Faculté des bioingénieurs

Comparison between bulk and single-cell mass spectrometry-based proteomics

Is there a need for identification optimization ?

Auteur : Deflandre Guillaume
Promoteur : Gatto Laurent
Co-promoteur : Ghislain Michel
Superviseur : Grégoire Samuel
Lecteurs : Morsomme Pierre, Vanderaa Christophe
Année académique 2023-2024
Mémoire de fin d'études présenté en vue de l'obtention du diplôme de Bioingénieur : chimie et bioindustries

Acknowledgements

I am deeply grateful to my promotor, Laurent Gatto, for accepting me in his lab and allowing me to realize this project. I have been blessed with amazing guidance, be it through our weekly reunions or through his general advice. I have truly learnt to better my skills in a variety of ways I could have only dreamt of before starting my master thesis.

I express my gratitude to my co-promotor, Michel Ghislain, for his advice to contact the CBIO lab which allowed me to realize a project in my field of interest.

I would also like to thank my supervisor Samuel Grégoire, who accompanied me through every step of my thesis. Samuel offered his precious help and answered my every trouble with ease.

I am also grateful to the members of the CBIO lab who have aided me in the context of my project, Julie Devis and Philippe Hauchamps, as well as the other members for welcoming me and including me in the lab activities, lunches and discussions.

Special thanks to my loved ones, my parents and friends who have encouraged me throughout the project.

Thank you, dear reader and member of my jury for taking the time to read this master thesis of mine. I hope to bring you insight and interest in the topics I will discuss in this report as a sign of thanks.

Finally, I would like to thank the faculty of bioengineering, UCLouvain and de Duve institute for offering me the possibility to realize this project within a structured and well-supervized manner.

Abstract

Single-cell proteomics (SCP) has emerged as a powerful tool for elucidating cellular heterogeneity, offering opportunities beyond traditional bulk sample analysis. However, the application of current peptide identification algorithms crafted for bulk samples may lead to false discoveries in SCP. Challenges such as reduced peak counts, lower peak intensities, and degraded signal-to-noise ratios raise the question: do current peptide scoring methods in search engines adequately perform in the context of SCP?

To address these limitations, we explore the effectiveness of database search engines and rescoring tools with the use of Bioconductor packages *PSMatch* and *Spectra*. Rescoring tools take profit of as many mass spectrometry-based features as possible, such as spectral characteristics and retention time models, which can be particularly relevant to mitigate the poor quality of SCP spectra. We used *MS²Rescore* to generate new features, *Mokapot* to rescore the SCP peptides as well as the above-mentioned packages to assess the efficiency of rescoring tools and potentially improve current scoring methods in the context of SCP.

While the need for identification optimization at search engine level is not identified, optimization of rescoring tools is endorsed: our findings demonstrate a notable increase in confidently identified peptides upon rescoring. In addition, we suggest a four-step methodology to evaluate the usefulness of current and new potential features. Finally, our results shed light on the differences between bulk and single-cell samples whilst providing insights that can inform more accurate and reliable data interpretation in the context of SCP.

Contents

List of Figures

List of Tables

List of Acronyms

- API: Application Programming Interface
- CLI: Command Line Interface
- CPU: Central Processing Units
- ECDF: Empirical Cumulative Distribution Function
- FDR: False Discovery Rate
- FN: False Negatives
- FP: False Positives
- HPLC: High-Performance Liquid Chromatography
- LDA: Linear Discriminant Analysis
- MS: Mass Spectrometry
- PP-plot: Probability plot
- PTM: Post-Translational Modifications
- PSM: Peptide Spectrum Match
- SC: Single-Cell
- SCP: Single-Cell Proteomics
- S/N: signal-to-noise ratio
- SVM: Support Vector Machine
- TOF: Time-of-Flight
- TN: True Negatives
- TP: True Positives
- cps: counts per second
- m/z: mass-to-charge ratio

1 Introduction

Breathing, moving, eating, sleeping, recovering, ... All the activities that make living possible depend on the intricate interactions that occur between and within cells. Proteins are the core actors of cellular function, omnipresent and indispensable to any cell. They enable cellular communication and catalyze biochemical reactions. Understanding the identity and function of these proteins is paramount to unraveling the complexities of cellular dynamics and elucidating the mechanisms underlying physiological and pathological processes. This is precisely the goal of proteomics.

1.1 Single-cell proteomics

Traditionally, proteomic analyses have been conducted on bulk samples: samples comprised of millions of cells. Such analyses provide insights into the average protein composition of complex mixtures. However, this average protein abundance across heterogenous cell populations are not representative of each cell in the sample. Indeed, such an approach masks the heterogeneity that exists among individual cells within a population (Boekweg and Payne, 2023). Opposite to bulk, single-cell proteomics (SCP) is meant to explore cellular diversity. Single-cell (SC) resolution is necessary to characterise rare populations showing distinct expression profiles (Figure 1a) or studying continuous expression profiles (Figure 1b) (Vanderaa, 2024).

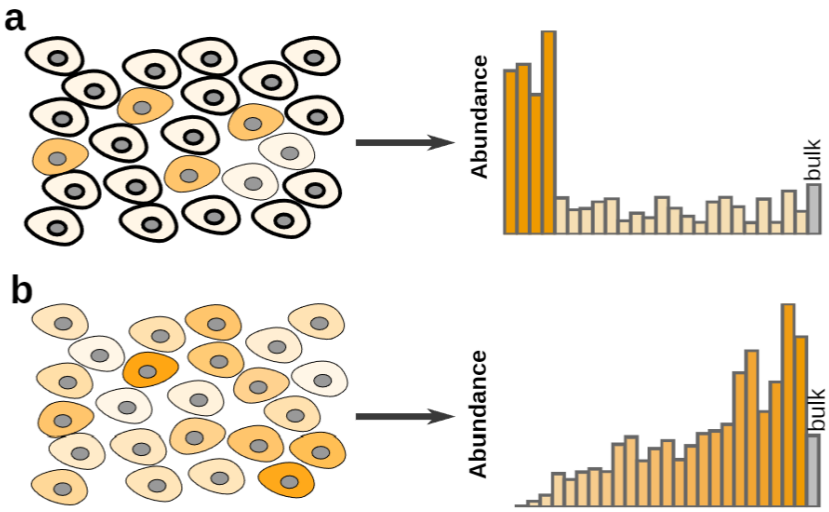


Figure 1: Single-cell analyses provide new biological insights. **a.** Single-cell analyses allow the identification of rare cell populations (dark orange). **b.** Single-cell proteomics enables identification of continuous expression profiles (Vanderaa, 2024).

However, SCP doesn't come without hurdles. Especially the significantly lower quantities of proteins in a SC sample make SCP challenging. Unlike DNA and RNA, which can be replicated in large quantities,

proteins cannot be amplified. The inherent difference in sample size between bulk and SC analyses begs the question: do current peptide and protein identification techniques still perform well in the context of SCP?

1.2 Mass spectrometry-based proteomics

Mass spectrometry (MS) is the current state-of-the-art to comprehensively study proteins. Mass spectrometry is compatible with various sample sizes and provides a comprehensive proteome analysis. Moreover, MS doesn't require prior knowledge on the proteins present in a sample. In comparison, flow cytometry and cell imaging are limited to a predefined number of protein targets and their corresponding antigen-specific antibodies.

Computational MS-based proteomics can be subdivided into two main areas: (i) the identification and quantification of peptides, proteins and post-translational modifications (PTMs) and (ii) downstream analysis (Sinitcyn et al., 2018). This report will focus on the identification of peptides.

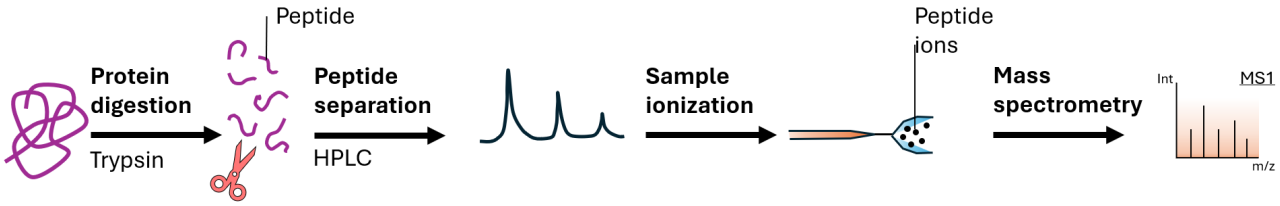


Figure 2: Bottom-up mass spectrometry-based proteomic experiment.

Two common approaches exist in MS-based proteomics: (i) bottom-up and (ii) top-down proteomics. Top-down proteomics focus on one or two proteins at most whereas bottom-up proteomics allow identification of the whole proteome in a sample. This project was focused on the identification of the whole proteome in a sample and thus, for the remainder of this report, we will refer to bottom-up proteomics as MS-based proteomics.

In MS-based proteomics, proteins undergo digestion using one or more enzymes (typically trypsin), thus generating a mixture of peptides to be identified (see Figure 2).

Before injection into the mass spectrometer, peptides are separated by a reversed-phase high-performance liquid chromatography (HPLC) column coupled to the mass spectrometer. The more hydrophobic a peptide is, the longer its retention time will be (Steen and Mann, 2004).

A mass spectrometer has three main components:

1. Ionization source

2. Mass analyzer

3. Detector

Each of these components may vary depending on the mass spectrometer, but they all share the same purpose.

As only ions can be detected by MS, peptides are first charged through ionization. The number of charges and their location on the peptide can vary. Most commonly, tryptic peptides are doubly protonated and thus carry two protons.

The mass analyzer, like its name suggests, measures the mass of a selected ion. The most common method employed in mass analyzers is Time-of-Flight (TOF), which measures the time it takes for ions to travel a known distance. In TOF MS, ions of different masses will reach the detector at different times, allowing for their mass-to-charge (m/z) ratios to be determined. Another widely used mass analyzer is the ion trap, which traps ions in a magnetic or electric field and then measures their mass based on their frequency of oscillation or their time of ejection from the trap.

Once the masses of the ions are determined, the detector records their signal intensities. These data, typically represented as a mass spectrum, provide information on the intensity of the detected ions in counts per second (cps) along the m/z axis. The ions are also referred to as the precursor ions. These ions do not necessarily belong to the proteins of interest as some ions can come from common lab contaminations or from background noise.

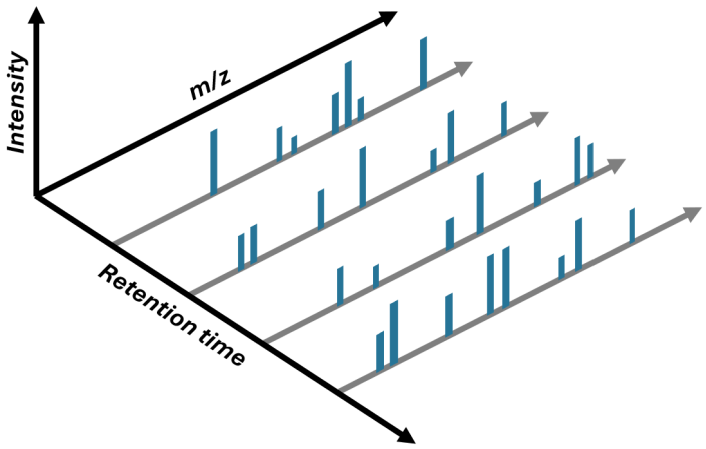


Figure 3: Three dimensional representation of mass spectrometry spectra. Peaks depend on signal intensities, mass-to-charge ratios and peptide retention time.

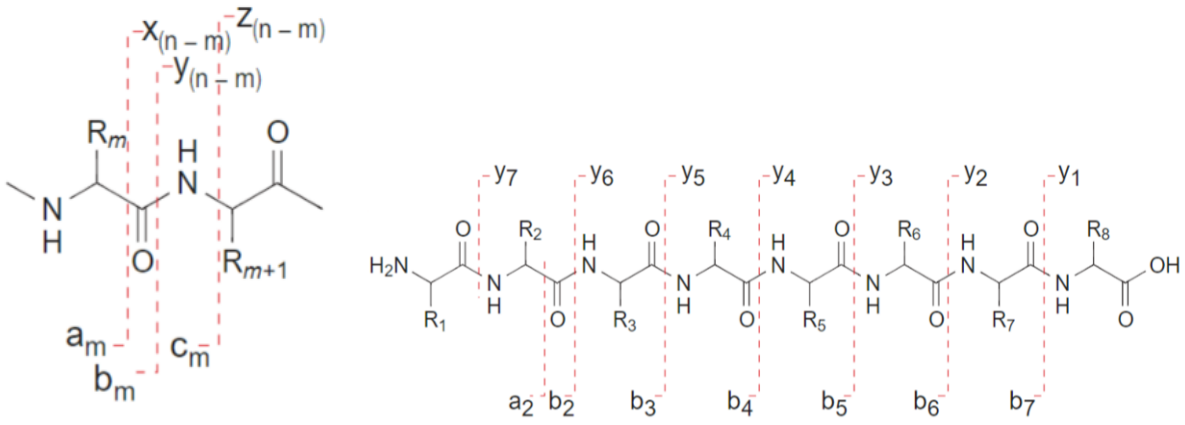
Because there is an additional step of retention time due to the separation by HPLC, MS peaks are represented in three dimensions (Sinitcyn et al., 2018). A spectrum's peaks depend on its retention

time, its m/z value and its intensity as can be seen on Figure 3. In a spectrum, the peak with the highest intensity is called the base peak.

1.2.1 Tandem mass spectrometry

After the masses of the precursor ions are determined by the mass analyzer in the first stage (MS1), selected ions of interest are subjected to further fragmentation within the mass spectrometer. During this process, the selected ions are collided with inert gas molecules, causing them to break apart into smaller fragments. These fragment ions are then analyzed in the second stage of mass spectrometry (MS2). This is called tandem-mass spectrometry and the additional fragmentation generates MS2 spectra. The selected precursor ion is then referred to as the parent ion and can, when not fully fragmented, still be present in the MS2 spectrum.

The fragments resulting from the fragmentation vary in type based on their location of cleavage. The different fragmentation types include: (i) a, b, c-ions at the N-terminal of the peptide and (ii) x, y, z -ions at the C-terminal of the peptide. The types are indexed with a number that represents the number of amino-acids that the fragment contains. Figure 4a illustrates an example of a peptide's labelled fragment types and their corresponding location on the peptide.



(a) a, b, c- and x, y, z-ions

(b) Common fragments observed in a spectrum

Figure 4: Fragmentation types observed in tandem mass spectrometry. The amino-acids side chains are represented by the letter R. Indexation of fragments is based on the number of R groups in that fragment. Commonly, the b- and y-ions are observed. Additionally, when the b_2 fragment is observed, the a_2 fragment is prone to be present too (Steen and Mann, 2004).

Even though six fragment types exist, it is very rare to observe them all. More often than not, only the b-and y-ions are observed (see Figure 4b) (Steen and Mann, 2004).

Because all amino-acids and their masses are known, the sequence of a precursor ion can be identified

through its fragmented peaks and their corresponding MS2 spectrum. Indeed, if all possible fragments were measured, a simple substraction of two consecutive b-ions or y-ions would result in a mass value matching one of the existing amino-acids. Doing this repeatedly until the whole peptide sequence is covered would then suffice to identify said sequence (a.k.a *de novo* peptide sequencing, see Figure 5) (Nesvizhskii, 2007).

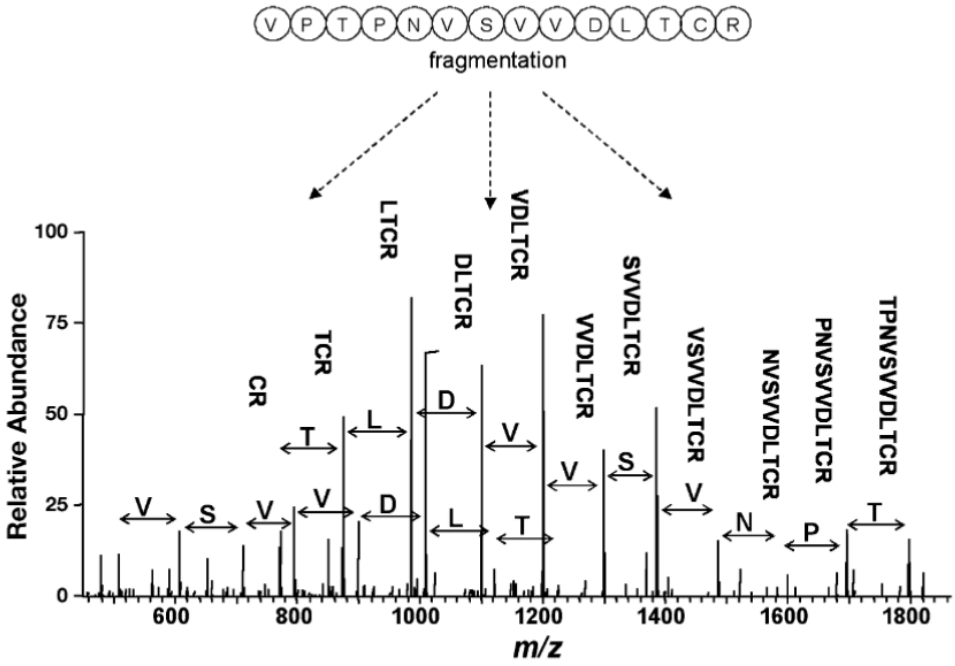


Figure 5: *De novo* peptide sequencing in tandem mass spectrometry. The mass difference of two consecutive b- or y-ion peaks correspond to the mass of an amino acid. Identifying all the amino acids enable the user to identify the sequence of the precursor ion for that spectrum (Nesvizhskii, 2007).

However, due to experimental imperfections, limits in resolution and missing fragment peaks, it becomes difficult to apply *de novo* sequencing. Indeed, one missing peak causes the mass difference between two peaks to match multiple combinations of amino-acids. The more missing peaks there are, the more combinations of masses are possible. Identifying peptides can then no longer be done through *de novo* techniques that are computationally heavy, which is where search engines come into play.

1.3 Search engines

Search engines match experimental spectra to known data, ultimately providing a score of similarity between matched spectra. The known data come from either (i) theoretical fragment masses or (ii) from existing experimental data that has already been matched to peptides. There are thus two main search engine types: (i) sequence database search engines and (ii) spectral library search engines respectively.

In spectral library searches, an experimental spectrum is matched to other already recorded spectra

(Chen et al., 2012). Each of these reference spectra belongs to a specific peptide with an identified sequence. Thus, such search engines cannot identify experimental spectra from peptides that have not already been identified in the spectral library. The peptide coverage is limited to the size of the spectral library.

In sequence database search engines, theoretical fragment masses are compiled by fragmenting a given peptide sequence. A theoretical spectrum is then generated by measuring the possible m/z values of the resulting fragments. The given peptide sequences are provided by digesting proteins from a given database. If a peptide sequence from the experimental sample is not in the given database, none of the theoretical spectra match the experimental spectrum. A sequence database search engine's strength lies in its ability to identify peptides not yet found in spectral libraries thanks to its significantly bigger peptide coverage. The more peptides are in a database, the more peptide coverage there is. However, the larger the provided database is, the more mismatching of spectra by chance are prone to occur.

Throughout this project, a specific sequence database search engine has been used: *Sage* (Lazear, 2023). *Sage* is rather newly developed and claims to be the fastest search engine yet with highly accurate peptide identification coverage. More on the exact parameters and functioning of *Sage* will be mentioned in section 2.2.3 as well as in the sections below. Because the project is limited to *Sage*, the following sections will refer to sequence database search engines as simply search engines. Within its category, different search engines use different spectral matching and scoring methods. To keep this introduction relevant to the project, only methods applicable to *Sage* will be developed.

An overview of *Sage*'s key steps is depicted in Figure 6. The steps will be further developed in the coming sections.

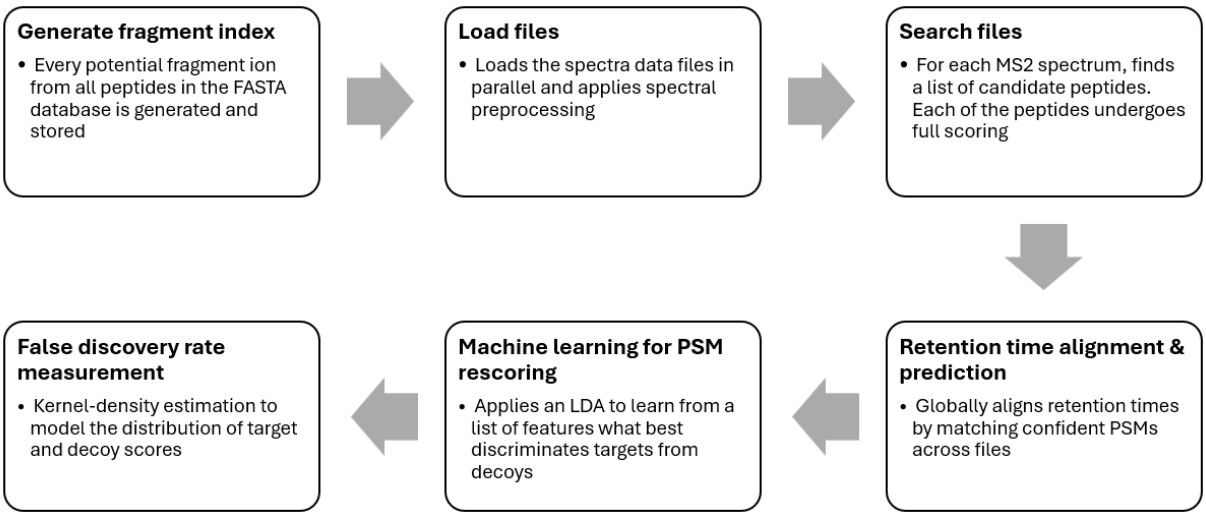


Figure 6: Overview of search engine Sage's key steps and algorithms.

1.3.1 *In silico* repetition of the experiment

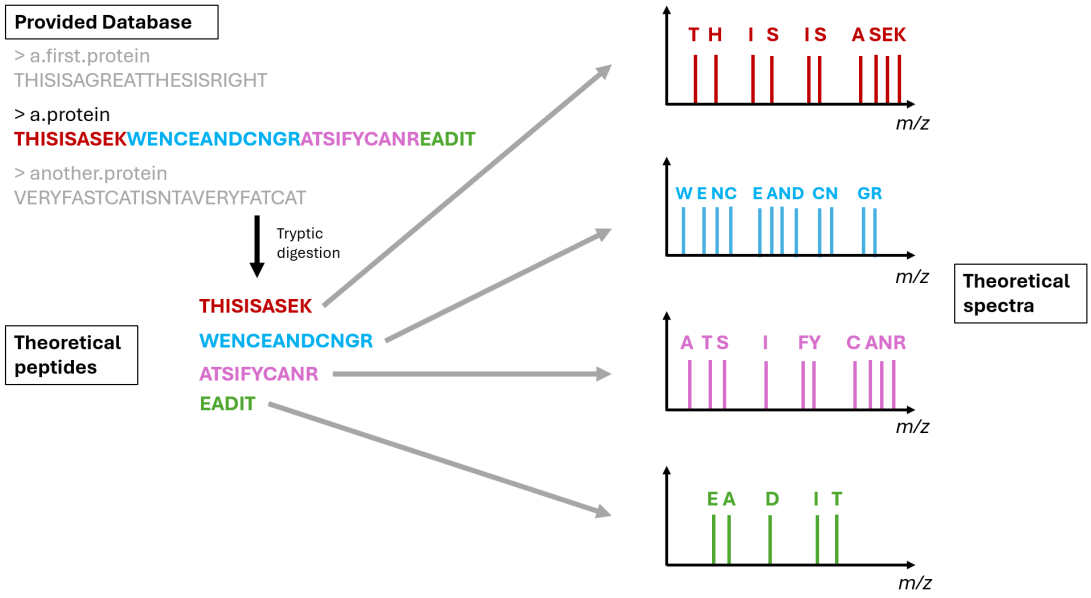


Figure 7: *In silico* repetition of a mass spectrometry experiment. A provided database of proteins of interest is first digested into theoretical peptides. These peptides are further fragmented into their amino acid sequences, generating theoretical spectra.

First, search engines generate every possible theoretical peptide based on the proteins in the provided database (see Figure 7). For instance, if the peptides from the sample of interest come from human proteins, the human proteome is provided to the search engine. In case trypsin was used to digest the initial proteins, the search engine cleaves all the proteins from the proteome into theoretical peptides according to trypsin behaviour (cleaving at the C-terminal side of lysine and arginine residues except when neighbouring proline).

The experiment is thus repeated *in silico*, generating theoretical peptides expected to include matches for every experimental peptide. The corresponding fragment m/z values are then computed based on the generated theoretical peptides, similar to precursor fragmentation in MS.

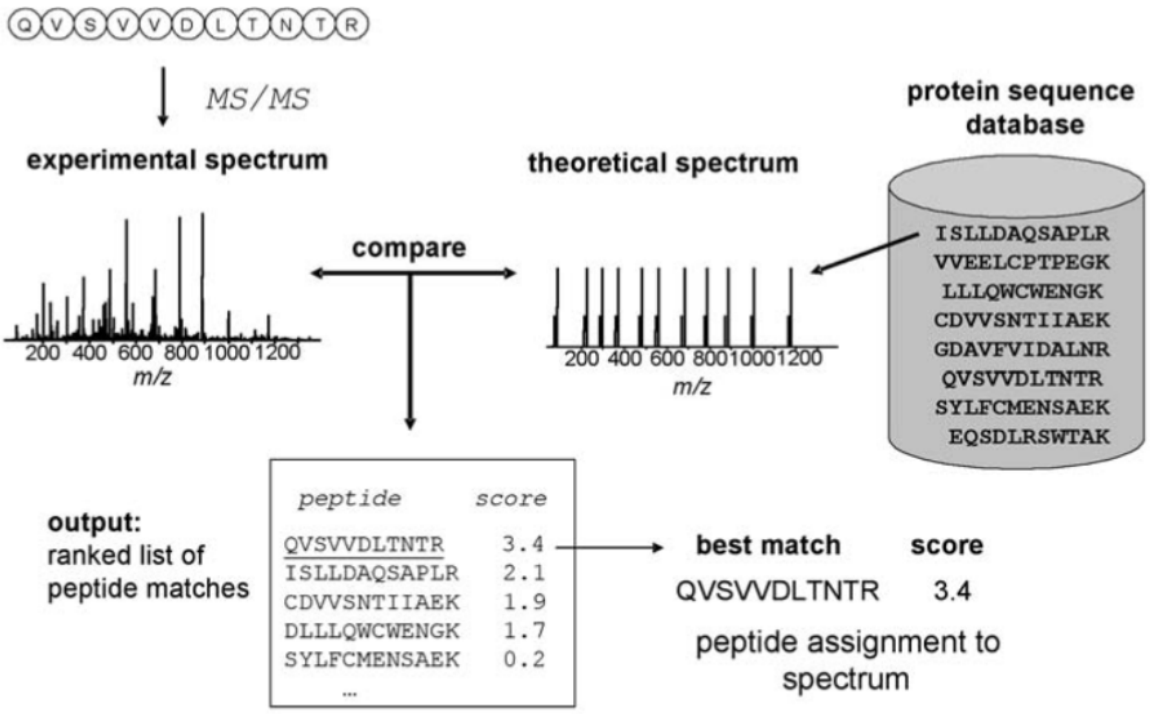


Figure 8: Tandem mass spectrometry database searching. Acquired MS2 spectra are correlated against theoretical spectra. A scoring method is used to measure the degree of similarity between the spectra. Candidate peptides, also called peptide spectrum matches, are ranked according to the computed score, and the best match is selected for further analysis (Nesvizhskii, 2007).

