Charles University in Prague
Faculty of Science

BACHELOR THESIS

Tomáš Jelínek

Linking Proteomes to Phenotypic Traits

Department of Zoology

Supervisor of the bachelor thesis:	Prof. Mgr. Pavel Stopka, Ph.D.
Study programme:	Bioinformatika
Study branch:	B-BINF

Prague 2024

Prohlašuji, že jsem tuto bakalářskou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů. Tato práce nebyla využita k získání jiného nebo stejného titulu.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V . . . . dne . . . .
Podpis autora

I would like to thank my supervisor, Prof. Mgr. Pavel Stopka, Ph.D., alongside his wife and my consultant, Mgr. Romana Stopková, Ph.D., for their guidance and support. I would also like to thank my family, who has supported me in my studies, and my friends for their help. Also, my thanks go to Metacentrum [1] for providing computational resources.

[1] Computational resources were provided by the e-INFRA CZ project (ID:90254), supported by the Ministry of Education, Youth and Sports of the Czech Republic.

Title: Linking Proteomes to Phenotypic Traits

Author: Tomáš Jelínek

Supervisor: Prof. Mgr. Pavel Stopka, Ph.D., Department of Zoology

Abstract: Proteins, as the main functional molecules of the cell, play a critical role in shaping phenotypic traits. This thesis investigates the application of proteomics data, obtained via LC-MS/MS, to understand the link between protein expression and resulting phenotypes. Various bioinformatic approaches are discussed, including data preprocessing, normalization techniques, and missing value imputation methods, to ensure the reliability and accuracy of downstream analyses. Furthermore, this work describes how one can gain insights into phenotypic traits across species or medical conditions by employing differential expression analysis, evolutionary modeling using the Ornstein-Uhlenbeck process, and machine learning algorithms.

Keywords: proteomics, phenotype, evolution, bioinformatics

Název práce: Propojování proteomických dat s fenotypy

Autor: Tomáš Jelínek

Vedoucí bakalářské práce: Prof. Mgr. Pavel Stopka, Ph.D., Katedra zoologie

Abstrakt: Proteiny, jakožto hlavní aktéři v buněčných procesech, mají zásadní roli ve formování fenotypových znaků. Tato práce zkoumá aplikaci proteomických dat, získaných pomocí LC-MS/MS, pro pochopení vztahů mezi expresí proteinů a výslednými fenotypy. Diskutuje různé bioinformatické postupy, od předzpracování dat po techniky normalizace a metod imputace chybějících hodnot, pro zajištění co nejvyšší kvality dat pro následující analýzy. Dále tato práce popisuje, jak lze získat vhled do oněch fenotypových znaků u různých druhů zvířat či patologických stavů za užití analýzy diferenciální exprese, evolučního modelování pomocí Ornstein-Uhlenbeckova procesu nebo algoritmů strojového učení.

Klíčová slova: proteomika, fenotyp, evoluce, bioinformatika

Contents

Introduction

In an organism, all cells share the same genetic information; nevertheless, they can fundamentally vary. This variation is present because different genes are expressed in cells at different times and amounts. Differences in gene expression are believed to be a major contributor to the phenotypic diversity observed between species. These differences, along with DNA polymorphisms in coding sequences, have profound effects on how organisms develop and function [1, 2].

The study of gene expression is not straightforward and is a complex process. This relates not only to data acquisition but also to data processing and analysis. In this thesis, we focus on proteomics, which is the study of the entire set of proteins expressed by a genome, cell, tissue, or organism at a particular time. Proteins are key players in almost all biological processes. Abnormalities in protein expression can lead to diseases as it is one of the most prominent factors in pathologies, and understanding the proteome can help to develop new treatments and diagnostics [3]. The aim is to describe how to utilize the proteomics data to better understand gene expression itself, how it relates to the evolution of species of interest and most importantly, uncover its link to phenotypic traits.

Most techniques to achieve this have been developed for the RNA-seq data. However, fundamental differences exist between the acquisition, processing, and even meaning of RNA-seq and proteomics data. Also, several challenges are solely related to proteomics data; therefore, applying these methods to proteomics data is not straightforward [3]. Understanding each step in the data processing workflow is therefore crucial for interpreting the results correctly, avoiding common pitfalls, and understanding the limitations of the data.

In this thesis, we propose a proteomic processing pipeline to achieve this goal. The first chapter describes proteomic data acquisition techniques using LC-MS/MS experiments, how to match the obtained spectra to peptides, and how to quantify the proteins. The second chapter focuses on data preprocessing, including quality control, describing which normalization method to choose and which missing value imputation algorithm to use so the data can be used for further analysis. The last chapter describes how to analyze the data to obtain the most information regarding the phenotypes. This includes differential expression analysis, machine learning, and evolution modeling using Brownian motion and Ornstein-Uhlenbeck processes.

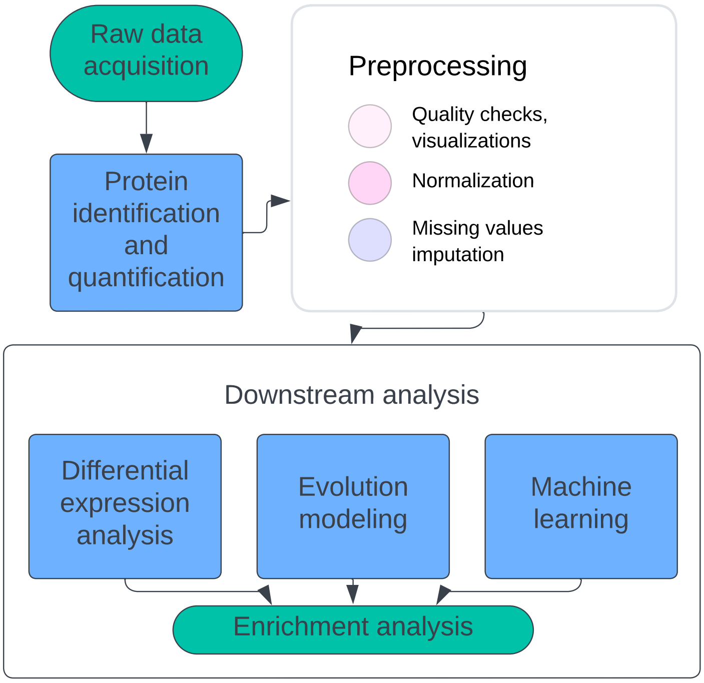
Raw data acquisition
Protein identification and quantification
Preprocessing
Quality checks, visualizations
Normalization
Missing values imputation
Downstream analysis
Differential expression analysis
Evolution modeling
Machine learning
Enrichment analysis

Figure 1 Proposed proteomic processing pipeline.

Throughout this thesis, proteomic data analysis from sperm cells across 34 passerine species acquired via LC-MS/MS LFQ will be used to exemplify the discussed methods.

Chapter 1

LC-MS/MS Proteomics, Protein Identification and Quantification

The first proteomic studies date back to 1975 when O'Farrell introduced two-dimensional gel electrophoresis (2-DE). Since then, tremendous progress has been made, primarily due to the development of mass spectrometry (MS) and liquid chromatography (LC) techniques, which have increased the sensitivity and accuracy of protein identification and quantification by several orders of magnitude [4]. LC-MS/MS is currently the go-to method for proteomic studies [4, 3, 5, 6].

There are several types of proteomics:

- **Structural proteomics** focuses on the 3D structure of proteins and their complexes.
- **Functional proteomics** studies the function of proteins and their interactions.
- **Expression proteomics** studies the expression of proteins in a given cell or tissue and compares it to other cells or tissues.

Here, the emphasis is on the usage of expression proteomics. Nevertheless, results obtained from the expression proteomics can be coupled with results from the other types of proteomics to give a more comprehensive understanding of the biological system under study.

1.1 Mass Spectrometry

To begin with, it is essential to clarify that the method described here is known as "bottom-up proteomics," which involves the digestion of proteins into peptides before analysis. This contrasts with "top-down proteomics," where intact proteins are analyzed directly. While top-down approaches are also valuable, they are not as commonly used in this context as bottom-up approaches and thus will not be discussed further here [3].

First, proteins must be extracted from biological samples and then digested into peptides, usually by trypsin. Then, peptides are purified from salts, detergents, and other contaminants. Nevertheless, the sample preparation for each experiment can differ depending on the biological system under study and the proteins of interest. Extra care is needed, for example, when dealing with membrane proteins or serums, as described by Chandramouli and Qian [7].

Peptides are then separated by liquid chromatography, and this is then followed by tandem mass spectrometry (LC-MS/MS), where resolved peptides are ionized and analyzed in two stages by the mass spectrometer. In the first stage (MS1), mass-to-charge (m/z) ratios of the peptides are measured, and in the second stage (MS2), the peptides are further fragmented, and their subsequent m/z ratios are measured [3, 7, 6].

1.2 Protein Identification

1.2.1 Peptide identification

After the raw data (fragmentation spectra) are collected, peptide sequences must be identified. Two main approaches are used for this purpose:

- **Searching against the fragmentation spectra database**, where peptide spectrum match (PSM) score is calculated for each peptide against all theoretical spectra from the database. The peptide with the best score can be considered a candidate for the peptide sequence [3].

- **De novo sequencing**, where the peptide sequence is reconstructed solely from the fragmentation spectra without a database. This is usually done by Graphical Probabilistic Models or Hidden Markov Models [3].

1.2.2 Protein inference

Once the peptide sequences are identified, they are mapped to the protein sequences. This task is not straightforward, as the same peptide can be present in multiple proteins. To handle these so-called degenerate peptides, many different models were built to assign these peptides to correct proteins.

Several software tools can be used for peptide identification and subsequent protein inference. Chen et al. [3] comprehensively reviews the most popular ones and discusses their advantages and disadvantages. In our study of passerine sperm cells, MaxQuant[8] was used for protein identification along with its integrated search engine Andromeda[9].

