ADVANCED STRATEGIES FOR MODELING LOW-INTENSITY SIGNALS FROM MULTI-OMICS DATA

by

Fabian Bong

Submitted in partial fulfillment of the requirements
for the degree of Master of Science

at

Dalhousie University
Halifax, Nova Scotia
August 2024

I dedicate this thesis to my parents, who, without a doubt, are the reason I have been able to make it this far; to my girlfriend, whom I have to thank for strengthening my trust in myself; and to my younger self, the person who decided to take this path in life.

Table of Contents

List of Tables

List of Figures

Abstract

Since the turn of the 21st century, research in the biomedical sciences has dramatically shifted and become reliant on high throughput bioanalytical measurements for investigating bio-molecular events that are presumed to be self-orchestrated and organized in hierarchies. These measurements yield large datasets which, consequently, require specialized statistical (bioinformatics) approaches with which to extract the most meaningful information given observational or designed experiments. At the outset, data acquisition outpaced model development, and ad-hoc approaches were used to glean the most obvious information. With the maturity of bioinformatics, important questions are beginning to emerge regarding the extent to which significant biological information can be extracted from these data given challenges such as: (a) their divergent distributional characteristics, (b) the large dynamic range in their signals, and (c) allowable effect sizes given limitations in sample size and associated variability. Here, two approaches are introduced to overcome these challenges. First, kurtosis-based projection pursuit, augmented with classification and regression trees (kPPA-CART) is proposed as a robust, easy-to-implement approach to model multi-omics data that are derived from next-generation sequencing (NGS) and mass spectrometry (MS). Most of the available methods for unsupervised multi-omics integration suffer from the inability to model low-intensity (low count) features and instead focus on highly variable (dominant) features. Comprehensive benchmarking of existing multi-omics integration tools against kPPA-CART was performed using simulated data where the changes involved in a hypothetical biological phenomenon are associated with low-intensity signals and small effect sizes. The results show that kPPA-CART provides a superior recovery of this information. The application of this method is supported by the development of an R Package (https://github.com/FabianBong/KPPACart) and an easily accessible web tool (https://intmove.vercel.app/) that allow experimentalists to implement kPPA-CART without the need for computational training. Second, to the extent that measurement uncertainties affect data analysis strategies, distributional assumptions accompanying many methods for -omics data analysis assume an independent, identical, and normally distributed (*iid* normal) structure for the noise. When this assumption is violated, one practical solution is to incorporate the true structure of the measurement error variance in the analysis. However, this requires extensive replication which can become prohibitive. Here, two approaches (Frequentist and Bayesian) are introduced for developing a parametric estimate of error variance incorporating shot and proportional noise for LC-MS data using, as a base, empirical replicate measurements. This thesis provides evidence that both methods accurately recapitulate the parameters of the variance function while accounting for sensitivity differences between replicate samples, and enable test statistics from the exponential family of distributions to be conducted without loss of generality.

List of Abbreviations Used

BC Breast cancer.

CART Classification and regression trees.

CCA Canonical correlations analysis.

CpG Areas of high concentration of Cytosine and Guanine.

DIABLO Data integration analysis for biomarker discovery using latent components.

DNA Deoxyribonucleic acid.

EMOGEA Error modelled gene expression analysis.

ESS Effective sample size.

FA Factor analysis.

GC-MS Gas chromatography–mass spectrometry.

GCCA Generalized canonical correlation algorithm.

HCA Hierarchical cluster analysis.

HMC Hamilton monte carlo.

ICA Independent component analysis.

IntNMF Integrative non-negative matrix factorization.

IPCA Independent principal component analysis.

JDR Joint dimensionality reduction.

JIVE Joint and individual variation explained.

kPPA Kurtosis projection pursuit analysis.

kPPA-CART Kurtosis-based projection pursuit analysis, augmented with classification and regression trees.

LC-MS Liquid chromatography–mass spectrometry.

LOD Limit of detection.

MA Minus/Average normalization.

MBC Metastatic breast cancer.

MCIA Multiple co-inertia analysis.

MCMC Markov chain monte carlo.

MDS Multi-dimensional scaling.

METABRIC Molecular Taxonomy of Breast Cancer International Consortium.

ML Machine learning.

MLPCA Maximum likelihood principal component analysis.

MOFA Multi-omics factor analysis.

mRNA Messenger ribonucleic acid.

MS Mass spectrometry.

NGS Next-generation sequencing.

NNMF Non-negative matrix factorization.

NUTS No-u-turn-sampler.

PAM50 A list of fifty genes that classify breast cancer into different subtypes.

PCA Principal component analysis.

PI Projection index.

PLS-DA Partial least squares - discriminant analysis.

PP Projection pursuit.

PPF Projection pursuit features.

RNA Ribonucleic acid.

RNA-Seq RNA sequencing.

S/N Signal to noise ratio.

SC Silhouette coefficient.

TCGA The Cancer Genome Atlas.

TME Tumor microenvironment.

TOF Time-of-flight.

TOF-MS Time-of-flight mass spectrometry.

VF Variance function.

Chapter 1

Introduction

Beginning in the late 1990s, the paradigm of quantitative biological measurements has profoundly shifted from traditional single sample (and variable) measurements to high throughput measurements, powered by rapid advances in bioanalytical technologies and exponential growth of computational power. This shift has cleaved space for a novel discipline within biological sciences referred to as bioinformatics, which has set itself the goal to analyze and manage the large datasets accompanying these measurement strategies. Research in bioinformatics involves a significant investment of time and computational resources to separate biological from technical variation following simultaneous measurements of biomolecular components of an organism using bioanalytical technology platforms. Whereas the former (biological variation) reflects the organism's observable characteristics (phenotype) [38] it is confounded by the latter. Conceptually, changes in an organism's physico-chemical or physiological environment would correlate with changes in the composition of its biomolecular milieu. At the time of writing this thesis, biomolecules that have received the most attention in academic literature are genes, proteins, and metabolites, which are known to play concerted roles in maintaining biological homeostasis and are directly modulated as an organism exhibits changes in their phenotype [92].

This interesting and delicate functional interplay between genes, proteins, and metabolites is formulated as "The Central Dogma of Biology" (Figure 1.1). It was conceived in 1985 by Francis Crick who is believed to be the father of molecular biology. The dogma serves as a guiding principle for understanding fundamental processes governing the flow of biological information. Briefly, it encapsulates the unidirectional flow of genetic information, postulating that the organism's phenotype is encoded in a sequence of deoxyribonucleic acids (DNA) that make up genes [25]. DNA consists of a chain of nucleotide polymers that are made up of pyrimidine

(**C**ytosine, **T**hymine, and **U**racil) or purine (**A**denine and **G**uanine) bases attached to a sugar-phosphate backbone. To form the commonly observed double-stranded helix, two single strands of chained nucleotide polymers hybridize, matching bases **A**-to-**T** and **G**-to-**C** through the formation of hydrogen bonds [108]. Following the flow of information, the DNA is then transcribed into messenger ribonucleic acid (mRNA), commonly referred to as gene transcripts.

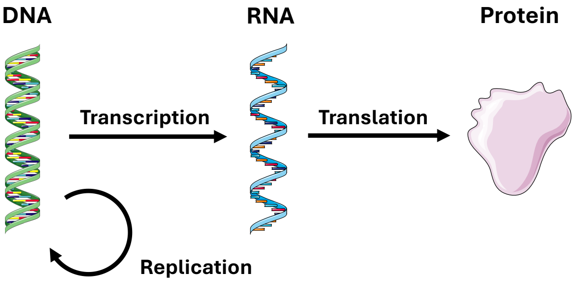

Figure 1.1: Concept figure of "The Central Dogma of Biology". Generally, DNA is exposed to transcription resulting in RNA which in turn is translated to a functional protein. The diagram was created with assets from the Servier Medical Art library [1], available under CC BY 4.0.

On a macro level, the transcripts (mRNA) are comparable to a single strand of DNA with the exception of **T**hymine bases being replaced by **U**racil. The formed mRNA, can be translated into proteins, which are the functional units of a cell and are made up of peptide chains of amino acids [3]. Proteins typically function in tandem with metabolites to actively change the biomolecular milieu in cells, resulting in changes of an organism's observable characteristics [25]. Over the years, and with increased knowledge about the fundamental principles of molecular biology, nuanced expansions have been made to the basic processes, embracing for example, that constitutional and somatic changes can affect the different steps within the Central Dogma of Biology.

The linear flow of information, advanced by the central dogma of molecular biology, ignores the occasional occurrence of changes to the pre-determined sequence of genes within the genome (*i.e.*, mutations), or chemical modifications to the structure of DNA (*i.e.*, epigenetic modifications). Genetic mutations include constitutional changes that refer to heritable, direct alterations in an organism's blueprint which profoundly affect its biological makeup. On the other hand, somatic mutations refer to changes in genetic sequences through a process involving spontaneous and cumulative (but permanent) alterations to one or several sections of genomic DNA. There is a wide array of tutorials and reviews that provide detailed overviews of the concept of genetic mutations and how they relate to changes in phenotypes [86]. For the purpose of this thesis, an important consideration is that genetic mutations explicitly change the information stored within the organism's blueprint. This change propagates into gene transcripts as well as proteins and can be quantified using bioanalytical technology platforms. For example, two groups of people can have differential responses to drug treatments if they exhibit rare variants that allow them to hyper-metabolize a drug [86]. Such an observation is measurable by quantifying the expression of genes, *e.g.*, the Cytochrome P450 family, which is responsible for the metabolism of many drugs. In diseases such as cancer, somatic mutations can manifest themselves through deletions, amplifications, or re-arrangements of regions of a gene that may be responsible for maintaining and controlling the division of cells [86]. This information can also be manifested in the expression levels of genes within the cell cycle which can be measured analytically.

Other non-mutational alterations to the genetic sequences exist and are driven by modifications of the chemical structure of individual DNA molecules (rather than deletions or amplifications of sequences of genes), and are referred to as epigenetic changes. These include, for example, the addition or removal of methyl- or acetyl groups to the sugar-phosphate backbone of DNA molecules. The consequence is that methylation of the 5' carbon (the fifth carbon) of a cytosine nucleotide, for instance, will impair the transcription of genes with such modifications. Knowing this provides opportunities for specialized high throughput bioanalytical approaches to measure genes that harbor epigenetic changes such as DNA methylation [46].

The quest to measure the sequence of the full complement of genes that comprise living things (the genome) started as early as 1960 pioneered by Frederick Sanger who, in 1967, successfully sequenced the 120-nucleotide long sequence of *Escherichia coli* (E. coli) [84]. This short sequence created a stepping stone to understanding pathogenetic properties that relied on the integrity of the genomic sequence. However, Sanger's original method soon lost traction as its simplicity did not meet the requirements necessary to sequence larger nucleotide sequences. Thus, bioinformatics, a field dedicated to developing high-tech instruments for fast, detailed, and high-throughput quantification was born. Today, bioinformatics-centered research has developed technology that allows the human genome to be sequenced within time frames as short as 5 hours.

These advancements in bioanalytical technologies, since Sanger's original genomic sequencing, are (un)fortunately accompanied by a deluge of data. The unintended consequence of this is the emergence of bioinformatics not just as an experimental biology-focused field but as a sub-specialty of data science dedicated to extracting meaningful biological information from these data. Researchers commonly split acquired data into groups depending on what *stage* of the Central Dogma the data represents. Each *stage* of the Central Dogma is referred to as a specific *-ome* and has their own dedicated and intricate measurement apparatus. The genome encompasses concepts associated with the DNA, such as both somatic and constitutional alterations as well as epigenetic changes, and is investigated through high-throughput sequencing and microarray technologies. The transcriptome, mainly referring to mRNAs, is quantified with similar approaches. In contrast, the proteome and metabolome, which focus on proteins and metabolites, respectively, rely on diverse iterations of separation and mass spectrometry methodologies for qualitative and quantitative analysis [92].

The first of the advanced and widely-used bioanalytical technologies to quantify changes in the genome was the gene expression microarray. These were characterized by tens of thousands of single-stranded DNA segments (each representing a gene) that were immobilized onto known locations on a glass slide. mRNA's were then extracted from the samples of interest, labeled with fluorescence dyes, and hybridized to the microarray. Such an experiment would result in competitive hybridization of labeled sample mRNAs to their complementary sequences on the chip and, upon fluorescence excitation, the extent of this hybridization could be quantified. To achieve best results, the selected probes must be tailored to the research question of interest. If data is required for non-targeted probes after the experiment, a new microarray must be prepared. Additionally, microarrays tend to stretch financial and temporal resources while containing large amounts of noise. These shortcomings have led to the advent of Next Generation Sequencing (NGS) which enables non-specific, parallel sequencing of millions of DNA and RNA fragments in a single experimental procedure [81].

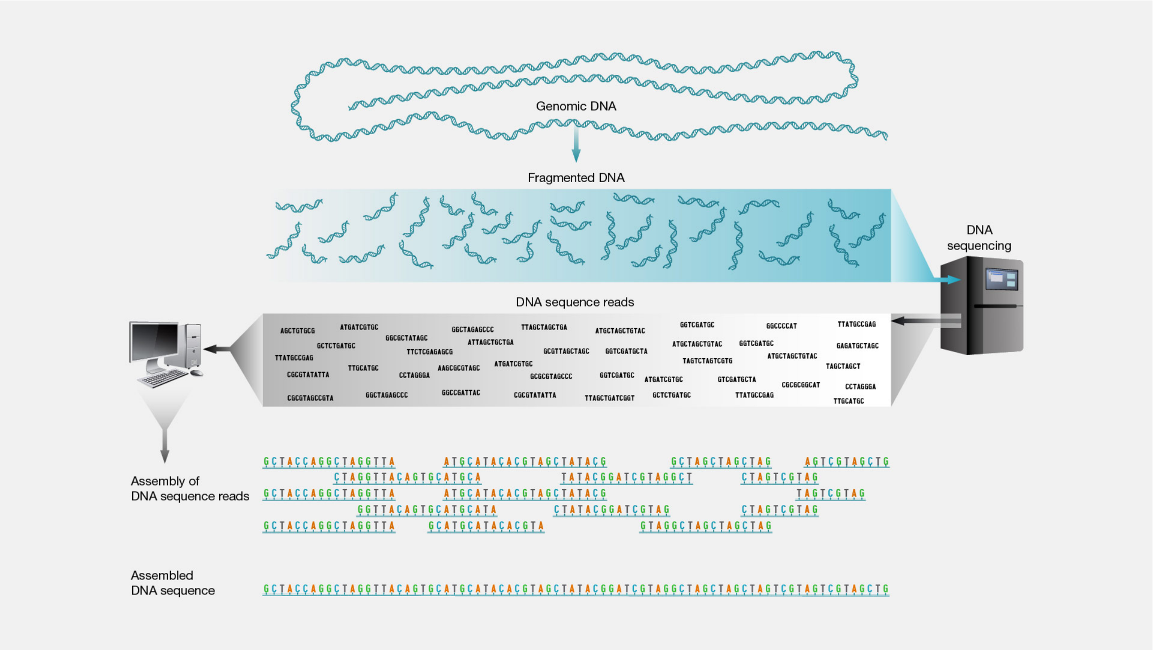

Figure 1.2: Shotgun sequencing visualized by means of a concept figure. After sample extraction, the DNA is fragmented, sequenced and assembled based on overlapping reads to recreate the initial strand in its entirety (Figure Courtesy: National Human Genome Research Institute) [41].

NGS is an umbrella term encompassing different forms of high throughput sequencing and has been described in detail since its inception [85]. In brief, the procedure is as follows. First DNA/RNA is isolated from samples and randomly fragmented into short sequences using physical or enzymatic methods. Second, these sequences are prepared by ligating adapters to both ends of the strand to allow for duplication via the polymerase chain reaction. Once a sufficiently large amount of duplicated fragments are present, the sequencing step can begin. After collecting all base reads, the numerous shotgun sequences are assembled into the original sequence by identifying overlaps between different reads. This results in a full recovery of the original sequence and avoids targeting solely specific areas of the DNA, as in microarrays [85]. Figure 1.2 provides a simplified visual representation of the method. Slight modifications to the experimental procedure enable quantification of somatic, constitutional, and epigenetic alterations.

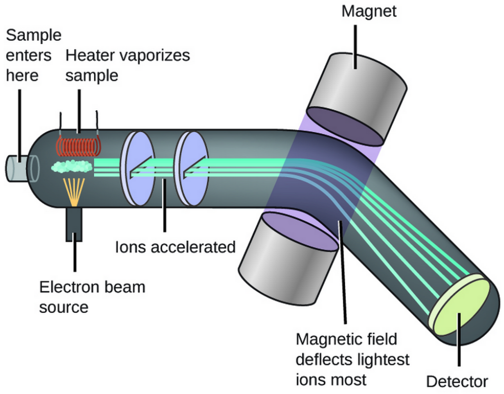

Figure 1.3: Mass Spectrometry explained by means of a concept figure. The highly concentrated sample is injected, vaporized and accelerated through a drift free region until it hits the detector. A magnet is used to filter ions that are outside the m/z region of interest. Ions that are too light are deflected against the bottom of the tube while ions that are too heavy will collide with the top of the tube [34].

Data collection for the metabolomic and proteomic landscape is often facilitated by mass spectrometry (MS) [49]. The experimental procedure begins with the extraction of proteins from the biological sample of interest followed by digestion into smaller peptide fragments with the help of enzymes or physical force. To increase sensitivity, the resulting fragments are enriched (highly concentrated) via methods such as solid-phase extraction. The solution of peptides can be injected into the mass spectrometer where the fragmented proteins are ionized and accelerated using a magnetic or electrical field, subsequently separating based on their mass-to-charge (m/z) ratio (Figure 1.3[1]). The further (shorter) a polypeptide travels, the lighter (heavier) it is [101]. The first instrument available at larger scales was the Time-Of-Flight (TOF) mass spectrometer in the 1960s. TOF-MS works by measuring the time required for a single ionized compound to travel through a drift-free, tubular region and combine the result with the strength of the electric field and length of the travel path to estimate a compound's molecular mass and relative abundance [24]. Although still found in labs around the world, new methods such as the Quadrupole MS or Orbitrap MS (founded on the same underlying principles) have started to replace TOF instruments and provide higher mass accuracy and resolving power [75].

[1] Access for free at https://openstax.org/books/chemistry/pages/1-introduction. Unmodified according to CC BY 4.0.

As MS instruments solely separate based on a compound's m/z ratio, it is common to directly couple techniques such as liquid chromatography (LC) or gas chromatography (GC) with mass spectrometers (LC-MS/GC-MS). This extension of MS enables the separation based on retention times in the chromatographic process. Thus, two peptides with a similar m/z are now separable given their behavior in the chromatographic column varies. Other tactics to increase specificity of MS, given the large number of similar peptides, include tandem MS during which two (or more) stages of MS are applied independently. The choice of instrument is highly dependent on the research question and hypothesis [75].

Independent of the choice of instruments required to answer the research question, the resulting data matrix, $\mathbf{X}$, is of dimension m by n, where m is the number of samples and n is the number of features (genes, peptides, proteins, etc...). As -omics technologies continue to advance, it has become clear that data probing multiple levels of biological complexity can be integratively analyzed (multi-omics integration) to provide a more complete picture of biological processes accompanying a given pathophysiological change. Computationally, for single omics experiments, the columns of $\mathbf{X}$ will refer to a single omics type, whereas in multi-omics experiments the columns are usually pertinent to a variety of data types. Thus, in multi-omics integration $n_{\text{total}} = \sum_i^{n_{\text{omes}}} n_i$, where n_i is the number of features collected for the n^{th} omics type. Depending on the workflow, it is possible to transpose $\mathbf{X}$ into a shape of n by m.

A characteristic of the matrix, $\mathbf{X}$, is that it epitomizes big data which raises two fundamental questions, *i.e.*, what is a good dataset, and what biologically interesting information can be extracted from the data. Addressing these two questions requires significant computational developments to understand statistical significance and isolate (biological) information from noise given the diverse and heterogeneous nature of the data. I address these two questions in this thesis by introducing: (a) an algorithm for unsupervised, robust sample classification via projection pursuit using kurtosis as a projection index (kPPA) and coupling it to classification and regression trees (CART), and (b) a framework for fitting variance models based on replicate measurements devoid of biological significance to recapitulate systematic variation introduced in -omics data by the analytical measurement procedures.

In the context of -omics analysis provided here, the role of bioinformatics is, given a data matrix **X**, to explore the data for quantitative information that relates an observed biological phenomenon to the acquired measurements that comprise **X**. Unfortunately, the wide variety of distributions resulting from different –omics measurement platforms can cause analysis to be highly complex. This has led to the development of online tools and packages that aim to streamline analysis of multi-omics data. Most of these tools, nonetheless, assume flawless experimental design, high sample availability and significant biological differences between samples representing each phenotype. In reality, those assumptions cannot always be met. For example studies focused on rare diseases commonly do not have multiple samples; thus, making estimation of variance and subsequent analysis of statistical significance difficult. Additionally, diseases with marginal changes in the molecular makeup may go unnoticed due to the widespread use of variance as an estimator of difference.

Elaborating on problem a), this thesis aims to dislodge the common approach of unsupervised classification from reliance on variance as the quantity of interest (such as Principal Component Analysis). The impact of this reliance is highly significant in datasets with a large range in value-intensity, a common occurence in multi-omics data. High intensity signals overshadow low intensity signals due to their high variances. A common remedy involves auto scaling the data. However, this inflates noise; thus, making detection of biologically important clusters harder. Bioinformaticians require a method that enables robust, but unsupervised, clustering while allowing low intensity signals to contribute to biologically interesting solutions. This approach shall also provide feature importance, comparable to loadings in PCA, for interpretability of clusters. This concept is explored in Chapter 2.

