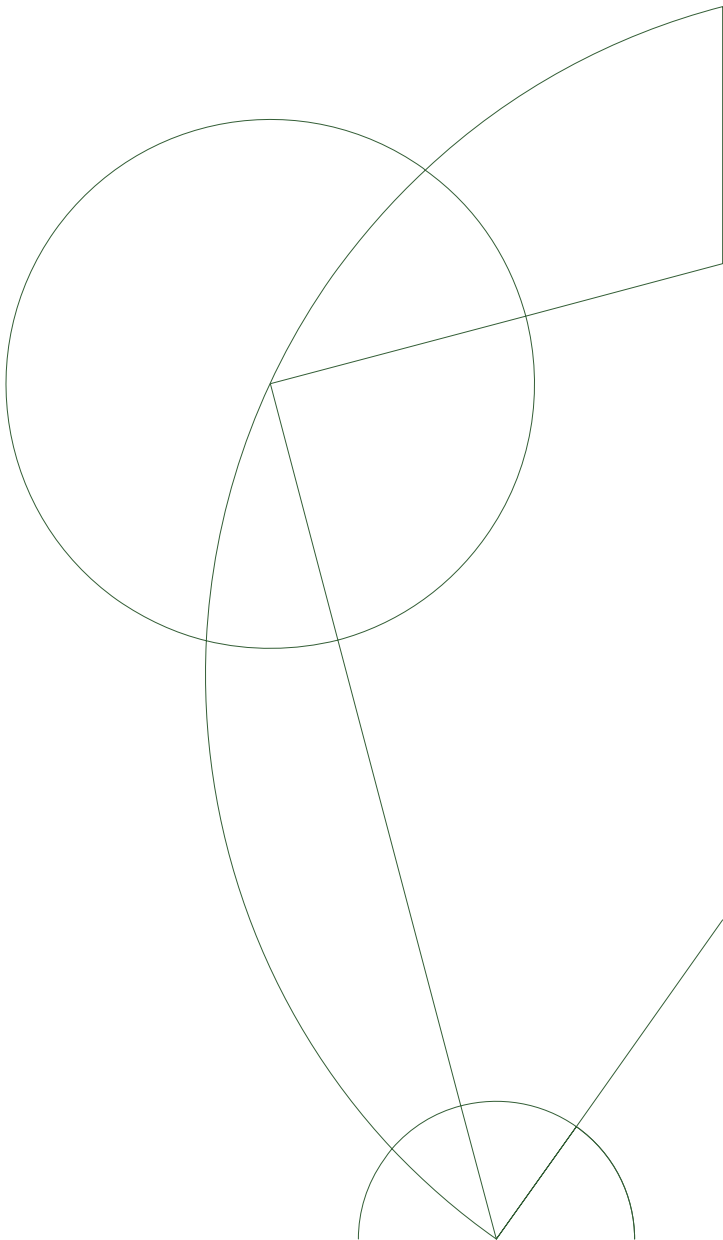
FACULTY OF SCIENCE
UNIVERSITY OF COPENHAGEN

Bioinformatics Master Thesis

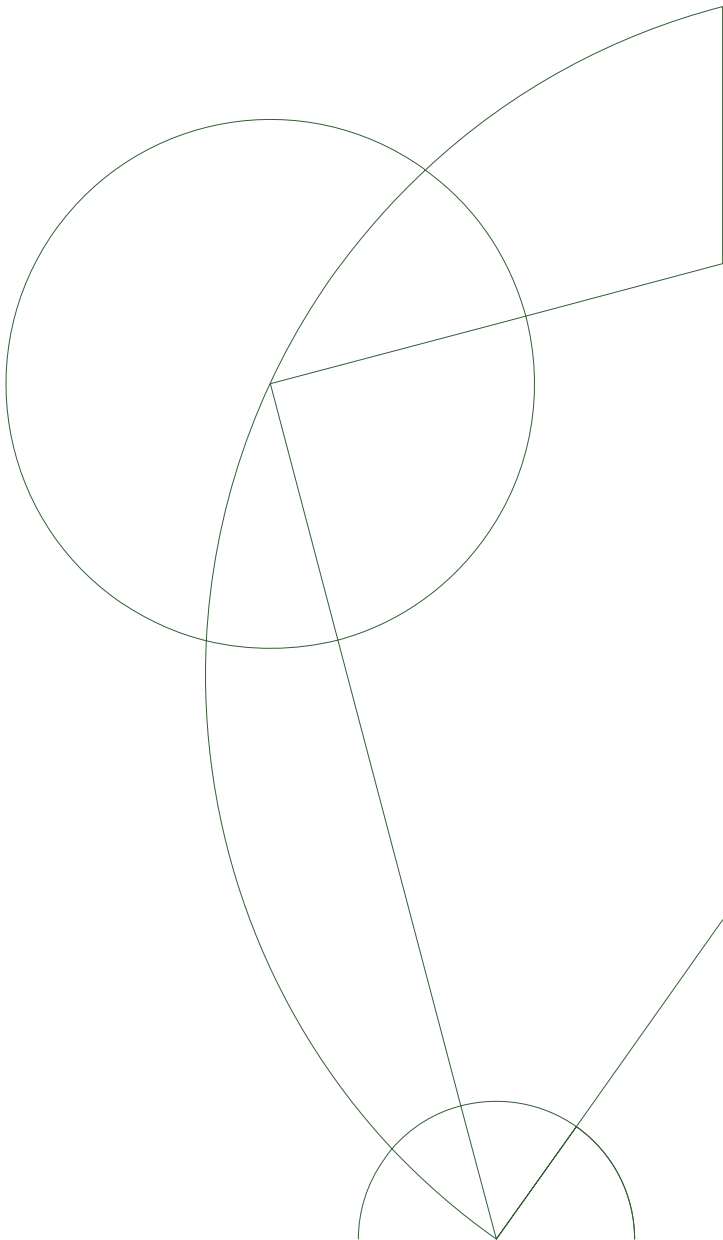
Development of a label-free quantification proteomics pipeline

Antonio Ortega Jiménez <ntoniohu@gmail.com>



Supervisors

Thomas Hamelryck <thamelry@gmail.com>
Mathias F. Gruber <mafg@novozymes.com>



Contents

Abbreviations

FT-ICR Fourier Transform Ion Cyclotron Resonance

HPLC High pressure liquid chromatography

IT Ion Trap

m/z mass charge ratio

MS Mass Spectrometry

MS/MS Tandem mass spectrometry

MS1 first tandem MS analyzer

MS2 second tandem MS analyzer

NZ Novozymes A/S

Q Quadrupole

RT Retention time

SC Spectral Counting

TOF Time of Flight

TPP Trans Proteomic Pipeline

XIC Extracted Ion Current

Preface

Introduction

0.1 Aminoacids and proteins

Proteins represent the last link in the central dogma of biology, where information encoded in DNA, is transcribed to RNA for posterior translation into proteins at the ribosome.

Proteins are made up of 20 basic units, called aminoacids. All aminoacids share a common chemical structure, where a carbon atom (C_α) is covalently bonded to a hydrogen atom, a carboxyl group, an amino group, and last but not least, a radical, also called side chain of the aminoacid. The side chain differs between aminaocids and generates them from each other. A slight deviation from this pattern exists in proline, where the radical is bound to the nitrogen atom, making it an iminoacid. Even though the side chains are all different, they can be classified into four different groups: aliphatic, polar, positively charged and negatively charged (see figure 1).

Two aminoacids are joined together through the formation of a peptidic (covalent) bond between them. Such a linkage is formed by removal of the elements of water (dehydration) from the α-carboxyl group of one amino acid and the α-amino group of another [1]. The remaining α-amino and α-carboxyl groups are available for linkage to other aminoacids, and in this way peptidic chains or peptides can be created.

