy construction and utilization still remains an open problem [66].

Several studies have observed that the choice of the decoy generation method may affect the FDR estimation [126, 123, 66]. In addition to the repeat-preserving decoy method discussed in Chapter 3, we explored five additional decoy-generation methods from the existing literature. These methods are detailed below:

Random Shuffling: Random shuffling stands as the most prevalent random decoy generation method, characterized by the random reordering of amino acids in each protein sequence within the database. These decoys do not preserve repeats, resulting in an increase in the number of unique decoy peptides. Consequently, the search space of the decoy is widened compared to that of the target.

Normalized Shuffling: After performing random shuffling, the count of unique peptides in both the target and decoy databases, denoted as N_t and N_d respectively, is recorded. The False Discovery Rate (FDR) is then conventionally calculated (as the ratio between the decoy and the target hits) and further normalized by multiplying it by a factor of $\frac{N_t}{N_d}$. Due to the expanded set of unique decoy peptides, numerous genuine target PSMs may be discarded during the search due to competition from the decoy set. While coefficients may help correct FDR overestimation, the target PSMs lost in this process remain unrecoverable.

Reversal: Reversal represents one of the most commonly employed decoy generation methods. In this approach, each target protein sequence is reversed to produce the decoy. The sequence reversal method preserves target database repeats, but introduces additional issues as decoy sequences lack randomness. This can lead to matching ion series as complements in both target and reversed sequences, and even identical matches between reverse and true peptide sequences.

Shifted Reversal: After the reversal, each amino acid K (lysine) or R (arginine) is swapped with its preceding amino acid in the sequence. This operation usually alters the amino acid composition and total amino acid mass of each peptide, thus addressing the main limitations associated with reversal decoys. In our experiments (see Section 3.3.3), we observed that the shifted reversal method retains noticeable sequence similarities between the target and decoy databases. This similarity can lead to high-scoring matches between decoy peptides and MS/MS spectra, suggesting that significant decoy matches arise systematically rather than randomly.

Trans-Proteomic Pipeline (TPP): TPP's default decoy generation program (decoyFastaGenerator.pl) assumes that enzyme specificity is unknown while generating decoys [32, 113]. Specifically, the decoy generation program uses its built-in default enzyme specificity, which involves digesting after G or F but not before N. This approach retains a substantial portion, though not all, of the repeating peptides resulting from nonspecific enzyme digestion. We also assess the effectiveness of this method in Section 3.3.3.

2.6 Application of Decoy-Free Re-Training Approach

On the one hand, the concept of decoy is incorporated in different fields of proteomics. While there is a well-established utility for decoys in addressing specific challenges, it is equally fascinating to explore the potential benefits these fields may derive from the innovative decoy-free training approach proposed in this thesis.

2.6.1 Improvement of Peptide Identification Rate

In recent research, post-processing algorithms aimed at re-scoring and re-ranking PSMs to enhance sensitivity have gained significant popularity. Our hypothesis centers on the potential utilization of additional PSMs per spectrum to facilitate the learning of a novel scoring function, all without the reliance on target-decoy labeling. Notably, this approach does not eliminate the presence of decoys; rather, it refrains from utilizing any information that distinguishes between target and decoy PSMs.

2.6.2 Decoy-Free Estimation of FDR

Although the performance of our decoy-free training-based FDR estimation is significantly dependent on dataset-specific optimization, this avenue of research remains promising. Our proposed method involves estimating the likelihood of correctness for each spectrum, which in turn enables the estimation of true and false distributions. We leverage this probability-based approach to estimate the FDR, employing a model trained through a decoy-free methodology.

2.6.3 Spectral Library Search

In certain proteomics studies, such as proteogenomics [52], spectral library search, and DIA workflows, the implementation of decoy-based false discovery rate estimation can be complex. While techniques like reverse and random decoy spectra construction exist in the literature [132], accurately determining spectral attributes, such as retention time, remains a challenge [33]. Therefore, the feasibility of employing decoy-free FDR estimation in spectral library search and DIA identification methods arises primarily from the absence of widely accepted and clearly defined decoy methods for FDR computation in these domains.

2.7 Conclusion

In this chapter, we embarked on a concise exploration of the essential components of peptide identification and validation, particularly within the context of bottom-up proteomics. This foundational exploration sets the stage for the fundamental concepts that will guide our research journey in the chapters to come.

We began by introducing the peptide identification process in bottom-up proteomics. Within this framework, we examine crucial elements such as peptide spectrum matches and the significance of post-translational modifications. Additionally, we introduced several notable database search engines, which were integrated into our study.

Recognizing the inherent challenges of noise and potential instrumentation errors in mass spectrometry, we underscored the importance of validating identified peptides. Rigorous and accurate validation procedures are vital to establish the credibility of biological research findings, a prerequisite for drawing meaningful conclusions in the field of proteomics. We navigated through the technique of false discovery rate and FDR estimation, employing target-decoy database search strategies to ensure the reliability of the identifications.

With a solid foundation established, we ventured into the heart of our research journey. In this thesis, we introduced the concept of decoy generation methods, exploring their properties and the compelling need for preserving repeatability in decoys for proteomic analyses. Furthermore, we unveiled the innovative approach of decoy-free re-training, a novel strategy harnessing the potential of multiple PSMs per spectrum. This novel decoy-free retraining method holds promise and has the potential to catalyze advancements across diverse domains within the proteomics field.

Chapter 3

Repeat-Preserving Decoy Database for False Discovery Rate Estimation in Peptide Identification [90]

The sequence database searching method is widely used in proteomics for peptide identification. To control the false discovery rate (FDR) of the search results, the target-decoy method generates and searches a decoy database together with the target database. A known problem is that the target protein sequence database may contain numerous repeated peptides. The structures of these repeats are not preserved by most existing decoy generation algorithms. Previous studies suggest that such discrepancy between the target and decoy databases may lead to an inaccurate FDR estimation. Based on the de Bruijn graph model, we propose a new repeat-preserving algorithm to generate decoy databases. We prove that this algorithm preserves the structures of the repeats in the target database to a great extent. The de Bruijn method has been compared with a few other commonly used methods and demonstrated superior FDR estimation accuracy and improved number of peptide identification.

3.1 Introduction

Identifying protein and peptide sequences is fundamentally important in proteomics research. When a protein sequence database is available, database searching is the most widely used method for peptide identification. In the sequence database searching method, a reference peptide sequence database is constructed from protein sequences by in-silico digesting them into peptides following protease specificity rules. Peptide identification is realized by matching the experimental spectra with the theoretical fragmentation patterns of peptides in the reference database. Many sequence database search software tools have been developed [99, 85, 29, 49, 42].

The database search strategy requires a robust method to assess false discovery rate (FDR) in identification. For FDR estimation, the target-decoy method is the most adopted method. [39, 77] In this strategy, MS/MS spectra are searched against a concatenated sequence database comprising of target protein sequences and artificially generated decoy sequences. The decoy sequence database contains biologically nonexistent sequences that share some statistical attributes with the target sequences. Ideally, random peptide spectrum matches (PSMs) are expected to hit both the decoy and the target sequences with the same probabilistic distributions. Therefore, the number of decoy matches becomes an estimation of the number of false matches in the target database. When the set of target matches above a score threshold are reported, the FDR is estimated by the ratio between the number of decoy matches and the number of reported target matches above the same score threshold.

A number of decoy generation methods have been proposed since introduction of this validation method. [89, 40] A decoy sequence database is normally constructed by reversing or shuffling the target protein sequences. During the shuffling, the enzymatic cleavage sites may be optionally preserved. Decoy sequences can also be produced by complete randomization of each protein sequence. Combination of multiple decoy strategies is also possible. However, the optimal way of decoy generation and utilization still remains an open problem. [66] Several studies observed that the choice of the decoy generation method may affect the FDR estimation. [126, 123, 66]

The characteristics of an ideal decoy sequence database are compiled by Elias and Gygi. [40] While creating a decoy, preservation of the target amino acid residue composition is a desired feature. [39] Decoy database should also statistically mimic the target database, except for the sequence itself [132]. Therefore, the principle is to find decoys with similar peptide mass [126], peptide length, and protein length [40] distributions. The shared peptides among different proteins in the target database should be preserved in the decoy database. [66] Otherwise, multiple occurrences of the same peptide in the target database may produce different sequences in the decoy database. Consequently, the search space in the decoy database will become larger than that of the target database. This causes the random PSMs to appear more frequently in the decoy database, and leads to FDR overestimation.

Sequence reversal and random sequences (including sequence shuffling) are two popular methods for the generation of decoy databases. One advantage of the sequence reversal method is that the repeats in the target database is preserved. However, a major problem of the sequence reversal method is that the decoy sequences are not random. Taking into account of the neutral loss ions, internal fragment ions, and possible post-translational modifications, it becomes very easy for the search algorithm to match an ion series as its complementary one in the target and reversed sequences, respectively. It is also possible for a peptide sequence in a reverse database to match identically against a true peptide sequence (i.e., palindromic peptide sequence) in the original target database. [2] Both these increase the false hit rate in the decoy database and cause FDR overestimation. The data in this paper also suggest even the approximate but not identical matches can cause FDR estimation errors. It was showed that FDR estimated by sequence reversal was inaccurate. [44] A comparison of three different decoy database construction strategies, random sequences, sequences from unrelated species, and sequence reversal also showed that the reversal method had the worst performance. [123] To reduce the problem caused by the reversal decoy method, the MaxQuant software proposed the following modification. [27] After sequence reversal, all amino acids K and R are swapped with their previous amino acid in the sequence. This operation usually alters the amino acid composition and the total amino acid mass of each peptide, and therefore reduces the aforementioned adversarial effects. Throughout this paper, we refer to this decoy method by *shifted reversal*. As to be shown in this paper, shifted reversal still does not completely remove the adversarial effects of the reversal method.

The random sequence method is also problematic because it does not preserve the repeats. The repeated occurrences of the same peptide in the target database may cause the appearances of different decoy sequences in database. [40, 126] An earlier study showed that the decoy database may contain as many as double number of unique peptides than that of the target database. [39, 126, 38] As a result, this method overestimates the FDR. [126]

Two methods were proposed in Wang et al. [126] to minimize the FDR estimation error caused by the different numbers of unique peptides in target and decoy. The first method is to normalize the FDR by the ratio between the unique and total peptides. In particular, when the random sequences are produced by shuffling the amino acids within each target sequence, we refer to this normalization method as *normalized shuffling*. Although the normalization can correct the FDR overestimation, we demonstrate in this paper that it causes the reduction of number of peptides identified by database searching. The second method generates decoys after removing the redundant tryptic peptides from the target sequence database. This only works if the enzyme digestion rule is known and strict. In addition, the target and decoy databases become statistically very different (e.g. their lengths are different) after the redundancy removal and the effect of this discrepancy is unknown.

An alternative to Wang's proposal is to memorize the corresponding decoy sequence for each peptide during the decoy generation, and reuse the same decoy sequence when the peptide appears again later in the database. This should both solve the problem caused by repeating peptides and avoid the changes to the protein sequence length. In fact, the method has been successfully used in the Trans-Proteomic Pipeline (TPP) [32, 113]. However, in all of these three approaches reviewed above, the programs require a strict enzyme digestion rule to compute and memorize the unique peptides in memory. Since missed and nonspecific cleavages are usual in proteomics sample preparation, it would be ideal if a method does not rely on the enzyme digestion specificity and completeness. This is the primary goal of this paper. When the digestion rule is unknown or nonspecific, TPP uses a default digestion rule (cut after G or F but not before N). This should preserve many but not all of the repeating peptides obtained with a nonspecific enzyme digestion. The effectiveness of this method is also studied in this paper.

In this paper, we propose a novel method, called de Bruijn decoy, to generate a randomized decoy database. The method is based on the de Bruijn graph model. De Bruijn graph [31] is a graph that can be used to efficiently capture the repeated substrings in a sequence. Previously, de Bruijn graph has been used in genome assembly [131, 20, 81, 14, 3, 21, 65], protein sequence assembly [55, 54], and in compression of protein sequence databases. [36]

The new method is proved mathematically to preserve the structure of the repeats in the target database to a great extent, regardless of the enzyme digestion specificity. Therefore, it avoids the FDR overestimation problem in the random sequence method. Meanwhile, the generation of the decoy sequences are considerably random, thus avoiding the problems in the sequence reversal method. The method is also remarkably easy to implement in a computer program. The method is compared with five other commonly used decoy methods, random shuffling, normalized shuffling, reversal, shifted reversal, and TPP. Our data demonstrated excellent FDR estimation accuracy, and an improved number of peptide identifications by using the new de Bruijn decoy method.

3.2 Methods

3.2.1 de Bruijn Decoy

A k-mer is a continuous stretch of k letters. For a protein sequence database and a given positive integer k, the corresponding de Bruijn graph is constructed as follows. Each protein sequence is attached with a special leading sequence consisting of k dash symbols (The leading sequence is added for the convenience of our algorithm, but is not in the standard de Bruijn graph construction). Then, each distinct k-mer in the sequences contributes a vertex. Each distinct $(k + 1)$-mer $a_1a_2 \cdots a_{k+1}$ contributes a directed edge connecting the two vertices corresponding to the two k-mers, $a_1a_2 \cdots a_k$ and $a_2a_3 \cdots a_{k+1}$. The edge is labeled with a_{k+1}. Figure 3.1 shows an example.

Each target sequence corresponds to a directed path in the de Bruijn graph. To generate the decoy database, the edge labels are replaced by random amino acids. For each target sequence, the algorithm follows the corresponding path in the graph, and concatenates the new edge labels together to produce a decoy sequence. Therefore, if a sequence is repeated multiple times in the target database (such as the sequence "REPEAT" in Figure 3.1(a)), their corresponding paths in the de Bruijn graph will overlap and produce multiple decoy sequences that share a repeat that is only slightly shorter than the original repeat (such as the sequence "YLPQ" in Figure 3.1(d)). The parameter k controls the degree of randomness and the length of the preserved repeats. The precise behaviour on this repeat-preserving property and the effect of parameter k will be discussed in the discussion section.

The random labeling of edges can be a simple shuffling of all edge labels, or is controlled as follows to ensure that the resulting amino acid frequencies are approximately the same in the target and decoy databases, respectively. For each amino acid a, let $n(a)$ be the number of a's occurrences in the target database. Let N be the total number of amino acids in the target database. The algorithm relabels the edges one by one. For each edge e, let $k(e)$ be the number of paths using e. The algorithm chooses a random amino acid a as the edge's new label following the probability distribution $p(x) = n(x)/N$ for each amino acid x; then it reduces $n(a)$ by $k(e)$.

The above method does not need to know the digestion enzyme. However, if the digestion enzyme is known and it is desired to preserve the digestion sites, only very minor modification to the procedure is needed. For example, if trypsin is used and the amino acids K and R need to be preserved, the algorithm will only randomly replace edge labels that are not equal to K and R, while keeping K and R unchanged.

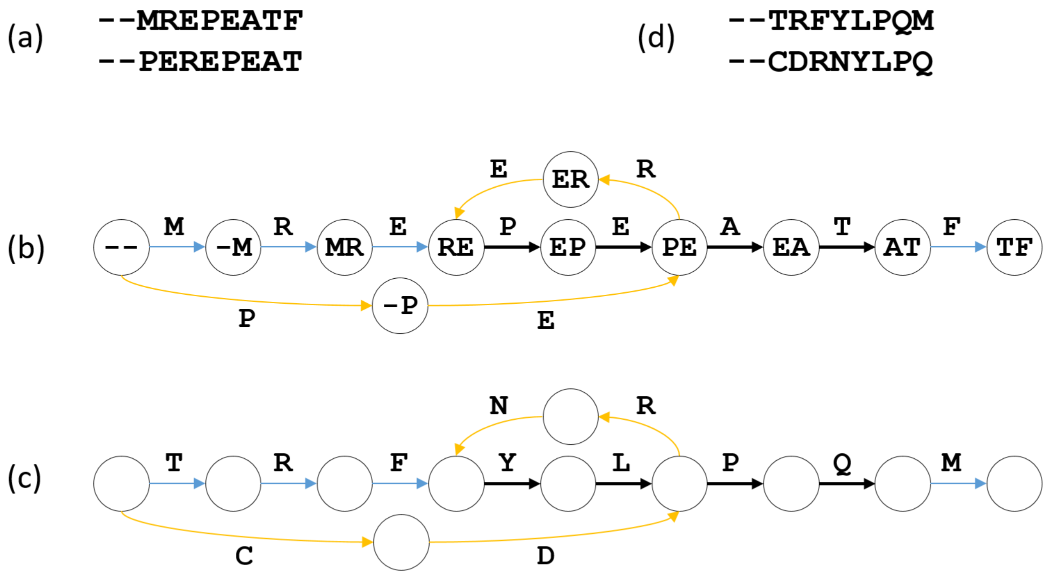


Figure 3.1: An illustration of the de Bruijn graph method. (a) An example target protein database containing two sequences. (b) The corresponding de Bruijn graph with $k = 2$. Each target sequence corresponds to a path in the graph. The edges from the first sequence, the second sequence, and shared by both sequences are in blue, orange, and black, respectively. (c) The edge labels are randomly replaced with other amino acids. (d) The decoy protein sequences are obtained by following the paths of the two target proteins in the re-labeled graph.

To implement this method in a computer program, a hash map data structure is used to keep all the $(k+1)$-mers in the database sequences. Then, each $(k+1)$-mer is mapped to a random amino acid. Lastly, for each target sequence, each of its $(k+1)$-mers is sequentially used to query the hash map. The obtained amino acids of such queries are concatenated together to produce a decoy sequence.

3.2.2 Other Decoy Methods

Five additional decoy generation methods existing in the literature were compared with de Bruijn decoy method. They are outlined in the following:

1. *random shuffling*. The amino acids in each protein sequence in the database are randomly shuffled.

2. *normalized shuffling.* After random shuffling, the number of unique peptides in the target and decoy databases are counted and recorded by N_t and N_d, respectively. After the FDR is calculated in the usual way (the ratio between decoy and target hits), it is further normalized by multiplying a factor $\frac{N_t}{N_d}$.

3. *reversal.* Each target protein sequence is reversed to produce the decoy.

4. *shifted reversal.* After reversal, each amino acid K or R is swapped with its previous amino acid in the sequence.

5. *TPP.* TPP's default decoy generation program (decoyFastaGenerator.pl) pretends that the enzyme specificity is unknown. More specifically, the decoy generation program uses its built-in default enzyme specificity - digesting after G or F but not before N.

3.2.3 MS Data and Sequence Database Search

Two LC-MS/MS datasets were used in the experiments. The first dataset was from a HeLa lysate sample, acquired on an Orbitrap QEHF instrument with 2-hour LC gradient and a top-30 DDA method. Identifications from three technical replicates were used together for analysis. The raw files are converted to mzML files using MSConvert. [15] The mass spectrometry proteomics data have been deposited to ProteomeXchange via MassIVE (ID: PXD015028). Link to the dataset is ftp://MSV000084207@massive.ucsd.edu.

The second dataset was from a Yeast cell lysate (downloaded from ProetomeXchange project ID PXD009740. [7]

The human and yeast proteome sequence databases were downloaded from UniProt with proteome ids UP000005640 and UP000002311, respectively. The cRAP (common Repository of Adventitious Proteins) sequence database, which represents a list of common protein contaminants found in proteomics labs, was downloaded from https://www.thegpm.org/crap/.

The MS-GF+ search engine [75] was used to perform database searches. The following search parameters were used: Precursor mass tolerance was 10 ppm. Precursor isotope errors of -1, 0, 1, 2 were allowed. Both termini of the peptides were required to be tryptic sites unless stated otherwise. Maximum allowed missed cleavage was set to 5. Constant modification of carbamidomethylation on cysteine (C) and variable modifications of methionine oxidation, pyro-glu formation from peptide n-terminal glutamine, deamidation of asparagine and glutamine, initial protein methionine loss with or without N-terminal acetylation were employed.

3.3 Results

3.3.1 Numbers of Unique Peptides in Decoy

An ideal decoy method should generate similar numbers of total and unique decoy peptides as there are in the target database. Table 3.1 summarizes the numbers of the total and unique peptides in different decoy databases derived from the same target database. The human proteome database was used as the target database, which contains 71,785 protein sequences. Each protein in the target and decoy databases is in-silico digested to count the number of peptides. We have ignored the peptide sequences that contain 'X' in the calculation. The in-silico digest conditions were (1) both termini of the peptides are tryptic sites, (2) the mass range of peptides is between 350 and 4000 Da, and (3) the maximal number of allowed missed cleavages is 5.

#Peptides **(thousands)**	Target	Random Shuffling	Normalized Shuffling	Reversal	Shifted Reversal	TPP	de Bruijn
Total	9109	8944	8944	9165	9106	9025	9155
Unique	4276	8239	8239	4309	4334	5843	4306

Table 3.1: Number of peptides in a target and the decoy databases generated with different methods: random shuffling, normalized shuffling, reversal, shifted reversal, TPP, and de Bruijn. Normalized shuffling and random shuffling have the same numbers because they use the same decoy sequences. Target database contains many repeats. The number of unique peptides in the de Bruijn decoy is similar to that of the number of unique peptides target database.

Table 3.1 shows the number of peptides and unique peptides in the target and the decoy databases generated by different methods. It clearly shows that the target database contain many repeats. In fact, about half of the total peptides in target database are due to repeats. This causes the decoy database generated by the random shuffling and normalized shuffling to contain more unique peptides than the target. This problem is avoided or reduced by the other four methods, which utilize certain rules to generate the decoy sequences. The same phenomenon is observed when semi-tryptic rule is used for digestion, where only one of the two termini is required to satisfy the trypsin digestion rule. The semi-tryptic results are provided in Table 3.2. The in-silico digest conditions for these results were the same, except both termini of the peptides are semi-tryptic sites.

We note that the difference between the numbers of target and reversal decoy peptides in Table 3.1 is not a counting error. It is only because a non-tryptic digestion site between

#Peptides (thousands)	Target	Random Shuffling	Normalized Shuffling	Reversal	Shifted Reversal	TPP	de Bruijn
Total	145,522	149,431	149,431	147,226	146,595	145,320	146,686
Unique	63,426	125,369	125,369	63,695	63,942	85,971	64,195

Table 3.2: The numbers of semi-tryptic peptides in a target and the decoy databases generated with different methods: random shuffling, normalized shuffling, reversal, shifted reversal, TPP , and de Bruijn. Normalized shuffling and random shuffling have the same numbers because they use the same decoy sequences. The number of unique semi-tryptic peptides in the de Bruijn decoy is similar to that of the number of unique peptides target database.

K/R and P may become a digestion site after reversal. It is also noteworthy that in order to test its performance when enzyme specificity is unknown or nonspecific, the TPP decoy method used its default parameter setting without specifying the enzyme.

The de Bruijn decoy also has an additional advantage as it produces fewer peptides shared by the target and decoy databases in comparison to other decoy methods. As to be shown later, identical or similar peptides between target and decoy databases may cause additional challenges in accurate FDR estimation. Table 3.3 summarizes the number of unique peptides shared in target and different decoy databases with different peptide lengths ≥ 5. De Bruijn decoy shares a fewer number of overlapping peptides (unique) with the target database. TPP method produced a significant number of shared long peptides because certain areas in the sequence database are enriched with letter G. For example, the longest peptide “GGFGGGRGRGGGFRGRGRGGGGGGGGGGGGR” has a length of 32, and belongs to the protein “FBRL_HUMAN”. TPP’s default partition rule (after G and F but not before N) is therefore unable to produce enough randomness in these areas.

3.3.2 Statistical Behaviors of de Bruijn Decoy

In addition to the number of unique peptides, more statistical behaviors of the de Bruijn decoy database were compared with the target database. It was found that the length and mass distributions of the unique decoy peptides were nearly identical to the ones of unique target peptides (Figure 3.2 and Figure 3.3).

Another key requirement for the target-decoy method to work is that the false target hits and the decoy hits should have similar score distributions. This requirement is examined in the following experiments.

	Decoy				
Length	Random Shuffling/ Normalized Shuffling	Reversal	Shifted Reversal	TPP	de Bruijn
5	62,321	43,191	43,911	53,711	39,684
6	16,363	11,955	11,640	14,505	8,930
7	1,348	1,704	1,604	1,683	964
8	109	482	385	224	109
9	5	89	107	60	14
10	1	116	101	30	2
11	0	14	20	21	1
12	0	65	69	23	0
13	0	6	8	15	0
14	0	7	1	17	0
15	0	4	5	9	0
16	0	2	0	8	0
17	0	0	0	8	0
18	0	1	4	6	0
19	0	0	0	5	0
20	0	1	0	3	0
>20	0	0	0	35	0

Table 3.3: Numbers of unique tryptic peptides at different lengths shared between the target and each decoy database. Normalized shuffling and random shuffling have the same numbers because they use the same decoy sequences. De Bruijn decoy shares a fewer number of overlapping peptides (unique) with the target database.

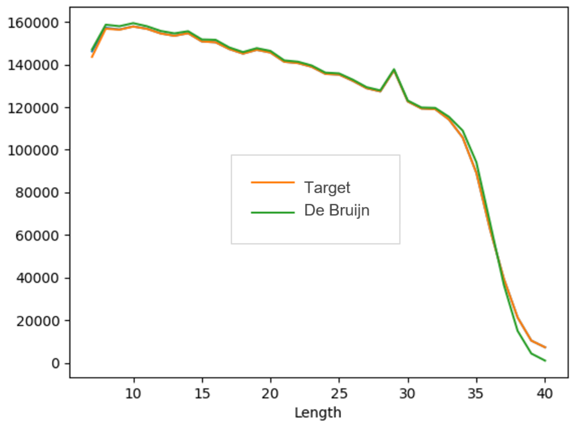


Figure 3.2: Overlap of unique peptide length distributions in de Bruijn decoy and target databases.

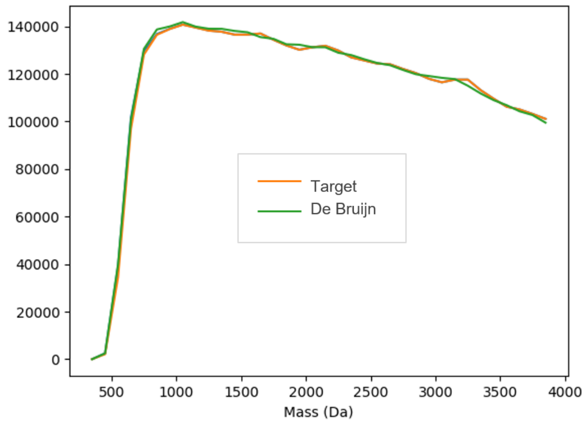


Figure 3.3: Overlap of mass distributions in de Bruijn decoy and target databases.

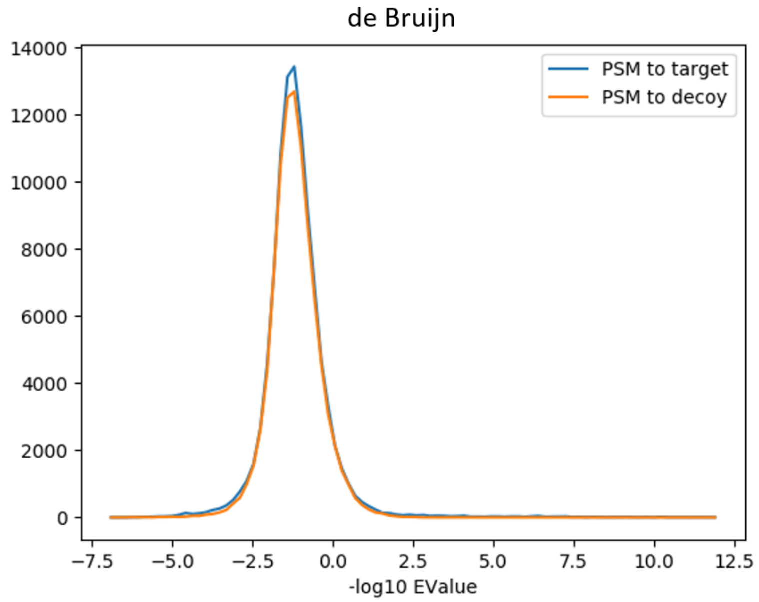


Figure 3.4: E-Value Distributions of PSMs found by matching yeast dataset with human and corresponding de Bruijn decoy databases. The distribution of the target and decoy overlap as the sample and the search database correspond to different species.