The second part of this thesis addresses problem b). Given a single, biologically important LC-MS measurement, it is impossible to determine statistics of interest such as uncertainty or benchmarking values like the limit of detection (LOD). This reveals the need for an approach resulting in a variance function (VF) which allows the estimation of the missing statistics mentioned above. The function is estimated by collecting multiple sample measurements of a reference solution. Since replicates of LC/GC-MS experiments are subject to instrument (or sample) dependent changes in sensitivity, the fitting process requires proper scaling to account for different sensitives with the goal to avoid inflating variance. Various scaling methods have been used in prior applications which I argue to be unreliable due to inability to account for the error in the measurements. Thus, I propose a method composed of iterative scaling, based on Maximum Likelihood Principal Component Analysis (MLPCA), followed by fitting the VF. Chapter 3 and 4 explores the problem in more detail and suggests two approaches; one based on Frequentist statistics; one based on Bayesian statistics.

Chapter 2

Augmented Kurtosis-based Projection Pursuit: A Novel, Advanced Machine Learning Approach for Multi-omics Data Analysis and Integration

2.1 Introduction

Large-scale -omics data deriving from genomics, epigenomics, transcriptomics, etc., have revolutionized biological studies allowing many new hypotheses to be derived from these measurements. These data capture a systems-wide molecular overview of biological processes that enable screening for disease [77] [92], prognostic forecasting [77], discovery of biomarkers [21], disease subtyping [19] [65], drug repurposing [107] and so on. In principle, each -omics measurement provides a specific insight into a "layer" of functional biological organization where, *e.g.*, genomics data provide a systemic, organism-wide blueprint of the phenotype; transcriptomics offers insights into which proteins are likely to be expressed and so on. Many studies reported in the literature employ single-omics measurements that aim to decipher, *e.g.*, causes of known pathologies, refine pathological classifications or stratify their risk, and/or to select appropriate treatments. Conceptually, extending single -omics measurements to capture additional layers of biological complexity (multi-omics) should lead to the inference of significantly more valuable, system-wide information. Rationally, changes observed via gene expression measurements should have a quantifiable correlation with epigenomics measurements that indicate differential methylation at CpG islands (areas of high concentration of **C**ytosine and **G**uanine) around promoters or enhancers for those same genes.

Although integration and visualization of multi-omics data should yield significantly more robust biological insights, analyzing these data is a remarkable challenge. This is unsurprising because experimentally, -omics data arise from a variety of analytical sources with non-standard distributional characteristics, quality, linear dynamic range and (often) large differences between the number of variables per study. Although extensive reviews exist that catalogue limitations associated with multi-omics integration and visualization [68] [78] [90], consensus appears to be that high data dimensionality, missing values, imbalance in sample design, and data storage are the most pertinent. The Karakach Lab [89] and others [69] [113] [109] [88] have maneuvered these challenges and performed high level multi-omics data integration and demonstrated increased recovery of biological information from integrated data compared to data arising from individual sources. Smilde et al. [88] fused gas chromatography–mass spectrometry (GC–MS) and liquid chromatography–mass spectrometry (LC–MS) metabolomics data to increase the coverage of metabolites in a study to identify bottlenecks in phenylalanine production in E. coli NST74. Nam et al. [69] described an approach for integrating transcriptomics and metabolomics for breast cancer biomarker identification in which t-statistics were used to determine differentially expressed gene transcripts from breast cancer vs normal subjects. Van den Berg et al.[103], in a simulation study, devised a maximum likelihood method for integrating functional genomics data given that such data exhibit different noise characteristics. Most recently, Heo et al. [45], reviewed the literature and offered a compendium of computational frameworks that have been used for multi-omics integration with a specific focus on cancer research. Many of these approaches are, nonetheless, based on Machine Learning (ML) principles that are highly influenced by sample sizes, data distributions, and rely on signals that are above a large threshold of signal-to-noise ratio (S/N). In many applications, the latter problem is avoided by pre-selecting a set of highly variable features, implicitly focusing the analysis on the most dominant signals at the expense of the low intensity ones. The Karakach Lab has recently demonstrated that low intensity signals in a biomolecular context can have significant biological impact and developed error modelled gene expression analysis (EMOGEA) as a framework that ameliorates dependence on dominant signals for data classification [8].

There continues to be no consensus on the *de facto* approach for performing multi-omics analysis. However, it can be reasonably argued that any methods for integrating multi-omics data must identify biologically meaningful class (dis)similarities and the concomitant features to allow high-risk, high-reward -omics studies to be conducted. For my purposes, I define high-risk, high-reward -omics studies as those for which: (a) the expected biological differences are subtle, and/or (b) pertinent biological information is embedded in the low intensity signals. Whereas experimentalists overcome the former by performing experiments for which large effect sizes are anticipated, there is no mechanism (other than EMOGEA [8]) to comprehensively deal with the latter to the best of our knowledge. A common strategy is to exclude the low intensity signals from analyses as they are assumed to be dominated by noise.

Here, I propose kurtosis-based projection pursuit analysis (kPPA) coupled to Classification and Regression Trees (CART), as an approach to deal with -omics data exhibiting small effect sizes and low feature intensities. I specifically employ kPPA-CART to integrate and visualize data deriving from the most common -omics platforms: transcriptomics by RNA-seq, epigenomics by DNA methylation chips or Bisulfite sequencing and proteomics by reverse phase protein arrays or mass spectrometry (MS). At the outset, I emphasize that kPPA is an “unsupervised” data exploration approach that finds patterns in input data without *a priori* knowledge of class membership, unlike “supervised” methods that train a model using labeled training data with known class membership. This absence of labels avoids biasing the model to find ubiquitous information. The output of kPPA are projections of the original samples into “interesting” directions which, when plotted against each other will depict clustering of similar samples. I augment kPPA’s clustering with Classification and Regression trees which takes, as an input, the kPPA cluster information to perform a quasi-supervised classification and decipher feature importance. While kPPA shows remarkable sample clustering, I demonstrate that the addition of CART allows for extraction of data that contain meaningful biological information. I apply this approach to two datasets. First, I use multi-omics data consisting of transcriptomics by RNA-seq and proteomics by MS reported in Takemon et al.[97] in which the molecular changes that take place in the kidney during the aging process were measured. Using these data as a base, I generate additional artificial data where I simulate the effects of varying noise and effect sizes and benchmark methods for multi-omics analysis. Later I employ kPPA-CART to the experimental data to reveal additional novel insights. Second, I model Breast cancer data from The Cancer Genome Atlas (TCGA) [17] to show the extent to which this approach is superior in identifying cancer subtypes from both individual -omics platforms and through high-level integration.

The goals of the work presented here are three-fold. First, I show the importance of the kPPA-CART approach for "distribution agnostic" data classification that is specifically suited for -omics data with a large dynamic range such as transcriptomics by RNA-seq or proteomics by MS. I benchmark our kPPA against common multi-omics data integration and visualization approaches using: (a) the well-behaved multi-omics data from Takemon et al. [97] to study the relationship between kidney function and age among diversity outbred male and female mice using transcriptomics and proteomics, and (b) an artificial dataset simulated based on the same Takemon data to investigate performance of kPPA-CART given varying effect sizes. I then employ the Silhouette Coefficient [10] as measure of model performance in each case. Second, I use the TCGA data to answer the crucial question of whether I can classify cancer samples into distinct molecular subtypes given the data. Subsequently given the subtypes, can I then: (1) stratify their risk and forecast their prognosis, and (2) infer the molecular mechanisms that uniquely drive each subtype. Finally, I provide a web-based application to allow experimentalists to explore -omics data using kPPA-CART, allowing the flexibility for feature engineering, analysis of the -omics data (individually or integratively) followed by functional analyses, and generation of publication ready figures and reports.

2.2 Methods

Mathematically, -omics data are exceptionally heterogeneous. They exhibit signals that cover a large linear dynamic range, with unique distributional characteristics, nonuniform measurement error structures, and biological variations. For example, transcriptomics measurements acquired using RNA sequencing technologies (RNA-seq) follow an over-dispersed Poisson distribution (Negative Binomial), while those acquired using microarray technologies and MS-based proteomics measurements exhibit a log-normal one [57]. These factors make data integration computationally challenging requiring transformations, filtering, imputation of missing values, normalization, and/or scaling prior to downstream analyses to (a) limit the influence of outliers, (b) reduce the number of features considered, and (c) prevent data from one measurement platform from dominating the results of the integration.

Several computational methods and packages exist to address some of these challenges to enable integration of multiple of the -omics measurements, including *mixOmics* [82], *MoCluster* [64], MultiOmics factor analysis (*MOFA*) [6] and Multiple co-inertia analysis (*MCIA*) [7]. *mixOmics* is an R package containing a collection of computational tools that classify sample groups, identify discriminant features, and predict the class membership for new samples. Methods within this package include DIABLO (Data Integration Analysis for Biomarker discovery using Latent Components) which has the mathematical underpinnings of PLS-DA (Partial Least Squares - Discriminant Analysis), a supervised clustering algorithm that is based on identifying latent variables maximizing class separation [87]. In addition, *mixOmics* extends the Generalized Canonical Correlation Algorithm (GCCA) using sparsity constraints in which linear combinations of variables with high correlations are identified for class separation [99]. *MoCluster* applies multiblock multivariate analysis to define latent variables that represent shared patterns across -omics datasets. It then uses distance-based measures to quantify the separation between genes and clusters them accordingly and is sensitive to the choice of the distance metric.

The packages mentioned here comprise a mix of supervised and unsupervised classification methods and our kurtosis-based projection pursuit analysis (kPPA) falls under the latter. Merits for each data classification approach have been extensively discussed in the literature [18] [102] [100]. However, a clear advantage for unsupervised methods is that they are not only less sample intensive than their supervised counterparts, but they are also less prone to overfitting, especially when used by non-experts. For example, Westerhuis et al. [112], showed that although PLS-DA requires enough samples to allow model development, validation and prediction, many applications of this approach in the literature did not include the validation and prediction steps, leading to unreliable conclusions. Most -omics studies are sample-limited due to cost, unavailability of samples (e.g. clinical samples), or other ethical factors, making unsupervised classification approaches the best strategy for hypothesis generating analyses under these circumstances. The most widely used unsupervised methods include principal components analysis (PCA; 73,024 PubMed hits), independent component analysis (ICA; 42,089 PubMed hits), hierarchical cluster analysis (HCA; 21,2865 PubMed hits), and multi-dimensional scaling (MDS; 17, 495 PubMed hits), among others [18] [102] [100]. Although from the inception of -omics approaches in biomedical research PCA has been the dominant method for visualizing high-dimensional data in lower dimensional spaces, it is based on maximizing the variance along the projection vectors, a drawback that limits its effectiveness in separating classes. The Karakach Lab has recently shown, for example, that high intensity signals will dominate the output of PCA and offered a solution that required measurement uncertainty to be incorporated in the analysis [8]. This problem can also be circumvented using projection pursuit (PP) analysis, which uses different criteria to identify projection vectors. While there are examples of the application of this technique to other fields [39] [28], it is not nearly as widely applied as PCA, ICA and HCA, in part due to the requirement of a high sample-to-variable ratio and balanced, binary data sets. These issues are underlined by inaccessibility of a user-friendly format that can be implemented by experimentalists. Many authors (see citations below) have previously described efficient and straightforward algorithms to carry out PP analysis, allowing this tool to be readily adapted. This algorithm is based on minimizing kurtosis as the preferred projection index for our multi-omics applications and I briefly highlight the base algorithm for optimizing this projection index and refer the reader to Daszykowski et al. [28], Croux et al. [26] [27], Hou et al. [47], Driscoll et al. [30] and Wentzell et al. [111] for more detailed descriptions.

At the outset, I stated that like PCA, PP is a subspace estimation method that seeks to identify “interesting” projections in a low-dimensional subspace that can reveal the latent information in these data. Unlike PCA where the directions of the greatest variance in the data are assumed to represent this information, PP does not unambiguously define how to determine what is interesting, instead relying on the experimentalist to explore projections (including variance) that can maximize information recovery [29]. PP can be mathematically formulated as follows. Let $\mathbf{X}_{m \text{ x } n}$ be a matrix of measurements, e.g., representing n genes transcripts measured for each of the m samples in a study. In the general principle of subspace estimation, $\mathbf{X}_{m \text{ x } n}$, can be represented as a product of two matrices of equal rank but lower dimension such that:

$$\mathbf{X} = \sum_{i=1}^{p} \mathbf{t_i}\mathbf{p}_i^T + \mathbf{E} \tag{2.1}$$

where the column vectors $\mathbf{t}_i$ and $\mathbf{p}_i$, represent scores and loadings respectively. Loadings explain contributions of the individual data variables to construction of each latent factor. In this equation $\mathbf{E}$ is the residual matrix, *i.e.*, part of the data that is, in principle, devoid of information and is unexplained by a model of p factors.

In the base PP framework suggested by Croux [26], identifying the latent variables in Eq. 2.1 follows two critical algorithmic steps. First, all the row vectors in $\mathbf{X}$, $\mathbf{x}_i$, are normalized to unit length to represent the set of all possible directions in the data. The data are then projected onto these directions to form a set of latent factors, following which a projection index (PI) is calculated for each of the latent factors. The factor with the highest (lowest) projection index is chosen as the direction of the data that contains the most information. Second, new directions are found in the space of the data deflated (Step 3c, Table 2.1) by the first most important latent factor using the same procedure as the first step, to ensure the orthogonality of projection pursuit features (PPFs). This procedure continues until a pre-determined number of directions is achieved. Table 2.1 provides the algorithmic details for construction interesting projections in this base framework.

In this work, I employed kurtosis as the projection index (PI) and implemented the approach described in Hou et. al. [47] [111] for optimizing the PI, which differs from the base algorithm. The mathematical details for this optimization are provided in references [47] [30] [111], importantly using the quasi-power method to minimize kurtosis, k. Illustratively, if one considers a sphered data matrix $\mathbf{X}_{m \text{ x } n}$ (m samples by n features), the univariate kPPA algorithm begins by randomly selecting a unit length projection vector, $\mathbf{v}_0$ (n x 1), as a starting point of the quasi-power method. An iterative procedure is then used to update the projection vector until kurtosis is minimized. The projection vector at each iteration is updated such that:

$$\mathbf{v}_{k+1} = \left[\sum_{i=1}^{m} (\mathbf{v}_k^T \mathbf{x}_i \mathbf{x}_i^T \mathbf{v}_k^T)(\mathbf{x}_i \mathbf{x}_i^T) \right]^{-1} (\mathbf{X}^T \mathbf{X}) \mathbf{v}_k \tag{2.2}$$

where $\mathbf{x}_i$ is the i^{th} row vector of $\mathbf{X}$, and the product $\mathbf{x}_i \mathbf{v}_k$ is the score, $\mathbf{t}_i$, (latent factor) for sample i of m, given the current projection vector. Kurtosis is subsequently calculated using this latent factor such that:

$$k = \frac{\frac{1}{n} \sum_{i=1}^{n} (\mathbf{t}_i - \bar{\mathbf{t}})^4}{\left(\frac{1}{n} \sum_{i=1}^{n} (\mathbf{t}_i - \bar{\mathbf{t}})^2 \right)^2} \tag{2.3}$$

where is $\bar{\mathbf{t}}$ the mean of all the m latent factors calculated. It is to note that Equation 2.2 will converge to a minimum; however the found minimum is not guaranteed to be global [47].

Step 1	Sphere the data (mean center and scale to unit variance)
Step 2	Construct a matrix, $\mathbf{A}$, containing normalized rows of $\mathbf{X}$ such that: $\mathbf{p}_i = \frac{\mathbf{x}_i}{\\|\mathbf{x}_i\\|}$, where $\mathbf{x}_i$ is the i^{th} row of $\mathbf{X}$ and $\\| \cdot \\|$ is the Euclidean norm.
Step 3	a. project objects on all possible directions from Step 2 such that $\mathbf{t}_i = \mathbf{X}\mathbf{p}_i^T$ and calculate the projection index for each direction. b. find the directions that minimize (maximize) projection index. c. Deflate the original data and create a new data space $\tilde{\mathbf{X}}$ such that $\tilde{\mathbf{X}} = \mathbf{t}_i \mathbf{p}_i^T$ and perform a Gram-Schmidt orthogonalization from $\mathbf{X}$.
Step 4	Terminate the procedure if the number of interesting directions found is lower than p, (the user pre-determined number of PPFs to extract) otherwise go to step 2.

Table 2.1: Base algorithm for Projection Pursuit.

2.3 Results

2.3.1 Implementation

Implementing kPPA-CART proceeds as a nested, $N_i(\forall\ i = 1, 2, \ldots, p)$, iterative procedure with the following steps. For a data matrix $\mathbf{X}_{m \text{ x } n}$ consisting of m samples and n features (e.g., gene IDs, Protein IDs, methylation sites etc.), I first select f random features to form a smaller matrix, $\tilde{\mathbf{X}}_{\text{m x f}}$, where f is chosen to represent the smallest possible subset of features presumed to contain the most meaningful biological information to cluster the data. Second, I employ multi-dimensional scaling (MDS) to denoise these data. The number of components in MDS, k_{MDS}, is chosen to be approximately one fifth the number of samples due to limitations of kPPA in cases of fat data. Following this, I obtain the latent factors, $\mathbf{t}$, that minimize kurtosis via an iterative procedure that updates the projection vector until convergence. Because this is an unsupervised classification, k-means clustering is employed on the two-dimensional scores to assign arbitrary clusters to samples which are then fed into a pseudo-supervised random forest (RF) classification [13]. The number of clusters to be fit in k-means is determined based on experimental expectations. This allows a feature importance score to be extracted following training of the random forest classifier. Finally, the index and importance score of the features used in the RF class prediction are stored in a vector $\mathbf{v}$ as a value pair, the features selected in the first step returned to the matrix $\mathbf{X}$, and the whole procedure is repeated p times. After finishing p iterations, the final kPPA clustering is generated based on the f features that provide the highest average importance score across all iterations. The output of kPPA-CART is a set of projection vectors that best cluster the data (similar to principal component analysis: PCA scores) and a subset dataset $\mathbf{X}_{\text{small}}$, comprising the original data with all samples but only containing the f features with the highest average importance scores from all p iterations. A schematic of this procedure is shown in Figure 2.1 and a web best implementation can be accessed directly here (https://intmove.vercel.app/).

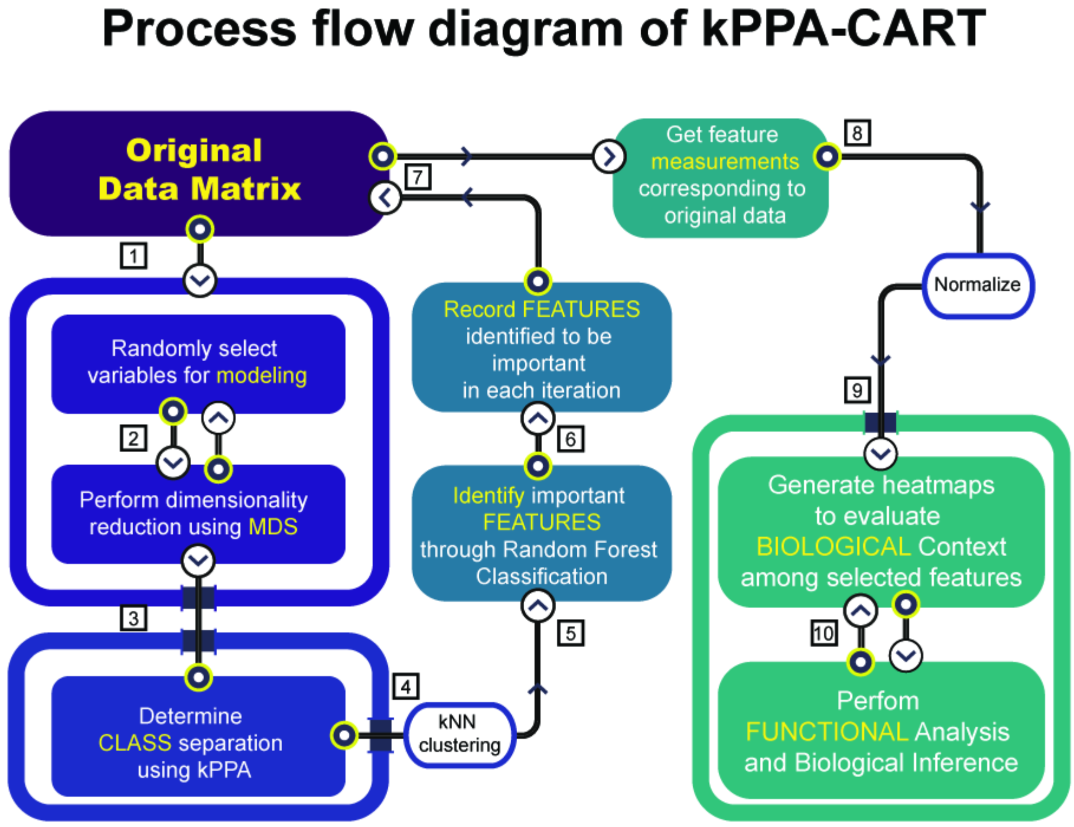
Process flow diagram of kPPA-CART
Original Data Matrix
1
Randomly select variables for modeling
2
Perform dimensionality reduction using MDS
3
Determine CLASS separation using kPPA
4
kNN clustering
5
Identify important FEATURES through Random Forest Classification
6
Record FEATURES identified to be important in each iteration
7
Get feature measurements corresponding to original data
8
Normalize
9
Generate heatmaps to evaluate BIOLOGICAL Context among selected features
10
Perfom FUNCTIONAL Analysis and Biological Inference

Figure 2.1: Workflow for kPPA-CART indicating the iterative process of implementing kurtosis-based projection pursuit analysis. In the first instance, a preset number of variables is selected from the data matrix **X** and used to find projections minimizing data kurtosis after application of MDS. Class membership is then determined via k-means clustering. Using these class IDs, a pseudo-supervised random forest classification is performed to determine feature importance for the first iteration. The most important features are recorded, and the iterative process proceeds with the selection of second set of random features that are then put through MDS, kPPA, k-means and random forest classification. After all iterations are run, the features that have the most importance across all iterations are selected for further biological functional validation. Within this framework, the set of variables for each iteration and number of iterations must be chosen such that each feature has a chance to be used at least 10 times in the model development. This number can be determined via a formulation similar to a "coupon collector's problem" [33].