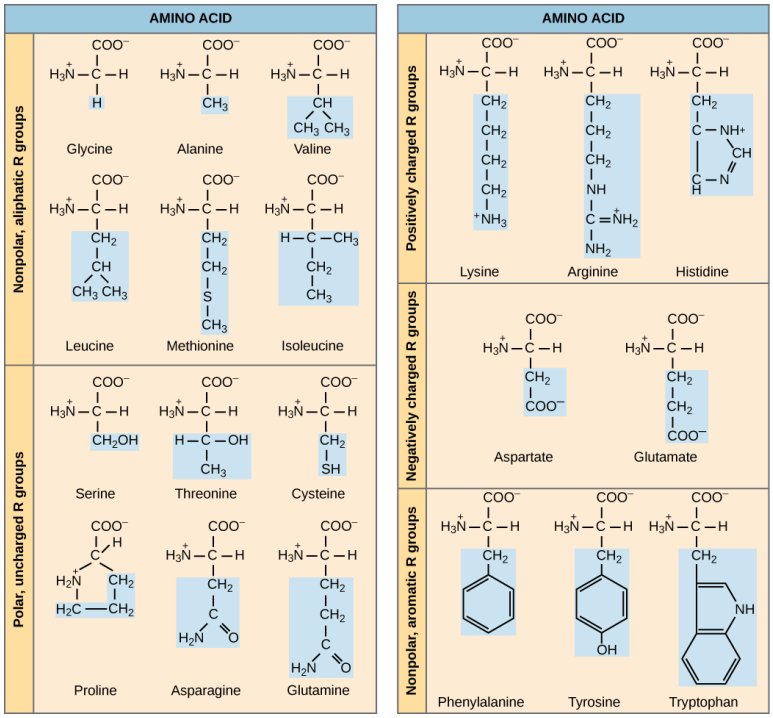


Figure 1 CAPTION AND REFERENCE

While there are 20 basic units that constitute the majority of naturally observable proteins, their side chains can be modified both by physiological processes and by experimental procedures cite. One frequent instance of such modifications is the oxidation of methionine.

0.2 The protein-focused biotechnology industry

Proteins carry out most of the cell's molecular functions, they work as molecular agents that can perform an extremely wide range of tasks. The advent of biotechnology has sought to take advantage of this power, either by using proteins as present in natural conditions (wild type) or engineered by humans. This potential economic activity is carried out by several biotech companies, including Novozymes A/S (NZ).

NZ is a company whose line of business consists of the development of enzymatic products performing chemical transformations in different industrial processes. The application of these products, instead of conventional chemical-based solutions, has the advantage that they require less chemical substances, potentially simplifying industrial processes, reducing their costs and their environmental impact. Notorious examples of such applications include waste-water treatment, household care and the baking industry.

The advancement of the way NZ does protein research is thus key to place the organization ahead of its competitors. The refinement of the currently used tools and the development of new ones could be of great significance for the company.

Protein research can be approached from different angles. This thesis exploited the combination of mass spectrometry (MS) and proteomics workflows (see chapter 1) for the qualitative and quantitative characterization of protein samples.

0.3 Objectives of the Thesis

In line with the goal of making NZ more competitive, this project aimed at the following objectives:

1. Develop an open-source, Linux based and easily deployable pipeline for the analysis of MS data, starting at the raw high-throughput data files and ending in the biological interpretation of the results.
2. Evaluate this pipeline with a benchmark dataset to assess if the pipeline is able to reflect the biological phenomena captured in the data.
3. Establish a label-free quantification probabilistic model that provides

relative abundance estimates and a measurement of their uncertainty based on the available data.

0.4 Structure of the Thesis

An overview over the MS and following computational data analysis steps is presented in 1. The pipeline development and its benchmark are explained in chapters 2 and 3, while the modelling problem is introduced in chapter 4. Finally, a conclusion of the work is given in chapter 4.

Chapter 1

Mass spectrometry and shotgun proteomics overview

Life systems consist of complex systems, meaning their behaviour cannot be easily explained by analyzing the individual elements alone. Moreover, they present multiple layers of complexity, given by the nature of the elements that make it up. The layer provided by proteins is one of them, and its study is called proteomics. It is a complex layer because thousands of different proteins can be present in a single cell at any time, and their exact composition and quantities constantly change, responding to the stimuli of the surrounding environment. The study of the protein-specific complexity is called proteomics. With proteomics, one endeavors to infer the protein composition of a sample, and eventually quantify its protein amounts.

It may be useful to divide the existing approaches into two types of paradigms: top-down and bottom-up. In the top-down paradigm, intact proteins are directly used for the analysis. In the bottom-up paradigm, the proteins are first cleaved into smaller parts, and these parts are then used for identification, characterization, and quantification. These smaller parts are called peptides. [2] CITE 1.6.1 COMPUTATIONAL METHODS. Such peptides acquire physicochemical properties fitting the requirements of the down-

stream analytical methods, mainly mass spectrometry (MS), which performs the data acquisition. The bottom-up paradigm is most often used because peptides are much more suitable to analysis by mass spectrometry, as explained in 1.2.3. The top-down paradigm will be ignored in the rest of the manuscript.

MS is performed by means of a mass spectrometer, an ensemble of pieces of equipment that can acquire mass measurements for plenty of sample components. A detailed explanation of the sample processing required prior to MS is given in section , while an overview on mass spectrometers is given in section . The result of the MS analysis is a dataset that, with adequate computational analysis tools, is enough to perform the inference steps required to gather knowledge about the original protein sample. These inference steps can be condensed to the peptide and protein inference problems, explained in section . A third computational problem needs to be solved if quantitative, and not just qualitative information, is to be gained from the experiment. This is the quantification problem, explained in section CITE.

A summary of the bottom-up approach MS analytical pipeline is provided in the rest of the chapter. It can be divided into two main steps:

1. MS analysis and data generation. Sections 1.1 to 1.3.
2. Computational analysis of data. Sections 1.4 to 1.7.

1.1 Sample processing

An MS experiment starts with the generation of protein mixture samples. They are first separated in order to sort the proteins via physicochemical criteria. This is most frequently carried out via SDS-PAGE based on mass or isoelectric point. Once the electrophoresis is completed, protein bands can

be excised from the gel. Each band will contain a subset, or even only one of the proteins originally available, thus making the downstream analysis simpler [2] REFERENCE COMPUTATIONAL METHODS CHAPTER 2

The reduced complexity protein mix is extracted in a denaturalized state so as to remove biases due to the divergent properties acquired by folded proteins. Then, proteins are subjected to digestion with specific enzymes, that can cut the aminoacidic chains following a predictable pattern. Trypsin is the most frequently used for this task, which cuts peptidic bonds whenever a positively charged residue, either Lysine (K) or Arginine (R), lies on the carboxyl side of the peptidic bond. Even though enzymes are very specific, the cleaving process is far from perfect, as there could be: [2] COMPUTATIONAL METHODS 3.2

1. Missed cleavages
2. Unsuspected cleavages during the maturation/life cycle of the protein.
3. Unexpected cleavages occurring either in the wet-lab procedure of the proteolytic treatment.
4. Naturally occurring, intentionally or unintentionally induced chemical modifications.

Item 1 can happen due to steric inaccesability or the presence of specific aminoacids that can weaken the enzyme's function. This is the case of Trypsin whenever the residue on the other side of the peptidic bond is Proline. This variability, though limited, needs to be taken care of in downstream analysis, as it could introduce biases in peptide observability. The other

The result of this process is a mix of peptides following a length distribution given by the cleavage sites frequency and each protein's aminoacidic com-

position. For Trypsin, the average peptide length is 10 residues, as roughly 1/10 residues are either R or K. As explained in 1.2.3, this length distribution is fitted to the resolution of the MS analyzer, thus optimizing the throughput of the method. An overview over mass spectrometers follows.

1.2 The mass spectrometer

The mass spectrometer consists of three main parts: an ion source, a mass analyzer, and a detector (see figure 1.1).

