Design and implementation of analysis pipeline for single cell type proteomics data

By

Lukas Gamp

in partial fullfilment of the requirement
for the degree of MSc
in Bioinformatics

mm yy

Abstract

(the spacing is set to 1.5)

no more than 250 words for the abstract

- a description of the research question/knowledge gap – what we know and what we don't know
- how your research has attempted to fill this gap
- a brief description of the methods
- brief results
- key conclusions that put the research into a larger context

Contents

Acknowledgements

Thank you for following this tutorial!

I hope you'll find it useful to write a very professional dissertation.

1 Introduction

1.1 From the Central Dogma of Molecular Biology to Omics and Systems Biology

Before understanding the underlying principle biological systems undergo and the initial exploration of molecular biology, the processes of live were roughly inferred and not well understood. The central dogma of molecular biology serves as the cornerstone of biological processes, providing a framework for understanding how genetic information is converted into functional proteins. This framework consists of two major processes, which involve transcription, converting DNA into RNA, and translation, converting RNA into functional proteins. In between those processes regulation takes base on a molecular level, by altering structure with chemical adaptations. The regulation is undertaken to provide biological organisms with an highly efficient output of their genetic products called proteins. Crick's formulation of the central dogma has guided research in molecular biology for over 60 years (Cobb 2017). Over the past six decades, researchers have explored various aspects of this principle, collectively referred to as omics, to comprehensively analyze biomolecules in diverse contexts. Omics research has emerged as an approach for comprehensively studying biomolecules and their interactions within biological systems. Omics encompasses various fields, including genomics, transcriptomics, proteomics, metabolomics, and others, each focusing on a specific class of biomolecules. By analyzing multiple omics layers, researchers can gain a holistic view of biological processes, uncover regulatory mechanisms, and identify key players in complex biological networks with computational methods. Systems biology takes a quantitative approach to investigate the interactions, dynamics, and emergent properties of biomolecular networks. Imagining biological systems as complex, multi-layer networks, the ensemble of proteins, known as the proteome, plays a critical role in this interactive structure with many functions such as structural integrity, catalyzing chemical reaction and regulation of cellular funcitons.

1.2 Proteomics

The central dogma of molecular biology offers insights not only at the level of individual terms but also provides a hierarchical understanding of biology. Hierarchical organization is the underlying principle in all topics of biology. Proteins exhibit hierarchical structures that contribute to their functionality. At the primary structure level, proteins are composed of smaller building blocks called amino acids. Combining them results in the protein secondary structure such as alpha helix and beta

sheet. These secondary structures further assemble to create higher-order tertiary structures, which represent specific domains within the protein. Finally, the quaternary structure describes the functional state of the protein as a whole at a given time. For instance pathogens interact with cells of an organism a cascade of biochemical reactions takes place. These reactions influence the outcome of the cellular proteome in regards of localization, abundance, post-translational modifications and more (Beltran et al. 2017). By studying the proteome, which encompasses all the proteins present in a biological system at a specific moment, we can gain a comprehensive understanding of the hierarchical organization and functional dynamics of biological processes.

1.2.1   Bulk proteomics

Higher organisms are composed of specialized cells organized into tissues, such as skin, muscle, and blood. Each tissue consists of cells with specific functions, resulting in variations in protein expression. Bulk proteomics is a technique used to analyze the protein composition of a tissue sample, which contains all types of cells present in that particular tissue. This approach finds valuable applications for instance in oncology, where it can provide insights into the protein profile of a tumor. By studying oncogenic biomarkers and expression patterns, this knowledge can contribute to the development of screening methods and the customization of treatments based on observed expression patterns. Additionally, bulk proteomics can aid in identifying the progression of tumor development (Kwon et al. 2021). Bulk analysis expression profiles provide scientists with an average protein abundance across all cells in a sample, offering a comprehensive view of protein expression. Additionally, compared to single-cell methods, bulk samples typically contain a greater number of proteins. In fact the expanded library can be used to validate the performance of single-cell type and single-cell proteomics methods and bioinformatics pipeline (Schoof et al. 2021). However, in this type of analysis, the precise properties of individual cell types can only be inferred and remain obscured.

1.2.2   Single-cell type proteomics

Taking tissue samples can lead to an averaging effect across the entire cellular ensemble, making it difficult to discern specific cell types. To overcome this limitation, cell sorting techniques such as fluorescence-activated cell sorting (FACS), magnetic-activated cell sorting (MACS), and buoyancy-activated cell sorting (BACS) are employed (Liou et al. 2015). These techniques rely on the detection of surface proteins on cells and the specific binding of antibodies to these proteins. Antibodies, typically derived from the immune system of other species, bind to antigens in a

lock-and-key manner. Antibodies are widely used in molecular biology and find applications in cell isolation. Depending on the desired application, the antibodies can be labeled with fluorescent markers (FACS), magnetic beads (MACS), or biotin (BACS). These methods have varying technical requirements, with flow cytometers being the most complex and costly. In MACS, magnetic beads are used to isolate cells from a solution by utilizing a magnet. However, mechanical forces involved in MACS can potentially damage the cells. BACS, a method developed in 2015, employs microbubbles to carry the antibodies for cellular isolation. The method showed improvements compared to MACS in regards of viability after sorting. It is important to note that FACS is considered the gold standard for validating the effectiveness of MACS and BACS due to its high purity and yield. FACS was used in the validation of these two methods (Liou et al. 2015, Sutermaster & Darling 2019). Single-cell type proteomics reflects the protein ensemble of a specific cell type at a particular time, providing a focused perspective on the studied field compared to bulk methods (Maes et al. 2020).

1.2.3 Single-cell proteomics

Cell sorting techniques primarily rely on extracellular proteins to identify and separate different cell types. However, the intracellular content of cells can still exhibit variability in experiments. Single-cell proteomics (SCP) addresses this limitation by providing a snapshot of the underlying processes within an individual cell, eliminating this variability and allowing for the elucidation of the principles of cell type behavior, such as differentiation. Differentiation, a critical process in various biological systems such as the immune system, involves the acquisition of specialized functions by specific cell types. Investigating differentiation in multiple cells can be challenging due to the inherent heterogeneity during the initiation of this process. However, by analyzing one cell at a time and observing it throughout the differentiation process, researchers can capture the complete trajectory of differentiation at a single-cell level. This approach provides valuable insights into the dynamic changes occurring within cells during differentiation. In 2021, Specht et al. developed the SCoPE2 sample preparation and analysis pipeline, performing macrophage-monocyte differentiation experiments (Specht et al. 2021). This pipeline streamlined the sample preparation process, reducing labor time and providing a reference model for subsequent SCP projects. SCP has proven to be a powerful tool for studying a wide range of biological phenomena, including drug responses, infectious diseases, and organism development. Over the past decade, single-cell techniques, including SCP, have become indispensable tools for the global research community (Minakshi et al. 2019).

1.3 Capturing the proteome

An early approach of qualitative analysis of the cellular proteome involved labeling with fluorescent antibodies and imaging. The major disadvantage of this technique was the limitation to only stain a few proteins per cell and the requirement of a-priori knowledge of the protein to analyze. For quantification procedures such as Western blots, immunoassays or cytometry by time of flight (CyTOF) have been used. Other disadvantages are the ability to permeate cells, accessibility and binding of the epitope and the creation of specific antibodies for a given protein (Budnik et al. 2018). Early quantitative methods in molecular biology were assays like the UV-vis-, Bradford- and/or bicinchoninic-acid-assay measuring the entire protein concentration. These methods are still in use to obtain total concentration for sample preparation. Another quantitative technique involved RNA-sequencing. RNAseq refers to the task of examining the transcriptome of bulk tissue sample with deep-sequencing technologies. RNA in a sample includes mRNAs, non-coding RNAs, and small RNAs. One of the main disadvantages is, that the mapping relies on the knowledge of the DNA sequence of the observed organism (reference genome) (Wang et al. 2009). Another disadvantage is, that the amount of protein translated from the determined mRNA can only be estimated (Gygi et al. 1999). However, today these predictions can be done with computational methods such as LASSO and the accuracy of the estimate highly depends on selection and validation of this method (Magnusson et al. 2022). When observing the protein ensemble of a cellular environment with a top-down approach like DNAseq or RNAseq, the characteristics (e.g. post-translational modifications or charge) and local concentration of the proteins remain hidden. The cause of this phenomenon is that only a fraction of the DNA is transcribed into mRNA and not all mRNA is further translated into protein. Furthermore multiple proteins can be derived from a single DNA-sequence with alternative splicing and different proteins coming from non related sequences acting like building blocks can form protein complexes and/or Protein-RNA complexes. Cells are highly efficient and do not waste energy by producing obsolete proteins, moreover the degradation of proteins is an important energy recovery mechanism. A bottom-up strategy can elucidate these properties of the protein ensemble at a given time leading to proteomics. When studying impact of an external signal or stressor to a cell the abundance and properties of a protein or protein-ensemble can enlighten the underlying pathway leading to a biological response. Since some proteins do not even function before post-translational modifications (PTMs) happen, a knowledge about these can be crucial for further studies. Mass spectrometry (MS) fusions the advantages of the above mentioned techniques, because it combines the specificity of qualitative assays which are even able to sense (PTMs) with the quantitative accuracy of protein concentration assays. Furthermore MS does not require any a-priori

knowledge of the protein ensemble of the specimen, making it the method of choice for untargeted approaches.

1.3.1 Mass Spectrometry

Mass spectrometry enables qualitative and quantitative analysis of the entire repertoire of a biological sample. Mass spectrometers measure the mass to charge ratio (m/z) of a particle from a fragmented larger molecule. This is achieved by a physical procedure done with a device which is made of three major components: the ion source, an analyzer, and a detector. The ion sources charges molecules and accelerate them through a magnetic field. The analyzer separates particles according to their mass to charge ratio and the detector senses charged particles and amplifies their signal (Parker et al. 2010). However before acquiring any data, sample preparation and pre-processing is required. After sample preparation the first step involves tryptic digestion. It is a widely used technique in shotgun proteomics and involves the enzymatic cleavage of proteins into smaller peptides using the protease enzyme trypsin. This process is known as proteolysis. The enyme cleaves specifically at the caroboxyl side of arginine and lysis residues. Afterwards a charge remains on the residues making them detectable for the mass spectrometer. It is proven that proteins which are digested to the limit, meaning no further proteolysis can be induced, provide the best results in such bottom up approaches (Brownridge & Beynon 2011, Ünige A. Laskay et al. 2013).

To observe multiple samples in one run, labeling is needed to identify each sample within a batch. Since mass spectrometry is not a quantitative technique by itself, the peak height or area does not reflect the abundance of a peptide. Physicochemical properties of the peptides can change the ionization efficiency and detectability of the target. However, when comparing the same analyte between multiple runs of labeled peptides, differences in the reporter ion intensity reflect the abundance of those. Labels should be chosen to change solely the mass of the sample and to not affect folding or other inherent properties of the peptide. The two techniques for labeling peptides are metabolic labeling and isobaric labling. In metabolic labeling the cells were fed with aminoacids containing heavy isotopes, it is the method of choice in order to label peptides at the earliest possible level. This atoms can be heavy nitrogen in aminoacids or salts in fertilizer for plants. Mass shifts are proportional to the isotopes incorporated during biomass production and are visible after proteolytic cleavage. Stable isotope labeling in cell culture (SILAC) was presented in the early 2000s. This method used heavy aminoacid enriched media to feed cells, in order to quantitatively analyze expression profiles. The limitation of this method is, that it can only be applied to a human cell culture (in vitro) or to model organisms (in vivo). Isobaric labeling solely applies to in vitro

techniques, with TMT and iTRAQ. The data analyzed by the developed pipeline primarily utilized TMT tags. Tandem mass tag (TMT) reagents enable differentiation of multiple samples in a single mass spectrometry (MS) run. Each sample is individually labeled with TMT reagents and then pooled together, this procedure is called multiplexing. TMTs have the same charge and differ only by their isotopic masses, the peaks observed for each sample are called reporter ions (RI). Each RI and sample is interpreted as one channel in downstream analysis. This technique allows for the identification, enrichment, and quantification of low-abundance peptide ions, which is particularly valuable in single-cell techniques. With this technique it is possible to quantify proteins and differ low abundant proteins from background noise. However, a drawback of isobaric labeling is the presence of co-fragmentation signals in the spectrogram, which requires data normalization to remove unwanted contributions (Marx 2019, Budnik et al. 2018). Isotopic distribution in TMTs, reflecting the natural distribution, can be corrected during data acquisition by considering a defined spread in other channels. Single-cell proteomics by mass spectrometry (SCoPE-MS) uses a isobaric labeling with a tandem mass tag (TMT) to enhance signal intensity of a protein species. Loading of a single-cell channel (SCC) with the right amount can be balanced to the protein concentration of a single cell, leading to single-cell proteomics (Ye et al. 2022).