Each generated theoretical spectrum is matched against an experimental spectrum. The matched spectra are then referred to as Peptide Spectrum Matches (PSMs). Every PSM is given a score based on its similarity with the theoretical spectrum (see Figure 8). The scoring method that is used depends on the search engine, but it most often relies on the number of matched fragment peaks (with a shared m/z value) and on their intensities in the experimental spectrum. The generated theoretical spectra strictly serve to identify the number of matched fragments, since they don't have any peak intensities (peaks are either present or absent). The PSMs are then ranked based on their score and the best match (the PSM with the best score) is kept. If the matching is done correctly, this PSM's peptide sequence corresponds to the precursor ion from the experimental MS2 spectrum. The whole process is then repeated on another experimental spectrum.

1.3.2 Scoring methods

Search engines need a scoring method to apply a score to- and rank the generated PSMs. Different scoring methods exist, each with their pros and cons. Because *Sage* uses *X!Tandem*'s hyperscore, it is the one that will be developed here.

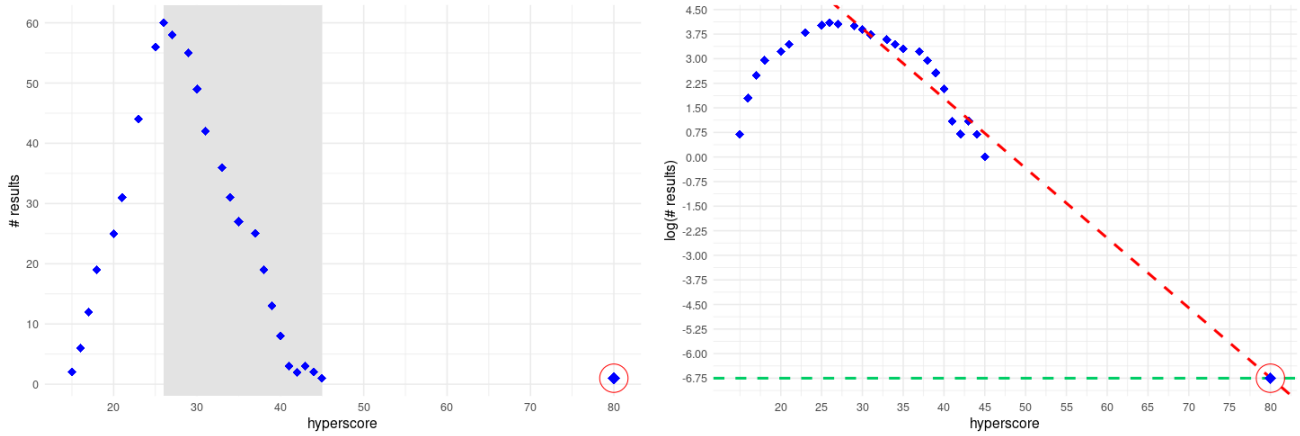
X!Tandem is a program proposed by Craig and Beavis (2003). The scoring method it uses is called the hyperscore and it is the fastest scoring method yet. It is based on Equation (1) where N_b and N_y represent the number of matched fragments (type b and y), I_i the fragment ion intensities from the experimental spectrum and P_i the absence or presence of a peak in the theoretical spectrum ($P_i \in [0; 1]$).

$$HS = (\sum_{i=0}^{n} I_i P_i) N_b! N_y! \tag{1}$$

According to Equation (1), the sum of the intensities for the number of matched peaks is calculated before multiplying it by the factorials of the number of each matched fragment type. A greater number of matched fragments with high intensities thus increases the hyperscore of the PSM. High intensities of non-matched fragments are not penalized as they are not taken into consideration in the hyperscore.

Ranking of PSMs is done based on their score. The first rank is assigned to the PSM with the highest hyperscore and the second rank to the PSM with the next best score etc... The best PSM is thus the PSM that ranks first.

For a given experimental spectrum, there can only be one matching peptide sequence. All the other sequences are wrong. These incorrect matches form a random distribution in the histogram of hyperscores (see Figure 9a). The first ranking PSM's score should then naturally be distinctively higher than the next best score if that PSM is indeed a correct match. Ideally, plotting the histogram of hyperscores should show a clear distancing between the highest score and the next best one. If the distance is too narrow, the significance of the first ranking PSM is lost.



(a) Histogram of hyperscore

(b) Measure of significance of the first ranking PSM

Figure 9: Histogram of peptide spectrum matches (PSMs) based on their attained hyperscore. (a) The x-axis represents the hyperscore and the y-axis represents the amount of PSMs with that same score. The first ranking PSM is circled in red. The grey area represents the descending part of the random distribution that undergoes a linear regression represented in red in figure (b). The expectancy value of the first ranking PSM is measured based on this regression line and equals $e^{-6.75}$, according to the intercept between the green and red lines (adapted from Wojtkiewicz et al. (2013)).

A measure of significance of the first ranking PSM can thus be made. The expectancy value (E-value) of the first ranking PSM is measured based on the regression line of the descending part of the random distribution. The line is represented in red in Figure 9b. The E-value represents the probability that the first ranking PSM belongs to the random distribution. The E-value of the first ranking PSM from the figure equals: E-value $= e^{-6.75} = 0.0012$. A lower E-value indicates a higher confidence in the PSM, as it suggests that the observed hyperscore is unlikely to occur at random (Fenyö and Beavis, 2003).

For every experimental spectrum, the search engine will provide the first ranking PSM and its corresponding peptide sequence. The matches reaching a high score by chance accompanied by high E-values can occur and rank first. These false positives need to be addressed thoroughly through the appriopriate statistical techniques described in the following section.

1.3.3 Target-Decoy competition

A decoy database search is used to estimate how many False Positives (FP) are among the list of first ranking PSMs. Multiple approaches exist in current search engines. Traditionally, a simple competition at PSM level is done, but according to Lin et al. (2022) double competition proves to be most efficient. The terms simple and double competitions refer to a competition of scores that occurs at different stages of the identification process from the search engine. The competition is done through the addition of a concatenated decoy and target database.

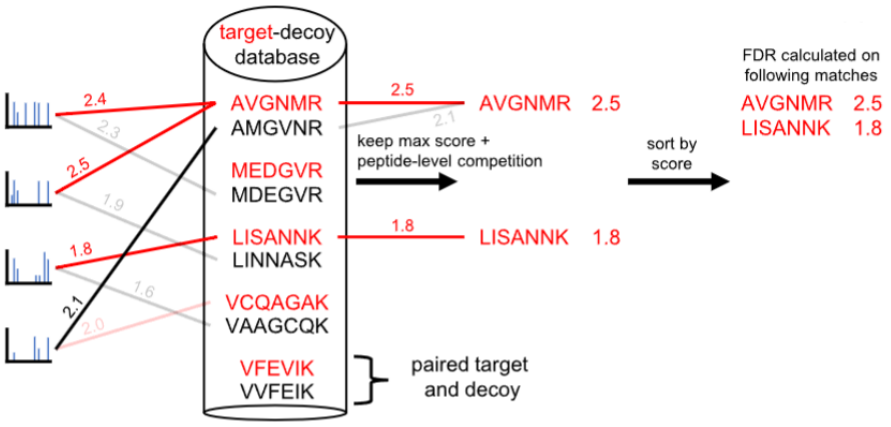


Figure 10: Target-Decoy double competition. The experimental spectra are matched against theoretical peptides coming from a concatenated database of shuffled (decoys) and true target sequences. The scores based on spectral similarity are displayed in red for targets and in black for decoys. A first competition occurs at PSM level and another at peptide level, always keeping the highest score. The final list of peptide spectrum matches is ranked by score for further false discovery rate analysis (Lin et al., 2022).

The decoy database is composed of decoy peptides, i.e. sequences that aren't expected to be found in the proteome under study. These 'made up' sequences are generated by shuffling or reversing the real tryptic peptide sequences. The database search is then made based on a mixture of target and decoy sequences, thus making them compete at PSM level. At his level, only the first ranking PSM is kept. This is shown in the first step (left) of Figure 10.

Naturally, most of the decoys' spectra don't match with any experimental spectra and their given score is thus very low. This means that most of the first ranking PSMs are target PSMs. However, some decoy PSMs do attain the first rank for an experimental spectrum at random.

The double competition makes it possible to clearly differentiate FP from TP. It does this by making the paired peptides (target-decoy pairs) compete against each other a second time at the peptide level (second step (right) of figure 10. Hence, the term double-competition: the first competition occurs at PSM-level and the second competition at peptide-level.

Sorting the final list of first ranking PSMs for each experimental spectrum by score allows for further false discovery analysis. The amount of decoys within the list of first ranking PSMs shows the confidence that target PSMs with a similar score hold. To understand these confidences and where to put a threshold, the concept of False Discovery Rates (FDR) needs to be introduced.

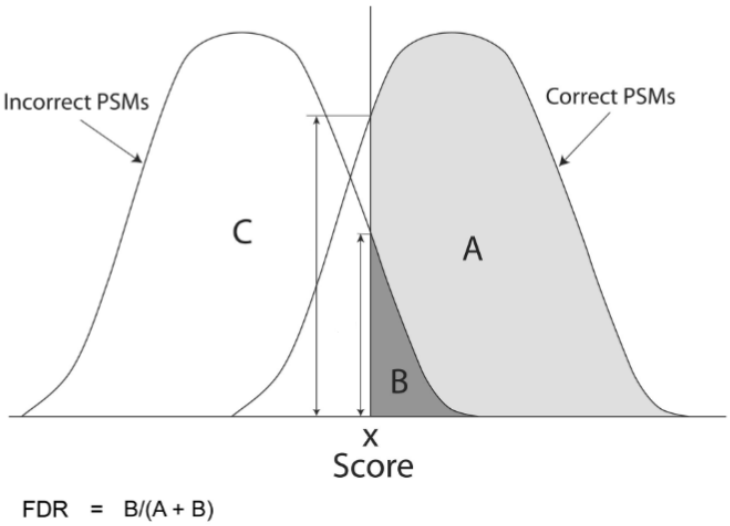


Figure 11: Representation of False Discovery Rates (FDR) in the context of correct and incorrect Peptide Spectrum Matches (PSMs). The FDR is based on a score assigned to each PSM (Käll et al., 2008).

$$FDR = \frac{FP}{FP + TP} \tag{2}$$

With $FP =$ the number of False Positives and $TP =$ the number of True Positives above a score x.

Suppose that a hundred PSMs ranked by score have a decoy at the the 100^{th} position. The FDR of the other 99 target PSMs equals to $\frac{1}{100} = 0.01$ or 1%. This means that among 100 identified PSMs, no more than 1 should be incorectly identified. Generally, a threshold at 1% FDR is preferred as it shows enough confidence in the identified PSMs. The PSMs identified above this threshold should not be considered in the peptide identification since too many FP are found among them (Käll et al., 2008).

This double competition methodology and the calculation of FDR is also implemented in *Sage*. The score that the FDR is based on however, is not the hyperscore but a discriminant score (see 1.3.4).

1.3.4 *Sage*'s discriminant score

A linear discriminant analysis (LDA) is a dimension reduction technique that focuses on maximizing the separability between two or more classes (Izenman, 2008). It a supervised learning algorithm aiming to identify a linear combination of features that optimally segregates classes within a dataset. In the case of search engines in proteomics, the classes are the defined target and decoy PSMs. The idea behind the LDA is to further diminish the false positives and false negatives by assigning a new and better score, essentially rescoring the PSMs. Once the new scores are generated, an increase in amount of

confidently identified PSMs at a same FDR is observed.

To do so, the tendencies and classifications of target and decoy PSMs are identified through the LDA based on a list of features. Adding meaningful features increases the separation between target and decoy PSMs modeled by the LDA. In *Sage*'s LDA, a total of 16 features are provided (listed in Table 1).

Table 1: Features passed on to the LDA in Sage's search engine

Feature	Transformation	Description
charge	none	Reported precursor charge.
fragment_ppm	none	Average ppm (delta mass) for matched fragment ions compared to theoretical ions.
precursor_ppm	none	Difference between experimental mass and calculated mass, reported in parts-per-million.
hyperscore	log(1+x)	X!Tandem hyperscore for the PSM.
delta_next	log(1+x)	Difference between the hyperscore of this candidate and the next best candidate.
isotope_error	none	C13 isotope error.
poisson	log(1+(-x))	Probability of matching exactly N peaks across all scored candidates (Pr(x=k)).
matched_intensity_pct	log(1+x)	Fraction of MS2 intensity explained by matched b- and y-ions.
matched_peaks	none	Number of matched theoretical fragment ions.
longest_b	log(1+x)	Longest consecutive b-ion series.
longest_y	log(1+x)	Longest consecutive y-ion series.
longest_y_pct	none	Longest y-ion series, divided by peptide length (as a fraction).
peptide_len	log(1+x)	Length of the peptide sequence.
missed_cleavages	none	Number of missed cleavages.
aligned_rt	none	Globally aligned retention time.
delta_rt_model	sqrt(clamp(x, 0.001, 0.999)	Difference between predicted and observed retention time.

Because LDA assumes normal distributions of the provided features, *Sage* first applies data transformation. The corresponding modifications are mentioned in Table 1.

Most of the features provide information on the matched spectra themselves (differences in m/z values between matched peaks, number of fragments, etc.) whilst others provide information on the HPLC retention time (`delta_rt_model` & `aligned_rt`). The latter is done through an integrated (simplified) model for predicting peptide retention time. *Sage* performs a linear regression to determine retention coefficients for each amino-acid. According to *Sage*'s documentation, this generally corresponds to a 1-3% boost in PSM identifications.

1.4 Rescoring

The concept of rescoring is not specific to *Sage* and it has long been explored and developed. As is mentioned in the previous section, the ultimate goal of rescoring is to expand the number of confidently identified PSMs and peptides. *Sage* does this through an LDA by using the information that multiple features hold. These features and machine learning techniques are key for rescoring! Some tools strictly focus on feature generation whilst others elaborate machine learning techniques that use those features to increase the PSM counts under a same FDR threshold.

1.4.1 *MS²Rescore*

The rescoring tool that has been resorted to in this project is *MS²Rescore* (Buur et al., 2023). *MS²Rescore* is a modular and user-friendly platform for Artificial Intelligence (AI)-assisted rescoring of peptide identifications. *MS²Rescore* can be presented as a *Python* package which offers the possibility to add one's own generated features for rescoring. It is also accessible through the Command Line Interface (CLI). It can be divided into two sections: (i) the feature generators and (ii) the rescoring engines.

1.4.1.1 Feature generators

The feature generators that are proposed by *MS²Rescore* include *DeepLC* (Bouwmeester et al., 2021) and *MS²PIP* (Degroeve and Martens, 2013).

DeepLC generates six features based to the HPLC retention time from spectra. Retention time prediction systematically improves peptide identification and is crucial to any rescoring method (Klammer et al., 2007; Gessulat et al., 2019; Chen et al., 2023).

MS²PIP is a tool for predicting the intensity of the most important fragment ion signal peaks from a peptide sequence. By doing so, it generates a total of 71 features focused on matched peak intensities.

Every feature generated by aforementioned tools is described and mentioned in Table 7 from the provided appendices.

1.4.1.2 Rescoring engines

The rescoring engine that is used when running *MS²Rescore* is chosen by the user and can either be *Mokapot* (Fondrie and Noble, 2021) or *Percolator* (The et al., 2016). Both engines depict highly correlated results as their algorithms are very similar (see Figure 12).

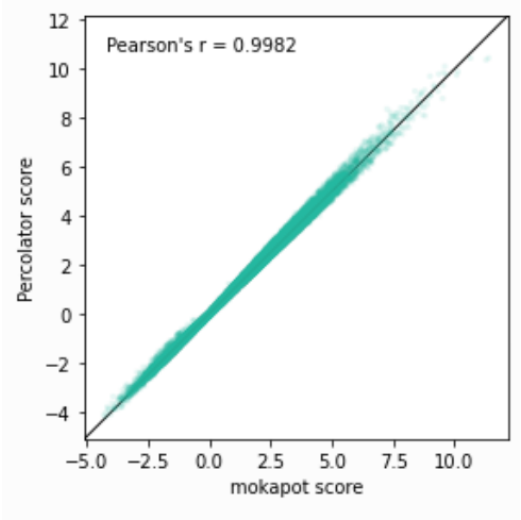


Figure 12: Correlated resulting scores between *Mokapot* and *Percolator* rescoring engines (Fondrie and Noble, 2021).

Both engines use a set of confident PSMs as positive examples and decoy PSMs as negative examples to iteratively train a Support Vector Machine (SVM) that discriminates between them. The method is semi-supervised because the decoys have negative labels but the target labels must be inferred. The engines' algorithm randomly splits the provided list of PSMs into three subsets (the cross-validation bins) and trains three separate SVM classifiers, each trained on two of the three subsets and tested on the remaining subset. The final score for each PSM is calculated based on the scoring vector of the SVM classifier to which that PSM belongs to.

1.5 Objectives

This project aims to explore the inherent differences between single-cell and bulk samples within MS-based proteomics. Specifically, it investigates the effectiveness of current identification methodologies in the context of SCP.

To address this challenge, an evaluation of spectral differences between single-cell and bulk samples is needed. Indeed, due to the lower input samples introduced in the mass spectrometer, single-cell spectra tend to have less annotated peaks and lower peak intensities than bulk spectra (Boekweg et al., 2022).

Moreover, a deeper understanding of feature characteristics utilized for rescoring is essential. To achieve this, a customized approach to rescoring using an LDA was developed. Additionally, the exploration of potential features was made to increase the separation between target and decoy first ranking PSMs. Furthermore, an assessment of feature quality was conducted to determine the significance of individual features.

2 Methods

2.1 Datasets

Two datasets were used in this study. Each dataset comprised bulk, bulk dilutions, and single-cell samples, allowing for comprehensive analysis across different levels of resolution. Each experiment for every sample was done on HeLa cells. HeLa cells are a unique line of epithelial human cells derived from cervical cancer cells taken from Henrietta Lacks in the 1950s (Lucey et al., 2009).

2.1.1 Boekweg dataset

The Boekweg dataset refers to the data used in Boekweg et al. (2022). The files providing the bulk samples (three replicate analyses of HeLa digest) were available in the MassIVE repository MSV000087689. The low input samples, consisting of 2 and 0.2 ng aliquots of HeLa protein digest standard, as well as single HeLa cells are available on the ProteomeXchange (Deutsch et al., 2023) partner repository MassIVE MSV000087524. The four datasets are thus comprised of three to six replicates of a traditional bulk sample, a 10 cell equivalent, a 1 cell equivalent and true SC samples prepared using nanoPOTS (Zhu et al., 2018). In the article, the peptides were identified using MetaMorpheus. The search parameters specified in the article were kept for our analysis. The downloaded *.raw* files were converted to *.mzML* files using *MSConvert.* All samples are label-free and were acquired on an Orbitrap Exploris 480 Thermo Fisher mass spectrometer (except the bulk samples which were analysed by an Orbitrap Fusion).

2.1.2 Liang dataset

The Liang dataset refers to the data provided by Liang et al. (2021) using the autoPOTS workflow. The analyses mentioned in this report made use of the files specific to the MaxQuant search related to the article. The files are comprised of HeLa samples containing 1, 10, 150 and 500 cells. The files containing three replicate analyses of HeLa digest of each sample are available on the PRIDE project PXD021882. Default search parameters were used unless explicitely mentioned in the article. The downloaded *.raw* files were converted to *.mzML* files using *MSConvert.* All samples are label-free and were acquired on an Orbitrap Exploris 480 Thermo Fisher mass spectrometer.

2.1.3 FASTA files

The human proteome provided to sage was downloaded from the UniProt website as a *.fasta* file. The exact *.fasta* file can be found on the UniProt website (UP000005640 - december 2022).

To further extend the robustness of searches and identifications, a concatenated version of the human proteome and the common Repository of Adventitious Proteins (cRAP) (Mellacheruvu et al., 2013) was used. The cRAP consists of a list of proteins commonly found in proteomics experiments that are present either by accident or through unavoidable contamination of protein samples. The list of proteins includes common laboratory proteins (i.e. bovine serum albumin); proteins added by accident through dust or physical contact (i.e. keratin); and proteins used as molecular weight or as mass spectrometry quantitation standards (i.e. horse cytochrome c).

2.2 Tools and packages

The whole process, from the initial *.mzML* replicate sample files of a certain sample size to the final resulting *.tsv* file, is pictured in Figure 13.

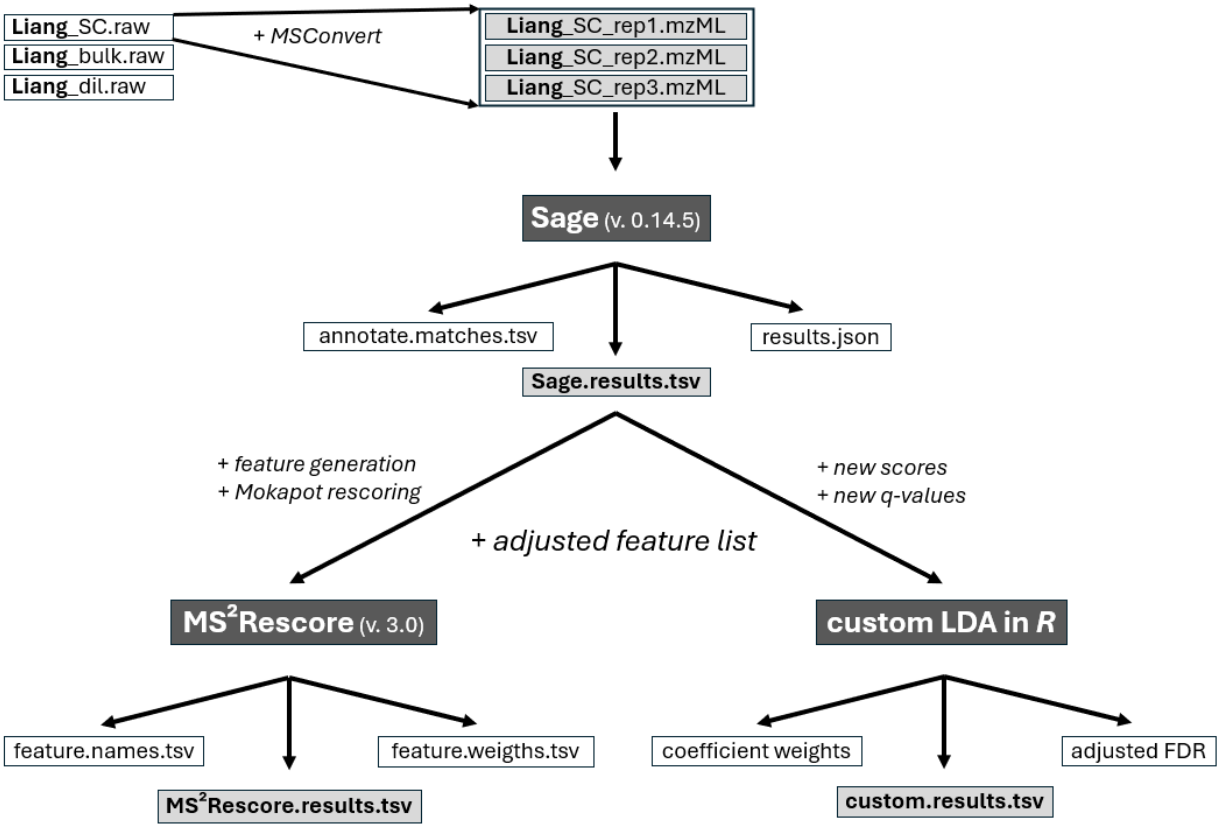


Figure 13: Steps and tools used during the processing of sample files into a resulting tsv file containing the peptide spectrum matches and their corresponding features, scores and false discovery rates.

To generate the resulting files, different tools for different uses have been explored. Below, you will find the main programs that have been used.

2.2.1 *MSConvert*

The program *MSConvert* was used to convert *.raw* files into *.mzML* files.

2.2.2 *R* programming language and packages

Statistical computation, data visualisation and general programming for all analyses mentioned in this report have been done in *R*-4.3.3 (R Core Team, 2024). Although the *R* community is vast and the packages relied on are numerous, a few have been especially useful:

- The *Spectra* package (version 3.18) (Rainer et al., 2022), originating from the Bioconductor project (Huber et al., 2015), allows one to create objects storing MS data. In simple terms, the stored objects follow a strict infrastructure that allows subsetting, processing, visualizing and comparison of spectral data.
- The *PSMatch* package (version 3.18) (Gatto et al., 2022), also originating from the Bioconductor project, offers facilitated handling of PSMs and provides functions to calculate and visualise MS2 fragment ions.
- The *MASS* package (version 7.3-60.0.1) (Venables and Ripley, 2013), although much more global and rich in variety of statistical functions it provides, has mainly been used for its `lda()` and `predict()` functions for LDA and scoring calculations.
- The *tidyr* (version 1.3.1) (Wickham et al., 2024) and *dplyr* version(1.1.4) (Wickham et al., 2023) packages for overall data wrangling.
- The *ggplot2* package (version 3.5.1) (Wickham, 2016) for data visualization.
- The *TargetDecoy* package (version 1.10.0) (Debrie et al., 2024) for evaluating target and decoy distributions.
- The *seqinr* package (version 4.2-36) (Charif and Lobry, 2007) provides functions to read and write *.fasta* formatted files. Through these functions, the concatenated *.fasta* file containing both the human proteome and the common contaminants was generated.

2.2.3 Search engine

Making use of its incomparable running speed and precise peptide identification coverage, *Sage* (version 0.14.5) (Lazear, 2023) has been the only database search engine used throughout this project. *Sage* is a recent open-source search engine written in *rust* (Matsakis and Klock II, 2014).