1.3 Protein Abundance Quantification

Most experimental quantification methods fall into two following categories:

Labeled methods These methods utilize isotopic or chemical labels to differentially mark proteins or peptides from various samples, enabling their quantification through mass spectrometry. Labeled approaches are further subdivided based on the stage of mass spectrometry where quantification occurs:

- In **MS1-based labeling**, peptides from different conditions are chemically tagged before the LC-MS/MS analysis, so there are shifts in their m/z ratios. These labeled peptides are then co-analyzed in a single LC-MS/MS run, and the quantification is done in the MS1 stage by comparing the isotopic variants in the MS1 spectra, with each variant representing a different condition [3]. Techniques such as ICAT, SILAC, and ICPL are among this category's most widely used ones [3].
- In **MS2-based labeling**, the quantification occurs in the MS2 stage. Here, peptides are labeled with tags that not only change their m/z ratios but also release reporter ions upon fragmentation. This makes it possible to quantify relative protein abundances in a single LC-MS/MS run from all conditions - usually, up to 11 samples can be in one experiment when using TMT [3, 10].

Label-free methods In label-free quantification (LFQ), spectra for different samples are obtained from separate LC-MS/MS runs, contrary to labeled methods, and do not use any labels. In data-dependent acquisition (DDA), the most abundant precursor ions are selected for fragmentation. On the contrary, Data-independent acquisition (DIA) selects a window of m/z ratios for fragmentation at each chromatographic time point, which can increase the number of quantified peptides selected without bias. The downside of DIA is that the resulting fragmentation spectra are more complex and harder to interpret than those from DDA; therefore, DDA is more commonly used [3, 11]. In DDA LFQ methods, the quantification can be divided into two main categories:

- **Spectral counting** estimates the protein abundance from the number of MS/MS spectra of peptides mapped to the given protein [11].
- **Peak intensity-based approach** quantifies the proteins according to the ion intensities of the detected peptides [11].

Zhao et al. [11] comprehensively evaluated the performance of 7 commonly used LFQ methods and found that MaxQuant in MaxLFQ mode had the best accuracy and precision but had high missing values. Meanwhile, using SEQUEST as a search engine, the Proteome Discoverer performed superiorly in quantifiable low-abundance proteome coverage [11].

However, LFQ bears an inherent problem - samples are not directly comparable due to many unwanted variations and biases, so normalization is needed [3, 12, 13, 14]. So-called spike-in standards can help with this problem of absolute protein quantification in LFQ experiments. Nevertheless, this method increases the costs and complexity of the process [3].

When it comes to choosing between labeled and label-free methods, several factors need to be considered:

- **Number of samples:** Labeled methods are typically limited in the number of samples that can be analyzed simultaneously, while LFQ offers greater flexibility in this regard.
- **Accuracy and precision:** Labeled methods generally provide higher accuracy and precision, especially for low-abundance proteins.
- **Cost and time:** Labeled methods require specialized reagents and expertise, making them more expensive and time-consuming than LFQ.
- **Research question:** The specific research question and the desired level of quantitative accuracy will influence the choice of method.

Labeling methods are the way to go when one has money, time, and few samples or when high accuracy is needed. On the other hand, when one has many samples, LFQ is the better choice, and it is also more cost-effective and less time-consuming. Another advantage of LFQ is that it requires less specialized manipulation and treatment of samples, which lowers the risks of altering the proteins in any manner in the labeling process [15].

Chapter 2

Proteomic Data Preprocessing

Once the proteins and their abundances are known, it is usually desired to apply certain cutoffs for minimum peptide numbers matched to specific proteins to increase the reliability of the data [3]. Often, software tools mentioned earlier responsible for identifying and quantifying proteins allow setting these cutoffs and providing values for the number of peptides matched to each protein, Q values measuring the false discovery rate (FDR) for each protein, and other valuable statistics. However, data quality checks need not only rely on these tools, and other software such as MsQuality developed by Naake, Rainer, and Huber [16] can be used to assess the quality of the raw data.

It might also be desirable to inspect the data visually and perform some exploratory data analysis (EDA) to understand the data better. In our case of passerine sperm, we noticed that the proteins with the highest intensities across species were HBAA and LOC100222646, which are blood proteins and should not be in our sperm samples, indicating blood contamination.

2.1 Normalization

Nevertheless, as mentioned earlier, expression data in the current state are not directly comparable, and normalization is needed. This step is crucial as choosing different normalization methods can lead to different conclusions in subsequent analyses [17].

Normalization aims to deal with systematic biases in the data arising from technical variations, such as differences in sample preparation and handling, spectrometer calibration, or even temperature changes. Unfortunately, the exact reasons for these biases are often unknown and, therefore, cannot be accounted for by adjustment of the experimental design [18].

Many of the normalization methods used in proteomics come from transcriptomics. Most of these methods assume that most proteins are not differentially expressed, but this does not always hold [18, 19].

Some of the most popular normalization methods are described below.

Quantile Normalization Quantile normalization aligns the distribution of protein expression levels across multiple samples. For n samples, this technique adjusts the expression so that the distribution of quantile values is consistent, forming a line in n-dimensional space along the unit vector $\frac{1}{\sqrt{n}}(1,\ldots,1)$. The k-th quantiles for all samples $\mathbf{q}_k = (q_{k1},\ldots,q_{kn})$ are projected onto the unit diagonal $\mathbf{d} = \frac{1}{\sqrt{n}}(1,\ldots,1)$ as follows:

$$\operatorname{proj}_{\mathbf{d}}\mathbf{q}_k = \left(\frac{1}{n}\sum_{j=1}^{n} q_{kj},\ldots,\frac{1}{n}\sum_{j=1}^{n} q_{kj}\right)$$

as described by Bolstad et al. [20].

Median Normalization This method assumes that the samples are proportionally related and share equivalent medians, adjusting the data by a scaling factor accordingly [18].

Cyclic Loess Normalization In Cyclic Loess normalization, each pair of samples is MA-transformed. Log-ratios M are plotted against mean log intensities A and then normalized iteratively. Specifically, Cyclic Loess cyclically performs this transformation, iterating through all possible sample pairs and repeating the cycle three times to ensure stability [18].

Variance Stabilizing Normalization (VSN) Originally developed by Huber et al. [21] for microarray data, VSN is based on the assumption that the variance of the expression levels is proportional to the mean. This method aims to make the variances nondependent of their mean and bring it to the same scale for all samples [18]. It assumes that this transformation can be achieved through affine-linear mappings. For which the parameters are calculated using maximum likelihood estimation [21].

The methods mentioned here are manipulating data after log2-transformation (except VSN) as other methods in literature usually do. R packages such as *proteiNorm*[22] or *NormalyzerDE*[23] provide user-friendly interfaces to normalize proteomic data using several methods of choice. Nevertheless, choosing the right normalization method is not an easy task. *NormalyzerDE* and *proteiNorm* provide visualizations comparing the results of different normalization methods, which can be used in decision-making processes to determine which method to use. Providing Pooled intragroup Coefficient of Variation (PCV), Pooled intragroup Median Absolute Deviation (PMAD), Pearson and Spearman correlation, MA-plots, and several other metrics. Descriptions on how to interpret them can be found in [23].

However, the normalization method should not be chosen only based on these metrics, as they can be slightly deceiving. For example, it is to be expected that Cyclic Loess will have the nicest MA plots, as this is the metric against which the algorithm optimizes. Simultaneously, having a high intragroup correlation is meaningless if the intergroup correlation is also very high. For these reasons, Valikangas, Suomi, and Elo [18] compared the performance of popular normalization methods on spike-in datasets. VSN normalization consistently outperformed other methods in terms of AUC when finding differentially expressed proteins; also, it has decreased PMAD significantly more than other methods and had the highest Pearson correlation coefficient between technical replicates. Chawade, Alexandersson, and Levander [23] also considered VSN one of the most suitable methods for the normalization of proteomic data. However, VSN consistently underestimated the logFCs of the spike-in proteins in benchmarks by Valikangas, Suomi, and Elo [18], which can be seen as a potential downside, particularly when examining the exact logFCs of proteins as they note.

For our data, we have chosen the VSN normalization, as it had good performance based on the report generated by *NormalyzerDE* and also because it had great performance in the benchmarks mentioned earlier.

2.2 Missing Values

Unlike RNA-seq, proteomics is significantly challenged by missing values (MVs), detrimentally affecting the outcome of downstream analyses and even rendering certain methods inapplicable due to their inability to handle MVs. This issue is primarily attributed to protein abundances falling below the detection threshold and various technical constraints of mass spectrometry; this includes sample loss during preparation, peptide miscleavage, and poor ionization efficiency [24]. Furthermore, MVs in proteomics can stem from coverage missingness, which occurs when a protein is not observed in any sample despite its known presence, in addition to inconsistency missingness, where a protein is observed in at least one instance but not others [25].

MAR and MNAR MVs can be broadly classified into missing at random (MAR) and missing not at random (MNAR). MAR MVs often arise from technical limitations and stochastic fluctuations independent of protein abundance [24]. Conversely, MNAR MVs are typically abundance-dependent, attributed to the non-measurability of specific peptides [24]. The distinction between MAR and MNAR is crucial for understanding the underlying reasons for data incompleteness and informs the selection of appropriate mitigation strategies.

Addressing Coverage Missingness Coverage missingness presents a great challenge in proteomics. Unlike inconsistency missingness, which can be somewhat mitigated through missing value imputation (MVI) algorithms, coverage missingness requires more comprehensive approaches. These may include data integration techniques, leveraging observed proteins alongside reference networks and prior knowledge to infer the presence of unobserved proteins [25]. However, in many cases, compensating for coverage missingness is unnecessary, as the focus is on the proteins observed, such as in our case of passerine sperm cells.

As dealing with inconsistency missingness is a more common problem [25], we will focus on this type of MVs in the following section and how to handle them.

2.2.1 Missing Values Imputation

Many different imputation methods have been developed to address MVs in proteomics, each with its strengths and weaknesses. The choice of imputation method is crucial, as it can significantly impact the results of downstream analyses [24, 25]. Nevertheless, it is nontrivial to determine which method will yield the most accurate results. Examples of imputation methods are provided below. See Jin et al. [24] and Kong et al. [25] for more comprehensive reviews.