2.3.2 Benchmarking

The Takemon multi-omics dataset [97] includes the transcriptome and the proteome profile of kidney tissues harvested from genetically diverse mice at different ages. I employ this dataset to demonstrate the conditions under which standard exploratory analysis approaches become inadequate. In many analyses, PCA or MDS is typically performed either on an entire dataset or on a pre-selected set of highly variable features. The unfortunate outcome is that low intensity signals, even if they exhibit biological significance, will be dominated by the high intensity ones, and are hardly picked up as contributing significantly to the biology. To demonstrate this, I selected a percentage: 3%, 5%, 10%, 13%, 15%, 18%, 20%, 25% and 30% of the lowest variance features from the Takemon proteomics and transcriptomics data separately and analyzed them integratively using methods recently benchmarked by Cantini et al. [18], with the goal of evaluating their performance in comparison to kPPA-CART. To further show the effect of selecting highly variable features, I performed the reverse analysis where I selected a percentage (3%, 5%, 8%, 10%, 12%, 15%, 18% and 20%) of the highest variance features from the same data (proteomics and transcriptomics separately) and repeated the integrative analyses. Figure 2.2A and B, show the outcome of these benchmarking analyses respectively. The methods by Cantini et al. [18] are referred to as joint Dimensionality Reduction (JDR) methods because they are used to analyze data that have been integrated via row- or column-wise concatenation (high-level data fusion) of individual -omics measurements. The methods in Cantini's work rely on different mathematical formulations to identify latent factors in the data, including PCA, Factor Analysis (FA), independent component analysis (ICA), co-inertia analysis, Gaussian latent model, matrix-tri-factorization, Non-negative Matrix Factorization (NNMF), or canonical correlations analysis (CCA). I reasoned that because these methods continue to receive the most attention in the literature, they would form the best basis against which kPPA-CART would be benchmarked. Specific methods I tested include: Multi-Omics Factor Analysis (MOFA) [6], iCluster [55], Integrative NMF (IntNMF) [20], Multiple Co-Inertia Analysis (MCIA) [7], IPCA [114], an approach that combines ICA and PCA for multi-omics data analysis, MDS, Joint and Individual Variation Explained (JIVE [60], which I denote as PCA since they are practically identical).

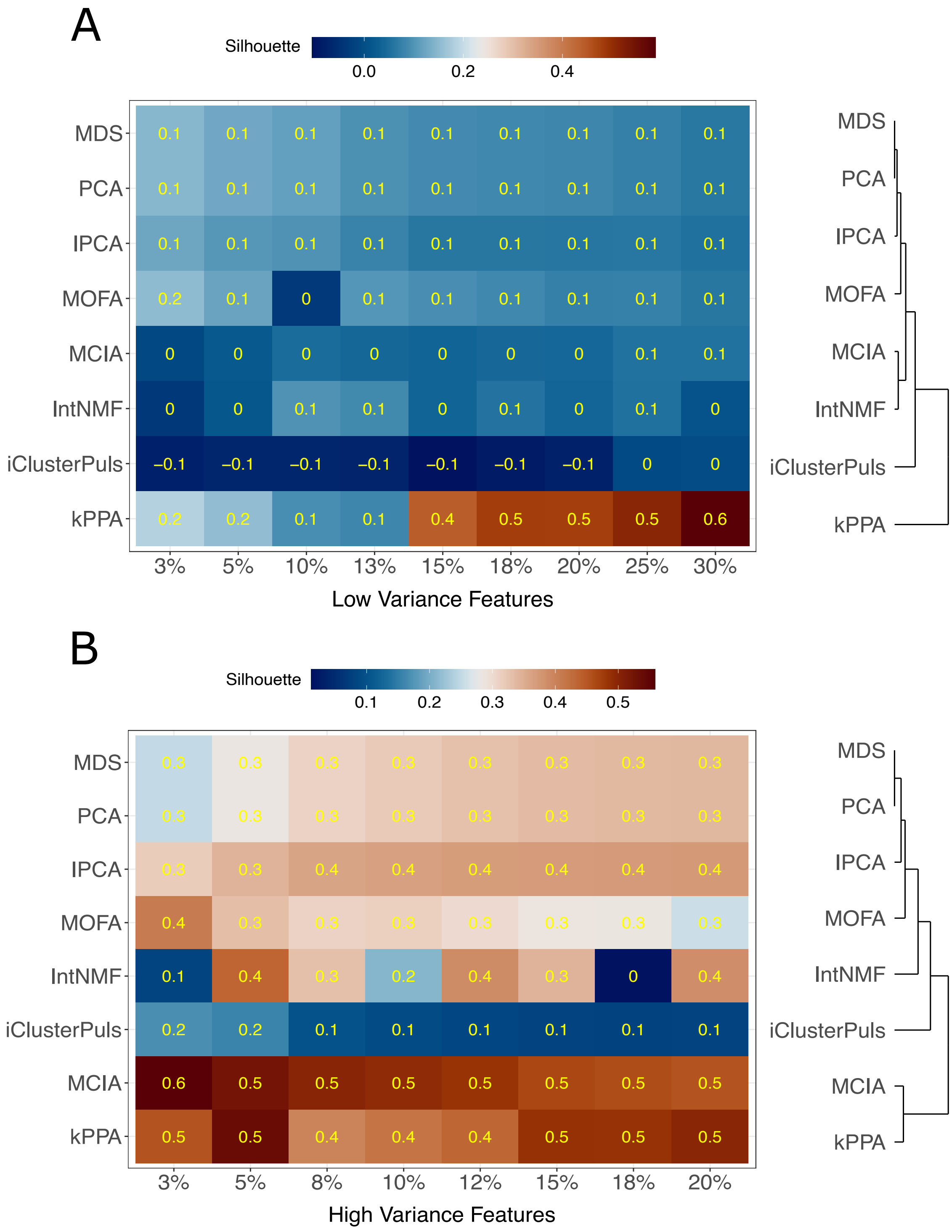

Figure 2.2: The sillhouette coefficient for each method of interest as a function of the percent of: **A)** features exhibiting the lowest variance. **B)** features exhibiting the highest variance. In **A)**, most methods are not able to capture significant sample classification using low-variable features, but at 15% of the lowest variable features, kPPA-CART can begin to model these differences. For highly variable features shown in **B)**, we see that with only the top 3% of the features, class similarity is evident.

Our results in Figure 2.2A-B are based on the Silhouette Coefficient [10] and show that low intensity (low variable features) remain difficult to model using standard tools. Specifically, I show that kPPA-CART begins to show reasonable clustering at about 15% for low variance features (Figure 2.2A) and at 3% for high variance features (Figure 2.2B). While other methods provide equivalent quality of separation for high variance features, the figures provide evidence that that kPPA-CART reveals meaningful data classification even when only 15% of the least variable features are used. No other method can provide a similar performance, even in the case of using up to 30% of the least variable features of each -omics type. Figure 2.3A shows sample clustering obtained with PCA, IntNMF, MOFA and kPPA using 15% of the least variable features. In this, I see that kPPA-CART can separate the data based on age and sex, while PCA minimally shows the effect of sex on these data. Moreover, PC1 vs. PC2 do not show any observable differences perhaps because of potential outliers; providing another area where kPPA-CART shines. MOFA and IntNMF capture the same information and potential outliers as PCA. Figure 2.3B provides the separation for the same methods given the top 3% variable features. While PCA and MOFA show acceptable clustering for age and sex, kPPA-CART still performs best. IntNMF struggles to provide any clear separation based on age. Interestingly, PCA, MOFA and IntNMF do not show the same outliers that were present across PC/Component 1 in the bottom 15% of low intensity signals.

To further benchmark kPPA-CART, I simulated data with differential effect sizes to capture cases where the differences between study groups is small. I changed this effect size to vary from 0.2 to 5 (0.1, 0.3, 0.5, 0.7, 0.9, 1.1, 1.4, 2, 5) using data from thirty (30), six (6) month-old female diversity outbred mice of which a random set comprising 100 features were selected to be differentially expressed (50 features were chosen from proteomics data and another 50 were chosen from transcriptomics data). To ensure that this effect size is not the same across all the features, I added noise drawn from a normal distribution to the mean effect size for the block of features arbitrarily designated as up- or down-regulated. I subsequently tested the methods described in Cantini's publication [18] to identify clusters from these data.

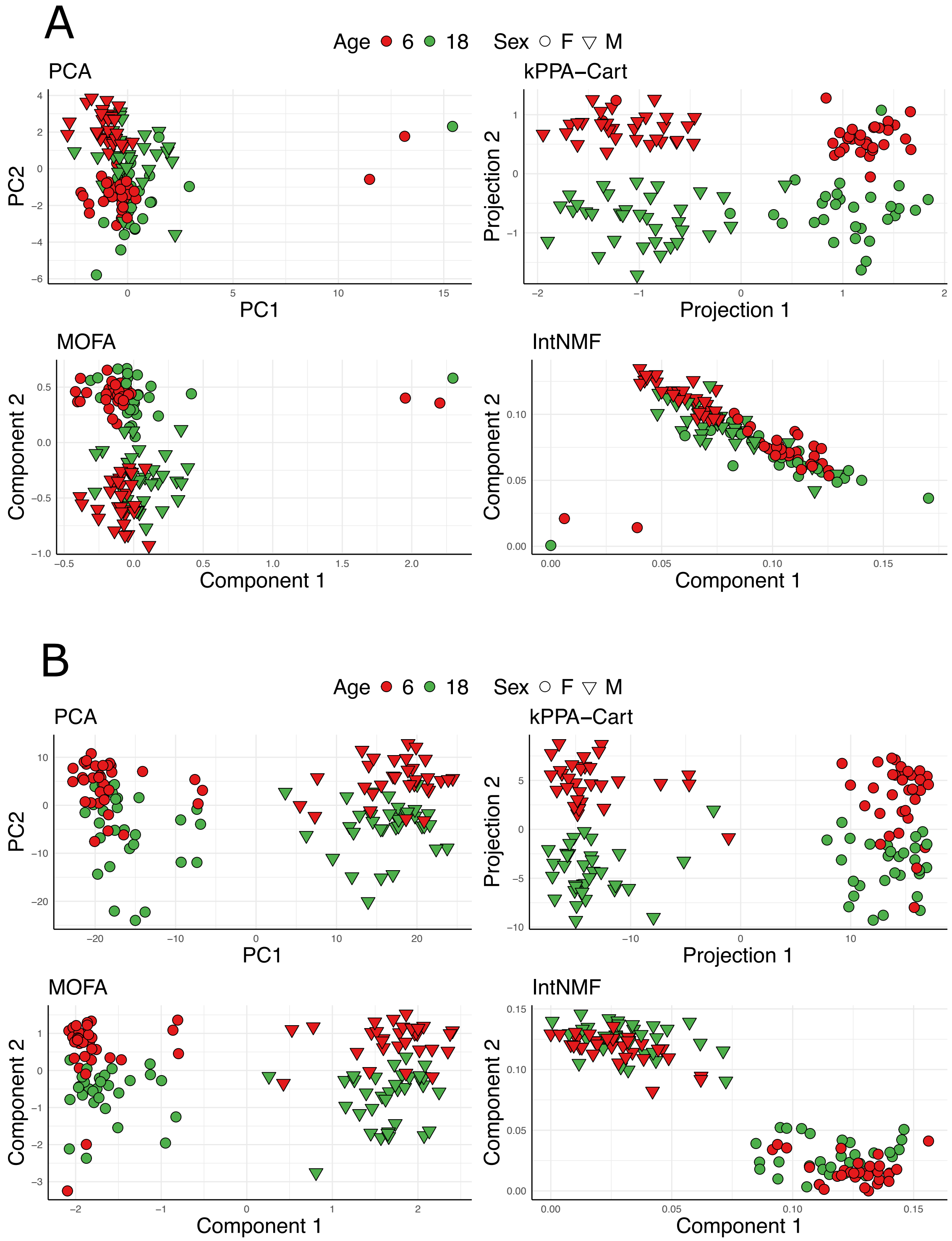

Figure 2.3: Clustering capabilities of PCA, kPPA-CART, MOFA and IntNMF for: **A)** the bottom 15% of least variable features. **B)** the top 3% of most variable features. Within both subplots kPPA-CART outperforms all of the provided methods.

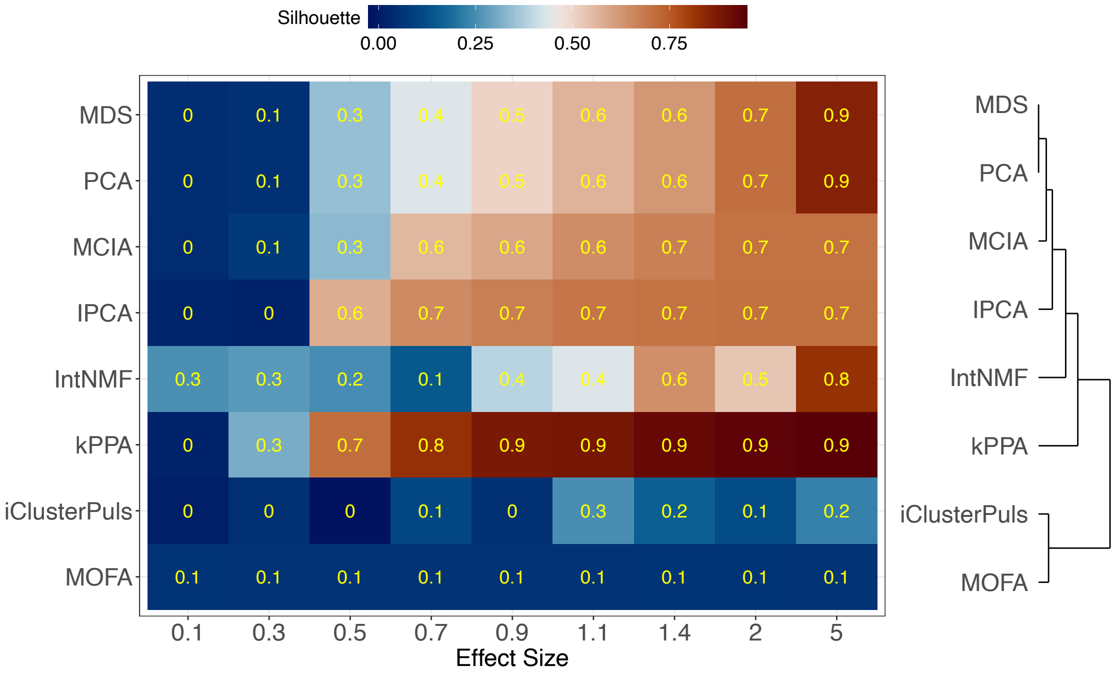

Figure 2.4: The sillhouette coefficient for each method of interest as a function of effect size. The effect size describes separation between two groups. While kPPA-CART provides a high clustering (SC=0.9) at an effect size of 0.9, no other method shows a similarly high SC until an effect size of 5.

In Figure 2.4 I show the sillhouette coefficent (SC) determined by each method as a function of effect size which indicates that kPPA-CART is the top performer. In detail, over the range of effect sizes, it is evident that kPPA-CART almost consistently achieves the highest SC. For low effect size, IntNMF seems to provide some clustering but does not show the expected increase in clustering ability with an increase in effect size. The higher the effect size, the better variance-based methods perform. Moreover, given knowledge of the features that are differentially expressed, the rate at which kPPA-CART recovers these features (true positive rate) can be calculated. The true positive rate steadily increases reaching more than 90% for any effect size that is larger than 0.7 and reaches 100% for effect sizes of 2 and larger. This provides significant evidence that kPPA-CART is capable of detecting biological features useful in understanding the phenotype of the organsim.

2.3.3 Modelling the Aging Process in Mouse Kidneys

Using the Takemon data to simulate artificial measurements by modifying parameters such as effect size and variable features has allowed us to mimic common challenges encountered in -omics research. The original experiment consists of important biological information obtained via a multi-omics experiment. Using diversity outbred mice, the fundamental objective of this study was to understand the molecular changes that take place in the kidney during the aging process, and if these are different between sexes. MS-based and RNA-seq data were acquired from flash frozen kidneys of animals in 3 age categories: 6 weeks (n = 30 males and n = 33 females); 12 week (n = 31 males and n = 31 females); and 18 weeks (n= 34 males and n = 29 females).

Following standard pre-processing steps for RNA-seq and MS-based proteomics, the data were analyzed starting with the individual datasets (proteomics followed by transcriptomics) and subsequently combining them to increase the information recovery associated with multi-omics integration. Without subdividing the data into age categories, Figure 2.5A(i) shows PCA results for proteomics with two clusters along PC2 that largely separate male from female mice regardless of age. When I analyze the data comparing two ages at a time (i.e., 6- and 12-week-old mice; 6- and 18-week-old mice, and 12- and 18-week-old mice), PCA gives similar classification patterns where sex is the dominant factor, while age cannot be modeled (Figure 2.5A (ii-iv)), with outliers in two of the categories. Figure 2.5B shows kPPA analysis of the same data with a much clearer separation between the two sexes; however, an apparent age separation is only possible when comparing 6 versus 18 months old mice (Figure 2.5B(iii)). For a similar analysis with RNA-seq data, the results indicate a weaker class dissimilarity in these data as shown in Figure 2.5C and D, with no age separation with either PCA or kPPA.

This was followed by integrating the two datasets, starting with the subset consisting of 6- and 18-week-old mice for which proteomics measurements showed a separation via kPPA analysis. Figure 2.6 show the classification of integrated data using PCA (left panel) and kPPA (right panel), while Figure 2.7 is a boxplot of four of the most important (protein and RNA) features (genes) for this classification. For

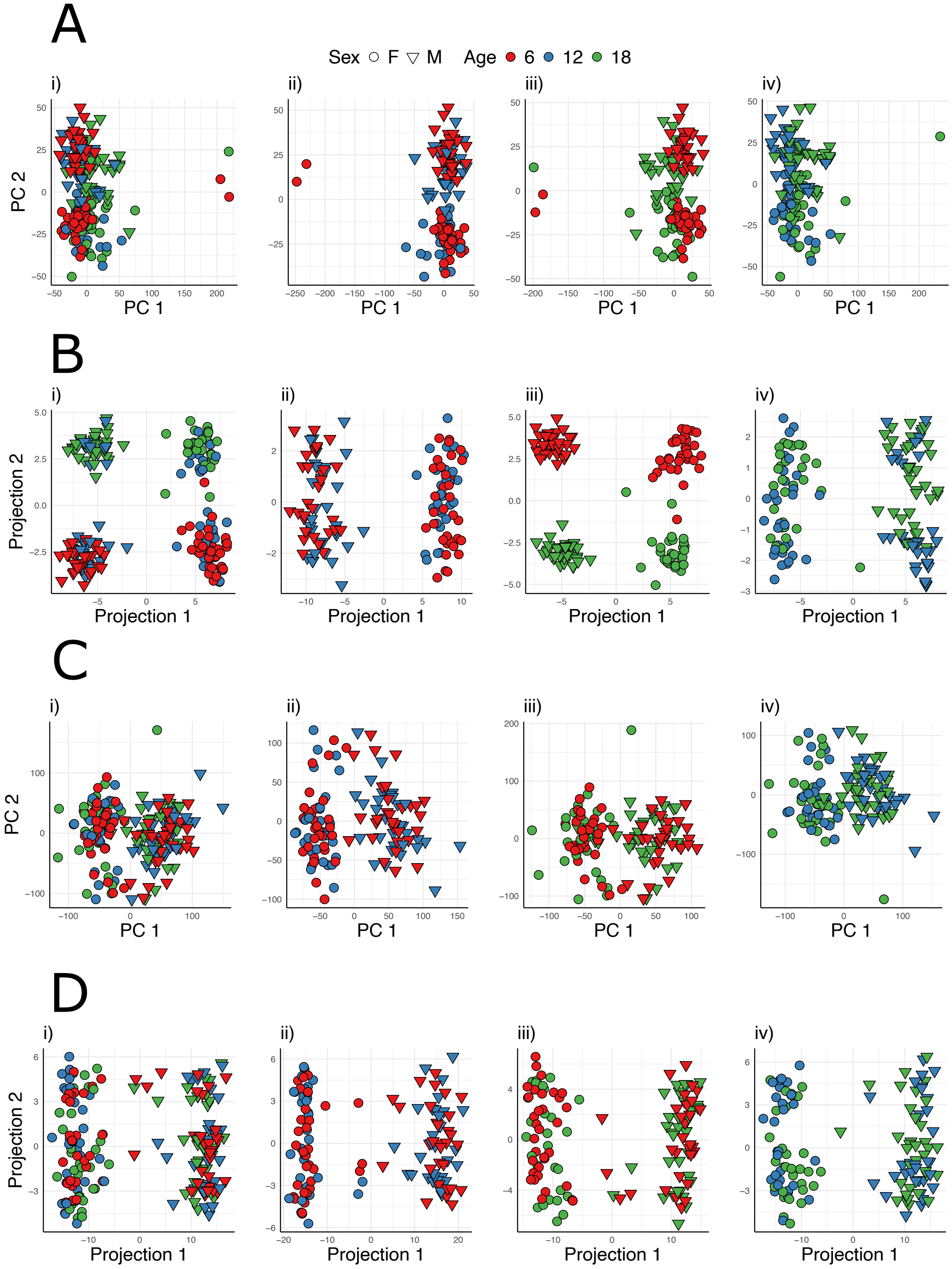

Figure 2.5: Comparison of clustering ability of PCA and kPPA-CART applied to the proteomics and transcriptomics dataset by Takemon et al. [97] **A)** PCA applied to proteomics. **B)** kPPA-CART applied to proteomics. **C)** PCA applied to transcriptomics. **D)** kPPA-CART applied to transcriptomics.