Before ionization peptides need to be further separated according to their chemical properties, size or species by liquid chromatography (LC), because the number of peptide processed simultaneously by the MS device is limited. In the case of mass spectrometry, liquid chromatography especially high performance LC (HPLC) is the technique of choice and replaces the instrumentation-wise simpler technique gel electrophoresis. In HPLC the molecules eluate through a narrow column with high pressures (50-350 bar), compared to LC where the separation is gravity dependent. Similar to gel electrophoresis molecules can be separated upon different properties depending on the type of the chromatography column. In a typical proteomics experiment peptides are separated with reversed phase chromatography (RPLC). The principle behind RPLC is a column with a nonpolar stationary phase which interacts with the nonpolar residues of the peptides. Peptides were fractionated according to their hydrophobicity.

To analyze a biological sample consisting of peptides in solution the liquid needs to be vaporized into gas phase. Two techniques are capable of this procedure. Electrospray ionization (ESI) pushes the analyte through a capillary while applying an electric current to the liquid, vaporizing the sample to a charged aerosol. Peptides are further fragmented according to their chemical properties and can be further handled in the mass spectrometer. The fragmented peptides are now in charged droplets separated based on their surface charge, splitting further into smaller droplets until they reach the

gas phase as ions. Two physical models describe the process from gas phase to ion called "The ion evaporation model" (IEM) and "The charge residue model" (CRM). In the ion evaporation model the droplets shrink by evaporation until ions are expelled (Iribarne 1976). The model had its limitation by explaining same evaporation rate constant among ions with different chemical properties. In the charge residue model the assumption of one molecule per droplet leads to an ionization rate constant, which is independent of the ion itself and relies solely on the generation of the droplet and the efficiency of the solvent (Wilm 2011).

Subsequently after the ionization peptides are accelerated through the magnetic field generated by the mass analyzer. Six general types of mass analyzers are availaible on the market, there are Quadropole, time of flight (TOF), magnetic sector, electrostatic sector, quadropole ion trap and ion cyclotron resonance -mass analyzer. The function of the mass analyzer is to separate ions based on their mass to charge ratio (m/z).

Before the signal of the separated ions can be obtained an optional subsequent step involves trapping the ions an electric and/or magnetic field in order to detect ions based on their (m/z) consecutively. Ion trapping enables the technique of coupled mass spectrometers, where ions were further fragmented by collision induced dissociation and analyzed subsequently. The detector transforms the impact energy of the ion into an electric signal, called spectrum which can be processed computationally.

1.3.2 Data processing

Given the high resolution of MS data, algorithms are employed to convert the raw signal into an interpretable form. Software packages like MaxQuant (Cox & Mann 2008) are commonly used to process the data, providing it for further analysis and statistical testing. Other software solutions include Protein Discoverer by Thermo Fisher.

The data obtained from mass spectrometry has three dimensions: m/z ratio, intensity, and retention time. To separate peaks from each other, algorithms are used to identify local minima in the data. The centroid of each peak is determined by fitting a Gaussian peak shape, which can be interpreted as locating the peaks of each m/z spectrum as a function of retention time. The centroid of a peak corresponds to an isotope.

In order to decipher the isotopic distribution of a biomolecule, MaxQuant employs a vertex-based approach. This process, referred to as de-isotoping, utilizes graph theory. It creates a vertex for each individual peak and connects them with their potential isotopic counterparts. This is achieved by

determining the mass proportion of an average amino acid to its respective isotope using a concept known as averagine (Senko et al. 1995). By applying this procedure, the number of data points is reduced by a factor of ten, and each peak represents a small biomolecule.

The subsequent step in data acquisition involves the detection of labels for quantification purposes. Isotopic pairs of the label (such as N13, N14, N15) present in the tag or aminoacid are identified by convolving the two measured isotope patterns with the theoretical isotope patterns. Through an iterative process using a least-squares method, the best fit is determined, enabling the identification of the specific channel or sample.

The intensity-weighted average of the MS peak centroids, as described as 3D peak identification, corresponds to the mass of the peptide. The accuracy of mass measurements relies on the specific analyzer used. For an Orbitrap-type analyzer (which was used during this project), MaxQuant applies a correction value of 1 ppm (parts per million). Autocorrelation between centroids is accounted for by considering only well-identified peptides. Published data indicates that the mass precision within an MS experiment typically ranges around 10^{-7}.

Each peptide is represented by its individual fingerprint in the MS spectrum, which is based on the sequence and modifications of amino acids. The m/z ratio of the amino acids can be calculated, and the resulting peaks are interpreted as an amino acid sequence. However, only peptide fragments are visible in the spectrum due to peptide fragmentation. To identify proteins, peptides are matched against a sequence database. Sequence databases are typically stored as .fasta files and can be downloaded from the UniProt website (www.uniprot.org). The availability of gene sequences in databases and the computational matching of peptides against those sequences enable the identification of alterations in a sample at the protein level. These alterations can rely on the sequence level or could be to post-translational modifications (PTMs) such as phosphorylation, methylation (Aebersold & Mann 2003). The peptide identification (P-) score reflects the goodness of fit between the data and the identified sequence in the database, taking into account the length of the peptide. This score is used to calculate the posterior error probability, which is then utilized to estimate the false discovery rate (FDR). The FDR calculation provides a measure of the rate at which false identifications are expected in the dataset. Within the identified peptide groups, a "razor protein" refers to the protein with the highest number of ambiguously identified peptides. Quantification is performed by considering only unique peptides for comparison. Posterior error probabilities, which represent the likelihood of the identified peptide being a random event, are multiplied together, and only distinct sequences with the highest scores are considered for further analysis. Various metrics that indicate the performance of the peptide search can be utilized in

downstream analysis.

1.3.3 Downstream analysis

With advancements in mass spectrometry technology and computational methods, proteomic analysis has emerged as a powerful tool for investigating complex protein samples. However, analyzing proteomic data poses challenges in data processing, statistical analysis, and interpretation. This thesis aims to address these challenges by exploring computational methods for downstream analysis of proteomic data. The methods employed in this study will be thoroughly explained in the methods section, with a specific focus on developing an interactive environment equipped with a graphical user interface. This user-friendly interface will enable researchers, including those with limited computational expertise, to navigate and comprehend the data effectively.

The analysis pipeline comprises a series of steps to streamline the data. It begins with quality control measures at the peptide spectrum match level, involving the removal of contaminants and irrelevant channels. Significance levels of peptide spectrum matches serve as confidence thresholds for match selection. The data is then aggregated at the peptide level, followed by additional filtering procedures based on metrics such as reporter ion intensity, peptide covariance to razor proteins, and missing rates for peptides.

To derive meaningful biological insights from the dataset, statistical testing plays a crucial role. Selecting appropriate statistical tests and ensuring proper data transformation and normalization are essential steps. Many statistical methods rely on data distribution and require careful consideration to preserve the biological interpretation of the results. This thesis employs various transformation and normalization techniques, aiming to maintain data integrity and avoid biased interpretations. Additionally, an entropy-based visualization approach will provide users with a valuable tool for validating the computations and selected thresholds for the filtering steps.

Handling missing values poses further challenges in the pipeline development since a missing signal for a peptide results in a missing protein, which is highly unlikely in a biological context. Several options for missing value handling will be included and thoroughly explained in the methods section.

Dimensionality reduction and visualization techniques are widely used to cluster data. In this project, multiple approaches will be utilized, and users will have the ability to visually and interactively observe the data.

It is important to acknowledge that downstream analysis in proteomics lacks a standardized approach, and the selection of computational methods depends on the specific dataset and research

objectives. Furthermore, statistical analysis and the interpretation of biological meaning often present misconceptions that need to be addressed. For instance, batch correction, a commonly practiced method, has been overestimated in numerous studies, leading to misleading results (Nygaard et al. 2016). This thesis aims to contribute to the field by providing researchers with a comprehensive understanding of computational methods in proteomics analysis and their application in an interactive environment.

2 Materials and Methods

2.1 Wetlab - Materials and Methods

For testing and development of the pipeline the following materials and methods were performed. For the sample preparation protocol see appendix.

2.1.1 Celltypes

Macrophages from 12 healthy donors and 12 HIV patients with metabolic syndrome and 12 HIV patients without metabolic syndrome. Cell sorting was performed with FACSAria Fusion (BD Biosciences).

2.1.2 Digestion

Protein digestion was performed using trypsin (Promega, Madison, WA) buffered in Triethylammonium bicarbonate buffer (TEAB). Reagents were handled with MANTIS automatic dispenser (Formulatrix, Bedford, MA).

2.1.3 Labeling

Samples of test dataset were labeled with 12-plex TMTpro (Thermo Fisher) in anhydrous acetonitril (ACN) solution.

2.1.4 Pooling

Carrier proteome was added to the 36 TMTpro sets before pooling sample types for each of the 36 MS-runs.

2.1.5 Liquid chromatography

In order to separate peptides according to their chemical properties, size or species a liquid chromatography (LC) is recommended before ionization. For the reference data used for developing the pipeline the following intrumentation was used. EASY-Spray™ HPLC column, ES802A, 2 μm $\times$ 75 μm ID $\times$ 250 mm (ThermoFisher Scientific) Acclaim PepMap trap column, 2 μm $\times$ 75 μm ID $\times$ 2 cm (ThermoFisher Scientific)

2.1.6   Mass Spectrometry

2.1.7   Ionization

For the reference data ESI ionization was used.

2.1.8   Mass spectrometer

For the reference data used for developing the pipeline an Orbitrap™ Fusion™ Lumos™ tribrid mass spectrometer (ThermoFisher Scientific) was used. The instrumentation uses two subsequent runs in the analyzer (=Orbitrap) to increase mass accuracy.

2.2   Computational methods

2.2.1   Data-acquisition and raw-file analysis

For the data-acquisition Thermo Fisher acquisition software was used. Raw-file analysis of the data was done with MaxQuantCmd 2.2.0.0 (Cox & Mann 2008). The default settings were used. TMTpro labels were corrected according to the isotopical distribution of the Lot-number provided by Thermo Fisher.

2.2.2   Data processing

Further analysis is done with R and respective packages such as bioconductor. Please find version info in the appendix.

evidence file

sample annotation

read data into q–feature object

quality control 1

filter out pool samples
replace zeros with NA
filter out contaminants
filter by q–value cutoff

aggregate peptide spectrum matches to peptides

join multiple batches

filter by:
reporter ion intensity cutoff
pepitde covariance to razor proteins
peptide missing rate

quality control 2

aggregate peptides to proteins

transform data

options:
log2
log10
sqrt
quadratic
BoxCox
None

options:
col–Median,row_Mean
CONSTANd
None

normalize data

options:
KNN
drop rows
replace with mean
replace with median
replace with 0

missing value handling

ComBat
PCA
UMAP

differential expression
pathway enrichment
ontology enrichment

batch correction & dimensionality reduction

14

2.2.2.1 Reading the data After processing MaxQuant creates a directory containing all results as .txt file. The evidence.txt file include all peptide to spectrum matches (PSM) with their respective proteins and statistical parameters.