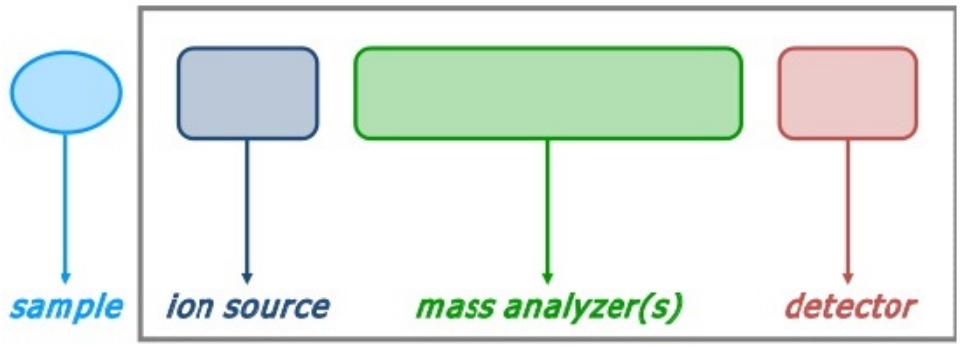


Figure 1.1 Schematic view of a mass spectrometer. Taken from [1]

1.2.1 The ion source

All mass spectrometers exploit the physical properties of mass and electric charge exhibited by the analyzed components. Ionization of the analytes is absolutely essential prior to any measurement, as analytes left uncharged will be unobservable to the equipment. This step is performed in the ion source [2] CITE 5.1 COMPUTATIONAL METHODS. The most frequent ionization methods in proteomics are Matrix-Assisted Laser Desorption-

[1] https://www.slideshare.net/joachimjacob/bits-introduction-to-mass-spec-data-generation

Ionization (MALDI) and Electro Spray Ionization (ESI) CITE. Most peptides ionized by MALDI will acquire a single charge, whereas ESI can provide multiple charges (+2, +3, etc) too. Thus, the charge exhibited by an ion is not obvious when produced via ESI. Moreover, ESI can be run online with the right robotic equipment, while MALDI demands waiting time for vacuum generation. Finally, due to the chemical nature of the matrix components, MALDI ionizes more easily peptides containing aminoacids featuring aromatic rings (PYW), thus introducing a bias. Bias in ESI is less predictable. This is known as the competitive ionization problem. REF ALL THIS

The acquired charge yields a mass/charge (m/z) ratio, a property that can be applied in the downstream component separation and measurement steps.

1.2.2 The mass analyzer

The plethora of ion separation methods is reflected upon the range of different analyzers available, mainly time of flight (TOF), Ion trap (IT) and quadrupole (Q). These apply different principles to perform the same task: separation (analysis) of the ion mix by the m/z ratio.

Moreover, two other analyzers exist which combine mass analysis with intensity measurement. These are Fourier Transform Ion Cyclotron Resonance (FT-ICR) and Orbitrap. They both register cylotron resonance frequencies that are Fourier transformed into the spectrum space. Remarkably, FT-ICR exhibits great resolving power, at the cost of high maintenance costs and difficult operability.

1.2.3 The detector

Detectors measure the intensity of an incoming ion signal. The ion's m/z ratio is known thanks to the previous mass analysis step. Performed for enough m/z ratios, the detector can produce a MS spectrum, which shows the intensity of ion current over an m/z range. Some topics in signal detection in MS need to be discussed.

On the one hand, the precision of the signal measurement is given by its mass resolution It is conventionally defined as the closest distinguishable separation between two peaks of equal height and width [3]. The resolution decreases as the m/z ratio increases because small increments in the m/z ratio become negligible at high m/z ratios. This is one of the reasons why proteins are better fit for analysis when digested into peptides, as m/z are reduced, thus increasing the mass resolution.

On the other hand, due to the natural occurrence of isotopes, particularly ^{13}C, the same peptide will induce the measurement of several signals with very close m/z values. They constitute the isotopic envelope of the ion SEE FIGURE, and represent the signal created by peptides containing an increasing number of ^{13}C atoms. Every time a ^{12}C is replaced by ^{13}C, the mass increases by 1 Da. Even though the natural abundance of ^{13}C is 1.1 %, the sheer number of carbon atoms in a peptide makes it likely that at least one or even more carbon atoms will be ^{13}C, eventually driving the pure ^{12}C signal to comparatively small intensity values, and down to intensities below the background noise. Such event can be problematic if it entails that the ^{13}C peak is confused for the ^{12}C peak.

The resolution achieved by modern equipment allows for the distinction of each individual signal in most isotopic envelopes. Remarkably, the separation across peaks in the envelope can be used to infer the charge of the

peptide, as increases of 1 Da at charge 1 will induce a separation of 1 m/z, while at charge 2 it will be $1/2 = 0.5$ m/z, at 3 $1/3 = 0.33$ m/z, and so on.

It is up to the MS technician to decide on the best pieces of equipment according to their availability and particularities of the dataset.

1.3 Mass spectrometry workflows

The MS workflow diverges based on the simplicity of the original protein sample. When it consisted of a single protein, Peptide Mass Fingerprinting (PMF) is used, otherwise tandem MS (MS/MS) shall be performed.

1.3.1 Peptide Mass Fingerprint (PMF)

If the original sample was known to contain a single protein, PMF, or *protein-centric* proteomics, is conducted. In PMF, the mixture of peptides can be already transferred to the spectrometer, where a spectrum containing a peak for every m/z ratio present in the ionized peptide mix will be recorded. The resulting spectra can be considered a pattern, or fingerprint, of the peptides making up the original protein, thus enabling the identification of the original protein.

1.3.2 High pressure liquid chromatography-Tandem MS

High pressure liquid chromatography (HPLC)

If presented with the problem of analyzing a mixture of proteins, the capacities of mass spectrometers are easily overwhelmed by a too complex mixture, resulting in the analysis of only a minor part of the total protein of the sample. This can be surmounted by splitting the initial sample into

fractions, and using a series, or tandem, of spectrometers in the analysis. The spectrometers are used to analyze each obtained fraction separately, using different schemes.

Fractionation is achieved by different methods of separation CITE 1.7, most commonly via HPLC methods, like reverse phase chromatography (separating on hydrophobicity) and strong cation exchange chromatography (separating on isoelectric point) [2] CITE 4.2 computational methods.

HPLC methods work by loading the peptide mix in a column containing a stationary and a solid phase. These phases create an environment where peptides interact differently based on their physico-chemical properties, set by the nature of the phases. The output of the column, called elute, will consist of subsets or fractions of peptides leaving the column at different retention times (RT) i.e the amount of time passed before the peptide is observed in the mass spectrometer. Therefore, the input to the machine will consist of a simplified mix of peptides at a time. Notably, the same peptide could elute in several contiguous fractions.

First MS analysis

The tandem MS MS/MS) analysis starts at the first MS scan (MS1), when the peptide mix accesses the analyzer. It is ionized in the ion source and enters the first mass spectrometer. While different ways of handling the peptides are available, we will focus on the product ion scan method. In this protocol, the first analyzer is used to select ionized peptides within a narrow m/z window.

Fragmentation

As the protein sample complexity increases, the MS1 information on the peptide fractions will not be enough to correctly map each to the original protein, thus rendering a correct interpretation of the resulting spectra impossible. Therefore, more information needs to be extracted from them.

Learning the peptide sequences would provide an additional feature that would make the peptide to protein mapping possible. This can be achieved by peptide fragmentation, which triggers the breakage of bonds along the peptidic chain, turning the peptides into smaller fragments. A given bond will be more likely to break the less stable it is. This makes (I) the C_α-CO, (II) the peptidic CO-NH, and (III) the C_α-NH bonds the more likely to break (see figure 1.2). A nomenclature REF was introduced to name these fragments:

Provided enough mass resolution, the mass difference between contiguous fragments of the same series i.e a1 and a2, can be related to the mass of the extra residue in the longer fragment. If this comparison process is repeated for enough fragment pairs of the same peptide, a sequence can be read, finally providing sufficient information to perform an accurate peptide to protein mapping.