Naïve Imputation

Naïve imputation is a straightforward approach where missing values are filled in with simple guesses like zero, mean or median values. While easy to do, using a constant value can sometimes make it hard to see differences between samples, which might not be great for understanding actual protein levels. Some basic methods, like MinProb, work well when data mostly lacks information because it's not strong enough to be detected. This method puts in random values that are very low, fitting for when missing data is because something was too faint to see. Surprisingly, a method called SampMin, which puts the lowest seen values in place of missing ones, has shown to be pretty effective, especially when trying to find out which proteins are more or less abundant than usual. This might be because it puts in values close to what the instruments can just about detect, making it a more accurate guess for missing data due to low intensity [25].

Feature-based and Ensemble Imputation Approaches

Feature-based and ensemble imputation approaches offer more advanced strategies for handling missing values in proteomic data, with methods like K-nearest neighbors (KNN), GSimp, and Random Forests (MissForest) being particularly effective [25], will be here described in more detail. It is important to note that many other methods exist [25, 24].

K-nearest Neighbors (KNN) KNN imputes missing values by identifying the k closest samples in the dataset and averaging their values to fill in gaps. This method considers the similarity between samples based on features like protein levels, making it particularly useful for datasets where similar samples can provide meaningful information for imputation. KNN variants, such as KNN-Euclidean (KNN-EU), KNN-Correlation (KNN-CR), and KNN-Truncation (KNN-TN), adapt the basic approach to more closely fit the specific characteristics of proteomic data, offering flexible solutions to address Missing at Random (MAR) and Missing Not at Random (MNAR) types of missingness [25].

GSimp GSimp enhances the Quantile Regression Imputation of Left-Censored Data (QRILC) by incorporating a two-step refinement process. This approach uses an elastic net model combined with a random Gibbs sampler to provide a more accurate imputation for data assumed to have linear relationships. GSimp is particularly effective for datasets where parameters are estimated under the assumption of normality, but its accuracy can be bad if the data does not adhere to these assumptions [25].

Random Forests (MissForest) R package *MissForest*[26] utilizes the Random Forest algorithm to handle missing data by constructing multiple decision trees and using their aggregated predictions to impute missing values. This method has shown to perform exceptionally well with MAR data [25, 24], outperforming other imputation methods in terms of the ability to recover the original distribution of the dataset. While it may not be as effective for left-censored MNAR data compared to QRILC, MissForest's robustness makes it a preferred choice for proteomic datasets that contain a mix of MAR and MNAR missingness [25].

Kong et al. [25] in their paper provide a decision tree as to which method to use under what circumstances. However, this decision tree is not based on rigorous benchmarking but on assumptions about the methods and data. Jin et al. [24] tested the performance of several popular imputation methods on proteomic data where they introduced different rates of MVs and their types. They found that *MissForest* consistently outperformed other methods with the lowest NRMSE, high TPs, and average FADR < 0.05.

Nevertheless, caution is needed when using Random Forest imputation methods, as when the percentage of MVs is too high (e.g., $> 30\%$), the imputation can be inaccurate and misleading. Regardless of the chosen method, the yield of an experiment will increase as sensitivity is expected to increase in subsequent analyses, but unfortunately, the false discovery rate will increase too, and one has to account for this fact [27].

2.3 EDA and Visualizations

Now that the proteomes are directly comparable, starting with a deeper exploratory analysis (EDA) is possible. First of all, it might be helpful to view correlations between samples as one can detect outliers or other issues in the data. From figure 2.1, it can be seen that several samples such as S7, S19, S52, and several others seem to be exhibiting low overall correlation - they have low intragroup and even outgroup Pearson correlation even though the other replicates seem to be okay. This could have occurred for several reasons, such as poor sample extraction, preparation, or contamination. Keeping data like that could obscure the results by falsely increasing variance.

Both from figure 2.1 and 2.2 outlier species with distant proteome profiles can be seen - e.g., *Pyrrhula pyrrhula*, *Locustella lusciniodes* or *Lanius collurio*. These species might have curious evolution histories or phenotypes forcing proteome to be that way, and it might be interesting to investigate them further.

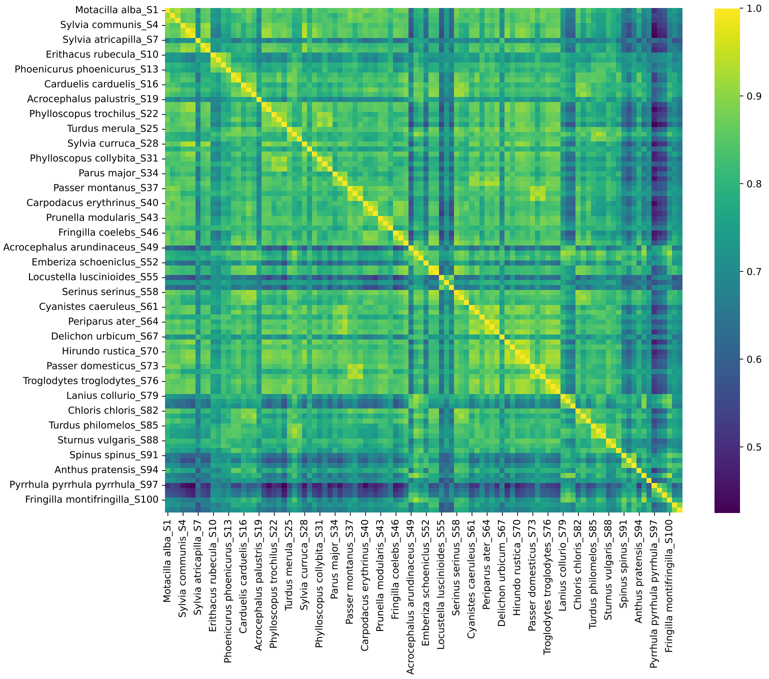

Figure 2.1 Pearson correlation plot of the passerine sperm cells proteome samples.

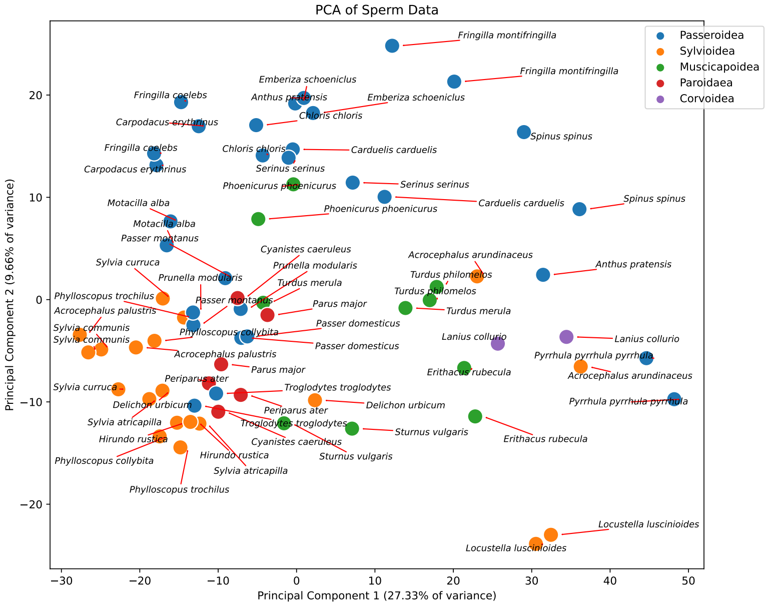

Figure 2.2 Principal component analysis of the passerine sperm cells proteome samples. Only two samples (ones with the highest pairwise Pearson correlation) are shown per species, as the plot would be too cluttered otherwise.

2.3.1 Gene expression phylogenies

Brawand et al. [28] showed that gene expression data (from RNA-seq) can be used to infer phylogenetic relationships between mammalian species. First of all, the gene expression profiles cluster by tissue and then secondly by species. Suggesting that regulatory changes accumulate over time, such that more closely related species have more similar expression levels. This also shows that gene expression has strong tissue and even sex-specific constraints [29]. Pal, Oliver, and Przytycka [29] reached the same conclusion when exploring gene expression in the Drosophila genus. Nevertheless, testis showed the highest variance, and the reconstructed phylogeny did not perfectly match the real phylogeny in the case of the mammals [28].

Both teams used Spearman correlation to measure the similarity between gene expression profiles and then constructed a phylogeny using the neighbor-joining method from the distance matrix. For proteomics normalized and log2-transformed data, we have considered using Euclidean distance as a more appropriate metric. They used Spearman correlation because it is more robust to outliers[28, 29], but in log2-transformed data, this problem diminishes. Furthermore, using Spearman correlation on VSN normalized data would have the same result as on unnormalized data, as VSN, in the end, does a generalized logarithm of affine-linear mappings on the vectors. When x is greater than y, then too $glog_2(ax+b) > glog_2(ay+b)$, implying that the ranks of the vector would not change. Therefore, the Spearman correlation would be the same.

As Witten [30] explains, squared Euclidean distance is equivalent to a log-likelihood ratio statistic under a Gaussian model for the data. Given the model

$$X_{ij} \sim N(\mu_{ij}, \sigma^2), X_{i'j} \sim N(\mu_{i'j}, \sigma^2),$$

we have the log-likelihood ratio statistic for testing $H_0 : \mu_{ij} = \mu'_{ij}$ against $H_a : \mu_{ij} \neq \mu'_{ij}$, it is proportional to

$$\begin{aligned} \sum_{j=1}^{p} \left(X_{ij} - \frac{X_{ij} + X_{i'j}}{2} \right)^2 + \sum_{j=1}^{p} \left(X_{i'j} - \frac{X_{ij} - X_{i'j}}{2} \right)^2 &\propto \sum_{j=1}^{p} (X_{ij} - X_{i'j})^2 \\ &= ||\mathbf{x}_i - \mathbf{x}_{i'}||^2. \end{aligned}$$

This demonstrates that squared Euclidean distance is a natural choice for data following a Gaussian distribution as it corresponds to the log-likelihood ratio test statistic under the null hypothesis. Thus, in the context of normalized log2-transformed proteomics data, Euclidean distance may provide a more sensitive and accurate reflection of the phylogenetic relationships inferred from gene expression levels.