Example fo basic parameters and derivations include:

- Peptide sequence
- Mass to charge ratio (m/z) for all scans (eg. MS1, MS2)

  – Mass

- Retention time
- Precursor Ion Fragment

  – source of the detected ion also referred as mother ion

- Fraction of total spectrum
- Base peak fraction
- Reporter intensity (RI)

  – corrected RI

- Posterior error probability (PEP)

2.2.2.2 Object oriented programming In order to streamline the analysis of multiple experiments, object oriented programming was applied. The approach in R is to create a so called Q-feature object, which contains all variables and metadata in a hierarchical structure. The structure enables sub setting for further analysis (Vanderaa & Gatto 2021).

2.2.2.3 Zero values Peptides with low abundance are often set to zero during analysis. However, assigning a value of zero may incorrectly suggest that the sample does not contain the respective peptide. Given that it is highly unlikely for a biological cell of a comparable type and function to not contain a particular protein, replacing the zero value with "not applicable" (NA) is crucial for understanding and interpreting MS data.

2.2.2.4 Exclude reverse matches/contaminants Peptide sequences matching to the reverse protein sequences (=decoy database) are considered as possible contaminants. These matches were be excluded from further analysis.

2.2.2.5 Filter according to precursor ion fraction (PIF) During mass spectrometry, the ions detected in MS1 are further fragmented through collision during multiple MS runs. The resulting product ions are derived from precursor ions (also known as mother ions or parental ions). Contaminant peptides can co-migrate in this process and were distinguished by the lower fraction of their respective precursor ions (Tannous et al. 2020). These peptides need to be filtered out during the analysis pipeline. A cutoff value, referenced in the SCoPE2 pipeline (Specht et al. 2021), is applied in the user interface, but it can be adjusted according to the needs of the biologist.

2.2.2.6 Filter by q-value The next step for quality control is the exclusion of samples with a high false discovery rate (FDR). When applying multiple statistical testing (e.g. t-Test) the obtained p-values can be considered as biased, because the probability to observe a significant will iteratively increase with each test performed. Corrections in statistics are an approach to compensate for the multiplicity of testing. There are many ways to do this compensation like the Bonferroni method or Benjamini-Hochberg's FDR. In Mass spectrometry the common "way to go" is calculating a false discovery rate, by dividing false PSMs (=hit of the decoy database) through the total number of PSMs above the peptide-spectrum matching score. The peptides spectrum matching score is defined as -10log10(p). Whereas the p-value is defined that the hit is done by chance. The calculation of the score is highly dependent on the data acquisition method used. MaxQuant uses Andromeda , an integrated search engine. Proteome Discoverer from Thermo Fisher utilizes different engines such as Mascot or Minora. As published by J.Cox in 2011 Mascot and Andromeda showed similar performance when comparing FDR values as a function of coverage. However the observed performance can be lower when dealing with a decreased coverage (Cox et al. 2011).The threshold for accepting an FDR of an individual PSM is described as q-value.

2.2.2.7 Peptide spectrum match (PSM) aggregation to peptides In data science, aggregation refers to a row-wise operation that merges data based on a particular column using a specific function. In the context of processing from peptides to spectrum matches, the desired column is the peptide sequence. To account for different distributions across multiple assays, the median of the channel is used as the function to aggregate multiple matches into one.

2.2.2.8 Join assays when observing multiple comparable batches at once Sample size is often a limiting factor in hypothesis testing. A strict quality control and the fact that TMT reagents are only available up to 18-plex can reduce the number of observed samples below the critical

threshold, leading to an early end of analyses. To overcome this limitation, the provided software is capable of processing multiple runs simultaneously, allowing for testing of multiple batches and increasing the number of samples that can be included in the analysis.

2.2.2.9 Calculate reporter ion intensity (RI) and filter according to median RI

Columns which do not meet the desired intensity can be filtered by a threshold set on the RI. The median RI can also be used to check if an entire channel has a lower detection level. This can be due two reasons. One is the expression level of the given proteins in a cell. Meaning, that the expression of the observed cell type is simply lower than the other type. Another one could be a spillage of TMT detection in other channels due incorrect or missing correction of the TMT isotopes.

2.2.2.10 Calculate and filter according to median coefficient of variation (CV) per cell/channel

Depending having a bulk sample or single-cell sample choosing a minimum of observed peptides and a cutoff value for the CV, changes the level of confidence in the peptide data. The coefficient of variation of a peptide is considered as the ratio of the standard deviation to the mean and describes the relationship of the observed peptide signal over multiple proteins (=razor proteins). Peptides having a high coefficient of variation over many razor proteins are considered as noise and need to be filtered out before statistical analysis.

2.2.2.11 Remove peptides with high missing rate

Although missing value imputation can be performed during the analysis of multiple batches, peptides with missing detections across channels can be problematic for quantification. The proteomic composition of a biological sample is similar between replicates and even across groups. However, the threshold of missingness (described as a fraction of the row) can be set in the user interface and adjusted to enable different experimental designs.

2.2.2.12 Aggregation of peptides to proteins

Similar to the already explained previous aggregation step the peptides will be further processed into their respective proteins after the quality control on the peptide level is performed. Finally an expression matrix for every protein and their respective channel column is returned. Each channel refers to a sample and reflects contains the intensities of each protein found in the biological specimen.

2.2.2.13 Transformation of protein expression data Data transformation applies a function to each value of a matrix or array, so that:

$$y_i = f(x_i)$$

Depending on the distribution of the values in the observed expression set, different transformations can be applied to fulfill assumptions for statistical testing. In the shiny application, various procedures are implemented and can be further expanded upon request from the user. The macrophage analysis done by Specht et al. mentioned in their SCoPE2 publication (Specht et al. 2021) uses the logarithm to the base 2 to spread a compacted distribution and remove skewness in the dataset. This transformation was used as a reference when comparing methods. However, using the logarithm to the base 10 may be an easily interpretable way of defining expression data and will also be facilitated by other transformation methods such as the boxcox method, which is also implemented in the application. When observing a wide distribution of small and large values of the expression set in a histogram, a square root transformation can help increase the variability of smaller values and decrease the variability of larger values. Practically impossible, but in data-driven science, occasionally negative expression values may be present, and they can be converted into positive values by taking each one to the power of two. Depending on the distribution, the user has to decide which transformation to consider and verify that statistical assumptions are met after application with visualizations such as histograms, qqplots, and MA-plots. For statistically inexperienced users, the boxcox transformation can help with this decision. Developed in 1964 by Box and Cox (Sakia 1992), this method applies a linear model against 1 on the data to determine the statistical parameter lambda by the maximum likelihood method. The log-likelihood method takes the 95% confidence interval for the parameter lambda and the final lambda is chosen as the value with the highest log-likelihood value.

λ	Transformation
-2	$\frac{1}{x^2}$
-1	$\frac{1}{x}$
-0.5	$\frac{1}{\sqrt{x}}$
0	$\log x$
0.5	$\sqrt{x}$
1	x
2	x^2

Depending on the size of lambda, a certain function will be automatically applied to each value of the expression set in order to introduce normality. In terms of usability, the boxcox method is the most efficient transformation method, taking the decision of which calculation to apply away from the user.

2.2.2.14 Normalization The term normalization is ambiguous in data-science and will be explained briefly in this chapter. Starting with converting the data to the Z-distribution (Gaussian result), where 0 represents the mean and 1 represents the standard deviation, in vectorization every feature (or protein) is represented as a vector that points in a specific direction in the unit sphere, normalization can have different meanings. Normalization is considered as part of the scaling procedure and ensures that every feature contributes equally to our statistical model and hinders large values from biasing the model in a particular direction. However, this could also have negative consequences for the modeling procedure, as it may reduce the impact of important features on the dataset. Depending on the algorithms later used in the data processing, the method of choice for normalization can change the results significantly. The K-means clustering algorithm, for instance relies on the distances between distinct data points by minimizing variance of the squared euclidean distances. This method will be applied in the missing value imputation which will be the next step applied to our data. Therefore, we want to ensure that we obtain the most accurate imputed values possible. The first normalization method available in the user interface is column-wise median and row-wise mean normalization. This method was chosen by Specht et al in the SCoPE2 publication 2021 (Specht et al. 2021) and worked as a reference during the development process to benchmark other methods. When dividing each value of an expression set by the median of the particular column, the procedure is considered as a column-wise median normalization. Although the mean-normalization works the same way but takes the mean as the second variable. In the analysis pipeline both procedures were applied on the dataset. However when the logarithm as the transformation method was chosen as transformation method, the pipeline automatically switches from a division to a subtraction, since: $\log_a \left(\frac{u}{v}\right) = \log_a u - \log_a v$. After applying this normalization method, each value can be interpreted as a distance or fraction of the mean or median of the corresponding column or row, enabling fair comparisons between features and samples. The second method for normalization included is the CONSTANd method, which was proven suitable for relative quantification in the field of proteomics (Maes et al. 2016, Houtven et al. 2021), but can also be applied on RNAseq data. CONSTANd uses a technique called matrix raking, which employs the RAS algorithm. The expression set reflects the nonnegative real (m, n) matrix A were the bi proportional constrained matrix problem will be solved by finding the (m, n) matrix B which equals

to $diag(x) * A * diag(y)$. Whereas $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$. The solution to this problem involves finding the row sum of $B = u_i$ and the column sum of $B = v_j$, where i and j denote the row and column indices of the matrix, respectively (Bacharach 1965). In other words, the matrix will be alternatively manipulated on rows and columns until the mean of both equals 1. In order to apply the method and yield true results, one has to consider 3 major assumptions between sample types. This assumption can be observed in the MA-plot which indicates the differences of two samples. If we have the intensities of two samples, R and G, we can plot them on a graph with the x-axis as $M = \log_2(\frac{R}{G})$ and the y-axis as $A = \frac{1}{2} \log_2(RG)$. A comparison between all sample types of identically processed sets is needed before calculation and the assumptions must be fulfilled.

- **1 The majority of proteins are not differentially expressed**

  Density of the dots decreases when looking from the middle of the cloud towards the edges.

- **2 Up- and downregulation is balanced around the mean expression**

  The scatterplot of the data points is symmetrical around the mean on the horizontal axis.

- **3 Systematic bias correlates with the magnitude of expression**

  The symmetry axis is approximately horizontal.

If the MA-plot does not visualize any of these violations the CONSTANd method can be applied on the dataset and the biological meaning of the sample types can be assessed. The method is utilized by the bioconductor package handler. However if the dataset already consists of balanced features and samples or the user wants to benchmark the methods applied, normalization can be skipped also.

2.2.2.15 Missing value handling   When analyzing multiple batches simultaneously, it is possible to encounter situations where a particular protein is not detected in one of the batches. From a biological perspective, it is highly unlikely that a cell completely lacks a single protein, as it would imply a complete loss of function for that protein. Therefore, an intensity value of 0 is close to impossible for most proteins. In order to handle this dark space in the expression set, several methods can be chosen in the application. A variety of function employs replacing the missing values with the mean or median of the rest of the matrix. This procedure introduces no bias in the data if the missingness is low, although differential expression for the sample carrying the missing feature can not be expected as well. Even a more conservative approach is dropping the rows for the feature with the missing value in one of the observed samples. A common practice in proteomics is utilizing the K-nearest neighbor algorithm (=KNN), which imputes the missing values by a euclidean metrics

of the neighboring values in the columns where the feature is not missing. K denotes for the count of neighboring values of a gene and can be selected upon preference. The euclidean distance between two points is defined as $d = \sqrt{a^2 * b^2}$, the K nearest neighbors average value will be assigned to the intensity of the missing feature. Since the average is a statistical metric which can be biased easily by skewed distributions, normalization is an important task before performing missing value imputation. KNN was benchmarked for the regression of protein abundance intensities and showed an Area under the Curve (AUC) above 0.7 when testing on a human proteome test set (Lan et al. 2013).