Fragmentation in proteomics is performed via (I) collision-induced (CID) or (II) electron-induced (EID) dissociation. CID is an ergodic fragmentation technique where peptides enter a collision cell containing an inert gas. Given enough kinetic energy, hits between ionized peptides and the gas will trigger the fragmentation of the peptide into smaller units. [2] PAG 123 COMPUTATIONAL METHODS. Since the kinetic energy is randomly distributed across the peptide, the weakest bonds will break first. This results in the production of mostly b and y fragments, as well as the loss of

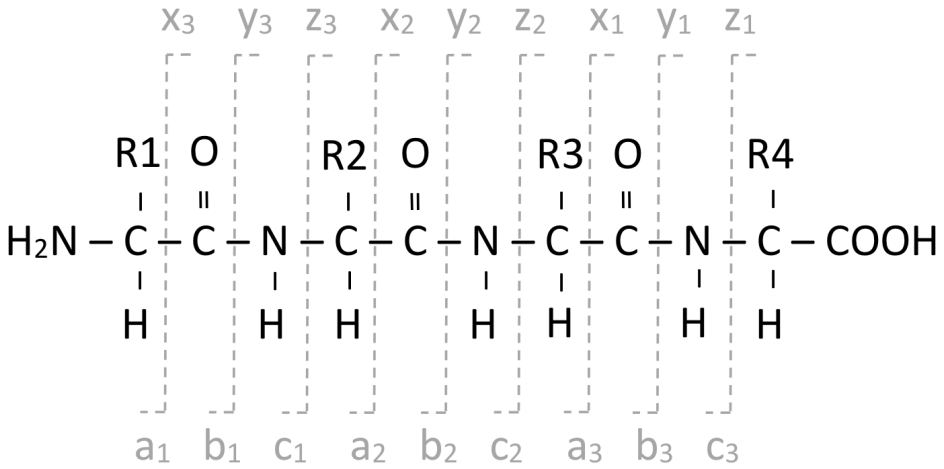


Figure 1.2 The common fragments and their relation to the peptide sequence can be organized into 2 groups of 3 series each. The abc fragments keep the N-terminal residue, while xyz keep the C-terminal one. The a/x, b/y and c/z series are produced by the breakage of bonds I, II and III respectively. Specific fragmentation techniques make fragments belonging to one series more likely than others. Other fragments are possible but much less likely. Taken from CITE COMPOMICS TUTORIAL

any chemical modifications [2] CITE pag 134 computational methods.

EID is produced by the hits against the molecule, which tend to occur on the areas of the peptide more positively charged. Thus, the fragmentation process is not ergodic and returns TYPE ions. More importantly, chemical modifications are not lost in the process.

Second MS analysis

The produced fragments enter the second analyzer MS2, where a m/z spectrum of the fragments is recorded. Thus, unlike in PMF, where the spectrum recorded reflects the m/z ratios acquired by the protein peptides cleaved by the enzyme, tandem MS spectra on product ion scan mode

record the m/z ratio of the fragments produced by an ionized peptide with a given m/z ratio. The m/z ratio of this precursor ion is changed during the run, thus, multiple spectra are obtained where PMF would create only one. The MS2 spectra will encode the sequence of each peptide as mentioned in 1.3.2, and with the right tools, it can be deconvoluted and interpreted as explained in 1.4.

1.4 Spectra processing: search engines

MS search engines are capable of performing the peptide to spectrum matching (PSM). They do it by building statistical models that find the peptides within a protein database that best explain the observed spectra. *De novo* search engines exist that don't require a reference proteome, but better results are obtained when using one..DEVELOP THAT.

Matching is possible by predicting the cleavage pattern of each protein sequence in the database *in silico* based on the cleavage pattern of the enzyme used, coupled with the simulation of the expected spectrum based on the predicted peptides.

Given the stochastic nature of the protein cleavage and spectra recording processes, the resulting spectra exhibit variability manifested in missing peaks or spurious ones. Furthermore, random (wrong) matches can be returned by the PSM process when running against a sufficiently big database. This translates to the generation of multiple matches, of which one, if any, will be correct. Therefore, the lists of matches need to be somehow ranked. The issue is addressed by search engines through the deployment of statistical models that provide scoring systems measuring the goodness of the match. Assuming the correct protein is present in the database, a good scoring system should give the best score to the right

peptide. Under this circumstances, if repeated for several peptides, enough evidence for the presence of individual proteins can be collected.

Multiple search engines exist that implement different matching and scoring algorithms. The most modern ones include MS-GF+, MS-Amanda, Comet, X!Tandem or Andromeda CITE THE ENGINES. Notably, the results of each individual search engine can be combined to gather their strengths, at the expense of an increased computational cost and time REF THIS!!

1.5 Validation and quality control

The scoring system implemented by search engines provides the best matches, but they are bound to contain false identifications. Nevertheless, these scores can be used to apply a filter that aims at minimizing the amount of errors.

A common filter is the false discovery rate (FDR), usually set to 1%, indicating tha the application of this filter one out of a hundred filtered matches are expected to be false positives.

The most commonly used method to compute the FDR of a list of matches is the target-decoy search. Using this method, the search engine replicates the matching process, using the same spectra, but instead against a decoy database. This decoy database is generated by reversing or more generally applying a randomization technique upon the sequences present in the original database (target). This guarantess that while the sequences in the decoy database won't overlap with those in the target, their basic properties of the decoy (size, composition, etc) will remain identical to the target.

All matches to the decoy are by definition wrong. Since database proper-

ties are kept the same, the amount of matches to the decoy exhibiting less than a given score s can be regarded as an estimate of the number of false identifications ($\hat{n}_{fp}$) in the equivalent list of target results. This is because the existence of shared properties entails that random matches are equally likely to happen in both databases CITE COMPOMICS TUTORIAL 1.5. Together with the number of PSMs passing a threshold score ($n_{tp} + n_{tp}$), the FDR can be computed using the formula below.

$$FDR = \frac{\hat{n}_{fp}}{n_{fp} + n_{tp}} \tag{1.1}$$

Equation 1.1 can be used to select the number of positives (the denominator of the fraction) that make the FDR equal to a predefined value, frequently 0.01 or 1 %.

1.6 Peptide and protein inference

Two steps in protein identification can be distinguished:

1. **Peptide inference**: infer the peptides present in the sample.
2. **Protein inferece *proper***: based on the inferred peptides, infer what proteins generated them. This is not trivial as peptides are degenerate and could map to more than one protein.

Peptide inference is performed during the PSM process. The ensemble of proteins most likely to have generated the list of peptides stemming from the filtered PSMs can be inferred using different algorithms. The degenerate nature of peptides is dealt with the Occam's razor principle, which states that the most likely solution is the simplest one. Thus, protein inference algorithms aim at explaining the maximum amount of peptides using the least amount of proteins.

1.7 Protein quantification

The combination of all the aforementioned computational analyses yields a list of protein identities that reports the protein composition i.e qualitative information of the original sample. However, in most proteomics applications, quantitative data can be of great interest, as many biological phenomena are manifested mainly through changes in the protein abundances, rather than protein presence alone. For instance, cancer cells in response to a drug could modulate the abundance of several proteins without removing them from the cytosol or introducing new ones.

Protein quantification pipelines can be classified based on whether isobaric labelling was used (label-based) or not (label-free). These are explained in subsection 1.7.1. If the label-free approach is employed, more distinctions can be made based on:

- The proxy used for quantification: spectral counting (SC) or extracted ion currents (XIC). These are explained in 1.7.2
- The way the data are brought to the protein level from the peptide level: summarization-based vs. peptide-based.

1.7.1 Label-based and label-free approaches

Two paradigms exist in protein quantification: label-based and label-free. In label-based quantification, originally identical peptides from a number of different samples are made distinguishable by their masses via the incorporation of a label. All label-based methods simultaneously analyze several samples in each experiment, removing the difficulties associated with between-run variability [2] CITE PAGE 237 COMPUTATIONAL METHODS The finite number of "plexes" available for a given label sets the limit

to how many samples can be differentially quantified [4]. Different techniques, like Stable Isotope Labeling by Amino acids in Cell culture (SILAC) CITE or Isotope-Coded Affinity Tags (ICAT) CITE, differ in the nature of the label and the way it is introduced. Remarkably, peptide labeling costs can be quite high. This, together with the limited amount of samples that can be compared makes a case for label-free quantification.

In the label-free quantification approach, peptides from different samples are not labelled differently and are thus distinguished by their presence in different, independent MS runs. In order to account for inter-run variability in peptide identifications and RT, a match-between-runs (MBR) processing can be carried out for pair-wise transfer of peptide identities from runs where identification was successful to those were it wasn't due to the randomness in the identification process. The transfer can be made based on shared precursor mass and corrected retention times CITE MOFF. Furthermore, in order to collect consistent XIC, a feature extraction step can be executed to extract the apex intensity of the identified peak clusters 1.3.