Every application of *Sage* was done on the Command Line Interface (CLI):

```
sage --annotate-matches --batch-size N configuration.json
```

Depending on the dataset, *Sage*'s input (*configuration.json*) file differed slightly. The different configuration files can be found in appendix C (section 6.3.1). The following chunk serves as an illustration of a typical configuration file:

```
{
  "database": {
    "bucket_size": 8192, ## How many fragments are in each internal mass bucket
    "enzyme": { ## Default is trypsin
      "missed_cleavages": 2, ## Number of missed cleavages to allow
      "min_len": 7, ## Minimum AA length of peptides to search
      "max_len": 30, ## Maximum AA length of peptides to search
      "cleave_at": "KR", ## Amino acids to cleave at
      "restrict": "P" ## Do not cleave if this AA follows the cleavage site
    },
    "fragment_min_mz": 150, ## Minimum mass of fragments to search
    "fragment_max_mz": 2000, ## Maximum mass of fragments to search
    "peptide_min_mass": 500, ## Minimum monoisotopic mass of peptides to fragment
    "peptide_max_mass": 5000, ## Maximum monoisotopic mass of peptides to fragment
    "ion_kinds": ["b", "y"], ## Which fragment ions to generate and search?
    "min_ion_index": 2, ## Do not generate b1/b2/y1/y2 ions for preliminary search
    "max_variable_mods": 3, ## Limit k-combinations of variable modifications
     "static_mods": { ## Static modifications applied
      "C": 57.0215 ## Apply static modification to cysteine
    },
    "variable_mods": { ## Variable modifications applied before static mods
      "M": 15.994,
      "[": 42.0 ## Applied to protein N-terminus
    },
    "decoy_tag": "rev_", ## Apply a tag for decoys generated or recognize tag
    "generate_decoys": true, ## Generate decoys
    "fasta": "../fasta/UP000005640.fasta" ## Path to fasta file
```

```
  },
  "precursor_tol": {
    "ppm": [-20, 20] ## Precursor tolerance in either "ppm" or "da"
  },
  "fragment_tol": {
    "ppm": [-20, 20] ## Fragment tolerance in either "ppm" or "da"
  },
  "isotope_errors": [0, 2], ## C13 isotopic envelope to consider for precursor
  "min_peaks": 15, ## Only process MS2 spectra with at least N peaks
  "max_peaks": 150, ## Take the top N most intense MS2 peaks to search
  "max_fragment_charge": 1, ## Maximum fragment ion charge states to consider
  "min_matched_peaks": 4, ## Minimum # of matched b+y ions used for reporting PSMs
  "predict_rt": true, ## Use retention time prediction model as an feature for LDA
  "output_directory": "/home/gdeflandre/sage_result/PXD021882/cell1/",
  "mzml_paths": [ ## Path to mzML files
  "../data/PXD021882/HeLa_1cell_E12.mzML",
  "../data/PXD021882/HeLa_1cell_F8.mzML",
  "../data/PXD021882/HeLa_1cell_E14.mzML"]
```

All resulting *.tsv* files include only the first ranking PSMs, containing both `targets` and `decoys`, and no FDR filters were applied before any analysis was launched.

2.2.4 Rescoring

All *mzML* files used to run *Sage* were also used to run *MS²Rescore* (version 3.0) (Buur et al., 2023). Calling *MS²Rescore* was done on the CLI:

```
ms2rescore -p N --psm-file sage.results.tsv -s /path/to/mzML --psm-file-type 'sage'
```

2.2.4.1 Feature generators

Both *MS²PIP* and *DeepLC* were applied on the input files, generating their respective features (see Table 7 from appendix B).

Every *MS²Rescore* run has been done on a *sage.results.tsv* file. *MS²Rescore* retains only the specific features mentioned in Table 1 in that file's content, while adding newly generated features to the file

before passing them on to the rescoring engine. Additional custom features were added one at a time (see section 2.6).

2.2.4.2 Rescoring engines

By default, features are passed on to *Mokapot* for rescoring. Since *Percolator* and *Mokapot* depict highly correlated results, with *Mokapot* claiming to perform slightly better when applied on SC data, all runs were performed with *Mokapot.* Because the seed in the Support Vector Machines (SVM) of *Mokapot* isn't fixed, replicate runs don't give exactly the same results. To counterbalance the variance in resulting files, each rescoring analysis was repeated five times for better statistical robustness. The *feature.names.tsv* and *feature.weights.tsv* files resulting from *Mokapot* were used to assess feature importance as described in section 2.5.

2.3 Comparison between bulk and single-cell proteomics data

Assessing the basic differences between bulk and SC proteomics data was done through different approaches.

Because only confident, target PSMs are of interest, a systematic filtering of the provided files was done unless the analysis required to keep all PSMs. In the case of assessing sequence identification differences, only target PSMs below 1% FDR were kept. The FDR was applied on the `spectrum_q` column (the FDR at PSM level) in *Sage*'s resulting *.tsv* file. The distribution of PSMs among the different groups was visualized using an UpSet plot (Gehlenborg, 2019).

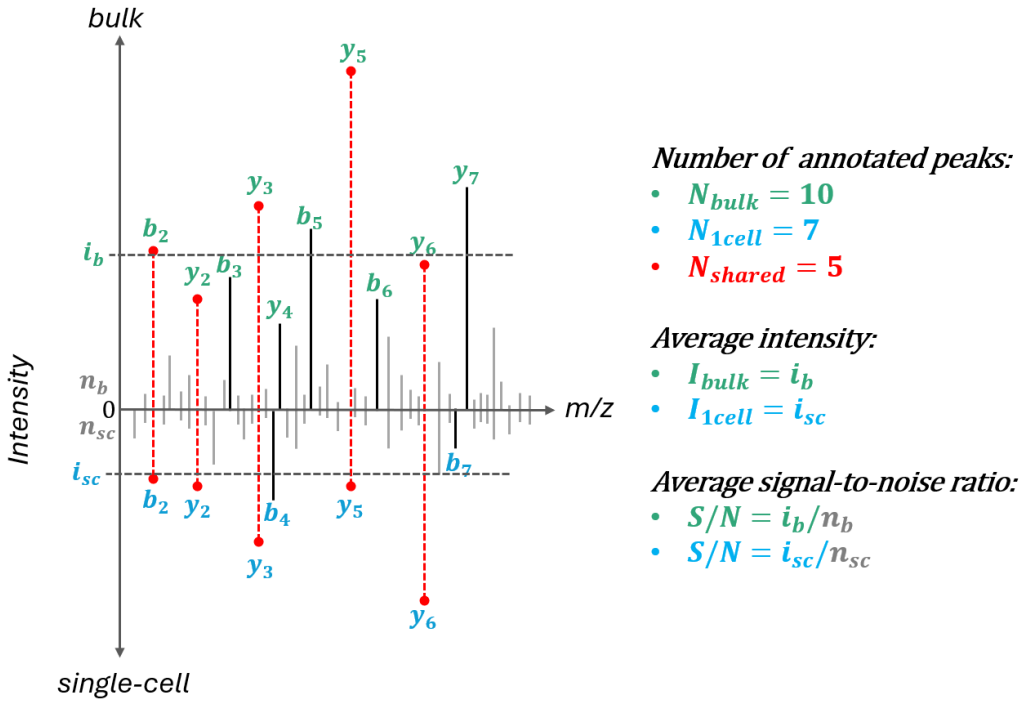
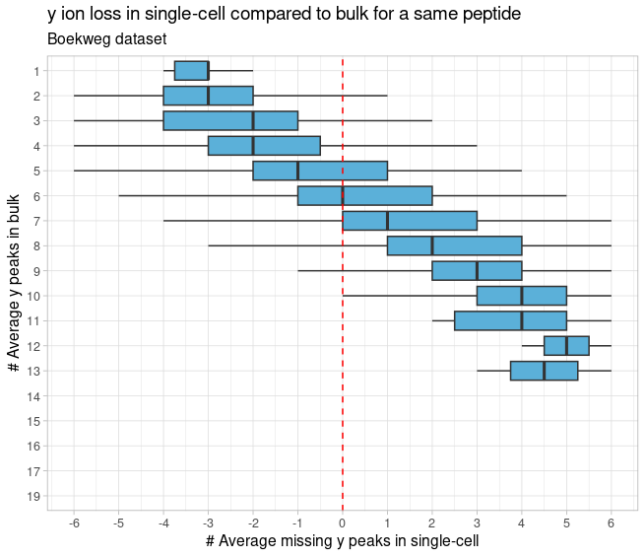


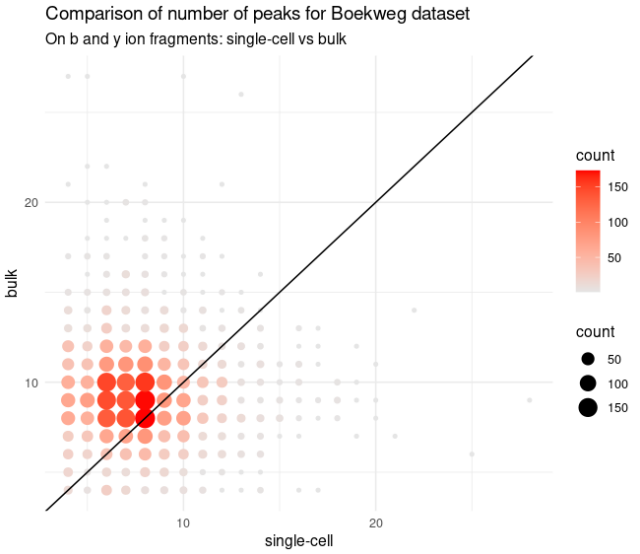
Figure 14: Comparison of spectral characteristics from single-cell and bulk spectra.

Comparing spectral characteristics was done using the *Spectra* package. The spectra were generated for multiple peptide sequences (without modifications) shared between bulk and SC data. For the generation of spectra, the same filtering methods mentioned above were applied. As pictured in Figure 14, spectral characteristics include:

- The number of annotated peaks
- The average intensity
- The average signal-to-noise ratio



(a) Reproduced graphic from the Boekweg article

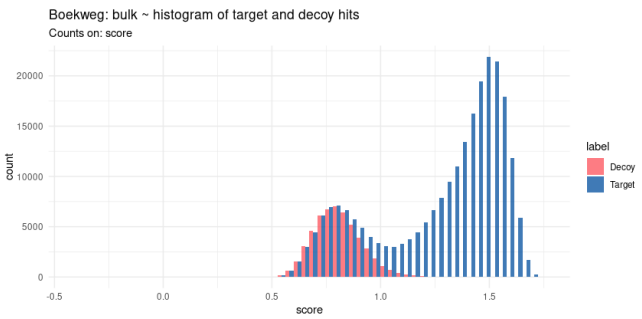


(b) Novel approach to identify fragment losses

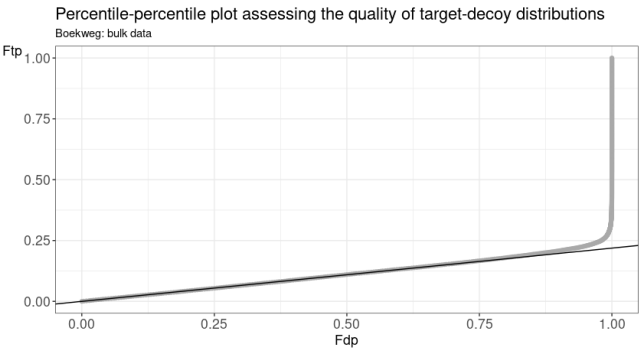
Figure 15: Evaluation on annotated fragment loss in single-cell compared to bulk spectra for shared sequences.

To assess the evolution on the number of annotated peaks, we reproduced a figure from Boekweg et al. (2022) (Figure 15a). The graph in question shows that the losses in y-ions for shared sequences between bulk and SC spectra (x-axis) get more prominent the more peaks are present in the bulk spectrum (y-axis). Because the article only mentions the loss in number of y-ions, an additional approach to identify the loss in both b-and y-ion annotated peaks was done. Here, this visualization was repeated in a more straight-forward manner, depicting the number of annotated peaks on both axes (Figure 15b). More on this decision and why it is important will be discussed in section 4.

The evaluation of target-decoy distributions was done through the use of percentile-percentile plots (PP-plots) from the *TargetDecoy* package (Debrie et al., 2024).



(a) Distributions of PSMs based on their score



(b) PP-plot assessing the deviation of distributions

Figure 16: Quality assesment of target-decoy distributions for Boekweg: bulk dataset.

The PP-plot from Figure 16b reflects the quality of the target-decoy separability in Figure 16a. Percentile-percentile plots are used to determine how well one dataset fits the probability distribution of another dataset. In this case, the PP-plot displays the empirical cumulative distribution (ECDF) from the target distribution in function that of the decoy distribution. Both axes display the probability distribution functions for the decoy (fdp) and target (ftp) distributions. Percentile-percentile plots have the property that they show a straight line through the origin if and only if both distributions are equivalent. Any deviation from this straight line indicates that the distributions differ (Meloun and Militký, 2011).

2.4 Custom Linear Discriminant Analysis and FDR

To apply rescoring with a customized list of features, we ran an LDA in *R*. Before running an LDA, the features were transformed if necessary. The transformations were kept as they were from *Sage*'s LDA (see Table 1).

The resulting LDA scores for each PSM were added to the original *.tsv* file. The PSMs were then ordered by descending values of scores and a new `spectrum_q` value was measured based on the ordered scores. The FDRs were calculated in the same manner as *Sage*'s FDRs (from highest to lowest score, stopping at each PSM to measure its corresponding FDR). An arbitrary value of `8.145312e-05` was attributed to all target PSMs up until the first decoy PSM was hit (following *Sage*'s methodology).

The computed custom LDA allows for the addition of new features as well as removal of any number of initial features. By default, the custom LDAs were done on all of *Sage*'s initial features. To assess feature quality, the LDA was done on all-but-one feature lists: leaving one different feature out each time. This method allowed for measuring the importance of a feature (referred to as feature quality) based on the resulting amount of confidently identified PSMs at a given FDR. A greater loss in PSMs would mean a greater importance of the removed feature. Each analysis was done on both SC and bulk data from the two different datasets.

Unless features were added to the initial *.tsv* file and new scores were attributed to the PSMs, *MS²Rescore* was systematically run with the original `sage_discriminant_score`, as computed by *Sage*.

2.5 Feature quality assessment

The importance of a feature was further measured in accordance to a suggested four-step methodology. The four measurements are to be judged in their integrity, as a whole, rather than independently. The four methods are proposed in no particular order.

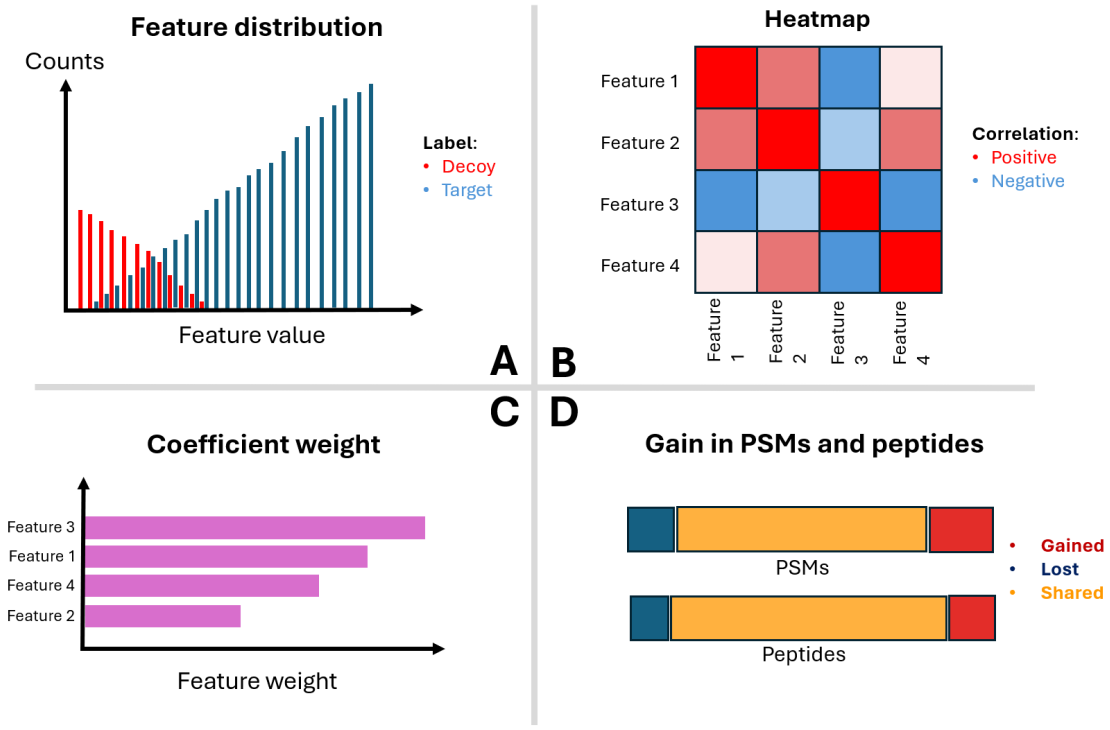


Figure 17: Suggested four-step methodology for feature quality assessment.

- **Clear absence or presence of target-decoy discrimination (Figure 17A)**

The first step consists in observing trends or obvious classification of target and decoy PSM counts in function of their corresponding feature value. Noticeable classifications would then correspond to features of higher importance.

- **Heatmap of correlations (Figure 17B)**

The heatmap of correlations were computed using the *ComplexHeatmap* package (Gu et al., 2016). A traditional three-color shade was used: red being positively correlated, blue being negatively correlated and white being completely uncorrelated. The features were first modified according to *Sage*'s documentation.

- **LDA coefficients' weights (Figure 17C)**

The custom LDA's coefficients correspond to the weights of each feature in explaining target-decoy separation. The absolute value of each coefficient indicates the importance of the corresponding feature in discriminating between targets and decoys. The sign of the coefficients indicates the direction of the relationship between the feature and the likelihood of belonging to a particular class.

- **Absolute gain in confidently identified PSMs and peptide sequences (Figure 17D)**

The last step in this suggested methodology refers to the absolute gains in confidently identified target PSMs and peptide sequences acquired after addition of a feature. In order to precisely understand the changes in PSM and peptide counts, a subdivision corresponding to the type of activity was made: PSMs and peptide sequences can either be retained, lost or added to their original file. To measure these differences, only the target PSMs filtered at 1% FDR were taken into account.

2.6 Addition of potential features

Potential new features were discussed with Dr. Didier Vertommen (MASSPROT core facility manager, de Duve institute).

- **Ratio of parent ion intensity over base peak intensity in MS2 spectra**

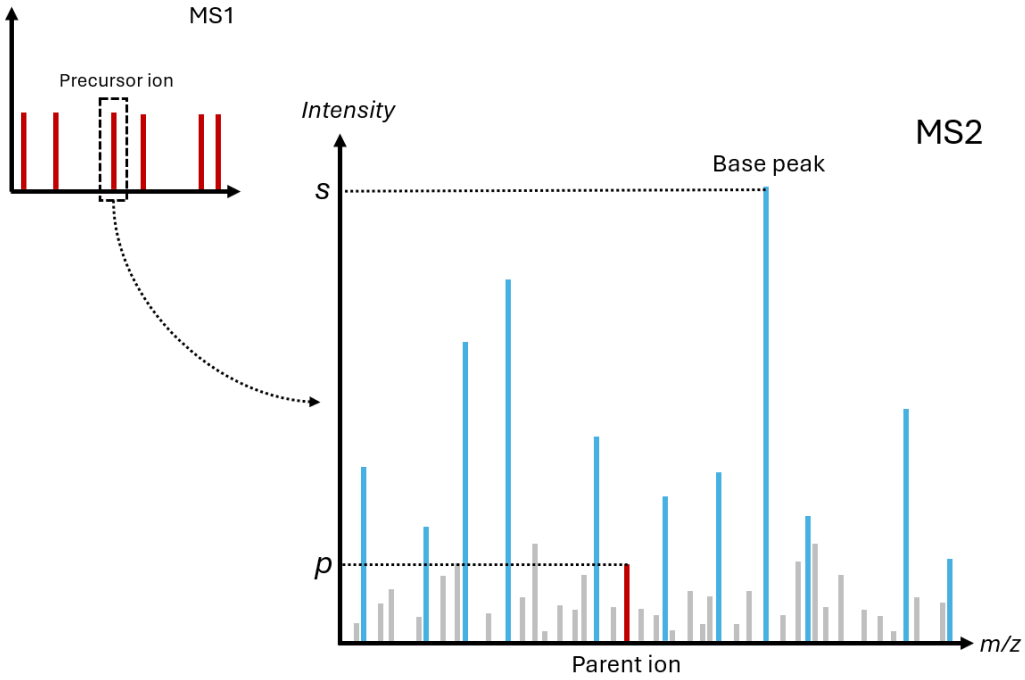


Figure 18: Presence of the parent ion in an MS2 spectrum.

This feature was computed by taking the sum of intensities of experimental m/z peaks within 5 ppm of the theoretical parent ion m/z. For every PSM, the ratio of this resulting intensity over the base peak intensity was measured (p/s according to Figure 18). The value of this ratio is then assigned to the PSM. A value of 0 corresponds to a lack of parent ion in the MS2 spectrum.

- **Peak symmetry around parent ion in MS2 spectra**

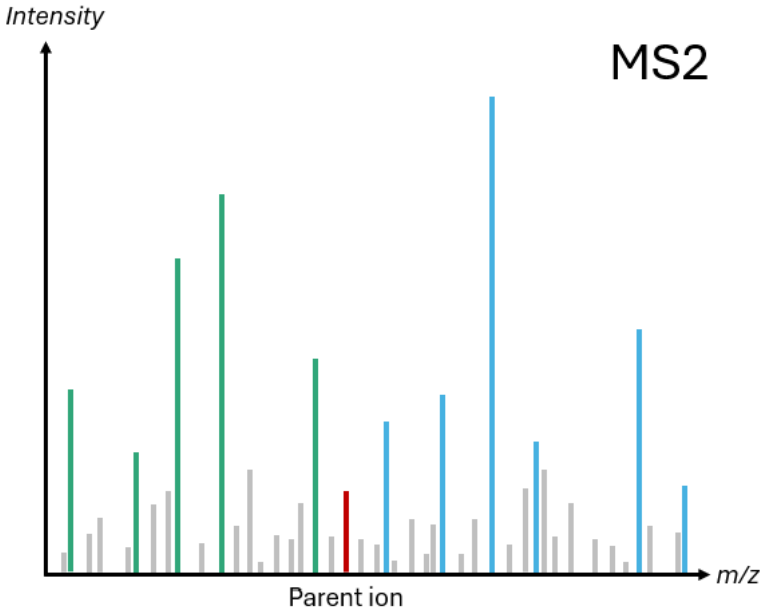


Figure 19: Symmetry of annotated peaks around the parent ion in an MS2 spectrum.

This feature was computed by taking the absolute value of a substraction of the number of annotated peaks right to the parent ion with the number of annotated peaks left to the parent ion :

$$\text{Symmetry} = |N_{\text{peaks above parent ion m/z}} - N_{\text{peaks below parent ion m/z}}| \tag{3}$$

Because only a small proportion of PSMs had a reported precursor charge higher than 2, in which case the symmetry around the parent ion theoretically shouldn't be equal, these were ignored altogether. Although their values were not representative of their spectral symmetry, their small proportion made little impact on the LDA.

Both new features were computed and added to the initial *.tsv* file, generating two different tables consisting of the initial features and the newly added feature. Each resulting file underwent the feature quality assessment methodology and underwent rescoring through *MS²Rescore*. After rescoring, filtering at 1% FDR and target PSMs is done.

An additional "random" negative control was also added. This random feature was generated by providing random deviates between 0 and 1 and should not in any way have an impact on the rescoring results.

3 Results

3.1 Comparison between single-cell and bulk proteomic data

In this section, the identification and spectral differences between bulk and SC data are analyzed. The objective is to confirm the differences identified in the literature (Boekweg et al., 2022). A more in-depth analysis is made if any conclusion deviates from these differences.

3.1.1 Basic data differences

Table 2 shows how the variations in SC and bulk proteomics stand out in the amount of confidently identified PSMs. For SC files, an average of 5000 target PSMs are identified whereas this number is as high as 80000 target PSMs for bulk files. Proportionally, Boekweg's dataset identifies more PSMs at a 1% FDR than Liang's dataset.

Table 2: General overview of a basic data analysis for different datasets

dataset	type	label	PSMs	sequences	proteins	PSMs 1% FDR	sequences 1% FDR	proteins 1% FDR
Liang	single-cell	Target	5610	4492	6266	2145	1388	583
Boekweg	single-cell	Target	4424	3758	3158	3691	3176	1686
Boekweg	bulk	Target	87295	44958	15978	64819	27494	6507
Liang	500 cells	Target	73875	52918	16580	53122	34713	6628
Liang	single-cell	Decoy	3360	3081	NA	21	19	NA
Boekweg	single-cell	Decoy	520	477	NA	37	34	NA
Boekweg	bulk	Decoy	19651	17510	NA	648	583	NA
Liang	500 cells	Decoy	19471	18315	NA	531	509	NA

3.1.2 Shared sequences

The number of sequences found in each sample and how these sequences are shared among samples is shown in Figure 20. The UpSet plot illustrates the evolution of shared sequences between bulk, its diluted samples, and actual SC data filtered at 1% FDR.

Expectedly, half the sequences found in bulk were not found in lower input samples, whereas most of SC's sequences are shared with the other experiments. Do keep in mind that in SC analyses, the quantity of a specific peptide may not always be sufficient to reach the detection threshold. It is likely that sequences are missed due to this technical limitation.

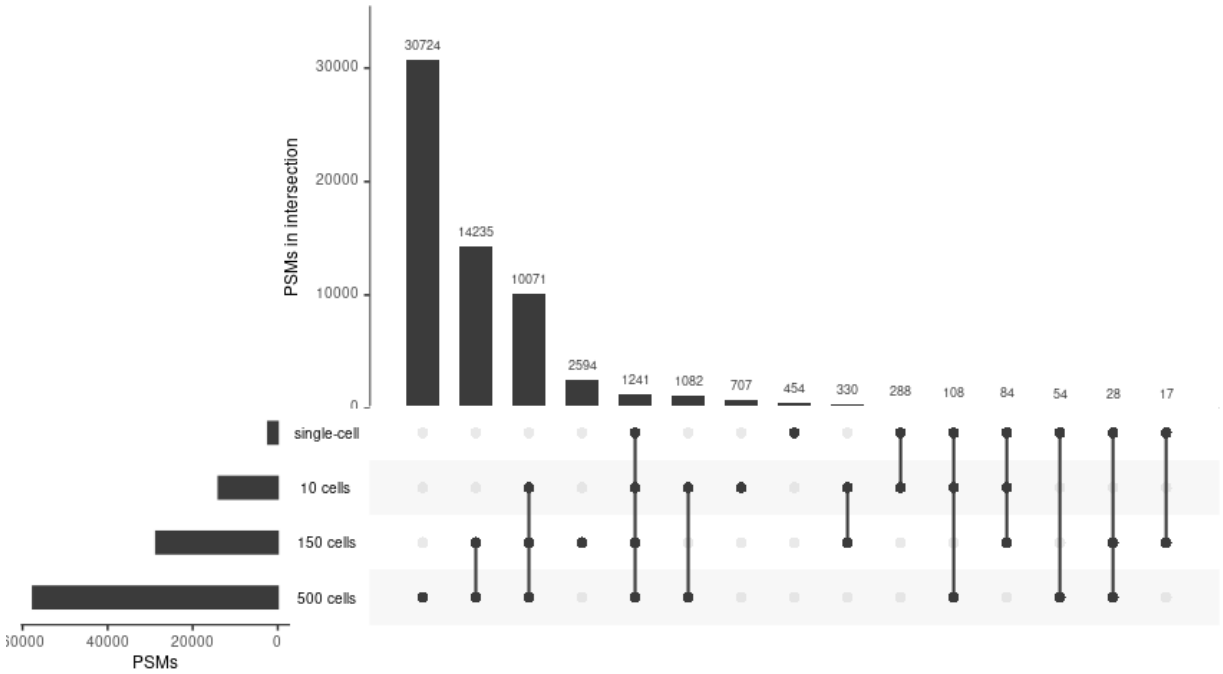


Figure 20: Shared sequences between samples with different numbers of cells from the Liang dataset. Data have been filtered at 1% FDR and show targeted PSMs only.

A gradual trend can be seen when looking at the number of unique sequences identified between all experiments: bulk samples identify 30724 sequences in addition to the 25500 sequences shared with the other samples. As the number of cells decrease, the number of identified sequences also decrease. This gradual trend can be explained by the sheer number of peptides introduced in the MS.

Although these sequences were filtered at 1% FDR, a fifth of the total number of SC sequences are uniquely identified: 453 sequences can only be found in SC samples and not in the other experiments. This is a high number considering bulk data should in fact include all possible sequences from any similar SC analysis.