In the first experiment, the yeast dataset was searched against the human sequence database plus the corresponding de Bruijn decoy with MS-GF+. The obtained target hits were filtered by removing peptides that appear either in the yeast or the cRAP database. The use of an irrelevant database and the filtration ensure that almost all the target hits are false hits. The score distributions of the remaining PSMs from the human and the decoy databases were plotted in Figure 3.4. As expected, the distributions of the target and decoy hits overlap very well. The minor deviation near the peaks of distributions ($-\log_{10} x \approx -1$) would not have a significant effect on the FDR estimate since the E-Value cutoff threshold is generally much larger.

In the second experiment, the HeLa dataset was searched against the human sequence database plus the de Bruijn decoy. This represents the practical scenario where both the MS data and sequence database are from the same species. Figure 3.5 illustrates the three E-Value distributions: (1) “PSMs to target”, which contains both true matches and false matches from the target database, (2) “PSMs to decoy”, which is considered as random match distribution, and (3) “true matches”, which is obtained by subtracting the

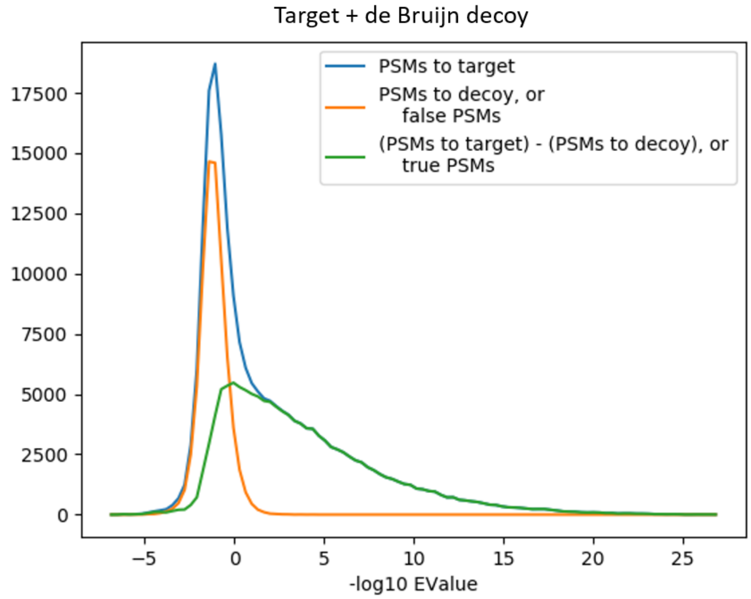


Figure 3.5: Sensible E-Value Distribution estimated with de Bruijn Decoy (HeLA Sample)

distribution of "PSMs to decoy" from "PSMs to target". The figure shows that the de Bruijn decoy can be used to produce a sensible estimation of the false and true target hits.

In contrast to Figure 3.4, there are more PSMs to target than to decoy at the score around 0 in Figure 3.5. This suggests that when the right target database is used, MS-GF+ search engine's results nearby score of 0 (or E-value of 1) still contain many correct PSMs. However, since the search engine did not score it high enough, they would have to be filtered out in order to achieve the desired FDR cutoff.

3.3.3 Database Search Performance Comparison among Different Decoy Methods

In this section, we compare the performance of the database search using different decoy methods for result validation. The HeLa dataset was searched against the human database plus the decoy database generated with each decoy generation method.

Table 3.4 summarizes the numbers of peptide-spectrum matches with different decoy methods at 1% FDR. Notice that when the FDR of the normalized shuffling is calculated,

	Random Shuffling	Normalized Shuffling	Reversal	Shifted Reversal	TPP	de Bruijn
E-value cutoff	0.1122	0.2599	0.14094	0.14646	0.12984	0.2039
Target	94,198	100,255	98,573	98,922	96,847	**101,801**
Decoy	951	(*)1,968	995	999	978	1,028
Expected Correct	93,247	(*) 99,234	97,578	97,923	95,869	**100,773**

Table 3.4: Numbers of PSMs at 1% FDR with different decoy methods. The de Bruijn decoy method outperformed all other decoy methods by achieving the highest number of expected correct PSMs. (*)A normalization factor of 0.519 is multiplied to calculate the FDR for normalized shuffling method.

	Random Shuffling	Normalized Shuffling	Reversal	Shifted Reversal	TPP	de Bruijn
E-value cutoff	0.09319	0.20526	0.1128	0.12013	0.1036	0.16367
Target	96,392	102,323	100,773	101,253	98,833	**104,105**
Decoy	973	(*)2061	1,017	1,022	998	1,051
Expected Correct	95,419	(*)101,280	99,756	100,231	97,835	**103,054**

Table 3.5: Numbers of PSMs at 1% FDR with different decoy methods when semi-tryptic rule is used. (*) A normalization factor of 0.506 is multiplied to calculate the FDR for the normalized shuffling method. The de Bruijn decoy method that we proposed surpasses other decoys in reporting a higher number of expected correct PSMs.

the number of decoy hits was discounted by multiplying a normalization factor of 0.519.

Table 3.4 and Table 3.5 shows that use of de Bruijn decoy leads to an increased number of identified peptides in comparison to other decoy methods. For Table 3.5, the search conditions were same except both the termini were considered semi-tryptic. In addition, de Bruijn achieved the highest number of correct (expected) PSMs for a wide range of FDR thresholds (Table 3.6, Table 3.7, and Table 3.8, (*)A normalization factor of 0.519 is multiplied to calculate the FDR for the normalized shuffling method.).

Table 3.8, 3.7, 3.6 outline the number of PSMs, decoy identifications, target identifications, and expected correct identifications with different decoy methods for five distinct % FDR values ranging from 0.1% to 10%. For all the cases de Bruijn achieved the highest number of correct (expected) PSMs. The results corroborate that de Bruijn behaves consistently across various %FDR levels.

The reasons for the lower database search performances in Table 3.4 for random shuffling

	Random Shuffling	Normalized Shuffling	Reversal	Shifted Reversal	TPP	de Bruijn
Total PSM	287,913	287,913	287,841	287,871	287,875	287,860
Total FP	104,141	10,4141	80,070	79,854	91,301	70,221
% FDR	Expected correct					
0.1	76,564	80,576	79,060	79,279	80,286	83,383
0.5	87,889	93,457	92285	93,116	91,458	95,118
1	93,247	99,234	97578	97,923	95,869	100,773
5	104,217	113,014	111251	111,354	107,446	115,388
10	106,977	120,575	117897	117,672	111,607	124,472

Table 3.6: Numbers of expected correct at five different % FDR thresholds with different decoy methods. The de Bruijn decoy method we proposed surpasses other decoys in reporting higher number of expected correct PSMs across different FDR thresholds.

	Random Shuffling	Normalized Shuffling	Reversal	Shifted Reversal	TPP	de Bruijn
Total PSM	287,913	287,913	287,841	287,871	287,875	287,860
Total FP	104,141	10,4141	80,070	79,854	91,301	70,221
% FDR	Target					
0.1	76,640	80,656	79,139	79,358	80,366	83,466
0.5	88,332	93,931	92,751	93,586	91,919	95,598
1	94,198	100,255	98,573	98,922	96,847	101,801
5	110,006	119,632	117,431	117,540	113,415	121,798
10	120,349	137,619	132,634	132,381	125,557	140,031

Table 3.7: Numbers of target identifications at five different % FDR thresholds with different decoy methods.

	Random Shuffling	Normalized Shuffling	Reversal	Shifted Reversal	TPP	de Bruijn
Total PSM	287,913	287,913	287,841	287,871	287,875	287,860
Total FP	104,141	10,4141	80,070	79,854	91,301	70,221
% FDR	Decoy					
0.1	76	155	79	79	80	83
0.5	443	913	466	470	461	480
1	951	1,968	995	999	978	1,028
5	5,789	12,747	6,180	6,186	5,969	6,410
10	13,372	32,824	14,737	14,709	13,950	15,559

Table 3.8: Numbers of decoy identifications at five different % FDR thresholds with different decoy methods.

and TPP methods were because of the larger numbers of unique peptides in the decoy databases than in the target. Since the enzyme information was not provided to TPP, it could not correctly preserve all repeating peptides. The reasons for reduced performance of the normalized shuffling and shifted reversal are less obvious, and are rationalized in the following.

For the random shuffling method, the reason is the unbalanced numbers of unique peptides in the target and decoy databases. [39] The normalized shuffling method attempts to fix the random shuffling method's problem by discounting the decoy matches. At least in theory, this should estimate the FDR correctly. However, there is a subtler problem that cannot be fixed by the normalization. Due to an enlarged number of unique decoy peptides, an average spectrum is evaluated against more decoy peptides than target ones. Consequently, some borderline quality but true target PSMs may be outcompeted by a decoy sequence that happens randomly, merely due to a larger number of decoy sequences. This causes an unbalanced loss of many borderline quality but true target PSMs. As a result, the number of identified target peptides is adversely affected. This may be one of the reasons for the drop of the identified target peptides for the normalized shuffling method in Table 3.4.

For the reversal method, the performance drop may come from the high correlation between the target and decoy peptides and the fragment ions in their spectra, respectively. The adverse effect of such a correlation to the FDR estimation was also reported in earlier literature. [123]

The shifted reversal method attempts to fix the reversal method's problem by shifting the K and R amino acids by one position. This should have solved the problem as the precursor masses of the reversed peptides were randomly changed during this process. However, Table 3.4 shows that the shifted reversal method still has a lower number of PSMs. To understand the true reason, additional experiments were carried out to study the differences between the results of the shifted reversal method and the de Bruijn method.

By using the same searching method as described before, the first replicate of the HeLA sample is analyzed with the shifted reversal decoy and de Bruijn decoy, respectively. At 1% FDR, MS-GF+ reported $34,177$ and $35,129$ target PSMs, by using the shifted reversal decoy and de Bruijn decoy, respectively. The E-value thresholds were 0.15603 and 0.21643 while using shifted reversal and de Bruijn as decoys, respectively. Figure 3.6 shows the numbers of decoy PSMs below a given E-value for both methods.

Figure 3.6 indicates that the shifted reversal method resulted in an increased number of decoy matches at the same E-value, which is consistent with its reduced database search performance. The decoy matches with E-value below 0.01 were specifically examined. As shown in Figure 3.7(A), the shifted reversal resulted in 30 decoy matches with E-value at most 0.01. Seven (23.33%) of the 30 spectra have their corresponding significant target matches if the de Bruijn decoy was used instead. The target and decoy sequence pairs of these 7 spectra were examined manually. For 2 of them, the target and decoy sequences are identical. For 4 others, the target and decoy sequences share significant similarity. The comparisons of these five pairs of peptides are shown in Figure 3.8. The other one pair does not share any apparent similarity.

Figure 3.8 suggests that the shifted reversal method still yields non-negligible sequence similarities between the target and decoy databases. The decoy peptides that are similar to the target ones may match the MS/MS spectra with high scores. Therefore, significant decoy matches are produced not randomly but rather systematically. This may explain why the database search performance with shifted decoy method is dropped.

In contrast, this phenomenon is insignificant for the de Bruijn decoy. In fact, as shown in Figure 3.7(B), only 4 significant de Bruijn decoy matches have corresponding target matches found by Andromeda's reversal [28] (shifted reversal) method. None of them has significant similarity between the target and decoy sequences.

A common practice in the target-decoy method is to remove the decoy matches if the decoy peptides also appear in the target database. Such practice cannot completely resolve the problem where many target and decoy peptide pairs are similar (isobaric) but not identical, as shown in Figure 3.8. To confirm this, the statistics for Table 3.4 were repeated after removing the PSMs that MS-GF+ reported an equal score for the best

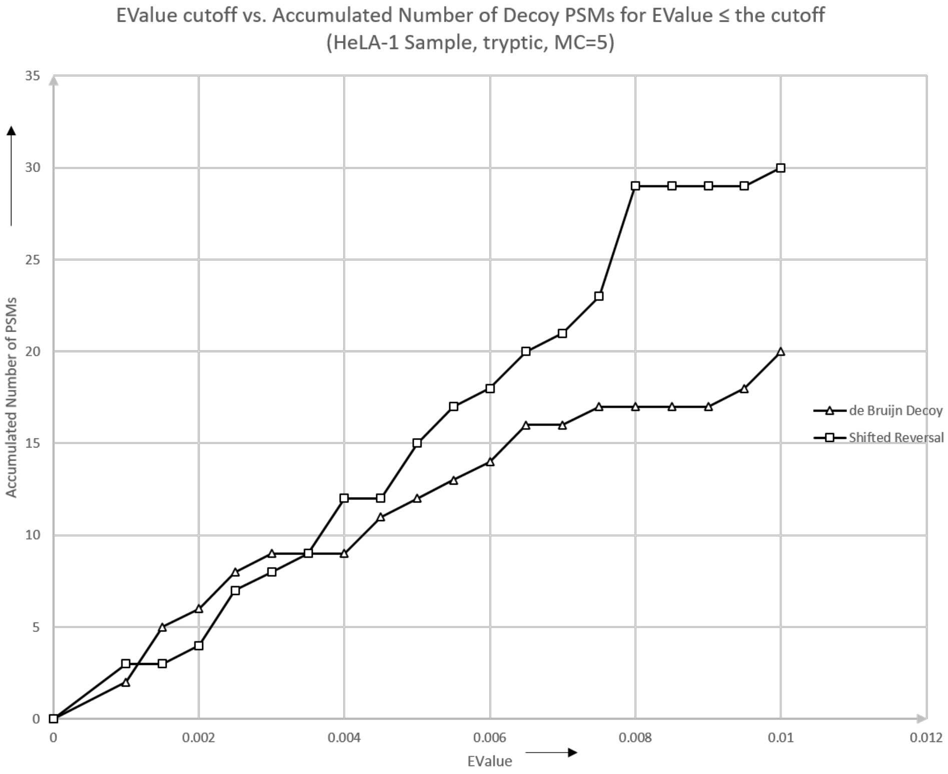


Figure 3.6: Number of decoy PSMs below a given E-value for the shifted reversal and de Bruijn decoy methods, respectively. Our proposed de Bruijn decoy reports fewer decoys at the same E-value compared to the shifted reversal method.

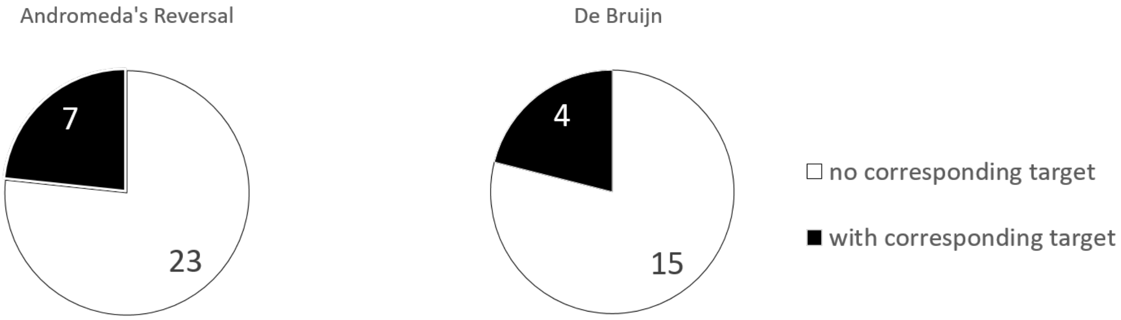


Figure 3.7: (A) Significant (E-value ≤ 0.01) decoy matches produced by shifted reversal (Andromeda's reversal) that have (and have no) corresponding target matches found by de Bruijn decoy at 1% FDR. (B) Significant (E-value ≤ 0.01) decoy matches produced by de Bruijn decoy that have (and have no) corresponding target matches found by shifted reversal at 1% FDR.

```
DRYDSDRYR     EA[L]A[Q(+0.98)]LQ(+0.98)[    RE   ]K
DRYDSDRYR     EA[I]A[   E    ]LQ(+0.98)[Q(+0.98)R]K

LKEELEEAR    [DI][   D    ][I]HEVR      [LY]DAY[EL]K
LKEELEEAR    [ID][N(+0.98)][L]HEVR      [YL]DAY[IE]K

R[    RE   ]EEMMIR
R[Q(+0.98)R]EEMMIR
```

Figure 3.8: Alignments between six significant decoy peptides found by shifted reversal and their corresponding target sequences found by de Bruijn decoy. For each alignment, the top and bottom sequences are the target and decoy sequences, respectively. The numbers in the brackets indicate PTMs. The square brackets indicate the top and bottom sequences in the same alignment block are isobaric.

	Random Shuffling	Normalized Shuffling	Reversal	Shifted Reversal	TPP	de Bruijn
E-Value Cutoff	0.11782	0.27158	0.14796	0.15224	0.13013	0.21122
Target	94,519	100,579	98,932	99,208	96,838	102,080
Decoy	954	(*)1,975	999	1,002	978	1,031
Expected Correct	93,565	(*)99,554	97,933	98,206	95,860	101,049

Table 3.9: Numbers of PSMs at 1% FDR with different decoy methods (decoy matches removed if the decoy peptides also appear in the target database). (*)A normalization factor of 0.519 is multiplied to calculate the FDR for the normalized shuffling method.

target and best decoy peptides, respectively. The results are shown in Table 3.9. The number of identified peptides at 1% FDR for each decoy method only increased marginally in Table 3.9 when compared with that of Table 3.4. However, the relative performance of different decoy methods did not change.

3.3.4 Discussion

Problems Caused by Not Preserving the Repeats

When generating the decoy database, the simple random shuffling method does not preserve repeats in the target database. For the human proteome database, it is shown in Table 3.1 that the number of unique peptides in the decoy database is almost twice as many as the number of unique peptides in the target database. This is consistent with the results previously reported [39, 126, 38]. The excessive number of unique decoy peptides will cause the following adversary effects to the target-decoy method:

1. It reduces the probability that a spectrum matches the correct target peptide.
2. It inflates the estimated false discovery rate due to a higher chance of generating a high-scoring decoy peptide-spectrum match.

The adversary effects are demonstrated in Table 3.4. The random shuffling method seriously overestimates FDR and causes a reduced number of PSMs reported at 1% FDR. The normalized shuffling method (unique peptide coefficient) proposed in the literature [126] attempts to correct this overestimation through multiplying the number of decoy hits by the unique peptide ratio. Our data showed that this simple fix leads to another problem that reduces the number of identified target peptides. Because of the enlarged set of unique decoy peptides, many true target PSMs are lost during the search due to the competition from the decoy. Although the coefficient may correct the FDR over-estimation, the lost target PSMs are lost forever.

Repeat-Preserving Property of de Bruijn Decoy

Our proposed de Bruijn decoy method preserves the structures of the repeats in the target database to a great extent. More precisely, the de Bruijn decoy method using k-mer vertex labels ensures the following repeat-preserving property:

If two proteins contain a common peptide sequence of length L, their corresponding de Bruijn decoy proteins will contain a common peptide sequence with length at least $L - k$.

To prove this repeat-preserving property, suppose two proteins share a length-L peptide sequence $a_1 a_2 \cdots a_L$. In the de Bruijn graph, the two proteins' corresponding paths will start overlapping at vertex $a_1 a_2 \cdots a_k$. In fact, all the vertices $a_i a_{i+1} \cdots a_{i+k}$ are shared for $i = 1, 2, \cdots, L - k + 1$. Thus, the two paths share a sub-path of length at least $L - k$ edges. The shared sub-path will produce a shared peptide sequence of length at least $L - k$ during the generation of the decoy sequences.

Balance between Randomness and Repeat Preservation

Any repeat-preserving decoy generation method needs to maintain a certain level of correlation between different parts of the target database, and therefore cannot be completely random. However, a significant level of randomness is useful. In fact, when the decoy database is generated with a deterministic rule (such as the reversal and shifted reversal methods), our data showed that the false positive matches in the decoy database may be systematically increased. A detailed examination of the high-scoring false decoy matches suggested that this is a result of the subtle similarities between the generated decoy peptides and the target peptides (and the peptide-spectrum-match scoring method used by the search engine). Conversely, it is entirely possible that another deterministic generation method may systematically decrease the decoy hits. In either case, the FDR estimation is systematically biased. Randomness will help mitigate the systematic bias.

In our de Bruijn decoy method, the parameter k can provide fine control to trade between the level of randomness and the repeat-preserving capability. Since each edge is produced by a $(k + 1)$-mer in the database sequences, the number of edges in the graph is upper bounded by 20^{k+1}. Here 20 is the number of different amino acids. Therefore, the

	$k = 2$			$k = 3$	$k = 4$
E-Value cutoff	0.2039	0.20361	0.19795	0.2006	0.20069
Target	101,801	101,526	101,092	101,257	101,201
Decoy	1,028	1,025	1,021	1,022	1,022
Expected correct	100,773	100,501	100,071	100,235	100,179

Table 3.10: Numbers of PSMs at 1% FDR with de Bruijn decoys produced by applying different k values. $k = 2$ is repeated three times. Consistent results in multiple de Bruijn decoy generation runs demonstrate its robustness and reliability, unaffected by k value variations or iteration changes.

decoy generation is completely determined by up to 20^{k+1} random parameters (replacements of edge labels). Increasing k will increase the number of random parameters and therefore the randomness level. However, at the same time, the preserved length $(L - k)$ of a length-L repeat is also reduced.

Our experiments (Table 3.10) show that $k = 2$ can provide satisfactory results as compared to $k = 3$ or 4. Also, the number of identifications at 1% FDR is fairly stable when the decoy generation is repeated with a different random seed. Additionally, Table 3.11 represents the number of semi-tryptic peptides for three different de Bruijn decoys. Despite the incorporation of the randomization process in decoy generation, the numbers do not vary significantly. Results from three different de Bruijn decoys constructed for $k = 2$ confirm that the results do not vary through different runs of de Beuijn decoy generation. Furthermore, the number of identifications for different k values underlines that k-mer length does not have a substantial impact on the results.

#Peptides (thousands)	Target	de Bruijn		
Total	145,522	146,686	146,954	146,545
Unique	63,426	64,195	64,156	63,970

Table 3.11: The numbers of semi-tryptic peptides in a target and three different decoy databases generated with the de Bruijn method ($k = 2$) in three repeats. The number of unique peptides remains consistent with the target across all three instances.

Programming Simplicity and Efficiency

Although de Bruijn decoy method is based on a rigorous mathematical model, the software implementation is surprisingly simple. It only requires sequentially scanning through the target protein database twice. The first time builds the hash map for all the occurring $(k+1)$-mers. The second time translates each protein sequence into a decoy sequence. Most programming languages have the built-in support to the hash map data structure. This makes the programming fairly straightforward. The memory footprint is also extremely small. The protein sequences can be sequentially read from files, and the decoy sequences can be written to files on the fly while they are generated. The hash map is the only large object to be kept in the main memory, which contains no more than 20^{k+1} entries. When $k = 2$, there are only 8000 entries. This makes it consume less than 1M bytes of memory. An example implementation of the algorithm in Java can be found at https://github.com/johramoosa/deBruijn.

3.3.5 Conclusion

We present a new repeat-preserving strategy for decoy sequence generation for improved FDR estimation. A mathematically rigorous and easy-to-implement method, de Bruijn decoy, is proposed to generate decoy sequences that preserve the repeat structures in the protein sequences. Experimental results demonstrated the importance of the repeat-preservation property and the good performance of the de Bruijn decoy method.

3.4 Supporting Information

This section includes supplementary files related to the chapter.

3.4.1 Supporting xlsx Files

The following xlsx files are provided as supporting information. Truncated tables are provided here, showing PSMs where at least one method achieved an EValue ≤ 0.01. The complete files are available to download free of charge at https://pubs.acs.org/doi/10.1021/acs.jproteome.9b00555.

- SupportingInformation1: Top ranked decoy PSMs found by de Bruijn and shifted reversal decoy method for tryptic search

Table 3.12: Decoy PSMs reported within 1%FDR, Highlighted part denotes the PSMs with EValue ≤ 0.01; List sorted by EValue, Sample: HeLa 1. A truncated table is provided here, showcasing PSMs where at least one method achieved an EValue ≤ 0.01. For the full table, please refer to https://pubs.acs.org/doi/10.1021/acs.jproteome.9b00555

de Bruijn Decoy				Shifted Reversal			
Scan Num	Peptide	Protein	EValue	Scan Num	Peptide	Protein	EValue
21746	K.RMSVSQ+0.984FIQ+0.984RK.L	DeBruijnsp|P49913|CAMP_HUMAN	0.00051	8691	R.DRYDS DRYR.D	ReverseMQsp|P23588|IF4B_HUMAN	0.00003
38967	R.AQ+0.984AVDYRN+0.984KWRPMLSYK.F	DeBruijnsp|Q8N9B8|RGF1A_HUMAN	0.00072	113466	K.PPLQQAEARERKQ+0.984KPPSQ+0.984EVEDRVR.K	ReverseMQsp|E9PQR5|NPIB8_HUMAN	0.00027
61310	R.C+57.021RAIDAYFGHN+0.984MHVPRR.E	DeBruijnsp|Q6P3X3|TTC27_HUMAN	0.00114	26705	R.KSTG RDNR.A	ReverseMQsp|Q5JTC6|AMER1_HUMA	0.0008
36416	R.-17.027QEPLGSQ+0.984GPGQMIR.R	DeBruijnsp|Q9P2M7|CING_HUMAN	0.00118	88592	R.VTQTASFLM+15.995TRRQ+0.984TPLK.L	ReverseMQsp|Q9NVP1|DDX18_HUMAN	0.00172
51480	K.KELDSGQ+0.984QIK.E	DeBruijnsp|Q68DX3|FRPD2_HUMAN	0.00136	25506	R.RFFFDKNC+57.021YK.L	ReverseMQsp|Q9H2C1|LHX5_HUMAN	0.00204
103141	-.M+15.995SVC+57.021MPTLSPRLLHLELTRMKR.E	DeBruijnsp|Q9Y3A5|SBDS_HUMAN	0.00163	13530	R.HIAEE PDHR.S	ReverseMQsp|Q29RF7|PDS5A_HUMAN	0.00209
17042	K.RPLTN+0.984GALDELR.K	DeBruijnsp|Q8TE99|PXYP1_HUMAN	0.00221	17892	K.LKEEL EEAR.E	ReverseMQsp|Q8NEG2|CG057_HUMAN	0.00239
48551	R.TIGDLREIN+0.984DPSLPR.A	DeBruijnsp|Q9UKX5|ITA11_HUMAN	0.00246	6375	-.RMFKHIFLDHRK.L	ReverseMQtr|H0Y5L8|H0Y5L8_HUMAN	0.00257
45273	K.KMQN+0.984MSQSQK.K	DeBruijnsp|Q6PL18|ATAD2_HUMAN	0.00264	37513	R.LTVTEQKQ+0.984RSTLEAAQHEEQ+0.984LR.G	ReverseMQsp|Q8N137|CNTRB_HUMAN	0.00315
74602	M.LTRLP DYRK.E	DeBruijnsp|O75971|SNPC5_HUMAN	0.00418	72194	R.MTLLLMKEKTWEEEK.R	ReverseMQsp|Q5CZC0|FSIP2_HUMAN	0.00352

de Bruijn Decoy				Shifted Reversal			
Scan Num	Peptide	Protein	EValue	Scan Num	Peptide	Protein	EValue
8928	K.KLKLTEQ+0.984LQ+0.984K.R	DeBruijnsp|Q07890|SOS2_HUMAN	0.00427	26399	K.EAIAELQ+0.984Q+0.984RK.N	ReverseMQsp|Q8NCU4|CC191_HUMAN	0.00359
103377	-.M+15.995SVC+57.021MPTLSPRLLHLELTRMKR.E	DeBruijnsp|Q9Y3A5|SBDS_HUMAN	0.00452	32170	K.IDN+0.984LHEVR.P	ReverseMQsp|P30532|ACHA5_HUMAN	0.00386
51467	K.KMHF DKLK.K	DeBruijnsp|Q8NC60|NOA1_HUMAN	0.00513	38974	K.SPWNM+15.995LYELIGAM+15.995PK.R	ReverseMQsp|Q5TGI4|SAMD5_HUMAN	0.00496
110037	R.ILEQMINRLGRAMVEPGRRGAR.E	DeBruijnsp|Q5VZP5|DUS27_HUMAN	0.00564	34862	-.-17.027QVLHELC+57.021HN+0.984LR.F	ReverseMQtr|D6RBW5|D6RBW5_HUMA	0.00496
48841	R.LETQ+0.984EISQSIETMK.K	DeBruijnsp|Q86UQ4|ABCAD_HUMAN	0.006	38321	K.KQ+0.984VHDIER.S	ReverseMQsp|P02538|K2C6A_HUMAN	0.00496
102359	-.M+15.995SVC+57.021MPTLSPRLLHLELTRMKR.E	DeBruijnsp|Q9Y3A5|SBDS_HUMAN	0.0062	96605	R.TNLDDLDALKMQ+0.984 DRNRENTVEQ+0.984C+57.021KLNYYEQKITTK.A	ReverseMQsp|Q9BQS8|FYCO1_HUMAN	0.00504
20327	R.YTQ+0.984QIEEVTEN+0.984R.S	DeBruijnsp|Q9P273|TEN3_HUMAN	0.00725	45737	K.YLDA YIEK.L	ReverseMQsp|A3KN83|SBNO1_HUMAN	0.00544
41848	R.C+57.021ALKSSWISQ+0.984RWIDVGYR.G	DeBruijnsp|Q5T200|ZC3HD_HUMAN	0.0093	104900	K.AQLEDELAHFVRKQ+0.984EDVERSWKQ+0.984LTSILNEINK.N	ReverseMQsp|Q03001|DYST_HUMAN	0.00575
95606	K.DQ+0.984ILRQHSEYHTVR.E	DeBruijnsp|Q9Y2U9|KLDC2_HUMAN	0.00999	69616	R.FKYRN+0.984RQLKYDK.Q	ReverseMQsp|Q5VV17|OTUD1_HUMAN	0.00606
102663	-.M+15.995SVC+57.021MPTLSPRLLHLELTRMKR.E	DeBruijnsp|Q9Y3A5|SBDS_HUMAN	0.01028	22191	K.QLN+0.984QMEANLNENHRER.N	ReverseMQsp|O75330|HMMR_HUMAN	0.00628

de Bruijn Decoy				Shifted Reversal			
Scan Num	Peptide	Protein	EValue	Scan Num	Peptide	Protein	EValue
68002	K.PILIFPEC+57.021EGIFLPR.V	DeBruijnsp|Q2WGJ9|FR1L6_HUMAN	0.01074	21274	R.RQ+0.984AAALKAK.P	ReverseMQsp|Q6ZN01|MASTR_HUMA	0.00666
49802	R.TITM+15.995C+57.021VHDLHETILPN+0.984SR.V	DeBruijnsp|Q8IUK5|PLDX1_HUMAN	0.01123	14611	K.RQ+0.984REEMMIR.H	ReverseMQsp|Q13227|GPS2_HUMAN	0.00709
25175	K.ITSETLQ+0.984R.C	DeBruijnsp|Q8NEG4|FA83F_HUMAN	0.01219	46869	K.QLVSC+57.021M+15.995FVEM+15.995R.P	ReverseMQsp|Q9NR31|SAR1A_HUMAN	0.00748
77624	R.TTHDNQ+0.984KEHIDETIVK.K	DeBruijnsp|Q96BQ5|CC127_HUMAN	0.01301	68333	R.APSFLLNLVK.A	ReverseMQsp|P07101|TY3H_HUMAN	0.00754
25815	R.NMLN+0.984LPKLN+0.984TKFR.S	DeBruijnsp|Q9NZM1|MYOF_HUMAN	0.0133	43792	R.IHSHELIAMLK.A	ReverseMQsp|Q9UHL9|GT2D1_HUMAN	0.00762
8632	K.YETQ+0.984Q+0.984ENQR.Q	DeBruijnsp|Q8NBZ9|NEAS1_HUMAN	0.01345	6729	K.YYNPHDQMHQ+0.984RVR.L	ReverseMQsp|Q9H6I2|SOX17_HUMAN	0.00765
28048	K.WC+57.021KNLN+0.984VSK.V	DeBruijnsp|Q14789|GOGB1_HUMAN	0.01356	25139	R.DEQSFIADKAEVK.S	ReverseMQsp|Q9HBZ2|ARNT2_HUMA	0.00773
86897	R.RREHLEEVTQ+0.984GWFHVPTR.I	DeBruijnsp|Q15772|SPEG_HUMAN	0.01398	46905	R.TLQQQREEKLEQHLR.L	ReverseMQsp|Q01664|TFAP4_HUMAN	0.00783
96972	K.VMLSVFN+0.984DIDNRSIHR.I	DeBruijnsp|P09758|TACD2_HUMAN	0.01409	33669	R.AQ+0.984VLKFVC+57.021DLHGR.A	ReverseMQsp|Q6ZVH7|ESPNL_HUMA	0.00796
90492	K.EAKIHALILM+15.995R.G	DeBruijnsp|P51530|DNA2_HUMAN	0.01436	107108	K.C+57.021QRIEVVLLQEEM+15.995R.S	ReverseMQsp|P78316|NOP14_HUMAN	0.00956