2.2.2.16 Batch correction Since mass spectrometry experiments are highly influenced by several factors, such as sample processing, the technician, manufacturer and production of the device, the reagents etc., working in multiple batches introduces further variance in the protein data. This variance would overshadow the biological differences across sample types when performing dimensionality reduction. A common misconception in biology is, that batches need to be corrected before differential expression analysis, which is the final aim of the developed pipeline. It appears convenient to remove the batch effect before performing any statistical analysis, which is in turn prone to errors especially if the sample groups are not split equally between batches. In this case group differences influence the batch effect reducing statistical power after the correction., the both effects are so called interdependent. The opposite could happen if the experimental design is heavily unbalanced leading to group differences induced by batch correction. Finally it is recommended to include the batch factor should in the linear model as a co factor and statistically quantify it. (Nygaard et al. 2016) However it could be of interest to observe biological relevance of the factor of interest in the dimensionality reduction visualization, so a batch correction option was included to meet this need. To address the non-biological experimental variations, also known as technical factors, the R package sva (Leek et al. 2012) with the ComBat function (Johnson et al. 2007) was used. The expression of a gene g for the batch m_i and sample j can be defined as $Y_{ijg} = a_g + X\beta_g + \gamma_{ig} + \delta_{ig}\epsilon_{ijg}$, where γ and $\delta\epsilon$ represent the systematic error caused by the batch. The Bayesian framework is employed to remove this systematic error by shrinking the effect and pooling information over the features g. Ideally the effects a and β remain as natural as they are and should reflect the true biological difference between experimental obtained samples or groups. Before applying batch correction, the data is standardized using Z-transformation. By the method of moments (= the mean and the variance) the two batch effect parameters γ and $\delta\epsilon$ were estimated and can be subtracted from each value for the particular feature driving statistical moments at a comparable level between batches. The ComBat function allows the user to specify a desired effect,

which represents the actual biological variance to be preserved while removing non-biological variance. This is achieved by constructing a design matrix, which is provided as prior information about the dataset within the Bayesian framework. However, in cases of poor experimental design, the factors contributing to unwanted and desired variation may be confounded. In the worst case, if only one sample type per run is observed, it becomes challenging for the program to distinguish between batch-induced variance and biological variance. In such cases, the user is advised to adjust the experimental layout for future experiments. The detection algorithm for confounding effects is facilitated by QR matrix decomposition. The matrix $A = QR$ is decomposed into an orthogonal matrix Q and an upper triangular matrix R. By comparing the rank of matrix A with the number of biologically relevant factors, it is possible to determine the presence of confounding effects in the dataset. The rank is defined as the maximum number of linearly independent columns. The same approach is used in the limma package (Phipson et al. 2016) to identify non-solvable coefficients in experimental designs.

2.2.2.17 Dimensionality reduction After cleaning the data and crystallizing from impurities such as noise and unwanted effects, the biological differences between sample types can be obtained. Proteomic data consists of multi dimensions (=features) which are overall hard to understand by a glance. An approach to observe the differences without a closer sight of exact effect on the proteinaceous ensemble, but still keep a high level of information can be dimensionality reduction. After the cleaning procedures any additional effect caused by technical or biological replication should be removed, however it could be found be highlighting according to the particular factor. This makes the dimensionality reduction a useful tool before starting any statistical analysis by checking for involuntarily introduced bias.

2.2.2.17.1 Principle component analysis (PCA) The most common approach in data science for any kind of high dimensional data is the PCA. For the pipeline the bioconductor package scater (McCarthy et al. 2017) was used. Every principle component can be understood as a vector pointing in a specific direction derived as the eigenvector from the covariance matrix of the expression set. The covariance matrix indicates for the co linearity between the variance of all elements compared. After calculating the covariance matrix the eigenvectors for the covariance matrix will be obtained by solving the quadratic equations to obtain the eigenvalues λ. Since $Av = \lambda v$ the eigenvector v can be obtained by solving the equation for each element of the vector. The eigenvectors v never changes is direction and explains the covariance within the dataset and so

also the mathematical differences between the samples. The eigenvectors are sorted according to their eigenvalues in decreasing order and therefore decreasing explained variance. For graphical visualizations the principle components are plotted against each other. The first two components, which explain the majority of variance in the dateset highlight the difference between samples.

2.2.2.17.2 Uniform manifold approximation & projection (UMAP) UMAP (McInnes et al. 2018) is a dimensionality reduction technique that was developed in recent years and is based on Riemann geometry. It shares similarities to the t-distributed stochastic neighbor embedding (t-SNE) in terms of graphical visualization. Riemann geometry is a mathematical framework that deals with three dimensional functions using volumes, areas and vectors, which are also concept of topological data analysis. Before performing UMAP, three assumptions need to be considered:

1. There exists a manifold on which the data would be uniformly distributed. → This assumption implies that the probability for each value in the biological dataset is the same.

2. The underlying manifold of interest is locally connected. → In other words, similar samples will have a similar proteinaceous ensemble.

3. Preserving the topological structure of this manifold is the primary goal. → This means that the homeomorphic structure (the property of being able to transform one shape into another without tearing or gluing) is locally preserved, and local values in the dataset can be compared with each other.

(McInnes et al. 2018)

The UMAP algorithm begins by constructing a graph with a defined neighborhood parameter, typically denoted as k, which can be selected computationally. UMAP employs the k-nearest neighbor descent algorithm (NN-descent). Similar to the k-nearest neighbor algorithm (KNN), used in the missing value imputation part of this project, NN-descent recursively iterates to find the smallest distance to the respective neighbors within the defined neighborhood size k. Increasing the value of k gradually strengthens the clustering of the data, but at a certain point, fine structure may be lost, and only rough estimates of the underlying principle can be observed. The mathematical graph in UMAP is defined by nodes connected by edges, with each node internally connected to at least one other node. Edges are distinguished from each other not only by their connection but also by their importance, which is often referred to as weight. Whereas the weight function is calculated as $w((x_i, x_{i_j})) = \exp(\frac{-max(0, d(x_i, x_{i_j} - \rho_i)}{\sigma_i}$ and determine the parameter ρ for the weight function that:

$\rho_i = \min\left\{d(x_i, x_{i_j}) \mid 1 \le j \le k, d(x_i, x_{i_j}) > 0\right\}$. Furthermore σ will be set to meet the following conditions: $\sum_{j=1}^{k} \exp(\frac{-\max(0, d(x_i, x_{i_j}) - \rho_i)}{\sigma_i}) = \log_2(k)$ and acts as a normalisation factor. The obtained graph $\overline{G}$ is called the fuzzy simplicial set which is obtained by approximating the geodesic distance (= locally length-minimizing curve) of the data points, leading to a topological approximation. The second phase of the algorithm adjusts the layout of the weighted graph to a representative view, but preserves the characteristics and shows the underlying topological principle of the data. In order to obtain a visual representation of the graph, the algorithm applies repulsive forces $\frac{2b}{(\epsilon + ||y_i - y_j||_2^2)(1 + a||y_i - y_j||_2^{2b})}(1 - w((x_i, x_j)))(y_i - y_j)$ among vertices and attractive forces $\frac{-2ab||y_i - y_j||_2^{2(b-1)}}{1 + ||y_i - y_j||_2^2}w((x_i, x_j))(y_i - y_j)$ along edges. Where a and b are denoted as hyperparameters. By iterating through possible conformations until converging towards a local minimum with decreasing repulsive and attractive forces, the UMAP reaches it's aim to reveal the underlying properties of the data. The application utilizes the bioconductor package scater for performing UMAP (McCarthy et al. 2017).

2.2.2.18 Validation of the methods applied to the dataset

In the context of the validation of methods applied to a dataset, the mutual information calculation is used to assess the impact of computations on the dataset and the information it carries. Mutual information, denoted as $I(X; Y)$, is a measure of the dependence or information shared between two variables, X and Y, in this case sample types. It is calculated using the equation $I(X; Y) = H(X, Y) - H(X|Y) - H(Y|X)$, where H represents the entropy, $H(X|Y)$ and $H(Y|X)$ denote for the both conditional probabilities and $H(X, Y)$ for the joint entropy. Entropy, as a concept in information theory, quantifies the level of disorder or uncertainty associated with a random event. Events that are highly likely to occur are considered less informative than events that are more unexpected or less likely. In a Venn diagram $I(X; Y)$ (the mutual information) corresponds to the intersection of the two conditional entropys $H(X|Y)$ and $H(Y|X)$. To calculate the mutual information between two variables X and Y we can alternatively express the above equation as $I(X; Y) = D_{KL}(P_{(X,Y)} || P_X \otimes P_Y)$. D_{KL} denotes for the Kullback-Leibler divergence which is a concept of relative entropy between two variables and tells us how much 2 probability distributions differ from each other and $P_X \otimes P_Y$ represents the product of the marginals. The underlying principle of divergence is the integral asymmetry in Bayesian inference, which means that divergence does not satisfy the triangle inequality, and differences are not the same depending on the starting point. By comparing the mutual information before performing any calculations with the mutual information calculated after the computations, it is possible to assess whether the applied methods have significantly altered the dependence or

information content. The goal is to choose a method which preserves the information contributed by each biological sample type to the experiment. The R package "infotheo" is utilized to perform the estimates for the entropy of the two variables. Mutual information is returned as natural unit of information (=nat), a measurement proportional to the to the Shannon entropy $(1nat = \frac{1}{ln2}shannons)$. Before calculating the pairwise mutual information of the expression matrix, values were discretized into bins with the discretize function of above package. The mutual information is calculated within each sample type, and it can also be calculated between a selection of sample types to determine if the computations performed affect the information content. After that all pairwise mutual information within the expression set before and after the pipeline is obtained. The pairwise differences can be observed as a boxplot for a particular sample type indicating the change in dependence within and also compared to another sample type.

2.2.2.19   Statistics

2.2.2.19.1   Differential expression   After visualizing the biological tendency in the dataset, the next step is hypothesis testing. Several experimental designs can be tested against the null hypothesis h_0, which states that there is no significant difference between the sample types. Before proceeding with any statistical testing, it is important to observe the data using either a quantile-quantile norm plot (qq-norm plot) or a histogram. The quantile-quantile plot (qq-plot) serves a vital role in exploratory statistics by visualizing whether two datasets follow the same underlying distributions. In the case of statistical preparation for parametric tests, the first distribution is the dataset of interest, while the second dataset is the normal distribution. If the dataset follows an exact normal distribution, the values will align with the red line in the chart. However, values at the ends of the distribution may deviate from the normality line, indicating differentially expressed genes. These genes can be observed both in the histogram and the qq-norm plot.

Parametric statistical tests rely on the assumption of normality in the data, which needs to be achieved before building a model. To achieve normality, it is recommended to try various combinations of transformation and normalization methods. By observing the results in the plots, one can determine which combination of data processing methods satisfies the conditions required for the tests.

2.2.2.19.2   Differential expression   To test for differential expression, the analysis utilizes the "limma" package from R Bioconductor (Phipson et al. 2016). This package employs a linear model

approach to determine the fold expression between sample groups. Before initiating the analysis, two matrices need to be computed. The design matrix identifies samples based on their sample type and defines the experimental design. An algorithm within the program's backend logic is utilized to automatically generate this matrix according to the user's selection in the interface. The expression matrix contains intensities for each identified protein and is obtained at the end of the pipeline. By calling a function with the design and expression matrices as arguments, the contrast matrix is calculated, enabling the user to decide which comparisons to consider and also perform tests on multiple factors. When observing multiple batches at once, as already explained in the chapter batch correction this effect needs to be taken into consideration when performing differential expression analysis. This is done by the multi factor option in the user interface by selecting the batch as a co factor and inclusion in the design matrix. The advantage of this procedure is, that we can quantify the biological relevance of working in multiple batches and avoid publishing false positive results. The multi factor option uses an additive model as a design matrix, whereas continuous variables (such as size, weight etc.) can also be used to build the statistical model.