3.1.3 Comparison of spectral features

As search engines are based on spectral features, it is crucial to understand the characteristics and spectral differences of both bulk and SC spectra. Figure 21 shows a comparison of the spectra obtained for a same peptide identified at a 1% FDR shared between bulk and SC. The observations made in this section are general principles within both datasets.

Generally speaking, bulk spectra contain more peaks than SC spectra do. The blue peaks in the middle panel illustrate the shared peaks within a 20 ppm tolerance. The spectrum on top belongs to bulk data and the spectrum on the bottom to SC data. Even though most of the annotated peaks are

shared between bulk and SC, there remains a lack of annotated peaks in SC spectra. However, what stands out the most is the clear difference in annotated peak intensities between both spectra. For bulk, intensities range up to 80.000 counts per second (cps) whereas for SC, they only go up to 2500 cps for non-annotated peaks and 1000 cps for annotated peaks. Moreover, the signal-to-noise (S/N) ratio worsens in SC spectra: annotated peaks barely outrange the average noise in comparison to the average bulk S/N range. Additionally, a difference in fragment intensities can be assessed. Indeed, the relation between fragment intensities in bulk differ from those in SC. For instance, the b_2-ion in the bulk spectrum is also the most intensely annotated peak, whereas for the SC spectrum, it is the y_8-ion.

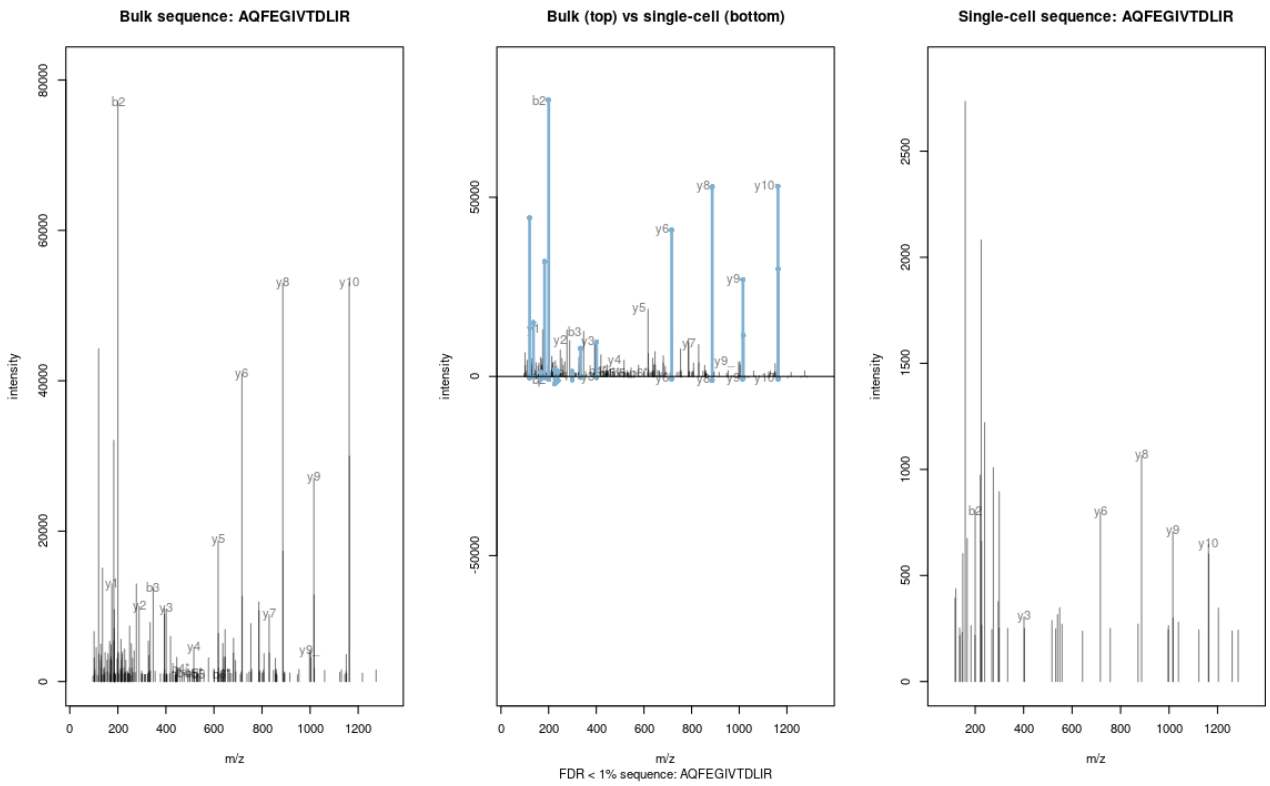
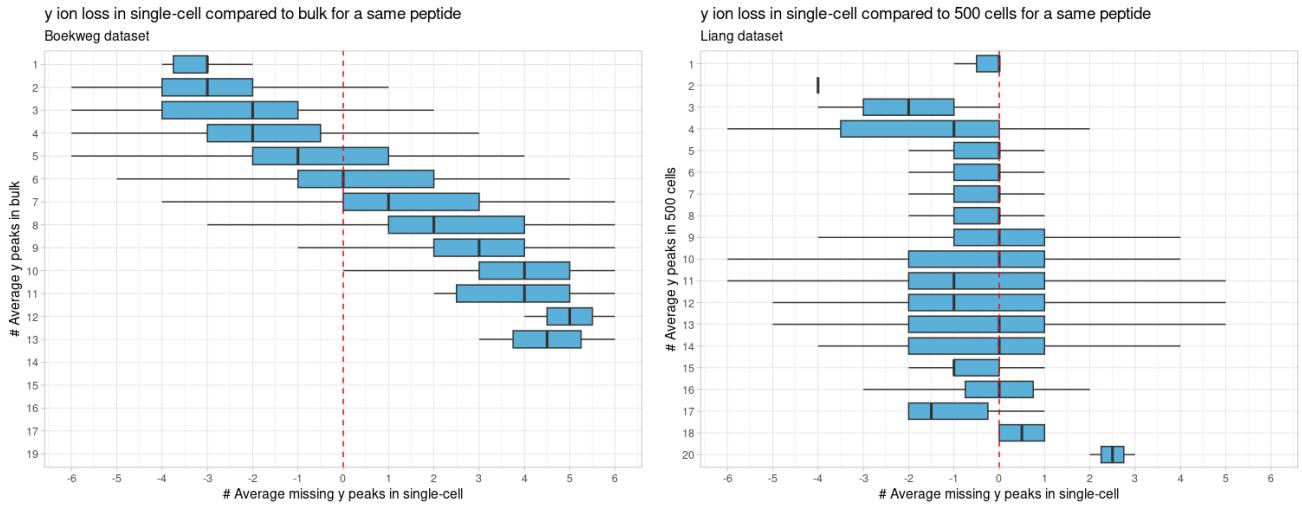


Figure 21: Side by side comparison of spectra for a shared peptide sequence confidently identified for both bulk (left) and single-cell (right) data. A mirrored plot of the spectra (middle) illustrates the shared peaks between both samples (in blue).

These spectral differences are important as fragment intensities are often used for rescoring through machine learning models which have been trained on bulk data. Moreover, search engines' scoring models are mainly based on peak intensities and the number of matched peaks between experimental and theoretical spectra. In this sense, SC MS2 fragments might not be adapted to said models.

3.1.4 Annotated peak loss

To dive deeper into the peaks data from bulk and SC, we first sought out to replicate the peak loss analysis from Boekweg et al. (2022) and transpose it to the Liang dataset. The graphics in the article were meant to asses y-ion peak loss in SC compared to bulk (see Figure 22).



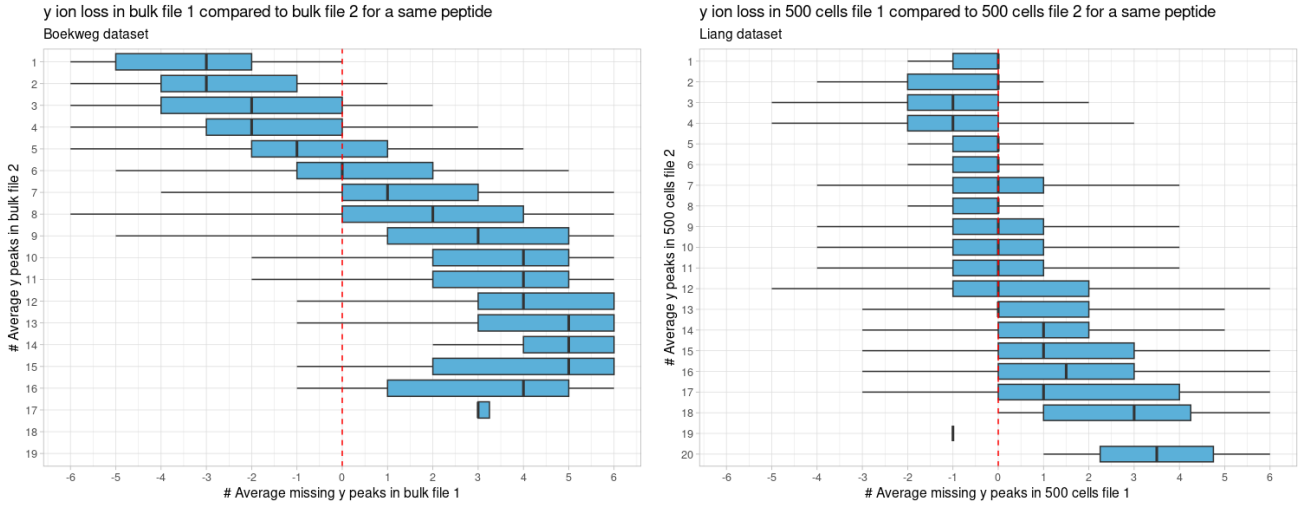
(a) For Boekweg dataset

(b) For Liang dataset

Figure 22: Reproduction of graphics from the Boekweg article: y-ion loss in single-cell compared to bulk spectra of shared sequences at 1% FDR. The y-axis shows the number of y-ions in the bulk spectrum and the x-axis shows if any peaks are lost or gained in single-cell for a spectrum of that same peptide. A positive value corresponds to peak loss and a negative value corresponds to peak gain.

For the Boekweg dataset, we successfully replicated their results: losses in y-ions for shared sequences between bulk and SC spectra get more prominent the more y-ion peaks are present in the bulk spectrum. This is not the case for the Liang dataset, which shows no differences between the number of peaks, if not a slight peak gain in SC.

To adress the difference between Figures 22a and 22b, similar analyses between two replicates with the same cell number were performed (see Figures 23 and 24). It is expected to have no distinct y-ion peak losses between replicates.

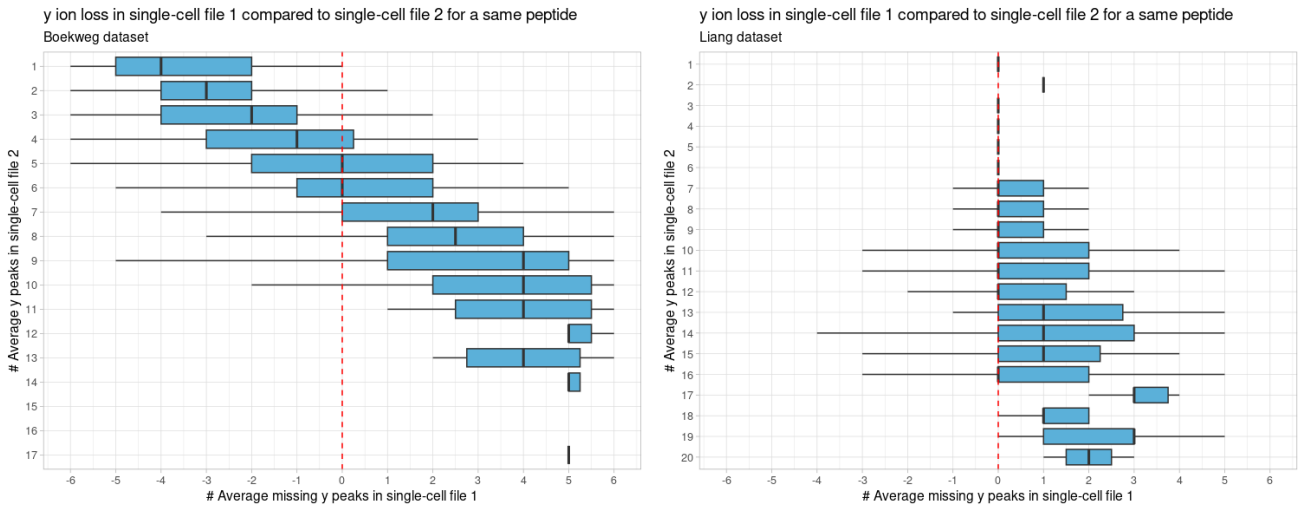


(a) For Boekweg dataset

(b) For Liang dataset

Figure 23: Y-ion loss in bulk: replication 1 compared to bulk: replication 2 of a same dataset at 1% FDR. Analysis done on the Boekweg (left) and Liang (right) datasets.

However, the figures from bulk-on-bulk y-ion analysis show the same patterns from Figure 22a. The conclusions from the Boekweg article are not specific to SC data, but a natural feature of MS-based proteomics data. The more peaks are found for a sequence in a sample, the more likely it is to lose peaks in another sample for the same sequence. With enough data available, this occruance is natural. The same observations are made on SC-on-SC y-ion loss comparisons as can be seen in Figure 24. The less pronounced trend in the data for the Liang dataset can be explained by its lack of shared sequences.



(a) For Boekweg dataset

(b) For Liang dataset

Figure 24: Y-ion loss in single-cell: replication 1 compared to single-cell: replication 2 of a same dataset at 1% FDR. Analysis done on the Boekweg (left) and Liang (right) datasets.

A more straight-forward approach to depict annotated peak loss is proposed in Figure 25. Both axes illustrate the total amount of annotated peaks in the spectra from shared sequences between bulk and SC files. Instead of only considering y-ions, both b- and y-ions are here considered: giving an idea of the overall annotated peak loss. Bear in mind, the amount of shared sequences between bulk and SC files is much more limited in the Liang dataset than it is in the Boekweg dataset. This is also reflected in the difference in scale of the legends in Figure 25. Still, it gives an overview of the global behaviour of peak loss within a dataset.

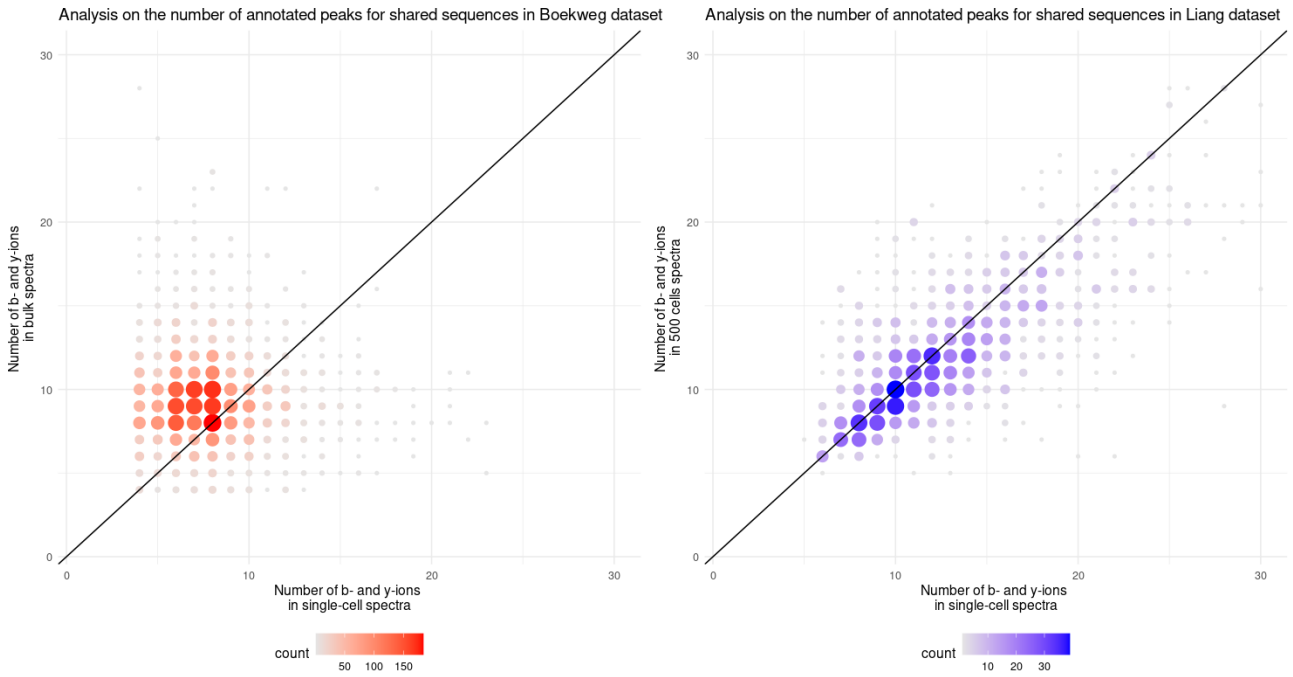


Figure 25: Annotated peak loss in single-cell compared to bulk spectra. Both axes show the number of annotated b- and y-ions for a shared sequence between bulk and single-cell spectra. Values above the diagonal (x = y) through the origin depict a loss of peaks in single-cell spectra compared to bulk. Analysis done on both the Boekweg (red) and Liang (blue) datasets.

According to Figure 25, there is indeed a slight loss of annotated peaks in SC spectra from the Boekweg dataset. Most of the shared sequence are plotted above the diagonal line, which means that there are less annotated peaks in SC spectra than in bulk spectra.

The Liang dataset shows a correlation of the number of annotated peaks between the SC and 500 cells samples. There doesn't appear to be any distinct losses in the number of annotated peaks. A possible explanation for this lack of peak loss could be that 500 cells is not enough to show spectral characteristics typical of bulk data (which can range up to millions of cells). Moreover, there is a distinct difference in the amount of shared sequences between both datasets. The Liang dataset only has 1442 shared sequences between bulk and SC files, whereas Boekweg's dataset goes up to 4295

shared sequences. Additionally, the figure demonstrates a lower amount of identified peaks in Boekweg's dataset's average number of peaks compared to Liang's dataset's average number of peaks. Once again, further analysis is needed to better understand where the losses originate from and why these differences between both figures are observed.

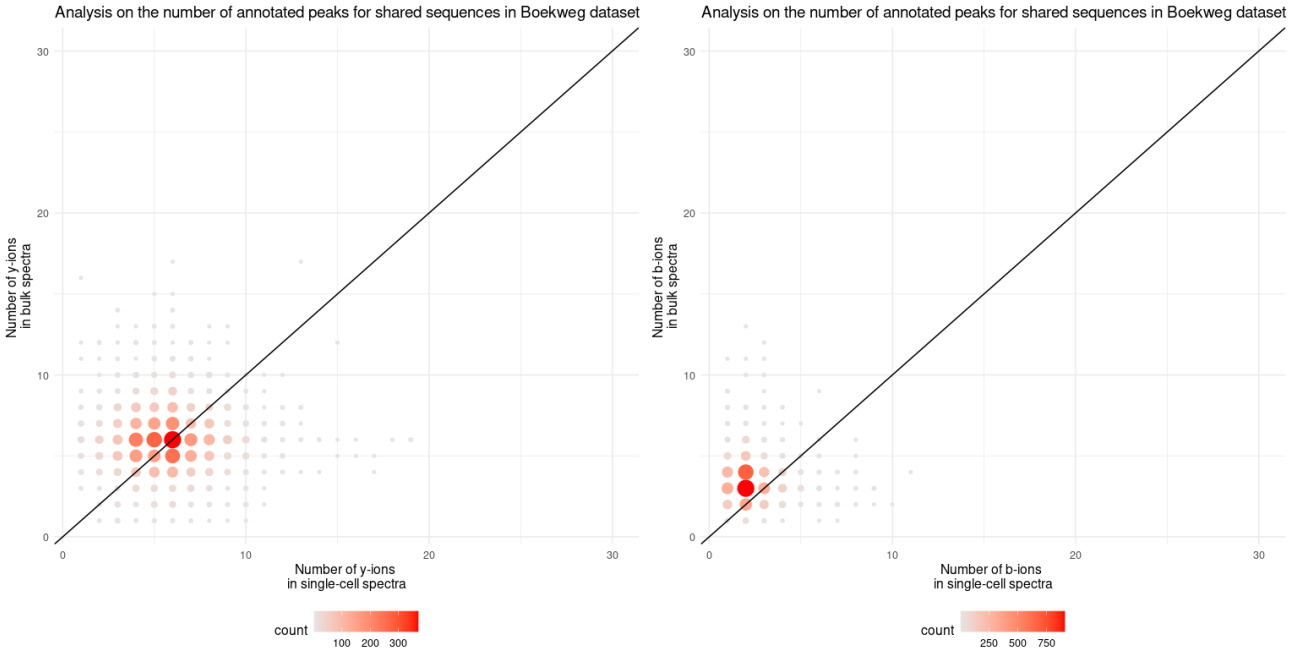


Figure 26: Annotated y-ion (left) and b-ion (right) peak loss in single-cell compared to bulk spectra respectively. Values above the diagonal through the origin depict a loss of peaks in single-cell spectra. Analysis done on the Boekweg dataset applied on the y-ions and b-ions.

Figure 26 shows that it is only the b-ions that influence the total peak loss in SC for the Boekweg dataset. The y-ions depict a symmetric dispersion around the diagonal through the origin: this will be referred to as a random deviation. This random deviation is also shown in Figure 27. In between samples, random deviation around the diagonal is expected: similar files have similar amounts of annotated peaks although these numbers can vary slightly due to random technical variations. Within samples of a same number of cells, no notable peak loss is observed.

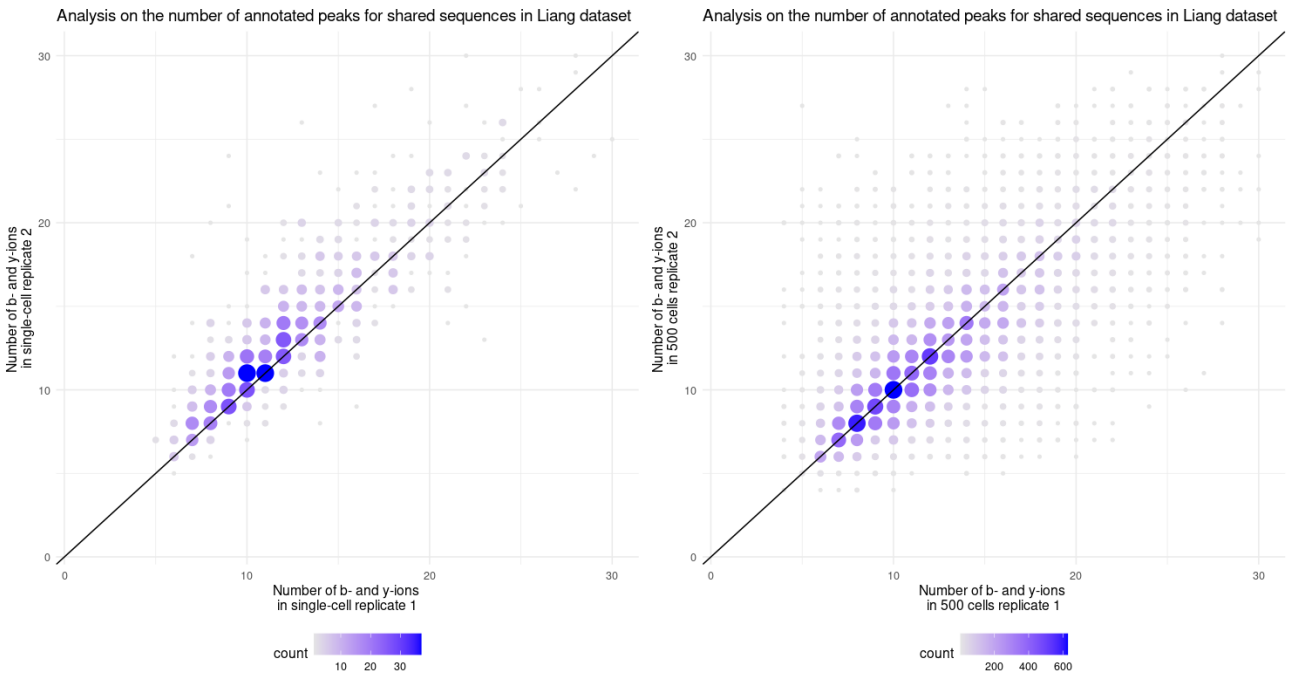


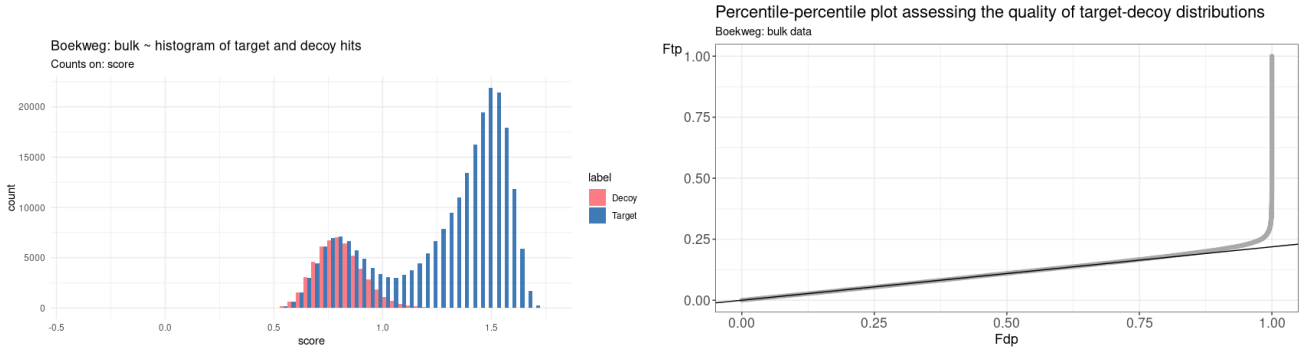
Figure 27: Similar files have similar amounts of annotated peaks. Random deviation between spectra is expected. Analysis done on the single-cell (left) and 500 cells (right) files from the Liang dataset.

Detailed distributions of b- and y-ions in SC spectra are depicted in Figure 49 in appendix A.

3.1.5 Target-Decoy separability

The number of annotated peaks are not the only feature used in identification. The target-decoy approach is responsible for defining the amount of confidently identified PSMs by measuring a PSM's corresponding FDR. It is crucial to understand the differences between bulk and single-cell samples in this regard.