- SupportingInformation2: PSMs matched as target by de Bruijn, and decoy by shifted reversal for tryptic search

- SupportingInformation3: PSMs matched as target by shifted reversal, and decoy by de Bruijn for tryptic search

Table 3.13: Decoy PSMs reported within 1%FDR, Highlighted part denotes the PSMs with EValue ≤ 0.01; List sorted by EValue, Sample: HeLa 1. A truncated table is provided here, showcasing PSMs where EValue ≤ 0.01. For the full table, please refer to https://pubs.acs.org/doi/10.1021/acs.jproteome.9b00555

Shifted Reversal				
ScanNum	Charge	Peptide	Protein	EValue
8691	3	R.DRYDS DRYR.D	ReverseMQsp|P23 588|IF4B_HUMAN	2.67E-05
113466	3	K.PPLQQAEARERKQ+0.984 KPPSQ+0.984EVEDRVR.K	ReverseMQsp|E9P QR5|NPIB8_HUMAN	2.68E-04
26705	2	R.KSTG RDNR.A	ReverseMQsp|Q5J TC6|AMER1_HUMAN	0.00079837
88592	2	R.VTQTASFLM+15.99 5TRRQ+0.984TPLK.L	ReverseMQsp|Q9N VP1|DDX18_HUMAN	0.0017153
25506	2	R.RFFFDKNC+ 57.021YK.L	ReverseMQsp|Q9H 2C1|LHX5_HUMAN	0.0020413
13530	2	R.HIAEE PDHR.S	ReverseMQsp|Q29 RF7|PDS5A_HUMAN	0.0020934
17892	2	K.LKEEL EEAR.E	ReverseMQsp|Q8N EG2|CG057_HUMAN	0.0023912
6375	4	-.RMFKHI FLDHRK.L	ReverseMQtr|H0Y5 L8|H0Y5L8_HUMAN	0.002571
37513	5	R.LTVTEQKQ+0.984RST LEAAQHEEQ+0.984LR.G	ReverseMQsp|Q8N 137|CNTRB_HUMAN	0.0031529
72194	3	R.MTLLLMKE KTWEEEK.R	ReverseMQsp|Q5C ZC0|FSIP2_HUMAN	0.0035173
26399	2	K.EAIAELQ+0.9 84Q+0.984RK.N	ReverseMQsp|Q8N CU4|CC191_HUMAN	0.0035852
32170	2	K.IDN+0.9 84LHEVR.P	ReverseMQsp|P30 532|ACHA5_HUMAN	0.0038609
38974	2	K.SPWNM+15.995LYE LIGAM+15.995PK.R	ReverseMQsp|Q5T GI4|SAMD5_HUMAN	0.0049589
34862	2	-.-17.027QVLHELC+5 7.021HN+0.984LR.F	ReverseMQtr|D6RB W5|D6RBW5_HUMAN	0.0049603
38321	2	K.KQ+0.98 4VHDIER.S	ReverseMQsp|P02 538|K2C6A_HUMAN	0.0049642

Shifted Reversal				
ScanNum	Charge	Peptide	Protein	EValue
96605	4	R.TNLDDLDALKM Q+0.984DRNRENTVE Q+0.984C+57.021KLNYYEQKITTK.	ReverseMQsp|Q9B QS8|FYCO1_HUMAN	0.0050413
45737	2	K.YLDA YIEK.L	ReverseMQsp|A3K N83|SBNO1_HUMAN	0.0054371
104900	4	K.AQLEDELAHFVRKQ+0.984EDV ERSWKQ+0.984LTSILNEINK.N	ReverseMQsp|Q03 001|DYST_HUMAN	0.0057492
69616	3	R.FKYRN+0.9 84RQLKYDK.Q	ReverseMQsp|Q5V V17|OTUD1_HUMAN	0.0060608
22191	4	K.QLN+0.984QM EANLNENHRER.N	ReverseMQsp|O75 330|HMMR_HUMAN	0.0062838
21274	2	R.RQ+0.984 AAALKAK.P	ReverseMQsp|Q6Z N01|MASTR_HUMAN	0.0066566
14611	3	K.RQ+0.984 REEMMIR.H	ReverseMQsp|Q13 227|GPS2_HUMAN	0.007092
46869	3	K.QLVSC+57.021M+15 .995FVEM+15.995R.P	ReverseMQsp|Q9N R31|SAR1A_HUMAN	0.0074764
68333	2	R.APSFL LNLVK.A	ReverseMQsp|P07 101|TY3H_HUMAN	0.0075397
43792	3	R.IHSHEL IAMLK.A	ReverseMQsp|Q9U HL9|GT2D1_HUMAN	0.0076246
6729	4	K.YYNPHDQMHQ +0.984RVR.L	ReverseMQsp|Q9H 6I2|SOX17_HUMAN	0.0076488
25139	3	R.DEQSFIA DKAEVK.S	ReverseMQsp|Q9H BZ2|ARNT2_HUMAN	0.0077262
46905	4	R.TLQQQREE KLEQHLR.L	ReverseMQsp|Q01 664|TFAP4_HUMAN	0.0078342
33669	3	R.AQ+0.984VLKFV C+57.021DLHGR.A	ReverseMQsp|Q6Z VH7|ESPNL_HUMAN	0.0079628
107108	2	K.C+57.021QRIEVV LLQEEM+15.995R.S	ReverseMQsp|P78 316|NOP14_HUMAN	0.0095576

Table 3.14: Decoy PSMs reported within 1%FDR, Highlighted part denotes the PSMs with EValue ≤ 0.01; List sorted by EValue, Sample: HeLa 1. A truncated table is provided here, showcasing PSMs where EValue ≤ 0.01. For the full table, please refer to https://pubs.acs.org/doi/10.1021/acs.jproteome.9b00555

DeBruijn Decoy				
ScanNum	Charge	Peptide	Protein	EValue
21746	2	K.RMSVSQ+0.984 FIQ+0.984RK.L	DeBruijnsp|P49 913|CAMP_HUMAN	0.00051
38967	5	R.AQ+0.984AVDYRN+ 0.984KWRPMLSYK.F	DeBruijnsp|Q8N9 B8|RGF1A_HUMAN	0.00072
61310	4	R.C+57.021RAIDAYF GHN+0.984MHVPRR.E	DeBruijnsp|Q6P3 X3|TTC27_HUMAN	0.00114
36416	3	R.-17.027QEPLGSQ +0.984GPGQMIR.R	DeBruijnsp|Q9P 2M7|CING_HUMAN	0.00118
51480	2	K.KELDSGQ+ 0.984QIK.E	DeBruijnsp|Q68D X3|FRPD2_HUMAN	0.00136
103141	3	-.M+15.995SVC+57.021 MPTLSPRLLHLELTRMKR.E	DeBruijnsp|Q9Y 3A5|SBDS_HUMAN	0.00163
17042	3	K.RPLTN+0.9 84GALDELR.K	DeBruijnsp|Q8TE 99|PXYP1_HUMAN	0.00221
48551	3	R.TIGDLREIN+0 .984DPSLPR.A	DeBruijnsp|Q9UK X5|ITA11_HUMAN	0.00246
45273	2	K.KMQN+0.9 84MSQSQK.K	DeBruijnsp|Q6PL 18|ATAD2_HUMAN	0.00264
74602	2	M.LTRLP DYRK.E	DeBruijnsp|O759 71|SNPC5_HUMAN	0.00418
8928	3	K.KLKLTEQ+0.9 84LQ+0.984K.R	DeBruijnsp|Q07 890|SOS2_HUMAN	0.00427
103377	3	-.M+15.995SVC+57.021 MPTLSPRLLHLELTRMKR.E	DeBruijnsp|Q9Y 3A5|SBDS_HUMAN	0.00452
51467	2	K.KMHF DKLK.K	DeBruijnsp|Q8N C60|NOA1_HUMAN	0.00513
110037	3	R.ILEQMINRLGR AMVEPGRRGAR.E	DeBruijnsp|Q5VZ P5|DUS27_HUMAN	0.00564
48841	2	R.LETQ+0.984 EISQSIETMK.K	DeBruijnsp|Q86U Q4|ABCAD_HUMAN	0.006
102359	3	-.M+15.995SVC+57.021 MPTLSPRLLHLELTRMKR.E	DeBruijnsp|Q9Y 3A5|SBDS_HUMAN	0.0062
20327	3	R.YTQ+0.984QIE EVTEN+0.984R.S	DeBruijnsp|Q9P 273|TEN3_HUMAN	0.00725
41848	5	R.C+57.021ALKSSWIS Q+0.984RWIDVGYR.G	DeBruijnsp|Q5T2 00|ZC3HD_HUMAN	0.0093
95606	2	K.DQ+0.984IL RQHSEYHTVR.E	DeBruijnsp|Q9Y2 U9|KLDC2_HUMAN	0.00999

Chapter 4

FDR Estimation: Training with *Next-Best* PSMs

In proteomics, the control of the false discovery rate is a critical aspect that boosts our confidence in the reported Peptide-Spectrum Matches (PSMs). However, this field faces the significant challenge of lacking ground truth data. Decoy databases have traditionally played a pivotal role in addressing this challenge, ensuring the reliability of results. Intriguingly, in cases where target peptides are absent in a sample, they functionally resemble decoy peptides for that specific sample. Leveraging this insight, our approach seeks to harness additional PSMs per spectrum during training, independent of the conventional target-decoy labels.

This chapter provides an in-depth exploration of false discovery rate estimation using machine learning-based methodologies, mainly focusing on the utilization of multiple peptide spectrum matches per spectrum. Our primary aim is to harness this innovative approach to estimate FDR that is on par with traditional target-decoy FDR estimation without necessitating the reliance on target-decoy labeling.

In this chapter, we provide a brief overview of FDR estimation methods, recognizing both the value and limitations of the target-decoy method. In Section 4.1, we delve into some inherent constraints of the target-decoy approach, shedding light on scenarios where alternative FDR estimation methods may offer valuable insights without undermining the efficacy of the powerful target-decoy strategy. Subsequently, in Section 4.2, we review alternative FDR estimation methodologies documented in the literature. In Section 4.3, we explore the opportunities presented by the existence of multiple PSMs per spectrum. This section introduces the concept of *best* and *next-best* PSMs, accompanied by a discussion on the split database search method, which plays a pivotal role in generating a substantial pool of *next-best* PSMs (Section 4.4).

The chapter further delves into the introduction of our proposed machine learning-based methods for FDR estimation (Section 4.5). Within Section 4.6, we propose various strategies to enhance the accuracy of our estimation techniques. Finally, in Section 4.7, we present and comprehensively discuss the results obtained from our research in this domain.

4.1 Challenges of Target-Decoy Approach

Target-decoy database search has dominated bottom-up proteomics due to its simplicity. However, this widely used method can be susceptible to many inherent limitations.

4.1.1 Fairness in FDR Estimation: The Problem of Decoy Discrimination

The target-decoy approach (TDA) to estimate the FDR fails when the underlying theoretical assumptions are not fulfilled in practice. The performance of TDA depends on the decoy's ability to trick the database search engine [26]. An ideal decoy should be statistically similar to the target. However, the difference between the distribution of theoretical target and decoy spectra introduces bias [30]. Danilova et al. [30] showed that it is possible to systematically mislead the scoring function to underestimate the FDR by boosting the target peptide match scores.

Ideally, a scoring function will score all the true PSMs with high scores and the false PSMs with low scores regardless of the peptide or the spectrum in consideration. However, some spectra may receive higher scores in practice due to containing more peaks or their precursor mass, resulting in more candidate peptides [57]. Additionally, skewed scoring functions with the capacity to distinguish a decoy either favor target peptides or discriminate against decoy peptides [70, 59, 30]. In such cases, the target decoy method already encounters a bias. As the target PSMs are favored, the probability of random matches in the target and the decoy are no longer the same. Besides, a machine-learning-based algorithm that involves target and decoy peptides in training can become inclined toward target peptides.

4.1.2 Conflicts of TDA with The High Resolution Data

Because of high resolution and high mass accuracy mass spectrometer data, narrow parent ion mass tolerance is in practice, which results in a lower number of candidate peptides per spectra. FDR estimation using TDA can decrease inaccurately as the numbers of peptide candidates shrink [24]. Consequently, there will be an insufficient number of instances to estimate the false distribution. Therefore, researchers argue that the target-decoy method became less reliable as the mass spectrometers' accuracy improved [26].

4.1.3 Decoy Induced Variability

Randomly generated decoys can cause different expected correct PSMs each time the decoy is generated [72]. To ensure minimum variance, we need a good decoy generation algorithm. Because random decoys are different in each run, the corresponding PSMs also differ. As a result, the FDR estimation can become susceptible to high variability [73]. Intuitively, the use of a reversal decoy should be a simple solution to this problem. Apart from the fact that reversal decoy poses the risk of systematic target matches [90], Keich et al. [73] argue that reversal decoys only mask the problem rather than solving it. Reversal decoy generation can be considered a conservative version of shuffled decoy generation, where the shuffle pattern is fixed and predetermined rather than random. Hence, a reversed decoy is merely one of the instances among all the possible random shuffled decoys.

4.1.4 Difficulties of Ideal Decoy Generation: Why Ideal Decoy Generation is Hard?

Optimal decoy generation still remains an open problem [66]. Our effort to address this issue by preserving the repeats is not flawless either. The de Bruijn decoy generation algorithm requires the knowledge of the cleavage rules to maintain a balanced number of digested unique peptides in the target and the decoy. Additionally, the tradeoff between repeat-preservation and randomness, as detailed in Section 3.3.4, remains an ongoing concern.

An ideal decoy database needs to be similar to the target so that the database search engine fails to distinguish between a target and a decoy. However, they can not be the same either. A decoy peptide should contain all the underlying properties (observed or unobserved) of an original peptide, e.g., mass, length, composition, amino acid dependencies, etc. However, a clear guideline is lacking in the literature as to which decoy generation method is the best [71].

The generation of an ideal decoy is significant to ensure accurate FDR estimation in the target-decoy database search. However, limitations regarding the target-decoy database search approach and the difficulties of an ideal decoy generation method demand alternative FDR estimation methods.

4.2 Alternative FDR Estimation Methods

In this research, we emphasize that decoy databases play a crucial role in proteomics, addressing key challenges and ensuring the reliability of results. Decoys are invaluable for validating the quality of peptide-spectrum matches and estimating the false discovery rate accurately. It is important to note that, while we explore alternative methods, we fully acknowledge and appreciate the significance of the target-decoy database search method in proteomics research. Our goal is to complement, not replace, their use by proposing an alternate approach that can estimate the FDR without relying solely on decoy labeling.

Keller et al. [74] proposed a statistical model, known as PeptideProphet, to distinguish between the correct and incorrect peptides using the search score and the number of tryptic termini. The model estimates the probability of PSMs by establishing a Bayes classifier from two-component (correct and incorrect) probability mixture distributions. The algorithm is optimized to incorporate decoy database search results for a better estimation of the negative distribution, but does not require them. However, the model requires training data with peptide assignments of known validity. Coute et al. [26] proposed a framework that uses Benjamini-Hochberg (BH) method to transform the target PSM scores to produce adjusted p-values. The adaptive multiple-component Gaussian mixture modeling suggested by Renard et al. [102] to compute the confidence of PSMs without the need for decoys. The authors also proposed the incorporation of an additional model to account for the low-quality spectra that are unfit to match any peptide sequence reliably.

4.3 Exploring Database Search Results: Multiple Peptide Spectrum Matches Per Spectrum

Database search engines assign scores depending on the similarity between the observed fragmentation spectrum and the theoretical spectra of the peptides of a reference database.

No. of PSMs	No. of PSMS at 1% FDR	No. of dissimilar PSMs at 1% (possibly co-eluting)
97,284	36,440	1,820

Table 4.1: Numbers of PSMs at 1% FDR reported by default target-decoy method with original database search and modified spectra database search. The count of dissimilar PSMs obtained from these two approaches is minimal, suggesting a low occurrence of potential co-eluting peptides.

A higher degree of similarity leads to a higher matching score. During a database search, all potential candidate peptides are examined for each experimental spectrum, and the top-scoring peptide is chosen. Consequently, it is common for the search engine to report multiple PSMs for a single spectrum. In some instances, these PSMs arise from similar peptides. However, when the peptides differ significantly, two intriguing possibilities emerge: they may represent co-eluting peptides, or they could be random matches.

4.3.1 Mixture Spectra and Co-eluting Peptides

Multiple peptide precursors often co-elute simultaneously in the same tandem mass (MS/MS) spectrum [127]. For example, two dissimilar and different peptides with the same m/z co-eluting in the same RT window. In this case, both peptides should be reported as true PSMs.

To accomplish this, we modify the spectra by eliminating the ions associated with the top-ranked PSM, which is the initially reported match. Subsequently, we search the database again with the modified spectra. This approach enables us to uncover additional peptides associated with the chimeric spectra. Analyzing our baseline HeLa dataset [90], during a fully tryptic database search, we identify additional peptides at 1% FDR that are dissimilar to the best-ranked peptide in approximately 1.87% of the spectra. These are most likely mixture spectra containing co-eluting peptides. It is worth noting that, for more accurate results, while calculating the *next-best* peptides (refer to Section 4.4.2), we can disregard these co-eluting peptides from the list of candidates. However, in our case, none of the *next-best* peptides from the split database search were similar to the co-eluting peptides found from spectra edit. The results are presented in Table 4.1.

4.3.2 Random Peptide Matches

If not co-eluting, theoretically, one spectrum can match only with one single peptide. Because of noises and noise-related uncertainties, the search engine might report some close variations of the peptide. Nevertheless, those variations result from a single peptide. However, when the peptides are dissimilar but matched to one spectrum, the highest-ranked one is most likely the correct match; the rest are highly probable random matches. So, if we can generate a list of high-probability random PSMs, we already have enough information to estimate the false distribution.

4.4 Estimation of False Distribution: Training with *Next-Best* Peptides

Usually, a spectrum has only one correct peptide assignment (except for the rare case of a spectrum produced by a mixture of two or more peptides). Thus, although many peptides are scored against each spectrum, most search engines will only keep one top-scoring or the *best* peptide for each spectrum. Usually, only these top-scoring PSMs from all spectra are used for FDR control. The remaining peptides for each spectrum are seldom correct and were traditionally discarded. However, we realize that these discarded peptides create an excellent opportunity to estimate the score distribution of the top-scoring false PSMs.

We have proposed a method for identifying the false distribution using the discarded peptide assignment data. Analysis of the retained information can provide a distribution of the *next-best* peptides (formally introduced in Section 4.4.2). As a result, theoretically, we should be able to obtain four distributions to explore. (1) Score distribution of the *best* peptides for true spectra (A_1), (2) Score distribution of the *next-best* for true spectra (A_2), (3) Score distribution of *best* peptides for false spectra (B_1), and (4) Score distribution of *next-best* peptides for false spectra (B_2). Among these, only the first one is the correct one. So, our goal is to produce an estimation of the first distribution A_1, i.e., top-ranked score distribution of the true spectra A, as accurately as possible. We will consider A_2 and B_1 to fit the top-ranked false spectra distribution. The rest of the distributions are updated according to this estimation. We will repeat this process until convergence. However, to achieve a satisfactory approximation of the false distribution, we need to ensure that we have *next-best* peptides for the majority of the spectra, as discussed in Section 4.4.5. We employ machine learning-based methods to identify a well-distinguished separation between the true and false distributions. We aim to closely replicate the separation achieved by the target and decoy distributions. This approach enhances our ability to accurately discriminate between correct identifications and false positives, even in the absence of decoy labels. Consequently, we hypothesize that the *next-best* PSMs serve as a suitable approximation for false positives.

4.4.1 True and False Spectra

Due to instrumental errors and noise, some of the MS/MS spectra are merely random signals, and others are generated from actual peptides belonging to the sample. The spectra containing the random signals are the false spectra, B, while the spectra resulting from the actual peptides are true spectra, A. Our main goal is to identify the true spectra, A. However, distinguishing A from B is not trivial in reality. In other words, we do not have a straightforward method to separate the spectra into A and B. Our intention is to assign each spectrum with score x, a probability denoted as $f(x)$, indicating its likelihood of being correct or belonging to group A. Consequently, the complementary probability, $1 - f(x)$, represents the probability that the spectrum is incorrect or belongs to group B. We can derive an estimate of the FDR from the prediction provided by $f(x)$.

4.4.2 Definition: *Next-Best* Peptides

Among the list of peptides for one spectrum, the *best* or the rank-one peptide is the highest-ranked (scored) peptide. Peptides that do not resemble the *best* peptide become candidates for the *next-best* peptide. The *next-best* or the rank-two peptide is the highest-ranked peptide from the candidates of *next-best* peptide. We define these peptides as *next-best*, regardless of their position in the initial list of candidates.

Initially, we assessed the dissimilarity between the *best* peptide and candidate peptides using several distance metrics. Naturally, the first distance metric is the primary sequence similarity. Furthermore, we have computed theoretical ion matches and considered factors such as isobaric peptides (I, L) and post-translational modifications (PTMs) that lead to similar mass values.

Our preliminary studies show a clear distinction in the distribution of the *best* and the *next-best* peptides as presented in Figure 4.1. The overlapping shape matches the distribution of target and decoy peptides. In order to gain a deeper understanding of the correlation, we have presented the distribution of various features associated with both the *best* and *next-best* in the Figure 4.2, 4.3, 4.4, and 4.5. Due to space constraints, the features are divided into two figures, namely, Figures 4.2 and 4.3. The three plots for each feature depict scatter plots focusing primarily on the correlation of their scores

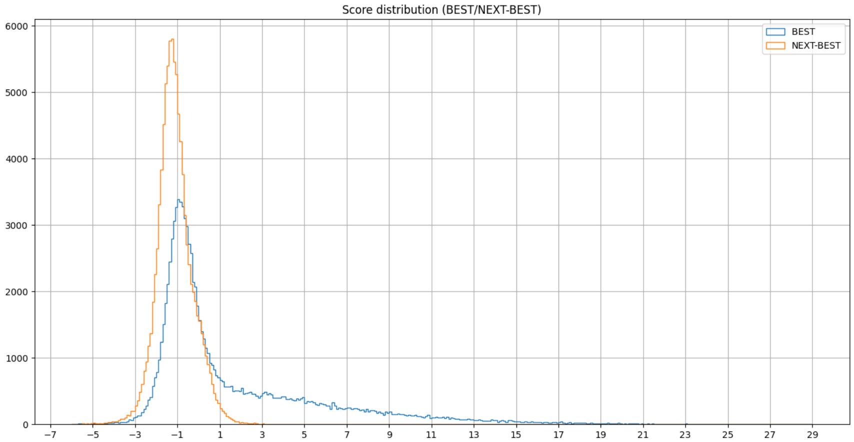
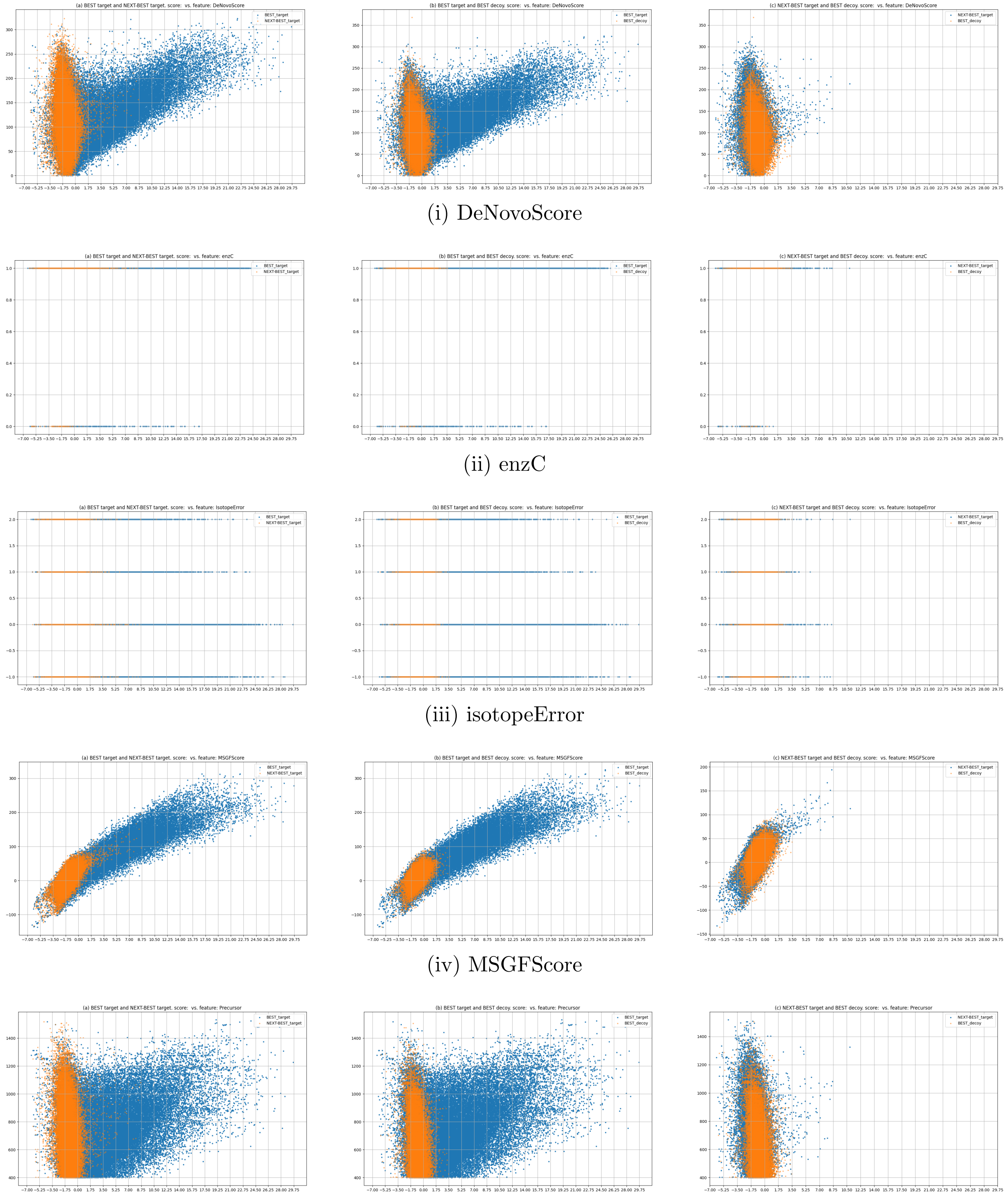


Figure 4.1: Score distribution of *best* (Blue) and *next-best* (Orange) peptides.

versus the feature for different PSM categories. Upon observing the feature distributions for *best* target and *next-best* target PSMs in part (a), as well as target-decoy distributions in part (b) of Figure 4.2 and 4.3, it becomes evident that they exhibit a similar separation pattern. Furthermore, in part (c) of the figures, it is notable that the decoy distribution of *best* PSMs and the target distribution of *next-best* PSMs almost overlap, suggesting that the *next-best* PSMs can serve as a reliable approximation of the false distribution. The results mentioned here are computed using the HeLa dataset, as detailed in Section 4.7.1. Figure 4.4 displays the distribution of 'decoy' for *best* and *next-best* for different features. On the other hand, Figure 4.5 displays the distribution of 'decoy' for *best* and all PSMs for *next-best* for different features. Further statistical analysis, including the use of correlation coefficients, may be necessary for a more precise quantification of these relationships. The details of these features can be found at [75].