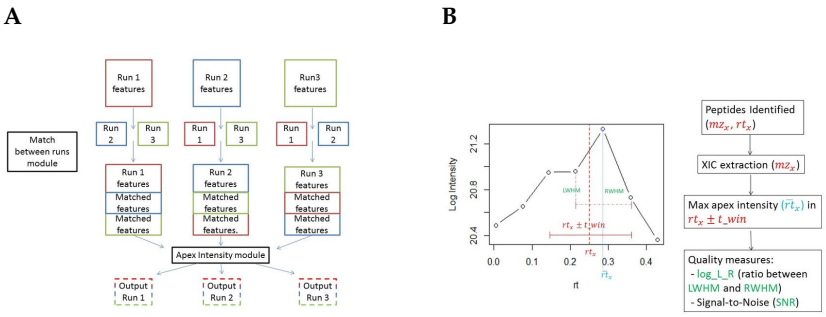


Figure 1.3 Illustration of the MBR and apex intensity extraction steps. **A** the information gathered from matches in replicate runs is collected for a reanalysis of the spectra. **B** A retention time window can be used to screen the peak clusters and highest MS1 intensity of a peak cluster (apex intensity). Taken from CITE MOFF

1.7.2 SC and XIC based quantification

Quantification can be spectral counting or XIC based. Spectrum counting based quantification is the simplest quantification method in proteomics. It relies on the rationale that highly abundant peptides will have a higher intensity and are thus more likely to trigger the acquisition of MS/MS spectra. As a result, peptides from abundant proteins are more likely to be identified and in more spectra. Two approaches were hence followed: count the number of peptides identified for a given protein – like in the emPAI index method – or count the number of spectra ascribed to a protein as in the NSAF method CITE COMPOMICS 4.1. These methods have the advantage that they are very simple to implement and don't require any further data processing.

XIC based methods rely on intensity measurements at any level of the MS workflow as proxies for protein abundance CITE. A wide range of algorithms are available to process these data and output estimates of protein abundance. All of them require an intensive preprocessing step, usually including (I) taking the log_2 intensity to make the data distributions symmetrical and thus make it fit for diverse parametric tests, and (II) quantile normalization to address between-runs variability in the intensity measurements. They can be classified in the bases of which MS level is used as proxy for the protein abundances and on whether or not a summarization step is performed to aggregate peptide-level data into protein-level data, or not.

Although the abundance of proteins and the probability of their peptides being selected for MS/MS sequencing are correlated to some extent, XIC-based methods should clearly be superior to spectral counting given sufficient resolution and optimal algorithms. These advantages are most prominent for low-intensity protein/peptide species, for which a continuous in-

tensity readout is more information-rich than discrete counts of spectra. [4] For this reason, only the XIC approach will be regarded in the rest of the manuscript.

1.7.3 XIC-peptide-based models for label-free quantification

The data collected in the mass spectrometer refers to peptides originating from a latent protein composition, given by the original sample. However, the data interpretation requires the transfer of these peptide-level data into the protein level. This can be done by either (I) performing an aggregation of the peptide-level data, where a summary value of the peptide-level data is taken as representative for the protein-level data, or (II) performing the protein quantification directly at the peptide-level by means of linear regression models.

As stated in [5]. *Peptides originating from the same protein can indeed be considered technical replicates and theoretically should lead to similar abundance estimates. However, the summarization of the peptide intensities into protein expression values is cumbersome, and most summarization-based methods do not correct for differences in peptide characteristics or for the between-sample differences in the number of peptides that are identified per protein. This might introduce bias and differences in uncertainty between the aggregated protein expression values, which are typically ignored in downstream data analysis steps.*

It is for this reason that peptide-based models offer the statistical framework required to learn as much from the data as possible. This translates into improved results when compared to the other aforementioned methods [5].

Chapter 2

A label-free quantification proteomics pipeline

Summary

A pipeline making use of the set of tools published by the Compomics and StatOmics groups SearchGUI, PeptideShaker, moFF and MSqRob CITE THEM, was developed to support complete label-free protein quantification analyses using the most recent advances in the field with open-source software. The pipeline can be run on Linux computer clusters to perform (I) peptide to spectrum matching against a reference database, (II) quality control and filtering, (III) MBR and feature extraction, (IV) protein inference and (V) relative quantification. Its output can be passed to follow-up analyses in R or Python to get a biological interpretation of the results. A benchmark of its performance was accomplished using the proteome benchmark dataset published in [4]. The results exhibited an accuracy similar to those achieved by the MaxLFQ [4] software, excepting a bias produced by the sample fractionation of this dataset.

2.1 Introduction

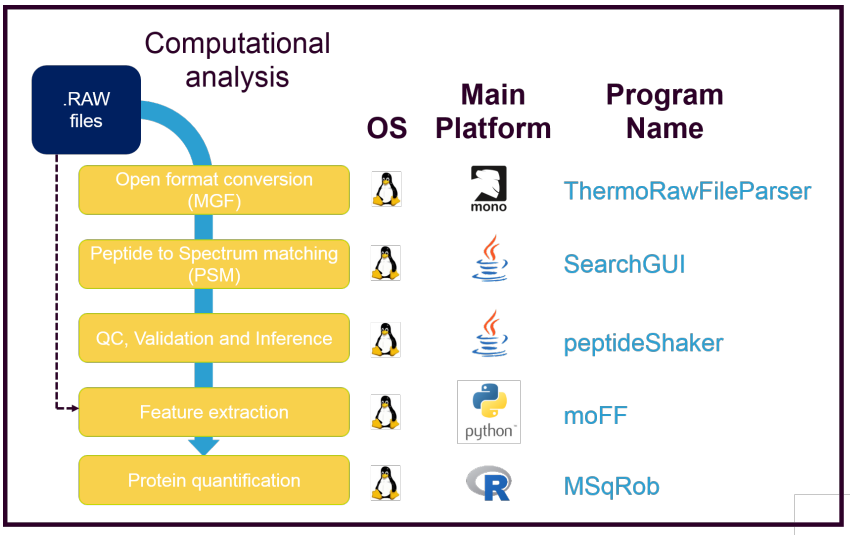


Figure 2.1 Pipeline

Several proteomics pipelines are available on the internet under different licensing conditions. Many are released as closed-source software, where information on how the program works is kept from the user. This is a serious drawback as it hinders the study of the implemented models and its customisation. Open-source, free alternatives, like the Trans Proteomic Pipeline (TPP) or openMS CITE ALL are nevertheless available for the community. MaxQuant, a free but closed-source proteomics analysis suite CITE MAXQUANT, has been extremely successfully adopted by the scientific community due to its ease of use and a comprehensive pipeline.

However, still only few of these tools have a fully Linux-supported, well-documented, command-line version available, which would make customised, automatic streamline analyses much easier to perform. As a GUI would not be required to run the pipeline, it would also be scalable to big datasets. Finally, having free licenses would imply that anyone, provided the technical knowledge, can perform the computational analysis independently, thus

speeding the data to knowledge turnover in NZ or any organization making use of it.

The development of a pipeline achieving these goals will be described in this chapter.

2.2 Materials and Methods

2.2.1 Data generation and loading

The proteome benchmark dataset from [4] was reanalysed starting at the output RAW files available at the PRIDE repository [1]. Briefly, the *Homo sapiens* and *E. coli (strain K12)* proteomes were mixed in 1:1 (condition L) and 1:3 (condition H) proportions, with 3 replicates for each combination. Moreover, each of the 3 replicates of the 2 conditions was analysed over 24 fractions. This experimental setting thus generated a total of $2 \times 3 \times 24 = 144$ RAW files. One file was missing in the repository. The ThermoRawFileParser CITE THERMORAWFILEPARSER program was used to convert RAW files to the open format MGF (Mascot Generic Format).