Unfortunately, Pal, Oliver, and Przytycka [29] did not mention how they handled having more replicates for each species, but Brawand et al. [28] constructed all possible phylogenies where the species profile was represented by one of the replicates. After that, a majority-rule consensus tree was constructed from these phylogenies. However, this method is not feasible when the number of species and replicates is high. In our case of passerine sperm cells, this would amount to 3^{34} possible phylogenies, which are too much to calculate. Therefore, using the Monte Carlo approach can be appropriate for such cases or simply averaging the replicates for each species and constructing the phylogeny. Nonetheless, we could not obtain a meaningful majority-rule consensus tree from the Monte Carlo approach as our data had too much variability. Therefore, we have calculated the average tree [31] using the *phytools* package [32] in R. After obtaining the tree, we have compared it to the real phylogeny of the species. This can be done using *phytools* cophylo function.

As constructing phylogenies from genomic data is a tedious and time-consuming job, I have developed an open-source Python package *PhyloBuilder*[1] that can speed up the process of obtaining relevant FASTA files and aligning them for the subsequent tree construction.

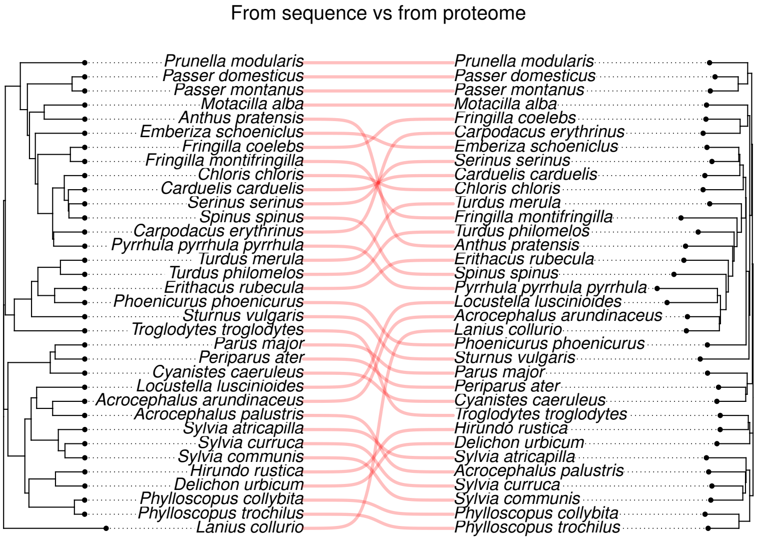

Figure 2.3 Comparison of the real phylogeny (1000 trees were downloaded from `www.birdtree.org` and average tree using *phytools* was computed from them) of the species and the phylogeny constructed from the proteomic data.

[1] `https://github.com/Desperadus/PhyloBuilder`

Chapter 3

Downstream Analysis

After data preprocessing, one can finally start analyzing the data to obtain information regarding the phenotypes. This chapter describes several methods that can be used for this purpose. First, an explanation of enrichment analysis is needed, as this method will be used later to interpret the results from subsequent methods.

3.1 Enrichment Analysis

Enrichment analysis is a method used to identify genes, proteins, or metabolites over-represented in a set of interest compared to a background set. The set of interest is usually a list of genes that have the same function or are involved in the same pathway [3]. This is done because after the differential expression analysis, OU modeling, or machine learning, we have a list of genes that showed up as significant (differentially expressed, under selection, or important for the model predictions), that we want to interpret. As discussed in the seminal work by Subramanian et al. [33], traditional gene expression analysis methods, which might focus on individual genes showing, for example, significant differential expression, face several limitations. These include the modest biological differences being obscured by noise, the uneasy task of interpreting long lists of significant genes without a unifying biological theme, and the potential to overlook crucial effects on pathways where sets of genes act together [33]. The authors argue that cellular processes often impact multiple genes within a pathway, making the collective effect more significant than changes in any single gene. Therefore, looking at the data more holistically is needed, rather than focusing on individual genes alone.

3.1.1 Gene Ontology (GO)

The most widely used method for enrichment analysis is the Gene Ontology (GO) project. GO terms are a set of predefined categories that classify genes and their products into three main domains: Molecular Function, Cellular Component, and Biological Process [3, 34]. Each GO term provides a specific descriptor representing the role of genes and gene products, and the terms are connected to other terms in a directed acyclic graph (DAG) structure, forming a hierarchy [34]. When specific terms are over-represented in a set of interest than we would expect by chance, it suggests that the proteins are together involved in a specific biological process, molecular function, or cellular component. However, as Chen et al. [3] points out, the GO terms usually represent ORF products rather than the mature proteoforms themselves. One must carefully examine the GO terms to ensure they relate to the proteoform of the corresponding gene. If specific protein annotations are missing, homology-based methods can transfer annotations from similar proteins, with several tools available for this purpose [3].

3.1.2 Pathways

Knowledge about regulatory pathways or diseases in which proteins participate can also be utilized for enrichment analysis. Kyoto Encyclopedia of Genes and Genomes (KEGG) is a widely used database representing molecular functions as interaction networks [3, 34]. Beyond KEGG, Reactome is another resource for pathway enrichment analysis. Reactome offers a detailed and curated database of biological pathways across various organisms [3, 34]. Enrichment analysis using pathways can provide a more detailed view of the biological processes and pathways affected by the significant genes.

ClusterProfiler [34] is an R package designed for performing enrichment analysis with GO terms or KEGG pathways. Additionally, it allows using the *enricher* and *GSEA* functions for enrichment analysis with any user-provided gene annotations, including those from databases like Reactome. The package also provides complementary functions that enable the user to compare the results among different conditions or groups and visualize the results as enrichment map networks, GSEA enrichment plots, UpSet plots, and many more [34].

3.2 Differential Expression Analysis

Differential expression analysis is one of the most commonly employed methods in the downstream analysis of gene expression data. It is used to identify whether differences in means of genes between two conditions are significantly higher than one would expect by chance caused by biological or technical variation [35]. For example, suppose one explores a particular type of cancer. In that case, it is possible to compare the gene expression profiles of cancerous and healthy tissues to identify genes that are up- or down-regulated in the cancerous tissue with respect to the healthy tissue. This might provide valuable insights into the molecular mechanisms of the disease and help find genes responsible for the resulting negative phenotype, which drugs or other therapies could later target.

3.2.1 Statistical Methods

First of all, one must estimate the variance of the data. Non-log-transformed data seem to have a mean-dependent variance, Pavelka et al. [35] propose that the gene variances follow the power law relationship $\log(geneS.D.) = k \cdot \log(geneMean) + c + \epsilon$, where k is the slope of the power law, c is the intercept, and ϵ is an error term. Other models propose modeling the variance using a Quasi-Poisson or Negative Binomial distribution [15].

However, relying on the variance estimation from only one sample is not ideal, as it is susceptible to sampling error and will not explain all of the biological variation present in the condition. Therefore, having multiple replicates for each condition is crucial for more robust results. Ooijen et al. [27] showed that having less than three replicates has drastically decreased the predictive power of differential expression analysis of peptides.

The *limma* package in R/Bioconductor is a popular tool for analyzing gene expression data and estimating variances [36]. Limma employs an empirical Bayes approach to moderate the gene-wise variances towards a common or trended variance. This method borrows information across genes, leading to more stable and reliable variance estimates, especially for experiments with small sample sizes. Additionally, limma offers the ability to incorporate quality weights and model correlations between samples, further enhancing the accuracy and robustness of variance estimation and downstream differential expression analysis [36].

Once the variances are estimated, one can further proceed with the differential expression analysis. The most commonly used methods are moderated t-statistics and F-statistics to assess the significance of differential expression for each gene. For simple two-group comparisons, moderated t-tests are employed, comparing the mean expression levels between the groups while taking into account the estimated variances. For more complex experimental designs with multiple factors or groups, moderated F-tests are used to evaluate the overall significance of differential expression across the different conditions [36, 3, 27].

Regardless of the statistical model used, correcting for multiple hypothesis testing is necessary. The most commonly used methods for this purpose are the Benjamini-Hochberg (BH) correction and FDR estimation from permutation [3]. The result can be visualized using volcano plots, MA plots, or heatmaps. Further, the results can be enriched, as mentioned in the previous section about Enrichment Analysis 3.1. This might provide deeper insight into which types of genes are differentially expressed and what the underlying motif is. For example Harel et al. [37] have constructed proteomaps using KEGG pathway annotations and found out that people who have responded to anti-PD-1 and TIL therapy for melanoma had a higher expression of proteins involved in metabolism than those who have not responded. In figure 3.1, one can see a visualization of the results of differential expression between two conditions.

3.2.2 Available Tools for Differential Expression Analysis of Proteomics Data

Many tools have been developed for analyzing expression data from microarrays or RNA-seq, with most of them being directly applicable to proteomics data. However, several tools have been made directly with the proteomics data in mind.

Bai et al. [38] comprehensively evaluated various R packages and other software tools designed for differential expression analysis of LFQ proteomics data. While these packages offer diverse functionalities and approaches, it's crucial to acknowledge that their benchmarking results may not be directly comparable due to differences in normalization and missing value imputation methods. Therefore, the performance of the differential expression analysis is not the only thing being evaluated. They found that different packages resulted in different proteins being identified as differentially expressed; however, this could have also been caused by different normalization and MV imputation methods, as mentioned earlier. They have concluded that *MSstats* is one of the most well-maintained and documented packages with competitive performance, yet *Perseus*, *prolfqua*, and *LFQ-Analyst* had the best performance in their benchmarks [38].

Lin et al. [39] have done benchmarking too on LFQ spike-in datasets, comparing *EdgeR*, *DESeq2*, *limma*, *DEqMS*, *SAM* and *ROTS*. They have found that packages made for RNA-seq data such as *DESeq2*, *EdgeR*, and *ROTS* had the best performance on their datasets. *DEqMS* seems to have better performance than *limma* as it is able to incorporate information from PSM/peptide counts. Their paper and supplementary materials also provide tables of the performances when they used different normalization and MV imputation methods.