Assuming that each spectrum matches only one peptide, we can establish that the remaining peptides for each spectrum are most likely false matches. As a result, the *next-best* peptides can provide us with the A_2, and B_1 distribution. Thus, the *next-best* peptides allow us to estimate the top-scoring false distribution without the use of decoys. A slim chance remains that some of the *next-best* peptides are co-eluting peptides. Based on the results obtained from our experiments, as detailed in Section 4.3.1, it is evident that the number of co-eluting peptides remains remarkably low even after the implementation of the spectral editing process. Additionally, the probability of finding a co-eluting peptide decreases when we perform the split database search.



(i) DeNovoScore

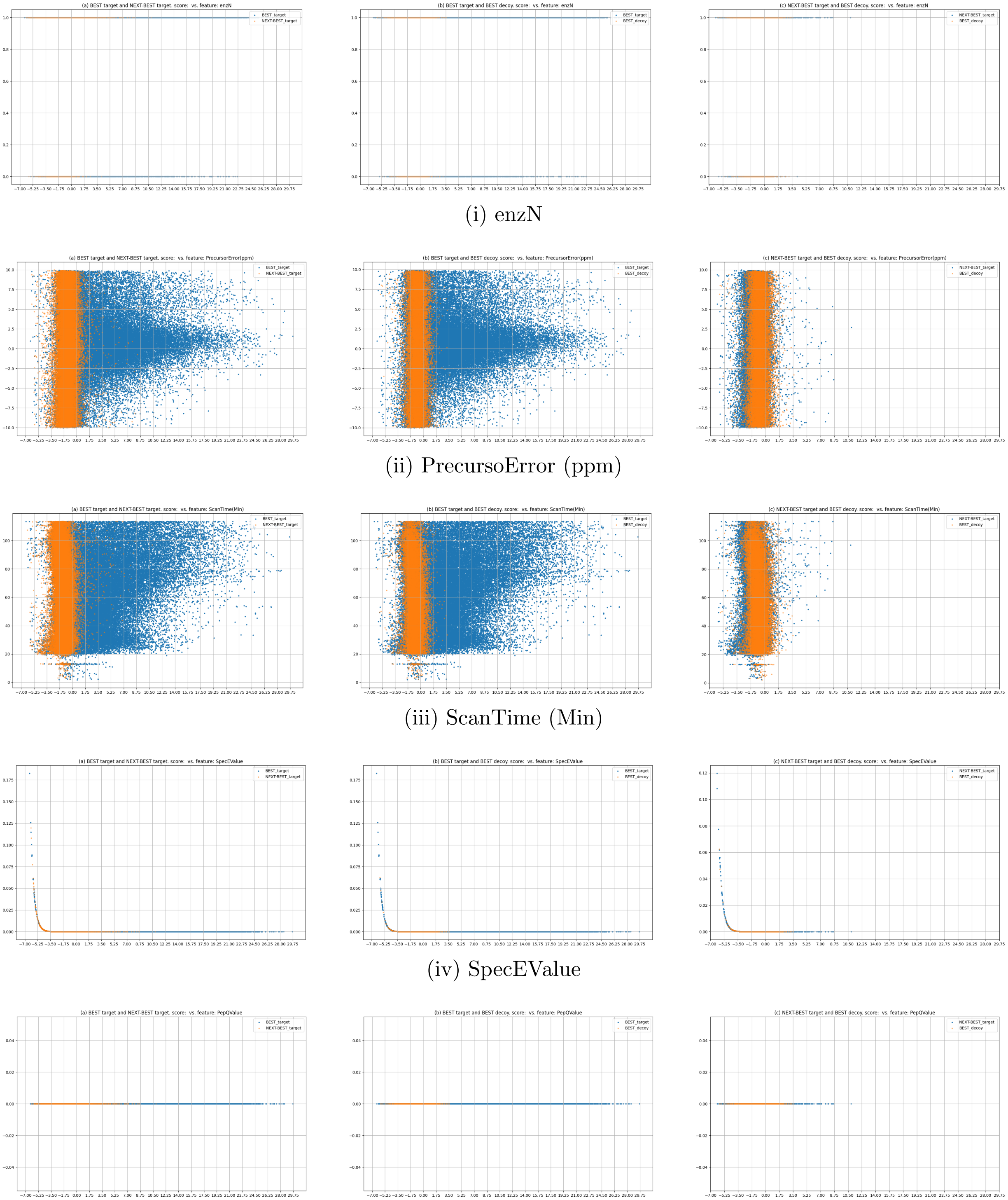
(ii) enzC

(iii) isotopeError

(iv) MSGFScore

(v) Precursor

Figure 4.2: Distribution of (a) *best* target and *next-best* target, (b) *best* target and *best* decoy, and (c) *next-best* target and *best* decoy for score vs. features: (i) DeNovoScore, (ii) enzC (boolean, is C terminal tryptic), (iii) isotopeError, (iv) MSGFScore, and (v) Precursor. The distributions of *next-best* targets and *best* decoys exhibit a significant overlap.



(i) enzN

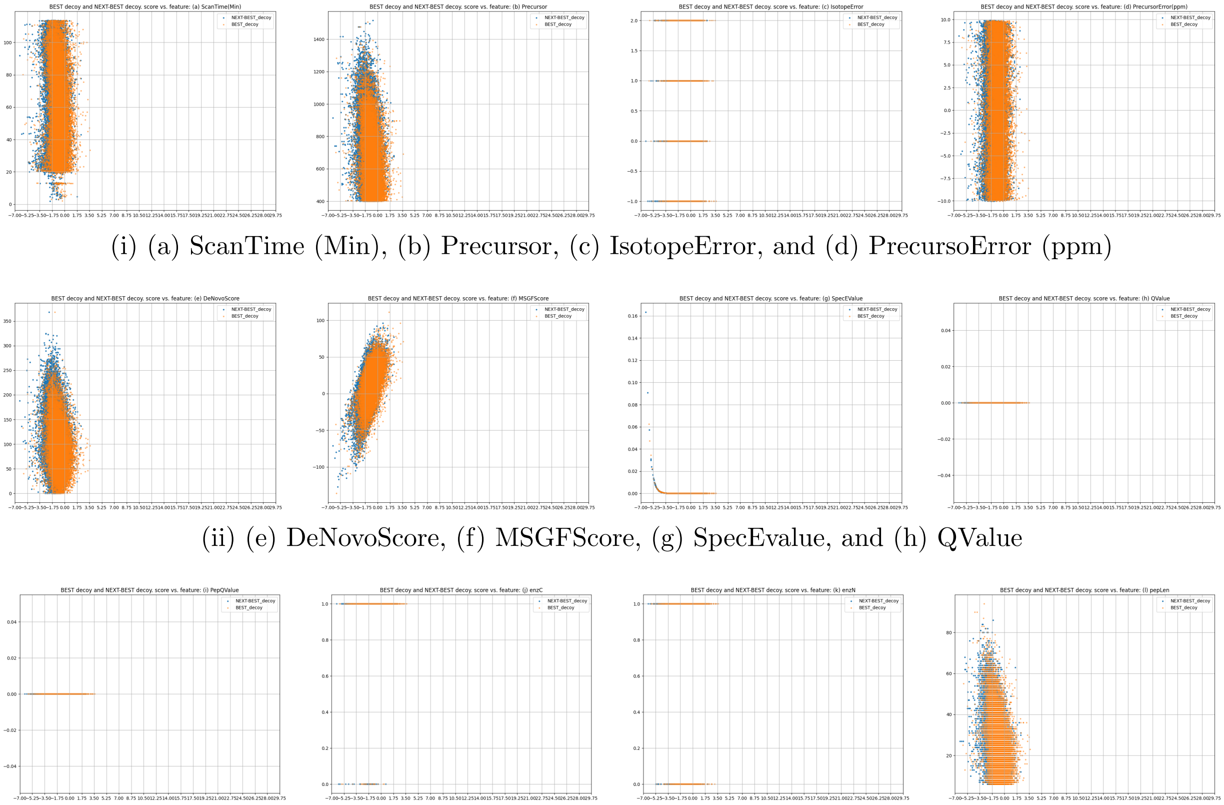
(ii) PrecursoError (ppm)

(iii) ScanTime (Min)

(iv) SpecEValue

(v) PepQValue

Figure 4.3: Distribution of (a) *best* target and *next-best* target, (b) *best* target and *best* decoy, and (c) *next-best* target and *best* decoy for score vs. features: (i) enzN (boolean, is N terminal tryptic), (ii) PrecursoError (ppm), (iii) ScanTime (Min), (iv) SpecEValue, and (v) PepQValue. The distributions of *next-best* targets and *best* decoys exhibit a significant overlap.

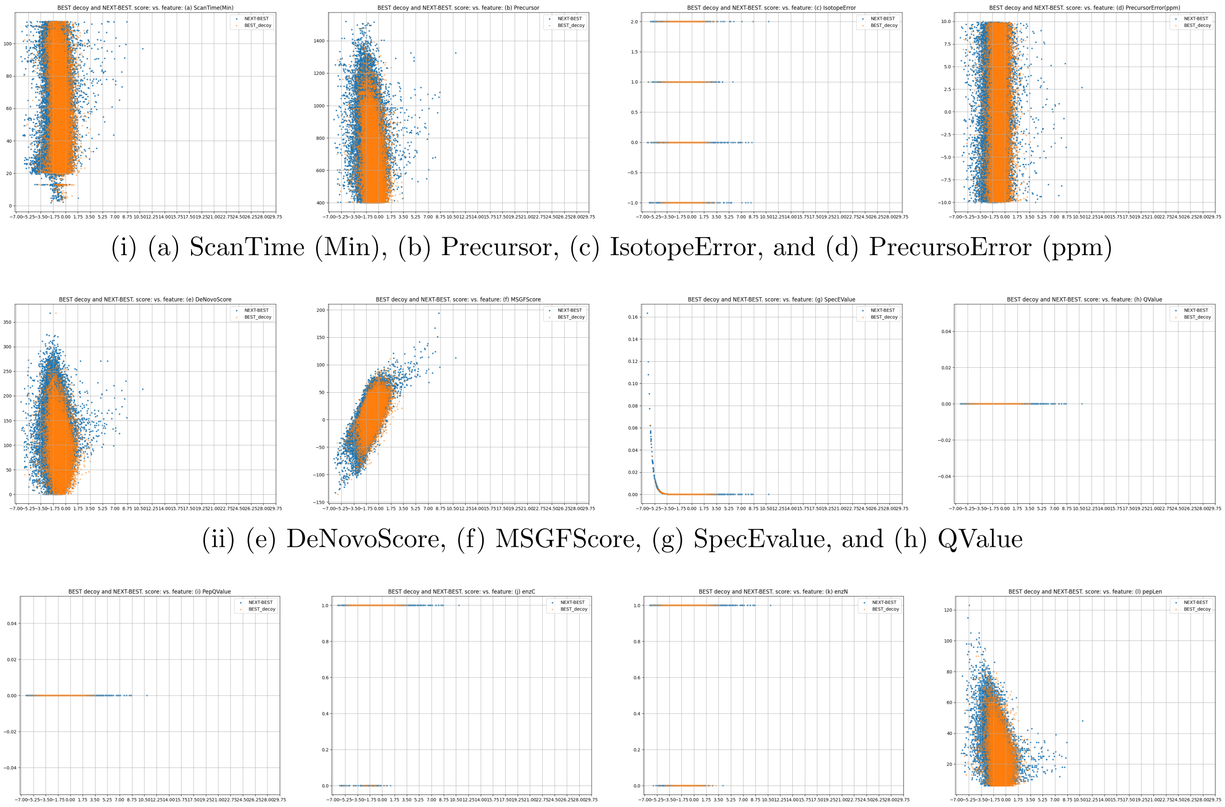


(i) (a) ScanTime (Min), (b) Precursor, (c) IsotopeError, and (d) PrecursoError (ppm)

(ii) (e) DeNovoScore, (f) MSGFScore, (g) SpecEvalue, and (h) QValue

(iii) (i) PepQValue, (j) enzC, (k) enzN, and (l) pepLen

Figure 4.4: Score distribution of only ‘decoy’ PSMs for both *best* and *next-best* across various features.



(i) (a) ScanTime (Min), (b) Precursor, (c) IsotopeError, and (d) PrecursoError (ppm)

(ii) (e) DeNovoScore, (f) MSGFScore, (g) SpecEvalue, and (h) QValue

(iii) (i) PepQValue, (j) enzC, (k) enzN, and (l) pepLen

Figure 4.5: Score distribution of *best* (top-scoring) decoy PSMs and all *next-best* PSMs for different features.

4.4.3 Definition: Third-Best Peptides

Similar to the calculation of *next-best* peptides, we can further calculate *third-best* peptides. The *third-best* peptide (also referred to as rank-three peptide) for a spectrum is defined as the highest-scoring peptide that differs from the *next-best* peptides and, consequently, from the *best* peptide as well. The score distribution of *third-best* PSMs compared to the *best* and *next-best* PSMs is presented in Figure 4.6. As expected, we can observe a shift between the *best* and *next-best*, and *next-best* and *third-best*, particularly when the score is higher. This observed shift aligns with our expectations and underscores the significance of these score differentials in our analysis. We can compute the shift between the *next-best* and *third-best* scores, allowing us to adjust the *next-best* scores to align more effectively with the false *best* PSMs. This compensation for the observed shift by augmenting the *next-best* distribution will be discussed in detail in Section 4.6.2.

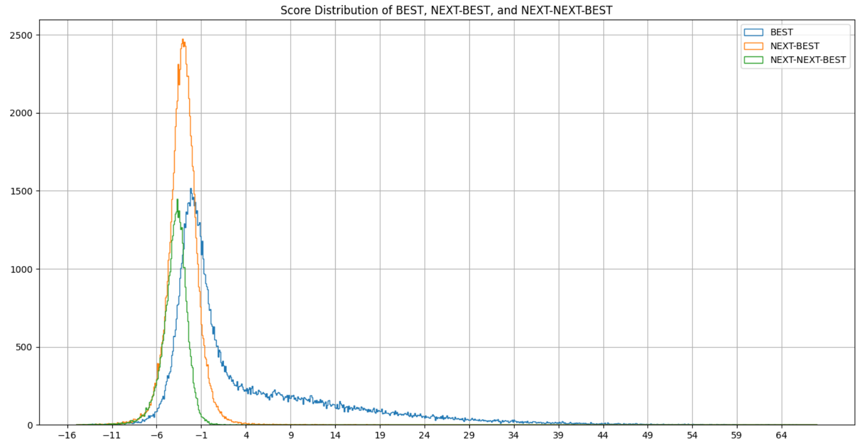


Figure 4.6: Score distribution of *best* (Blue), *next-best* (Orange), and *third-best* (Green) peptides.

4.4.4 Quantifying Peptide Similarity: Determining the '*Best*' and '*Next-Best*' for Each Spectrum

The first crucial step in generating the *next-best* candidate peptides involves assessing the degree of similarity between peptides. In our study, this assessment is based on two key aspects: (1) their sequence similarity and (2) their fragment ion similarity. To evaluate sequence similarity, we employ the computation of the edit distance, also known as the

Levenshtein distance [95], between the primary sequences of each peptide. It is important to note that when considering sequence similarity, we treat isobaric peptides (I, L) and post-translational modifications (PTMs) resulting in similar masses (as discussed in Section 2.3) as interchangeable, enhancing our ability to capture nuanced similarities.

In addition to sequence similarity, we also measure fragment-ion similarity by comparing the theoretical spectra of the peptides rather than their sequences. This approach provides a more comprehensive assessment of the similarity between the peptides. Further details on this peptide similarity measure can be found in Section 5.2.2. It is worth noting that this multifaceted approach enables us to determine and select *next-best* candidate peptides properly based on their resemblance to the *best* peptides, thereby enhancing the accuracy of our FDR estimation. By following this procedure to calculate the *next-best*, we ensure the faithful application of our hypothesis, as the reported *best* and *next-best* PSMs are maximally dissimilar.

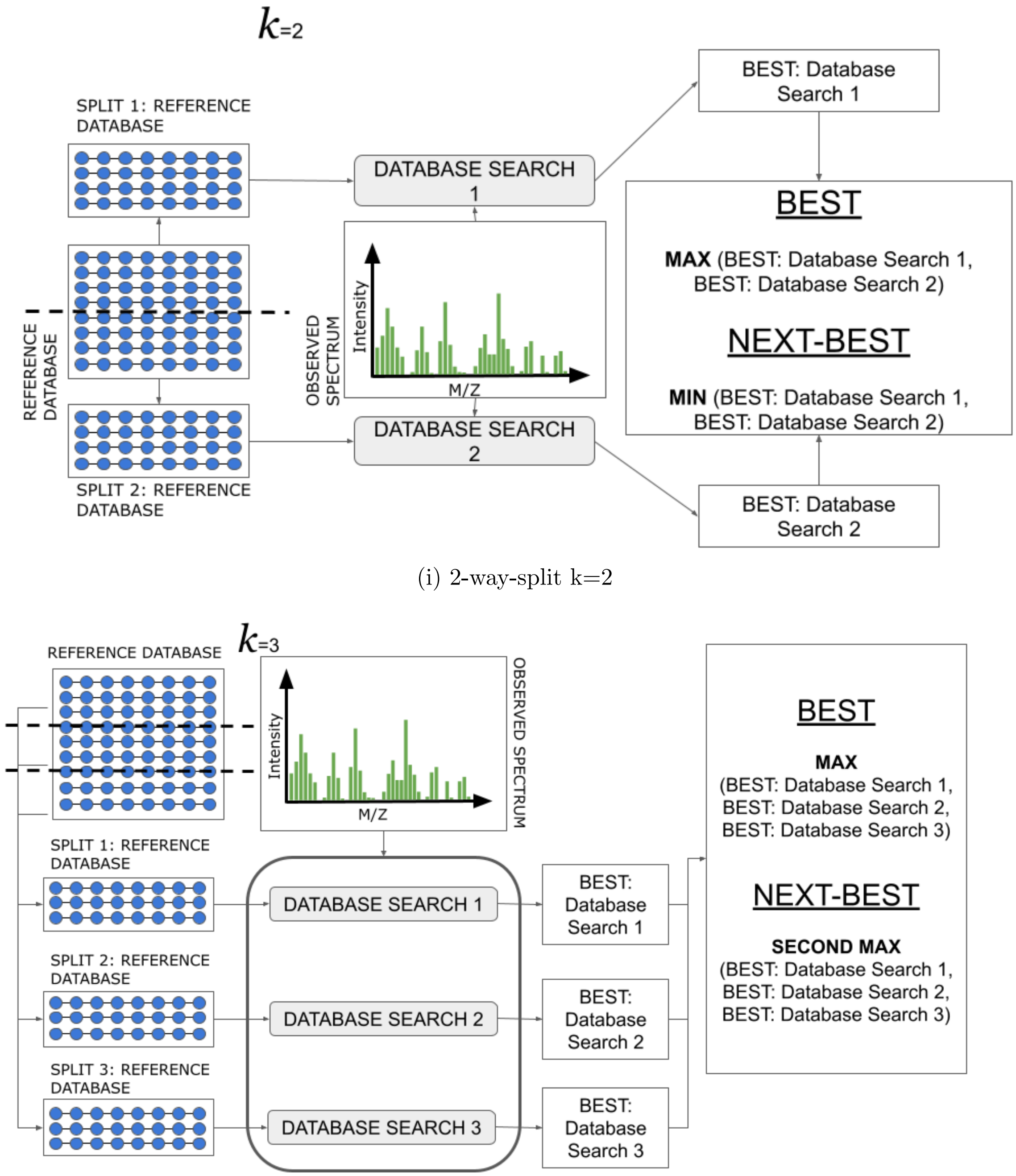
After calculating the similarity between the *best* peptide and the candidate peptides, we retain only one *best* and one *next-best* (one *third-best* when applicable, see Section 4.4.3) for each spectrum, eliminating any duplicates for each rank.

However, it is imperative to note that by excluding similar PSMs, there is a possibility that if no dissimilar PSMs are available for a specific spectrum, this process may result in an empty candidate list for the *next-best*, leading to the absence of *next-best* (or *third-best*) PSMs for that particular spectrum.

4.4.5 Split Database Search

Ensuring an adequate number of *next-best* peptides is essential to obtain a reliable estimation of the false distribution. However, it is worth noting that database search methods may not consistently provide multiple PSMs for all spectra. Some search engines solely report the top-ranked peptide identification. Indeed, even when multiple PSMs for a spectrum are reported, they often exhibit similarities in their characteristics and properties. This similarity in nature can be attributed to various factors, including shared sequences, post-translational modifications, or spectral features. For instance, when analyzing MS-GF+ database search results, we initially identified *next-best* peptides for less than 5% of the spectra. Furthermore, due to the similarity of the peptides, there may be instances where an empty candidate list is generated for the *next-best* peptides, as discussed in Section 4.4.4. Consequently, this leads to an imbalance in the training dataset.

To address this challenge and ensure an adequate supply of *next-best* peptides, we need to implement a strategic approach. We split the database into three parts and utilized


k=2
SPLIT 1: REFERENCE DATABASE
REFERENCE DATABASE
SPLIT 2: REFERENCE DATABASE
DATABASE SEARCH 1
OBSERVED SPECTRUM
Intensity
M/Z
DATABASE SEARCH 2
BEST: Database Search 1
BEST
MAX (BEST: Database Search 1, BEST: Database Search 2)
NEXT-BEST
MIN (BEST: Database Search 1, BEST: Database Search 2)
BEST: Database Search 2
k=3
REFERENCE DATABASE
Intensity
M/Z
OBSERVED SPECTRUM
SPLIT 1: REFERENCE DATABASE
SPLIT 2: REFERENCE DATABASE
SPLIT 3: REFERENCE DATABASE
DATABASE SEARCH 1
DATABASE SEARCH 2
DATABASE SEARCH 3
BEST: Database Search 1
BEST: Database Search 2
BEST: Database Search 3
BEST
MAX
(BEST: Database Search 1, BEST: Database Search 2, BEST: Database Search 3)
NEXT-BEST
SECOND MAX
(BEST: Database Search 1, BEST: Database Search 2, BEST: Database Search 3)

(i) 2-way-split k=2

(ii) 3-way-split k=3

Figure 4.7: Visualization of *best* and *next-best* calculations during split database searches with (i) k=2 and (ii) k=3.

them to perform three separate database searches. Therefore, we have at least one PSM from each database for each spectrum in the combined search result. The merged result is then sorted according to score to report the *best* and the *next-best* (*third-best*) peptides for each spectrum. Furthermore, if we split the database so that the number of shared peptides among the splits is the lowest, we can ensure the highest number of spectra with at least two dissimilar PSMs. This alternative search technique, accomplished through database pre-processing, enables us to identify the *next-best* peptides for the vast majority (over 85%) of the spectra [91]. Figure 4.7 illustrates the process for computing the *best* and *next-best* PSMs for each spectrum in the context of (i) a 2-way-split ($k = 2$) database search and (ii) a 3-way-split ($k = 3$) database search. It is worth noting that in our experiments, we specifically use the scenario where $k = 3$.

When splitting a target database, the process involves randomly dividing the original target database into k mutually exclusive subsets of proteins. However, the procedure for splitting a concatenated target-decoy database differs slightly. Initially, the target database is divided in the conventional manner. Subsequently, when splitting the decoy database, the original decoy database is divided in such a way that each split contains the same set of proteins as the target database.

4.5 FDR Estimation through Machine Learning Methods

In this section, we propose machine learning techniques to estimate the FDR of the identified peptides. Our objective is to achieve a separation comparable to that provided by the established target-decoy method. We leverage the scores from both the *best* and *next-best* peptide identifications as our positive and negative distributions, respectively, to derive a confidence score denoted as $f(x)$. This confidence score serves as a central component in our FDR estimation process.

4.5.1 Estimation of $f(x)$

We employ a machine learning approach to estimate the probability $f(x)$ for a spectrum with a score x. This probability indicates the likelihood that the spectrum is correct or belongs to the group A. We compute $f(x)$ as follows:

- Input: $X = x_1, x_2, \cdots, x_n$ for $i = 1, 2, \cdots, n$, where x_i is the score of a spectra S_i, and n is the total number of spectra.

- Output: $f(x)$, the probability that a spectra S with score x is correct.

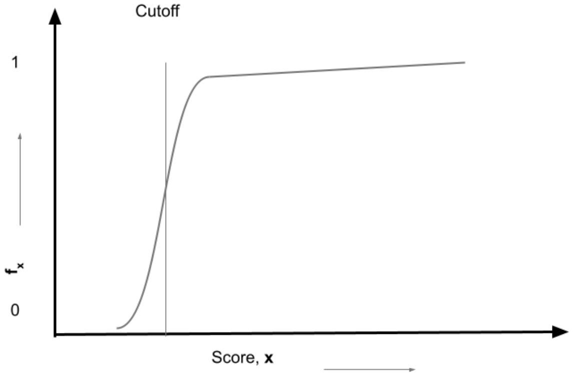


Figure 4.8: Expected $f(x)$

Subsequently, with the current estimate of $f(x)$, we can iteratively refine the sets A and B, continuously updating the calculation of $f(x)$ until convergence is reached. Figure 4.8 portrays our anticipated result, where the function displays a sharp incline with the score, eventually approaching a value close to 1 beyond the cutoff. This phenomenon means that any PSM with a score exceeding the cutoff is highly likely to be correct, approaching a probability close to 1. To compute $f(x)$, we have employed various methods including Kernel Density Estimation (KDE) and logistic regression. We will explore these methods in more detail in the following discussion.

4.5.2 Kernel Density Estimation (KDE) Based Estimation

Kernel density estimation is a non-parametric method that employs a kernel function to estimate the unknown probability density function of a given finite set of observations. Non-parametric methods are utilized when it is challenging to make specific assumptions about the data distribution. Unlike parametric density estimation, where parameters are assumed to fit a standard probability distribution, non-parametric methods do not rely on such assumptions.