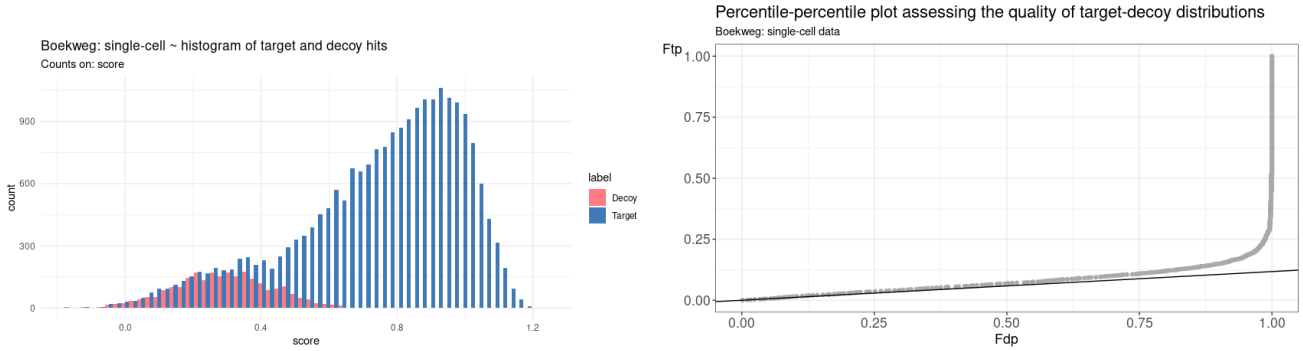
Figures 28a and 29a show the binomal distribution of target PSMs (counts as a function of the `sage_discriminant_score`), distinctively distancing true positives (blue) from false positives (red) for bulk and SC respectively. A clear separability and discrimination of TP and FP is key here, and this is only achieved when the first curve of the target binomal distribution follows the decoy distribution. The second curve should only appear and deviate from the decoy distribution at the end of the decoy distribution's tail.



(a) Distributions of PSMs based on their score

(b) PP-plot assessing the deviation of distributions

Figure 28: Quality assesment of target-decoy distributions for Boekweg: bulk dataset.



(a) Distributions of PSMs based on their score

(b) PP-plot assessing the deviation of distributions

Figure 29: Quality assesment of target-decoy distributions for Boekweg: single-cell dataset.

The quality of such distributions, meaning the separability between target and decoy distributions, can be assessed using PP-plots. Such plots are depicted in Figures 28b and 29b, reflecting the quality of their corresponding distributions. As a reminder, PP-plots have the property that they show a straight line through the origin if and only if both distributions are equivalent. Any deviation from this straight line indicates that the distributions differ (Meloun and Militký, 2011).

In other words, the sooner the PP-plot deviates from the line through the origin, the earlier the target distribution deviates from the decoy distribution, resulting in less confidentlty identified PSMs. Indeed, since a FDR is measured by the formula $FDR = \frac{FP}{TP+FP}$ above a given score, deviating from the decoy distribution results in a higher number of FP at the given score.

According to above-mentioned figures, bulk data clearly shows better quality distributions with clear separability of target and decoy PSMs compared to the SC distributions. However, these observations were not as obvious in the Liang dataset as the distribution of target PSMs in SC data showed weaker deviations (see Figure 51 in appendix A).

3.2 Custom Linear Discriminant Analysis

Sage's algorithm and scoring method is based on an LDA of a list of sixteen features mentioned in Table 1. Reproducing the scores and spectrum FDR is feasible in *R* and it allows for more flexibility in the selection of features. The results in this section are based on *Sage*'s LDA implemented in *R*.

3.2.1 Impact of *Sage*'s initial features

The objective of the LDA in *Sage*'s algorithm is clear: optimize the separability of targets and decoys given the data. The more information, also referred to as features, is added, the better the separation. Some of the features explain the target-decoy separability better than others. However, none of the features are to be disposed as every additional information can be used to separate targets from decoys.

Figure 30 shows the importance of each of *Sage*'s initial features based on a ratio of confidently identified PSMs: $\frac{\text{Number of PSMs without feature xx}}{\text{Number of PSMs with all initial features}}$. Considering bulk data, the features that seem to induce the largest loss of PSMs when removed are (in order): `poisson` and `delta_rt_model`. The same observations are made for the Boekweg dataset (see Figure 52 in appendix A).

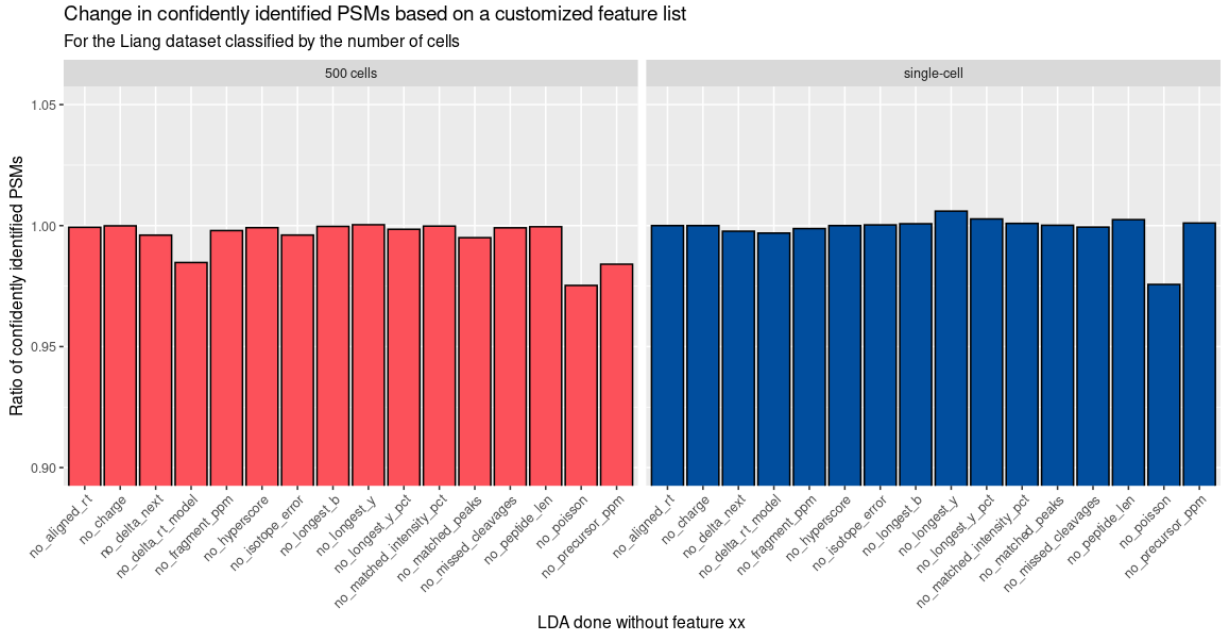


Figure 30: Overview of the impact of *Sage*'s initial 16 features on the number of confidently identified PSMs. A ratio ratio of confidently identified PSMs is shown: $\frac{\text{Number of PSMs without feature xx}}{\text{Number of PSMs with all initial features}}$. A low value means a high loss in PSMs and thus a greater importance of the removed feature. Analysis done on the Liang dataset.

Single-cell data differs slightly from bulk in a sense that it also identifies other important features (depending on the dataset, see Figure 52 in appendix A) whilst still showing that removing the `poisson`

and/or `delta_rt_model` features results in the least confidently identified PSMs. Does that mean that these features are also the most important ones? Yes, but reality is more complex than that. The importance of a feature can be assessed in four steps as seen in 3.3.

3.3 Quality measurement of a feature

This section suggests how the quality of a variable that is used as a feature in a scoring system can be measured in multiple ways. Here, the four most relevant measuring methods are explained. These methods are complementary and should not be looked at individually, although some are more relevant. The methods cited below can be used on all features, may they be from *Sage*'s initial LDA, a custom LDA, *MS²Rescore* or even newly added variables.

3.3.1 Target-decoy discrimination

A simple way to judge whether a feature is discriminatory towards target and decoy PSMs is by looking at the target-decoy distribution of the feature. Figure 31 shows how a histogram from a feature can explain the separability of targets and decoys. For instance, the `poisson` feature shows almost no decoys in low value ranges. Such clear separation is exceptional amongst the features as most of them show unclear or gradual separations rather than distinct separations.

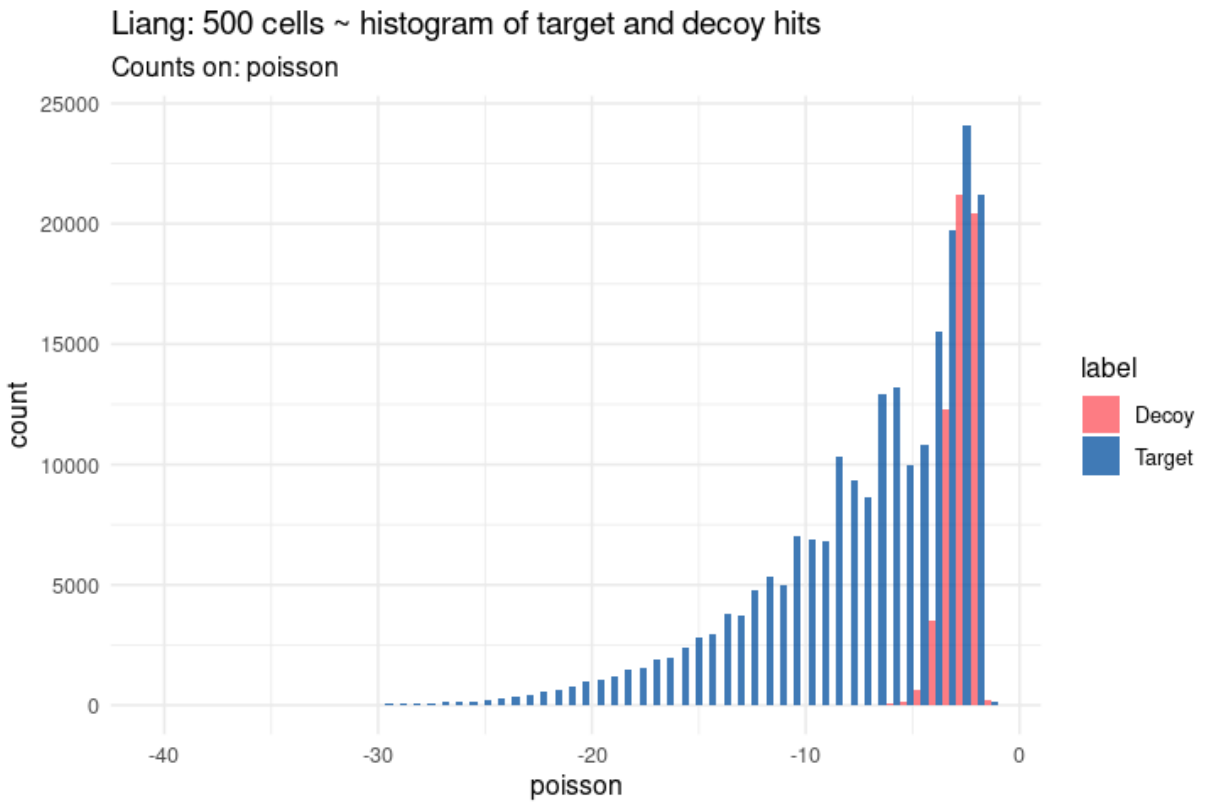
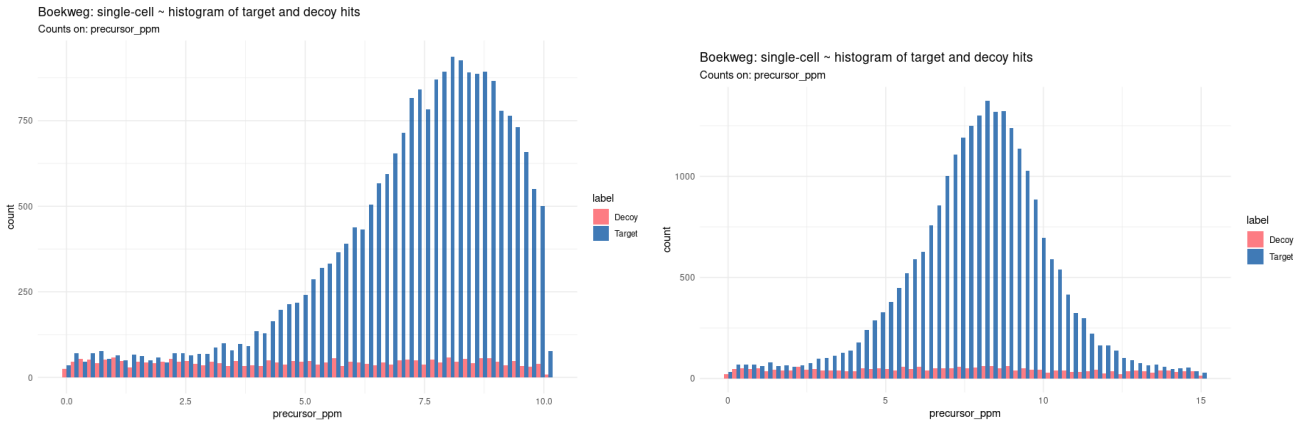


Figure 31: Histogram of the `poisson` feature for the Liang: 500 cells dataset. A clear separation of target and decoy PSMs is noticeable.

Such plots can also highlight the presence of anomalies in the data. For example, the Boekweg dataset had unexpectedly high `precursor_ppm` values (difference between experimental mass and calculated mass of precursor ion, reported in parts-per-million) where one would expect values centered around 0 (see Figure 32a). After further analysis and consulted insight from MS expert D. Vertommen, a clear presence of bad calibration of the MS was diagnosed. Resulting from this bad calibration, a lot of `precursor_ppm` with values higher than $10ppm$ weren't even considered due to a systematic error, which could lead to less discrimination into the LDA and thus less confidently identified PSMs. To counter this, the search parameters in *Sage*'s *.json* file were adapted in order to enable higher `precursor_ppm` tolerances, resolving the problem (see Figure 32b).



(a) Sage *precursor ppm* set at 10 ppm

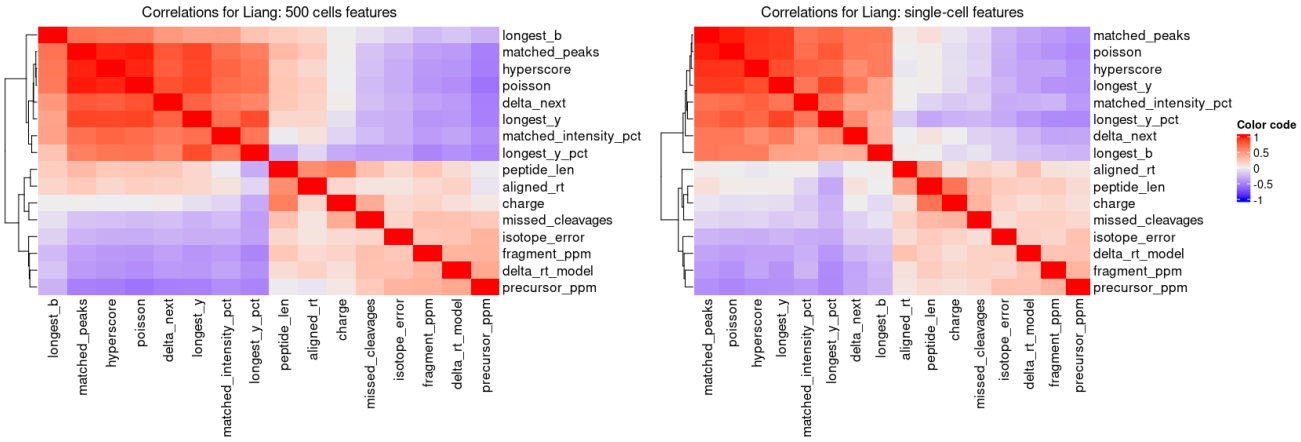
(b) Sage *precursor ppm* set at 15 ppm

Figure 32: Counts on the `precursor_ppm` feature for the Boekweg: single-cell dataset. The target PSMs are centered around higher ppm values than expected due to bad calibration of the mass spectrometer.

In a way, the histograms provide information on both the quality of a feature and the potential anomalies in the data.

3.3.2 Heatmap of correlations

Features rarely bring completely new information as they are often correlated with each other, either positively or negatively. Through an LDA, any new information or feature is used to better explain separability of two or more classes. This means that as long as two features are not 100% correlated, they have discriminative information that can be extracted. Figures 33a and 33b show the correlations between features in bulk and in SC respectively. Highly positively correlated features are depicted in red, highly negatively correlated features in blue and uncorrelated features in white. Ideally, a feature should be completely uncorrelated to other features for it to bring out most discriminative information.

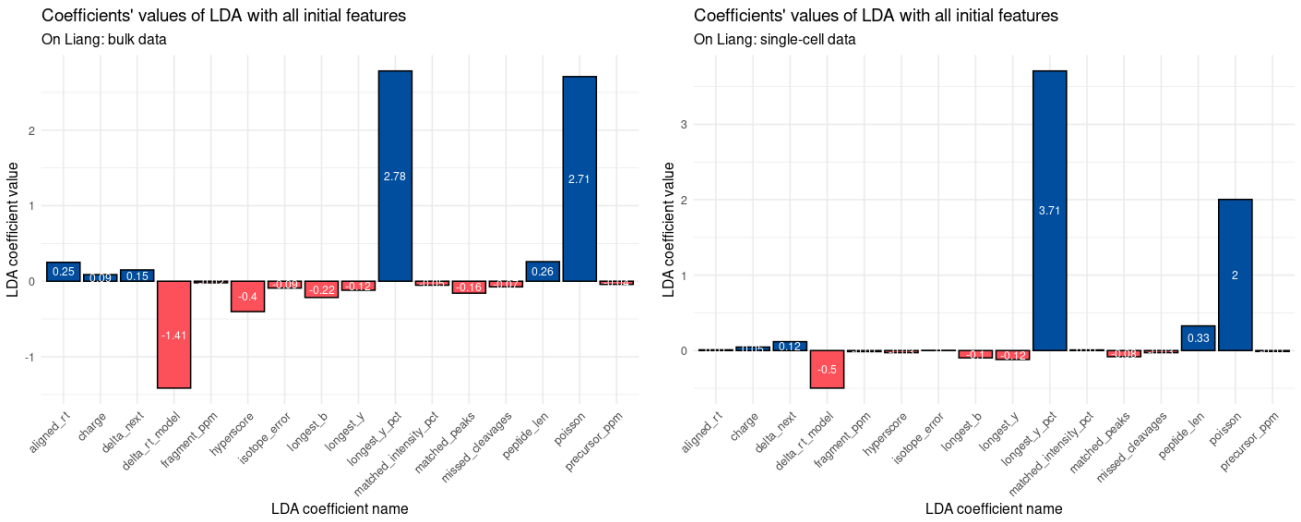


(a) For 500 cells (b) For single-cell

Figure 33: Heatmap of feature correlations for the Liang dataset.

3.3.3 LDA coefficients' weights

Complementarily to the heatmaps, the coefficients' weights reveal the importance of a feature based on its discriminative power. Figures 34a and 34b show the weights of the features in bulk and SC data. A coefficient is characterized by a sign that determines the direction of its impact (target or decoy) and by a its absolute value that determines the feature's weight. The feature with the highest weight is the `longest_y_pct` in both bulk and SC data. This is due to the correlation of features. For instance, `longest_y_pct` and `hyperscore` are two features that are highly correlated and so, both may be important but only one will provide the shared information to the LDA, ultimately lowering the weight of the `hyperscore` feature.



(a) For 500 cells (b) For single-cell

Figure 34: Linear discriminant analysis coefficients' weights for the Liang dataset.

The coefficient weights need the correlations between coefficients from section 3.3.2 to be taken into consideration in order to be correctly interpreted.

In conclusion, a heavy weight for a highly uncorrelated feature could mean a feature is important to discriminate target and decoy PSMs, yet a low weight with highly correlated features does not mean the feature is unimportant. The different possibilities and their impact on feature importance are summarized in Table 3.

Table 3: Potential importance of a feature based on its combination of correlation and coefficient weight

Importance	Heavy weight	Light weight
High correlation	+	+/-
Low correlation	++	-

3.3.4 Absolute gain in PSMs and peptides

Similar to what is discussed in 3.2.1 and maybe the most relevant way to assess the quality of a feature, is by looking at the amount of confidently identified PSMs and peptide sequences brought by adding a feature. In the following sections, this measurement will be shown through the use of figures much like Figure 35, reflecting the absolute gain in PSMs and peptide sequences in percentages of initial amount identified. Notice that PSMs and peptide sequences can be gained, lost or (majoritarily) retained.

Figure 35: Example plot of absolute gain in confidently identified PSMs and peptide sequences when comparing two database searches.

This approach is not only valid for feature quality analysis, but for rescoring in general, as will be seen in section 3.4.

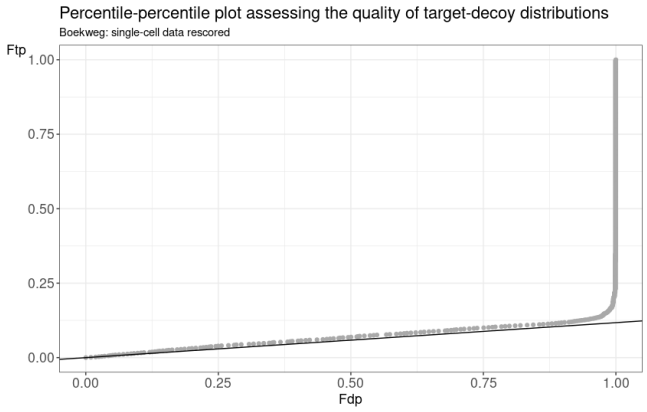
3.4 Impact of *MS²Rescore*

Sage already performs rescoring with an LDA, even if it is incomparable with the actual improvements obtained when using a tools dedicated to rescoring like *MS²Rescore*. This section will focus on the impact of rescoring tools on both bulk and SC data. Most of the feature importance measures have been mentioned in sections 3.1.5 and 3.3 and thus won't be detailed any further. We will focus on the observations that can be made from the different measures in the coming sections.

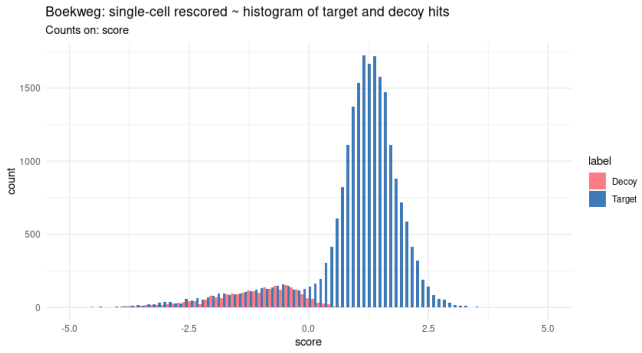
3.4.1 Impact on Target-Decoy distributions

The first noticeable improvement resulting from *MS²Rescore* lies in the loss of deviation between the decoy and target distributions (referring to section 3.1.5). Figure 36a shows a significant improvement in SC distribution quality compared to the PP-plot from Figure 29a, with significantly less deviation from the line going through the origin.

Still, the resulting distributions remain of lower quality than that of bulk data (Figures 36b and 37b), but their improvement after rescoring is significantly higher. Still, Figure 37a shows a slight enhancement in distribution quality compared to Figure 28a.

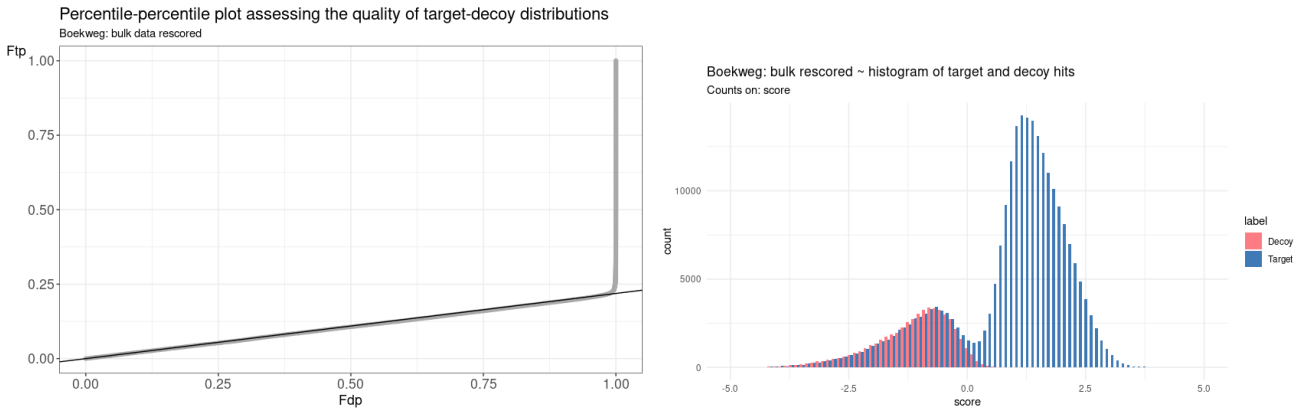


(a) PP-plot assessing the deviation of distributions



(b) Distributions of PSMs based on their score

Figure 36: Quality assesment of target-decoy distributions for Boekweg: single-cell rescored dataset.



(a) PP-plot assessing the deviation of distributions

(b) Distributions of PSMs based on their score

Figure 37: Quality assesment of target-decoy distributions for Boekweg: bulk rescored dataset.

When considering the better improvement in SC compared to bulk, it appears that rescoring has more importance on SC data than it does on bulk data. It is undeniable that for low-quality distributions, rescoring favors a boost in the amount of PSMs below the same FDR threshold.

3.4.2 Feature weight analysis

MS²Rescore uses *Sage*'s initial PSM file, and relies on *MS²PIP* and *DeepLC* to generate new features. Features are then provided to the rescoring algorithm *Mokapot*. It is then interesting to discover which features actually carry more weight during rescoring (in a similar way to section 3.3.3). The feature weights used in the 3 subsets of *Mokapot* (see section 1.4.1.2) are provided by *MS²Rescore*. The median of the three weights corresponding to each subset is a good representation of the importance of each feature. Figure 38 represents a global view of the feature weights based on their origin for SC (left) and bulk (right) data respectively. Akin to the weights pictured in section 3.3.3, features related to retention time show higher weights (for instance `rt_diff_best` from the *DeepLC* feature generator). Globally, the weight distribution between feature generators is very similar for both SC and bulk data. Only the intensity of the `rt_diff_best` feature is notably lower in SC than it is in bulk.

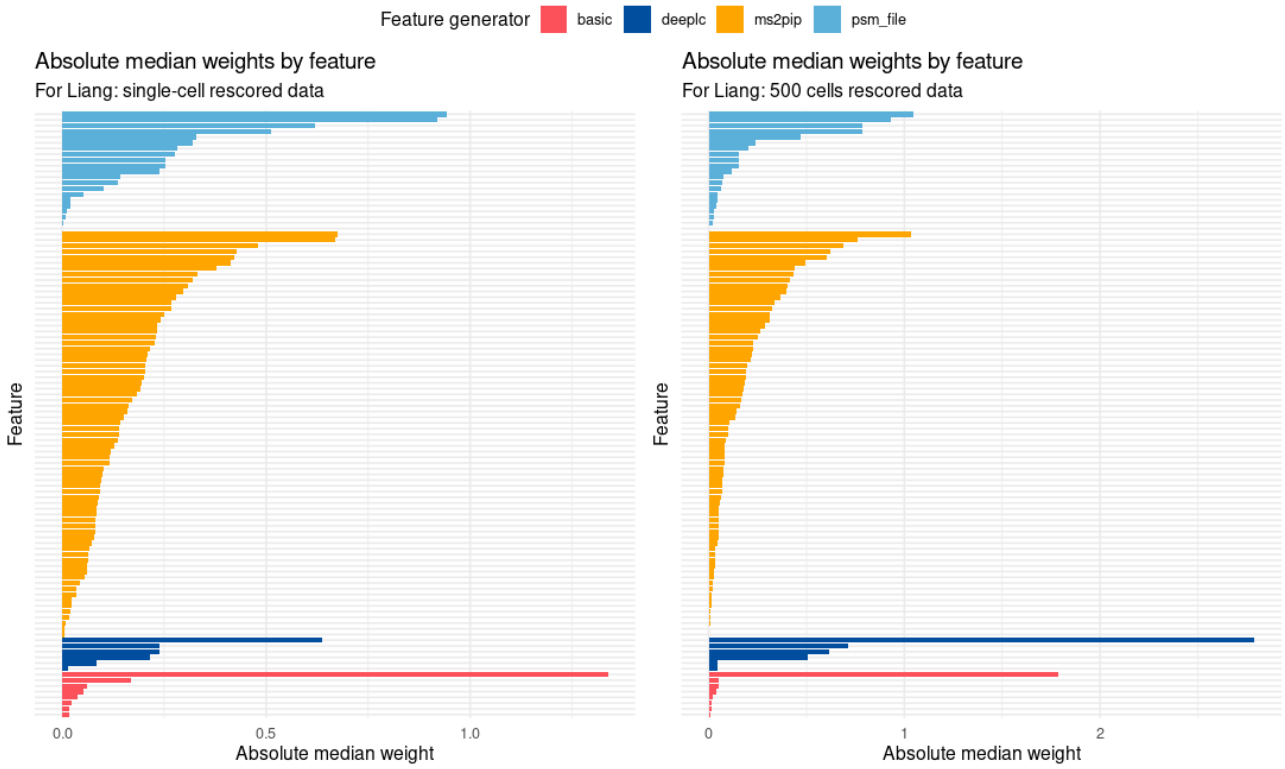


Figure 38: Feature weight reflecting the importance of each feature in rescoring. The weights were provided by *MS²Rescore*. Analysis done on the Liang dataset's single-cell (left) and bulk (right) files.