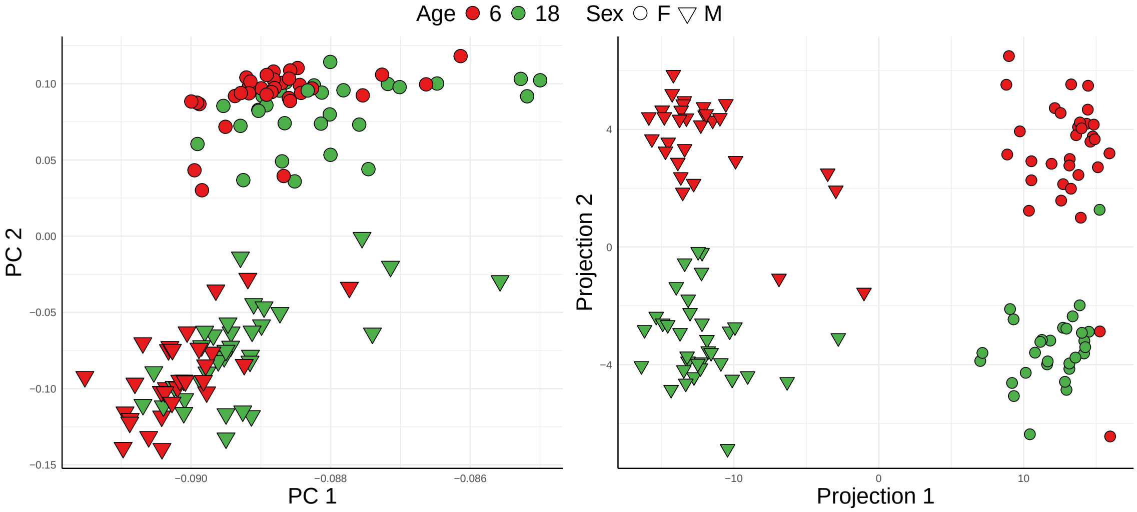

Figure 2.6: Integrative analysis of the Takemon dataset for 6- and 18-week-old mice. The left panel shows the resulting PCA separation. The right panel provides the clustering of kPPA-CART.

example, in Figure 2.7A it is apparent that at 6 weeks, there is no difference in the abundance of Cyp2d26 and Coenzyme A synthase (Coasy), among female mice, while for male mice the abundance of these two proteins is vastly different. This observation is consistent up to 18 weeks. In contrast, there is an age-related change in the relative abundance of Slc7a7 compared to Cyp2d26 where the male and female mice at 6 weeks show a small difference in the abundance levels of these two proteins at week 6, but this difference increases at week 18. Consistent with previous results, the top 4 most important RNA transcripts only show difference in sex but not age. Finally, Figure 2.8 is a heatmap of features selected by kPPA after passing through the classification and regression trees (CART) protocol. These figures indicate that kPPA-CART indeed identifies features with biological (rather than simply mathematical) importance.

2.3.4 Modelling Breast Cancer Multi-omics Data

Next, the kPPA-Cart approach was applied to well-established experimental data that simultaneously answer a known question in biomedical research. To provide context, I chose to focus on Breast Cancer (BC) as it affects a disproportionate number of women world-wide (1 in 8) with most deaths occurring among black women with metastatic breast cancer (MBC) [115]. Already, significant effort has gone into

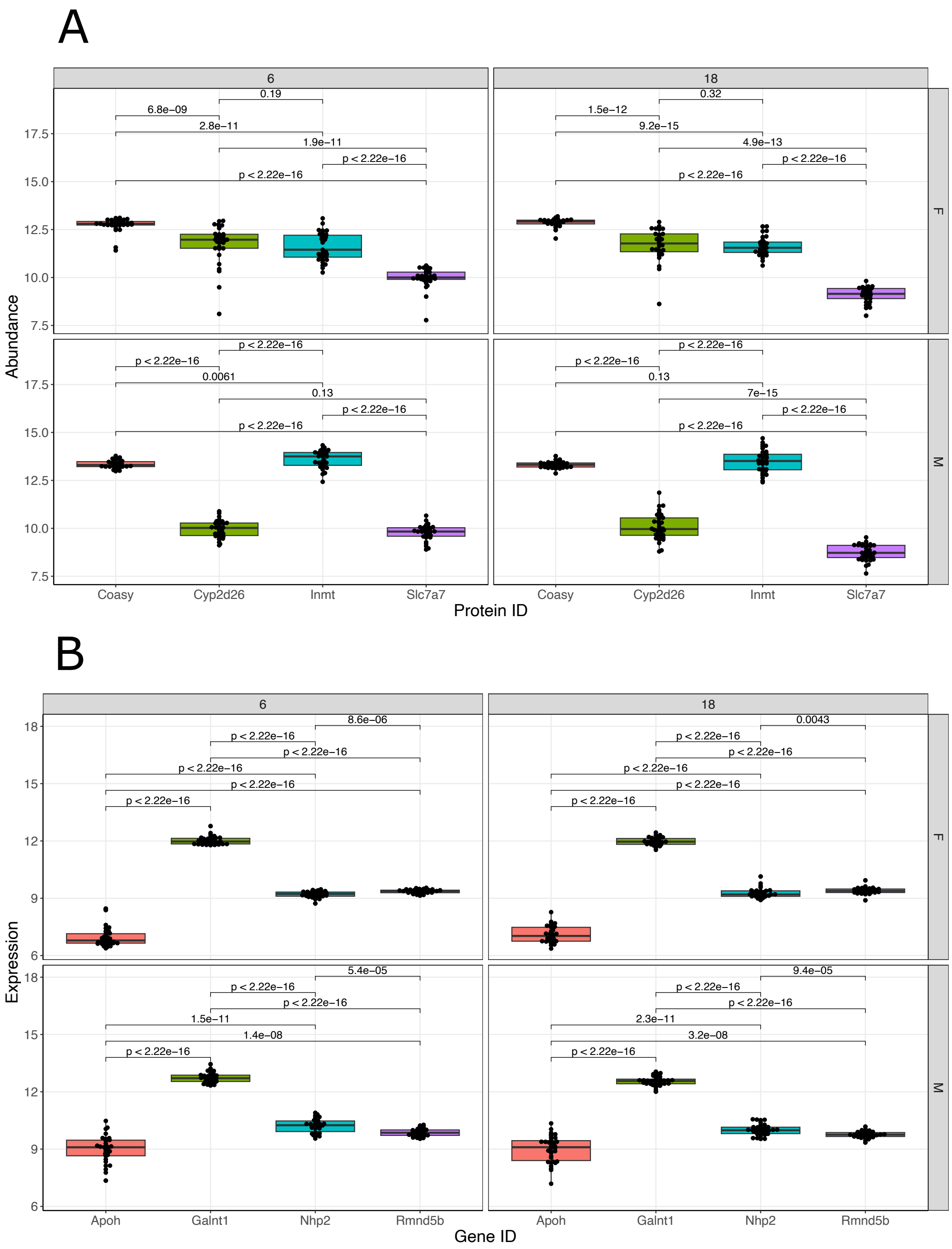

Figure 2.7: In each subplot, the statistical difference were obtained by means of a t-test and can be used to compare relative expression and abundance levels between phenotypes. **A)** Abundance of the four of the most important proteins for each phenotype. **B)** The expression levels for four of the most important genes for each phenotype.

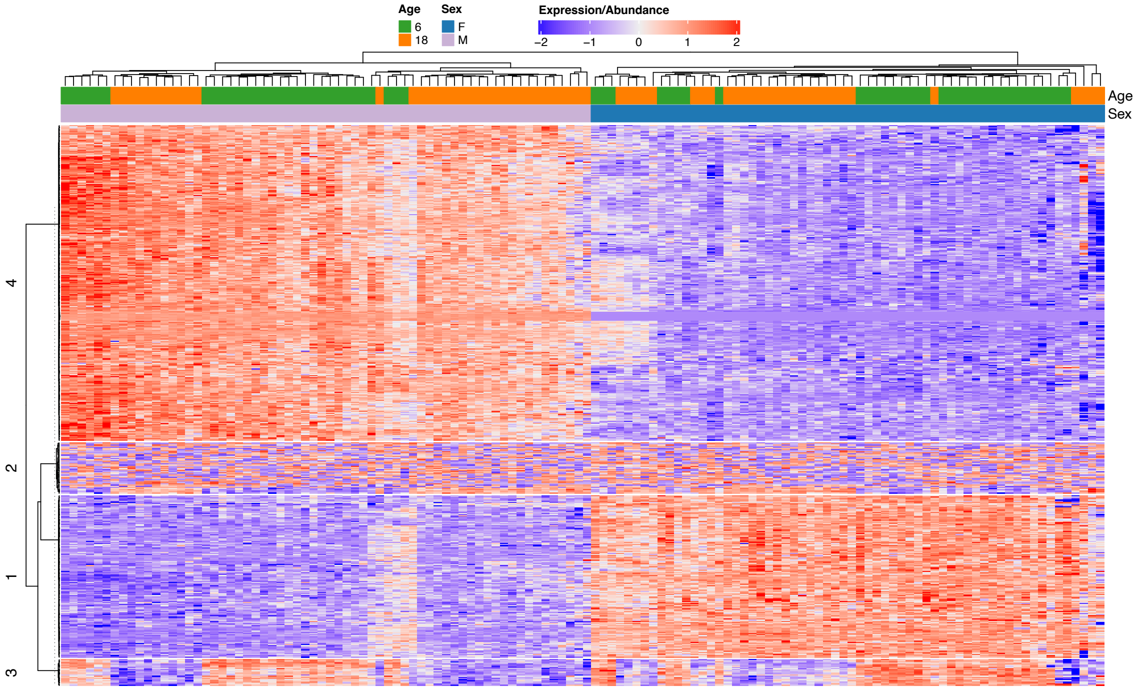

Figure 2.8: Heatmap of the most important features (including genes and proteins) as determined by kPPA-CART. The data has been scaled across rows. A checkerboard pattern separating the samples based on sex and age is observable.

understanding the molecular basis of MBC and yielded a significant number of public data generated by large consortia, including TCGA [17], Molecular Taxonomy of Breast Cancer International Consortium (METABRIC) [66] and AURORA US Metastatic Project [37]. In combination with early seminal work by Perou et al. [76] that identified four BC subtypes (normal-like tumors, i.e., basal-like, HER2-enriched, luminal, further divided into two subgroups), it becomes possible to stratify patients into treatment categories. Perou et al.'s molecular subtypes were identified by selecting a set of self-consistent “intrinsic” genes within a tumor subtype followed by hierarchical clustering to separate patients into transcriptionally distinct groups. This 50-gene signature has been referred to as the PAM50. Unfortunately, only samples in large retrospective studies could be classified by this signature. In the effort to develop more reliable classifiers that could identify the subtype of a single tumor, later studies identified new gene expression patterns and prognostic models [94] [50] [74] [43], and these have largely continued to be used.

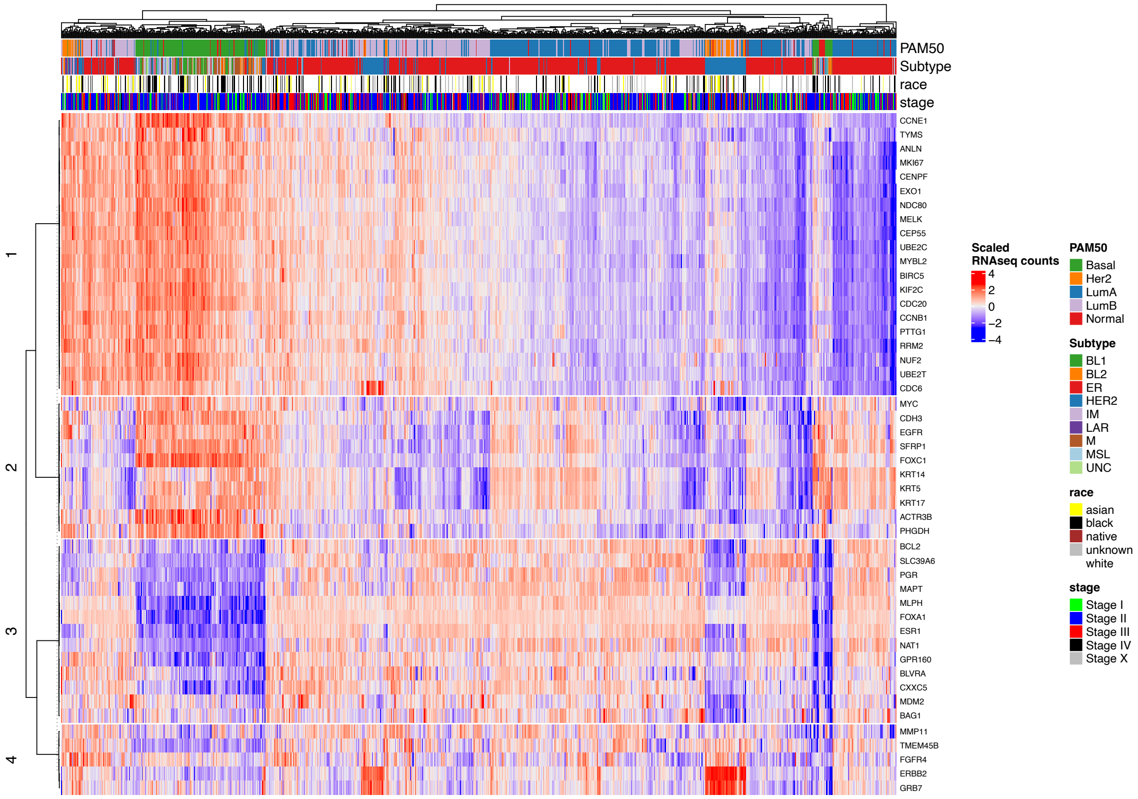

Figure 2.9: A heatmap of the genes that are accepted to stratify PAM50 phenotypes according to Perou [76].

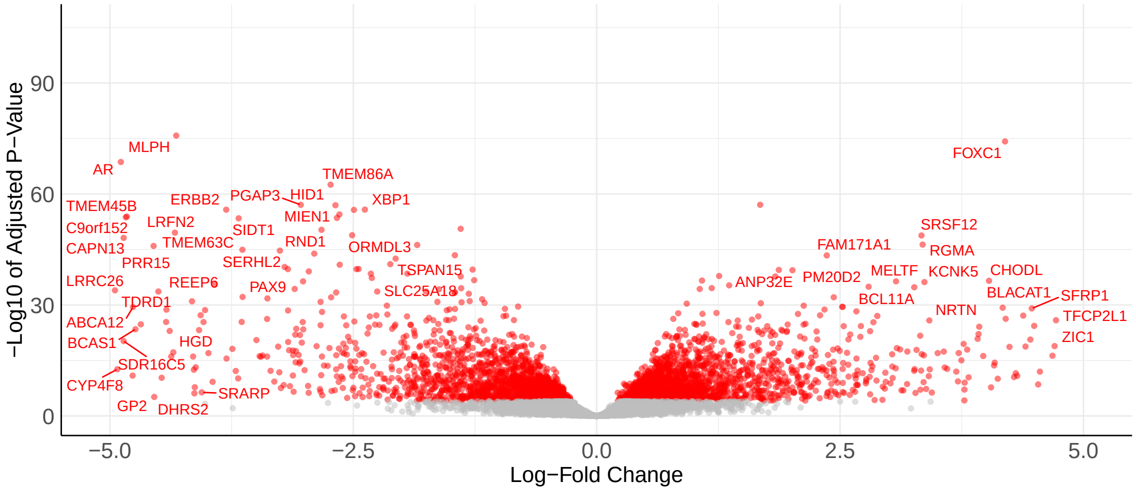

Figure 2.10: Volcano plot resulting from differential expression analysis from edgeR with the Basal PAM50 subtype as a reference.

kPPA-CART was applied to the PAM50 classification that accompanies many RNA-seq data in public repositories. I start with TCGA RNA-seq data which comprise of 1057 measurements of samples of primary breast cancer tumors from a cohort of female participants (59 years median age). A total of 735 participants self-identified as white, 168 as black or African American, 58 as Asian, and 1 as Native American, while 94 were unidentified. Among the PAM50 classifications, 162 samples were labelled “Basal”, 52 as “Her2”, 449 as “Luminal A”, and 340 as “Luminal B” while, 52 were labelled “Normal”. At the outset, I show a heatmap of the original genes used to define the PAM50 subtypes in Figure 2.9 to serve as a reference against which: (a) the conventional approaches for analyzing RNA-seq data, and (b) kPPA-CART are used to recapitulate these subtypes. The first step consists of filtering for low expressed genes and normalization to stabilize variance prior to analysing the data via edgeR [116]. Taking the Basal group as reference, Figure 2.10 and 2.11 show a volcano plot and heatmap, respectively, that correspond to differential expression of genes within this classification along with the heatmap of the most differentially expressed genes where a subset of the original panel of PAM50 genes is marked. The left panel of Figure 2.12 shows classification via standard approach (PCA), while the right panel of Figure 2.12 shows the kPPA-CART classification of a balanced (52 random samples per class, except normal type) subset of the data. Further, Figure 2.13 provides a heatmap of the 350 most important variables that classify the four BC subtypes, with the original panel of PAM50 genes marked.

To verify the ability of the features selected by kPPA-CART to classify PAM50 phenotypes, I randomly chose 40 samples from each of the four groups as a training set to perform a new kPPA-CART analysis, identified the most important variables, trained a random forest classifier based on the the training set, and predicted the remaining (12 of each group) samples. In Table 2.2 I show the confusion matrix depicting the accuracy of prediction. This step verifies that the features picked out by kPPA-CART carry importance when classifying the PAM50 subtype.

However, apart from classifying samples, the features should also carry biological significance. To determine the biological relevance of the genes identified via

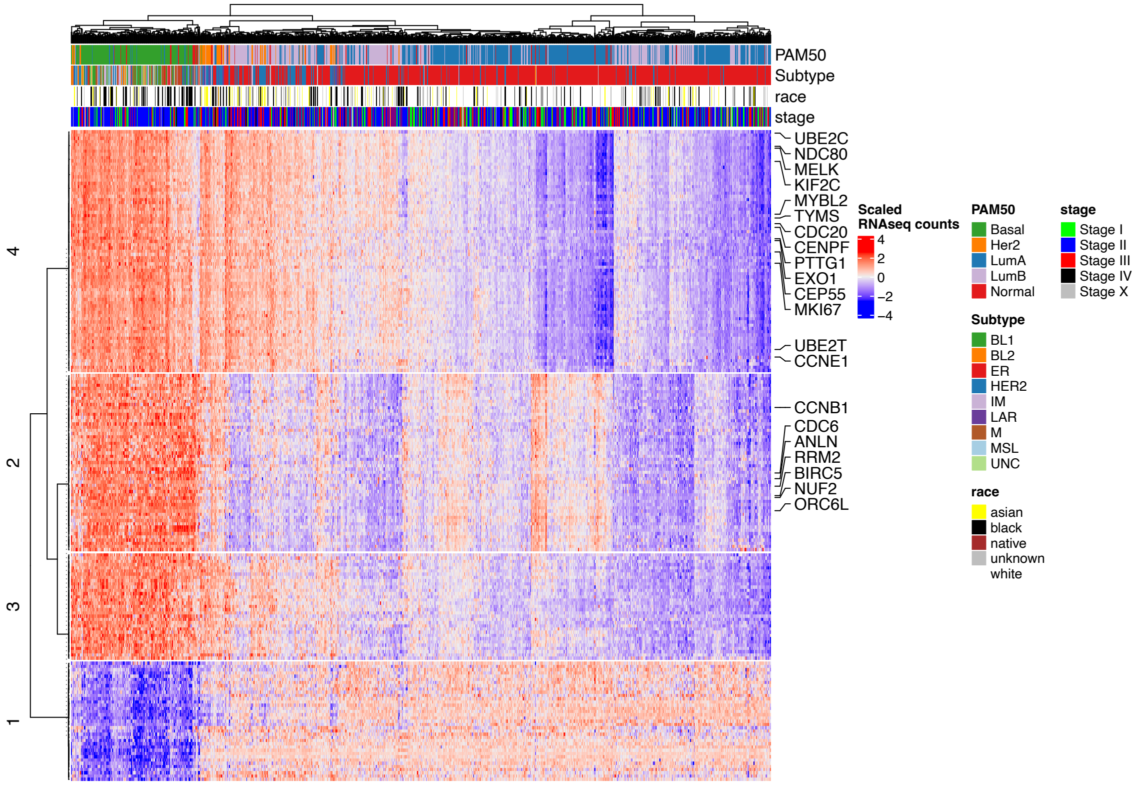

Figure 2.11: Heatmap of the most differentially expressed genes based on differential expression analysis that uses the basal group as a reference. The genes indicated on the right are part of the previously established panel of PAM50 genes.