In cases where it is not feasible to make explicit assumptions about the data, non-parametric algorithms come into play. These algorithms are applied to approximate the probability distribution of the data without assuming a predefined distribution shape or parameters. Essentially, non-parametric algorithms work without requiring prior knowledge of the data characteristics and aim to deduce distribution based on the available finite dataset.

To find the KDE, the kernel function is first generated at every data point. The distribution is then estimated using the sum of the value derived from the kernel function at each data point u. Kernel, $K(u)$ is a function that satisfies the following three properties.

1. The function must be symmetrical, i.e., $K(-u) = K(+u)$.
2. The area under the curve of the function must be equal to one. $\int_{-\infty}^{\infty} K(u)du = 1$.
3. The value of the kernel function can not be negative. $K(u) \geq 0$, for all $-\infty < u < \infty$.

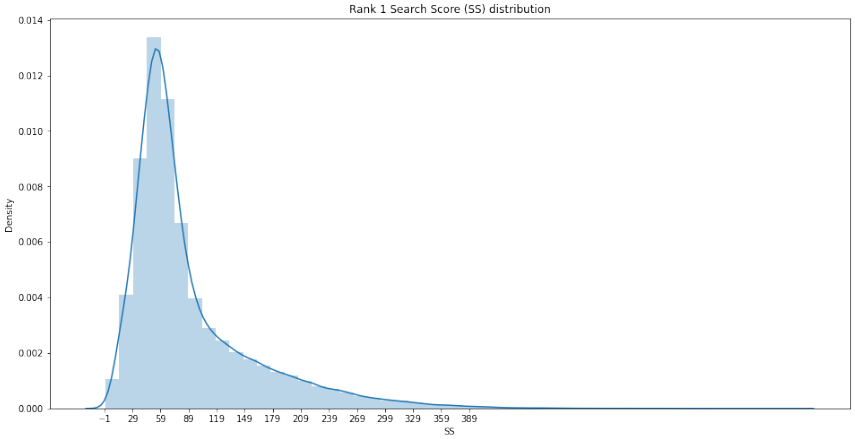


Figure 4.9: Score distribution of *best* PSMS (rank-one) using both KDE and Histogram

The bandwidth parameter, a crucial aspect of kernel density estimation, significantly influences the accuracy of fitting the data. Regulating the bandwidth alters the shape of the kernel, and choosing an appropriate bandwidth is essential for obtaining accurate results. A smaller bandwidth leads to a narrow kernel function that captures many details in the density estimation, potentially including high variances. Conversely, a larger bandwidth introduces bias and smoothes out fine details in the estimation.

To visualize the distribution of a data sample, a simple method is to use a histogram. In Figure 4.9, we provide a combined representation of the histogram and kernel density estimation (KDE) for rank-one (*best*) peptides. This visual representation illustrates the effectiveness of kernel distribution estimation as a robust approach for modeling the underlying distribution in our analysis.

In our approach, we employ two distinct models, namely $model_A$ and $model_B$, to perform kernel distribution estimation for two critical components: true spectra A, denoted as $P(A)$, and false spectra B, denoted as $P(B)$. Subsequently, we calculate the confidence score $f(x)$ by averaging the probabilities obtained from these models, which translates to averaging $P(A)$ and $1-P(B)$. This process allows us to assess the likelihood of correctness for each spectrum, thereby facilitating the estimation of the FDR.

4.5.3 Logistic Regression Based Estimation

Logistic regression is a supervised learning algorithm, i.e., the training data is labeled. The algorithm models the relationship between the dependent variable and one or more independent variables by estimating the probabilities using a logistic regression equation. This type of analysis helps us predict the likelihood of an event. The sigmoid function is used in order to map predicted values to probabilities. Therefore, it assigns a value between 0 to 1 to any real value.

In our research, we aim to predict whether a spectrum is true, or if it belongs to the set A as defined in Section 4.4.1. We estimate $f(x)$, which represents the probability that a spectrum with a score x is correct, specifically, the probability that a spectrum containing a *best* (rank-one) peptide with a score x belongs to the set of true spectra. Upon examining Figure 4.8, it becomes evident that we require $f(x)$ to approach 1 when the score is higher than the cutoff score (to be determined from the FDR of the search results) and closer to 0 when the score is lower. Therefore, logistic regression is a good fit for our problem.

In the logistic regression step, we label A_1 as 1, and the combined set of $(A_2 + B_1)$ as 0. For each spectrum with the *best* score x, we calculate the probability of it belonging to class 1. This process yields an estimation of A, and subsequently, we can derive the values of A_1, A_2, B_1, and B_2. In the subsequent iteration, we utilize this updated information to recalculate the confidence score $f(x)$, and this iterative process continues until convergence is achieved.

Furthermore, adding polynomial-order terms increases the capacity of the logistic regression model, which allows the model to learn complex decision boundaries that are otherwise impossible using linear regression. We have integrated various combinations of polynomial terms, including x^2,x^3, $x^{\frac{1}{2}}$, and $x^{\frac{1}{3}}$, in addition to the original score x.

4.5.4 FDR Estimation from $f(x)$

Initially, we calculate the probability $f(x)$ for a spectrum with a score x, which represents the likelihood of the spectrum being correct. Subsequently, we use $f(x)$ to derive the FDR estimation through the following computational steps.

1. The spectra are sorted according to the Search Score, x, where higher values of x indicate better scores.
2. Estimated FDR from $f(x)$,
$$V = \frac{\sum(1 - f(x))}{\#peptide\ with\ score\ \geq x} \tag{4.1}$$
3. 1% FDR approximate from $f(x)$: all results where $V \leq 0.01$.

To assess the performance of our proposed method, we must benchmark it against the standard target-decoy approach. To facilitate this comparison, we computed the default FDR using the following steps for this part of our study in order to compare it with our estimated FDR derived from $f(x)$:

1. When FP represents the number of false positives and TP represents the number of true positives, the default false discover rate is calculated as:
$$FDR = \frac{FP}{(TP + FP)}$$
2. FP is calculated using the following equation:
$$FP = 2 \times \#decoy \tag{4.2}$$
3. Therefore, the default false discovery rate is given by:
$$FDR = \frac{2 \times \#decoy}{(\#target + \#decoy)} \tag{4.3}$$

The reason this default FDR calculation differs from the standard calculation presented in Equation 2.1 is that the estimated FDR calculation (using $f(x)$) in Equation 4.1 includes the total false distribution, encompassing both false positives and false negatives, in contrast to the default FDR calculation paradigm. In the standard target-decoy approach, the total false PSMs are calculated using Equation 4.2, and we utilize this approach to ensure a fair comparison between the performance of the estimated FDR using our algorithm and the default target-decoy method.

4.5.5 Validation of Hypotheses Using Synthetic In-Silico PSMs

In this section, we conducted preliminary testing using synthetic PSMs to validate our method's potential. The lack of ground truth in proteomics presents a significant validation challenge. Nevertheless, synthetic sets of true and false PSMs can serve as proxy ground truth for initial hypothesis validation in our method.

We explore a hypothetical scenario where two sets of synthetic spectra are generated, each associated with two scores: a rank-one and a rank-two score. One set has a higher mean score, designating it as the true spectra, while the other serves as a collection of random spectra. Combining these sets, we create rank-one and rank-two score distributions for all spectra, both true and random. Our method aims to differentiate between hypothetical true and false spectra using only the distributions of rank-one and rank-two scores, without prior knowledge of the spectra labels. Demonstrating this separation in a theoretical scenario would serve as validation for our hypothesis, showcasing the effectiveness of our method under ideal conditions.

4.6 Improvement of Logistic Regression Approach

When our FDR estimation performs as anticipated, and our method can achieve the same level of separation as the target-decoy method, we expect the estimated A_1 and the expected correct (calculated using the traditional target-decoy method) distributions to overlap to the greatest extent possible.

Our algorithm exhibits superior performance with logistic regression compared to KDE, as demonstrated in the results presented in Section 4.7. However, there remains a gap in the alignment between the expected correct values and the estimated A_1, particularly for lower scores. To address this by introducing a non-linear component to logistic regression for increased complexity, we propose the incorporation of non-linear features that are polynomials of the score x. Additionally, to compensate for the underestimated false distribution in comparison to decoys, we suggest augmenting the *next-best* data, which serves as our false samples for estimating the separation. The details of these proposed enhancements will be discussed in the following sections.

4.6.1 Integration of Non-linear Polynomial Features

The motivation for adding polynomial features and their interactions into the mix is to increase the model's capacity. Including polynomial terms allows us to learn decision boundaries that we would otherwise be unable to learn simply using the original features. This is because a linear decision boundary (which is what logistic regression fits) learned on nonlinear transformations of features will ultimately be nonlinear in terms of the original features. Even if the polynomial terms prove to be of limited value, the model still possesses the ability to learn a decision boundary that remains linear in the original features by simply disregarding the polynomial terms. This flexibility allows the model to adapt and find the most suitable representation for the data, ensuring robustness in its decision-making process. We have incorporated different combinations of x^2,x^3, $x^{\frac{1}{2}}$, and $x^{\frac{1}{3}}$ along with x. The results are presented in Figures 4.13, 4.14, and 4.15.

4.6.2 Augmentation

In practice, the set of *best* PSMs can also contain random matches, complicating our goal of estimating the truly random PSMs, i.e., the spectra that belong to B. To approximate this, first, we will introduce the concept of first-order incorrect PSMs, denoted as A_2 and B_1. These are considered first-order incorrect PSMs because, for each spectrum, they are the highest scoring among the incorrect PSMs. Similarly, we define B_2 as second-order incorrect PSMs, as these are the second-highest scoring incorrect PSMs for each spectrum belonging to B.

Initially, we do not have knowledge about the separation between A and B. Therefore, we model A using the *best* PSMs, which includes some random matches, and we model B using the *next-best* PSMs, which incorporates corresponding second-order incorrect PSMs. Consequently, we observe a shift between our estimated B_1 and decoy (known false).

However, it is worth noting that for higher scores, the B_1 decoy-only distribution tends to be underestimated, as illustrated in Figure 4.10. This implies that some decoys are erroneously considered highly likely to be targets or included in A_1. This underscores the importance of refining our methods to achieve more accurate estimates.

In this plot (Figure 4.10), we exclusively display the decoys from B_1 to facilitate a comparison with the decoys obtained through the default target-decoy method. It is important to note that B_1 represents the distribution of random false matches, and when applied to target-decoy data, it encompasses both the random matches from the target and the true decoys. An alternative visualization approach would involve doubling the distribution of decoys obtained from the regular target-decoy method while retaining both the target and decoys from B_1.

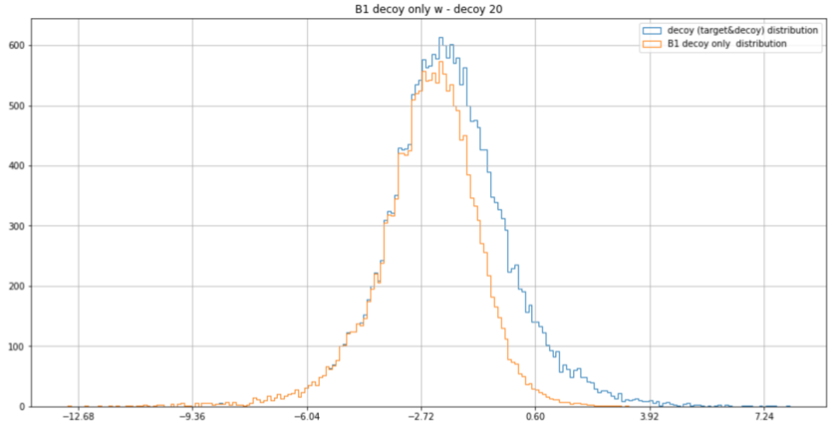


Figure 4.10: Underestimation of decoy only B_1 distribution for higher score compared to decoy distribution.

To further refine this approximation, we examine the differences between the *third-best* (or rank-three) and *next-best* PSMs, , aiming to account for shifts caused by the omission of first-order incorrect PSMs in the heuristics. We utilize this information to transform the *next-best* PSMs, enabling them to provide a more accurate representation of the random spectra. The *next-best* data undergoes augmentation with the inclusion of R_2'. This set denoted as R_2', comprises c samples that are randomly selected from the original *next-best* set. The scores assigned to these samples in R_2' are determined as $x' = ax + b$, where x represents the score for the *next-best* PSM, and a and b are parameters used for score transformation.

Optimization

Indeed, a critical consideration in our augmentation approach is the selection of optimal values for the augmentation parameters a, b, and c. The effectiveness of the augmentation approach significantly relies on these parameter values. To identify the optimal values for the parameters a, b, and c, we employ the basin-hopping optimization technique [125]. This optimization methodology enables us to fine-tune these parameters to achieve the best possible representation of the *next-best* PSMs in our FDR estimation process. The proposed method involves the following approach:

Parameter Ranges: Initially, specific ranges are defined for each of the parameters. The ranges for a and b are determined by analyzing the score distribution of both the *best* and *next-best* sets. The range for c is derived from the sizes of the sets *best*, *next-best*, and *third-best*.

Basin-Hopping Optimization: Basin-hopping optimization is applied to search for the optimal values of a, b, and c. This optimization method aims to minimize the mean-square-error (mse) between the transformed data (i.e., augmented *third-best*) and the target data (*next-best*).

Cost Function: A cost function is defined based on the binned difference between the transformed augmented *third-best* data (using the current a, b, and c) and the *next-best* data. The goal is to find parameter values that minimize this cost function, ensuring a close alignment between the augmented data and the target data.

Parameter Optimization: The basin-hopping optimization iteratively explores parameter combinations within their defined ranges to minimize the cost function. This process continues until an optimal (to some degree) set of values for a, b, and c is discovered. The "temperature" parameter is adjusted through intuition and manual tuning.

Utilization of Optimized Parameters: After identifying the optimal values for a, b, and c, these parameters transform the augmented *next-best* data, as outlined above. This optimized augmentation procedure enhances the ability of the *next-best* data to closely approximate the false distribution, aligning it more effectively with the decoy distribution from the target-decoy database search.

The process of augmenting the data is indeed intriguing. When the parameters are optimized, this method enables us to achieve a substantial overlap between the 'expected correct' distribution from the target-decoy database search and our estimated true distribution (as presented in Section 4.7.2). However, it is crucial to acknowledge that the optimization process is time-intensive, rendering it impractical for every new dataset. Consequently, there is a pressing need to explore and develop a more generalized approach that can be readily applied across diverse datasets without extensive parameter optimization. This pursuit of a more universal solution holds the potential to streamline and enhance the applicability of our methodology.

4.7 Results and Discussion

In this section we will first introduce the dataset and experimental setup for the results presented next. Then we will present the results of KDE, logistic regression, non-linear feature addition, and augmentation one by one. Additionally, we will present the results from synthetic data signifying our method's effectiveness in theory or ideal data.

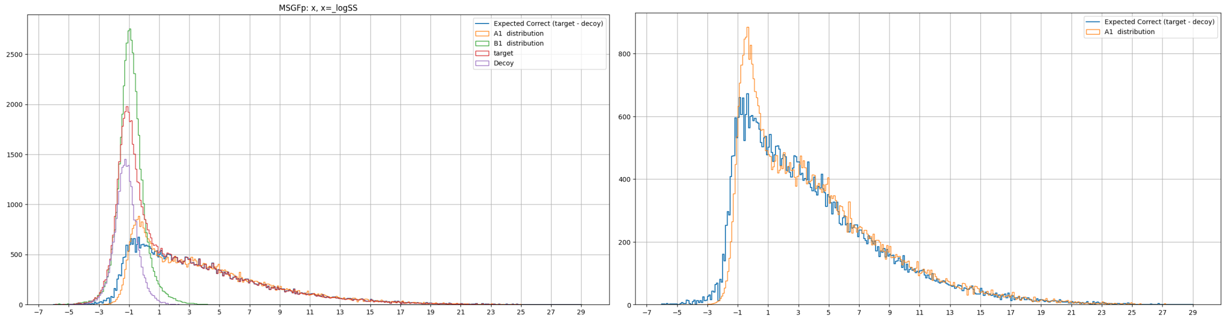
4.7.1 Dataset

In our experimental setup, we exclusively utilized the initial replicate of the Human HeLa dataset (ProteomeXchange project ID: PXD015028) as detailed in the study by Moosa et al.[90], unless specified otherwise. Particularly, we made adjustments during the split database search phase, primarily to accommodate changes in the database, where different searches were performed with different splits. Additionally, to validate our initial hypothesis, we conducted experiments using synthetic data.

It is important to note that our algorithm is ideally designed for target-only search results. To generate an adequate number of *next-best* PSMs, we perform a 3-way split database search using only the target database, excluding the decoys. However, for the purpose of comparing our results with traditional target-decoy methods and validating the effectiveness of our approach, we employed a target-decoy 3-way split database strategy. Therefore, for our experimental results, we conducted a 3-way split target-decoy database search, as detailed in Section 4.4.5, using the HeLa replicate one dataset while maintaining the same search parameters as specified in Moosa et al.'s manuscript [90].

4.7.2 Experimental Results

In this section, we delve into the results, exploring the outcomes of our extensive efforts in FDR estimation for peptide identification. We have explored various methodologies, including Kernel Density Estimation (KDE), logistic regression, the incorporation of polynomial terms in logistic regression, and the augmentation of *next-best* samples. Through



(i) Expected Correct, A_1, B_1, target, and decoy

(ii) Expected Correct and A_1

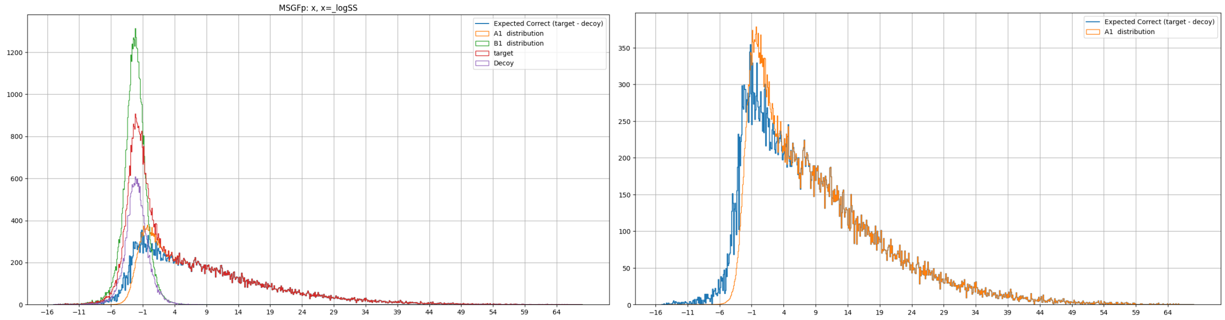
Figure 4.11: Distribution of estimated A_1, including (i) the distribution involving estimated B_1, targets and decoys and (ii) the Expected Correct distribution (target − decoy) estimated with KDE. Significant overlap is observed for higher scores in expected correct distributions.

a rigorous analysis of these techniques, our objective is to shed light on their effectiveness and uncover any improvements achieved.

In our approach, we designate A_1 as the positive distribution and $(B_1 + A_2)$ as the negative distribution. Using these distributions, we assign a probability of correctness to each spectrum. With this probability information, we iteratively calculate the distribution for A_1, B_1, A_2, and B_2. This process is repeated until convergence is achieved. Ultimately, from the probabilities assigned to each spectrum that indicates the correctness, we derive the final distribution of A_1, which represents the distribution of correctly identified spectra. Ultimately, this allows us to estimate the FDR.

When applying KDE, the expected false distribution estimation diverges from our expectations, as illustrated in Figure 4.11. This discrepancy becomes apparent when comparing the expected correct values derived from the target-decoy method with the estimation of A_1, as demonstrated in Figure 4.11 (ii). Note that the expected correct distribution is obtained by subtracting the decoy distribution from the target distribution.

Despite logistic regression outperforming KDE, it still falls short of our expectations. The differences between A_1 and the expected correct distribution are still evident, as observed in Figure 4.12. To further improve the performance, we propose two strategies. Firstly, the integration of non-linear polynomial features, and secondly, augmenting the *next-best* distribution to achieve a more accurate representation of the false distribution as discussed in Section 4.6. It is worth mentioning that we have used the "liblinear" solver for logistic regression.



(i) Expected Correct, A_1, B_1, target, and decoy

(ii) Expected Correct and A_1

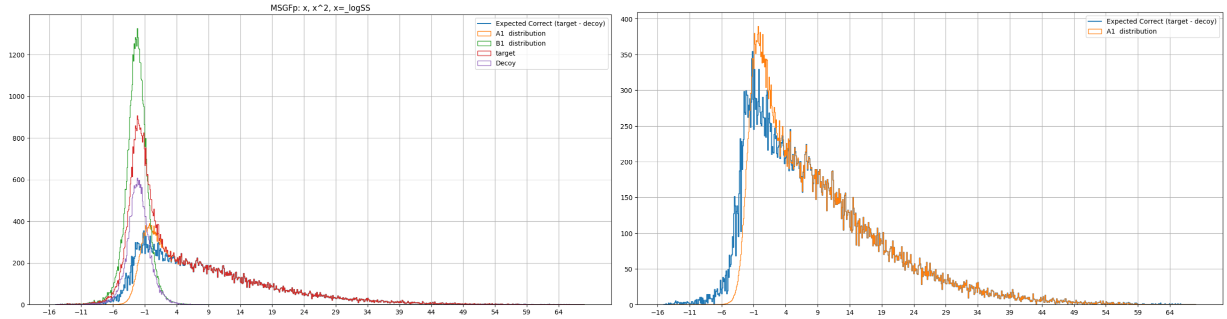
Figure 4.12: Distribution of estimated A_1 and B_1, including (i) the distribution involving targets and decoys and (ii) the Expected Correct distribution (target − decoy) estimated with logistic regression with feature x, augmentation not applied. Significant overlap is observed for higher scores in expected correct distributions.

Polynomial features in logistic regression

As observed in Figure 4.14, the inclusion of polynomial terms helps mitigate the disparity between the expected correct values and the estimated A_1 more effectively. However, it remains short of our desired accuracy. While the addition of x^2 or $x^{\frac{1}{2}}$ or $x^{\frac{1}{2}}$ (4.13, 4.16, 4.15) does not exhibit any noticeable differences compared to only x (Figures 4.12), further comparison between Figure 4.13 (ii) and Figure 4.14 (ii) sheds light on the significance of the additional term x^3. The latter figure demonstrates that the inclusion of this term contributes to a more pronounced reduction in the observed discrepancy. Consequently, we showcase the results obtained when only the x^3 term is added as a polynomial component, as illustrated in Figure 4.17. In particular, this configuration exhibits a slight improvement compared to Figure 4.14, where both x^2 and x^3 are incorporated as polynomial terms. However, this nuanced performance comparison does not yield specific insights or reasons for the observed differences. Further investigation is needed to elucidate the underlying factors that contribute to the varying performance between the inclusion of different non-linear terms in our logistic regression based FDR estimation approach.

Augmentation results including optimization

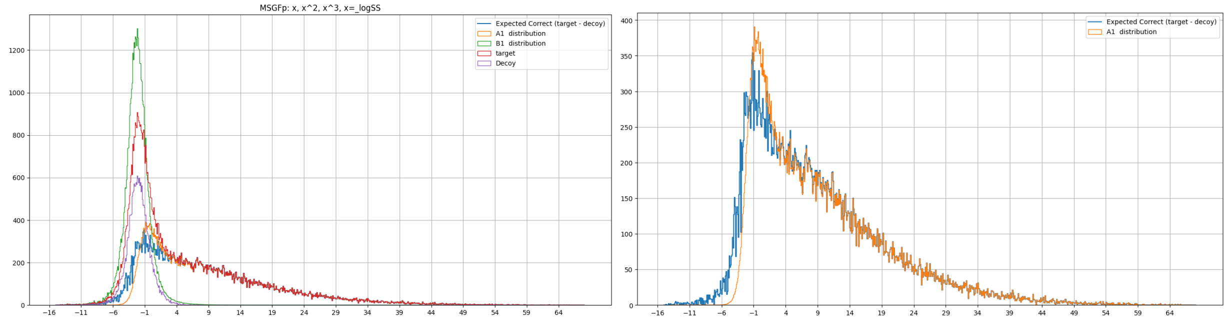
Our additional effort to improve the performance of logistic regression involves augmenting the *next-best* data, addressing the discrepancy arising from second-order incorrect PSMs, as elaborated in Section 4.6.2. In this section, we present the corresponding results of



(i) Expected Correct, A_1, B_1, target, and decoy

(ii) Expected Correct and A_1

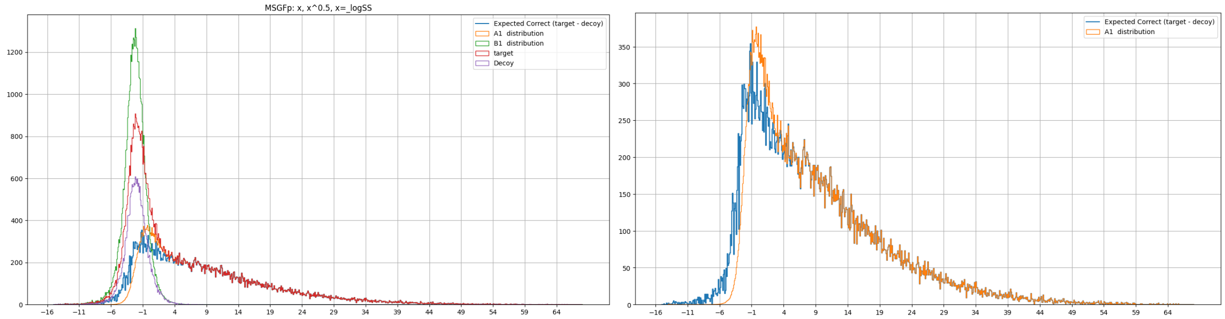
Figure 4.13: Distribution of estimated A_1 and B_1, including (i) the distribution involving targets and decoys and (ii) the Expected Correct distribution (target − decoy) estimated with logistic regression with features x and x^2. Significant overlap is observed for higher scores in expected correct distributions.



(i) Expected Correct, A_1, B_1, target, and decoy

(ii) Expected Correct and A_1

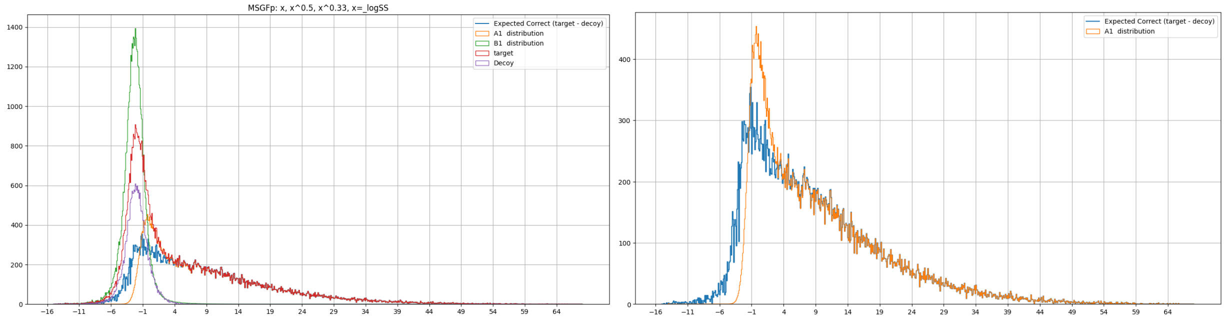
Figure 4.14: Distribution of estimated A_1, including (i) the distribution involving estimated B_1, targets and decoys and (ii) the Expected Correct distribution (target − decoy) estimated with logistic regression with features x, x^2, and x^3. Significant overlap is observed for higher scores in expected correct distributions.