For better readability and for clarity's sake, a zoomed in version of the figures is depicted in Figure 39. Only the features that are used by *Sage*'s LDA will be discussed since those are the ones we have been working with so far.

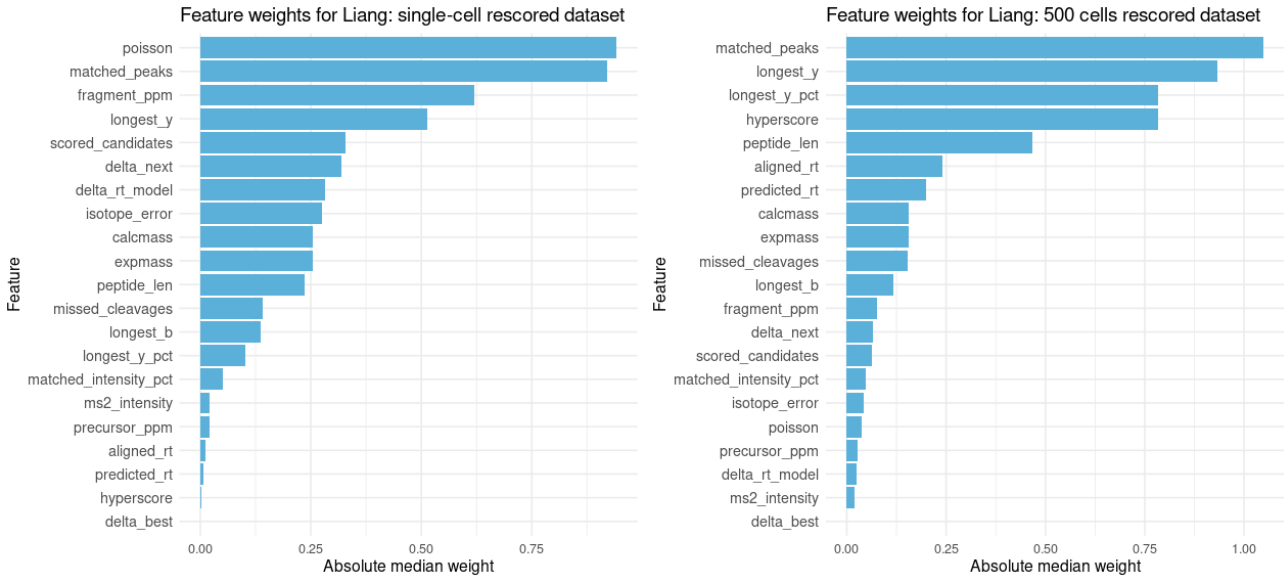
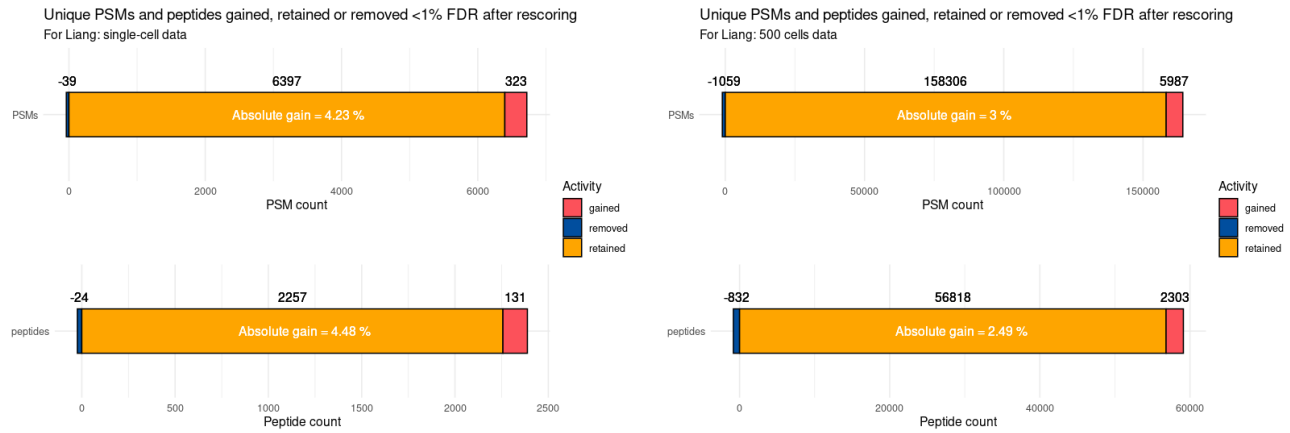


Figure 39: Feature weight reflecting the importance of features (from the PSM file) in rescoring. The weights were provided by *MS²Rescore*. Analysis done on the Liang dataset's single-cell (left) and bulk (right) files.

Compared to *Sage*'s LDA coefficient weights, one of the leading features for both SC and bulk data happens to be the `matched_peaks` feature. This feature is used when calculating the `hyperscore` and thus the two features are highly correlated (see section 3.3.2). For SC's weight distribution, the lower weight for the `hyperscore` could be explained through this correlation. A similar reasoning can explain the low `poisson` and high `hyperscore` feature weights for the bulk data. Similar observations are made on the Boekweg's dataset (see Figure 53 in appendix A).

3.4.3 Increase in PSM and peptide counts

Now, the most important aspect of a rescoring tool is its ability to increase the number of PSMs under the same FDR threshold. As evidenced by Figure 40, rescoring on SC brings more PSMs and peptide sequences than it does on bulk data in terms of percentages: SC shows an absolute increase of 4.23% and 4.48% respectively where bulk only shows an increase of 3% and 2.49%.



(a) For single-cell

(b) For 500 cells

Figure 40: Variation in confidently identified PSMs and peptide sequences after rescoring using *MS²Rescore*. Analysis done on Liang dataset.

Through these analyses, we noticed that the seed of *Mokapot*'s SVMs is not fixed, resulting in inconsistent gains in between replicate runs. For bulk data, these variations in gains are minor. On the contrary, SC data showed larger variations in between runs. Figure 41 demonstrates the mean absolute gains of five runs for each data type after rescoring.

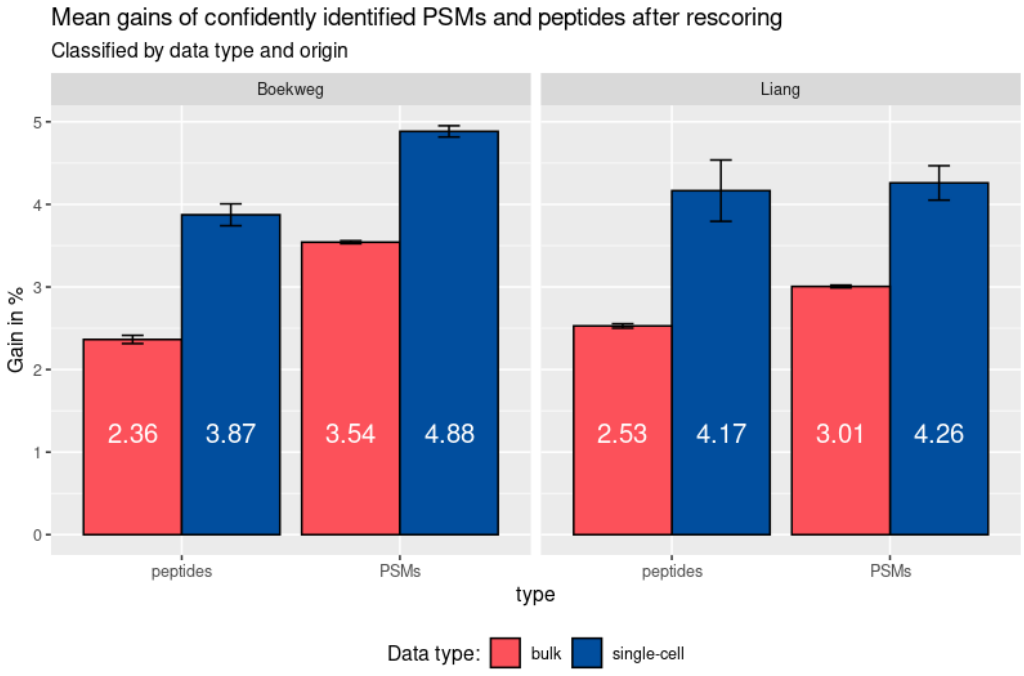


Figure 41: Absolute gains in percentages after five replicated runs of rescoring with *MS²Rescore*, classified by origin. The mean gains in confidently identified PSMs and peptides as well as their standard deviations are depicted.

Following on from previous observations, rescoring on SC data seems to systematically provide more relative gains than it does on bulk data. The gains in SC typicaly rise more than 1% above the gains depicted in bulk, regardless the type of gain (PSM or peptide).

Note that the link between PSM and peptide gains differ when comparing data types. There seems to be a higher ratio of $\frac{\text{peptide gain}}{\text{PSM gain}}$ in SC than there is in bulk (see Table 4).

Table 4: Ratio of absolute gains in peptides over PSMs for different datasets and data types

Data origin	Single-cell	Bulk
Boekweg	0.79	0.67
Liang	0.98	0.84

Hence, it seems more fruitful to identify one more PSM in SC than it is in bulk. For every 10 PSMs identified, 8 to 10 peptide sequences are identified as well for SC analyses compared to only 7 or 8 in bulk.

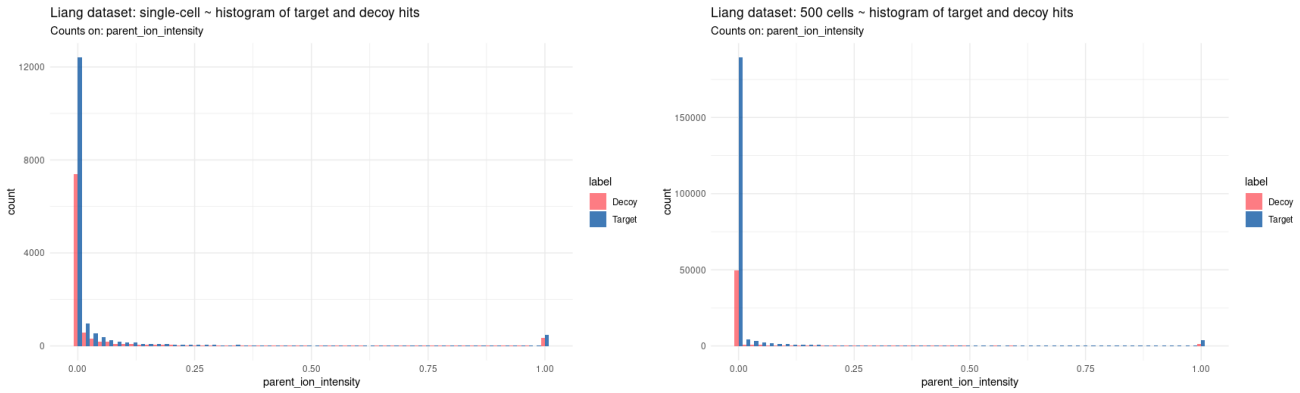
3.5 Potential of additional features

This section assesses the quality of potential features proposed by Dr. Vertommen (see section 2.6). The features discussed below include the parent ion intensity and symmetry around the parent ion in MS2 spectra. Their quality was assessed according to section 3.3 and their potential implementation will be discussed in section 4.

3.5.1 Feature counts

- **Parent ion intensity:**

There is no apparent difference between the target and decoy distributions along the parent ion intensity over base peak value (see Figure 42). In both SC and in bulk, around 95% of the spectra do not have any parent ion left. When comparing the ratio of $\frac{\text{decoys}}{\text{targets}}$ at `parent_ion_intensity` $= 0$ and at `parent_ion_intensity` $= 1$, an increase in decoys is noticeable. For SC this ratio goes from 0.595 to 0.713 and for bulk it goes from 0.262 to 0.335. There is thus a higher proportion of decoys for MS2 spectra that have the parent ion as their base peak. The overall value for this ratio is also higher in SC than it is in bulk.

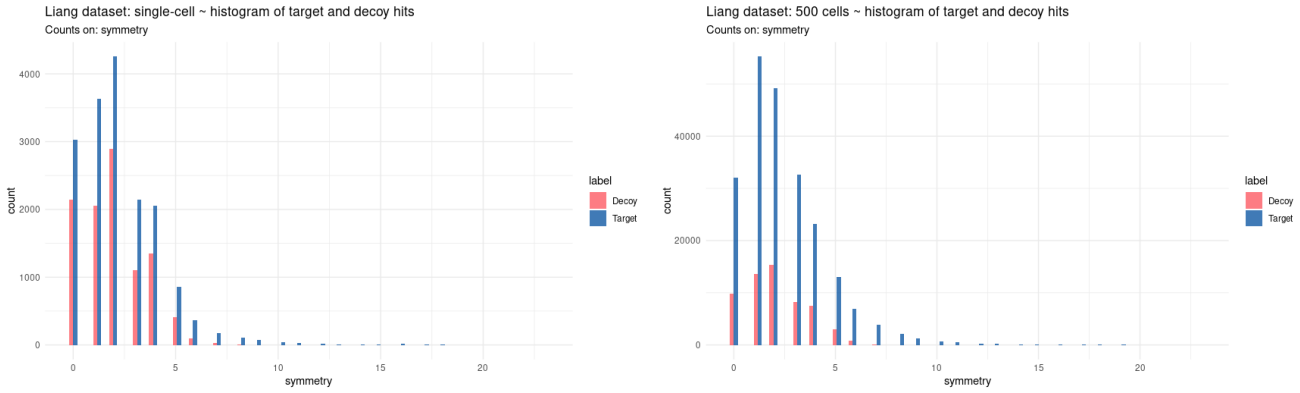


(a) For single-cell (b) For 500 cells

Figure 42: Counts of target and decoy PSMs for parent ion intensity feature applied on Liang dataset.

- **Symmetry around the parent ion m/z:**

Not much can be interpreted from Figure 43 except that most of the spectra have good symmetry with one to three annotated peak differences around the parent ion m/z value. Keep in mind not many conclusions can be pulled from this graph alone for most of the initial features as well.



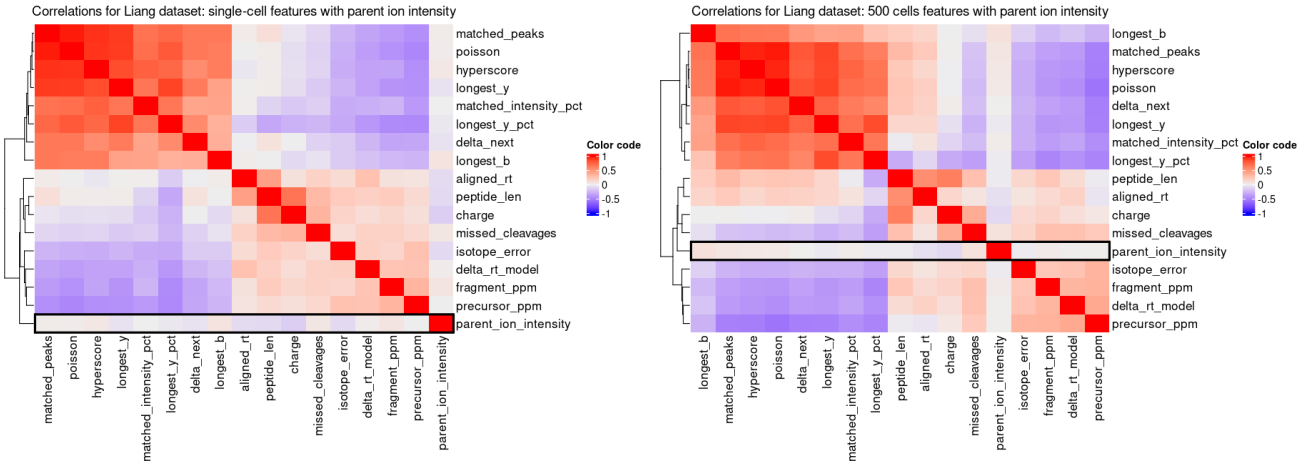
(a) For single-cell (b) For 500 cells

Figure 43: Counts of target and decoy PSMs for symmetry feature applied on Liang dataset.

3.5.2 Heatmaps

- **Parent ion intensity:**

Figure 44 shows how uncorrelated the `parent_ion_intensity` feature is respective to the other features. This absence of correlation might be related to the fact that most spectra don't have any parent ion remaining and thus no direct correlation can be extracted from it. The same observations are made for both SC and bulk data.

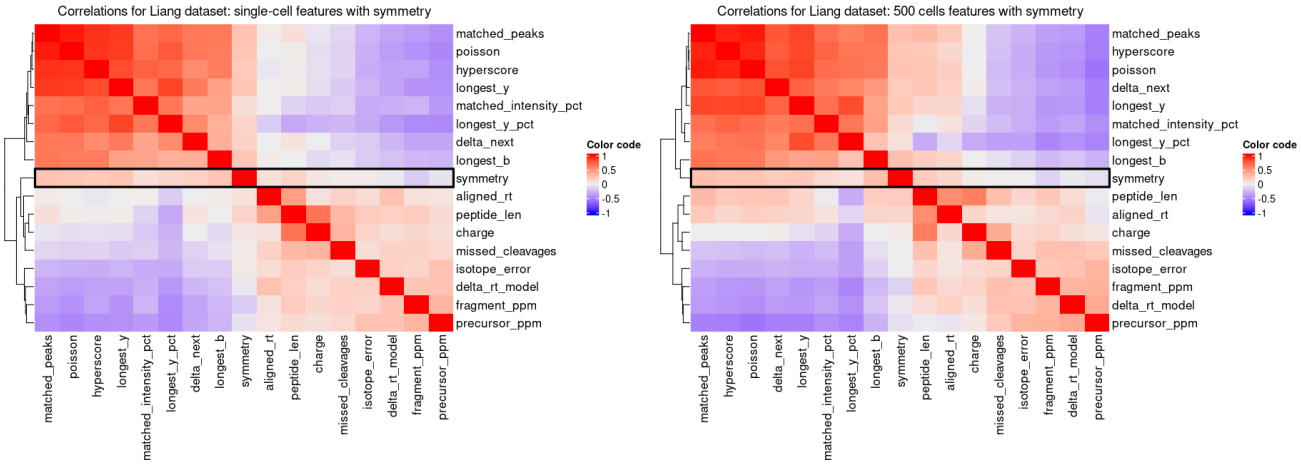


(a) For single-cell (b) For 500 cells

Figure 44: Correlation of features with parent ion intensity feature applied on Liang dataset.

- **Symmetry around the parent ion m/z:**

The `symmetry` feature however, depicts some mild correlations with the other features as is shown in Figure 45. Specifically the features containing peak information show some correlation (for instance the `matched_peaks` and `longest_y` features) with the symmetry feature. The same observations are made for both SC and bulk data.



(a) For single-cell (b) For 500 cells

Figure 45: Correlation of features with symmetry feature applied on Liang dataset.

3.5.3 Feature weights

- **Parent ion intensity:**

For both SC and bulk, the `parent_ion_intensity` feature has one of the lowest weights (among the

initial features) during rescoring.

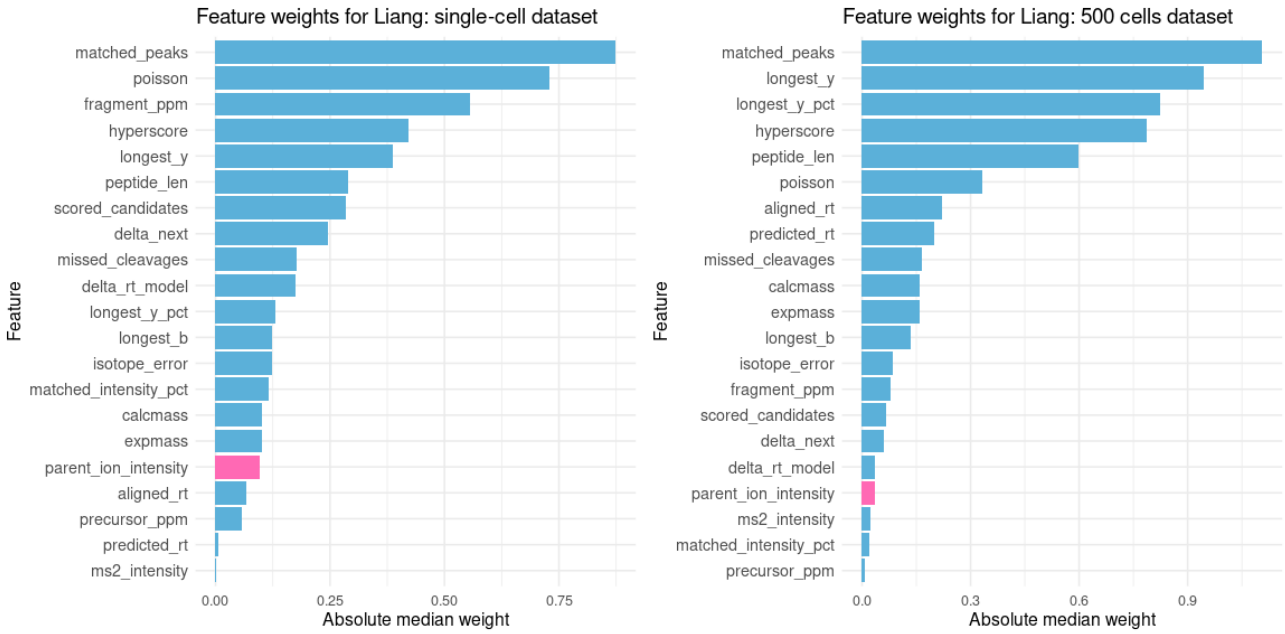


Figure 46: Feature weights with parent ion intensity feature applied on Liang dataset. The parent ion intensity feature weight (pink) is shown for both the single-cell (left) and 500 cells (right) datasets.

- **Symmetry around the parent ion m/z:**

There is a clear difference in feature weight between SC and bulk data for the `symmetry` feature. As is pictured in Figure 47, the `symmetry` feature has a weight matching the average feature whereas it is very low for bulk data.

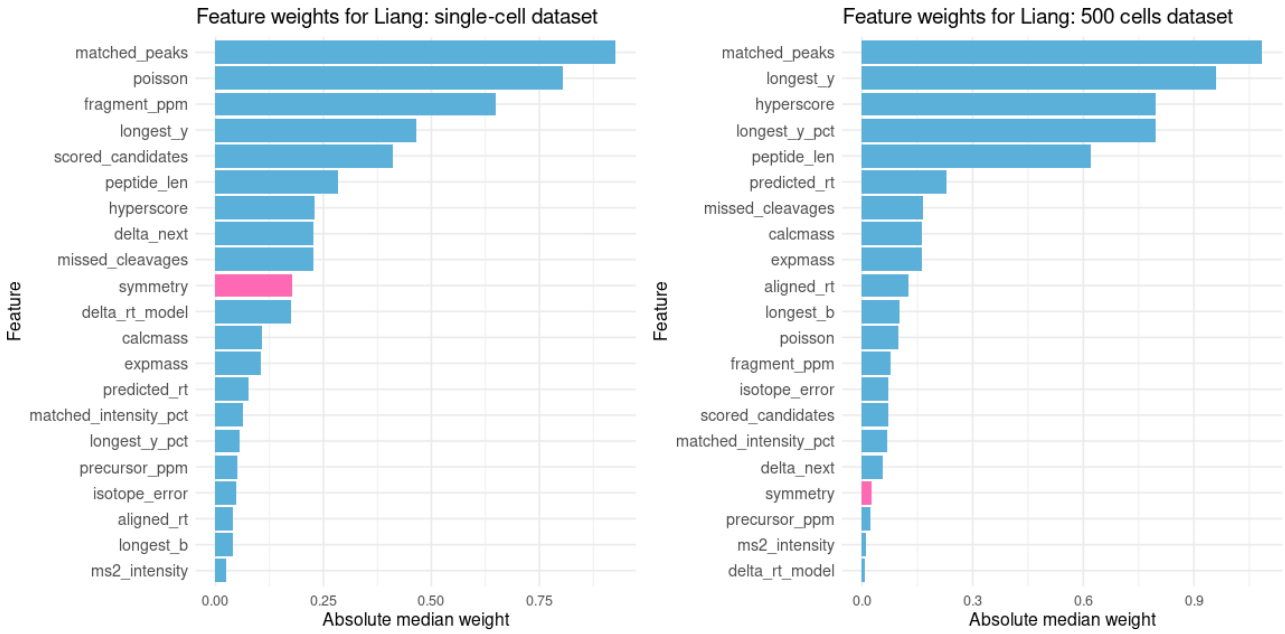


Figure 47: Feature weights with symmetry feature applied on Liang dataset. The symmetry feature weight (pink) is shown for both the single-cell (left) and 500 cells (right) datasets.

3.5.4 Absolute gains in PSMs and peptide sequences

An overview of the absolute gains for various rescoring types is depicted in Figure 48. For both added features and the random variable (for negative control, see section 2.6), and for both SC and bulk, no apparent rise in absolute gains can be observed. Some very slight increases are noticeable, but these are not sufficient to generalize a conclusion. For instance, the `symmetry` feature seems to add the most PSMs and peptides for SC when applied on the Liang dataset. Overall, the features do not depict a substancial impact on the rescoring output.