The subsequent step involves creating a linear model between groups using log-ratios of their expression values. In a two-way design, the expression of the gene y can be explained as $Exp_Y = \beta_0 + \beta_1 X_1 + \epsilon$, where β_0 represents the intercept, which can be interpreted as the mean expression of the gene. β_1 represents difference in mean of the treatment (or condition) on the discrete or continuous variable X_1, and ϵ acts as an error term. The interface allows for additive models with multiple factors, allowing the user to add factors according to various scenarios. For example, in a two-factor model, it would look like: $Exp_Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon$. The principle remains the same for models with multiple factors. When visualizing the linear model, the x-axis would represent the gene for the sample types or other factors, whereas the y-axis would account for the expression value. By drawing a line through the cloud of data points the model is constructed by minimizing the quadratic distance to every data-point, called the least-square method or also ordinary least-square method. With the regression line the above mentioned equation can be constructed.

Next the statistical module of the program performs an empirical Bayes model utilizing the eBayes (Smyth 2004) function of the Limma package. The eBayes function utilizes a moderated t-statistics following t-distributed values. The advantage of this method over the posterior odds is the reduction of hyperparameters to estimate used for the modelling process, also called shrinkage. In this context, the hyperparameters to estimate are the coefficient β_{gj} for gene g and sample i, as well as the variance σ_g^2 across genes g. In the Bayesian approach, probabilities are updated after obtaining new data, referred to as conditional probabilities. For estimation, it is assumed that $\frac{1}{\sigma_g^2} \sim \frac{1}{d_0 s_0^2} \chi_{d_0}^2$,

representing the prior information of the model. The probability that $\beta_{gj} \neq 0$ is denoted as p_j, where p_j represents the expected proportion of truly differentially expressed genes. The expected distribution of log-fold changes follows the distribution $\beta_{gj}|\sigma_g^2, \beta_{gj} \neq 0 \sim N(0, v_{0j}\sigma_g^2)$. The posterior mean of σ_g^2 given s_g^2 is calculated as $s_g^{-2} = E(\sigma_g^2|s_g^2) = \frac{d_0 s_0^2 + d_g s_g^2}{d_0 + d_g}$. This is where the shrinkage occurs, as the posteriors affect the prior values based on their respective sizes and degrees of freedom. The moderated t-statistic is defined as $\tilde{t}_{gj} = \frac{\hat{\beta}_{gj}}{\hat{s}_g \sqrt{v_{gj}}}$. Once the values of $\tilde{t}$ and s^2 are calculated, the posterior odds are computed, providing the odds that a particular gene is differentially expressed. The odds for gene g being differentially expressed are denoted as $O_{gj} = \frac{p(\beta_{gj} \neq 0|\tilde{t}_{gj}, s^2 g)}{p(\beta_{gj}=0|\tilde{t}_{gj}, s^2 g)}$, where the numerator represents the probability that the gene is differentially expressed, and the denominator represents the probability that the gene is not differentially expressed (Smyth 2004).

Limma outputs a table with all proteins, their respective logfold change, p-values and adjusted p-values. The user can choose different methods for the correction of the p-values, which is necessary when testing the same hypothesis multiple times. A common method in omics studies is the false discovery rate (=fdr) by Benjamini Hochberg (Benjamini & Hochberg 1995). By ranking all the p-values from smallest to largest the p-values are adjusted sequentially using the formula $p_{(i)} = \frac{r_i}{m}Q$. The rank in the p-value table is denoted as r_i , m reflects the total number of tests and Q represents false discovery rate which is set to 5%. The second option to choose is the Benjamini & Yekutieli (=BY) (Benjamini & Yekutieli 2001)method, which is a further development of the fdr by the same statistician and has a wider range of application based on the dependencies for the test. The third and most conservative option is the Holm method (Holm 1979) which controls the error sequentially in a family wise manner, which is considered as traditional in regards to the above mentioned experimental wise error rates. The above mentioned correction methods are ordered in the way from the most liberal to the most conservative one. The selection is supposed to give the user the possibility questioning results on multiple levels.

2.2.2.20 Protein set enrichment analysis   After differential expression the upcoming question could be, in which context the up or down regulated proteins are. Enriched proteins can be understood as an over-representation (ORA = over-representation analysis) within a specific set. This set can be a pathway or a selection of common denominators in biological contexts, such as cell types or oncological factors. The statistical testing to identify the overabundant proteins of a specific set is similar to a χ^2 test developed by Karl Pearson (Pearson 1900) or the exact Fisher test (Sprent 2011) for tables with 4 fields in small sample sizes. Within the list are all truly differentially expressed genes, an annotation is the identification of a protein in the desired biological context. We

expect that the values populate the fields in the table as follows:

	in list	not in list	totals
with annotation	(A+B)(A+C)/N	(A+B)(B+D)/N	A+B
without annotation	(C+D)(A+C)/N	(C+D)(B+D)/N	C+D
	A+C	B+D	N

Under the null hypothesis H_0 states that the distribution and population of the table is random and there is no clear tendency with significant evidence. Against that the alternative hypothesis H_A says that there is an underlying tendency the table is populated. This leads to our test statistics, the observed values ($=O$) in the table are significantly different to the expected ($=E = \frac{(A+B \vee C+D)}{N}$ ones. One approach is approximating $\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$ and deriving the p-value by the area under the density curve of the χ^2 distribution $Q = \sum_{i=1}^{k} Z_i^2$. In the distribution k is defined by the degrees of freedom ($=$df), what can be obtained by $k = (columns - 1) * (rows - 1) = df$ of our table. Another approach to obtain an exact p-value would be Fisher's exact test, which states that the margins are distributed according to the hypergeometric distributions. The test statistic leads directly to the p-value with no need for approximation: $p = 1 - \sum_{i=0}^{A-1} \frac{\binom{B}{i}\binom{D-B}{A+C-i}}{\binom{D}{A+C}}$ Here D denotes either for the background distribution which can be the total number of proteins obtained in our experiment, all proteins with annotation or a custom selected background. The background indicates all proteins which could be positive tested for in our observed sample. The above mentioned three options are implemented in the user interface. After obtaining the p-values for each protein set, the subsequent critical step involves the correction for multiple testing, considering the inherent risk of false positive results. Multiple testing correction methods adjust the significance thresholds to account for the increased probability of detecting false positives when performing multiple statistical tests. In this protein set enrichment analysis, the user is provided with a selection of established correction methods, akin to those commonly utilized in the field of differential expression analysis, ensuring robust and reliable interpretation of the results. Once the enriched protein pathways are determined, they can be visually represented using bar or dotplots, and a network of the found pathways. In the network ontologies are depicted as nodes, whereas edges show the strength of the interaction by thickness corresponding to the number of associated proteins.

2.2.2.20.1 Pathway based protein enrichment analysis The pathway-based protein enrichment analysis is performed using the R Bioconductor package clusterProfiler (Wu et al. 2021). The analysis utilizes the "Kyoto Encyclopedia of Genes and Genomes" (KEGG) database, which is

accessed through a function that maps the UniProt IDs to corresponding pathways.

2.2.2.20.2 Custom ontology based protein enrichment analysis For custom ontologies the protein enrichment analysis is done using the R bioconductor package piano (Väremo et al. 2013). In opposite to the pathway enrichment, piano maps proteins to a pre-selected gene collection set, which can be either downloaded from gsea-msigdb.org or custom made for the particular experiment and explains conditions, phenotype or other desired properties. For statistical testing the above explained Fisher exact test is used.

3   Results

After runring MaxQuant mode with MS2 search for TMTpro reporter ions with a runtime of 6 hours, the data was analyzed using the developed shiny application. Using the application starts with uploading the tabdelimited evidence and sample annotation file. The application is ready to run with default values. Fine tuning will be done subsequently.
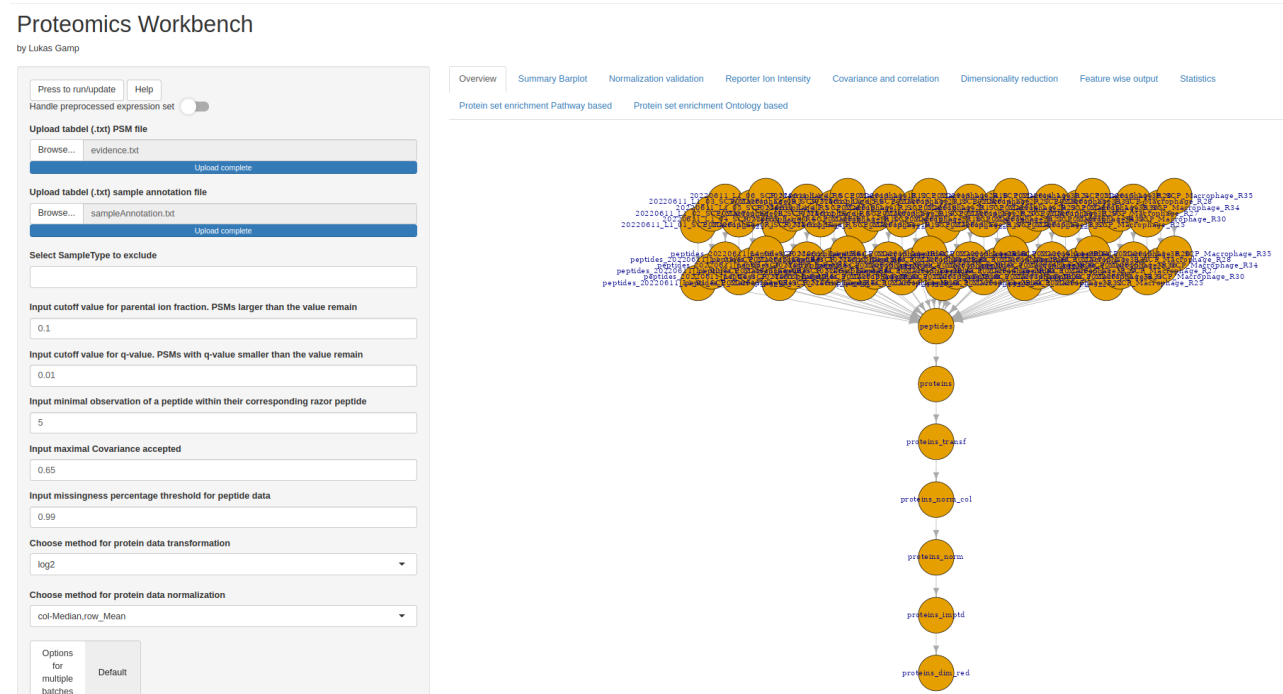

Figure 1: Summary showing the process of the analysis pipeline

On the left side of the user interface the analysis pipeline settings ca be changed. With the update button the user can apply the changes made and observe alterations in the dataset directly. The summary visualization looks cluttered since 36 files were processed, in smaller datasets the nodes of the network graph can be distuingished clearly.

By running the application with SCoPE2 (Specht et al. 2021) reference cutoffs the following results were visualized.

The default settings yield a total number of unique proteins below $\approx 10^3$ aggregated from $10^{4.5} \approx$ peptide spectrum matches also aggregated over $10^{3.5} \approx$ peptides. The aggregation is set fixed by median.