(i) Expected Correct, A_1, B_1, target, and decoy

(ii) Expected Correct and A_1

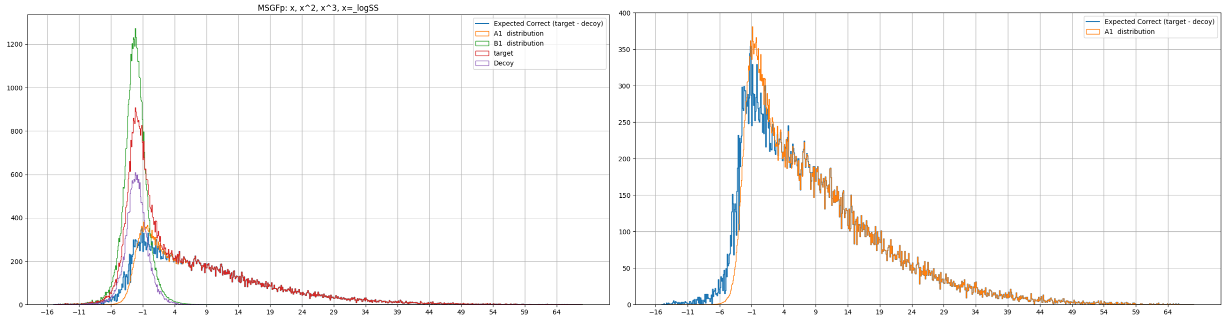
Figure 4.15: Distribution of estimated A_1, including (i) the distribution involving estimated B_1, targets and decoys and (ii) the Expected Correct distribution (target − decoy) estimated with logistic regression with features x and $x^{\frac{1}{2}}$. Significant overlap is observed for higher scores in expected correct distributions.



(i) Expected Correct, A_1, B_1, target, and decoy

(ii) Expected Correct and A_1

Figure 4.16: Distribution of estimated A_1, including (i) the distribution involving estimated B_1, targets and decoys and (ii) the Expected Correct distribution (target − decoy) estimated with logistic regression with features x, $x^{\frac{1}{2}}$, and $x^{\frac{1}{3}}$. Significant overlap is observed for higher scores in expected correct distributions.



(i) Expected Correct, A_1, B_1, target, and decoy

(ii) Expected Correct and A_1

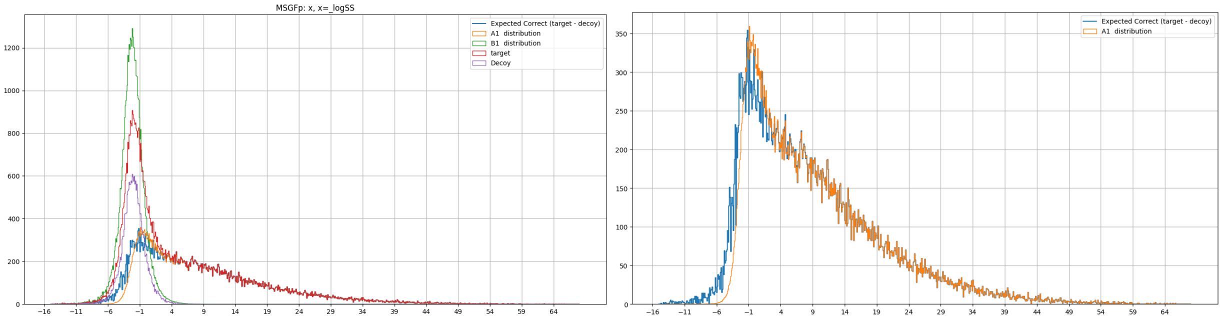
Figure 4.17: Distribution of estimated A_1, including (i) the distribution involving estimated B_1, targets and decoys and (ii) the Expected Correct distribution (target $-$ decoy) estimated with logistic regression with features x, x^3. Significant overlap is observed in expected correct distributions.

this augmentation process. Through basin-hopping optimization, we have determined the values of 1.0506, 1.7971, and 0.18085 for the parameters a, b, and c, respectively. Although the distribution of A_1 and the expected correct values do not achieve complete overlap, we are able to achieve a highly accurate approximation following augmentation as exhibited in Figure 4.18. It is worth mentioning that during the augmentation process, no other modifications are made to the regular logistic regression method.

Hypothesis Validation Results Using Synthetic Data

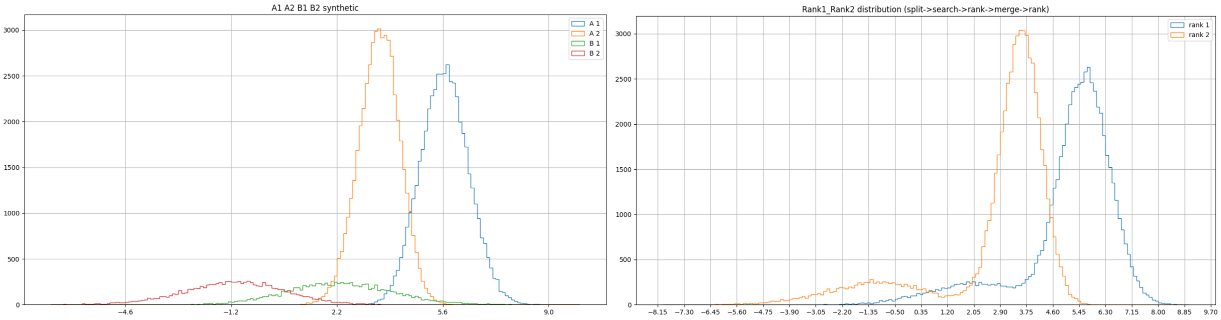
To evaluate the effectiveness of our method under ideal conditions, we have generated four sets of synthetic data representing A_1, B_1, A_2, and B_2. The distribution of this synthetic data is visually depicted in Figure 4.19. It is important to note that our algorithm can easily calculate the rank-one and rank-two for each hypothetical spectrum by simply comparing the designated scores, which is similar to the *best* and *next-best* in an experimental spectrum. However, the labels A and B remain concealed from the algorithm. Our objective is to assess whether our algorithm can generate a distribution of A and B from the rank-one and rank-two information that closely aligns with the ground truth of the synthetic data. Achieving this alignment in a theoretical scenario would demonstrate the efficacy of our method under ideal conditions, thus validating our hypothesis.

The results obtained using synthetic data, when applying our regular logistic regression without any augmentation or non-linear features are displayed in Figure 4.20. In the case



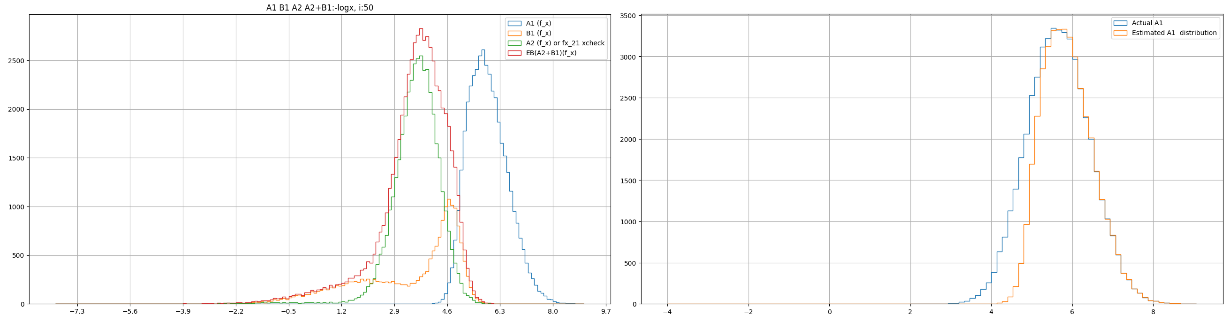
(i) Expected Correct, A_1, B_1, target, and decoy (ii) Expected Correct and A_1

Figure 4.18: Distribution of estimated A_1, including (i) the distribution involving estimated B_1, targets and decoys and (ii) the Expected Correct distribution (target − decoy) estimated with logistic regression with **augmentation** of *next-best*. Significant overlap is observed in expected correct distributions.



(i) A1B1A2B2 (ii) R1R2

Figure 4.19: In-sillico synthetic distribution of (i) spectra A, and B with rank-one (A_1, B_1), and rank two (A_2, B_2) scores and (ii) combined rank-one, and rank-two scores.



(i) Estimated A_1, B_1, A_2, B_2

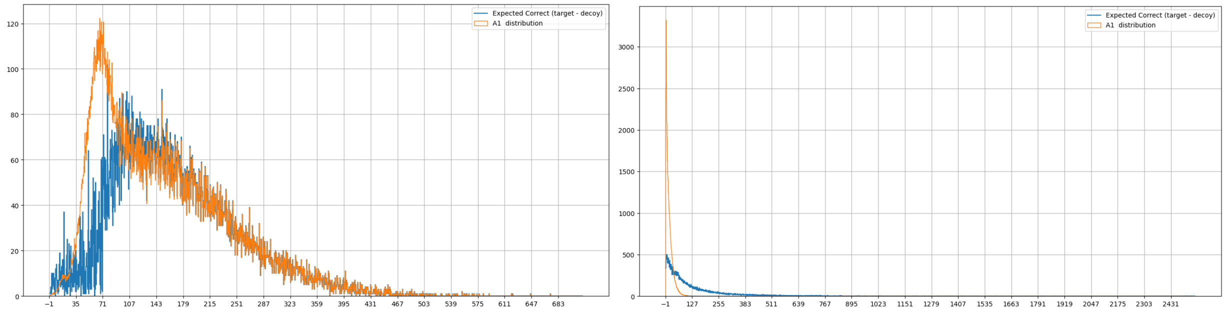
(ii) Actual distribution of A vs. Estimated A_1

Figure 4.20: (i) Estimated A_1, B_1, A_2, B_2 from the synthetic distribution (ii) Actual distribution of A and Estimated A_1 without the knowledge of A, and B labels. Significant overlap is observed between the estimated correct distribution and the true distribution, especially for higher scores.

of theoretical data, our method clearly distinguishes between A and B. The estimation of true and false distributions was carried out without prior knowledge of the actual A and B distributions. Furthermore, as illustrated in Figure 4.20 (ii), it is evident that for higher scores, the estimated A_1 distribution aligns almost perfectly with the actual distribution of A. While these results form the basis of our hypothesis, it is important to note that they do not guarantee that the method will perform similarly in real proteomics analysis data. Indeed, to comprehensively assess the method's effectiveness in practical applications, we have conducted further investigation and validation using a real-world proteomics dataset. When applied to experimental proteomics data, the method demonstrated promising results. Nevertheless, as discussed earlier, there is room for further improvement.

4.7.3 Performance Across Different Search Engines

Building upon the promising results obtained for MS-GF+ [75] search results, we endeavored to assess the applicability of our approach to other search engines, namely MS Amanda [34] and Comet [42]. However, a noteworthy observation is that optimized augmentation is highly dataset-specific, and augmentation tailored for one dataset does not readily generalize to a different dataset. As demonstrated in Figure 4.21, this dataset-specific optimization challenge has resulted in suboptimal performance, with a noticeable gap in the results. Moreover, it is worth highlighting that the optimization process is time-consuming, making it impractical to perform for every dataset, especially when the



(i) MS Amanda

(ii) Comet

Figure 4.21: (i) Estimated A_1 vs. Expected Correct distribution for different search engines using optimized a, b, and c as obtained in Section 4.7.2 for augmentation. Estimated correct differs from the expected correct distribution.

improvements in results are marginal.

4.8 Conclusion

In conclusion, this chapter has presented an initial exploration of alternative false discovery rate estimation in the field of proteomics. We utilize machine learning-based algorithms trained on the *best* and *next-best* PSMs, with the goal of closely mimicking the separation observed in target and decoy distributions. Our results, particularly when applied to synthetic data and experimental datasets, have unveiled exciting and promising insights into the potential of our FDR estimation methodology. Notably, our best-performing algorithm, which involves logistic regression and data augmentation, has demonstrated the capacity to distinguish between true and false identifications. Furthermore, the inclusion of polynomial term features has shown promising results.

Nevertheless, it is essential to recognize that although these results are promising, there is still a requirement for further enhancements and comprehensive analysis to achieve the desired outcome that can be applied universally across various types of datasets. The complexities and nuances of experimental proteomics data call for refinement of our methodology. Possible avenues for improvement include refining the augmentation process to bridge any remaining gaps, improving the estimation of f_x, use of different machine learning methods to estimate f_x, and exploring alternative equations or methods to estimate FDR using f_x. In essence, this chapter sets the stage for ongoing research and development in the pursuit of more accurate and reliable FDR estimation techniques in peptide identification.

Chapter 5

Improving Peptide Identification Rate by Machine Learning with Next-Ranked Peptide Spectrum Matches [91]

To improve the peptide identification rates in the database search analysis of bottom-up proteomics data, many proposed implementation of machine learning algorithms. These machine learning-based methods train a new scoring function after the initial search to re-score and re-rank the peptide spectrum matches (PSMs). Generally, the retraining uses selected peptide-spectrum matches from the target and decoy databases as positive and negative training examples, respectively. However, this exposes the target-decoy information to the scoring function, potentially invalidating the false discovery rate (FDR) estimation. We propose a novel method for retraining without revealing the target-decoy information. Our approach considers the top-ranked and the next-ranked peptides for the same spectrum as positive and negative examples, respectively. We demonstrate that this leads to a much-improved identification rate while maintaining an accurate FDR estimation.

5.1 Introduction

Traditionally in bottom-up proteomics, peptide identification involves solving two subproblems: 1) defining a peptide spectrum match (PSM) scoring function; and 2) selecting a subset of top-scoring PSMs that are statistically significant. [127] We aim to define a re-scoring function that will rank the PSMs more accurately to achieve a better set of top-scoring PSMs. Because of noises and instrumentation errors, some spectra are random rather than originating from the proteins in the sample. These random spectra result in random PSMs. Thus, peptide identification search results contain an aggregation of two trends, the true and the false positives. The PSMs where the spectra matched to their source peptides are correct, whereas those matched with peptides they did not originate from are incorrect. The next step is to separate these trends to distinguish true PSMs from false ones. Although the ground truth is unknown, controlling the false discovery rate warrants our confidence in the output PSMs. The target-decoy search [39] is a mainstream strategy to regulate the false discovery rate (FDR). In this method, a decoy database is incorporated with the original target database to approximate the FDR.

One way to improve the peptide identification rate without modifying any current steps is to re-score and re-order the PSMs, [25] which can be applied as a post-processor to the database search results. PeptideProphet, [74, 18] a semi-supervised learning approach for peptide identification, distinguishes between the correct and incorrect peptides based on various parameters such as search score, spectra quality, number of missed cleavages, and peptide sequence length. The algorithm optimizes negative distribution estimation using decoys but does not require them. However, training data with known peptide assignments is required. Shteynberg et al. [113] developed iProphet, which integrates outcomes derived from multiple database search engines applied to the same data to improve identification rates and error estimation. Another commonly used post-processor, Percolator, [70, 119] assigns statistical confidence measures to the PSMs using a semi-supervised machine learning algorithm. However, it uses decoy PSMs as negative samples during training. MSBooster [130] incorporates features additional to Percolator to re-score the PSMs. These additional features are derived from deep learning-based predictions. Similarly, inSPIRE [25] utilizes spectral prediction and retention time derived from Prosit [51] to re-score the PSMs. Additionally, Halloran et al. [60] introduced a deep semi-supervised learning method that significantly improves peptide identification accuracy by harnessing the powerful learning capabilities of deep neural network models.

These methods that learn a new scoring function by using the target and decoy PSMs as positive and negative examples may compromise the FDR calculation. This is because the learning algorithm may learn to treat the decoy sequences and the false target sequences differently by sequence patterns specific to the target sequence database or the decoy generation algorithm. As a result, the new score distributions of the decoy and false target PSMs may become different, creating bias in the FDR estimation.

Here, we aim to develop a new way to train the new scoring function without exposing the target-decoy information to the learning algorithm. During the database search for peptide identification, a spectrum is compared with many peptides in the database. Typically, only the top-ranked peptide is reported for each spectrum. The other peptides that are dissimilar to the top-ranked peptide and have lower scores at the same time are most likely false matches. This is intriguing, as it generates a catalog of nearly confirmed negative examples without needing to disclose the decoy information. Therefore, we propose to train the new scoring function by using the top-ranked PSMs as positive examples and the next-ranked PSMs (refer to Section 5.2.2 for details) as negative examples. Since the learning algorithm has no access to the target/decoy information, and therefore will not create the bias in the FDR estimation.

After the manuscript's submission, we were made aware a related method called Nokoi [52], which also explores the concept of using lower-ranked PSMs as negative samples for training a classification model. However, there are significant differences between Nokoi and our method. Nokoi does not consider peptide similarity when selecting negative samples, and it uses a classification model pre-trained on a previously generated fixed set of inhouse data. In contrast, our approach trains a new scoring function specifically tailored to each dataset, allowing for customization and adaptability to different datasets, species, and instruments.

5.2 Data and Methods

While database search engines can report multiple PSMs for one spectrum, these PSMs generally appear from similar peptides. However, there are two exciting possibilities when the peptides are dissimilar. First, the peptides are potentially co-eluting. Multiple peptide precursors are often co-eluted simultaneously in the same tandem mass (MS/MS) spectrum. [127] For example, two different and dissimilar peptides with the same m/z co-eluting in the same RT window. In this case, both peptides are true PSMs. If not coeluting, one spectrum can only match with one single peptide. So the additional PSMs are just random matches. In conventional practice, only the top-ranked PSMs are reported and analyzed for further investigations and utilized for FDR control. The remaining peptides for each spectrum are seldom correct and are disregarded by search engines. Nevertheless, we acknowledge that these additional peptides present a valuable opportunity to learn a new score function for the PSMs.

We propose a method to train a new scoring function using machine learning algorithms to leverage the retained information from the discarded peptide assignments. This enables us to obtain sufficient negative samples to train the scoring function and improve performance. We first sort the PSMs according to their scores, with the top-ranked PSM being the highest-scoring PSM for each spectrum. Among the remaining PSMs of that spectrum, we identify the highest-scoring PSM with a dissimilar peptide sequence to that of the top-ranked PSM as the next-ranked PSM (details in Section 5.2.2). While top-ranked PSMs can include both true and false positives, all next-ranked PSMs are almost always false, which is particularly evident for high-quality spectra. We hypothesize that integrating the next-ranked PSMs to re-score the PSMs can improve peptide identification rates. To train our new scoring function, we consider top-ranked PSMs as positive samples and next-ranked PSMs as negative examples. It is worth noting that this selection does not differentiate between target and decoy peptides. Machine learning techniques based on logistic regression are employed to learn the updated scoring function, which is then utilized to re-score the peptides. Only the top-scoring peptide is reported for each spectrum, and the target-decoy information is checked at the end to estimate FDR, following standard practice.

5.2.1 Split Database Search

To train our scoring function, we need a sufficient number of next-ranked peptides as negative samples. However, search engines usually only report the top-ranked peptide. In our initial analysis, we were able to identify the next-ranked peptides for less than five percent of the spectra. As such, it is imperative to devise a strategy to obtain the next-ranked peptides reliably. To address this challenge, we slightly modify the search process to generate multiple peptide candidates for each spectrum and rank them according to their initial scores. We divide the database into k parts and utilize them to conduct k separate database searches. This ensures that we obtain at least one PSM from each database for each spectrum in the combined search result. Subsequently, we sort the merged result by the initial score to identify the top-ranked and next-ranked peptides for each spectrum. By employing this technique with $k = 3$, we are able to identify the next-ranked peptides in a significant majority (over 86%) of the spectra, whereas in a complete database search, this percentage is less than 5%.

5.2.2 Next-ranked Peptides

Among the list of peptides for one spectrum, the peptide that receives the highest score or rank is referred to as the best or top-ranked peptide. Peptides that do not closely resemble the top-ranked peptide are considered as candidates for the next-ranked peptide. The next-ranked peptide is the highest-ranked peptide from the list of candidates, regardless of its original position in the initial list of candidates. To identify the differences between the top-ranked and the candidate peptides, various similarity measures were employed.

Peptide Similarity Measure.

The computation of peptide resemblance is a crucial step in generating the next-ranked candidate peptides. Two key factors considered for peptide resemblance are: 1) sequence similarity and 2) fragment ion similarity. Naturally, the similarity of the peptide sequence was assessed as the first distance metric, where sequence similarity was determined using the Levenshtein distance [95], which measures the edit distance between the primary sequence of each peptide. A distance threshold of 0.3 has been established to define peptide sequence similarity, where a lower edit distance indicates a higher level of similarity between two peptides. In calculating peptide sequence similarity, isobaric peptides (I, L) and post-translational modifications (PTMs) that result in a similar mass are treated as interchangeable (to be discussed shortly).

In addition, theoretical ion matches were computed to further evaluate peptide similarity. To evaluate fragment ion similarity, the theoretical spectra of the peptides were compared instead of their sequences. A similarity percentage was determined by comparing the matching ions between the theoretical spectra of the peptides. To assess fragment ion similarity, a threshold of 0.5 has been established. We considered both b- and y-ions in generating the similarity percentage. PSMs that meet the threshold criteria for both metrics within a given spectrum are considered as potential candidates for the next-ranked peptide. Therefore, a peptide is considered a candidate when it has more than 30% sequence difference from the top-ranked peptide and no more than 50% of the fragment ions from the theoretical spectra of both peptides coincide. These percentages have been chosen on the basis of intuitive reasoning.

Post Translational Modifications.

In functional proteomics, PTMs are critical biochemical modifications of proteins. PTMs can occur at any stage of a protein's existence, including during mass spectrometry analysis. Some PTMs may result in similar masses as other amino acid residues, posing challenges in distinguishing them based solely on mass spectrometry data. For instance, deamidation of asparagine ($N+0.984$) can result in mass nearly identical to aspartic acid (D), deamidation of glutamine ($Q+0.984$) can result in mass similar to glutamic acid (E), and oxidation of methionine ($M + 15.995$) can result in mass almost same as phenylalanine (F). As a result, if all other factors are identical, the m/z values will be the same for both peptides. To address this issue, we considered such pairs of peptides with similar masses due to PTMs as interchangeable in our next-ranked peptide calculation, treating them as having the same rank. As a result, if one such peptide is the top-ranked peptide, the other one will not be placed on the candidate list of next-ranked peptides.

5.3 Results

Based on our hypothesis, we consider top-ranked PSMs as positive samples and next-ranked PSMs as negative samples for training our scoring function with additional features. We then re-score the PSMs, rearrange the PSMs based on the new scores, and calculate the FDR from the target and decoy PSMs. Importantly, our algorithm does not require identifying the decoy PSMs to re-score or re-rank the PSMs. The decoy database is incorporated only for performance comparison with existing methods, and our algorithm remains blind to the target-decoy labelling until evaluation. In our experiments, we utilized four datasets: 1. Human HeLa (ProteomeXchange project ID: PXD015028) dataset, [90] 2. Mouse muscle spindle (ProteomeXchange project ID: PXD035552) dataset, [10] 3. Human pulmonary microvascular endothelial cells (ProteomeXchange project ID: PXD036260) dataset, [78] and 4. Human HeLa (ProteomeXchange project ID: PXD005280) dataset. [9] Uniprot databases were used as the target database in our experiments. We generated the decoy databases for target-decoy database search using a repeat-preserving decoy algorithm. [90] For our initial experiment, we utilized the first human HeLa dataset (ProteomeXchange project ID: PXD015028), the UniProt human database as the target database, and Comet [42] as the database search engine. We primarily sorted and ranked the PSMs using the 'expect' score ($-\log Ev$). This dataset contains approximately 97k top-ranked PSMs and 82k next-ranked PSMs. Since traditional database search engines typically only report top-ranked PSMs and dismiss the rest, to ensure sufficient next-ranked PSMs for our analysis, we employed a three-way split database search where we divide the database into three parts and conducted individual searches on each split. The HeLa dataset has around 72k top-ranked target PSMs, 25k top-ranked decoy PSMs, 44k next-ranked target PSMs, and 38k next-ranked decoy PSMs. Notably, the number of decoys in next-ranked PSMs is much higher than in top-ranked PSMs. We then used three additional features, namely 'enzC' (C-terminal tryptic), 'enzN' (N-terminal tryptic), and number of missed cleavages to train our scoring function. The default traditional database search algorithm using Comet search engine and the 'expect' score ($-\log Ev$) identified

ProteomeXchange ID	Sample type	Instrument type	# Taregt at 1%FDR		Improvement
			Default	Re-scored	
PXD015028	HeLa	Q Exactive	$31,931$	$34,512$	8.1%
PXD035552	Mouse	Q Exactive HF	$2,115$	$2,294$	8.5%
PXD036260	Human pulmonary	Q Exactive	$7,967$	$8,229$	3.3%
PXD005280	HeLa	Q Exactive	$11,289$	$12,408$	9.9%

Table 5.1: Numbers of target PSMs at 1% FDR reported by default method and our method using Comet as the database search engine for different datasets. The final column indicates the percentage of improvement achieved after re-scoring.

$31,931$ PSMs at 1% FDR. In contrast, our algorithm identified $34,512$ PSMs at 1% FDR after re-scoring, reporting 8.1% additional PSMs compared to the initial scoring method.

To ensure generalizability, we performed supplementary experiments using various species and samples. Specifically, we conducted experiments on human pulmonary microvascular endothelial cells and a distinct HeLa dataset, which yielded improvements of 3.3% and 9.9%, respectively. Additionally, our algorithm demonstrated uniform performance across diverse species, as evidenced by the noteworthy 8.5% improvement observed in the mouse dataset. Overall, our results, as presented in Table 5.1, suggest that our approach can consistently improve performance for both different samples within a given species and across different species.

We carried out another experiment to test whether overfitting plays a role in the improvement above. We divided the data into training and testing spectra with a $3:1$ ratio and trained the new scoring function only on the top and next-ranked peptides of the training spectra. We then tested the performance on the testing spectra. The retrained scoring function identified 7.9% additional PSMs at 1% FDR compared to the default scoring method. This improvement is very similar to the 8.1% improvement ratio obtained when the training and re-scoring were both on the whole dataset. This suggests that by hiding the target/decoy information from the training, we successfully avoided any significant overfit.

5.4 Conclusion

The presence of random spectra and incorrect PSMs due to poor spectra quality or background noises can affect the accuracy of peptide identification. Methods that use decoys as training data introduce potential bias in FDR calculation. In this study, we propose a novel post-processing algorithm to improve peptide identification without using target-decoy labeling. We suggest using the next-ranked PSMs, which are typically discarded in traditional methods, as negative samples to bypass the problem of decoy unmasking. The proposed method aims to retrain the scoring function after the initial database search to achieve a more accurate ranking of PSMs. Our study provides compelling evidence supporting the use of a curated machine learning algorithm to significantly enhance performance. By employing three meticulously selected features in our analysis, we achieved promising results. Nonetheless, our findings suggest that the integration of additional features has the potential to further augment our algorithm's efficacy.

Chapter 6

Advanced Machine Learning to Retrain and Re-score the PSMs

As indicated in Section 5.4, it has been proposed that the retraining method can benefit from the integration of advanced machine learning algorithms. In this chapter, we scrutinize the performance of our proposed re-scoring method when applied to datasets (PSMs) generated by different search engines (Section 6.1). Furthermore, we delve into the application of two advanced machine learning techniques to further substantiate the efficacy of utilizing *next-best* PSMs as suitable negative samples.

Our primary objective for this chapter is to enhance the performance of the re-scoring function by leveraging various well-established machine learning algorithms. Additionally, we offer a detailed comparison and discussion surrounding the utilization of *best* and *next-best* as positive and negative samples, respectively, within the framework of the Mokapot [45] algorithm.

6.1 Re-scoring PSMs: Results from Different Search Engines

To ensure the generalizability of our findings, we conducted a series of supplementary experiments that employed varying species, datasets, and database search engines. We have utilized search results from two distinct search engines, namely MS Amanda [34] and MS-GF+ [75], alongside Comet, to ensure that the effectiveness of our proposed method does not depend on the specific search engine.

MS Amanda is a specialized peptide identification algorithm tailored for high-accuracy and high-resolution mass spectrometry data. Notably, MS Amanda boasts remarkable accuracy, demonstrated by the substantial overlap in identified spectra compared to gold standard algorithms such as SEQUEST [43] and Mascot [99]. A notable benefit of MS Amanda, particularly with the introduction of MS Amanda 2.0 [35], is its exceptional speed in identification. Additionally, the algorithm has been expanded to facilitate a second search, enabling the identification of peptides within chimeric tandem mass spectra.

In contrast, MS-GF+ is optimized for a multitude of spectral types, which encompass diverse combinations of fragmentation methods, instruments, enzymes, and experimental protocols. It stands out as an MS/MS database search tool known for its sensitivity, capable of identifying more peptides compared to other database search tools and on par with spectral library search tools [75]. Notably, our results align with this claim, despite our original research goals being distinct from this specific evaluation.

6.2 Enhancing Re-scoring Function Performance: Advanced Machine Learning Models

In our pursuit of establishing the suitability of *next-best* peptide-spectrum matches as a representative approximation of the false distribution, we initially employed a logistic regression-based re-scoring function, as outlined in [91]. However, recognizing the potential for enhancing the efficacy of our re-scoring function, we sought to explore the application of advanced machine learning models.