Nevertheless, benchmarking a larger number of packages together, with different normalization and MV imputation methods on several datasets, is needed to determine which methodological approach is the best for differential expression analysis on proteomics data.

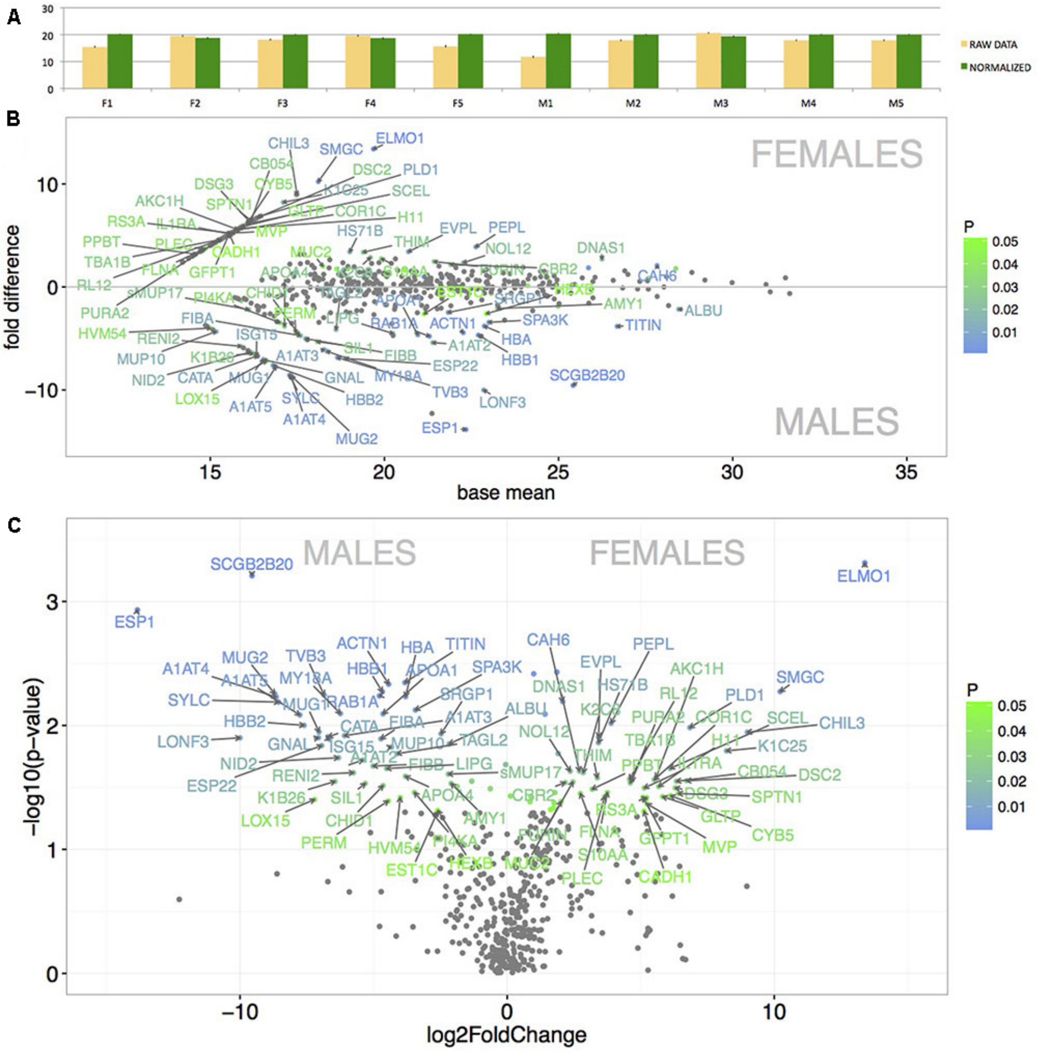

Figure 3.1 Visualization of differential expression analysis of soluble mouse proteome from nasal cavity done by Kuntová, Stopková, and Stopka [40] between male and female mice. In the B plot, one can see the MA plot, where the x-axis represents the average expression level of the gene, and the y-axis represents the fold change. In the C plot, one can see the volcano plot, where the x-axis represents the log-fold change, and the y-axis represents the $-\log_{10}$ of the p-value.

3.3 Evolution Modeling

Selection acting on proteins was and still is mainly studied on the sequence level. Methods like dN/dS ratios might give us insights about the selection acting on the protein sequence/structure, altering its function. However, these methods do not provide any information about the selection acting on the protein abundance, which might be just as important for the phenotype [3, 2, 1]. Therefore, methods that model the evolution of gene expression have been developed to study this phenomenon.

In the past, methods of continuous traits have used inappropriate models that only assumed a purely neutral evolution - Brownian motion (BM). Felsenstein identified two scenarios where the BM model might not accurately represent evolutionary processes: firstly, if selection continues over time, leading to correlated evolutionary changes across successive branches; and secondly, if various lineages experience identical selection pressures [41]. With this problem in mind, Hansen proposed to model the evolution of a continuous trait using the Ornstein-Uhlenbeck (OU) process, which can have multiple evolutionary optima [41]. Butler and King [41] have developed an R package *OUCH* that can be used to model the evolution of quantitative traits (such as gene expression) using the OU process.

3.3.1 Ornstein-Uhlenbeck Process

Consider a quantitative trait X (e.g., gene expression, height, tail length) evolving along one branch of a phylogenetic tree. The following differential equation defines the OU process:

$$dX(t) = \alpha[\theta - X(t)]dt + \sigma dB(t). \tag{3.1}$$

This equation expresses an increment of X in an infinitesimally short time interval. The change of X can be decomposed into two parts: deterministic and stochastic:

- The term $dB(t)$ represents a Wiener process, Butler and King [41] describe it as a *white noise*; that is, identically distributed random variables with mean zero and variance dt. The term σ is the intensity of the random fluctuations. One can imagine this as being the drift of the trait due to randomness.

- The term $\alpha[\theta - X(t)]dt$ is the deterministic part of the process. θ is the local optimal value of the trait in a fitness landscape, and α is the strength of the pull towards the optimal value. This term makes it so that the trait evolves in the *direction* of the optimal value. When the trait is under significant selection pressure, α will be high, leading to rapid evolution towards the optimal value. If there is no selection pressure, α will be zero, and therefore, the deterministic part of the process will completely disappear, and the evolution of the trait will be purely a stochastic BM process.

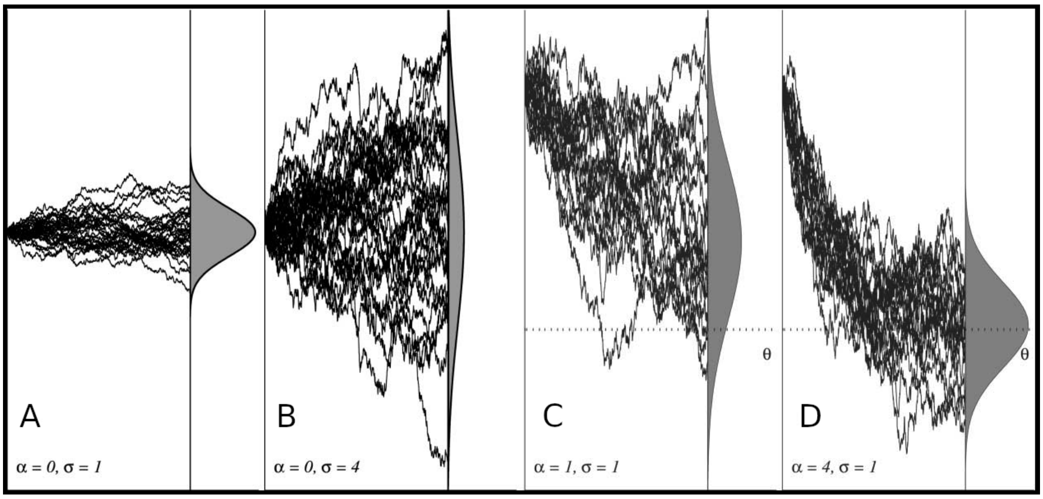

Figure 3.2 Visualizations are taken from Butler and King [41]. In the first two plots, A and B, one can see the effect of σ on the evolution of a trait that is not under selection. In the second two plots, C and D, one can see the effect of α on the evolution of a trait that has an evolutionary optimum θ. Each line represents a simulated evolutionary path of a trait under the OU model, where the y-axis is the trait value, and the x-axis is time.

Before one can use the OU model to infer the historical values of a given trait using maximum likelihood estimation or to test hypotheses about the selection acting on it, one needs three components to do the analysis [41]:

1) A set of data that includes the trait values for the species of interest In our case of passerine sperm cells, it is the gene expression values of a given gene for each of the bird species. OUCH does not account for interspecies variation and accepts only one value per species. Therefore, for the value of the gene, we have chosen the mean of the gene expression values of the given gene in the two species that had the highest pairwise Pearson correlation. Implications of not accounting for the variation and potential circumventions will be discussed later.

So that the data are more comparable for runs with different genes, we have made the expression values dimensionless as described by Cressler, Butler, and King [42]. Gene expression values were divided by subtracting a minimum value from the maximum value of a given gene's expression values.

2) A phylogenetic tree with branch lengths The tree can also be made dimensionless by dividing the branch lengths by the overall tree depth [42].

3) One or more hypotheses about the selective regimes (amount and location of optima) acting on the trait on each branch in the evolutionary history The selective regimes should be chosen based on a biological hypothesis and knowledge about the species in question. In our case of passerine birds, we have chosen to model the evolution of gene expression under the OU process with one optimum and more complex regimes, such as optima being different for each superfamily.

The problem is when one has to select regimes for the branches before the last speciation event, as the data are not available for the internal nodes of the tree (species are extinct). This can be dealt with with additional biological knowledge or logical anticipation, where one expects that when both species are promiscuous, the last common ancestor is also promiscuous and will have the same selection pressures acting. However, this is not, unfortunately, always the case. Another option is to generate multiple possible *"paintings"* (by painting, we mean assigning regimes (optima) to the branches) of the tree and then compare the likelihoods of the data under each painting. Cressler, Butler, and King [42] propose using a Monte Carlo approach to this. However, when the hypothesis is compatible with more than one painting, the uncertainty in the regime assignments remains, and rejection of any painting does not force rejection of the hypothesis [42].