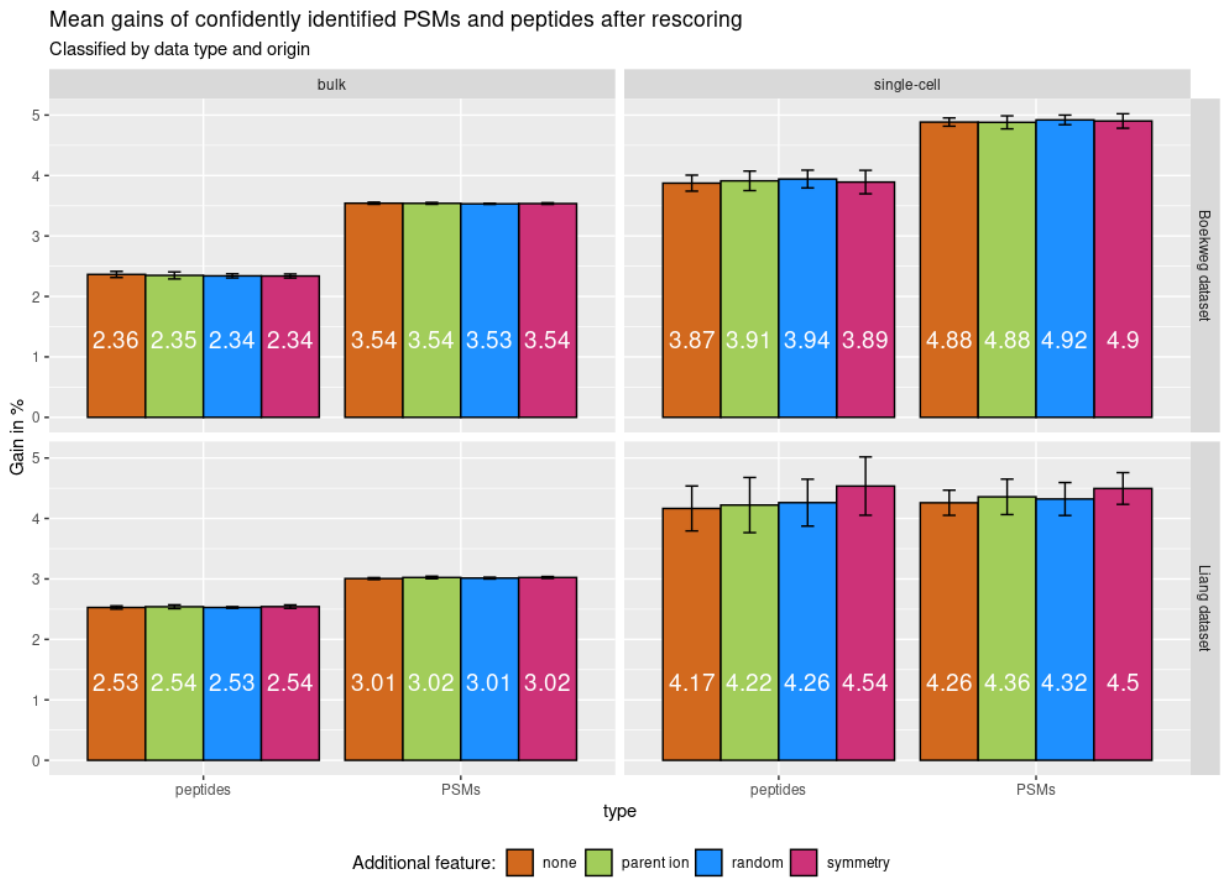


Figure 48: Absolute gains of confidently identified PSMs and peptides after rescoring with the added features. Five runs of each type were done. Analysis done on bulk and single-cell data of the Boekweg and Liang datsets.

3.6 Computing time

Table 8 in appendix B reveals the computing times of the different programs mentioned in this report for both SC and bulk data.

4 Discussion

4.1 Spectral differences

Bulk spectra inherently differ from SC spectra. Boekweg et al. (2022) demonstrated these differences in their article:

1. Loss in peak intensities
2. Intensity compression (lower signal-to-noise ratio)
3. Inconsistency in fragment intensities
4. Annotated peak loss in single-cell compared to bulk

The first three elements of this list were confirmed by generating and comparing spectra from bulk and SC samples (see section 3.1.3).

For the annotated peak loss however, it was demonstrated that the article used a misleading graphic. The first misleading element is that Figure 22 refers strictly to y-ion losses when an overall picture of peak loss is saught after. The second misleading element is that the figure simply shows a natural occurance of peak loss regardless of the data used for comparison. When a lot of peaks are annotated in a sample, it is natural that more peaks can be lost in another sample for a spectrum of the same sequence. Inversely, when few peaks are already annotated, it is unlikely to lose more peaks in another sample and still obtain a reliable identification. This natural occurance is not specific to SC data at all. This was demonstrated by generating the same graphic using replicates from the samples.

In order to truly asses annotated peak loss, a more straight-forward approach was done (in section 2.3). The analysis focuses on the total amount of annotated peaks rather than peak loss and considers both b- and y- ions rather than y-ions only. We confirmed that with this approach, the number of peaks between replicates was the same on average (for both bulk and single-cell replicates).

In this analysis, differences between the Boekweg and Liang datasets were observed. The expected correlation between replicates was also observed between the 500 cells and SC files from the Liang dataset. A possible explanation for this correlation is that 500 cells doesn't represent bulk samples well enough (as bulk samples generally represent thousands to millions of cells). It could be that 500 cells spectra show some similar spectral characteristics as SC spectra, notably in the number of annotated peaks.

Opposite to that, the Boekweg dataset displayed a slight loss of peaks: peptides from bulk spectra with

8-10 annotated peaks generally lose 2-4 peaks in SC spectra. The origin of the peak losses doesn't come from y-ion peaks like Boekweg et al. suspected, but rather from a loss in b-ions. Figure 49 from appendix A shows that SC spectra often have a lot of annotated y-ions (5-13) and few b-ions (1-6). Within the b-ions, mostly the b_2 fragment is observed (Figure 50 from appendix A).

Overall, results confirm the aforelisted spectral differences while highlighting the misleading use of figures from the Boekweg et al. (2022) article.

4.2 Importance of rescoring

The separability between target and decoy PSM distributions in SC data is worse than the one in bulk data (see section 3.1.5). This was demonstrated by using PP-plots in which SC files showed early deviations of distributions thus resulting in less confidently identified PSMs.

Because it is the ultimate goal to obtain the most confidently identified PSMs, target-decoy separability is improved through rescoring. On both bulk and SC samples, rescoring has proven to notably increase the amount of confidently identified PSMs and peptides. Results show that rescoring on SC samples systematically result in bigger increases than rescoring on bulk samples (Figure 41).

Moreover, the proportion of peptides gained per gained PSM is higher in SC than it is in bulk data (Table 4). This ratio even seems to gradually increase proportionally with the sample size: for Boekweg's bulk data, the ratio is lower than it is for Liang's 500 cells data. This further reinforces the assumption that 500 cells samples share characteristics from both bulk and SC samples.

However, it is to be noted that multiple runs are needed to truly identify the total gain in PSMs and peptides. *Mokapot* and *Percolator* include a randomized subsetting in their machine learning models and thus, replicate runs don't generate the same scores. These differences in scores are then reflected in the FDR and different amounts of PSMs and peptides are then confidently identified.

In this study, five rescoring runs were applied to truly assess the differences in gains. For bulk data, the sample standard deviation was negligible and the gains were almost identical at every run. This is also true for Liang's 500 cells. Single-cell replicated rescoring runs however, showed greater standard deviations. This difference is explained through the substantially lower amount of PSMs in the SC files provided to the rescoring engines. The SVMs for these files contain fewer PSMs and thus, variations in the training data induce variations in the testing model. The opposite is true for bulk: more data is provided to the SVMs and testing models barely change in between runs.

Another aspect when considering the use of rescoring is its runtime. For SC files, this runtime is

significantly lower than it is for bulk files. *MS²Rescore*'s runtime ranges from 20 to 30 minutes for SC files and 1.5 to 3 hours for bulk files.

Overall, the importance of rescoring is confirmed for both bulk and SC samples. It is even more prominent in SC samples whilst showing greater variances in between rescoring runs.

4.3 Importance of features

Whether rescoring is done through an LDA or through SVMs, spectral features are the key elements for increasing target-decoy separability. A customized approach to *Sage*'s LDA showed that removing certain features incurred greater losses in confidently identified PSMs. Two features distinctively showed the greatest losses in PSMs for both SC and bulk data: (i) the difference between the predicted and observed retention time and (ii) the probability of matching exactly N peaks across all scored candidates for that PSM. However, this customized approach did not suffice to assess whether those features were more specific to SC or to bulk data.

To further assess the importance of a feature, a more in-depth analysis was needed. This study proposes a four-step methodology to accurately achieve this (see section 2.5):

Clear absence or presence of target-decoy discrimination: This step allows a more general overview of the target-decoy discrimination by the feature. The most important features can manifest themselves here (for instance the two features mentioned above). Because bulk data contains abundantly more PSMs, the discrimination is also more prominent than it is in SC files. However, for most features, a clear discrimination between target and decoys PSMs is not observable. Still, this step also serves as quality control. By evaluating feature behaviour, it even allowed the identification of a bad calibration of the mass spectrometer for the Boekweg dataset. Adapting the search parameters resolved this issue.

Heatmap of correlations: Important features bring information that are not already carried within other features, and thus exhibits little correlation with those. This doesn't mean that highly correlated features are systematically less important though. In reality, the heatmaps of correlations must be interpreted in combination with the LDA coefficients' weights.

LDA coefficients' weights: Each feature has some weight associated to it when rescoring. Heavy weights systematically indicate a higher importance of a feature. However, low weights do not per se mean that the feature in question is unimportant. Table 3 demonstrates that, in combination with the heatmaps of correlations, an estimated importance of a feature can be acquired. The worst case scenario consists of features with very low weights and low correlations: meaning that the feature just doesn't hold any discriminatory power. A heavy weight with a low correlation however, is the mark of

an important feature.

Absolute gain in confidently identified PSMs and peptide sequences: Ultimately, researchers want to acquire the most confidently identified PSMs and peptide sequences. This is thus one of the most revealing steps of feature importance.

This methodology confirmed the importance of some essential features in both bulk and SC such as (i) the hyperscore, (ii) the probability of matching exactly N peaks across all scored candidates, and most importantly (iii) the difference between the predicted and observed retention time. However, only the features used by the search engine sage were assessed in the context of rescoring through an LDA. The feature generator *DeepLC* from *MS²Rescore* showed just how much more important retention time prediction actually is. The weights of features related to retention times were dominating in both bulk and SC data. It thus seems essential to constantly apply retention time prediction when rescoring, be it in bulk or SC. Still, no distinct features specific to SC data were identied.

4.4 Potential of additional features

It is undeniable, features are of great importance for rescoring. However, a spectrum can only hold so much information and most of the used features already cover this information. Adding more features that are highly correlated to already used features only presents a limited interest, as illustrated by our four-step methodology. An attempt at adding significant, uncorrelated features was thus made.

Parent ion intensity: In MS2 spectra, the presence of a highly abundant parent ion is a sign of poor fragmentation. Such spectra are likely to be discarded or associated with a bad score. A feature consisting of a ratio of parent ion intensity over base peak intensity was generated and added for rescoring, hoping to increase target-decoy separability. However, the feature had very little weight and showed low correlations with other features. Furthermore, the addition of the feature for rescoring did not have an apparent impact on the total gain in confidently identified PSMs and peptides. This can be explained by the 95% of MS2 spectra that don't actually have any remains of the precursor ion. The feature is then mainly comprised of zeroes, indicating the lack of parent ion which explains the non-correlation with other features. Although the idea behind the feature reflects the quality of a spectrum, in reality, this is rarely the case in confidently identified PSMs.

Symmetry around the parent ion: This feature reflects the symmetry around the parent ion. When a precursor ion is doubly charged, symmetry around the parent ion m/z is expected. Tandem mass spectrometry spectra without this symmetry indicate poor fragmentation and should be dealt with cautiously. The symmetry feature seemed more promising as a range of different values was obtained

throughout all PSMs. Low correlations with other features were observed in both bulk and SC data. The feature's weight was on the lower side for bulk data, yet at an average feature weight for SC data. Finally, the absolute gain in confident PSMs and peptides showed a slight increase, yet still minimal in comparison to the gains upon rescoring without the symmetry feature.

Another idea for a potential feature had been discussed before: the presence of the a_2 fragment when the b_2 fragment is observed. Once again, such a feature would reflect the quality of a spectrum: when both the b_2 and a_2 fragments are present in an MS2 spectrum, it means that the precursor ion is certainly a peptide that was well fragmented (Steen and Mann, 2004).

However, even though the idea behind the potential features was to reflect the quality of a spectrum and thus bring more target-decoy separability, this was not achieved upon rescoring with the two aforementioned potential features. The proposed four-step methodology confirmed this for the parent ion intensity feature more than for the symmetry feature. Adding only one or two features to an extensive list of 104 features covering most of a spectrum's information does not have a notable impact on the amount of confidently identified PSMs.

This being said, the potential features discussed above are not to be ignored altogether. It remains true that such features reflect the quality of a spectrum. The features are easy to implement and they can be useful for spectrum validation upon biological discoveries. Both features make use of the *Spectra* package and thus, could be implemented as a function in the package for additional spectrum validation.

4.5 Data quality

This study has limited itself to two datasets, namely the Boekweg et al. (2022) and Liang et al. (2021) datasets. Throughout the analyses, some differences or irregularities have been identified within and between the datasets. This begs the question: do these differences come from differing quality of the data, is it simply due to variations between similar datasets or is there some underlying explanation?

For Boekweg's bulk data, bad calibration of the mass spectrometer has been diagnosed. Figure 32 demonstrates this anomaly of the difference between experimental mass and calculated mass, reported in parts-per-million. On top of that, the analysis on the number of annotated peaks demonstrates a lower amount of identified peaks in Boekweg's dataset (4-15) compared to Liang's dataset (6-20).

Liang's 500 cells data showed some typical characteristics from both bulk and SC data. To start, Table 2 shows that both Boekweg's and Liang's bulk data have similar amounts of identifications (64819 and 53122 confidently identified PSMs respectively): indicating that both samples behave as typical bulk

samples. However, as is mentioned above, Liang's 500 cells showed correlation with SC data in the analysis on annotated peak loss. Moreover, the ratios of gained peptides per gained PSM in rescoring also showed typical SC behaviour (to a certain degree): is 500 cells enough to represent typical bulk characteristics?

It is theorized that 500 cells don't particularly belong to one category or the other, demonstrating characteristics from both bulk and SC data. Another possibility could be that one of the used datasets is of questionable quality. It then seems important to validate the analyses done in this study on new data comprised of both bulk and SC analyses.

4.6 Conclusion and future prospects

Through this study, a more in-depth understanding of database search engines and rescoring was obtained. The importance of features for rescoring has been demonstrated, and inherent differences between SC and bulk data have been observed. However, the analyses done in this study don't identify particular features specific to SC or bulk data. Finally, this study doesn't identify whether there is any need for identification optimization in the context of SCP. We cannot reject the fact that identification methods may, or may not, perform just as well in SCP than in bulk analyses. The elements brought in this study don't suffice to reject these scoring models.

At the level of rescoring however, the implementation of SC specific features could improve target-decoy separability. The study shows that rescoring performs better for SC than for bulk data. Although an extensive list of features already covers most of a spectrum's information, rescoring tools and models have been trained on bulk spectra specifically (Fondrie and Noble, 2021; The et al., 2016; Bouwmeester et al., 2021; Degroeve and Martens, 2013). It is possible that models and prediction tools trained on SC spectra could further increase target-decoy separability. Especially prediction tools focusing on fragmentation patterns and abundances could prove useful. Thus, other possible studies could include some of the non-investigated characteristics of MS data in this project: fragmentation patterns, intensity prediction, clustering of spectra, post-translational modifications (PTMs), etc...

Fragmentation patterns: For Boekweg's bulk data at least, a loss in annotated peaks was observed. If this loss truly is related to SC spectra, it might be interesting to discover how the fragmentation patterns differ from bulk spectra. Which ions are more probable to be lost or shared between the two sample sizes? Is there a definite fragmentation pattern or do SC spectral intensities occur randomly? These patterns could be assesed through clustering of bulk and SC spectra, identifying if for shared sequences, spectra are similar. If fragmentation patterns differ in SC, this could be accounted for by

search engines to improve identification in SC data.

Post-translational modifications: In proteomics, PTMs are crucial for identifying protein function, activity and interaction. At SC resolution, they become even more relevant and may provide new biological insights (associating the abundance of the modification with the cell). In this study, some PTMs were searched for (methylation, acetylation and oxydation), but many more exist. It could be interesting to see to what extent these PTMs can be found in samples. We would need to investigate the impact of PTM search on runtime and FDR. As adding modifications in open search exponentially increases the runtime and the computational ressources required, we would need to find a tradeoff between the number of PTMs to study and the time we are willing to dedicate to search, as well as identifying the more relevant PTMs to study.

5 Bibliography

H. Boekweg and S. H. Payne. Challenges and Opportunities for Single-cell Computational Proteomics. *Mol. Cell. Proteomics*, 22(4):100518, Apr. 2023. ISSN 1535-9476, 1535-9484. doi: 10.1016/j.mcpro.2023.100518.

H. Boekweg, D. Van Der Watt, T. Truong, S. M. Johnston, A. J. Guise, E. D. Plowey, R. T. Kelly, and S. H. Payne. Features of Peptide Fragmentation Spectra in Single-Cell Proteomics. *J. Proteome Res.*, 21(1):182–188, Jan. 2022. ISSN 1535-3893, 1535-3907. doi: 10.1021/acs.jproteome.1c00670.

R. Bouwmeester, R. Gabriels, N. Hulstaert, L. Martens, and S. Degroeve. DeepLC can predict retention times for peptides that carry as-yet unseen modifications. Feb. 2021.

L. M. Buur, A. Declercq, M. Strobl, R. Bouwmeester, S. Degroeve, L. Martens, V. Dorfer, and R. Gabriels. MS²Rescore 3.0 is a modular, flexible, and user-friendly platform to boost peptide identifications, as showcased with MS Amanda 3.0. Technical report, Chemistry, Nov. 2023.

D. Charif and J. R. Lobry. SeqinR 1.0-2: a contributed package to the R project for statistical computing devoted to biological sequences retrieval and analysis. In U. Bastolla, M. Porto, H. E. Roman, and M. Vendruscolo, editors, *Structural approaches to sequence evolution: Molecules, networks, populations*, Biological and Medical Physics, Biomedical Engineering, pages 207–232. Springer Verlag, New York, 2007.

Y. Chen, Z. Du, H. Zhao, W. Fang, T. Liu, Y. Zhang, W. Zhang, and W. Qin. SPPUSM: An MS/MS spectra merging strategy for improved low-input and single-cell proteome identification. *Anal. Chim. Acta*, 1279:341793, Oct. 2023. ISSN 0003-2670, 1873-4324. doi: 10.1016/j.aca.2023.341793.

Y.-Y. Chen, S. Dasari, Z.-Q. Ma, L. J. Vega-Montoto, M. Li, and D. L. Tabb. Refining comparative proteomics by spectral counting to account for shared peptides and multiple search engines. *Anal. Bioanal. Chem.*, 404(4):1115–1125, Sept. 2012. ISSN 1618-2642, 1618-2650. doi: 10.1007/s00216-012-6011-x.

R. Craig and R. C. Beavis. A method for reducing the time required to match protein sequences with tandem mass spectra. *Rapid Commun. Mass Spectrom.*, 17(20):2310–2316, 2003. ISSN 0951-4198. doi: 10.1002/rcm.1198.

E. Debrie, L. Clement, and M. Malfait. TargetDecoy: Diagnostic Plots to Evaluate the Target Decoy Approach, 2024.

S. Degroeve and L. Martens. MS2PIP: a tool for MS/MS peak intensity prediction. *Bioinformatics*, 29 (24):3199–3203, Dec. 2013. ISSN 1367-4803, 1367-4811. doi: 10.1093/bioinformatics/btt544.

E. W. Deutsch, N. Bandeira, Y. Perez-Riverol, V. Sharma, J. J. Carver, L. Mendoza, D. J. Kundu, S. Wang, C. Bandla, S. Kamatchinathan, S. Hewapathirana, B. S. Pullman, J. Wertz, Z. Sun, S. Kawano, S. Okuda, Y. Watanabe, B. MacLean, M. J. MacCoss, Y. Zhu, Y. Ishihama, and J. A. Vizcaíno. The ProteomeXchange consortium at 10 years: 2023 update. *Nucleic Acids Res.*, 51(D1): D1539–D1548, Jan. 2023. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkac1040.

D. Fenyö and R. C. Beavis. A method for assessing the statistical significance of mass spectrometry-based protein identifications using general scoring schemes. *Anal. Chem.*, 75(4):768–774, Feb. 2003. ISSN 0003-2700. doi: 10.1021/ac0258709.

W. E. Fondrie and W. S. Noble. mokapot: Fast and flexible semisupervised learning for peptide detection. *J. Proteome Res.*, 20(4):1966–1971, Apr. 2021. ISSN 1535-3893, 1535-3907. doi: 10.1021/acs.jproteome.0c01010.

L. Gatto, J. Rainer, and S. Gibb. *PSMatch: Handling and Managing Peptide Spectrum Matches*, 2022.

N. Gehlenborg. UpSetR: A more scalable alternative to venn and euler diagrams for visualizing intersecting sets, 2019.

S. Gessulat, T. Schmidt, D. P. Zolg, P. Samaras, K. Schnatbaum, J. Zerweck, T. Knaute, J. Rechenberger, B. Delanghe, A. Huhmer, U. Reimer, H.-C. Ehrlich, S. Aiche, B. Kuster, and M. Wilhelm. Prosit: proteome-wide prediction of peptide tandem mass spectra by deep learning. *Nat. Methods*, 16(6): 509–518, June 2019. ISSN 1548-7091, 1548-7105. doi: 10.1038/s41592-019-0426-7.

Z. Gu, R. Eils, and M. Schlesner. Complex heatmaps reveal patterns and correlations in multidimensional genomic data. *Bioinformatics*, 32(18):2847–2849, Sept. 2016. ISSN 1367-4803, 1367-4811. doi: 10.1093/bioinformatics/btw313.

W. Huber, V. J. Carey, R. Gentleman, S. Anders, M. Carlson, B. S. Carvalho, H. C. Bravo, S. Davis, L. Gatto, T. Girke, R. Gottardo, F. Hahne, K. D. Hansen, R. A. Irizarry, M. Lawrence, M. I. Love, J. MacDonald, V. Obenchain, A. K. Oleś, H. Pagès, A. Reyes, P. Shannon, G. K. Smyth, D. Tenenbaum, L. Waldron, and M. Morgan. Orchestrating high-throughput genomic analysis with Bioconductor. *Nat. Methods*, 12(2):115–121, Feb. 2015. ISSN 1548-7091, 1548-7105. doi: 10.1038/nmeth.3252.

A. J. Izenman. Linear discriminant analysis. In A. J. Izenman, editor, *Modern Multivariate Statistical*

Techniques: Regression, Classification, and Manifold Learning, pages 237–280. Springer New York, New York, NY, 2008. ISBN 9780387781891. doi: 10.1007/978-0-387-78189-1_8.

L. Käll, J. D. Storey, M. J. MacCoss, and W. S. Noble. Posterior error probabilities and false discovery rates: two sides of the same coin. *J. Proteome Res.*, 7(1):40–44, Jan. 2008. ISSN 1535-3893. doi: 10.1021/pr700739d.

A. A. Klammer, X. Yi, M. J. MacCoss, and W. S. Noble. Improving tandem mass spectrum identification using peptide retention time prediction across diverse chromatography conditions. *Anal. Chem.*, 79 (16):6111–6118, Aug. 2007. ISSN 0003-2700. doi: 10.1021/ac070262k.

M. R. Lazear. Sage: An Open-Source Tool for Fast Proteomics Searching and Quantification at Scale. *J. Proteome Res.*, 22(11):3652–3659, Nov. 2023. ISSN 1535-3893, 1535-3907. doi: 10.1021/acs.jproteome.3c00486.

Y. Liang, H. Acor, M. A. McCown, A. J. Nwosu, H. Boekweg, N. B. Axtell, T. Truong, Y. Cong, S. H. Payne, and R. T. Kelly. Fully Automated Sample Processing and Analysis Workflow for Low-Input Proteome Profiling. *Anal. Chem.*, 93(3):1658–1666, Jan. 2021. ISSN 0003-2700, 1520-6882. doi: 10.1021/acs.analchem.0c04240.

A. Lin, T. Short, W. S. Noble, and U. Keich. Improving Peptide-Level mass spectrometry analysis via double competition. *J. Proteome Res.*, 21(10):2412–2420, Oct. 2022. ISSN 1535-3893, 1535-3907. doi: 10.1021/acs.jproteome.2c00282.

B. P. Lucey, W. A. Nelson-Rees, and G. M. Hutchins. Henrietta lacks, HeLa cells, and cell culture contamination. *Arch. Pathol. Lab. Med.*, 133(9):1463–1467, Sept. 2009. ISSN 0003-9985, 1543-2165. doi: 10.5858/133.9.1463.

N. D. Matsakis and F. S. Klock II. The rust language. In *ACM SIGAda Ada Letters*, volume 34, pages 103–104. ACM, 2014.

D. Mellacheruvu, Z. Wright, A. L. Couzens, J.-P. Lambert, N. A. St-Denis, T. Li, Y. V. Miteva, S. Hauri, M. E. Sardiu, T. Y. Low, V. A. Halim, R. D. Bagshaw, N. C. Hubner, A. Al-Hakim, A. Bouchard, D. Faubert, D. Fermin, W. H. Dunham, M. Goudreault, Z.-Y. Lin, B. G. Badillo, T. Pawson, D. Durocher, B. Coulombe, R. Aebersold, G. Superti-Furga, J. Colinge, A. J. R. Heck, H. Choi, M. Gstaiger, S. Mohammed, I. M. Cristea, K. L. Bennett, M. P. Washburn, B. Raught, R. M. Ewing, A.-C. Gingras, and A. I. Nesvizhskii. The CRAPome: a contaminant repository for affinity purification-mass spectrometry data. *Nat. Methods*, 10(8):730–736, Aug. 2013. ISSN 1548-7091, 1548-7105. doi: 10.1038/nmeth.2557.

M. Meloun and J. Militký. 2 - the exploratory and confirmatory analysis of univariate data. In M. Meloun and J. Militký, editors, *Statistical Data Analysis*, pages 25–71. Woodhead Publishing India, Jan. 2011. ISBN 9780857091093. doi: 10.1533/9780857097200.25.

A. I. Nesvizhskii. Protein identification by tandem mass spectrometry and sequence database searching. In R. Matthiesen, editor, *Mass Spectrometry Data Analysis in Proteomics*, pages 87–119. Humana Press, Totowa, NJ, 2007. ISBN 9781597452755. doi: 10.1385/1-59745-275-0:87.

R Core Team. R: A language and environment for statistical computing, 2024.

J. Rainer, A. Vicini, L. Salzer, J. Stanstrup, J. M. Badia, S. Neumann, M. A. Stravs, V. Verri Hernandes, L. Gatto, S. Gibb, and M. Witting. A modular and expandable ecosystem for metabolomics data annotation in R. *Metabolites*, 12(2), Feb. 2022. ISSN 2218-1989. doi: 10.3390/metabo12020173.

P. Sinitcyn, J. D. Rudolph, and J. Cox. Computational Methods for Understanding Mass Spectrometry–Based Shotgun Proteomics Data. *Annual Review of Biomedical Data Science*, 1(Volume 1, 2018): 207–234, July 2018. ISSN 2574-3414. doi: 10.1146/annurev-biodatasci-080917-013516.

H. Steen and M. Mann. The ABC's (and XYZ's) of peptide sequencing. *Nat. Rev. Mol. Cell Biol.*, 5 (9):699–711, Sept. 2004. ISSN 1471-0072. doi: 10.1038/nrm1468.