The mutual information (MI) within the sample type of the healthy control group shows a significant reduction after the transformation and normalization. When comparing the MI with the sample type HIV with metabolic syndrome, the reduction is within the inter quantile range (IQR = 25 - 75
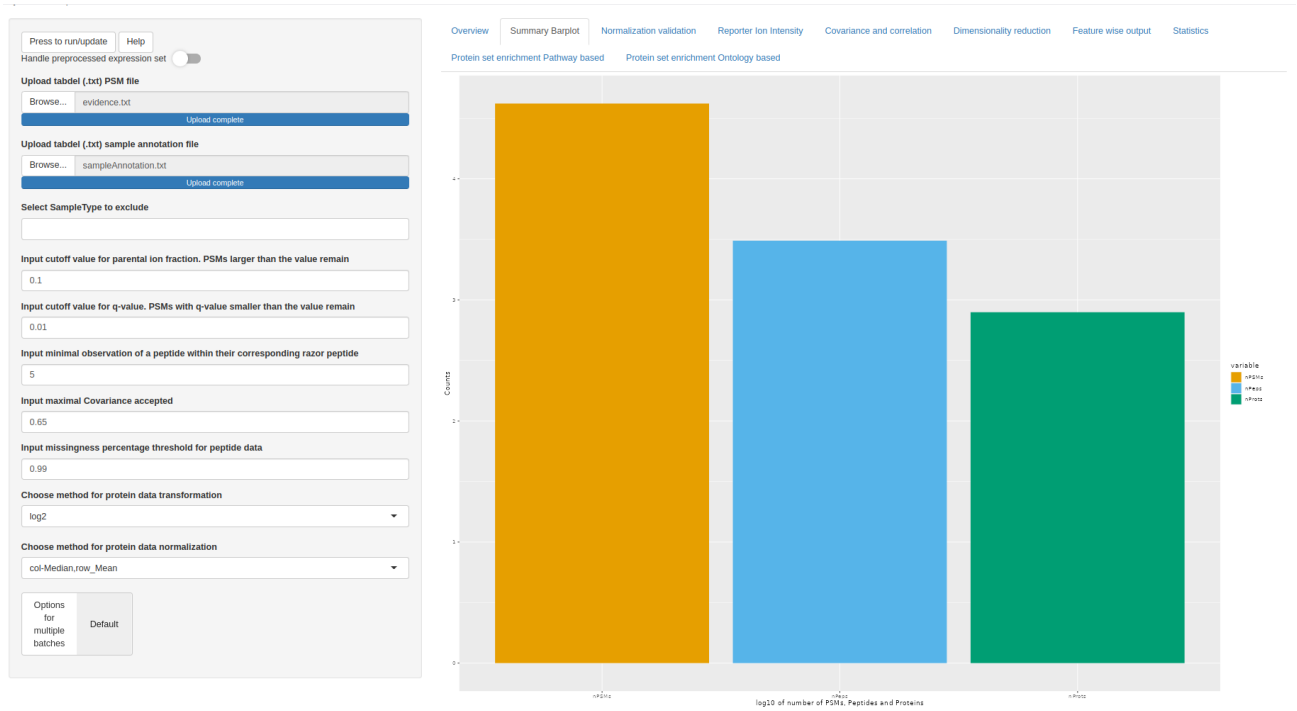
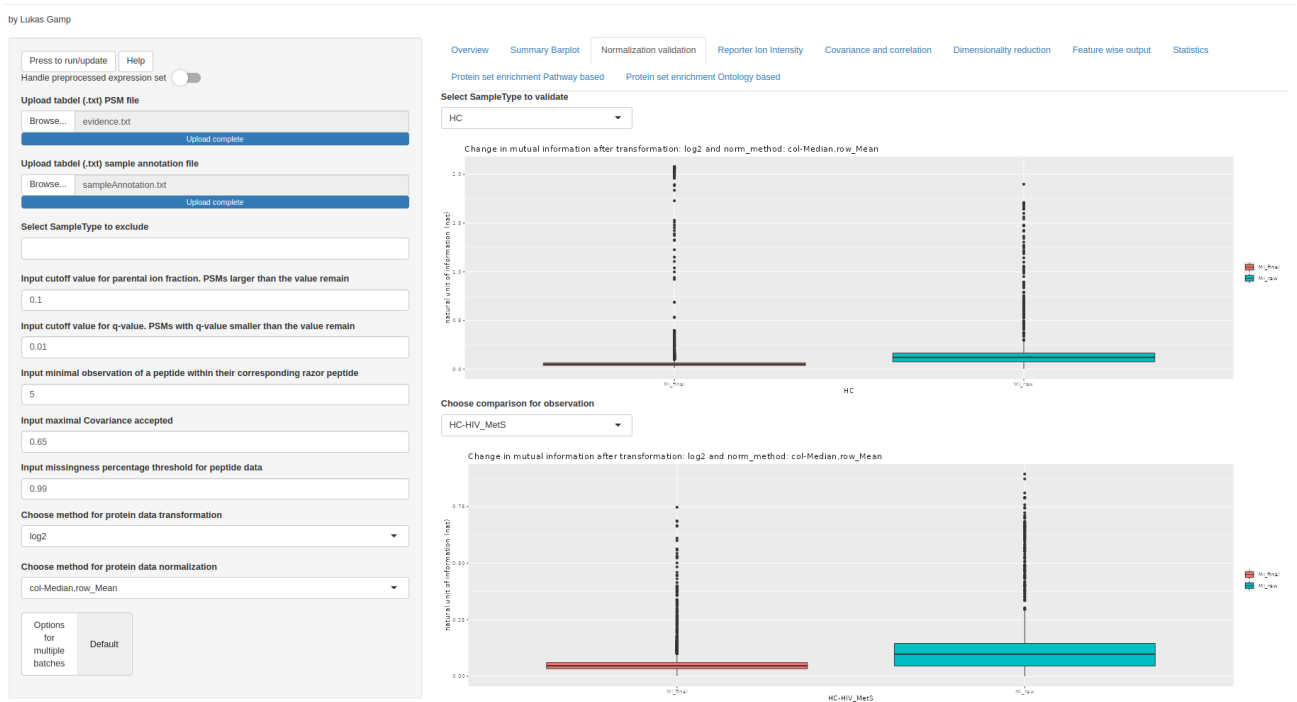Figure 2: Barchart showing the number of peptide spectrum matches, peptides and proteins

Figure 3: Mutual information with default setting
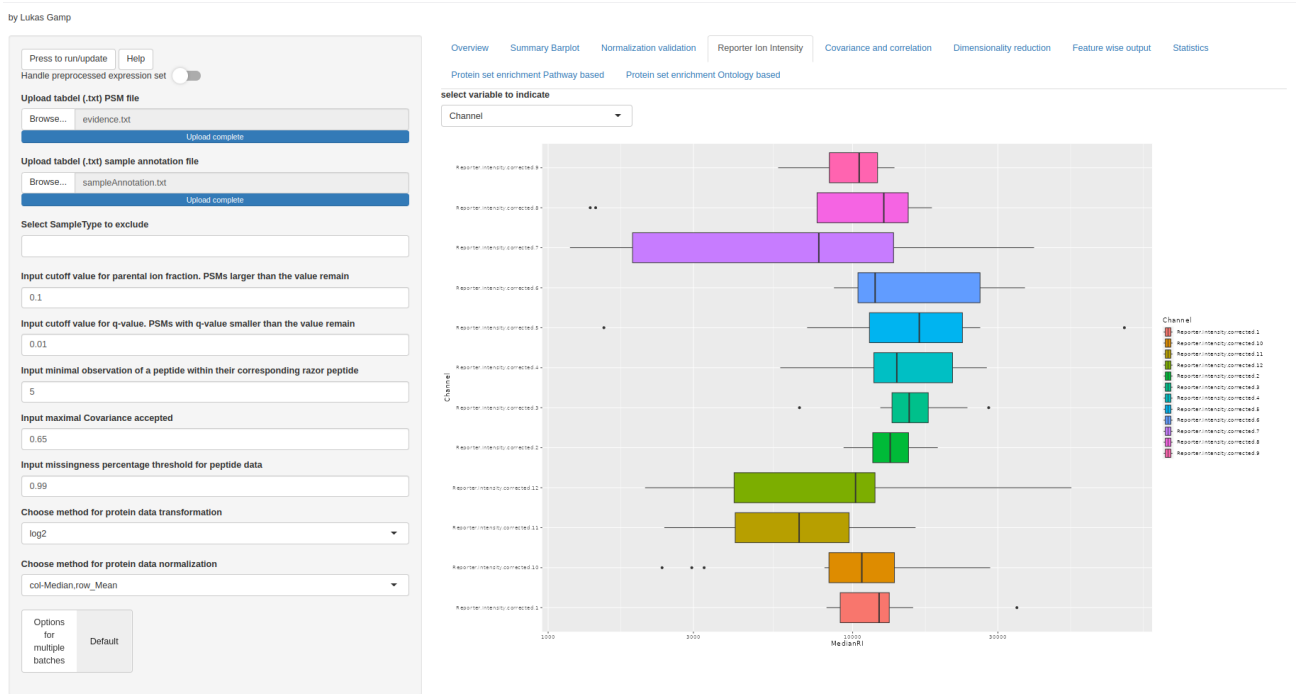
quantile) of the raw data MI.

Figure 4: Reporter ion intensity with selection on channel. The selected factor will be highlighted and indicated in the legend.

When comparing the median reporter ion intensities over channels show no systematic bias. Considering the 3 sample types occupy in every run the same channels, it is important to look at this visualization. Since the IQR overlap over the majority of samples, the change in expression with the factor channel can be considered as non-significant. However the channel will be later on tested for significance in the dimensionality redution plot and statistic module.

Comparing median reporter ion intensities with the focus on the batch (also Raw.file), it is obvious that the changes in signal are significant and need to be considered in further analysis. For the dimensionality reduction the user can choose in the sidepane the batch correction option ComBat (Leek et al. 2012). However this option is not chosen as default, since the conservative method includes the batch factor in the statistic module.