Having these three components, one can use the *OUCH* package to fit the model. The package uses a maximum likelihood approach to find the best fitting values (i.e., the gene expression values in the ancestral nodes and the parameters α, θ, and σ).

Hypothesis Testing

After fitting each model, for example, one for neutral evolution (pure BM stochastic process), another for stabilizing/conserved selection (OU process with one optimum), and one for more complex selection regimes - adaptive evolution (OU process with more optima), one can use the likelihoods of the data under each model to calculate statistical support. Given two models H_0, H_1, with parameters θ_0, θ_1, and likelihoods L_0, L_1, the likelihood ratio is defined as $L_1/L_0 = \lambda$. By the result of Wilks [43], the test statistic $-2\log(\lambda)$ is asymptotically χ^2 distributed with degrees of freedom equal to the difference in dimensionality between the two models (between the BM and OU models, this will be equal to the amount of selective optima in the OU model) [29].

Pal, Oliver, and Przytycka [29] consider gene to be under stabilizing selection if the likelihood ratio test between the OU model with one optimum and the BM model is significant, and under adaptive selection if the likelihood ratio test between the OU model with more optima and the OU model with one optimum is significant and also if it is significant against the BM model. However, this type of hypothesis testing biases to report the gene as under the stabilizing selection. Claiming that the gene is under stabilizing selection might not be the best interpretation, as it is somewhat under some kind of selection to which the extent and mechanism remains partially unknown.

As each gene is tested separately, the problem of multiple hypothesis testing arises, so a correction is needed. One can, for example, employ the Benjamini-Hochberg correction for this.

3.3.2 Approaches to OU Modeling and Their Limitations

Hansen originally proposed using the OU model for general continuous quantitative traits. Bedford and Hartl have suggested that it might also be an appropriate model for gene expression evolution [44]. Brawand et al. [28] have made one of the first studies with a large impact using the OU model to study the evolution of gene expression in 10 species of mammals and from 6 different tissues. They found that most of the genes were under the stabilizing selection, and only a smaller fraction was under adaptive selection. Nevertheless, different genes were under different selection pressures when expressed in different tissues. This suggests that the selection pressures acting on the gene expression are highly tissue-specific - even more than species-specific.

Chen et al. [44] expand on methodology from Brawand et al. [28] where they studied evolution across 17 mammalian species. They also considered using the OU model to predict deleterious levels of gene expression and tested it on identifying a gene responsible for muscular dystrophy.

However, the *OUCH* package only allows one to input one value for each species, which in turn cannot account for the intra-species variation. For this reason, [29] have developed an R package *EvoGeneX*, which uses additional parameter γ^2 to account for the intra-species variation. Unfortunately, the package has incorrectly declared dependencies, making installation difficult and requiring a repository fork. Also, it does not provide a function to easily create the regime assignment table, which *OUCH* does.

When inspecting the equation 3.1, one can see that the OU model is additive in the stochastic part, meaning that when the given feature is, for example, equal to 1000, then its evolution to value 2000 would be just as likely as its evolution from 10000 (if it ever were to reach this value) to 11000, when given the same amount time to evolve neutrally. In the case of RNA-seq data normalized using TPM or RPKM methods, we do not expect the evolution of gene expression to follow an additive model; rather, it tends to be multiplicative. Evolutionary changes, such as modifications in the promoter region or enhancer sequences (or proteins acting on them), typically alter gene expression through multiplicative factors. This is because changes in transcription factor affinity usually affect gene expression proportionally by a certain percentage, not incrementally in an absolute way. Therefore, taking a logarithm of the expression values might be a more appropriate scale for the OU model, as additive changes in the log scale are multiplicative in the original scale. Pal, Oliver, and Przytycka [29] and Brawand et al. [28] in their papers do not do this transformation to their data (at least it is not described in their methodology and supplementary materials); however, Chen et al. [44] have used a log-transformed data.

We could not find any papers that would have used the OU model to study the evolution of gene expression values obtained from LC-MS/MS proteomics data. Based on the previously discussed section, we have used VSN normalized data, which are already log-transformed, and we have used the *OUCH* package to model the evolution of gene expression in passerine birds. We have also tried doing the same analysis on exponentiated data from the VSN normalization, and interestingly, the results did not significantly differ when testing for constrained evolution. However, in a few cases, the results were very different. For example, in the case of the EZR gene, the protein was under stabilizing selection (stabilizing according to Pal, Oliver, and Przytycka [29] interpretation of test results, described earlier) when using the log-transformed data (BH adjusted p-value < 0.05) but not when using the exponentiated data.

When doing the OU modeling of adaptive evolution, where the optima were selected differently for each bird superfamily, we failed to find any statistically significant genes that would be under adaptive evolution (when compared to the stabilizing selection model). We hypothesize that this could have occurred due to several things:

- The adaptive evolution of the sperm proteome is not so dependent on the bird superfamily but rather on the species itself and more on its mating strategy. This assumption is supported by section 2.3.1, where we have shown that the proteomes follow the phylogeny of the birds only in a limited way.

- The adaptive model is more complex (has more degrees of freedom) than the stabilizing model; therefore, the likelihood ratio test is less in favor of the adaptive model.

- Overall, the data quality is insufficient to detect adaptive evolution.

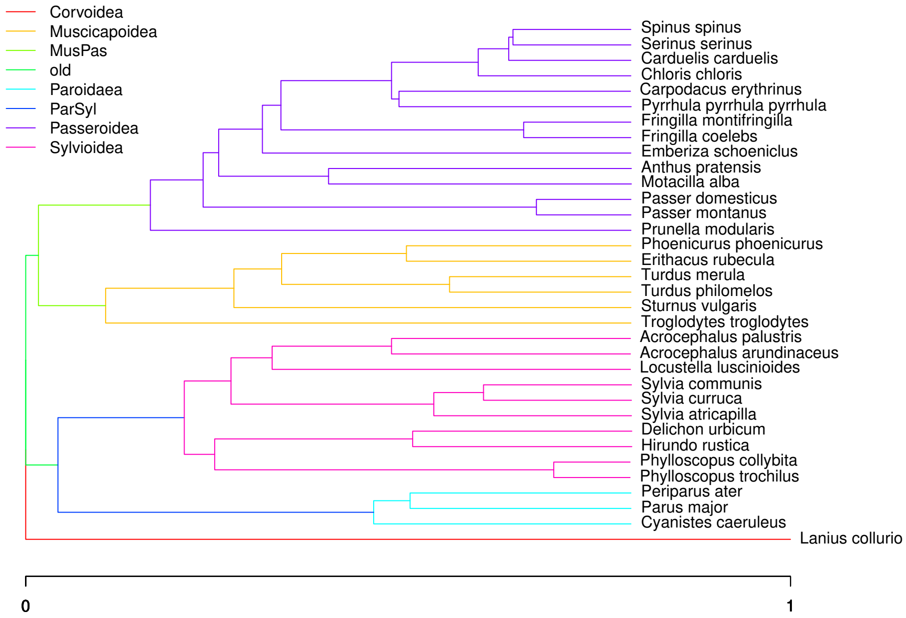

Figure 3.3 Painting of the regime assignments to the passerine bird tree. The painting has eight optima, one for each bird superfamily and then for their common ancestors. Note that there is no trifurcation in the tree; it is just that the branch length is too short to be seen.

3.4 Machine Learning

Over the years, machine learning (ML) has proven to be a powerful tool for analyzing large or highly complex datasets. It is used in various fields, from economics to biology. ML can be utilized for several goals in the context of gene expression data. One might care if a patient with cancer will respond to a particular therapy, or even if the patient has cancer at all, and if yes, at what stage. ML can also be used in explanatory analysis as certain algorithms can provide insights into which features are important for the prediction of a given phenotype and can then be considered as potential biomarkers.

Machine learning is mainly divided into two categories: supervised and unsupervised learning.

3.4.1 Supervised Learning

Supervised learning is a type of ML where the algorithm learns from labeled data. For example, one can have a proteomic profile of a patient and a label indicating whether the patient has cancer. The algorithm then learns to predict the label based on the proteomic profile. And then, one can use the model to predict the label of a new patient based on their profile. Bostanci et al. [45] have done exactly this, but using the RNA-seq data, with the model's accuracy being above 95% for both cancer and stage prediction.

Here, we propose a basic pipeline for supervised learning on proteomic gene expression data:

1. **Data Preprocessing:** Presumably, one of the most important things for the model's success is feature preparation when using gene expression data rather than the actual choice of algorithm [45, 37, 46]. Gene expression datasets usually contain hundreds if not thousands of features (genes); however, not all are relevant for predicting the phenotype. Therefore, feature selection is crucial. One can utilize information gain to assess how well individual features separate the classes based on entropy reduction when dealing with categorical target variables. This method quantifies the decrease in entropy—representing disorder or unpredictability—achieved by segmenting the dataset according to each feature, thus identifying those that are most effective at predicting the target variable [45]. Another option might be to use genes that are significantly differentially expressed between the conditions, as these genes might be important for the phenotype. ANOVA-based feature selection can be used for this purpose [37]. Evolutionary information can also be utilized to select features. For example, Cheng et al. [47] have used genes whose response to nitrogen treatment was conserved within and across different species when predicting the efficiency of nitrogen metabolism. We hypothesize that results from OU modeling can be utilized for this purpose as well. Lastly, one can utilize packages explicitly designed for feature selection regardless of the data type. One such example is *FeatureWiz*[48], which can add new features using autoencoders and then select the most relevant features using the SULOV method that finds highly correlated variables and keeps those with the highest mutual information score with the target variable. Lastly, by running a recursive XGBoost, it selects the most important features from the SULOV selected features. Features can also be further transformed using PCA, UMAP, SVD, autoencoders, or other dimensionality reduction methods. However, one then loses the interpretability of the model as the features no longer represent the individual genes.