To explore this avenue, we performed experiments with re-scoring functions implemented using XGBoost [17] as well as a straightforward neural network algorithm. These investigations were carried out within the framework of our baseline HeLa dataset, as detailed in Section 5.3. The objective of these experiments is twofold: firstly, to assess how these advanced machine learning models compare to the logistic regression approach for our study, and secondly, to strengthen the empirical evidence supporting the viability of employing *next-best* PSMs as negative examples to train our re-scoring function.

6.2.1 XGBoost

We utilize XGBoost [17], an ensemble machine learning algorithm based on decision trees, to implement our re-scoring function. The algorithm works by combining the predictions of multiple weak learners (typically decision trees) to create a strong predictive model. XGBoost, which stands for Extreme Gradient Boosting, is a renowned and versatile machine learning algorithm recognized for its exceptional performance across diverse domains, including bioinformatics. Its reputation for both speed and efficiency makes it a highly suitable choice for a wide range of applications, including the enhancement of PSM re-scoring in proteomics research.

6.2.2 Neural-network (NN) Model

Furthermore, we utilize a sequential neural network model to improve the performance of the PSM re-scoring function. This neural network architecture has been deliberately designed with a concise layer configuration, keeping it simple. The rationale behind this choice is to investigate the potential benefits that advanced machine learning models may bring to the performance of the re-scoring function. While this avenue holds great promise and is indeed exciting, we have decided to set it aside temporarily. Our rationale for this decision is twofold. Firstly, our foremost objective centers on establishing the adequacy of *next-best* as a proxy for the false distribution within the context of our proteomic research. By concentrating our efforts on this critical aspect, we ensure that the foundation of our methodology is solid and thoroughly validated.

Secondly, we lay the groundwork for a promising future direction. Specifically, we envision the development of a dedicated neural network model tailored explicitly to the task of re-scoring PSMs while using the *next-best* PSMs as negative samples.

6.3 Re-scoring PSMs: A Comparative Analysis of Mokapot with Proposed Positive and Negative Samples

In our ongoing efforts, we leverage Mokapot [45], a Python implementation of the well-established PSM post-processing method Percolator [70, 119], which is based on a semi-supervised learning algorithm. Mokapot is an open-source Python package, that can be found at https://github.com/wfondrie/mokapot.

In this phase of our study, we fine-tune Mokapot by training it with our proposed *best* and *next-best* PSMs, designated as positive and negative samples, respectively. We then proceed to compare its performance against the default approach, where targets and decoys are used as the positive and negative samples for PSM re-scoring. We maintain all other parameters at their default settings for Mokapot.

The only deviation from these default settings involves the modification of a specific parameter to additionally output the decoy confidence values. This adaptation is essential as it enables us to calculate the 1% FDR from the re-scored PSMs, utilizing the established method of false discovery rate calculation that considers both target and decoy counts. Subsequently, we analyze the resulting PSMs reported by Mokapot with the original target-decoy labeling. Importantly, it should be emphasized that our reporting of results does not factor in ranking information, specifically whether a PSM is classified as *best* or *next-best*. Similarly, we do not provide Mokapot with the target-decoy labels.

6.4 Results and Discussion

In our experimental setup, we worked with two distinct datasets derived from different species: 1. Human HeLa (ProteomeXchange project ID: PXD015028) dataset [90], and 2. Mouse muscle spindle (ProteomeXchange project ID: PXD035552) dataset [10]. Uniprot databases were used as the target database in our experiments. We generated the decoy databases for target-decoy database search using a repeat-preserving decoy algorithm [90]. Our baseline analysis was conducted on the initial Human HeLa dataset (ProteomeXchange project ID: PXD015028), using the UniProt human database as the target reference, as described in Chapter 5. During the database search process, we implemented the 3-way split database search method as outlined in [91].

In this chapter, we initiate our analysis by evaluating the effectiveness of our re-scoring approach when applied to PSMs obtained from two different search engines, MS Amanda and MS-GF+. In this assessment, we incorporate two boolean features, namely 'enzC' and 'enzN', which signify whether the C-terminal and N-terminal are tryptic, respectively. Additionally, the number of missed cleavages is utilized as a feature in some of our experiments, as mentioned in the respective results. These features serve as inputs for our machine learning-based re-scoring function. Subsequently, we provide a comprehensive performance comparison of various implementations of the re-scoring function as applied to Comet-generated PSMs. These results offer valuable insights into the effectiveness of advanced machine learning methods in contrast to the logistic regression-based approach discussed in Chapter 5.

In determining the number of training spectra, we also consider that the training data contains both top-ranked and next-ranked PSMs, while the testing dataset only contains the top-ranked PSMs. So, if we want 75% training data and 25% testing data, our number of training spectra will be 60% of the total spectra. A similar strategy was applied while testing for overfitting in the previous chapter (Chapter 5).

Data	Search Engine	# Target at 1%FDR		Improvement	Additional Features
		Target-Decoy	Re-scored		
HeLa PXD015028	MS Amanda	11,289	11,559	2.39%	‘enzC’ and ‘enzN’
HeLa PXD015028	MS-GF+	35,145	35,244	0.28%	‘enzC’ and ‘enzN’
Mouse PXD035552	MS-GF+	2,114	2,281	7.9%	‘enzC’ and ‘enzN’

Table 6.1: Numbers of Target PSMs at 1% FDR with different search engines and datasets. The second-to-last column represents the percentage of improvement achieved after the re-scoring.

Data	Re-scoring Function	# Target at 1%FDR		Improvement	Additional Features
		Target-Decoy	Re-scored		
HeLa PXD015028	Logistic Regression	31,931	34,512	8.1%	‘enzC’, ‘enzN’, and number of missed cleavages
HeLa PXD015028	Neural-network	31,931	34,415	7.78%	‘enzC’, ‘enzN’, and number of missed cleavages
HeLa PXD015028	XGBoost	31,931	33,226	4.06%	‘enzC’, ‘enzN’, and number of missed cleavages

Table 6.2: Number of target PSMs at 1% FDR with different re-scoring functions for Comet PSMs. The second-to-last column represents the percentage of improvement achieved after the re-scoring.

We integrated MS Amanda [34] and MS-GF+[75] as additional database search engines in our analysis. Despite using only two features, 'enzC' and 'enzN,' the MS Amanda [34] experiment resulted in a 2.39% improvement in PSM scoring accuracy (Table 6.1). Similarly, in the case of MS-GF+[75], where we also employed only 'enzC' and 'enzN,' the outcomes varied across different datasets. For instance, in the HeLa replicate one dataset (PXD015028), we observed a modest improvement of 0.28%, as shown in Table 6.1. However, it is noteworthy that in the Mouse dataset (PXD035552), a substantial improvement of 7.9% is observed.

Furthermore, we present a Venn diagram (Figure 6.1) illustrating the number of PSMs that overlap and those that remain exclusive among the 1% FDR results reported by the default scoring and re-scoring methods. For this diagram, we utilized the Comet-generated PSMs from our baseline HeLa dataset, as detailed in Section 4.7. Following the re-scoring process, we observed an increase of 4,376 new PSMs; however, we also experienced the loss of 1,711 PSMs at the 1% FDR threshold. To assess the quality of the gained and discarded PSMs, a closer examination of the results is necessary. Specifically, we identified that the number of decoys gained through re-scoring was 309, while the number of decoys removed amounted to 297. Figure 6.2 illustrates the score distribution of newly acquired PSMs and eliminated PSMs obtained at the 1% FDR threshold after re-scoring. The plot clearly indicates that post-re-scoring, we acquire a higher number of PSMs with higher scores and simultaneously exclude PSMs with lower scores.

As an additional experiment, we exclusively examine the baseline HeLa dataset, retaining all Mokapot parameters in their default settings. Nevertheless, we deviate from the default configuration by adjusting a specific parameter to generate confidence values for decoy PSMs. By default, Mokapot outputs confidence values only for target PSMs. This adaptation is pivotal in enabling the calculation of the 1% FDR from the re-scored PSMs when employing target-decoy labeling.

As part of our control results, we perform an experiment in which Mokapot is trained with the traditional approach: target PSMs are labeled as positive (label=1), and decoy PSMs are labeled as negative (label=-1). This approach represents the conventional standard.

Our primary objective is to assess how our proposed method integrates with Mokapot. To achieve this, we train Mokapot using our *best* and *next-best* PSMs as positive and negative samples, respectively. We accomplish this by assigning the label '1' to *best* and '-1' to *next-best*. Subsequently, we identified the results reported by Mokapot with our customized training data, and we analyzed these results using the available target-decoy labeling. It is important to note that our results reporting does not consider ranking

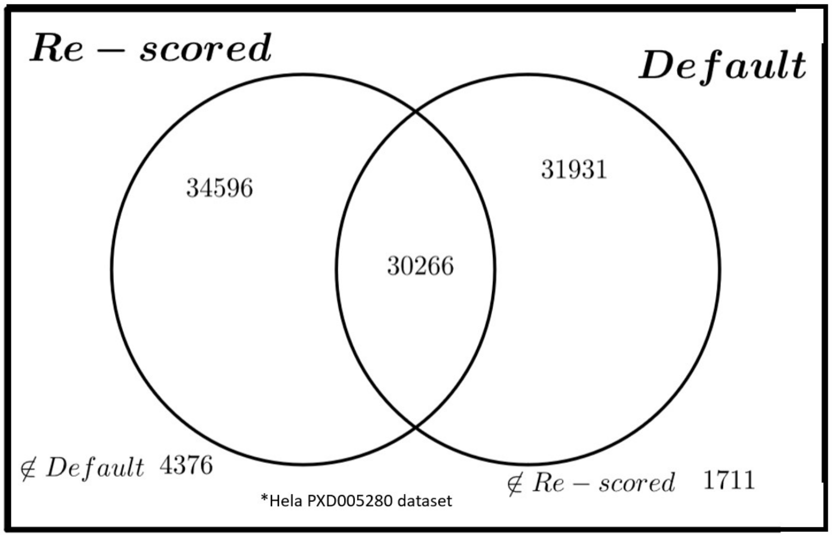


Figure 6.1: Overlap and discrepancy in PSMs between default scoring and re-scoring methods at 1% FDR using Comet-generated PSMs, using baseline HeLa data. The Venn diagram illustrates the number of PSMs reported in both the default and re-scored approaches, the number of PSMs that only appear after re-scoring ($\notin$ default), and the number of PSMs that are removed after re-scoring ($\notin$ re-scored).

information, specifically whether a PSM is categorized as *best* or *next-best*. Essentially, during training, we manipulate Mokapot into treating *best* and *next-best* as targets and decoys, respectively.

Following this, we proceed to compare the performance of this modified approach with the default method, where targets and decoys are employed as the positive and negative samples for PSM re-scoring. The results of this comparative analysis are presented in Table 6.3. We present results while training with different subsets of features that Comet reports. The feature descriptions listed in Table 6.4 are primarily sourced from [70]. For more comprehensive details on these features, refer to [41]. Interestingly, as shown in Table 6.3, Mokapot's performance improves significantly when trained with *best* and *next-best* PSMs after removing certain features. However, its performance remains relatively stable when trained with target and decoy PSMs.

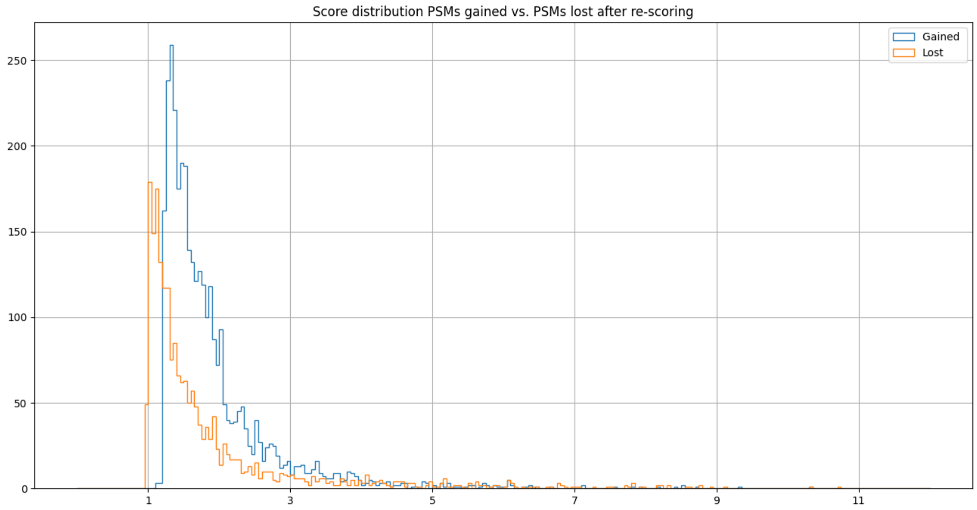


Figure 6.2: Distribution of the PSMs gained vs. lost after re-scoring.

6.5 Conclusion

In summary, this chapter explored the re-scoring of PSMs obtained from MS Amanda and MS-GF+ database search engines, revealing varying improvements in accuracy across diverse datasets. Our analysis of the score distribution among the gained and discarded PSMs provided valuable insights into the enhanced quality achieved through re-scoring, particularly in the identification of higher-scoring PSMs. These findings underscore the potential of advanced re-scoring methods in proteomics research, highlighting the significance of careful analysis and evaluation to optimize their effectiveness.

In addition, this chapter introduced the results of an enhanced re-scoring function implemented with advanced machine learning techniques such as XGBoost and neural networks. These findings enrich our understanding of the potential enhancements achievable within the re-scoring function through the application of advanced computational approaches, emphasizing the significance of optimizing re-scoring methodologies for accurate PSM identification.

Furthermore, the findings presented in this chapter indicate that the presence of inherent similarities among numerous proteins in the target database poses a challenge for cross-validation methods similar to the one used in Mokapot, as they may not entirely prevent the inadvertent leakage of target information.

best/next-best	target/decoy	Training Features
32,186	35,822	‘ExpMass’, ‘CalcMass’, ‘lnrSp’, ‘deltLCn’, ‘deltCn’, ‘lnExpect’, ‘Xcorr’, ‘Sp’, ‘IonFrac’, ‘Mass’, ‘PepLen’, ‘Charge’, ‘Charge1’, ‘Charge3’, ‘Charge4’, ‘Charge5’, ‘Charge6’, ‘enzN’, ‘enzC’, ‘enzInt’, ‘lnNumSP’, ‘dM’, ‘absdM’
35,739	35,759	‘enzN’, ‘enzC’, ‘enzInt’, ‘Charge’
35,855	35,745	‘enzN’, ‘enzC’, ‘enzInt’

Table 6.3: Number of target PSMs at 1% FDR reported by Mokapot when trained with various features and different sets of positive and negative samples (*best/next-best* in the first column, target/decoy in the second column). Improved performance after re-scoring with Mokapot when proposed *best/next-best* used as positive and negative samples.

Feature Name	Description
ExpMass	Experimental Mass
XCorr	Cross-correlation between theoretical and experimental spectra.
Sp	Initial score comparing the peptide to the predicted fragment ion values.
dM	The difference between the theoretical and experimental peptide mass in Daltons (Da).
absdM	The absolute value of the difference in theoretical and experimental mass.
ionFrac	The ratio of matched b- and y- ions
ln(NumSp)	The natural logarithm of the number of in-sillico database peptides present within the specified m/z range.
enzN	Boolean: Is the N terminal an enzymatic (tryptic) site?
enzC	Boolean: Is the C terminal an enzymatic (tryptic) site?
enzInt	Number of missed internal enzymatic (tryptic) sites or the number of missed cleavages.
Charge	The charge state.
Charge1–6	Six Boolean features indicating the charge state.
PepLen	Lenght of the peptide
deltaCn	The normalized difference in XCorr for this PSM compared to the subsequent ranked PSM for the same spectrum and charge.
delta_Icn	Similar to delta_cn, except the difference is calculated in relation to the lowest reported XCorr score for a specific spectrum and charge state.

Table 6.4: List of features for Mokapot including a brief description

Chapter 7

Conclusion

Our research, which aimed to improve peptide identification rates, was driven by two core, interconnected strategies. The first focused on enhancing decoy generation methods to address inherent biases and limitations, delving into the complexities of creating effective decoys. The second strategy diverged from traditional target-decoy paradigms, introducing a methodology based on the premise that an absent target peptide in a sample is functionally akin to a decoy peptide. This led to a novel approach to use search results beyond conventional labels, notably generating and utilizing additional PSMs for each spectrum. This approach not only aimed to refine peptide identification processes, but also ventured into new areas of false distribution modeling and FDR estimation. Collectively, these strategies signify a comprehensive effort to advance proteomics research in peptide identification and validation.

7.1 Summary of Contributions

Two scholarly articles have been published from this research: the first centers on developing a repeat-preserving decoy technique [90], while the second addresses the re-scoring of peptide spectrum matches through the use of *next-best* PSMs [91].

7.1.1 Repeat-Preseving Decoy

Current decoy generation methods lack one or more properties of an ideal decoy. This encouraged the use of de Bruijn graphs to generate repeat-preserving decoys. A repeat-preserving decoy is a type of decoy sequence generated for proteomic studies in such a way that it maintains the same repetitive patterns and sequences as the target protein but with differences that make it a decoy. This preservation of repeats ensures that the number and distribution of repeated elements in the decoy remain similar to those in the target sequence. The goal is to create a decoy that closely mimics the target's repetitive structure while still being distinct, allowing it to be used effectively in false discovery rate estimation during proteomics experiments.

The proposed method is not dependent on the specificity and completeness of enzyme digestion, which signifies a novel advancement in improving the quality of decoy database generation. This technique not only maintains a balance between the quantities of unique target and decoy peptides, but also preserves the statistical properties, thereby contributing to a more accurate and reliable false discovery rate estimation. An example implementation of the algorithm in Java can be found at https://github.com/johramoosa/deBruijn. Furthermore, the Human Hela dataset used in this project can be downloaded from ProteomeXchange (ProteomeXchange project ID: PXD015028).

7.1.2 Retraining with *next-best* PSMs

We introduce the concept of *next-best* peptide-spectrum matches, derived from data typically disregarded in conventional practice. While we continue to employ target-decoy labels for validation purposes, our hypothesis suggests avoiding their use during the learning phase. This aims to ensure an unbiased target-decoy false discovery rate validation process. Our approach not only provides the opportunity to utilize *next-best* PSMs for FDR estimation but also broadens its applicability to other domains within the field of proteomics, such as spectral library searches.

7.2 Limitations and Challenges

Although the repeat-preserving de Bruijn decoy effectively addresses the imbalance between the counts of unique targets and decoys, it does not fully resolve the inherent trade-off between repeat-preservation and randomness. As detailed in Section 3.3.4, any method for generating repeat-preserving decoys must maintain a certain degree of correlation between different segments of the target database, precluding total randomness. Nevertheless, an appreciable degree of randomness is desirable. For instance, our empirical data reveal that employing deterministic rules for decoy generation, such as reversal or shifted reversal techniques, can lead to a systematic increase in false positive matches. This increase is largely attributed to nuanced similarities between target peptides and their decoy counterparts as well as to the scoring algorithms employed by the search engine. Alternatively, the employment of a different deterministic generation method could conceivably lead to a systematic reduction in decoy hits. The inclusion of randomness serves to temper this systematic bias. Within the framework of our de Bruijn decoy methodology, the tunable parameter k allows subtle control over the trade-off between randomness and repeat preservation. Increasing k enhances the randomness by increasing the number of random parameters, albeit at the cost of decreasing the length of preserved repeats.

Although the conceptual utility of *best* and *next-best* categories is compelling, and our preliminary findings are promising, the generalizability of the method across various search engines and datasets is limited for FDR estimation. The optimization procedures are not only time-consuming but also dataset-specific, making the approach impractical for broader applications, especially when the gains are marginal.

Our experiments using *best* and *next-best* PSMs for re-scoring have yielded encouraging outcomes. However, the intricate relationship between the features and *next-best* peptide spectrum warrants further investigation. Our preliminary analyses indicate that the incorporation of certain features adversely impacts performance during retraining with *best* and *next-best* PSMs as presented in Table 6.3. Sophisticated deep-learning methodologies may be essential for deciphering these complex correlations.

7.3 Future Research Directions

We recognize the allure and potential of advanced machine learning models, which have demonstrated remarkable capabilities in various domains. However, our commitment to the current research objective of validating *next-best* PSMs as suitable negative samples has required us to temporarily delay the exploration of more complex neural network architectures.

It is important to note that the scope of our research is not limited to the current study. Beyond our existing research, we present a future avenue of research specifically dedicated to crafting a deep neural network model that is optimized for re-scoring PSMs, with an emphasis on deciphering the intricate relationships between features and *next-best* PSMs. In particular, the utilization of explainable AI could prove advantageous for elucidating the impact of various features.

Another interesting research direction can be to investigate methods to integrate proteomic data with other omics data, such as transcriptomics, to gain a more comprehensive understanding of the efficacy of the proposed re-scoring method. The exploration of advanced and more comprehensive methods to calculate PSM similarity, such as the incorporation of other omics data, holds significant potential benefits. For example, by combining proteomic and transcriptomic information, researchers can gain a deeper understanding of the relationships between gene expression and protein abundance. It is improbable that a peptide originating from an abundant protein is incorrect, rendering it an unlikely candidate to be classified as the *next-best* PSM.

Furthermore, the development of more efficient and universally applicable augmentation methods for FDR estimation using *next-best* PSMs could mitigate the need for dataset-specific optimization, thus making them applicable to a wider range of search engines.

7.4 Conclusion

In summation, this research has made noteworthy strides in the domain of proteomics, specifically addressing the complexities involved in peptide identification and validation. We have successfully implemented repeat-preserving de Bruijn decoys to tackle the issue of imbalance between the number of unique targets and decoys. Despite the trade-off of repeat preservation and randomness, our methods signify an important advancement in decoy database generation.

Our exploration of retraining techniques using *best* and *next-best* PSMs as positive and negative samples has shown promise. This method effectively curtails the leakage of target-decoy information, as manifested by the enhanced identification rates recorded in various experiments. Moreover, we recognize that machine learning models serve as a rich domain for subsequent research endeavors aimed at elucidating the intricate relationship between features and PSMs. While the technique has not yet achieved universal applicability for FDR across diverse search engines and datasets, the initial findings affirm its promise.

Beyond the current investigation, we introduce a series of subsequent studies focused on applying advanced machine learning techniques. This agenda includes the use of explainable AI to understand feature effects, the formulation of sophisticated re-scoring functions, and the development of more efficient FDR estimation techniques that are applicable to a wider range of search engines.

This research made novel contributions to the rapidly progressing field of proteomics. We are confident that the methodologies and findings of this work will catalyze ongoing advancements in peptide identification and validation techniques. Addressing existing limitations and providing innovative solutions, this work sets the stage for additional research in this essential domain.

References

[1] Erik Ahrné, Yuki Ohta, Frederic Nikitin, Alexander Scherl, Frederique Lisacek, and Markus Müller. An improved method for the construction of decoy peptide MS/MS spectra suitable for the accurate estimation of false discovery rates. *Proteomics*, 11(20):4085–4095, 2011.

[2] Nadia Allet, Nicolas Barrillat, Thierry Baussant, Celia Boiteau, Paolo Botti, Lydie Bougueleret, Nicolas Budin, Denis Canet, Stéphanie Carraud, Diego Chiappe, et al. In vitro and in silico processes to identify differentially expressed proteins. *Proteomics*, 4(8):2333–2351, 2004.

[3] Anton Bankevich, Sergey Nurk, Dmitry Antipov, Alexey A Gurevich, Mikhail Dvorkin, Alexander S Kulikov, Valery M Lesin, Sergey I Nikolenko, Son Pham, Andrey D Prjibelski, et al. SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. *Journal of computational biology*, 19(5):455–477, 2012.

[4] Harald Barsnes and Marc Vaudel. SearchGUI: a highly adaptable common interface for proteomics search and de novo engines. *Journal of proteome research*, 17(7):2552–2555, 2018.

[5] Markus Bauer, Gunnar W Klau, and Knut Reinert. Accurate multiple sequence-structure alignment of rna sequences using combinatorial optimization. *BMC bioinformatics*, 8(1):1–18, 2007.

[6] Daniel Becker, Matthias Selbach, Claudia Rollenhagen, Matthias Ballmaier, Thomas F Meyer, Matthias Mann, and Dirk Bumann. Robust Salmonella metabolism limits possibilities for new antimicrobials. *Nature*, 440(7082):303, 2006.

[7] Isma Belouah, Mélisande Blein-Nicolas, Thierry Balliau, Yves Gibon, Michel Zivy, and Sophie Colombié. Peptide filtering differently affects the performances of XIC-based quantification methods. *Journal of proteomics*, 193:131–141, 2019.

[8] Adele R Blackler, Aaron A Klammer, Michael J MacCoss, and Christine C Wu. Quantitative comparison of proteomic data quality between a 2d and 3d quadrupole ion trap. *Analytical chemistry*, 78(4):1337–1344, 2006.

[9] Bernhard Blank-Landeshammer, Laxmikanth Kollipara, Karsten Biß, Markus Pfenninger, Sebastian Malchow, Konstantin Shuvaev, René P Zahedi, and Albert Sickmann. Combining de novo peptide sequencing algorithms, a synergistic approach to boost both identifications and confidence in bottom-up proteomics. *Journal of Proteome Research*, 16(9):3209–3218, 2017.

[10] Bavat Bornstein, Lia Heinemann-Yerushalmi, Sharon Krief, Ruth Adler, Bareket Dassa, Dena Leshkowitz, Minchul Kim, Guy Bewick, Robert W Banks, and Elazar Zelzer. Molecular characterization of the intact mouse muscle spindle using a multiomics approach. *ELife*, 12:e81843, 2023.

[11] Daniel R Boutz, Andrew P Horton, Yariv Wine, Jason J Lavinder, George Georgiou, and Edward M Marcotte. Proteomic identification of monoclonal antibodies from serum. *Analytical chemistry*, 86(10):4758–4766, 2014.

[12] Natalie Castellana and Vineet Bafna. Proteogenomics to discover the full coding content of genomes: a computational perspective. *Journal of proteomics*, 73(11):2124–2135, 2010.

[13] Natalie E Castellana, Samuel H Payne, Zhouxin Shen, Mario Stanke, Vineet Bafna, and Steven P Briggs. Discovery and revision of Arabidopsis genes by proteogenomics. *Proceedings of the national academy of sciences*, 105(52):21034–21038, 2008.

[14] Mark J Chaisson and Pavel A Pevzner. Short read fragment assembly of bacterial genomes. *Genome research*, 18(2):324–330, 2008.

[15] Matthew C Chambers, Brendan Maclean, Robert Burke, Dario Amodei, Daniel L Ruderman, Steffen Neumann, Laurent Gatto, Bernd Fischer, Brian Pratt, Jarrett Egertson, Katherine Hoff, Darren Kessner, Natalie Tasman, Nicholas Shulman, Barbara Frewen, Tahmina A Baker, Mi-Youn Brusniak, Christopher Paulse, David Creasy, Lisa Flashner, Kian Kani, Chris Moulding, Sean L Seymour, Lydia M Nuwaysir, Brent Lefebvre, Frank Kuhlmann, Joe Roark, Paape Rainer, Suckau Detlev, Tina

Hemenway, Andreas Huhmer, James Langridge, Brian Connolly, Trey Chadick, Krisztina Holly, Josh Eckels, Eric W Deutsch, Robert L Moritz, Jonathan E Katz, David B Agus, Michael MacCoss, David L Tabb, and Parag Mallick. A cross-platform toolkit for mass spectrometry and proteomics. *Nature Biotechnology*, 30:918, oct 2012.

[16] Lu Chen, Nan Wang, Difei Sun, and Liang Li. Microwave-assisted acid hydrolysis of proteins combined with peptide fractionation and mass spectrometry analysis for characterizing protein terminal sequences. *Journal of proteomics*, 100:68–78, 2014.

[17] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[18] Hyungwon Choi and Alexey I Nesvizhskii. False discovery rates and related statistical concepts in mass spectrometry-based proteomics. *Journal of proteome research*, 7(01):47–50, 2008.

[19] Jacques Colinge, Alexandre Masselot, Marc Giron, Thierry Dessingy, and Jérôme Magnin. OLAV: Towards high-throughput tandem mass spectrometry data identification. *PROTEOMICS: International Edition*, 3(8):1454–1463, 2003.

[20] Phillip EC Compeau, Pavel A Pevzner, and Glenn Tesler. How to apply de bruijn graphs to genome assembly. *Nature biotechnology*, 29(11):987, 2011.

[21] Phillip EC Compeau, Pavel A Pevzner, and Glenn Tesler. Why are de Bruijn graphs useful for genome assembly? *Nature biotechnology*, 29(11):987, 2011.

[22] UniProt Consortium. UniProt: a hub for protein information. *Nucleic acids research*, 43(D1):D204–D212, 2015.