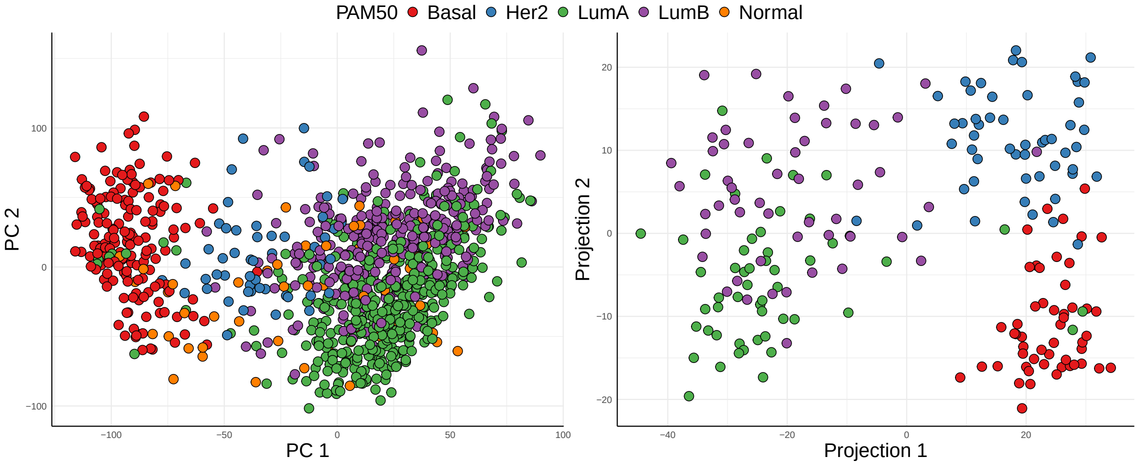

Figure 2.12: Analysis of the TCGA-BC RNA-seq (transcriptomics) data by means of PCA (left panel) and kPPA-CART (right panel). For kPPA-CART the dataset has been balanced and the samples classified as normal were removed.

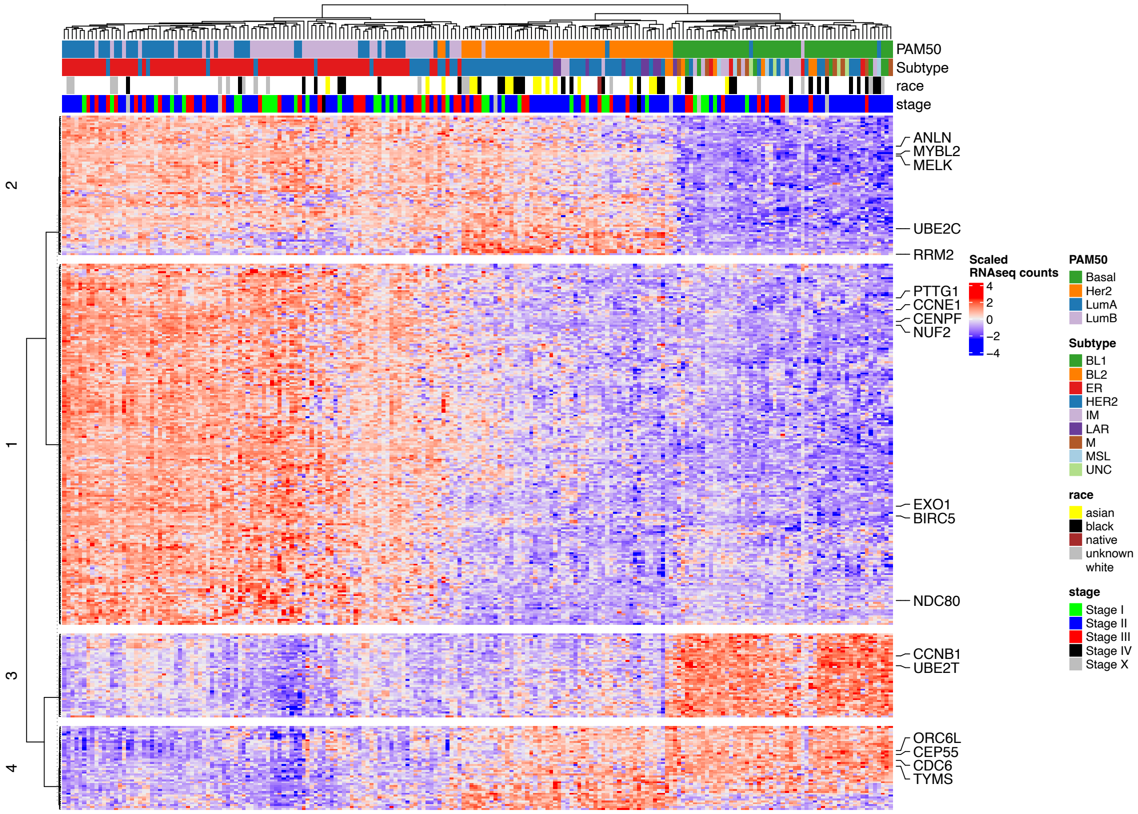

Figure 2.13: Heatmap of the 350 features determined by kPPA-CART to provide high importance in the classification of PAM50 subtypes. The genes indicated on the right are part of the previously established panel of PAM50 genes.

Type	Basal	Her2	LumA	LumB
Basal	12	1	1	0
Her2	0	11	0	2
LumA	0	0	9	0
LumB	0	0	2	10

Table 2.2: Confusion matrix resulting from the prediction of a test set containing 48 samples. The underlying random forest classifier was trained on a training set (160 samples) with 350 features that were determined to be significant in PAM50 stratification according to kPPA-CART.

kPPA-CART, I selected the top six most important features which are: GSTM1, ESR1, AGR3, CCDC170, C5AR2 and GATA3, and performed survival analysis using a readily available online tool called *KMPlot* [42]. These genes are shown to significantly impact patient overall survival despite not being part of the original PAM50 panel as evident in Figure 2.14. In *KMPlot*, the gene C5AR2 can be found under the

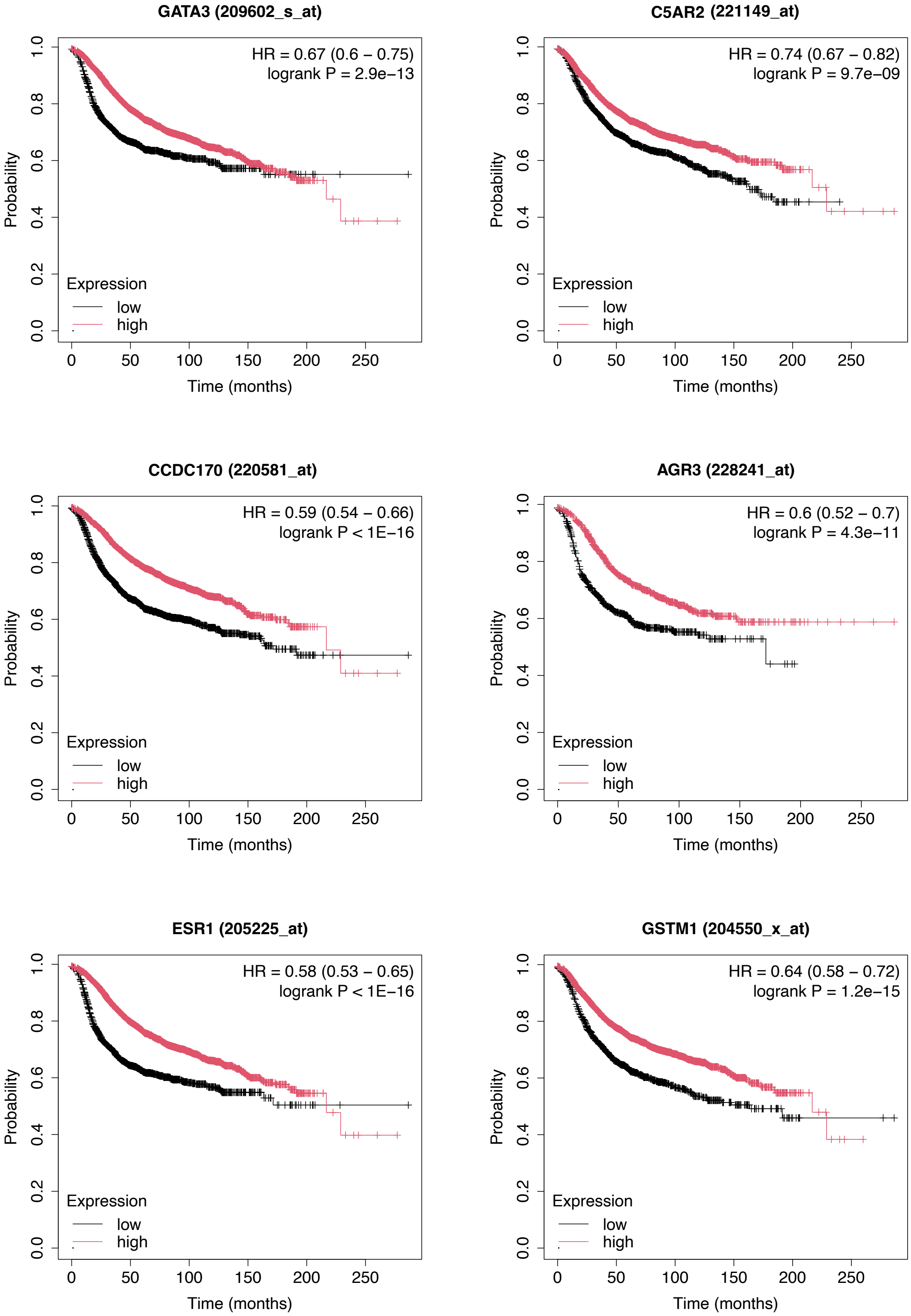

Figure 2.14: Kaplan-Meier survival plots [40] for the six most important genes identified by kPPA-CART analysis. The panels were created with the help of *KMPlot* [42].

synonym of GPR77 and CCDC170 can be found under the synonym of C6ORF97 [63]. Using the AURORA data as an independent validation set, a differential expression is observed between some of the genes in primary tumors compared to metastatic breast cancers visualized in Figure 2.15.

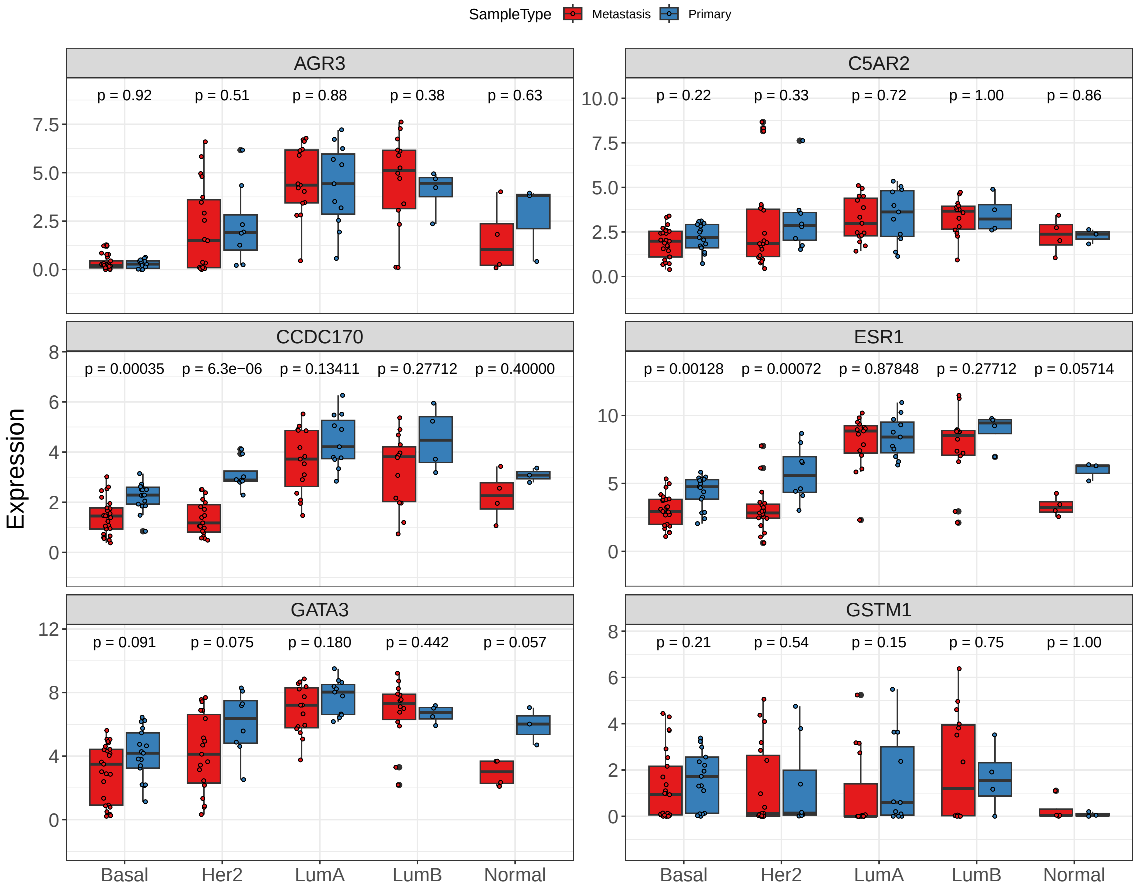

Figure 2.15: The expression levels of the six most important genes according to kPPA-CART for different PAM50 subtypes across metastatic and primary tumors. The displayed p-values are acquired by means of t-test. The data shown are extracted from the AURORA dataset [37].

Lastly, the gProfiler web tool [79] was used to get a high-level overview of the functional role of the genes extracted by kPPA-CART. In short, the results indicate that the genes of interest are involved in biological processes related to cancer progression, specifically system development as well as cell development and regulation of growth. Some of the genes identified by kPPA-CART, *e.g.*, GATA3 and GSTM1, play a crucial role in breast cancer as well as other cancers [58] [96] [22] [95]. Similar links to breast cancer can be made with some of the remaining genes, *e.g.*, ESR1 and C5AR2. Both genes have been reported as potential biomarkers in breast cancer [15] [117].

2.4 Discussion

I report the first implementation of kurtosis-based projection pursuit augmented with classification and regression trees (kPPA-CART) for the unsupervised analysis of -omics data to date. I have benchmarked the performance of kPPA against 7 of the most popular methods for multi-omics integration that have recently been expertly reviewed in Cantini et al. [18], and find that it offers superior sample classification. I specifically challenge these methods to model hard-to-access information in -omics data. In the most modern methods for analysis of omics data, low intensity signals are discarded, and the focus is turned to the highly variable features (HVF) which also happen to be the most dominant signals. I show that when the most prominent methods are employed to analyze low intensity signals, no biological groupings are discernible. Via kPPA, not only do I identify sample classification, but I also show that these features, which I identify via CART, carry biological significance.

Additionally, when inter-cluster dissimilarity is low (effect size), it is generally difficult to separate samples into their respective groups. This forces experimentalists to design experiments at the extreme ends of expected outcomes. For observational studies, effect sizes can be small, and kPPA-CART offers an opportunity to model a wide range of effect sizes for such studies or even studies where the effect sizes are implicitly small.

I have also employed kPPA-CART to model data derived from an experiment investigating the molecular changes that occur during the aging process of both male and female mice kidneys and show important biological changes. kPPA-CART applied to proteomics data shows four clusters of the data based on age and sex for 6- and 18-week-old mice. This separation is not clear when using standard approaches.

When I analyze mRNA data, I show that there is no clear separation, but when the data are combined and analyzed integratively, I continue to see the differences between male and female mice grouped according to their ages. I see that of the top 1000 features selected by CART, 215 represent protein features while the rest are drawn from the RNA set. This provides insights into which -omics type provides information that is most pertinent to the phenotype (clustering).

Extending our analysis to a more challenging breast cancer study, I show that kPPA-CART can identify novel features that classify BC data into: Basal, Her2, LumA and LumB subtypes better than the original features proposed by Perou et al. [76]. Compared to PCA classification, kPPA shows much crisper clusters of which features identified by CART exhibit biological importance in separating the two classes. Intriguingly, when the expression levels of these genes are compared across the PAM50 subtypes, they show only marginal differences but, via kPPA-CART, their biological significance can be identified.

2.5 Conclusion

This chapter explored a novel application of kurtosis projection pursuit analysis to enable clustering as well as feature extraction from high-dimensional multi-omics data, especially in cases where biological information is contained within low effect sizes and low variance features. After introducing the algorithm based on kPPA and classification and regrerssion trees, I have used simulations to show that kPPA-CART outperforms current methods. Further, I have applied kPPA-CART to two experimental data sets and provided evidence that the model is able to extract biological meaningful variables while providing coherent and correct clusters.

Chapter 3

Estimation of Measurement Error Models for LC-MS Data in Proteomics and Metabolomics

3.1 Introduction

The estimation of measurement error variance is critical to the evaluation of analytical data in several contexts that include the assessment of statistical significance, the assignment of uncertainty in derived results, the comparison of methods, and the evaluation of figures of merit such as limit of detection (LOD). Typically, measurement uncertainty is determined through the use of multiple experiments carried out at an appropriate level of replication that includes instrumental noise, as well as other potential sources of variation (sampling, biological variation, etc.). When measurements comprise a signal vector, the analytical errors may be characterized as correlated or independent, and may also be described as homoscedastic (uniform variance) or heteroscedastic (non-uniform variance). Consequently, the description of errors for a given instrument or method can represent a complex problem that is specific for a given technique.

In certain cases, especially where instrumental measurement errors are largely independent and the variance characteristics are somewhat predictable, the use of a variance function (VF) to describe the measurement error variance can be particularly convenient [98]. If we imagine a series of related measurements in a vector, $\mathbf{x}$ (e.g., a spectrum), then a specific measurement, x_i, can be defined in terms of the "true" (unknown) measurement, x_i^o, and the realization of the error, e_i, as given in Equation 3.1.

$$x_i = x_i^o + e_i \tag{3.1}$$

In VF modeling, it is assumed that the measurement error variance can be described in terms of a parameterized function of the measurement amplitude and (potentially) other variables as shown in Equation 3.2.

$$\sigma^2(x_i) = E[(x_i - x_i^o)^2] = f(x_i^o, ...) \approx f(x_i, ...) \tag{3.2}$$

Here, E denotes the expectation operator and f is the VF. Additional variables may be explicit (e.g., temperature) or implicitly expressed in the parameterization (e.g., which instrument is employed), but in practice the VF is usually expressed only as a function of the amplitude, x. It is assumed to be an exact function of the true value, which is unknown, but typically approximated for real measurements using the observed value. The form of function is dependent on the nature of the measurement, but common empirical forms are usually assumed. For example, a model used in liquid chromatography/mass spectrometry (LC-MS) measurements is given by Equation 3.3, which incorporates components of Poisson noise in the detector (first term) and source flicker noise (proportional) in the ion source (second term) [72] [104].

$$\sigma^2(x_i) = \beta_1^2 x_i^o + \beta_2^2 (x_i^o)^2 \tag{3.3}$$

Generally, the parameters ($\boldsymbol{\beta}$) associated with the VF are determined using replicated data to estimate the mean and standard deviation for each set of measurements and then fitting the model given in Equation 3.4.

$$s^2(x_i) = f(\bar{x}_i) + \delta_i \tag{3.4}$$

Note that the mean, $\bar{x}$, is used as an approximation of the true value in this estimation. Here, δ_i describes random error. Fitting the model parameters is generally carried out using a least squares method, although this is complicated by weighting issues and asymmetric residuals [98].

When a VF can be reliably used to model error variance, it has a number of important advantages. First, the VF is likely to give a more reliable estimate of measurement uncertainty than simple replication, since estimates of variance based on a limited number of samples are known to be highly uncertain themselves [98] and very susceptible to outliers. The use of hundreds or thousands of data points to estimate a VF can lead to a more dependable estimate as long as the model is valid. A second, very practical advantage is that measurement uncertainty can be estimated for cases where replicate measurements are unavailable, which is often a reality for experiments where repeat measurements are limited by sample availability or experimental design. Third, the functional form of the VF allows for a more well-defined determination of figures of merit such as LOD in the presence of heteroscedastic noise. Finally, the functional form of the VF facilitates a better elucidation of sources of variance for a particular instrument or method and permits some comparisons to be made across platforms. In spite of these advantages, however, it is important to be aware of the limitations of VFs. Notably, their utility is restricted to the context of their estimation. For example, a function developed simply to model instrumental noise will no longer be valid when other sources of variation (e.g., sample preparation) are present, although the model can still provide an indication of limiting uncertainty.