Observing the Median RIs with a focus on sample types, IQRs partly overlap so the binding efficiency of the TMTpro reagent is not affected by the sample type.
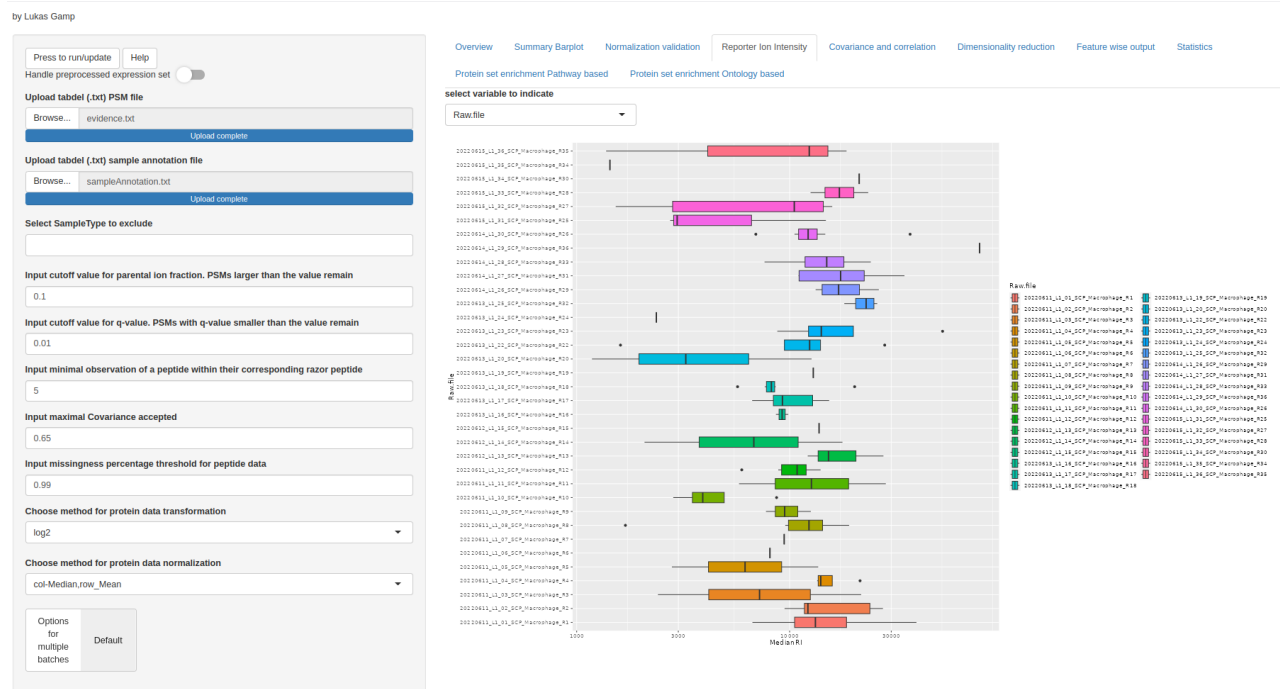
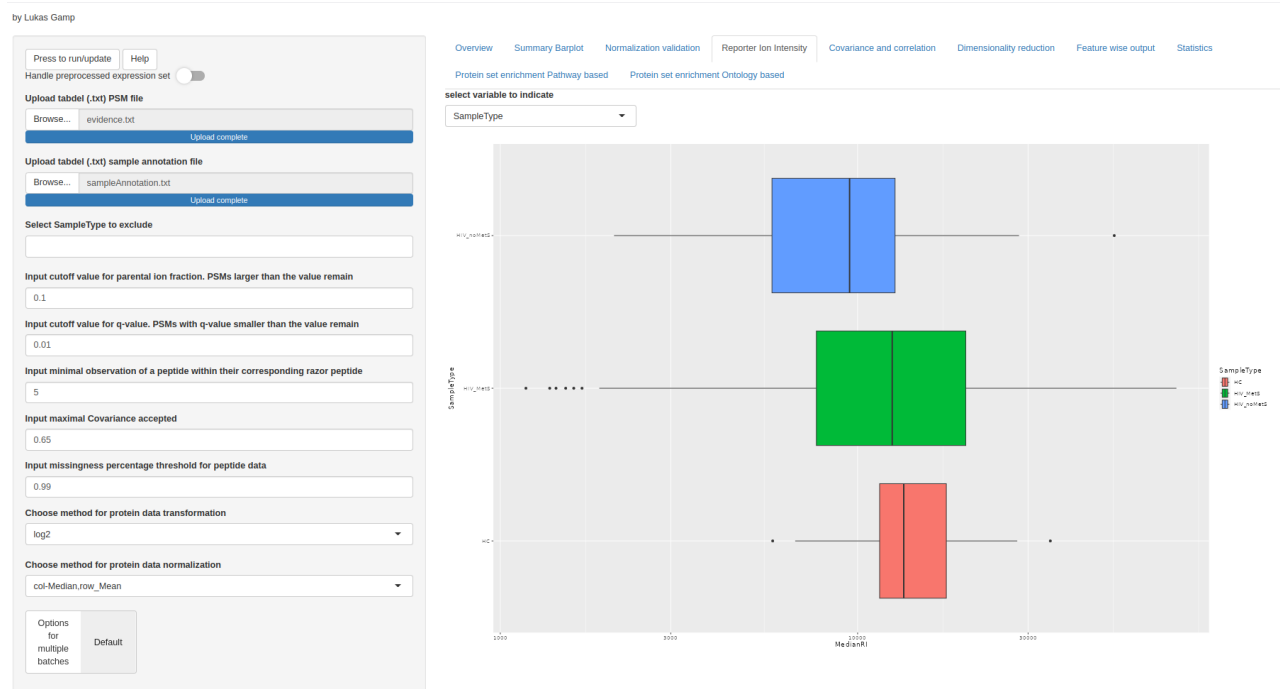Figure 5: Reporter ion intensity with selection on file

Figure 6: Reporter ion intensity with selection on sampletype

4 Discussion

The developed application and the proteomics experiment pipeline demonstrated how interactive engagement with the data not only enhances the experience of biologists but also improves the comprehension of the underlying significance of a biological dataset. By utilizing an entropy-based visualization approach, the conservation of information can be validated, allowing users to select appropriate methods and adjust thresholds, cutoffs, and techniques accordingly. With the selection of factors to indicate over multiple plots, biases can be identified on various levels. Additionally, the analysis can be examined on an individual protein basis, enabling the identification of suspicious results. The statistics module provides users with a simple tool to assess and visualize the results, offering clarity to dense scatter plots through hover functions. Enrichment mapping links statistically significant proteins to either the KEGG pathway database or individual research datasets, depending on user selection. These final results place the data in a biological context, which can be explored interactively in graphical visualizations such as network.

Information theory and the concept of mutual information provide a framework for understanding the conservation of entropy in a dataset. Traditionally, the validation of data transformation and normalization methods has been limited to observing distributions through histograms or Quantile-Quantile plots. However, the developed approach goes beyond these conventional methods, providing a deeper context for the variation in the data. By allowing users to choose the method that preserves the maximum amount of information, the approach enables researchers to extract the most valuable insights from their experiments.

Linear programming scripts are widely utilized for data analysis, and R-shiny provides an efficient platform for implementing these scripts in applications. This approach not only benefits scientists who are not proficient in programming, but it is also convenient for programmers. The developed user interface offers a visually appealing design, and the interactive menus enhance the user experience by eliminating the need to search for specific lines of code on a black screen. Modifying parameters of functions becomes effortless, and data can be visualized without rerunning the entire script. R-shiny executes functions on demand, which means that graphics, variables, and data frames are not recalculated if their dependencies remain unchanged. This feature makes the application both cost-effective and time-efficient.

The analysis pathway is visualized through multiple components, providing users with a comprehensive view of the data. The overview plot, generated using the Q-feature object-oriented programming in the scp R library (Vanderaa & Gatto 2021), serves as a visual representation of the

analysis process. This plot allows users to track the progression of the analysis and identify any missing files or samples. MS2 reporter ion intensities can be visualized as medians in form of a boxplot, allowing users to assess the distribution of intensities across different factors such as sample type, batch, or channel. This visualization helps identify potential biases or variations in the dataset.The feature-wise output further enhances the analysis pathway by enabling users to select individual proteins and explore their associated peptide spectrum, peptide and protein values over the course of the analysis. This feature provides a detailed view of the analysis results, allowing users to detect any missing or erroneous data points, particularly in channels with low or high abundance. By examining these individual protein profiles, users can evaluate the quality of the wet lab work and identify any possible errors or inconsistencies. The statistics module within the application provides users with a user-friendly interface for hypothesis testing, even without prior expertise in statistics. By leveraging the powerful R-package limma (Ritchie et al. 2015), the module offers robust statistical analysis methods, empowering users to perform tests and obtain results to validate their hypotheses.One key feature of the statistics module is its ability to handle multiple factors and therefore correct for batch effects. Users can extend the statistical analysis across various factors, allowing them to quantify the impact of different variables and identify differentially expressed proteins. To enhance the user experience, the visualization of statistical results has been improved, resulting in visually appealing and interactive displays. These visualizations enable users to explore and interpret the results effortlessly, aiding in the comprehension of the statistical findings. By providing intuitive and engaging visual representations, the module facilitates the understanding and communication of complex statistical analyses.

The enrichment analysis module within the application addresses the challenge of selecting protein sets for mapping by incorporating a database-fetching functionality. This feature streamlines the enrichment analysis process and eliminates the need for users to manually select sets to map proteins against. The first variant of the enrichment analysis module utilizes the KEGG database, which offers a comprehensive collection of pathways and their interactions. By accessing this database, users can gain valuable insights into the underlying pathways and their relationships. This automated mapping against the KEGG database provides a convenient and efficient way to perform enrichment analysis. The second variant of the module caters to sophisticated users who may have specific protein sets in mind based on their research question. This flexibility allows users to select and incorporate their own protein sets into the analysis, enabling research in multiple contexts and providing a comprehensive insight into the dataset. Both variants of the enrichment analysis module present the results in interactive plots, offering users a dynamic and immersive experience with the

data. Users can explore the enriched pathways and interact with the visualizations, gaining a deeper understanding of the relationships and connections within the dataset. Additionally, if desired, users can generate static plots and download them for use in presentations or other offline contexts.

One future research question could utilize artificial intelligence by predicting cellular behaviour, cell types, or even diseases. Running the application on a server or high-performance cluster would be advantageous to address this question and employ simple machine learning tools such as decision trees, random forests, or support vector machines. The application of artificial intelligence on biological data could reduce the hands-on time by identifying unclear results caused by experimental design, unclean preparation, or incorrect computation. Another aspect, not limited to research topics, would be to focus on enhancing usability. While a development time of 6 months for a pipeline and user interface is sufficient to meet the analytical needs, there is still room for improvement in terms of user-friendliness. It is crucial to make the application more intuitive and robust to prevent crashes or errors caused by user mistakes. Ensuring a user-friendly experience is particularly important for commercial software, as robustness and ease of use are essential factors for success. However, since all software should be open source and developers should have the freedom to satisfy their curiosity, we can overlook this fact.

5   Conclusion

Through this project, we have demonstrated that computational analysis can be made accessible to a wide range of biologists by providing an interactive environment. It is important to recognize that there cannot be a one-size-fits-all approach for proteomics datasets. Therefore, our application offers easy-to-interpret visualizations that allow users to assess the patterns and trends in their observed datasets. The default options provided in the application are based on successful analysis pipelines such as ScoPE2 (Specht et al. 2021), serving as a starting point for customization at multiple levels.

The developed pipeline and user interface serve as valuable tools for individual researchers and research communities, enabling them to analyze and visualize their data in meetings, even on non-performant computers. However, our future perspective involves deploying the application on a server, making it accessible to research groups around the world. This would enhance collaboration and facilitate the sharing of insights and findings across different geographical locations. By providing a centralized platform for proteomics analysis, we aim to foster scientific advancements and enable researchers to make significant contributions to their respective fields. Furthermore by having a central hub for proteomic data analysis and the use of artificial intelligence the biological meaning could be brought to a wider context. Since proteomic data could be saved without violating data protection laws, algorithms could be employed to learn from large datasets.

6 Bibliography

Aebersold, R. & Mann, M. (2003), 'Mass spectrometry-based proteomics', *Nature* **422**, 198–207.

Bacharach, M. (1965), 'Estimating nonnegative matrices from marginal data', *International Economic Review* **6**, 294.

Beltran, P. M. J., Federspiel, J. D., Sheng, X. & Cristea, I. M. (2017), 'Proteomics and integrative omic approaches for understanding host–pathogen interactions and infectious diseases', *Molecular Systems Biology* **13**, 922.

Benjamini, Y. & Hochberg, Y. (1995), 'Controlling the false discovery rate: A practical and powerful approach to multiple testing', *Journal of the Royal Statistical Society: Series B (Methodological)* **57**, 289–300.

Benjamini, Y. & Yekutieli, D. (2001), 'The control of the false discovery rate in multiple testing under dependency', *The Annals of Statistics* **29**.

Brownridge, P. & Beynon, R. J. (2011), 'The importance of the digest: Proteolysis and absolute quantification in proteomics', *Methods* **54**, 351–360.

Budnik, B., Levy, E., Harmange, G. & Slavov, N. (2018), 'Scope-ms: mass spectrometry of single mammalian cells quantifies proteome heterogeneity during cell differentiation', *Genome Biology* **19**, 161.

Cobb, M. (2017), '60 years ago, francis crick changed the logic of biology', *PLOS Biology* **15**, e2003243.

Cox, J. & Mann, M. (2008), 'Maxquant enables high peptide identification rates, individualized p.p.b.-range mass accuracies and proteome-wide protein quantification', *Nature Biotechnology* **26**, 1367–1372.

Cox, J., Neuhauser, N., Michalski, A., Scheltema, R. A., Olsen, J. V. & Mann, M. (2011), 'Andromeda: A peptide search engine integrated into the maxquant environment', *Journal of Proteome Research* **10**, 1794–1805.