2.2.2 Decoy database preparation and search

The spectra saved in the MGF files obtained in the previous step were passed to the MS-GF+ search engine CITE by means of the SearchGUI `SearchCLI` tool CITE utility. The search parameters were set using the `IdentificationParametersCLI`. In order to account for potential post-translational modifications, the search was conducted allowing for the following variable modifications: oxidation of M and deamidation of N and Q. Moreover, C carbamidomethylation was set as fixed modification. The

[1] https://www.ebi.ac.uk/pride/archive/projects/PXD000279

enzyme was set to semispecific Trypsin, allowing for a non-tryptic cleavage on any side of the peptide. Up to two missed cleavages were allowed. The precursor tolerance was 10 ppm and the fragment tolerance 0.5 Da.

The target database was created by combining the Uniprot proteomes for *E. coli (strain K12)* (UP000000625) and *Homo sapiens* (UP000005640), downloaded in June 2018. The decoy database was created using the `FastaCLI` utility in SearchGUI by reversing all sequences in the target.

2.2.3 Quality control and validation

The SearchGUI results were filtered using the default built-in checks available in the PeptideShaker utility `PeptideShakerCLI` [2]. By default, the FDR was set to 1%. PEP and confidence statistics were computed using the PeptideShaker built-in algorithms. Output was extracted via the Default PSM report txt file, available in the `ReportCLI` utility.

2.2.4 Data refinement

The moFF command line utility CITE was applied to perform (I) match-between-runs and (II) extract MS1 apex intensity of each peak cluster. This required passing the original RAW files, together with the Default PSM report from PeptideShaker. Output was exported to a peptide summary file, containing one row per peptide and for every peptide, the detected apex intensity in each sample.

[2] https://github.com/compomics/peptide-shaker/issues/300

2.2.5 Quantification

Relative quantification was performed using the MSqRob utility by passing the peptide summary file from moFF. Prior to quantification, the data was preprocessed using the `preprocess_MSnSet()` function. In a nutshell, (I) MS1 apex intensities were log_2 transformed, (II) peptides belonging to protein groups that contained one or more proteins that were also present in a smaller protein group were discarded CITE MAIN MSQROB, and (III) protein groups with only 1 peptide were dropped. Intensity normalisation was not performed as a systematic bias in intensities is indeed expected for all *E. coli* peptides from samples where the *E. coli*proteome was 3 times more abundant.

Once preprocessing is done, a ridge regression model with Huber weights and empirical Bayes estimation of the protein variance was fit to every protein individually. The significance of the treatment effect differences was assessed through a Student's T test, implemented in the `test.contrast()` function.

2.2.6 Code implementation

2.3 Results

2.3.1 Peptide to spectrum matching step identified thousands of spectra

The preprocessing of the RAW files produced by the mass spectrometer into the MGF open format enabled searchGUI to dispose of the registered spectra. PeptideShaker quality control and filtering capabilities (see figure 2.2) carried out the required search results validation. As expected, matches

to the target and decoy exhibited similar score distributions at low score values, while a divergence is observed at higher score values. Likewise, the m/z error was found to be closer to 0 on validated PSMs than on those which did not pass the 1 % FDR filter. The application of this filter implied that the FNR (false negative rate) was set to 5 %, i.e 5 out of every 100 discarded matches were estimated to be true positives.

The PSM confidence is defined as $1 - PEP$, where PEP stands for the Posterior Error Probability. Also known as local FDR, the PEP is an estimate of the probability of a given PSM of being an incorrect assignment. Thus, the confidence is the probability of the PSM being a correct assigment CITE NEVINZKY. The 1% FDR cutoff selected PSMs with a score higher than 78, which translated to a confidence of at least 65%.

The combination of both programs enabled the identification and validation (matching) of thousands of spectra with high confidence in all samples . However, when compared to the total amount of spectra available, the percentage of matched spectra was on average 37.4%, with a marked decrease starting at fraction 14 (see figure 2.3).

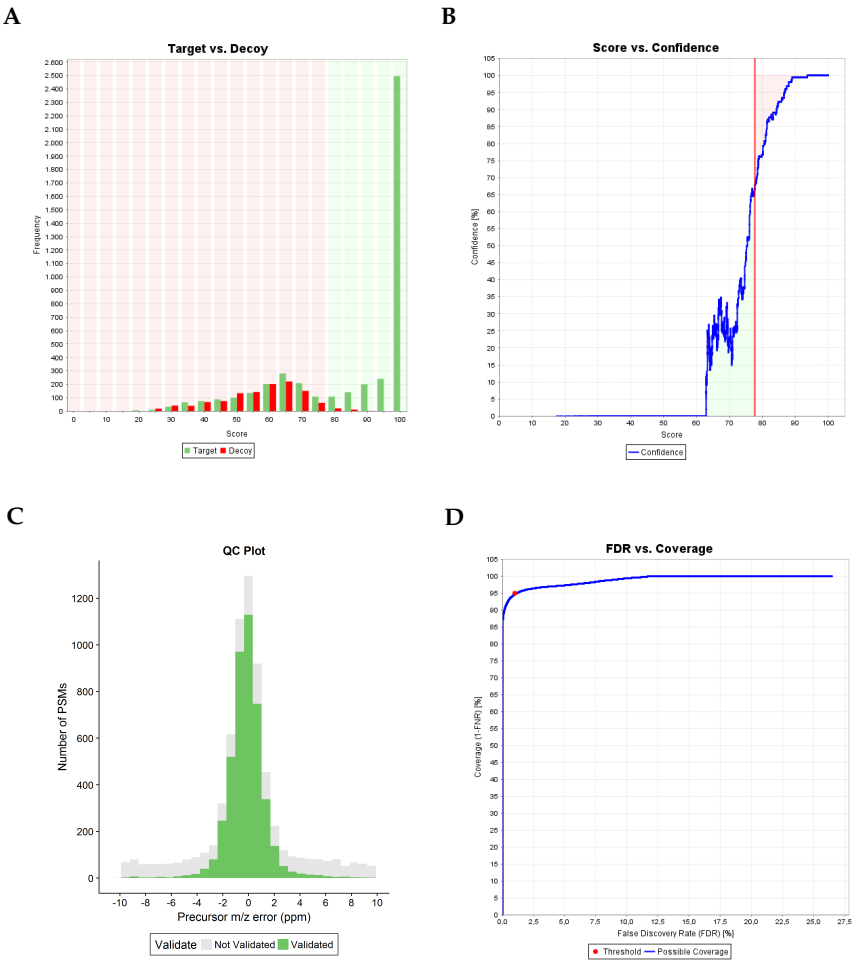


Figure 2.2 Quality control and validation of the 13th fraction of the third replicate in condition L. **A** Score distribution for matches to the decoy and the target databases. **B** Evolution of the PSM score with confidence. The implemented cutoff at FDR of 1 % is displayed with a red vertical line. **C** Distribution of the difference between the predicted and measured m/z values, segregated by validation status. **D** ROC curve built upon the number of false positives and negatives estimated from the decoy search. The cutoff is displayed as a red dot.

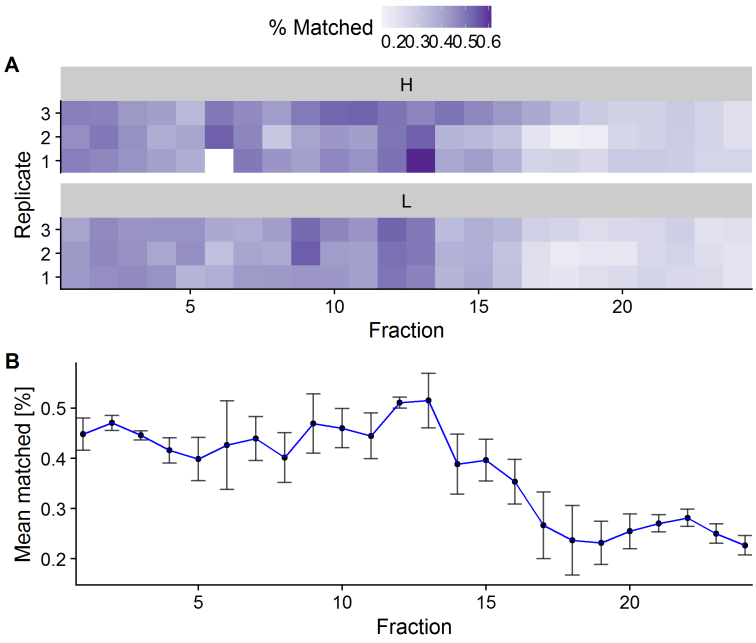


Figure 2.3 Percentage of matched spectra in all samples. The total number of spectra per sample ranged between 5894 and 20249. **A** Percentages are encoded with a blue palette, the darker, the higher, and viceversa. **B** The mean for each analysed fraction across conditions and replicates is displayed together with error bars to represent the standard deviation. The sixth fraction of the first replicate in condition H was missing in the PRIDE data repository.