The central purpose of this work is to propose an approach for estimating variance functions for LC-MS data typically recorded for proteomics and metabolomics experiments. Although traditional approaches based on Equation 3.4 have been used [5], these ignore the issue of sensitivity variation between experiments in MS data. The term “sensitivity” used in this context refers to the ratio of instrument response to a given analyte concentration and can encompass both variation in instrument performance (e.g., detector response) and variation in the apparent analyte concentration (e.g., due to variations in the amount of sample injected). While the former case can be regarded as a true change in instrumental sensitivity, the latter is only an apparent change, but the implications are the same for replicated experiments. Such changes can be observed when data for duplicate samples are plotted against one another and yield a line with a slope that deviates significantly from the theoretical value of unity. In some cases, these deviations can be small, but in others they can be substantial. Ordinarily, some type of normalization is performed to accommodate sensitivity changes at the data analysis stage, but such a procedure complicates estimation of the VF. If replicates are analyzed without normalization, the VF is likely to be dominated by the systematic (correlated) noise reflected in the sensitivity changes, which is generally not of interest. On the other hand, the use of normalization can result in combining measurements with different variances, leading to a mixture of distributions rather than true replicates drawn from a single distribution. Moreover, the normalization process itself will likely influence the resulting distribution. Therefore, the method proposed here couples the VF estimation with a normalization procedure based on maximum likelihood principal components analysis (MLPCA). Additionally, the new approach optimizes the VF parameters using the distribution of standardized residuals rather than traditional least squares objective functions. The novel method is demonstrated with diverse experimental datasets from proteomics and metabolomics studies, and validated through the use of simulations.

3.2 Background

3.2.1 Normalization

Figure 3.1 is intended to conceptualize the process of normalization for LC-MS data in the context of VF estimation. For the sake of illustration, we will assume duplicate signals are recorded, although more replicates would typically be used for the estimation of variance. Figure 3.1A shows the ideal case where there is no change in sensitivity between the two experiments. The vertical axis shows the signal amplitude of the replicate measurements, with red and blue points used to represent measurements from the duplicate experiments. This amplitude could be expressed in various ways, such as chromatographic peak height or area in terms of detector counts. The horizontal axis is labeled as a “reference amplitude”, $\mathbf{x}_{ref}$, which can be considered to be a “true” amplitude or population mean for the sake of conceptualization, but in practice would likely be represented by a sample mean. The purple line shows the ideal slope of unity, with dashed lines showing 95% confidence boundaries for an arbitrary VF with an amplitude dependence. Although the nature of the measurement and any low-level preprocessing will have implications for the characteristics of the VF, this will be considered later. The error bar shown in the middle of the plot reflects the population standard deviation (95% confidence interval) for a particular point and captures the true variance of the measurement. Note that the uniform spread of measurements along the x-axis shown in Figure 3.1 is atypical of most proteomics/metabolomics results, which are normally skewed to low intensities, but this is intended for illustration only.

The situation illustrated in Figure 3.1A is ideally suited for traditional VF modeling of instrumental measurement noise since the experiments are true replicates and the only source of variance is instrument noise (i.e., no changes in sensitivity). Figure 3.1B illustrates a more likely scenario where there is a change in instrumental sensitivity between experiments. Although the change is exaggerated here for the sake of illustration, it is not unrealistic for some experiments. The solid lines (red and blue) reflect the sensitivity of each experiment and the reference intensity (x_{ref}) could be considered to be the true mean for the purpose of illustration. The dashed lines represent the 95% limits for the instrumental noise in each experiment. With no normalization applied, it is clear that the estimated variance of a given pair of measurements will include both the random instrumental variations and the systematic differences introduced by changes in sensitivity. This is illustrated by the purple error bar that reflects the anticipated confidence intervals for a typical point, which are substantially enlarged over the example in Figure 3.1A. Using such measurements to estimate a VF is undesirable for a number of reasons. Principally, this approach would not yield a useful VF model for experiments where normalization is routinely performed to mitigate systematic variations. Any model developed would overestimate the variances under these circumstances and would also ignore the fact that the systematic variations reflect a high level of correlated noise, violating assumptions of independence that may be made in applying the VF model. Moreover, a VF model based on the mean of measurements under these conditions would be unlikely to capture the true functional relationship for instrument variability because it is not based on the mean instrument signal under fixed conditions.

If we reject the unacceptable approach illustrated in Figure 3.1B, the logical approach would be to normalize the replicate experiments before estimating the VF model. This is the situation illustrated in Figure 3.1C, where the two datasets in Figure 3.1B have been appropriately normalized so that they reflect the same sensitivity. In the simplest implementation, which will be assumed here, normalization involves

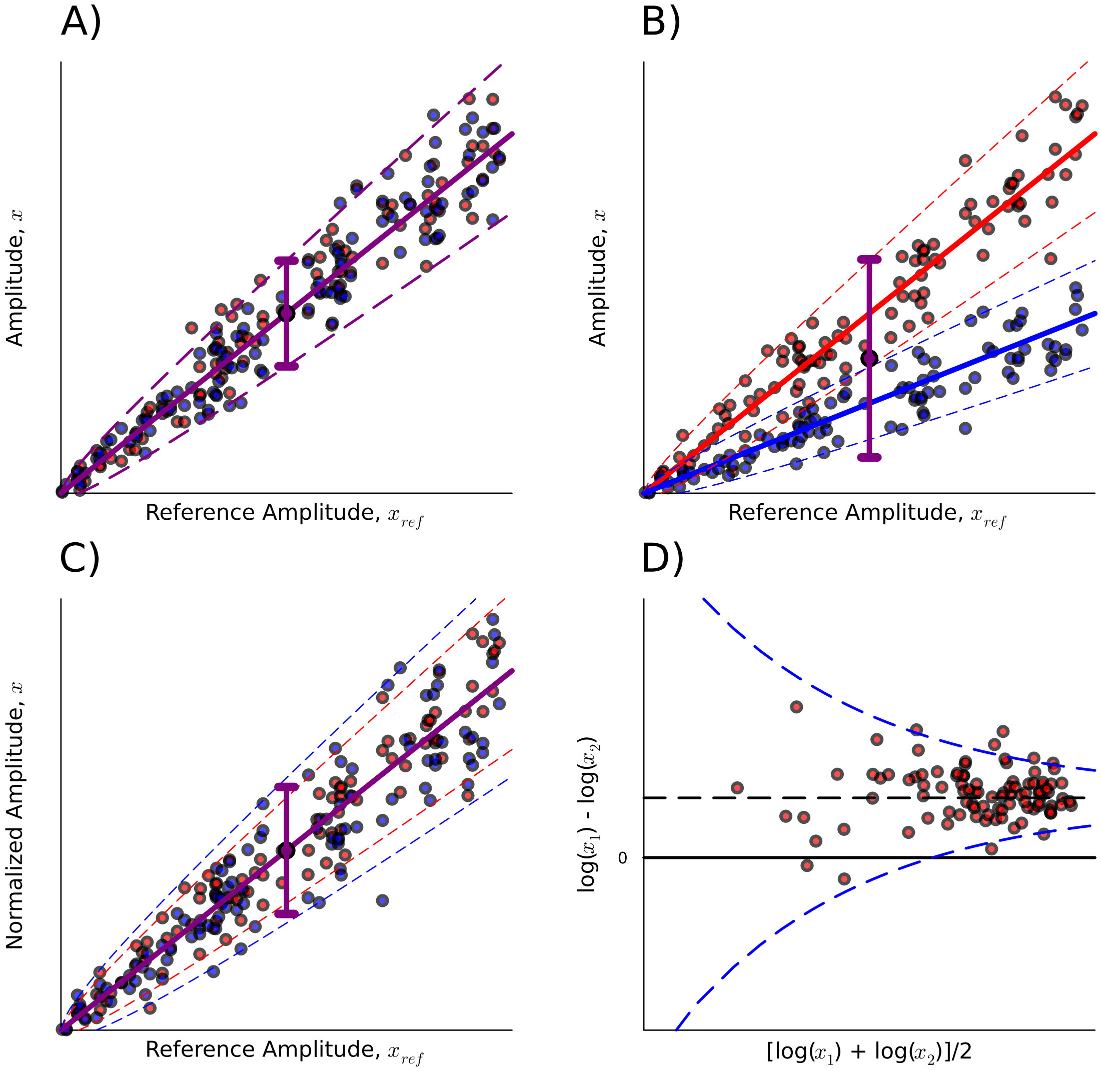

Figure 3.1: Illustration of normalization and variance function estimation for duplicate experiments. **A**) Variance estimation in the ideal case where there are no sensitivity changes and normalization is not required. **B**) Variance estimation in the presence of sensitivity differences but without normalization. **C**) Variance estimation in the presence of sensitivity differences following normalization. **D**) Normalization of the data in **B**) using the MA method. In **B**) and **C**), the red and blue dashed lines indicate the true 95% confidence intervals of each set of measurements individually.

the scaling of measurements from each experiment, as given in Equation 3.5.

$$x_{ij}^{Norm} = \alpha_j x_{ij} \tag{3.5}$$

Here, x_{ij} represents measurement i from experiment j and α_j is the normalization factor for that replicate set. The normalization factor is typically determined by a regression procedure, described in more detail below. The scaling is intended to adjust all of the replicates to the same sensitivity, although the actual reference scale is arbitrary. Typically, signals are normalized to the mean or the experiment with the highest sensitivity. Although there are issues associated with the normalization itself, for the sake of simplicity, it is assumed here that the normalization is performed perfectly so that there is no residual variance due to sensitivity change and only the instrumental noise variance remains. Since normalization involves scaling measurements to adjust the magnitudes of the observed intensities, errors in those measurements are subsequently scaled, which leads to different error distributions. This is represented by the red and blue confidence intervals shown by the dashed lines in Figure 3.1C. This results in a mixture of distributions and the uncertainty calculated through the combination of normalized measurements (indicated by the purple error bars for a representative point) will not reflect the true nature of the variance model. Although the divergence of individual standard deviations may be small, this can have important implications for VF modeling, depending on the nature of the model. If only proportional errors are observed (*i.e.*, the relative standard deviation, RSD, in the measurements is constant), then normalization will produce homogenous distributions. However, models are generally more complex and may not follow this proportional structure, especially at low intensities.

Given that VF modeling is intricately connected to the normalization process, a more detailed examination of normalization practices is required. Normalization procedures, many of which have been adapted from DNA microarray protocols for transcriptomics, vary widely and include the use of reference measurements (*e.g.*, internal standards, marker analytes), statistical transformations (e.g., median centering, variance stabilization, quantile normalization) and regression-based methods [51]. It is important to note that, in the more general context of -omics studies, normalization is applied to experimental measurements that are not replicates; that is, the experiments typically involve some change in the biology where differential expression of some analytes is expected. Therefore, the "correct" normalization is often ill-defined. For example, in some cases, one may be interested in changes in absolute concentration of an analyte, while in others, changes relative to other analytes may be of interest. Most normalization methods rely on an assumption that most analytes do not change their expression levels during the course of an experiment and can therefore be used to correct for systemic measurement artifacts. The most appropriate normalization method may depend on the system under study.

In the present context, where replicate experiments are being examined to obtain a model for measurement error variance, the goal is more well-defined since there should be no differential expression of analytes. Variations in the analyte measurements should therefore be random (except for sensitivity changes) and, at least to a first approximation, independent. A simple approach to determine normalization factors in a set of I measurement channels across each of J experiments would be to use a regression model of the form given in Equation 3.6 to fit the normalization factors for each experiment, α_j.

$$x_{ij} = \alpha_j x_i^{ref} + \epsilon_{ij} \tag{3.6}$$

In practice, there are several difficulties with this approach. First is the need to define a reference signal, x_{ref}, for each of the measurements. Ideally this would be the "true" (error-free) measurements for one of the experimental datasets, or perhaps the population mean for each measurement channel. Since these are unavailable, the actual (noisy) data for one set of experiments or the sample means can be used. However, this introduces uncertainty in the independent variable and, in the latter case, errors that are correlated with those of the dependent variable. A larger problem is that the measurement errors are typically proportional *i.e.*, they increase with the magnitude of the signal. Given that the distribution of measurements in -omics experiments often have the highest density for low-intensity signals, the higher variance in higher intensity (high leverage) signals will tend to increase the variance in estimates of the normalization factors, making them less reliable. A solution to this problem is to use weighted regression, but this requires prior knowledge of the model for measurement errors, which is the VF model we are trying to obtain.

In practice, the solution to the problem of heteroscedastic measurement errors is often to use a logarithmic transformation of the data. This assumes that the measurements exhibit a proportional error structure so that, by propagation of error, a logarithmic transformation will lead to a uniform error variance; that is if $\sigma_x = \gamma x$, then $\sigma(\log_b(x)) = \gamma/\ln(b)$. Empirically, the assumption of proportional errors is approximately valid for many experiments, especially for high-intensity measurements. The transformation of the model in Equation 3.6 now becomes the form represented in Equation 3.7.

$$\log_b(x_{ij}) = \log_b(\alpha_j) + \log_b(x_i^{ref}) + \delta_{ij} \tag{3.7}$$

The problem now becomes one of obtaining an estimate of the intercept in a case where the slope is unity, which can be solved using the mean of the difference between the first and last log terms. Commonly the logarithmic approach is often implemented through the so-called *MA* ("minus/average") normalization method adopted from transcriptomics [16]. The *MA* approach can also allow the normalization factor to effectively change as a function of intensity through to use of global or localized regression. In the current application which involves simple replicates from methods with good linearity, the normalization factor is assumed to remain fixed.

The logarithmic or *MA* approach is an effective normalization strategy for replicated experiments but is not without its deficiencies. As illustrated with the *MA* plot for the simple simulation in 3.1D, the variance tends to increase at low intensities where Poisson noise dominates, making a weighted regression more appropriate, although this is not possible without an error model.

Perhaps a more rational strategy to the problem of normalization in the present context is an errors-in-variables approach in which one set of measurements is regressed against the other (e.g., x_2 vs. x_1) while considering uncertainties in both variables. Noting that the intercept should be zero, this problem is most conveniently presented as a latent variable model such that:

$$[x_{i1}\ x_{i2}] = t_i \cdot [v_1\ v_2] + [e_{i1}\ e_{i2}] = t_i \cdot \mathbf{v}^T + \mathbf{e}_i^T \tag{3.8}$$

In this form, t_i represents the underlying latent variable (or score) for measurement pair i (proportional to x_{ref}) and the vector $\mathbf{v}$ contains the scale factors (loadings) for the two experiments. Typically, $\mathbf{v}$ is constrained to be unit length to remove ambiguities of scale in this formulation. This representation is easily extended for additional replicate experiments to give a multilinear equation in higher dimensional space and is presented in matrix form in Equation 3.9.

$$\mathbf{X} = \mathbf{t} \cdot \mathbf{v}^T + \mathbf{E} \tag{3.9}$$

Here, $\mathbf{X}$ and $\mathbf{E}$ have dimensions of I measurement channels by J experiments (replicates), $\mathbf{t}$ is an Ix1 score vector and $\mathbf{v}$ is a Jx1 vector of loadings with a Euclidean norm of unity. In the simplest case where the measurement errors in $\mathbf{X}$ are independent and identically distributed with a normal distribution (*iid* normal, $E(e_{ij}^2) = \sigma^2$), the maximum likelihood estimates for $\mathbf{t}$ and $\mathbf{v}$ are provided by simple principal components analysis (PCA) or singular value decomposition (SVD), where $\mathbf{v}$ is the first eigenvector. In this case, the maximum likelihood estimates of the measurements is provided by an orthogonal projection onto the loading vector, as given in Equation 3.10.

$$\hat{\mathbf{X}} = \mathbf{X}\mathbf{v}\mathbf{v}^T = \mathbf{t} \cdot \mathbf{v}^T \tag{3.10}$$

A major impediment to the implementation of this approach to normalization is that the measurement errors for LC-MS experiments deviate significantly from *iid* normal and show a large degree of heteroscedasticity (non-uniform variance). Therefore, to apply the latent variable model it is necessary to employ a method that incorporates more complex error structures. One of these methods, and the one employed in this work, is maximum likelihood principal components analysis (MLPCA), as described in the next section.

3.2.2 Maximum Likelihood PCA

MLPCA is a technique for fitting linear subspaces in higher dimensions using latent variables and considering known error characteristics of the multidimensional measurements [110]. The name is intended to highlight that the results are presented in a format analogous to PCA, but whereas PCA is intended to model the maximum amount of residual variance in the data with successive components irrespective of the source of the variance, MLPCA attempts to highlight the meaningful variance by partitioning it from the error variance using prior information. Consequently, many of the properties familiar in the application of PCA are not necessarily shared with MLPCA. Moreover, MLPCA is truly a maximum likelihood method only if the subspace model is valid and the measurement errors are normally distributed and have a known variance/covariance structure. The approach is closely related to other errors-in-variables methods, such as total least squares [93] and positive matrix factorization [73]. The theoretical foundations and practical application of MLPCA have been discussed elsewhere [110] and will be treated only briefly here.

The general framework of MLPCA is intended to accommodate a range of independent and correlated measurement error structures, although multivariate normality is assumed in all cases. For the normalization problem under consideration, we will assume independence of the measurement errors and a general non-uniformity of variance, which corresponds to "case C" in the descriptions of MLPCA algorithms [110]. For the moment, we will assume that the error variance for each measurement is known a priori. In this scenario, the problem of normalization becomes one of fitting the one-dimensional model described in the previous section in the J dimensional space of replicated experiments. For the case of duplicate experiments, this is illustrated in Figure 3.2 using the previous example. Maximum likelihood estimation requires minimization of the objective function, S^2, given in Equation 3.11.

$$S^2 = \sum_i \sum_j \frac{(x_{ij} - \hat{x}_{ij})^2}{\sigma_{ij}^2} = \sum_i (\mathbf{x}_{i\bullet} - \hat{\mathbf{x}}_{i\bullet}) \boldsymbol{\Sigma}_i^{-1} (\mathbf{x}_{i\bullet} - \hat{\mathbf{x}}_{i\bullet})^T = \sum_j (\mathbf{x}_{\bullet j} - \hat{\mathbf{x}}_{\bullet j})^T \boldsymbol{\Sigma}_j^{-1} (\mathbf{x}_{\bullet j} - \hat{\mathbf{x}}_{\bullet j}) \tag{3.11}$$

Here, σ_{ij} is the error standard deviation for measurement x_{ij} as previously defined,

$\mathbf{x}_{i\bullet}$ is a 1xJ row vector in $\mathbf{X}$ with $\mathbf{\Sigma}_i$ as the corresponding JxJ (diagonal) error covariance matrix (ECM), and $\mathbf{x}_{\bullet j}$ is an Ix1 column vector in $\mathbf{X}$ with $\mathbf{\Sigma}_j$ as the corresponding IxI ECM. This is similar to the objective function minimized in PCA except: (1) no weighting is used in PCA, and (2) PCA uses an orthogonal projection to estimate x, whereas MLPCA employs a maximum likelihood (oblique) projection as given in Equation 3.12.

$$\hat{\mathbf{x}}_{i\bullet} = \mathbf{x}_{i\bullet}\mathbf{\Sigma}_i^{-1}\mathbf{v}(\mathbf{v}^T\mathbf{\Sigma}_i^{-1}\mathbf{v})^{-1}\mathbf{v}^T \tag{3.12}$$

As before, $\mathbf{v}$ is the Jx1 loading vector which contains the scaling factors for the normalization.

These concepts are illustrated in Figure 3.2 for the simple two-dimensional case where the measurements from experiment 2 are plotted against the measurements from experiment 1. The uncertainty ellipses (red) are shown for three selected points (blue), with an expanded view for one of the points. Each ellipse represents the uncertainty of the corresponding point as determined by its 2x2 ECM (1σ, or 39% confidence interval), and the black line corresponds to the maximum likelihood estimate of $\mathbf{v}$ (i.e., its slope is v_2/v_1). The axes are shown at equivalent scales, so the longer horizontal axis in each ellipse corresponds to larger uncertainty in the first measurement, and the increasing size of the ellipses corresponds to their dependence of the uncertainty on the magnitude of the measurement. The red asterisks show the maximum likelihood projections of the blue points onto the model, where the direction of the projection depends on the relative magnitudes of the uncertainties. Although the figure illustrates the case of a one-dimensional model in two dimensions, it is easily extended to higher dimensions (more replicates), where the ECM is represented as a hyper-ellipsoid.

If the measurement error structure is accurately known for all of the measurements in Figure 3.2 and the model is correctly estimated, then the standardized residuals (see section 3.2.4) across all measurements should follow a standard normal distribution, as shown in inset (A) in the figure. This forms the basis of estimating the correct VF in this work, as described in section 3.2.5.