Another common problem with gene expression data is that the number of samples is too small, and even with smaller feature sets, the models still overfit. To mitigate this, one can artificially enlarge the dataset. For categorical target variables, one can use the SMOTE method to generate synthetic samples of the minority class (but not only as it can be done on the whole dataset) [45]. SMOTE generates synthetic samples by selecting two or more similar samples and creating a new sample that is a linear combination of the selected samples. For continuous target variables, the problem is more complicated. However, few methods, such as *SMOGN[49]* or Variational Autoencoders (VAE), have been developed to achieve this, yet their use in practice is limited.

In our case of passerine sperm cells, data imputed from MissForest performed better than when using original VSN normalized data with naive zero imputation.

2. **Model Selection:** After the feature selection, one can proceed with the data modeling. The choice of the model depends on the data and the problem at hand. Random Forests (RFs) and Gradient Boosted Decision Trees (GBDTs) have proven to be the most successful models for tabular data. Nevertheless, they can be more prone to overfitting than other models. However, one of the great advantages of these models is that they can provide feature importance [50], which can be used for interpretability and finding which genes are important for the prediction and, therefore, are likely to be involved in the phenotype [3]. However, genes highly correlated with these genes could have been removed in the feature selection process, yet they could play a crucial role in the phenotype. Of course, non-ensemble models such as SVMs, logistic or ridge regression, or deep neural networks (DNNs) can also work well with expression data [45, 37], and also allow for feature importance extraction when coupled with for example ANOVA-based feature selection. Nonetheless, it is not as straightforward as with RFs or GBDTs [51]. Tyanova et al. [51] have used SVMs to find proteins related to specific breast cancer subtypes and their processing pipeline is implemented in the *Perseus*[52] software. In the case of colon cancer predictions, 1D-CNNs and Bi-LSTM models have achieved the best performance, even beating out the Random Forest classifier [45].

3. **Model Evaluation:** It is always necessary to evaluate the model's performance on unseen data. The most common metrics for classification tasks are accuracy, precision, recall, F1 score, and AUC-ROC. RMSE, MAE, R-squared, or correlation coefficient can be used for regression tasks. The evaluation needs to be done on a separate dataset that was withheld from

the training process. Cross-validation can be utilized to evaluate the model on the whole dataset accurately. The dataset is divided into k folds, and the model is trained on $k - 1$ folds and evaluated on the remaining fold. This process is repeated k times, and the results are averaged. This method can provide a more robust evaluation of the model's performance [45].

Of course, this pipeline is not universal and exhaustive, and one might need to adjust it based on the data and the problem at hand. Nevertheless, it provides a good starting point for supervised learning on gene expression data. We used this to predict the sperm mid-piece length and other phenotypic traits of passerine birds. With only 102 samples, we were able to get meaningful predictions, signifying the relationship of the proteome with the phenotype. Unsurprisingly, when trying to predict plumage score, the model made basically random predictions (Pearson correlation of -0.05 using GBDT model and 0.02 using SVR), as the sperm proteome is not expected to be related to the plumage.

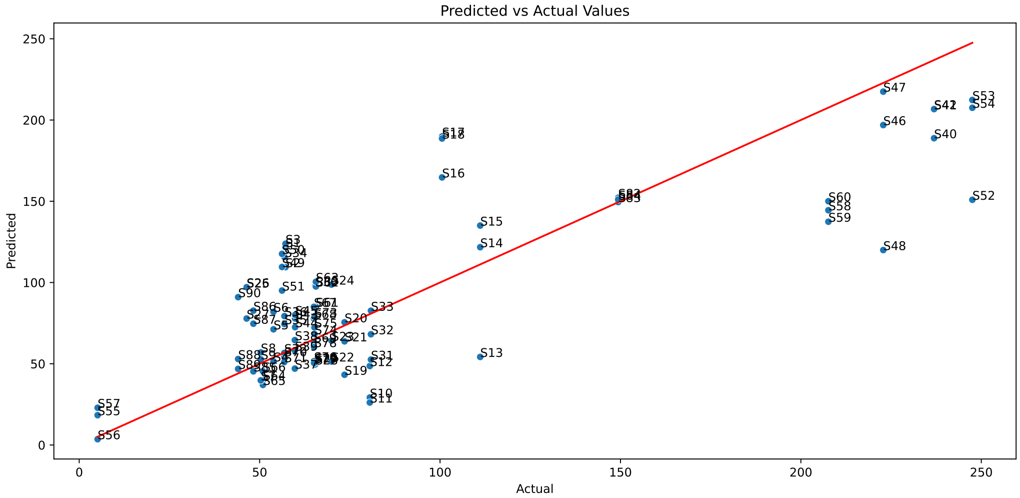

Figure 3.4 Plot of the observed sperm mid-piece lengths against the predicted values of mid-piece lengths of passerine birds using the Support Vector Regressor (SVR). Group k fold validation was done to evaluate the model. The model achieved an average of 0.73 Pearson correlation coefficient and 36.28 RMSE on the test sets. However, it can be seen the model doesn't generalize that well on rare expression profiles but works decently well on common ones.

3.4.2 Unsupervised Learning

Unsupervised learning is a type of ML where the algorithm learns from unlabeled data. Usually, the goal is to find certain patterns in the data, such as clusters of similar samples or features, or to reduce the dimensionality of the data. Unsupervised learning can go hand in hand with exploratory data analysis, as it can provide insights into the data structure and can aid with hypothesis generation or help with data quality assessment. As shown in 2.3, we used PCA to visualize the bird samples. Methods like t-SNE, UMAP or newly developed methods like PaCMAP [53] can also be used for data visualization and clustering purposes. However, it is important to note that methods like UMAP or PaCMAP assume that the data points lie on a Riemannian manifold, and this assumption might not always hold. Usually, these methods are used for single-cell transcriptomics data, where the assumption is more likely to hold.

K-means is another unsupervised learning algorithm that can assign data points to clusters. For example, it can be used to find subgroups of cancer patients based on their gene expression profiles. However, K-means has several limitations, such as the need to specify the number of clusters beforehand, the assumption that the clusters are spherical, and the sensitivity to outliers. To find the optimal number of clusters in K-means, one can use the elbow method, silhouette score, or gap statistics.

Another popular method is hierarchical clustering, which creates a tree-like structure (dendrogram) of the data points based on their similarity. When doing hierarchical clustering on genes, one can find groups of genes that might be co-regulated, co-expressed, or related to a given phenotype [51].

TDA methods utilizing algebraic topology, such as Mapper, can be used to visualize high-dimensional data in a lower-dimensional space and find clusters or topological features in the data. Li et al. [54] have used TDA to find and describe subgroups of patients with type 2 diabetes based on their electronic medical records (EMRs). Nonetheless, one can use proteome profiles instead of EMRs to calculate the distance between the patients. Yet, TDA methods have not been widely used in the context of proteomics, and this might be an exciting area for future research as TDA is proving very useful in single-cell transcriptomics.

Discussion and Conclusion

Discussion

This thesis explored the journey of interpreting and analyzing proteomic data to understand phenotypic traits. The complexities of LC-MS/MS, protein identification, quantification methods (both labeled and label-free), and the critical role of data preprocessing in ensuring reliable downstream analysis were described, and its potential shortcomings were discussed. Furthermore, several bioinformatics approaches to obtain biological insights from the proteomics data were presented with corresponding software tools to perform them.

However, these tools can differ significantly in their performance, and the choice of the tool can significantly impact the results. Therefore, several descriptions with corresponding assumptions and benchmarks of the tools were presented here to aid in selecting the right software or algorithm. Unfortunately, due to a lack of gold-standard datasets and usually a small sample size of compared tools, the benchmarking results can be biased, and more work in this area is needed to ensure that researchers can choose the right tool for their specific needs.

Nevertheless, in many cases, VSN has proven to be a reliable normalization method, beating other methods on several benchmarks and datasets. In the case of missing values imputation, MissForest has shown to be a good performer for data that contain both missing at-random and not-at-random values and GSimp has proven to be a good choice for left-censored data.

Also, new adaptations of methodologies here have been proposed to analyze proteomics data, such as constructing an evolutionary tree from proteomic profiles or using the Ornstein-Uhlenbeck process to model the evolution of protein abundances. Several problems with the OU modeling were identified that had not been addressed in previous studies, signifying the need for carefulness when interpreting the results from these methods and further research. Lastly, machine learning approaches were discussed, and how one can utilize them in overall proteomic data analysis. Furthermore, how they can help with medicinal diagnostics and biomarker discovery research was described.

Conclusion

The field of proteomics has great potential to help us to uncover relationships between proteins and phenotypic traits. However, the road to understanding these relationships is long and complex, and many challenges need to be addressed, as incorrect decisions can lead to misleading results. Also, analyzing the data correctly still requires much work and expert knowledge, making it difficult for researchers to utilize proteomics data to its full potential. Hopefully, this thesis has provided a comprehensive overview of the field and the tools available to researchers to make the journey easier.

Bibliography

[1] Mark S. Hill, Pétra Vande Zande, and Patricia J. Wittkopp. "Molecular and evolutionary processes generating variation in gene expression". In: *Nature Reviews Genetics* 22.4 (Dec. 2020), 203–215. ISSN: 1471-0064. DOI: 10.1038/s41576-020-00304-w.

[2] Jenny Chen et al. "A quantitative framework for characterizing the evolutionary history of mammalian gene expression." In: *Genome Research* (2019). DOI: 10.1101/gr.237636.118.

[3] Chen Chen et al. "Bioinformatics Methods for Mass Spectrometry-Based Proteomics Data Analysis". In: *International Journal of Molecular Sciences* 21.8 (Apr. 2020), p. 2873. ISSN: 1422-0067. DOI: 10.3390/ijms21082873.

[4] Paul R. Graves and Timothy A. J. Haystead. "Molecular Biologist's Guide to Proteomics". In: *Microbiology and Molecular Biology Reviews* 66.1 (Mar. 2002), 39–63. ISSN: 1098-5557. DOI: 10.1128/mmbr.66.1.39-63.2002.