[23] Bret Cooper. The problem with peptide presumption and low Mascot scoring. *Journal of proteome research*, 10(3):1432–1435, 2011.

[24] Bret Cooper. The problem with peptide presumption and the downfall of target–decoy false discovery rates. *Analytical chemistry*, 84(22):9663–9667, 2012.

[25] John A Cormican, Yehor Horokhovskyi, Wai T Soh, Michele Mishto, and Juliane Liepe. inSPIRE: An open-source tool for increased mass spectrometry identification rates using Prosit spectral prediction. *Molecular & Cellular Proteomics*, 21(12), 2022.

[26] Yohann Couté, Christophe Bruley, and Thomas Burger. Beyond target–decoy competition: Stable validation of peptide and protein identifications in mass spectrometry-based discovery proteomics. *Analytical Chemistry*, 92(22):14898–14906, 2020.

[27] Jürgen Cox and Matthias Mann. MaxQuant enables high peptide identification rates, individualized ppb-range mass accuracies and proteome-wide protein quantification. *Nature biotechnology*, 26(12):1367, 2008.

[28] Jürgen Cox, Nadin Neuhauser, Annette Michalski, Richard A Scheltema, Jesper V Olsen, and Matthias Mann. Andromeda: a peptide search engine integrated into the MaxQuant environment. *Journal of proteome research*, 10(4):1794–1805, 2011.

[29] Robertson Craig and Ronald C Beavis. TANDEM: matching proteins with tandem mass spectra. *Bioinformatics*, 20(9):1466–1467, 2004.

[30] Yulia Danilova, Anastasia Voronkova, Pavel Sulimov, and Attila Kertész-Farkas. Bias in false discovery rate estimation in mass-spectrometry-based peptide identification. *Journal of proteome research*, 18(5):2354–2358, 2019.

[31] Nicolaas Govert De Bruijn. A combinatorial problem. *Koninklijke Nederlandse Akademie v. Wetenschappen*, 49(49):758–764, 1946.

[32] Eric W Deutsch, Luis Mendoza, David Shteynberg, Terry Farrah, Henry Lam, Natalie Tasman, Zhi Sun, Erik Nilsson, Brian Pratt, Bryan Prazen, et al. A guided tour of the Trans-Proteomic Pipeline. *Proteomics*, 10(6):1150–1159, 2010.

[33] Eric W Deutsch, Yasset Perez-Riverol, Robert J Chalkley, Mathias Wilhelm, Stephen Tate, Timo Sachsenberg, Mathias Walzer, Lukas Käll, Bernard Delanghe, Sebastian Böcker, et al. Expanding the use of spectral libraries in proteomics. *Journal of proteome research*, 17(12):4051–4060, 2018.

[34] Viktoria Dorfer, Peter Pichler, Thomas Stranzl, Johannes Stadlmann, Thomas Taus, Stephan Winkler, and Karl Mechtler. MS Amanda, a universal identification algorithm optimized for high accuracy tandem mass spectra. *Journal of proteome research*, 13(8):3679–3684, 2014.

[35] Viktoria Dorfer, Marina Strobl, Stephan Winkler, and Karl Mechtler. MS Amanda 2.0: Advancements in the standalone implementation. *Rapid Communications in Mass Spectrometry*, 35(11):e9088, 2021.

[36] Nathan Edwards and Ross Lippert. Sequence database compression for peptide identification from tandem mass spectra. In *International Workshop on Algorithms in Bioinformatics*, pages 230–241. Springer, 2004.

[37] Martin Eisenacher. mzIdentML: an open community-built standard format for the results of proteomics spectrum identification algorithms. In *Data Mining in Proteomics*, pages 161–177. Springer, 2011.

[38] Martin Eisenacher, Michael Kohl, Michael Turewicz, Markus-Hermann Koch, Julian Uszkoreit, and Christian Stephan. Search and decoy: the automatic identification of mass spectra. In *Quantitative Methods in Proteomics*, pages 445–488. Springer, 2012.

[39] Joshua E Elias and Steven P Gygi. Target-decoy search strategy for increased confidence in large-scale protein identifications by mass spectrometry. *Nature methods*, 4(3):207, 2007.

[40] Joshua E Elias and Steven P Gygi. Target-decoy search strategy for mass spectrometry-based proteomics. In *Proteome bioinformatics*, pages 55–71. Springer, 2010.

[41] Jimmy K Eng, Michael R Hoopmann, Tahmina A Jahan, Jarrett D Egertson, William S Noble, and Michael J MacCoss. A deeper look into Comet—implementation and features. *Journal of the American Society for Mass Spectrometry*, 26(11):1865–1874, 2015.

[42] Jimmy K Eng, Tahmina A Jahan, and Michael R Hoopmann. Comet: an open-source MS/MS sequence database search tool. *Proteomics*, 13(1):22–24, 2013.

[43] Jimmy K Eng, Ashley L McCormack, and John R Yates. An approach to correlate tandem mass spectral data of peptides with amino acid sequences in a protein database. *Journal of the american society for mass spectrometry*, 5(11):976–989, 1994.

[44] Jian Feng, Daniel Q Naiman, and Bret Cooper. Probability-based pattern recognition and statistical framework for randomization: modeling tandem mass spectrum/peptide sequence false match frequencies. *Bioinformatics*, 23(17):2210–2217, 2007.

[45] William E Fondrie and William S Noble. mokapot: fast and flexible semisupervised learning for peptide detection. *Journal of Proteome Research*, 20(4):1966–1971, 2021.

[46] Ari M Frank. A ranking-based scoring function for peptide- spectrum matches. *Journal of proteome research*, 8(5):2241–2252, 2009.

[47] Joachim Friedrich, Thomas Dandekar, Matthias Wolf, and Tobias Müller. ProfDist: a tool for the construction of large phylogenetic trees based on profile distances. *Bioinformatics*, 21(9):2108–2109, 2005.

[48] Olivier Gascuel. BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data. *Molecular biology and evolution*, 14(7):685–695, 1997.

[49] Lewis Y Geer, Sanford P Markey, Jeffrey A Kowalak, Lukas Wagner, Ming Xu, Dawn M Maynard, Xiaoyu Yang, Wenyao Shi, and Stephen H Bryant. Open mass spectrometry search algorithm. *Journal of proteome research*, 3(5):958–964, 2004.

[50] Daniel Gerlach, Matthias Wolf, Thomas Dandekar, Tobias Müller, Andreas Pokorny, and Sven Rahmann. Deep metazoan phylogeny. *In silico biology*, 7(2):151–154, 2007.

[51] Siegfried Gessulat, Tobias Schmidt, Daniel Paul Zolg, Patroklos Samaras, Karsten Schnatbaum, Johannes Zerweck, Tobias Knaute, Julia Rechenberger, Bernard Delanghe, Andreas Huhmer, et al. Prosit: proteome-wide prediction of peptide tandem mass spectra by deep learning. *Nature methods*, 16(6):509–518, 2019.

[52] Giulia Gonnelli, Michiel Stock, Jan Verwaeren, Davy Maddelein, Bernard De Baets, Lennart Martens, and Sven Degroeve. A decoy-free approach to the identification of peptides. *Journal of proteome research*, 14(4):1792–1798, 2015.

[53] Patricia A Gonzales, Trairak Pisitkun, Jason D Hoffert, Dmitry Tchapyjnikov, Robert A Star, Robert Kleta, Nam Sun Wang, and Mark A Knepper. Large-scale proteomics and phosphoproteomics of urinary exosomes. *Journal of the American Society of Nephrology*, 20(2):363–379, 2009.

[54] Manfred G Grabherr, Brian J Haas, Moran Yassour, Joshua Z Levin, Dawn A Thompson, Ido Amit, Xian Adiconis, Lin Fan, Raktima Raychowdhury, Qiandong Zeng, et al. Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nature biotechnology*, 29(7):644, 2011.

[55] Manfred G Grabherr, Brian J Haas, Moran Yassour, Joshua Z Levin, Dawn A Thompson, Ido Amit, Xian Adiconis, Lin Fan, Raktima Raychowdhury, Qiandong Zeng, et al. Trinity: reconstructing a full-length transcriptome without a genome from RNA-Seq data. *Nature biotechnology*, 29(7):644, 2011.

[56] Alejandro Grajales, Catalina Aguilar, and Juan A Sánchez. Phylogenetic reconstruction using secondary structures of internal transcribed spacer 2 (its2, rdna): finding the molecular and morphological gap in caribbean gorgonian corals. *BMC Evolutionary Biology*, 7(1):1–9, 2007.

[57] Viktor Granholm and Lukas Käll. Quality assessments of peptide–spectrum matches in shotgun proteomics. *Proteomics*, 11(6):1086–1093, 2011.

[58] Viktor Granholm, William S Noble, and Lukas Käll. On using samples of known protein content to assess the statistical calibration of scores assigned to peptide-spectrum matches in shotgun proteomics. *Journal of proteome research*, 10(5):2671–2678, 2011.

[59] John T Halloran, Jeff A Bilmes, and William S Noble. Learning peptide-spectrum alignment models for tandem mass spectrometry. In *Uncertainty in artificial intelligence: proceedings of the... conference. Conference on Uncertainty in Artificial Intelligence*, volume 30, page 320. NIH Public Access, 2014.

[60] John T Halloran, Gregor Urban, David Rocke, and Pierre Baldi. Deep semi-supervised learning improves universal peptide identification of shotgun proteomics data. *bioRxiv*, pages 2020–11, 2020.

[61] Roger Higdon, Jason M Hogan, Gerald V Belle, and Eugene Kolker. Randomized sequence databases for tandem mass spectrometry peptide and protein identification. *Omics: a journal of integrative biology*, 9(4):364–379, 2005.

[62] Matthias Hochsmann, Bjorn Voss, and Robert Giegerich. Pure multiple RNA secondary structure alignments: a progressive profile approach. *IEEE/ACM transactions on computational biology and bioinformatics*, 1(1):53–62, 2004.

[63] Michael R Hoopmann, Gregory L Finney, and Michael J MacCoss. High-speed data reduction, feature detection, and MS/MS spectrum quality assessment of shotgun proteomics data sets using high-resolution mass spectrometry. *Analytical chemistry*, 79(15):5620–5632, 2007.

[64] Cendrine Hudelot, Vivek Gowri-Shankar, Howsun Jow, Magnus Rattray, and Paul G Higgs. RNA-based phylogenetic methods: application to mammalian mitochondrial RNA sequences. *Molecular Phylogenetics and Evolution*, 28(2):241–252, 2003.

[65] Zamin Iqbal, Mario Caccamo, Isaac Turner, Paul Flicek, and Gil McVean. De novo assembly and genotyping of variants using colored de Bruijn graphs. *Nature genetics*, 44(2):226, 2012.

[66] Kyowon Jeong, Sangtae Kim, and Nuno Bandeira. False discovery rates in spectral identification. *BMC bioinformatics*, 13(16):S2, 2012.

[67] Yu-Jen Jou, Chia-Der Lin, Chih-Ho Lai, Chao-Hsien Chen, Jung-Yie Kao, Shih-Yin Chen, Ming-Hsui Tsai, Su-Hua Huang, and Cheng-Wen Lin. Proteomic identification of salivary transferrin as a biomarker for early detection of oral cancer. *Analytica chimica acta*, 681(1-2):41–48, 2010.

[68] Howsun Jow, Cendrine Hudelot, Magnus Rattray, and Paul G Higgs. Bayesian phylogenetics using an RNA substitution model applied to early mammalian evolution. *Mol Biol Evol*, 19(9):1591–1601, Sep 2002.

[69] Thomas H Jukes, Charles R Cantor, et al. Evolution of protein molecules. *Mammalian protein metabolism*, 3:21–132, 1969.

[70] Lukas Käll, Jesse D Canterbury, Jason Weston, William S Noble, and Michael J MacCoss. Semi-supervised learning for peptide identification from shotgun proteomics datasets. *Nature methods*, 4(11):923, 2007.

[71] Lukas Käll, John D Storey, Michael J MacCoss, and William S Noble. Assigning significance to peptides identified by tandem mass spectrometry using decoy databases. *Journal of proteome research*, 7(01):29–34, 2008.

[72] Uri Keich, Attila Kertesz-Farkas, and William S Noble. Improved false discovery rate estimation procedure for shotgun proteomics. *Journal of proteome research*, 14(8):3148–3161, 2015.

[73] Uri Keich, Kaipo Tamura, and William S Noble. Averaging strategy to reduce variability in target-decoy estimates of false discovery rate. *Journal of proteome research*, 18(2):585–593, 2018.

[74] Andrew Keller, Alexey I Nesvizhskii, Eugene Kolker, and Ruedi Aebersold. Empirical statistical model to estimate the accuracy of peptide identifications made by MS/MS and database search. *Analytical chemistry*, 74(20):5383–5392, 2002.

[75] Sangtae Kim and Pavel A Pevzner. MS-GF+ makes progress towards a universal database search tool for proteomics. *Nature communications*, 5(1):1–10, 2014.

[76] Thomas Kislinger, Anthony O Gramolini, David H MacLennan, and Andrew Emili. Multidimensional protein identification technology (MudPIT): technical overview of a profiling method optimized for the comprehensive proteomic investigation of normal and diseased heart tissue. *Journal of the American Society for Mass Spectrometry*, 16(8):1207–1220, 2005.

[77] Thomas Kislinger, Khaled Rahman, Dragan Radulovic, Brian Cox, Janet Rossant, and Andrew Emili. PRISM, a generic large scale proteomic investigation strategy for mammals. *Molecular & Cellular Proteomics*, 2(2):96–106, 2003.

[78] Daria S Kostyunina, Simon C Rowan, Nikolai V Pakhomov, Eugene Dillon, Keith D Rochfort, Philip M Cummins, Malachy J O'Rourke, and Paul McLoughlin. Shear stress markedly alters the proteomic response to hypoxia in human pulmonary endothelial cells. *American Journal of Respiratory Cell and Molecular Biology*, (ja), 2023.

[79] Mohamed Lamkanfi, Thirumala-Devi Kanneganti, Petra Van Damme, Tom Vanden Berghe, Isabel Vanoverberghe, Joël Vandekerckhove, Peter Vandenabeele, Kris Gevaert, and Gabriel Núñez. Targeted peptidecentric proteomics reveals caspase-7 as a substrate of the caspase-1 inflammasomes. *Molecular & Cellular Proteomics*, 7(12):2350–2363, 2008.

[80] Eun-Young Lee, Do-Young Choi, Dae-Kyum Kim, Jung-Wook Kim, Jung O Park, Sungjee Kim, Sang-Hyun Kim, Dominic M Desiderio, Yoon-Keun Kim, Kwang-Pyo Kim, et al. Gram-positive bacteria produce membrane vesicles: proteomics-based characterization of Staphylococcus aureus-derived membrane vesicles. *Proteomics*, 9(24):5425–5436, 2009.

[81] Ruiqiang Li, Hongmei Zhu, Jue Ruan, Wubin Qian, Xiaodong Fang, Zhongbin Shi, Yingrui Li, Shengting Li, Gao Shan, Karsten Kristiansen, et al. De novo assembly of human genomes with massively parallel short read sequencing. *Genome research*, 20(2):265–272, 2010.

[82] T-Y Liu and YH Chang. Hydrolysis of proteins with p-toluenesulfonic acid determination of tryptophan. *Journal of Biological Chemistry*, 246(9):2842–2848, 1971.

[83] Xiaowen Liu, Yakov Sirotkin, Yufeng Shen, Gordon Anderson, Yihsuan S Tsai, Ying S Ting, David R Goodlett, Richard D Smith, Vineet Bafna, and Pavel A Pevzner. Protein identification using top-down spectra. *Molecular & cellular proteomics*, pages mcp–M111, 2011.

[84] Bingwen Lu and Ting Chen. A suffix tree approach to the interpretation of tandem mass spectra: applications to peptides of non-specific digestion and post-translational modifications. *Bioinformatics*, 19(suppl_2):ii113–ii121, 2003.

[85] Michael J MacCoss, Christine C Wu, and John R Yates. Probability-based validation of protein identifications using a modified SEQUEST algorithm. *Analytical chemistry*, 74(21):5593–5599, 2002.

[86] Brendan MacLean, Daniela M Tomazela, Nicholas Shulman, Matthew Chambers, Gregory L Finney, Barbara Frewen, Randall Kern, David L Tabb, Daniel C Liebler, and Michael J MacCoss. Skyline: an open source document editor for creating and analyzing targeted proteomics experiments. *Bioinformatics*, 26(7):966–968, 2010.

[87] Dominik Madej, Long Wu, and Henry Lam. Common decoy distributions simplify false discovery rate estimation in shotgun proteomics. *Journal of proteome research*, 2022.

[88] Dattatreya Mellacheruvu, Zachary Wright, Amber L Couzens, Jean-Philippe Lambert, Nicole A St-Denis, Tuo Li, Yana V Miteva, Simon Hauri, Mihaela E Sardiu, Teck Yew Low, et al. The CRAPome: a contaminant repository for affinity purification–mass spectrometry data. *Nature methods*, 10(8):730, 2013.

[89] Roger E Moore, Mary K Young, and Terry D Lee. Qscore: an algorithm for evaluating SEQUEST database search results. *Journal of the American Society for Mass Spectrometry*, 13(4):378–386, 2002.

[90] Johra M Moosa, Shenheng Guan, Michael F Moran, and Bin Ma. Repeat-preserving decoy database for false discovery rate estimation in peptide identification. *Journal of proteome research*, 19(3):1029–1036, 2020.

[91] Johra M Moosa and Bin Ma. Improving peptide identification rate by machine learning with next-ranked peptide spectrum matches. In *18th Conference on Computational Intelligence Methods for Bioinformatics & Biostatistics, CIBB 2023, Padova, Italy*, 2023.

[92] Tobias Müller, Sven Rahmann, Thomas Dandekar, and Matthias Wolf. Accurate and robust phylogeny estimation based on profile distances: a study of the Chlorophyceae (Chlorophyta). *BMC Evolutionary Biology*, 4(1):1–7, 2004.

[93] Tobias Müller, Rainer Spang, and Martin Vingron. Estimating amino acid substitution models: a comparison of Dayhoff's estimator, the resolvent approach and a maximum likelihood method. *Molecular biology and evolution*, 19(1):8–13, 2002.

[94] Tobias Müller and Martin Vingron. Modeling amino acid replacement. *Journal of Computational Biology*, 7(6):761–776, 2000.

[95] Gonzalo Navarro. A guided tour to approximate string matching. *ACM computing surveys (CSUR)*, 33(1):31–88, 2001.

[96] Norman Pavelka, Giulia Rancati, Jin Zhu, William D Bradford, Anita Saraf, Laurence Florens, Brian W Sanderson, Gaye L Hattem, and Rong Li. Aneuploidy confers quantitative proteome changes and phenotypic variation in budding yeast. *Nature*, 468(7321):321, 2010.

[97] Junmin Peng, Joshua E Elias, Carson C Thoreen, Larry J Licklider, and Steven P Gygi. Evaluation of multidimensional chromatography coupled with tandem mass spectrometry (LC/LC- MS/MS) for large-scale protein analysis: the yeast proteome. *Journal of proteome research*, 2(1):43–50, 2003.

[98] Yisu Peng, Shantanu Jain, Yong Fuga Li, Michal Greguš, Alexander R Ivanov, Olga Vitek, and Predrag Radivojac. New mixture models for decoy-free false discovery rate estimation in mass spectrometry proteomics. *Bioinformatics*, 36(Supplement_2):i745–i753, 2020.

[99] David N Perkins, Darryl JC Pappin, David M Creasy, and John S Cottrell. Probability-based protein identification by searching sequence databases using mass spectrometry data. *ELECTROPHORESIS: An International Journal*, 20(18):3551–3567, 1999.

[100] Paola Picotti, Mathieu Clément-Ziza, Henry Lam, David S Campbell, Alexander Schmidt, Eric W Deutsch, Hannes Röst, Zhi Sun, Oliver Rinner, Lukas Reiter, et al. A complete mass-spectrometric map of the yeast proteome applied to quantitative trait analysis. *Nature*, 494(7436):266, 2013.

[101] Sven Rahmann, Tobias Muller, Thomas Dandekar, and Matthias Wolf. Efficient and robust analysis of large phylogenetic datasets. In *Advanced data mining technologies in bioinformatics*, pages 104–117. IGI Global, 2006.

[102] Bernhard Y Renard, Wiebke Timm, Marc Kirchner, Judith AJ Steen, Fred A Hamprecht, and Hanno Steen. Estimating the confidence of peptide identifications without decoy databases. *Analytical chemistry*, 82(11):4314–4318, 2010.

[103] Koos Rooijers, Carolin Kolmeder, Catherine Juste, Joël Doré, Mark De Been, Sjef Boeren, Pilar Galan, Christian Beauvallet, Willem M de Vos, and Peter J Schaap. An iterative workflow for mining the human intestinal metaproteome. *BMC genomics*, 12(1):6, 2011.

[104] Naruya Saitou and Masatoshi Nei. The neighbor-joining method: A new method for reconstructing phylogenetic trees. *Mol. Biol. Evol*, 4(4):406–425, 1987.

[105] Michel Schneider, Lydie Lane, Emmanuel Boutet, Damien Lieberherr, Michael Tognolli, Lydie Bougueleret, and Amos Bairoch. The UniProtKB/Swiss-Prot knowledgebase and its plant proteome annotation program. *Journal of proteomics*, 72(3):567–573, 2009.

[106] Michael Schöniger and Arndt Von Haeseler. A stochastic model for the evolution of autocorrelated DNA sequences. *Molecular phylogenetics and evolution*, 3(3):240–247, 1994.

[107] Jörg Schultz, Stefanie Maisel, Daniel Gerlach, Tobias Müller, and Matthias Wolf. A common core of secondary structure of the internal transcribed spacer 2 (ITS2) throughout the Eukaryota. *Rna*, 11(4):361–364, 2005.

[108] Jörg Schultz, Tobias Müller, Marco Achtziger, Philipp N Seibel, Thomas Dandekar, and Matthias Wolf. The internal transcribed spacer 2 database—a web server for (not only) low level phylogenetic analyses. *Nucleic acids research*, 34(suppl_2):W704–W707, 2006.

[109] Philipp N Seibel, Tobias Müller, Thomas Dandekar, Jörg Schultz, and Matthias Wolf. 4SALE–a tool for synchronous RNA sequence and secondary structure alignment and editing. *BMC bioinformatics*, 7(1):1–7, 2006.

[110] Christian Selig, Matthias Wolf, Tobias Müller, Thomas Dandekar, and Jörg Schultz. The ITS2 Database II: homology modelling RNA structure for molecular systematics. *Nucleic acids research*, 36(suppl_1):D377–D380, 2007.

[111] Yufeng Shen, Nikola Tolić, Kim K Hixson, Samuel O Purvine, Ljiljana Paša-Tolić, Wei-Jun Qian, Joshua N Adkins, Ronald J Moore, and Richard D Smith. Proteome-wide identification of proteins and their modifications with decreased ambiguities

and improved false discovery rates using unique sequence tags. *Analytical chemistry*, 80(6):1871–1882, 2008.

[112] Gloria M Sheynkman, Michael R Shortreed, Brian L Frey, and Lloyd M Smith. Discovery and mass spectrometric analysis of novel splice-junction peptides using RNA-Seq. *Molecular & Cellular Proteomics*, pages mcp–O113, 2013.

[113] David Shteynberg, Eric W Deutsch, Henry Lam, Jimmy K Eng, Zhi Sun, Natalie Tasman, Luis Mendoza, Robert L Moritz, Ruedi Aebersold, and Alexey I Nesvizhskii. iProphet: multi-level integrative analysis of shotgun proteomic data improves peptide and protein identification rates and error estimates. *Molecular & cellular proteomics*, 10(12):M111–007690, 2011.

[114] Sven Siebert and Rolf Backofen. MARNA: multiple alignment and consensus structure prediction of RNAs based on sequence structure comparisons. *Bioinformatics*, 21(16):3352–3359, 2005.

[115] Jennifer A Siepen, Emma-Jayne Keevil, David Knight, and Simon J Hubbard. Prediction of missed cleavage sites in tryptic peptides aids protein identification in proteomics. *Journal of proteome research*, 6(1):399–408, 2007.

[116] Andrew D Smith, Thomas WH Lui, and Elisabeth RM Tillier. Empirical models for substitution in ribosomal RNA. *Molecular Biology and Evolution*, 21(3):419–427, 2004.

[117] Hanno Steen and Matthias Mann. The ABC's (and XYZ's) of peptide sequencing. *Nature reviews Molecular cell biology*, 5(9):699–711, 2004.

[118] Lars Terenius. Endogenous peptides and analgesia. *Annual review of pharmacology and toxicology*, 18(1):189–204, 1978.

[119] Matthew The, Michael J MacCoss, William S Noble, and Lukas Käll. Fast and accurate protein false discovery rates on large-scale proteomics data sets with percolator 3.0. *Journal of the American Society for Mass Spectrometry*, 27:1719–1727, 2016.

[120] John C Tran, Leonid Zamdborg, Dorothy R Ahlf, Ji E Lee, Adam D Catherman, Kenneth R Durbin, Jeremiah D Tipton, Adaikkalam Vellaichamy, John F Kellie, Mingxi Li, et al. Mapping intact protein isoforms in discovery mode using top-down proteomics. *Nature*, 480(7376):254, 2011.

[121] Pieter Vanormelingen, Eberhard Hegewald, Anke Braband, Michaela Kitschke, Thomas Friedl, Koen Sabbe, and Wim Vyverman. The systematics of a small spineless Desmodesmus species, d. costato-granulatus (Sphaeropleales, Chlorophyceae), based on its2 rdna sequence analyses and cell wall morphology 1. *Journal of phycology*, 43(2):378–396, 2007.

[122] Marc Vaudel, Julia M Burkhart, René P Zahedi, Eystein Oveland, Frode S Berven, Albert Sickmann, Lennart Martens, and Harald Barsnes. PeptideShaker enables reanalysis of MS-derived proteomics data sets. *Nature biotechnology*, 33(1):22–24, 2015.

[123] William H Vensel, Frances M Dupont, Stacia Sloane, and Susan B Altenbach. Effect of cleavage enzyme, search algorithm and decoy database on mass spectrometric identification of wheat gluten proteins. *Phytochemistry*, 72(10):1154–1161, 2011.

[124] Juan A Vizcaíno, Eric W Deutsch, Rui Wang, Attila Csordas, Florian Reisinger, Daniel Ríos, José A Dianes, Zhi Sun, Terry Farrah, Nuno Bandeira, et al. ProteomeXchange provides globally coordinated proteomics data submission and dissemination. *Nature biotechnology*, 32(3):223–226, 2014.

[125] David J Wales and Jonathan PK Doye. Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms. *The Journal of Physical Chemistry A*, 101(28):5111–5116, 1997.

[126] Guanghui Wang, Wells W Wu, Zheng Zhang, Shyama Masilamani, and Rong-Fong Shen. Decoy methods for assessing false positives and false discovery rates in shotgun proteomics. *Analytical chemistry*, 81(1):146–159, 2008.

[127] Jian Wang, Philip E Bourne, and Nuno Bandeira. MixGF: spectral probabilities for mixture spectra from more than one peptide. *Molecular & Cellular Proteomics*, 13(12):3688–3697, 2014.

[128] David L Wheeler, Colombe Chappey, Alex E Lash, Detlef D Leipe, Thomas L Madden, Gregory D Schuler, Tatiana A Tatusova, and Barbara A Rapp. Database resources of the national center for biotechnology information. *Nucleic acids research*, 28(1):10–14, 2000.

[129] Matthias Wolf, Marco Achtziger, Jörg Schultz, Thomas Dandekar, and Tobias Müller. Homology modeling revealed more than 20,000 rRNA internal transcribed spacer 2 (ITS2) secondary structures. *RNA*, 11(11):1616–1623, 2005.

[130] Kevin L Yang, Fengchao Yu, Guo C Teo, Vadim Demichev, Markus Ralser, and Alexey I Nesvizhskii. MSBooster: Improving peptide identification rates using deep learning-based features. *bioRxiv*, pages 2022–10, 2022.

[131] Daniel Zerbino and Ewan Birney. Velvet: algorithms for de novo short read assembly using de Bruijn graphs. *Genome research*, pages gr–074492, 2008.

[132] Zheng Zhang, Meghan Burke, Yuri A Mirokhin, Dmitrii V Tchekhovskoi, Sanford P Markey, Wen Yu, Raghothama Chaerkady, Sonja Hess, and Stephen E Stein. Reverse and random decoy methods for false discovery rate estimation in high mass accuracy peptide spectral library searches. *Journal of proteome research*, 17(2):846–857, 2018.