2.3.2 MBR and Apex intensity extraction capacitate for accurate quantification

The match between runs step allows for increased identifications by transferring successful matches between replicate runs. The results of this process for the 13th fraction of the L condition is shown in figures 2.4 and 2.5)

Once as many identifications as possible were gathered, a refinement of the measured MS1 intensity can be implemented to select the apex of every peak cluster, which yields robust intensity measurements for each sample (see figure 2.6). The extracted apex intensity can be used as a proxy for

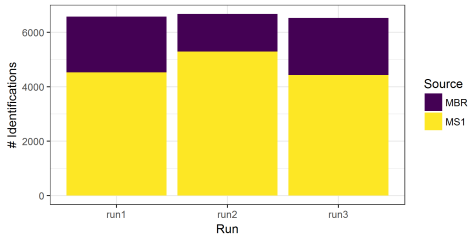


Figure 2.4 Count of identifications on each run segreagated by source. More than 4k spectra were identified and validated by SearchGUI+peptideShaker. The MBR acts as an imputation step where extra identifications are performed by gathering the information collected from identifications in other samples and reprocessing the spectra files. As a consequence, more identifications with a lesser fraction of missing datapoints are achieved. In this particular case, hundreds of identifications were accomplished

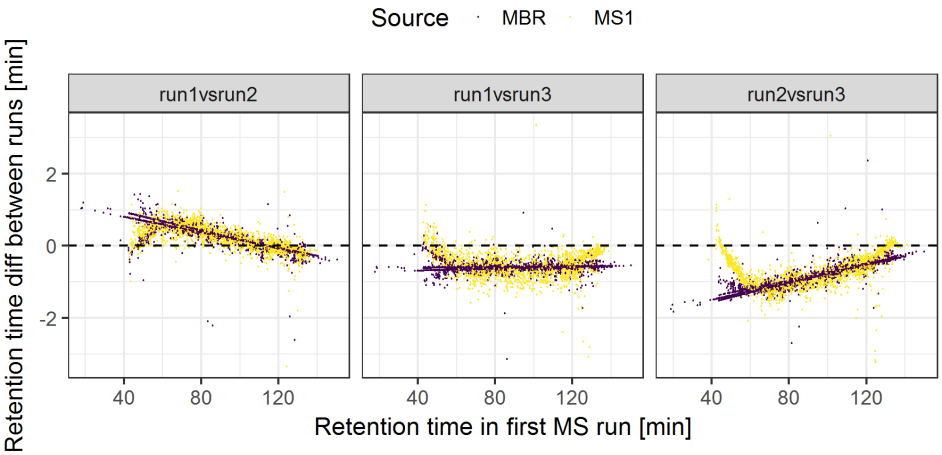
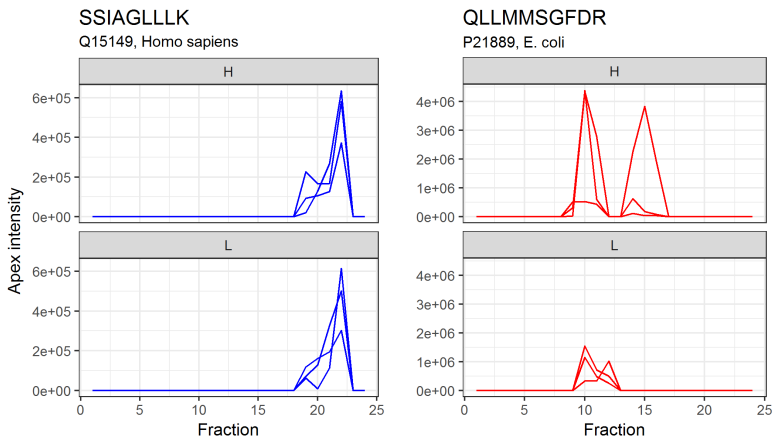


Figure 2.5 Match Between Runs with moFF. Every dot represents a peptide shared across 2 runs. The coordinate system illustrates its retention time on the first run on the x axis and the difference with the second run on the y axis. The color depicts whether the identification was carried out during the PSM process, or thanks to a cross-identification achieved by the MBR module.

peptide abundance in the following quantification step.

A



B

Figure 2.6 A The apex intensity profile across fractions for 2 different peptides, one from *Homo sapiens*, and one from *E. coli*. The figure illustrates the intrinsic intensity variability between technical replicates, particularly in the case of the *E. coli* QLLMMSGFDR peptide, as it was almost non-existent in one of the runs. **B** The expected pattern of overall similar intensities for the *Homo sapiens* data and 3-fold higher intensities for the *E. coli* data in condition H was observed, confirming the good performance of the protocol.

2.3.3 Quantification

The result of this step was a table where the $log_2(FC)$ estimate, together with test statistics are returned for every protein group.

2.4 Discussion

2.4.1 Improvement of the PSM process

The low attained matching rate manifests existing room for improvement in the currently available tools CITE THAT. Remarkably, it has been shown that the combined usage of multiple search engines can increase identifications, since the different statistical frameworks implemented in each of them compensate each other's caveats CITE THAT. Furthermore, *de novo* search engines are available and supported by SearchGUI. Their usage could contribute to further improvements in the identification rate.

The most important reason why many spectra remain unidentified is the presence of post-translational modifications (PTMs), which exponentially increase the search space, forcing most workflows to discard many of the peptides featuring a PTM. New approaches to the problem are emerging, mainly machine learning methods for the handling of unexpected modifications CITE ralf gabriels. Moreover, the prediction of MS2 peak intensities patterns from peptide sequences promises to increase the amount of evidence available during the PSM process, thus boosting correct identifications CITE MS2PIP.

Finally, the extremely low matching rates in the latter fractions translates to decreased contributions to the number of identifications. This indicated that decreasing the number of fractions would not have had a major impact in the experiment's depth.

2.5 Conclusion

Chapter 3

Pipeline benchmarking on NZ data

Summary

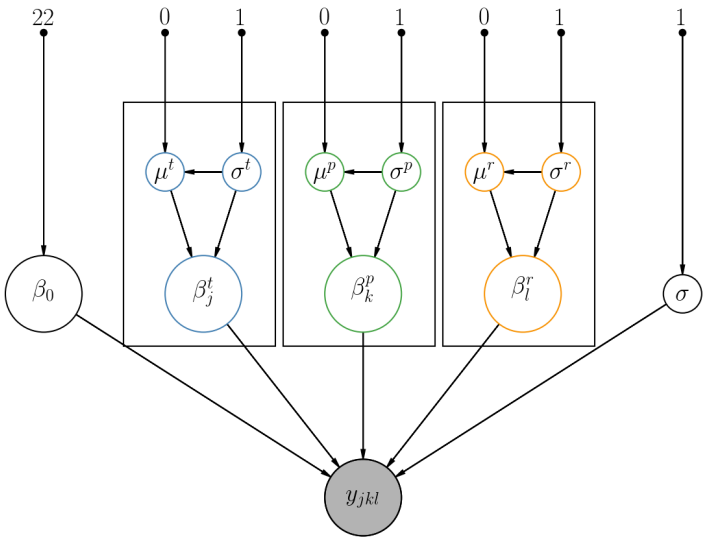
A benchmark dataset generated by NZ technicians was run through the pipeline presented in chapter 2 to showcase its performance. The experiment consisted of the application of two different treatments to THP-1 cell cultures. One group was subjected to an experimental procedure triggering the immunological response, while the other group was subjected to a negative control. The analysis of the resulting dataset through the aforementioned pipeline should thus reflect a change in the protein profile corresponding to an activation of the immune system in the first group when compared to the second. The results confirmed that the computational analysis successfully captured this response. It was concluded that the software can be used in future experiments where the expected biological phenomena is not known.