[5] Wenhong Zhu et al. "Mass spectrometry-based label-free quantitative proteomics." In: *BioMed Research International* (2010). DOI: 10.1155/2010/840518.

[6] Timothy Clough et al. "Statistical protein quantification and significance analysis in label-free LC-MS experiments with complex designs." In: *BMC Bioinformatics* (2012). DOI: 10.1186/1471-2105-13-s16-s6.

[7] Kondethimmanahalli Chandramouli and Pei-Yuan Qian. "Proteomics: challenges, techniques and possibilities to overcome biological sample complexity". In: *Human genomics and proteomics: HGP* 2009 (2009).

[8] Stefka Tyanova, Tikira Temu, and Juergen Cox. "The MaxQuant computational platform for mass spectrometry-based shotgun proteomics". In: *Nature protocols* 11.12 (2016), pp. 2301–2319.

[9] Jurgen Cox et al. "Andromeda: a peptide search engine integrated into the MaxQuant environment". In: *Journal of proteome research* 10.4 (2011), pp. 1794–1805.

[10] Jana Zecha et al. “TMT labeling for the masses: a robust and cost-efficient, in-solution labeling approach*[S]”. In: *Molecular & cellular proteomics* 18.7 (2019), pp. 1468–1478.

[11] Lei Zhao et al. “Comparative evaluation of label-free quantification strategies”. In: *Journal of proteomics* 215 (2020), p. 103669.

[12] Tommi Välikangas, Tomi Suomi, and Laura L. Elo. “A systematic evaluation of normalization methods in quantitative label-free proteomics”. In: *Briefings in Bioinformatics* (2016). DOI: `10.1093/bib/bbw095`.

[13] Yuliya V. Karpievitch, Alan R. Dabney, and Richard D. Smith. “Normalization and missing value imputation for label-free LC-MS analysis.” In: *BMC Bioinformatics* (2012). DOI: `10.1186/1471-2105-13-s16-s5`.

[14] Jurgen Cox et al. “Accurate Proteome-wide Label-free Quantification by Delayed Normalization and Maximal Peptide Ratio Extraction, Termed MaxLFQ”. In: 13 (Sept. 2014). ISSN: 1535-9476. DOI: `10.1074/mcp.m113.031591`.

[15] Matthew C Leitch, Indranil Mitra, and Rovshan G Sadygov. “Generalized linear and mixed models for label-free shotgun proteomics”. In: *Statistics and its Interface* 5.1 (2012), p. 89.

[16] Thomas Naake, Johannes Rainer, and Wolfgang Huber. “MsQuality: an interoperable open-source package for the calculation of standardized quality metrics of mass spectrometry data”. In: *Bioinformatics* 39.10 (Oct. 2023). Ed. by Macha Nikolski. ISSN: 1367-4811. DOI: `10.1093/bioinformatics/btad618`.

[17] Yusuf Khan et al. “Normalization of gene expression data revisited: the three viewpoints of the transcriptome in human skeletal muscle undergoing load-induced hypertrophy and why they matter”. In: *BMC Bioinformatics* 23.1 (June 2022). ISSN: 1471-2105. DOI: `10.1186/s12859-022-04791-y`.

[18] Tommi Valikangas, Tomi Suomi, and Laura L Elo. “A systematic evaluation of normalization methods in quantitative label-free proteomics”. In: *Briefings in bioinformatics* 19.1 (2018), pp. 1–11.

[19] Ciaran Evans, Johanna Hardin, and Daniel M Stoebel. “Selecting between-sample RNA-Seq normalization methods from the perspective of their assumptions”. In: *Briefings in bioinformatics* 19.5 (2018), pp. 776–792.

[20] Benjamin M Bolstad et al. “A comparison of normalization methods for high density oligonucleotide array data based on variance and bias”. In: *Bioinformatics* 19.2 (2003), pp. 185–193.

[21] Wolfgang Huber et al. “Variance stabilization applied to microarray data calibration and to the quantification of differential expression”. In: *Bioinformatics* 18.suppl_1 (2002), S96–S104.

[22] Stefan Graw et al. “proteiNorm–A user-friendly tool for normalization and analysis of TMT and label-free protein quantification”. In: *ACS omega* 5.40 (2020), pp. 25625–25633.

[23] Aakash Chawade, Erik Alexandersson, and Fredrik Levander. “Normalyzer: a tool for rapid evaluation of normalization methods for omics data sets”. In: *Journal of proteome research* 13.6 (2014), pp. 3114–3120.

[24] Liang Jin et al. “A comparative study of evaluating missing value imputation methods in label-free proteomics”. In: *Scientific reports* 11.1 (2021), p. 1760.

[25] Weijia Kong et al. “Dealing with missing values in proteomics data”. In: *Proteomics* 22.23-24 (2022), p. 2200092.

[26] Daniel J Stekhoven and Peter Bühlmann. “MissForest—non-parametric missing value imputation for mixed-type data”. In: *Bioinformatics* 28.1 (2012), pp. 112–118.

[27] Michiel P van Ooijen et al. “Identification of differentially expressed peptides in high-throughput proteomics data”. In: *Briefings in bioinformatics* 19.5 (2018), pp. 971–981.

[28] David Brawand et al. “The evolution of gene expression levels in mammalian organs”. In: *Nature* 478.7369 (2011), pp. 343–348.

[29] Soumitra Pal, Brian Oliver, and Teresa M Przytycka. “Stochastic modeling of gene expression evolution uncovers tissue-and sex-specific properties of expression evolution in the Drosophila genus”. In: *Journal of Computational Biology* 30.1 (2023), pp. 21–40.

[30] Daniela M Witten. “Classification and clustering of sequencing data using a Poisson model”. In: (2011).

[31] Francois-Joseph Lapointe and Guy Cucumel. “The average consensus procedure: combination of weighted trees containing identical or overlapping sets of taxa”. In: *Systematic Biology* 46.2 (1997), pp. 306–312.

[32] Liam J. Revell. “phytools 2.0: an updated R ecosystem for phylogenetic comparative methods (and other things).” In: *PeerJ* 12 (2024), e16505. DOI: `10.7717/peerj.16505`.

[33] Aravind Subramanian et al. “Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles”. In: *Proceedings of the National Academy of Sciences* 102.43 (2005), pp. 15545–15550.

[34] Tianzhi Wu et al. “clusterProfiler 4.0: A universal enrichment tool for interpreting omics data”. In: *The innovation* 2.3 (2021).

[35] Norman Pavelka et al. “Statistical similarities between transcriptomics and quantitative shotgun proteomics data”. In: *Molecular & Cellular Proteomics* 7.4 (2008), pp. 631–644.

[36] Matthew E Ritchie et al. “limma powers differential expression analyses for RNA-sequencing and microarray studies”. In: *Nucleic acids research* 43.7 (2015), e47–e47.

[37] Michal Harel et al. “Proteomics of melanoma response to immunotherapy reveals mitochondrial dependence”. In: *Cell* 179.1 (2019), pp. 236–250.

[38] Mingze Bai et al. “LFQ-based peptide and protein intensity differential expression analysis”. In: *Journal of Proteome Research* 22.6 (2023), pp. 2114–2123.

[39] Miao-Hsia Lin et al. “Benchmarking differential expression, imputation and quantification methods for proteomics data”. In: *Briefings in Bioinformatics* 23.3 (2022), bbac138.

[40] Barbora Kuntová, Romana Stopková, and Pavel Stopka. “Transcriptomic and proteomic profiling revealed high proportions of odorant binding and antimicrobial defense proteins in olfactory tissues of the house mouse”. In: *Frontiers in Genetics* 9 (2018), p. 297892.

[41] Marguerite A Butler and Aaron A King. “Phylogenetic comparative analysis: a modeling approach for adaptive evolution”. In: *The american naturalist* 164.6 (2004), pp. 683–695.

[42] Clayton E Cressler, Marguerite A Butler, and Aaron A King. “Detecting adaptive evolution in phylogenetic comparative analysis using the Ornstein–Uhlenbeck model”. In: *Systematic biology* 64.6 (2015), pp. 953–968.

[43] Samuel S Wilks. “The large-sample distribution of the likelihood ratio for testing composite hypotheses”. In: *The annals of mathematical statistics* 9.1 (1938), pp. 60–62.

[44] Jenny Chen et al. “A quantitative framework for characterizing the evolutionary history of mammalian gene expression”. In: *Genome research* 29.1 (2019), pp. 53–63.

[45] Erkan Bostanci et al. “Machine learning analysis of RNA-seq data for diagnostic and prognostic prediction of colon cancer”. In: *Sensors* 23.6 (2023), p. 3080.

[46] Tong Lim Shiuh et al. “Prediction of Thyroid Disease using Machine Learning Approaches and Featurewiz Selection”. In: *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* 15.3 (2023), pp. 9–16.

[47] Chia-Yi Cheng et al. “Evolutionarily informed machine learning enhances the power of predictive gene-to-phenotype relationships”. In: *Nature communications* 12.1 (2021), p. 5627.

[48] Ram Seshadri. *GitHub - AutoViML/featurewiz: Use advanced feature engineering strategies and select the best features from your data set fast with a single line of code.* `https://github.com/AutoViML/featurewiz`. source code. 2020.

[49] Nicholas Kunz. *SMOGN: Synthetic Minority Over-Sampling Technique for Regression with Gaussian Noise.* Version v0.1.2. 2020. URL: `https://pypi.org/project/smogn/`.

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

[51] Stefka Tyanova et al. “Proteomic maps of breast cancer subtypes”. In: *Nature communications* 7.1 (2016), p. 10259.

[52] Stefka Tyanova et al. “The Perseus computational platform for comprehensive analysis of (prote) omics data”. In: *Nature methods* 13.9 (2016), pp. 731–740.

[53] Haiyang Huang et al. “Towards a comprehensive evaluation of dimension reduction methods for transcriptomic data visualization”. In: *Communications biology* 5.1 (2022), p. 719.

[54] Li Li et al. “Identification of type 2 diabetes subgroups through topological analysis of patient similarity”. In: *Science translational medicine* 7.311 (2015), 311ra174–311ra174.