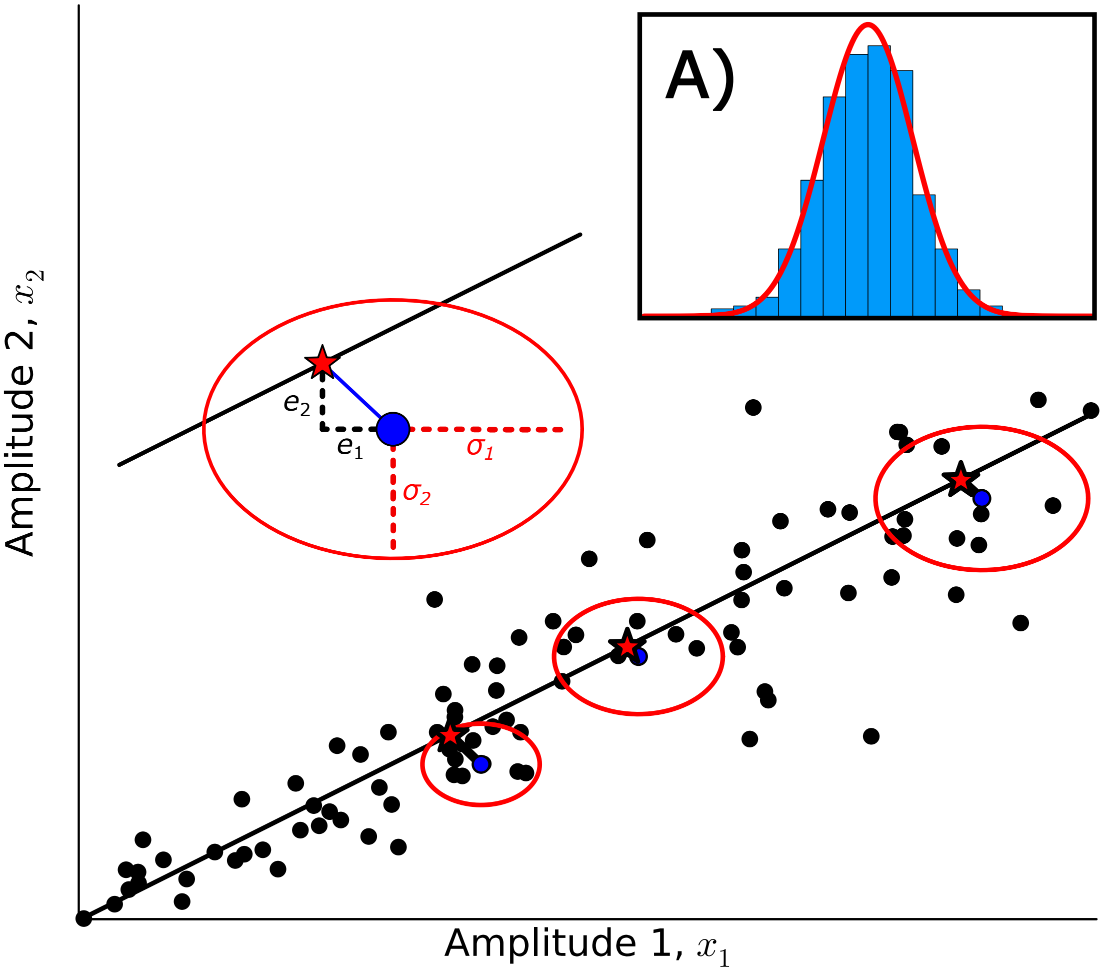

Figure 3.2: Illustration of normalization by MLPCA in two dimensions. The black line indicates the line of best fit ($\mathbf{v}$). For three selected pairs of measurements (blue dots) the uncertainty ellipses are shown (1 standard deviation) along with the maximum likelihood projection (red asterisk). An expanded view for the first point is shown. Inset **A**) shows the distribution of standardized residuals.

MLPCA estimates the model using an alternating least squares (ALS) algorithm and utilizes error variances specified for each measurement. In the present application, however, the error variance estimates should be obtained using a variance function provided to the algorithm. This requires a modification to the original software to what will be referred to as dynamic MLPCA, as described in the next section.

3.2.3 Dynamic MLPCA

Modification of the original MLPCA algorithm [110] for the present application is, on the surface, fairly straightforward, by simply replacing the matrix of measurement error variances passed to the algorithm with the VF and allowing the variance to be calculated for each measurement. However, two adjustments are required for this new implementation.

First, the VF can be used to estimate the measurement error variances directly from the measurements passed to the algorithm, but this is expected to be less accurate than using the measurements projected onto the model, which are not available until the model is estimated. To solve this problem, an iteratively re-weighted least squares strategy is invoked with an outer loop added to the ALS procedure within the algorithm. Initially, variance estimates are obtained from the original measurements and used to estimate the loading vector, $\mathbf{v}$, and the maximum likelihood estimates of the points, $\hat{\mathbf{X}}$, using Equation 3.12. These estimates are then used in the VF to update the variance estimates. This process is repeated until convergence of the solution, which normally requires only a few iterations.

A second, more subtle adjustment to the algorithm results from the need to accommodate the rare cases where the VF gives rise to invalid variances (*e.g.*, negative, zero or undefined) based on measured or projected values. This will depend on the form of the VF but is likely to occur only when measured/projected values are zero or negative. In these circumstances, the measurements are treated as missing and assigned infinitely large variance which corresponds to this data point not contributing to the MLPCA model. As long as there is at least one valid variance for a given variable (row of $\mathbf{X}$), a projected measurement can be calculated. However, if at any point all of the measurements or projections for a given variable are invalid, it is no longer possible to calculate that projection and none is returned from the algorithm. This is an infrequent occurrence, however.

The inputs to the dynamic MLPCA algorithm include the matrix of measurements, the VF and its parameters, and an initial estimate for the loading vector, $\mathbf{v}$ (optional). The most reliable way to obtain an initial estimate for $\mathbf{v}$ is to use a traditional normalization approach, such as the MA method discussed in section 3.2.1. Once the normalization factors have been obtained in the form of a vector, α, this can be transformed to a loading vector by scaling with the Euclidean norm as given in Equation 3.13.

$$\mathbf{v}_{init} = \boldsymbol{\alpha}/||\boldsymbol{\alpha}|| \tag{3.13}$$

The algorithm can accommodate missing measurements in $\mathbf{X}$ (represented as "not a number", or NaN in Julia [12]). If an initial estimate of $\mathbf{v}$ is not provided, it will be approximated using simple PCA applied to the rows of $\mathbf{X}$ with no missing measurements. The algorithm returns the vectors for the decomposition in standard SVD format ($\mathbf{U}$, $\mathbf{S}$, $\mathbf{V}$) from which the projected data can be calculated ($\hat{\mathbf{X}} = \mathbf{u}S\mathbf{v}^T$).

3.2.4 Standardization of Residuals

The foundation of the present approach is that when the normalization factors (in the form of the loading vector, $\mathbf{v}$) and the VF are correctly estimated, the residuals will follow a predicted distribution. In this case, the distribution of residuals can be considered individually (for each measurement) or collectively (for each set of replicates at a given channel). For the second approach, the strategy is to calculate the Mahalanobis distance, d_M, for each set of replicates, according to Equation 3.14, [61].

$$d_M(i) = \sqrt{(\mathbf{x}_i - \hat{\mathbf{x}}_i)\boldsymbol{\Sigma}_i^{-1}(\mathbf{x}_i - \hat{\mathbf{x}}_i)^T} \tag{3.14}$$

Here, $\mathbf{x}_i$ is a $1xJ$ row vector in $\mathbf{X}$ representing replicated experiments for a given variable and $\mathbf{\Sigma}_i$ is its corresponding error covariance matrix. In principle, the squared Mahalinobis distances should follow a χ^2 distribution with $J-1$ degrees of freedom (DOF) and this could be used to evaluate the error model [14]. A drawback of this approach, however, is that it requires a full set of measurements for each variable to have consistent DOF. It is very common for at least one measurement to be missing from each variable set, and the likelihood for this increases with the number of replicate experiments conducted. An advantage of MLPCA for normalization is that missing measurements (or outliers) can be accommodated by associating them with a large variance, which means they are excluded from the fit without the necessity of removing the other measurements in the same set. While the d_M can still be calculated in the presence of missing measurements, the statistical properties will not be consistent across all variables.

An alternative approach, used here, was to standardize the individual residuals and compare their distribution to a standard normal distribution. This raises the question of how to standardize the residuals. In an ordinary regression problem in which the measurement error variances in y are much larger than those in x, one would typically standardize based on division of the residuals by σ_y. However, for errors-in-variables methods, the standardization is more complex and depends on both the model (the loading vector, $\mathbf{v}$) and the uncertainties of all of the measurements in a set. This dependence structure is visible by examining the calculated residuals from the one-dimensional MLPCA model as shown in Equation 3.15.

$$\mathbf{\Delta x}_i = \mathbf{x}_i - \hat{\mathbf{x}}_i = \mathbf{x}_i - \mathbf{P}_i\mathbf{x}_i = (\mathbf{I}_J - \mathbf{P}_i)\mathbf{x}_i \tag{3.15}$$

Here, $\mathbf{x}_i$ is a $Jx1$ vector representing the J replicate measurements for channel i, with its maximum likelihood estimate, $\hat{\mathbf{x}}_i$, given by pre-multiplication by the JxJ projection matrix, $\mathbf{P}_i$, which is defined according to Equation 3.16.

$$\mathbf{P} = \mathbf{v}(\mathbf{v}^T\mathbf{\Sigma}_i^{-1}\mathbf{v})^{-1}\mathbf{v}^T\mathbf{\Sigma}_i^{-1} = K_i\mathbf{v}\mathbf{v}^T\mathbf{\Sigma}_i^{-1} \tag{3.16}$$

In this equation, $\mathbf{v}$ is the $Jx1$ loading vector defining the model, $\mathbf{\Sigma}_i$ is the JxJ error covariance matrix for measurement vector $\mathbf{x}_i$, and K_i is a scalar representing the term in parentheses. In the case of uncorrelated errors, this is given by Equation 3.17. Here it is interesting to note that $\mathbf{P}$ depends on $\mathbf{v}$ and $\boldsymbol{\Sigma_i}$ which must also propagate into the variance of the residual.

$$\begin{aligned} K_i &= (\mathbf{v}^T \boldsymbol{\Sigma}_i^{-1} \mathbf{v})^{-1} \\ &= \left([v_1 v_2 \dots v_3] \begin{bmatrix} \sigma_{i1}^{-2} & 0 & \dots & 0 \\ 0 & \sigma_{i2}^{-2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_{iJ}^{-2} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_J \end{bmatrix} \right)^{-1} \\ &= \left(v_1^2/\sigma_{i1}^2 + v_2^2/\sigma_{i2}^2 + \dots + v_J^2/\sigma_{iJ}^2 \right)^{-1} \\ &= \frac{1}{v_1^2/\sigma_{i1}^2 + v_2^2/\sigma_{i2}^2 + \dots + v_J^2/\sigma_{iJ}^2} \end{aligned} \tag{3.17}$$

For convenience, for the remainder of this derivation, the subscript i will be removed and it will be assumed that the equations refer to a particular set of replicates.

To determine the standardization factors, it is necessary to calculate the covariance matrix for $\boldsymbol{\Delta}\mathbf{x}$, whose diagonal elements will give the variance of the individual terms. This can be done by using either propagation of error and matrix derivatives or by using the expected value. The derivation shown in Equation 3.18 is based on the latter [4]. The constant multiplicative factor, $(\mathbf{I} - \mathbf{P})$, is replaced by $\mathbf{M}$ for simplicity.

$$\begin{aligned} \mathrm{Cov}(\boldsymbol{\Delta}_\mathbf{x}) &= E[(\boldsymbol{\Delta}_\mathbf{x} - E[\boldsymbol{\Delta}_\mathbf{x}])(\boldsymbol{\Delta}_\mathbf{x} - E[\boldsymbol{\Delta}_\mathbf{x}])^T] \\ &= E[(\mathbf{Mx} - E[\mathbf{Mx}])(\mathbf{Mx} - E[\mathbf{Mx}])^T] \\ &= E[(\mathbf{Mx} - \mathbf{M\bar{x}})(\mathbf{Mx} - \mathbf{M\bar{x}})^T] \\ &= E[\mathbf{M}(\mathbf{x} - \bar{\mathbf{x}})\mathbf{M}^T(\mathbf{x} - \bar{\mathbf{x}})^T] \\ &= \mathbf{M}E[(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^T]\mathbf{M}^T \\ &= \mathbf{M\Sigma M}^T \\ &= (\mathbf{I} - \mathbf{P})\boldsymbol{\Sigma}(\mathbf{I} - \mathbf{P})^T \\ &= \boldsymbol{\Sigma}_{\boldsymbol{\Delta}_x} \end{aligned} \tag{3.18}$$

In this equation, $\boldsymbol{\Sigma} = E[(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^T] = E[(\mathbf{x} - E(\mathbf{x}))(\mathbf{x} - E(\mathbf{x}))^T]$ [4] represents the error covariance matrix of $\mathbf{x}$. The standardization factors can be determined directly from the diagonal elements of $\boldsymbol{\Sigma}_{\boldsymbol{\Delta}\mathbf{x}}$, but it is convenient to reduce theses to algebraic expressions for greater clarity. Expanding the terms leads to Equations 3.19 and 3.20.

$$\begin{aligned}
(\mathbf{I}-\mathbf{P})\boldsymbol{\Sigma} &= (\mathbf{I}-K\mathbf{v}\mathbf{v}^T\boldsymbol{\Sigma}^{-1})\boldsymbol{\Sigma} \\
&= \boldsymbol{\Sigma} - K\mathbf{v}\mathbf{v}^T \\
&= \begin{bmatrix} \sigma_1^2 & 0 & \dots & 0 \\ 0 & \sigma_2^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_J^2 \end{bmatrix} - K \begin{bmatrix} v_1^2 & v_1v_2 & \dots & v_1v_J \\ v_1v_2 & v_2^2 & \dots & v_2v_J \\ \vdots & \vdots & \ddots & \vdots \\ v_1v_J & v_2v_J & \dots & v_J^2 \end{bmatrix} \\
&= \begin{bmatrix} \sigma_1^2 - Kv_1^2 & -Kv_1v_2 & \dots & -Kv_1v_J \\ -Kv_1v_2 & \sigma_2^2 - Kv_2^2 & \dots & -Kv_2v_J \\ \vdots & \vdots & \ddots & \vdots \\ -Kv_1v_J & -Kv_2v_J & \dots & \sigma_J^2 - Kv_J^2 \end{bmatrix}
\end{aligned} \tag{3.19}$$

$$\begin{aligned}
(\mathbf{I}-\mathbf{P})^T &= \mathbf{I}^T - (K\mathbf{v}\mathbf{v}^T\boldsymbol{\Sigma}^{-1})^T \\
&= \mathbf{I} - \boldsymbol{\Sigma}^{-1}\mathbf{v}\mathbf{v}^T K \\
&= \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} - K \begin{bmatrix} \sigma_1^{-2} & 0 & \dots & 0 \\ 0 & \sigma_2^{-2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_J^{-2} \end{bmatrix} \begin{bmatrix} v_1^2 & v_1v_2 & \dots & v_1v_J \\ v_1v_2 & v_2^2 & \dots & v_2v_J \\ \vdots & \vdots & \ddots & \vdots \\ v_1v_J & v_2v_J & \dots & v_J^2 \end{bmatrix} \\
&= \begin{bmatrix} 1 - Kv_1^2/\sigma_1^2 & -Kv_1v_2/\sigma_1^2 & \dots & -Kv_1v_J/\sigma_1^2 \\ -Kv_1v_2/\sigma_2^2 & 1 - Kv_2^2/\sigma_2^2 & \dots & -Kv_1v_J/\sigma_2^2 \\ \vdots & \vdots & \ddots & \vdots \\ -Kv_1v_J/\sigma_J^2 & -Kv_2v_J/\sigma_J^2 & \dots & 1 - Kv_J^2/\sigma_J^2 \end{bmatrix}
\end{aligned} \tag{3.20}$$

Combining the equations above gives Equation 3.21.

$$
\begin{aligned}
\mathbf{\Sigma_{\Delta x}} &= (\mathbf{I}-\mathbf{P})\mathbf{\Sigma}(\mathbf{I}-\mathbf{P})^T \\
&= \begin{bmatrix} \sigma_1^2 & -Kv_1v_2 & \dots & -Kv_1v_J \\ -Kv_1v_2 & \sigma_2^2 - Kv_2^2 & \dots & -Kv_2v_J \\ \vdots & \vdots & \ddots & \vdots \\ -Kv_1v_J & -Kv_2v_J & \dots & \sigma_J^2 - Kv_J^2 \end{bmatrix} \\
&\quad \times \begin{bmatrix} 1 - Kv_1^2/\sigma_1^2 & -Kv_1v_2/\sigma_1^2 & \dots & -Kv_1v_J/\sigma_1^2 \\ -Kv_1v_2/\sigma_2^2 & 1 - Kv_2^2/\sigma_2^2 & \dots & -Kv_1v_J/\sigma_2^2 \\ \vdots & \vdots & \ddots & \vdots \\ -Kv_1v_J/\sigma_J^2 & -Kv_2v_J/\sigma_J^2 & \dots & 1 - Kv_J^2/\sigma_J^2 \end{bmatrix}
\end{aligned}
\tag{3.21}
$$

For simplicity, we will only calculate the upper left element of $\mathbf{\Sigma_{\Delta x}}$, which gives the variance in Δx_1. Similar results can be verified for the other diagonal elements. The variance in question can be calculated from the inner product of the first row of the left-hand matrix and the first column of the right-hand matrix. This gives Equation 3.22.

$$
\begin{aligned}
\sigma_{\Delta x_1}^2 &= \sigma_1^2 - Kv_1^2 - Kv_1^2 + K^2v_1^4/\sigma_1^2 + K^2v_1^2v_2^2/\sigma_2^2 + \cdots + K^2v_1^2v_J^2/\sigma_J^2 \\
&= \sigma_1^2 - 2Kv_1^2 + K^2v_1^2\left(\frac{v_1^2}{\sigma_1^2} + \frac{v_2^2}{\sigma_2^2} + \cdots + \frac{v_J^2}{\sigma_J^2}\right) \\
&= \sigma_1^2 - 2Kv_1^2 + K^2v_1^2 \cdot K^{-1} \\
&= \sigma_1^2 - Kv_1^2 \\
&= \sigma_1^2 - \frac{v_1^2}{\frac{v_1^2}{\sigma_1^2} + \frac{v_2^2}{\sigma_2^2} + \cdots + \frac{v_J^2}{\sigma_J^2}} \\
&= \frac{v_1^2 + v_2^2\frac{\sigma_1^2}{\sigma_2^2} + \cdots + v_J^2\frac{\sigma_1^2}{\sigma_J^2} - v_1^2}{\frac{v_1^2}{\sigma_1^2} + \frac{v_2^2}{\sigma_2^2} + \cdots + \frac{v_J^2}{\sigma_J^2}} \\
&= \sigma_1^2 \cdot \frac{\frac{v_2^2}{\sigma_2^2} + \cdots + \frac{v_J^2}{\sigma_J^2}}{\frac{v_1^2}{\sigma_1^2} + \frac{v_2^2}{\sigma_2^2} + \cdots + \frac{v_J^2}{\sigma_J^2}} \\
&= \sigma_1^2 \cdot \frac{\sum_{i \neq 1} \frac{v_i^2}{\sigma_i^2}}{\sum_i \frac{v_i^2}{\sigma_i^2}}
\end{aligned}
\tag{3.22}
$$

This is one form of the expected variance of the residuals, but a more convenient form which is easily derived from the above is given in Equation 3.23 in general form applicable to every sample and feature.

$$\sigma^2_{\Delta x_{iK}} = \sigma^2_{iK} \cdot \left(1 - (v_K^2/\sigma^2_{iK}) / \sum_{j=1}^{J} v_j^2/\sigma^2_{ij}\right) = \sigma^2_{iK} \cdot \eta_{iK} \tag{3.23}$$

For a given measurement, x_{iK}, where i is the variable number and K is the replicate number, this equation gives the expected variance of the residual in terms of the measurement uncertainty from the error model (σ_{iK}) and an adjustment factor, η_{iK}. The adjustment factor can be regarded as a correction for degrees of freedom and depends on both the normalization vector, $\mathbf{v}$, and the uncertainties of each of the measurements in replicate set.

Although the derivation above appears complex, it can be appreciated intuitively by examining various limiting cases. If we consider the case of four replicates where the normalization factors are all unity ($v_j = v = 0.5$) and the measurements have equal uncertainties ($\sigma_j = \sigma$), the maximum likelihood projection (Equation 3.12) is the mean of the four measurements and Equation 3.23 gives $\eta = \frac{3}{4}$ for each measurement. This is consistent with estimation of variance around a mean, with the expectation value $E[(x - \bar{x})^2] = (N-1)\sigma^2/N$. If we modify the circumstances so that one of the measurements is missing (e.g., $\sigma_4 = \text{inf}$), then the projected vector is the mean of the other three measurements, with $\eta = \frac{2}{3}$ for those measurements and $\eta = 0$ for the missing measurement, which is inconsequential since no residual is calculated for the missing value. In the case of non-uniform measurement uncertainty (but retaining unity normalization factors), the projection becomes a weighted mean, and the adjustment factor reflects the contribution of each measurement in the projection. For example, if $\sigma_1 = 1$, $\sigma_2 = 2$, $\sigma_3 = 3$, and $\sigma_4 = 4$, the resulting adjustment factors are $\eta_1 = 0.298$, $\eta_2 = 0.824$, $\eta_3 = 0.922$, and $\eta_4 = 0.956$. Since more precise measurements are weighted more in the calculation of the projection, they are expected to have smaller residuals relative to their measurement uncertainty, as indicated in the increasing values. Reducing this to the simple bivariate case where $\sigma_y \gg \sigma_x$ (i.e., linear regression), η_y goes to unity and $\sigma_{\Delta y} = \sigma_y$, as expected.

The role of the loading vector, $\mathbf{v}$ (i.e., the normalization factors) in Equation 3.23 also relates to the way that the measurement vector is projected onto the model, with the projection direction being more highly aligned with the axes with higher values of v. Considering the simple bivariate case where $\sigma_x = \sigma_y$, as the slope of the normalization line decreases and becomes more horizontal (v_x increases relative to v_y), the projection becomes more vertical and η_x increases relative to η_y. At the limit of $v_x = 1$ (horizontal line), the projection is vertical, reflecting no variance in Δx, and η_y has a value of unity.