Chapter 4

Bayesian modelling of fold change estimates

Summary

The current label-free quantification methods, reviewed in section 1.7 all rely on frequentist statistics, which return point estimates of model parameters such as the estimate of the fold change across conditions. However, a Bayesian based approach to this problem is lacking in the literature. As a response to this shortcoming, a statistical model implemented in the probabilistic programming framework Pymc3 was developed and tested on the same benchmark dataset from chapter 2). The execution of the three steps required when doing Bayesian modelling, mainly (I) model implementation, (II) computation of posterior probabilities and (III) model checking, will be described for this particular problem in the present chapter, together with a discussion on its usability and its strengths.


22
0
1
0
1
0
1
1
μ^t
σ^t
μ^p
σ^p
μ^r
σ^r
β_0
β_j^t
β_k^p
β_l^r
σ
y_{jkl}

Conclusion

Appendix

```python
import pymc3 as pm
import numpy as np
import matplotlib.pyplot as plt
from theano import shared
import shutil
import os

def protein_model(observed_sh, feats_sh, x_treat_sh, x_pep_sh, x_run_sh, x_estimate_sh,
    n_peptides, model_name, n_draws=1000, n_chains=3,
    hierarchical_center=False, remove_backend=True, sequence=False):

    # Check working environment
    if not os.path.isdir("traces") or not os.path.isdir("plots/traceplots"):
        msg = "Please create a traces dir and a plots/traceplots dir before running this code"
        raise Exception(msg)

    if remove_backend and os.path.isdir(model_name):
        shutil.rmtree(model_name)

    # The number of proteins in this model is always one
    # i.e this model is fitted protein-wise
    n_prots = 1
    # The number of features is set to 9 for now
    # All peptides have 9 features, stored in feats_shared
    n_features = 9


    with pm.Model() as model:

        # Build a hyerarchical linear model of
        # log2(MS1 intensities) by accounting for:

            # Peptide effect
            # Run (batch) effect
            # Treatment effect
            # Remaining random effects

        # The difference in treatment effects is an estimate of the log2FC


        # Set a prior on the intercept
        intercept = pm.Normal("intercept", 22, 1)

        # Set a prior on the remaining random effects
        sigma = pm.HalfNormal('sigma', 1)

        ## Set priors on the peptide effect
        ################################################
        sigma_pep = pm.HalfNormal('sigma_pep', 1)
```

```
50
51          # Not using the sequence
52          if not sequence:
53              mu_pep = pm.Normal('mu_pep', mu=0, sd=sigma_pep, shape=(n_peptides, 1))
54
55          # Using the peptide sequence
56          else:
57              # sequence based modelling
58              mu_theta = pm.Normal('theta_generic', 0, sigma_pep, shape = 1)
59              theta = pm.Normal('theta', mu_theta, sigma_pep, shape = (n_features, 1))     # 9x1
60              theta_inter = pm.Normal('theta_inter', mu_theta, sigma_pep, shape = 1)
61              mu_pep = pm.Deterministic("mu_pep", theta_inter + feats_sh.dot(theta)) # n_peptidesx1
62
63
64          ## Set priors on the treatment and run effects
65          ################################################
66          sigma_treat = pm.HalfNormal('sigma_treat', 1)
67          mu_treat = pm.Normal('mu_treat', 0, sigma_treat)
68          sigma_run = pm.HalfNormal('sigma_run', 1)
69          mu_run = pm.Normal('mu_run', 0, sigma_run)
70
71          # Standard implementation of the hyerarchies
72          if hierarchical_center:
73              pep = pm.Normal("pep", mu_pep, sigma_pep) # n_peptidesx1
74              treat = pm.Normal('treat', mu_treat, sigma_treat, shape = (n_prots*2, 1))
75              run = pm.Normal('run', mu_run, sigma_run, shape = (n_prots*6, 1))
76
77          # Reparametrization to escape funnel of hell as noted in
78          # http://twiecki.github.io/blog/2017/02/08/bayesian-hierchical-non-centered/
79          else:
80              pep_offset = pm.Normal("pep_offset", mu=0, sd=1, shape = (n_peptides, 1))
81              pep = pm.Deterministic("pep", mu_pep + pep_offset * sigma_pep)
82              treat_offset = pm.Normal("treat_offset", mu=0, sd=1, shape=(n_prots*2, 1))
83              treat = pm.Deterministic("treat", mu_treat + treat_offset*sigma_treat)
84              run_offset = pm.Normal("run_offset", mu=0, sd=1, shape=(n_prots*6, 1))
85              run = pm.Deterministic("run", mu_run + run_offset*sigma_run)
86
87
88          # Model the effect for all peptides
89          # The sh variables consist of -1,0,1 matrices telling pymc3
90          # which parameters shall be used with each peptide
91          # In practice, the "clone" each parameter to fit the shape of observed_sh
92          # observed_sh is a n_peptides*6x1 tensor
93          # The first 6 numbers store the MS1 intensities of the first peptide in the 6 runs
94          # The next 6 those of the second peptide, and so on
95
96          estimate = pm.Deterministic('estimate', pm.math.sum(x_estimate_sh.dot(treat), axis=1))
97          treatment_effect = pm.Deterministic("treatment_effect", pm.math.sum(x_treat_sh.dot(treat), axis=
98          peptide_effect = pm.Deterministic("peptide_effect", pm.math.sum(x_pep_sh.dot(pep), axis=1))
99          run_effect = pm.Deterministic("run_effect", pm.math.sum(x_run_sh.dot(run), axis=1))
```

```
100
101         # BIND MODEL TO DATA
102         mu = pm.Deterministic("mu",
103             intercept + treatment_effect + peptide_effect + run_effect) #n_peptides*6x1
104         if hierarchical_center:
105             obs = pm.Normal("obs", mu, sigma, observed=observed_sh)
106         else:
107             obs_offset = pm.Normal("obs_offset", mu=0, sd=1, shape=(n_peptides*6,1))
108             obs = pm.Normal("obs", mu+obs_offset*sigma, sigma, observed=observed_sh)
109
110
111     print("Success: Model {} compiled".format(model_name))
112
113     with model:
114         # Parameters of the simulation:
115         # Number of iterations and independent chains.
116         n_sim = n_draws*n_chains
117
118         # Save traces to the Text backend i.e a folder called
119         # model_name containing csv files for each chain
120         trace_name = 'traces/{}'.format(model_name)
121         db = pm.backends.Text(trace_name)
122         trace = pm.sample(draws=n_draws, njobs=n_chains, trace=db,
123                           tune=2000, nuts_kwargs=dict(target_accept=.95))
124
125     # Save a traceplot
126     pm.traceplot(trace, varnames=["estimate"])
127     traceplot = "plots/traceplots/{}.png".format(model_name)
128     plt.savefig(traceplot)
129     plt.close()
130
131     return model
```

Bibliography

[1] David L. Nelson. *Lehninger : principles of biochemistry*. W. H. Freeman and Co,, New York :, 5 edition, 2008.

[2] L. Martens I. Eidhammer, K. Flikka and S.-O. Mikalsen. *Computational Methods for Mass Spectrometry Proteomics*. 2008.

[3] Alan G. Marshall, Greg T. Blakney, Tong Chen, et al. Mass Resolution and Mass Accuracy: How Much Is Enough? *Mass Spectrometry*, 2(Special_Issue):S0009–S0009, 2013.

[4] Jürgen Cox, Marco Y. Hein, Christian A. Luber, et al. Accurate Proteome-wide Label-free Quantification by Delayed Normalization and Maximal Peptide Ratio Extraction, Termed MaxLFQ. *Molecular & Cellular Proteomics*, 13(9):2513–2526, 2014.

[5] Ludger J.E. Goeminne, Andrea Argentini, Lennart Martens, and Lieven Clement. Summarization vs peptide-based models in label-free quantitative proteomics: Performance, pitfalls, and data analysis guidelines. *Journal of Proteome Research*, 14(6):2457–2465, 2015.