M. The, M. J. MacCoss, W. S. Noble, and L. Käll. Fast and Accurate Protein False Discovery Rates on Large-Scale Proteomics Data Sets with Percolator 3.0. *J. Am. Soc. Mass Spectrom.*, 27(11):1719–1727, Nov. 2016. ISSN 1044-0305, 1879-1123. doi: 10.1007/s13361-016-1460-7.

C. Vanderaa. *A principled approach and standardised software for mass spectrometry-based single-cell proteomics data analysis.pdf*. PhD thesis, UCLouvain, Jan. 2024.

W. N. Venables and B. D. Ripley. *Modern applied statistics with S*. Statistics and Computing. Springer, New York, NY, 4 edition, Mar. 2013. ISBN 9780387954578.

H. Wickham. ggplot2: Elegant graphics for data analysis, June 2016.

H. Wickham, R. François, L. Henry, K. Müller, and D. Vaughan. dplyr: A grammar of data manipulation, 2023.

H. Wickham, D. Vaughan, and M. Girlich. tidyr: Tidy messy data, 2024.

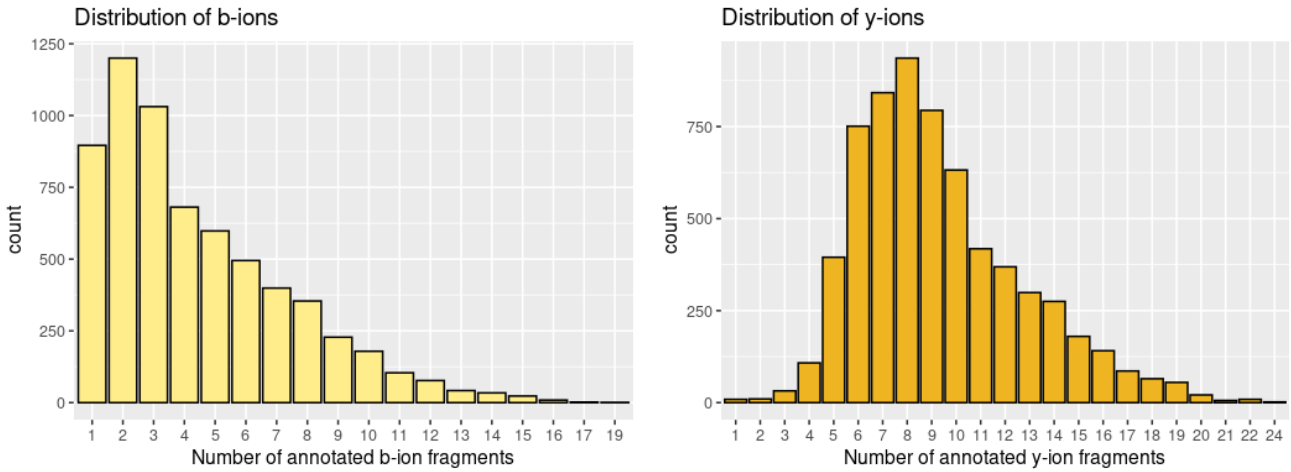
M. Wojtkiewicz, J. Wiederin, P. Ciborowski, and P. Olszowy. Comparison of proteome discoverer and PEAKS studio for phosphoproteome analysis. *J. Biomol. Tech.*, 24, May 2013. ISSN 1524-0215.

Y. Zhu, P. D. Piehowski, R. Zhao, J. Chen, Y. Shen, R. J. Moore, A. K. Shukla, V. A. Petyuk,

M. Campbell-Thompson, C. E. Mathews, R. D. Smith, W.-J. Qian, and R. T. Kelly. Nanodroplet processing platform for deep and quantitative proteome profiling of 10-100 mammalian cells. *Nat. Commun.*, 9(1):882, Feb. 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-03367-w.

6 Appendices

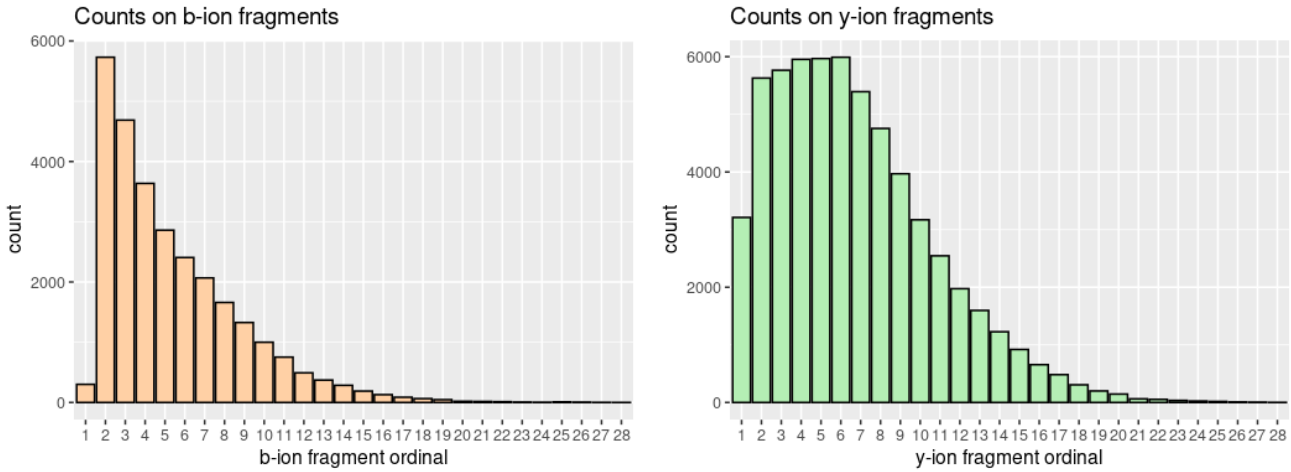
6.1 Appendix A: additional figures



(a) On b-ions

(b) On y-ions

Figure 49: Distribution of annotated fragments among single-cell spectra.



(a) On b-ions

(b) On y-ions

Figure 50: Counts on fragment indexes in single-cell spectra.

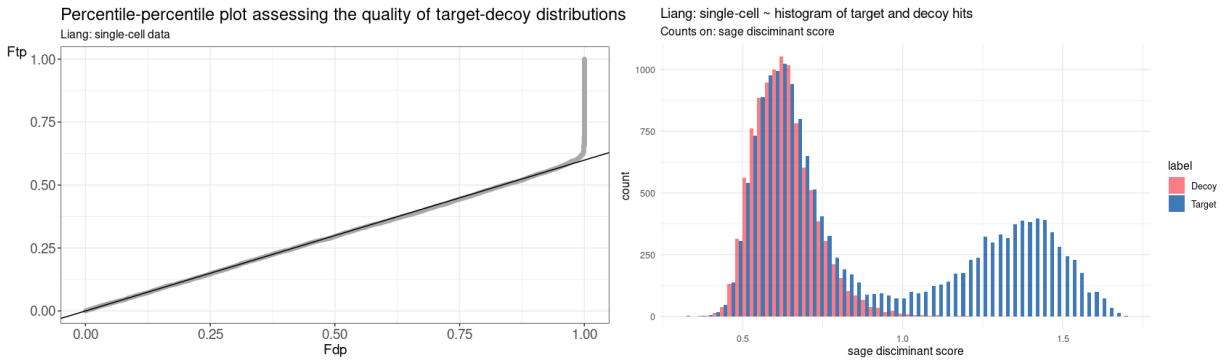


Figure 51: PP-plot and target-decoy distribution for Liang: single-cell dataset.

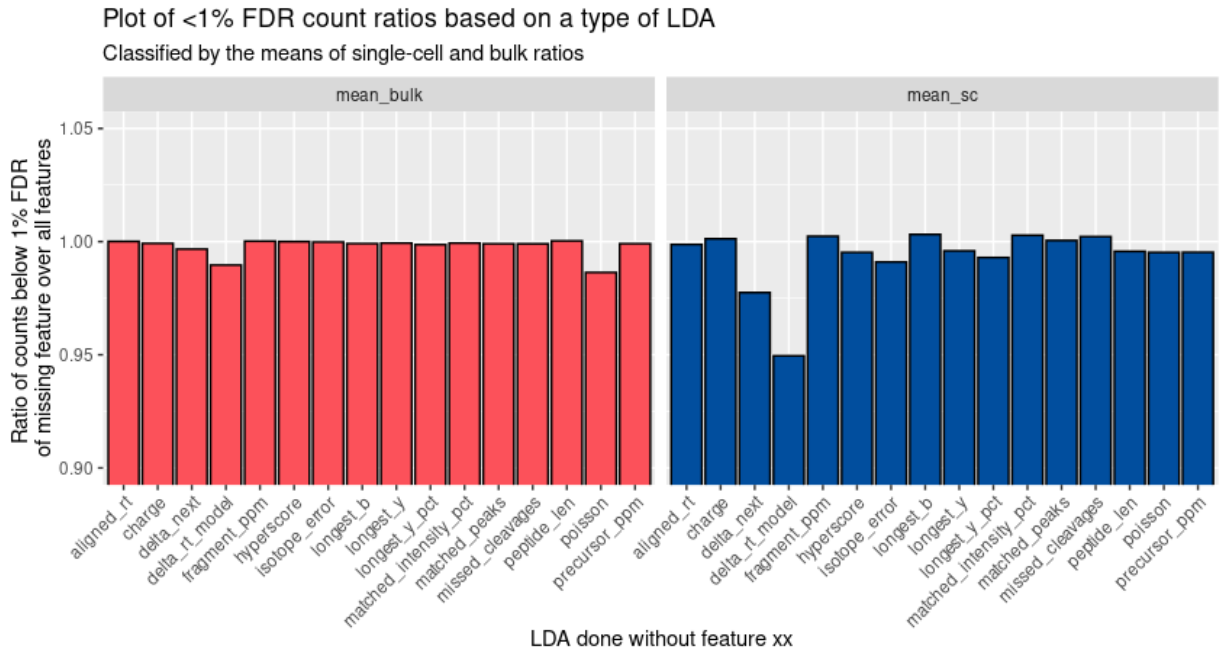
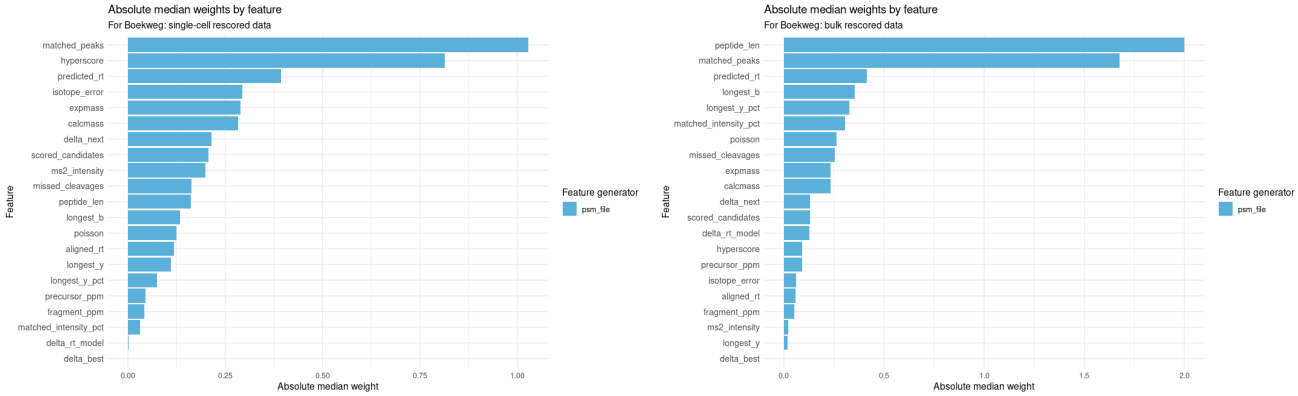


Figure 52: Overview of the impact of Sage's initial 16 features on the number of confidently identified PSMs. A ratio out of the number of PSMs based on the feature removed from LDA is shown. A low value means a high loss in PSMs and thus a greater importance of the removed feature. Analysis done on the Boekweg dataset.



(a) For single-cell

(b) For bulk

Figure 53: Weights of Sage's PSM file after rescoring with MS²Rescore on the Boekweg dataset.

6.2 Appendix B: additional tables

Table 5: Features generated by DeepLC

Feature	Description	Source
predicted_retention_time_best	Predicted retention time for the precursor peak	DeepLC
observed_retention_time_best	Observed retention time for the precursor peak	DeepLC
rt_diff_best	Difference between observed and predicted retention time for the precursor peak	DeepLC
rt_diff	Difference between observed and predicted retention time	DeepLC
predicted_retention_time	Predicted retention time	DeepLC
observed_retention_time	Observed retention time	DeepLC

Table 6: Features generated by MS²PIP - part 1

Feature	Description	Source
cos_iony	Cosine similarity on y-ions	MS²PIP
cos_ionb	Cosine similarity on b-ions	MS²PIP
cos	Cosine similarity on b- and y-ions	MS²PIP
dotprod_iony	Dot product on y-ions	MS²PIP
dotprod_ionb	Dot product on b-ions	MS²PIP
dotprod	Dot product on b- and y-ions	MS²PIP
iony_std_abs_diff	Standard deviation of the absolute differences (y-ions only)	MS²PIP
iony_mean_abs_diff	Mean of the absolute differences (y-ions only)	MS²PIP
iony_abs_diff_Q3	Quantile 3 of the absolute differences (y-ions only)	MS²PIP
iony_abs_diff_Q2	Quantile 2 of the absolute differences (y-ions only)	MS²PIP
iony_abs_diff_Q1	Quantile 1 of the absolute differences (y-ions only)	MS²PIP
iony_max_abs_diff	Maximum absolute difference (y-ions only)	MS²PIP
iony_min_abs_diff	Minimum absolute difference (y-ions only)	MS²PIP
ionb_std_abs_diff	Standard deviation of the absolute differences (b-ions only)	MS²PIP
ionb_mean_abs_diff	Mean of the absolute differences (b-ions only)	MS²PIP
ionb_abs_diff_Q3	Quantile 3 of the absolute differences (b-ions only)	MS²PIP
ionb_abs_diff_Q2	Quantile 2 of the absolute differences (b-ions only)	MS²PIP
ionb_abs_diff_Q1	Quantile 1 of the absolute differences (b-ions only)	MS²PIP
ionb_max_abs_diff	Maximum absolute difference (b-ions only)	MS²PIP
ionb_min_abs_diff	Minimum absolute difference (b-ions only)	MS²PIP
std_abs_diff	Standard deviation of the absolute differences	MS²PIP
mean_abs_diff	Mean of the absolute differences	MS²PIP
abs_diff_Q3	Quantile 3 of the absolute differences	MS²PIP
abs_diff_Q2	Quantile 2 of the absolute differences	MS²PIP
abs_diff_Q1	Quantile 1 of the absolute differences	MS²PIP
max_abs_diff	Maximum absolute difference	MS²PIP
min_abs_diff	Minimum absolute difference	MS²PIP
max_abs_diff_iontype	Ion type with maximum absolute difference	MS²PIP
min_abs_diff_iontype	Ion type with minimum absolute difference	MS²PIP
iony_mse	Mean square error on y-ions	MS²PIP
ionb_mse	Mean square error on b-ions	MS²PIP
spec_mse	Mean square error on b- and y-ions	MS²PIP
iony_spearman	Spearman correlation on b- and y-ions	MS²PIP
ionb_spearman	Spearman correlation on b- and y-ions	MS²PIP
spec_spearman	Spearman correlation on b- and y-ions	MS²PIP

Table 7: Features generated by MS²PIP - part 2

Feature	Description	Source
iony_pearson	Pearson correlation coefficient on y-ions	MS²PIP
ionb_pearson	Pearson correlation coefficient on b-ions	MS²PIP
spec_pearson	Pearson correlation coefficient on b- and y-ions	MS²PIP
cos_iony_norm	Cosine similarity on y-ions (log2-normalized spectrum)	MS²PIP
cos_ionb_norm	Cosine similarity on b-ions (log2-normalized spectrum)	MS²PIP
cos_norm	Cosine similarity on b- and y-ions (log2-normalized spectrum)	MS²PIP
dotprod_iony_norm	Dot product on y-ions (log2-normalized spectrum)	MS²PIP
dotprod_ionb_norm	Dot product on b-ions (log2-normalized spectrum)	MS²PIP
dotprod_norm	Dot product on b- and y-ions (log2-normalized spectrum)	MS²PIP
iony_std_abs_diff_norm	Standard deviation of the absolute differences (y-ions only) (log2-normalized spectrum)	MS²PIP
iony_mean_abs_diff_norm	Mean of the absolute differences (y-ions only) (log2-normalized spectrum)	MS²PIP
iony_abs_diff_Q3_norm	Quantile 3 of the absolute differences (y-ions only) (log2-normalized spectrum)	MS²PIP
iony_abs_diff_Q2_norm	Quantile 2 of the absolute differences (y-ions only) (log2-normalized spectrum)	MS²PIP
iony_abs_diff_Q1_norm	Quantile 1 of the absolute differences (y-ions only) (log2-normalized spectrum)	MS²PIP
iony_max_abs_diff_norm	Maximum absolute difference (y-ions only) (log2-normalized spectrum)	MS²PIP
iony_min_abs_diff_norm	Minimum absolute difference (y-ions only) (log2-normalized spectrum)	MS²PIP
ionb_std_abs_diff_norm	Standard deviation of the absolute differences (b-ions only) (log2-normalized spectrum)	MS²PIP
ionb_mean_abs_diff_norm	Mean of the absolute differences (b-ions only) (log2-normalized spectrum)	MS²PIP
ionb_abs_diff_Q3_norm	Quantile 3 of the absolute differences (b-ions only) (log2-normalized spectrum)	MS²PIP
ionb_abs_diff_Q2_norm	Quantile 2 of the absolute differences (b-ions only) (log2-normalized spectrum)	MS²PIP
ionb_abs_diff_Q1_norm	Quantile 1 of the absolute differences (b-ions only) (log2-normalized spectrum)	MS²PIP
ionb_max_abs_diff_norm	Maximum absolute difference (b-ions only) (log2-normalized spectrum)	MS²PIP
ionb_min_abs_diff_norm	Minimum absolute difference (b-ions only) (log2-normalized spectrum)	MS²PIP
std_abs_diff_norm	Standard deviation of the absolute differences (log2-normalized spectrum)	MS²PIP
mean_abs_diff_norm	Mean of the absolute differences (log2-normalized spectrum)	MS²PIP
abs_diff_Q3_norm	Quantile 3 of the absolute differences (log2-normalized spectrum)	MS²PIP
abs_diff_Q2_norm	Quantile 2 of the absolute differences (log2-normalized spectrum)	MS²PIP
abs_diff_Q1_norm	Quantile 1 of the absolute differences (log2-normalized spectrum)	MS²PIP
max_abs_diff_norm	Maximum absolute difference (log2-normalized spectrum)	MS²PIP
min_abs_diff_norm	Minimum absolute difference (log2-normalized spectrum)	MS²PIP
iony_mse_norm	Mean square error on y-ions (log2-normalized spectrum)	MS²PIP
ionb_mse_norm	Mean square error on b-ions (log2-normalized spectrum)	MS²PIP
spec_mse_norm	Mean square error on b- and y-ions (log2-normalized spectrum)	MS²PIP
iony_pearson_norm	Pearson correlation coefficient on y-ions (log2-normalized spectrum)	MS²PIP
ionb_pearson_norm	Pearson correlation coefficient on b-ions (log2-normalized spectrum)	MS²PIP
spec_pearson_norm	Pearson correlation coefficient on b- and y-ions (log2-normalized spectrum)	MS²PIP

Table 8: Computational time of the programs used to rescore.

Time	Sage[a]	MS²Rescore	Custom LDA[b]
Single-cell	6-20 sec	20-30 min	40-60 min
Bulk	20-60 sec	1.5-3h	3-6h

[a] With 1 static modification and 2 variable modifications
[b] Without parallelization

6.3 Appendix C: Additional code

6.3.1 Configuration files

6.3.1.1 Configuration file for Boekweg: single-cell data.

```
{
  "database": {
    "bucket_size": 8192,
    "enzyme": {
      "missed_cleavages": 2,
      "min_len": 8,
      "max_len": 30,
      "cleave_at": "KR",
      "restrict": "P"
    },
    "fragment_min_mz": 150,
    "fragment_max_mz": 2000,
    "peptide_min_mass": 500,
    "peptide_max_mass": 5000,
    "ion_kinds": ["b", "y"],
    "min_ion_index": 2,
    "max_variable_mods": 2,
     "static_mods": {
      "C": 57.0215
    },
    "variable_mods": {
      "M": 15.994,
      "[": 42.0
    },
    "decoy_tag": "rev_",
    "generate_decoys": true,
    "fasta": "../boekweg/fasta/UP000005640.fasta"
  },
  "precursor_tol": {
    "ppm": [-10, 10]
  },
  "fragment_tol": {
    "ppm": [-10, 10]
```

```
  },
  "isotope_errors": [0, 2],
  "deisotope": true,
  "min_peaks": 15,
  "max_peaks": 150,
  "max_fragment_charge": 1,
  "min_matched_peaks": 4,
  "report_psms": 1,
  "predict_rt": true,
  "output_directory": "/home/gdeflandre/sage_result/",
  "mzml_paths": [
    "../boekweg/data/raw/D19_15um30cm_SC1.mzML",
    "../boekweg/data/raw/D19_15um30cm_SC2.mzML",
    "../boekweg/data/raw/D19_15um30cm_SC3.mzML",
    "../boekweg/data/raw/D19_15um30cm_SC4.mzML",
    "../boekweg/data/raw/D19_15um30cm_SC5.mzML"]
}
```

6.3.1.2 Configuration file for Boekweg: bulk data.

```
{
  "database": {
    "bucket_size": 8192,
    "enzyme": {
      "missed_cleavages": 2,
      "min_len": 8,
      "max_len": 30,
      "cleave_at": "KR",
      "restrict": "P"
    },
    "fragment_min_mz": 150,
    "fragment_max_mz": 2000,
    "peptide_min_mass": 500,
```

```
  "peptide_max_mass": 5000,
  "ion_kinds": ["b", "y"],
  "min_ion_index": 2,
  "max_variable_mods": 2,
   "static_mods": {
    "C": 57.0215
  },
  "variable_mods": {
    "M": 15.994
    "[": 42.0
  },
  "decoy_tag": "rev_",
  "generate_decoys": true,
  "fasta": "../boekweg/fasta/UP000005640.fasta"
},
"precursor_tol": {
  "ppm": [-10, 10]
},
"fragment_tol": {
  "ppm": [-10, 10]
},
"isotope_errors": [0, 2],
"deisotope": true,
"min_peaks": 15,
"max_peaks": 150,
"max_fragment_charge": 1,
"min_matched_peaks": 4,
"predict_rt": true,
"output_directory": "/home/gdeflandre/sage_result/",
"mzml_paths": [
  "../boekweg/data/Hela_bulk/OR11_20160122_PG_HeLa_CVB3_CT_A.mzML",
  "../boekweg/data/Hela_bulk/OR11_20160122_PG_HeLa_CVB3_CT_B.mzML",
  "../boekweg/data/Hela_bulk/OR11_20160122_PG_HeLa_CVB3_CT_C.mzML"]
```

6.3.1.3 Configuration file for Liang: single-cell data.

```
{
  "database": {
    "bucket_size": 8192,
    "enzyme": {
      "missed_cleavages": 2,
      "min_len": 7,
      "max_len": 30,
      "cleave_at": "KR",
      "restrict": "P"
    },
    "fragment_min_mz": 150,
    "fragment_max_mz": 2000,
    "peptide_min_mass": 500,
    "peptide_max_mass": 5000,
    "ion_kinds": ["b", "y"],
    "min_ion_index": 2,
    "max_variable_mods": 3,
     "static_mods": {
      "C": 57.0215
    },
    "variable_mods": {
      "M": 15.994,
      "[": 42.0
    },
    "decoy_tag": "rev_",
    "generate_decoys": true,
    "fasta": "../fasta/UP000005640.fasta"
  },
  "precursor_tol": {
    "ppm": [-20, 20]
  },
```

```
  "fragment_tol": {
    "ppm": [-20, 20]
  },
  "isotope_errors": [0, 2],
  "deisotope": true,
  "min_peaks": 15,
  "max_peaks": 150,
  "max_fragment_charge": 1,
  "min_matched_peaks": 4,
  "predict_rt": true,
  "output_directory": "/home/gdeflandre/sage_result/PXD021882/cell1/",
  "mzml_paths": [
  "../data/PXD021882/HeLa_1cell_E12.mzML",
  "../data/PXD021882/HeLa_1cell_F8.mzML",
  "../data/PXD021882/HeLa_1cell_E14.mzML"]
```

6.3.1.4 Configuration file for Liang: 500 cells data.

```
{
  "database": {
    "bucket_size": 8192,
    "enzyme": {
      "missed_cleavages": 2,
      "min_len": 7,
      "max_len": 30,
      "cleave_at": "KR",
      "restrict": "P"
    },
    "fragment_min_mz": 150,
    "fragment_max_mz": 2000,
    "peptide_min_mass": 500,
    "peptide_max_mass": 5000,
    "ion_kinds": ["b", "y"],
```

```
  "min_ion_index": 2,
  "max_variable_mods": 3,
   "static_mods": {
    "C": 57.0215
  },
  "variable_mods": {
    "M": 15.994,
    "[": 42.0
  },
  "decoy_tag": "rev_",
  "generate_decoys": true,
  "fasta": "../fasta/UP000005640.fasta"
},
"precursor_tol": {
  "ppm": [-20, 20]
},
"fragment_tol": {
  "ppm": [-20, 20]
},
"isotope_errors": [0, 2],
"deisotope": true,
"min_peaks": 15,
"max_peaks": 150,
"max_fragment_charge": 1,
"min_matched_peaks": 4,
"predict_rt": true,
"output_directory": "/home/gdeflandre/sage_result/PXD021882/cell500/",
"mzml_paths": [
  "../data/PXD021882/HeLa_500cell_J5.mzML",
  "../data/PXD021882/HeLa_500cell_K16.mzML",
  "../data/PXD021882/HeLa_500cell_K8.mzML"]
```

Single-cell proteomics (SCP) has emerged as a powerful tool for elucidating cellular heterogeneity, offering opportunities beyond traditional bulk sample analysis. However, the application of current peptide identification algorithms crafted for bulk samples may lead to false discoveries in SCP. Challenges such as reduced peak counts, lower peak intensities, and degraded signal-to-noise ratios raise the question : do current peptide scoring methods in search engines adequately perform in the context of SCP ?

To address these limitations, we explore the effectiveness of database search engines and rescoring tools with the use of Bioconductor packages *PSMatch* and *Spectra*. Rescoring tools take profit of as many mass spectrometry-based features as possible, such as spectral characteristics and retention time models, which can be particularly relevant to mitigate the poor quality of SCP spectra. We used *MS²Rescore* to generate more features, *Mokapot* to rescore the SCP peptides as well as the aforementioned packages to assess the efficiency of rescoring tools and potentially improve current scoring methods in the context of SCP.

While the need for identification optimization at search engine level is not identified, optimization of rescoring tools is endorsed : our findings demonstrate a significant increase in confidently identified peptides upon rescoring. In addition, we suggest a four-step methodology to evaluate the usefulness of current and new potential features. Finally, our results shed light on the differences between bulk and single-cell samples whilst providing insights that can inform more accurate and reliable data interpretation in the context of SCP.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
Faculté des bioingénieurs
Croix du Sud, 2 bte L7.05.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/agro**