Gygi, S. P., Rochon, Y., Franza, B. R. & Aebersold, R. (1999), 'Correlation between protein and mrna abundance in yeast', *Molecular and Cellular Biology* **19**, 1720–1730.

Holm, S. (1979), 'A simple sequentially rejective multiple test procedure', *Scandinavian Journal of Statistics* **6**, 65–70.

Houtven, J. V., Hooyberghs, J., Laukens, K. & Valkenborg, D. (2021), 'Constand: An efficient normalization method for relative quantification in small- and large-scale omics experiments in r bioconductor and python', *Journal of Proteome Research* **20**, 2151–2156.

Iribarne, J. V. (1976), 'On the evaporation of small ions from charged droplets', *The Journal of Chemical Physics* **64**, 2287.

Johnson, W. E., Li, C. & Rabinovic, A. (2007), 'Adjusting batch effects in microarray expression data using empirical bayes methods', *Biostatistics* **8**, 118–127.

Kwon, Y. W., Jo, H.-S., Bae, S., Seo, Y., Song, P., Song, M. & Yoon, J. H. (2021), 'Application of proteomics in cancer: Recent trends and approaches for biomarkers discovery', *Frontiers in Medicine* **8**.

Lan, L., Djuric, N., Guo, Y. & Vucetic, S. (2013), 'Ms-k nn: protein function prediction by integrating multiple data sources', *BMC Bioinformatics* **14**, S8.

Leek, J. T., Johnson, W. E., Parker, H. S., Jaffe, A. E. & Storey, J. D. (2012), 'The <tt>sva</tt> package for removing batch effects and other unwanted variation in high-throughput experiments', *Bioinformatics* **28**, 882–883.

Liou, Y.-R., Wang, Y.-H., Lee, C.-Y. & Li, P.-C. (2015), 'Buoyancy-activated cell sorting using targeted biotinylated albumin microbubbles', *PLOS ONE* **10**, e0125036.

Maes, E., Cools, N., Willems, H. & Baggerman, G. (2020), 'Facs-based proteomics enables profiling of proteins in rare cell populations', *International Journal of Molecular Sciences* **21**, 6557.

Maes, E., Hadiwikarta, W. W., Mertens, I., Baggerman, G., Hooyberghs, J. & Valkenborg, D. (2016), 'Constand : A normalization method for isobaric labeled spectra by constrained optimization', *Molecular & Cellular Proteomics* **15**, 2779–2790.

Magnusson, R., Rundquist, O., Kim, M. J., Hellberg, S., Na, C. H., Benson, M., Gomez-Cabrero, D., Kockum, I., Tegnér, J. N., Piehl, F., Jagodic, M., Mellergård, J., Altafini, C., Ernerudh, J., Jenmalm, M. C., Nestor, C. E., Kim, M.-S. & Gustafsson, M. (2022), 'Rna-sequencing and mass-spectrometry proteomic time-series analysis of t-cell differentiation identified multiple splice variants models that predicted validated protein biomarkers in inflammatory diseases', *Frontiers in Molecular Biosciences* **9**.

Marx, V. (2019), 'A dream of single-cell proteomics', *Nature Methods* **16**, 809–812.

McCarthy, D. J., Campbell, K. R., Lun, A. T. L. & Wills, Q. F. (2017), 'Scater: pre-processing, quality control, normalization and visualization of single-cell rna-seq data in r', *Bioinformatics* **33**, 1179–1186.

McInnes, L., Healy, J. & Melville, J. (2018), 'Umap: Uniform manifold approximation and projection for dimension reduction'.

Minakshi, P., Kumar, R., Ghosh, M., Saini, H. M., Ranjan, K., Brar, B. & Prasad, G. (2019), *Single-Cell Proteomics: Technology and Applications*, Elsevier.

Nygaard, V., Rødland, E. A. & Hovig, E. (2016), 'Methods that remove batch effects while retaining group differences may lead to exaggerated confidence in downstream analyses', *Biostatistics* **17**, 29–39.

Parker, C. E., Warren, M. R. & Mocanu, V. (2010), *Mass Spectrometry for Proteomics.*

Pearson, K. (1900), 'X. <i>on the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling</i>', *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* **50**, 157–175.

Phipson, B., Lee, S., Majewski, I. J., Alexander, W. S. & Smyth, G. K. (2016), 'Robust hyperparameter estimation protects against hypervariable genes and improves power to detect differential expression', *The Annals of Applied Statistics* **10**.

Ritchie, M. E., Phipson, B., Wu, D., Hu, Y., Law, C. W., Shi, W. & Smyth, G. K. (2015), 'limma powers differential expression analyses for rna-sequencing and microarray studies', *Nucleic Acids Research* **43**, e47–e47.

Sakia, R. M. (1992), 'The box-cox transformation technique: A review', *The Statistician* **41**, 169.

Schoof, E. M., Furtwängler, B., Üresin, N., Rapin, N., Savickas, S., Gentil, C., Lechman, E., auf dem Keller, U., Dick, J. E. & Porse, B. T. (2021), 'Quantitative single-cell proteomics as a tool to characterize cellular hierarchies', *Nature Communications* **12**, 3341.

Senko, M. W., Beu, S. C. & McLaffertycor, F. W. (1995), 'Determination of monoisotopic masses and ion populations for large biomolecules from resolved isotopic distributions', *Journal of the American Society for Mass Spectrometry* **6**, 229–233.

Smyth, G. K. (2004), 'Linear models and empirical bayes methods for assessing differential expression in microarray experiments', *Statistical Applications in Genetics and Molecular Biology* **3**, 1–25.

Specht, H., Emmott, E., Petelski, A. A., Huffman, R. G., Perlman, D. H., Serra, M., Kharchenko, P., Koller, A. & Slavov, N. (2021), 'Single-cell proteomic and transcriptomic analysis of macrophage heterogeneity using scope2', *Genome Biology* **22**, 50.

Sprent, P. (2011), *Fisher Exact Test*, Springer Berlin Heidelberg.

Sutermaster, B. A. & Darling, E. M. (2019), 'Considerations for high-yield, high-throughput cell enrichment: fluorescence versus magnetic sorting', *Scientific Reports* **9**, 227.

Tannous, A., Boonen, M., Zheng, H., Zhao, C., Germain, C. J., Moore, D. F., Sleat, D. E., Jadot, M. & Lobel, P. (2020), 'Comparative analysis of quantitative mass spectrometric methods for subcellular proteomics', *Journal of Proteome Research* **19**, 1718–1730.

Vanderaa, C. & Gatto, L. (2021), 'Replication of single-cell proteomics data reveals important computational challenges', *Expert Review of Proteomics* **18**, 835–843.

Väremo, L., Nielsen, J. & Nookaew, I. (2013), 'Enriching the gene set analysis of genome-wide data by incorporating directionality of gene expression and combining statistical hypotheses and methods', *Nucleic Acids Research* **41**, 4378–4391.

Wang, Z., Gerstein, M. & Snyder, M. (2009), 'Rna-seq: a revolutionary tool for transcriptomics', *Nature Reviews Genetics* **10**, 57–63.

Wilm, M. (2011), 'Principles of electrospray ionization', *Molecular & Cellular Proteomics* **10**, M111.009407.

Wu, T., Hu, E., Xu, S., Chen, M., Guo, P., Dai, Z., Feng, T., Zhou, L., Tang, W., Zhan, L., Fu, X., Liu, S., Bo, X. & Yu, G. (2021), 'clusterprofiler 4.0: A universal enrichment tool for interpreting omics data', *The Innovation* **2**, 100141.

Ye, Z., Batth, T. S., Rüther, P. & Olsen, J. V. (2022), 'A deeper look at carrier proteome effects for single-cell proteomics', *Communications Biology* **5**, 150.

Ünige A. Laskay, Lobas, A. A., Srzentić, K., Gorshkov, M. V. & Tsybin, Y. O. (2013), 'Proteome digestion specificity analysis for rational design of extended bottom-up and middle-down proteomics experiments', *Journal of Proteome Research* **12**, 5558–5569.

7 Appendix

7.1 Cells sorting protocol

1. 96 single cells from each donor were sorted into a single plate. Each well contained a single cell. As such, 12 different plates containing single cells from 12 different donors were prepared.

2. Carrier plates were also prepared. In the carrier plates, 100 cells from one donor were sorted into each well. 24 such wells (3 columns) were prepared for each donor.

7.1.1 Digestion protocol

1. Prepare 25 ng/ μL trypsin in 100 mmol/L Triethylammonium bicarbonate buffer (TEAB).

2. Add 25 ng/μL trypsin: 2 μL to the carrier samples, 1 μL to the single cell samples.

3. Seal the plates.

4. Centrifuge (2 min at 120 G).

5. Incubate overnight at 37 °C.

7.1.2 Labeling protocol

1. Let the TMTpro labels come to room temperature so that all the condensation on the tube is gone. Dissolve the TMTpro reagents in anhydrous acetonitril (ACN) to obtain a concentration of 10 μg/μL.

2. Add 1 μL of TMTpro reagents according to the tables. For one set of plates, you will need 16 μL of each label. It is recommended to aliquot a slightly larger volume, around 20 μL.

3. Seal the plates.

4. Centrifuge for 2 minutes at 120 G.

5. Incubate for 1 hour at room temperature.

6. Prepare a 5

7. Add 1 μL of the 5

8. Seal the plates.

9. Centrifuge for 2 minutes at 120 G.

10. Incubate for 1 hour at room temperature.

11. Cover the plates with cap strips.

7.1.3   Pooling protocol

1. Wash a 10 μL glass microsyringe five times with ACN.

2. Use the microsyringe to combine TMTpro-labeled samples into 36 LC autosampler vials with built-in glass inserts.

3. Combine six carrier wells from each sample and add the carrier proteome to the 36 TMTpro sets.

4. Combine the first set by taking well A1 from the first single-cell plate, well A2 from the second plate, well A3 from the third plate, and so on. For the next set, take well A1 from the second plate, well A2 from the third plate, and so on. Combine 12 sets following this scheme. Combine the next 24 sets by following the same scheme for rows B and C.

5. Vacuum centrifuge the combined sets to dryness.

6. Dissolve the dried samples in 7 μL of buffer A and transfer them to LC-MS/MS for further analysis.

7.1.4   R-version info

```
R version 4.3.0 (2023-04-21)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 22.04.2 LTS


 [1] visNetwork_2.1.2          snowfall_1.84-6.2
 [3] snow_0.4-4                 piano_2.16.0
 [5] org.Hs.eg.db_3.17.0       AnnotationDbi_1.62.1
 [7] infotheo_1.2.0.1          clusterProfiler_4.8.1
 [9] plotly_4.10.1             sva_3.48.0
[11] BiocParallel_1.34.2       genefilter_1.82.1
```

```
[13] mgcv_1.8-42              nlme_3.1-162
[15] impute_1.74.1           CONSTANd_1.8.0
[17] limma_3.56.1            scater_1.28.0
[19] scuttle_1.10.1          reshape2_1.4.4
[21] dplyr_1.1.2             magrittr_2.0.3
[23] ggplot2_3.4.2           SingleCellExperiment_1.22.0
[25] scp_1.10.0              QFeatures_1.10.0
[27] MultiAssayExperiment_1.26.0 SummarizedExperiment_1.30.1
[29] Biobase_2.60.0          GenomicRanges_1.52.0
[31] GenomeInfoDb_1.36.0     IRanges_2.34.0
[33] S4Vectors_0.38.1        BiocGenerics_0.46.0
[35] MatrixGenerics_1.12.0   matrixStats_0.63.0
[37] spsComps_0.3.2.1        DT_0.28
[39] shinydashboard_0.7.2    shinyWidgets_0.7.6
[41] shiny_1.7.4
```