A few final comments can be made regarding Equation 3.23. First, note that, regardless of the parameters, the sum of the adjustment factors will be equal to one less than the number of replicates, $\sum \eta_j = J - 1$. Second, whenever all but one of the measurements in a set is missing, both the residual and the adjustment factor for the single measurement will be zero, so the standardized residual is undefined and should not be included in the distribution. Finally, the standardization should technically include an adjustment for the fact that v is estimated for the measurements, but for a large number of measurements, this should be negligible. The standardization formula given in Equation 3.23 has been tested extensively with simulations and found to be reliable.

3.2.5 Optimization of the Fit Parameters

The procedure described so far consists of providing estimates for the measurement uncertainties, σ_{ij}, using the variance function (e.g., Equation 3.3) with estimated parameters, $\boldsymbol{\beta}$, and then applying these uncertainties to estimate the normalization factors of the replicate datasets as $\mathbf{v}$ through MLPCA (section 3.2.2), with corresponding updates to the uncertainty estimates as described in section 3.2.3. The maximum likelihood estimates and residuals for each measurement are then calculated and standardized as described in section 3.2.4. In principle, if the error model and its parameters are correctly chosen, the standardized residuals should follow a standard normal distribution. Therefore, a suitable objective function is needed to evaluate the distribution of residuals and optimize the model parameters, $\boldsymbol{\beta}$.

In statistical analysis, there are many quantities suitable for assessing the similarity of some data to a standard normal distribution (or any distribution of choice). Examples include the Kolmogorov–Smirnov test or Pearsons χ^2 test [11] [61]. However, both tests require the optimization (*e.g.* minimization) of a test statistic and (or) p-value. While there is no explicit rule preventing such use of test statistics, it may cause unexpected difficulties during the optimization process. Therefore, this thsis explores the less commonly used Kullback–Leibler (KL) divergence as a suitable objective function.

The Kullback–Leibler divergence, also known as relative entropy, is a measure used in information theory to quantify the divergence of one probability distribution from a second, reference probability distribution. In the context of the problem statement provided here, the statistic enables a comparison of the distribution of standardized residuals to a standard normal distribution. Since the normalized residuals cannot be extrapolated to a continuous distribution, it is necessary to apply a discretized version of the KL divergence as defined in Equation 3.24. Here, $Q(x)$ is some reference distribution while $P(x)$ is any probability distribution to be compared [61].

$$D_{KL}(P||Q) = \sum_{x \in X} P(x) \log\left(\frac{P(x)}{Q(x)}\right) dx \tag{3.24}$$

To construct the required discrete probability distributions, an evenly spaced number of bins, n_{bins}, is created based on the first and 99th percentile of a standard normal distribution. The residuals are sorted into bins, counted, and normalized by the total number of residuals (data points), resulting in a discrete estimate of $P(x)$, where x represents the bins. $Q(x)$ is created using the same bins and the reference standard normal distribution. For a single bin $P(x)$ describes the observed fraction of residuals within that bin while $Q(x)$ describes the expected fraction of residuals. This allows the KL-Divergence to be calculated between $P(x)$ and $Q(x)$ (Equation 3.24). A smaller value indicates a closer fit between both distributions;thus, signifying a better fit.

There exists a number of important nuances when using KL-Divergence for optimization. If outliers are present the bins at the tail of the discretized distribution of observations, $P(x)$, may become inflated. This results, even with a small number of outliers, in a higher than expected KL-Divergence which consequently affects the estimation of $\boldsymbol{\beta}$ negatively. A simple remedy is to remove residuals that fall outside of the 1st percentile on the low end and 99th percentile on the high end by simply *cutting off* the tails. After the removal of values, both distributions must be adjusted to integrate (sum) to unity. Another area of concern are bins of $P(x)$ that do not contain any residuals, consequently causing the log term to become invalid. Therefore, any bins with zero-counts are removed from $P(x)$ and $Q(x)$. This is feasible since $\lim_{x \to 0^+} x \log(x) = 0$. The last consideration needs to be given to the number of bins. If n_{bins} is too small the intricacies of the normal distribution may not be captured well enough to provide a good fit. This is especially severe close to the global minimum. On the other hand, choosing n_{bins} too large can decrease the precision of the fit if the normalized residuals do not follow a standard normal distribution closely by over-fitting small discrepancies. Generally, $n_{\text{bins}} = 40$ provides good results and an easy to find global minimum.

The defined objective function, variance model (e.g. Equation 3.3) and non-normalized residuals need to be passed to any optimization (*e.g.* minimization) algorithm of choice. In each optimization step, a new $\boldsymbol{\beta}$ is suggested which, in turn, is used to estimate the (adjusted, section 3.2.4) variance. Residuals are normalized and KL-Divergence is calculated by using a standard normal distribution as the reference. These steps are repeated until convergence to a global minimum (meaning: no significant change in $\boldsymbol{\beta}$ and the objective function). Repeated test on simulated data suggest that the Nelder-Mead Simplex algorithm [36] reliably finds the global optimum and accurately estimates $\boldsymbol{\beta}$.

3.2.6 Iterative Optimization of Fit Parameters and Scaling Factors

The previous sections provided a thorough overview of the individual pieces required to build an algorithm that iteratively optimizes fit parameters, $\boldsymbol{\beta}$, and scaling factors, $\mathbf{v}$. To recall, this iterative process is necessary due to the dependence of MLPCA on

$\boldsymbol{\beta}$ and the dependence of $\boldsymbol{\beta}$ on $\mathbf{v}$ (a result from MLPCA). Figure 3.3 visualizes the algorithm by means of a flow chart.

As a starting point the scaling vector, $\mathbf{v}$, and parameter estimates, $\boldsymbol{\beta}$, must be initialized using educated guesses. The initial values are passed to dynamic 1-Dimensional MLPCA (refer to section 3.2.3) to calculate $\mathbf{u}$, S, $\mathbf{v}$, and, subsequently, $\hat{\mathbf{X}}$. The residuals ($\mathbf{R} = \mathbf{X} - \hat{\mathbf{X}}$) allow estimation of a new $\boldsymbol{\beta}$ (refer to section 3.2.5). If at least one parameter (of $\boldsymbol{\beta}$ or $\mathbf{v}$) exhibits a percent change in-between iterations that exceeds the set convergence limit, a new iteration is started; else the current optimization is completed.

Given the inconsistencies within LC-MS data, the algorithm contains mechanisms to treat missing values and outliers. With respect to the former issue the algorithm can handle any rows in $\mathbf{X}$ that provide at least one value; the minimum required for MLPCA to estimate a projection. All other rows in $\mathbf{X}$ must be discarded. The latter issue is treated by progressing into a new iteration (further called outlier pass) after setting any values in $\mathbf{X}$ with normalized residuals above four to NA. This approach of handling outliers (in addition to removing inflated tails while fitting the parameters) aims to curb the effect of extreme values on MLPCA. Should this cause a single row to have less than one available data point, the row is discarded.

Once a single iteration, including passes to remove outliers, is finished, it is recommended to start a new iteration with freshly initialized guesses for $\mathbf{v}$ and $\boldsymbol{\beta}$ while resetting the data matrix, $\mathbf{X}$. This practice is useful to combat scenarios in which a single iteration does not find the global optimum due to poor initial estimates. Such case will also lead to incorrect removal of outliers in each outlier pass, essentially nullifying the results. After performing multiple, independent runs and retaining the results, the run with the lowest final objective function may be retained.

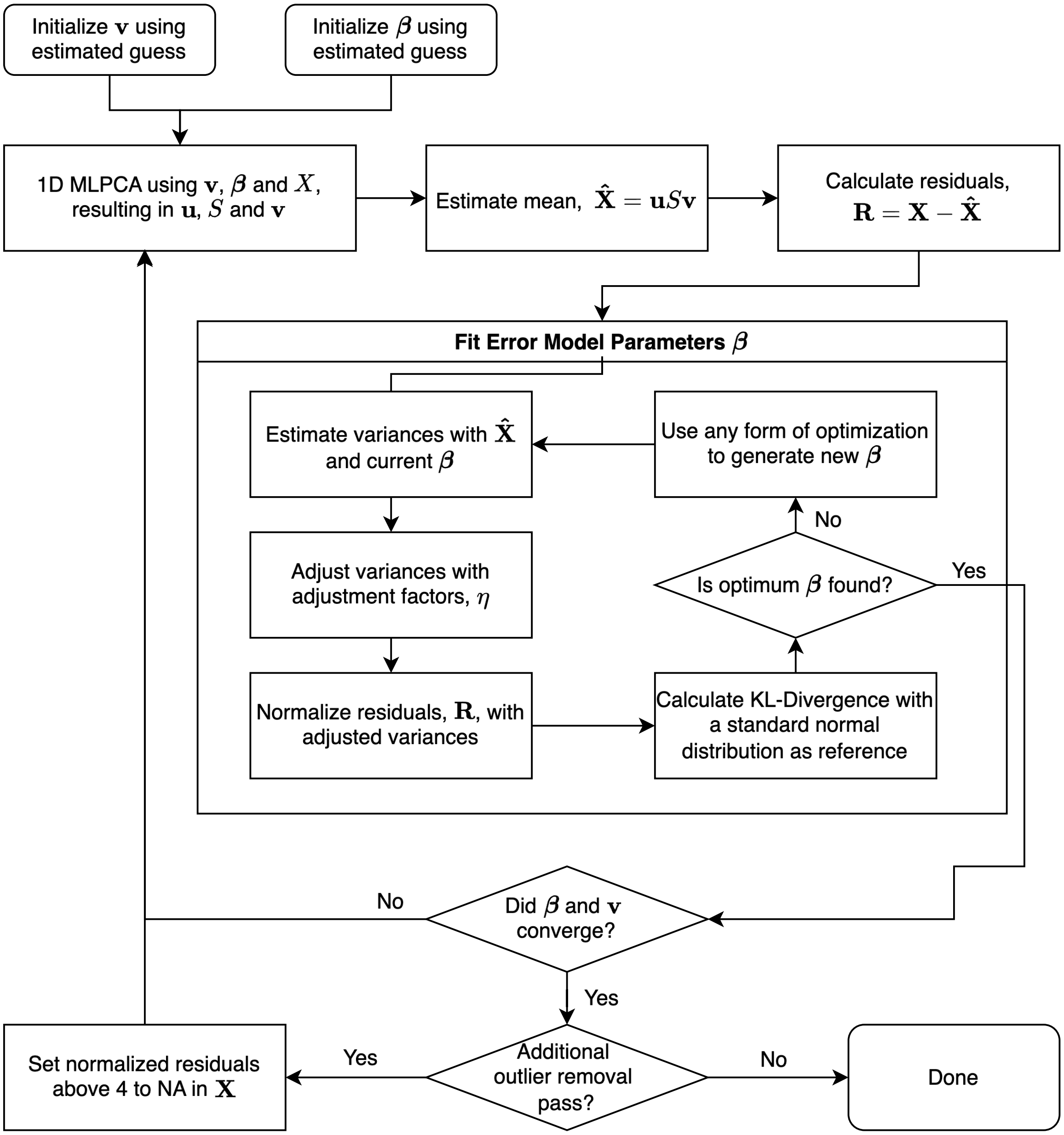
Initialize $\mathbf{v}$ using estimated guess
Initialize $\boldsymbol{\beta}$ using estimated guess
1D MLPCA using $\mathbf{v}$, $\boldsymbol{\beta}$ and X, resulting in $\mathbf{u}$, S and $\mathbf{v}$
Estimate mean, $\hat{\mathbf{X}} = \mathbf{u}S\mathbf{v}$
Calculate residuals, $\mathbf{R} = \mathbf{X} - \hat{\mathbf{X}}$
Fit Error Model Parameters $\boldsymbol{\beta}$
Estimate variances with $\hat{\mathbf{X}}$ and current $\boldsymbol{\beta}$
Use any form of optimization to generate new $\boldsymbol{\beta}$
No
Adjust variances with adjustment factors, η
Is optimum $\boldsymbol{\beta}$ found?
Yes
Normalize residuals, $\mathbf{R}$, with adjusted variances
Calculate KL-Divergence with a standard normal distribution as reference
No
Did $\boldsymbol{\beta}$ and $\mathbf{v}$ converge?
Yes
Yes
Set normalized residuals above 4 to NA in $\mathbf{X}$
Additional outlier removal pass?
No
Done

Figure 3.3: Flow chart visualizing the algorithm to iteratively fit variance function parameters ($\boldsymbol{\beta}$) and scaling factors ($\mathbf{v}$).

3.3 Results and Discussion

3.3.1 Simulation Studies

The data used in this section have been generated based on the following specifications. First, a hypothetical reference experiment is considered and represented by a column vector, $\mathbf{x}_{\text{ref}}$, (Ix1), corresponding to a single sample with I measurements. Then, the complete data matrix is obtained by the multiplication: $\mathbf{x}_{\text{ref}}\boldsymbol{\alpha}$, where $\boldsymbol{\alpha}$ is a row vector (1xJ) of scaling factors for each of the replicates J, giving rise to the matrix $\mathbf{X}_{base}$ of dimension IxJ. Second, a structured error is injected into $\mathbf{X}_{\text{base}}$ following Equation 3.25.

$$(\mathbf{X}_{\text{sim}})_{ij} = (\mathbf{X}_{\text{base}})_{ij} + \sqrt{VF((X_{base}))}_{ij}\epsilon_{ij}, \text{where } \epsilon \sim N(0,1) \tag{3.25}$$

In this equation, VF consists of parameters β_1 (shot noise component) and β_2 (proportional/flicker noise component) as defined in Equation (3.3) and which, for illustration purposes are set to 1 and 0.1 respectively. To account for sensitivity differences between replicate measurements, arbitrary scaling factors $\boldsymbol{\alpha} = (0.6, 0.8, 1.0, 1.2, 1.4)$ were set.

To recover these values, we used the model outlined in Figure 3.3 with 5 outlier passes, running the calculations up to 10 times (individually) and retaining the calculation with the lowest objective function. This has been done to avoid the algorithm getting stuck in local minima. After examining the detailed results for a single realization of error and to be able to asses accuracy and precision, different realizations of error were fit with the same model ten times; leading to ten individual estimates of $\boldsymbol{\beta}$. The code was executed in Julia [12].

The Ideal Case

In an ideal scenario the collected data are free of missing or outlying observations; equivalent to the unmodified, simulated data matrix $\mathbf{X}_{sim}$. The error model is fit to a single realization of error to examine diagnostics. The fit results in β_1 and β_2 to be 1.022 and 0.095, respectively. The scaling factors, obtained from MLPCA, are

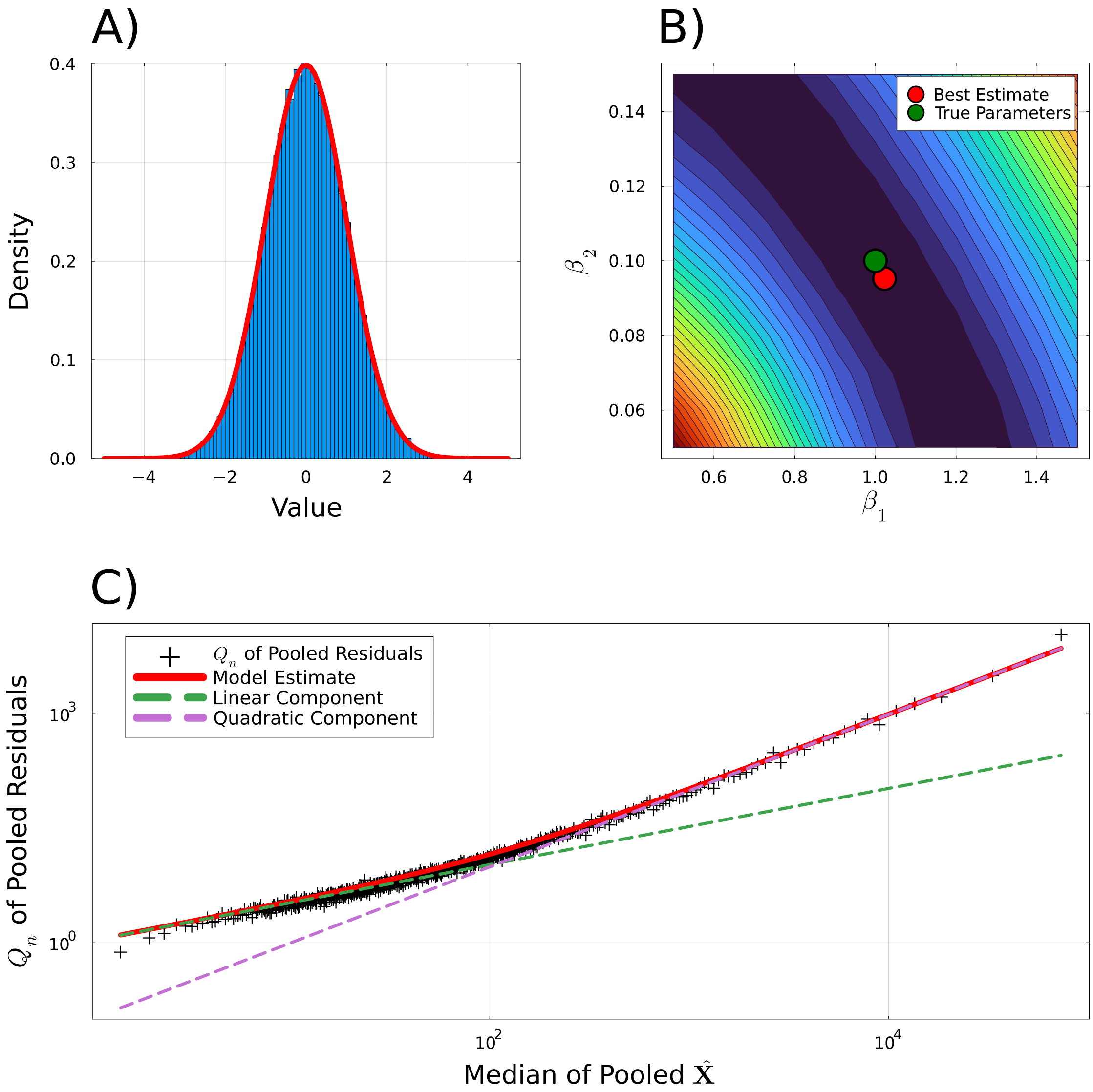

Figure 3.4: Diagnostic plots based on the error model fit to an ideal dataset following the introduced algorithm. **A)** A density histogram of the normalized residuals which should follow a standard normal distribution in a successful fit. The density of a standard normal distribution is overlaid in red. **B)** Contour plot showing the best estimate of the parameters, β_1 and β_2, as well as the true parameters (1 and 0.1, respectively). A dark blue corresponds to a low objective function. A bright red corresponds to a high objective function. **C)** The error model (red) superimposed on the robust standard deviation (Q_n) of the residuals. The axes are log-scaled. The quadratic and linear component are shown in purple and green, respectively.

$\mathbf{v} = (0.2586, 0.3439, 0.4298, 0.5162, 0.6034)$. Normalizing the scaling factors by 0.4298 recreates the true $\boldsymbol{\alpha} = (0.6, 0.8, 1.0, 1.2, 1.4)$. Figure 3.4 visualizes the results for the same realization of error. Subplot A provides evidence of the viability of the algorithm by showing that the normalized residuals follow a standard normal distribution. Further, subplot C provides visual evidence of a useful fit by showing the pooled Q_n [83] (robust estimate of standard deviation) against the median of pooled $\hat{\mathbf{X}}$ [1] [5]. The contour plot shown in subplot B underlines that the method for optimization of the variance model parameters is suitable to find the minimum. However, it also depicts a large area of objective function minima which may cause discrepancies between the true and fitted parameters, making multiple iterations with newly initialized starting values even more important.

Figures 3.7A and 3.8A show stability of the error model parameter and scaling factor estimates across multiple realizations of error. A slight bias towards the high end for β_1 and the low end for β_2 is observable. This trend stands in direct correlation with the trough of minima in Figure 3.4B and is most likely attributable to the optimization getting stuck in local minima. This issue can be resolved by adjusting the objective function (*e.g.* tuning n_{bins}) or using a different minimization algorithm; potential subjects of future research.

Missing Data

To simulate the effect of missing data on the model, 5% of data points have been randomly chosen and removed (set to NA). Any rows (channels) that result in less than two available data points are discarded due to inability of MLPCA to estimate a projection. The fit of a single error realization determines β_1 and β_2 to be 1.093 and 0.0848, respectively. The scaling factors are $\mathbf{v} = (0.2584, 0.3439, 0.4296, 0.5175, 0.6022)$. Normalizing by 0.4296 recreates the true $\boldsymbol{\alpha} = (0.6, 0.8, 1.0, 1.2, 1.4)$. Visual results of the fit are provided in Figure 3.5. All subplots follow the outlined expectations. Thus, it can be said that the error model remains viable for data with missing values. This is expected due to (1.) random removal of data points not influencing the

[1] In detail, the values of $\hat{\mathbf{X}}$ and $\mathbf{X} - \hat{\mathbf{X}}$ are flattened into a row vector ($\hat{\mathbf{x}}$ and $\mathbf{r}$) and are sorted so that the values in $\hat{\mathbf{x}}$ increase while also applying the same order to the residual vector. Then the Q_n [83] and median are calculated for bins of 100 for $\mathbf{r}$ and $\hat{\mathbf{x}}$, respectively.

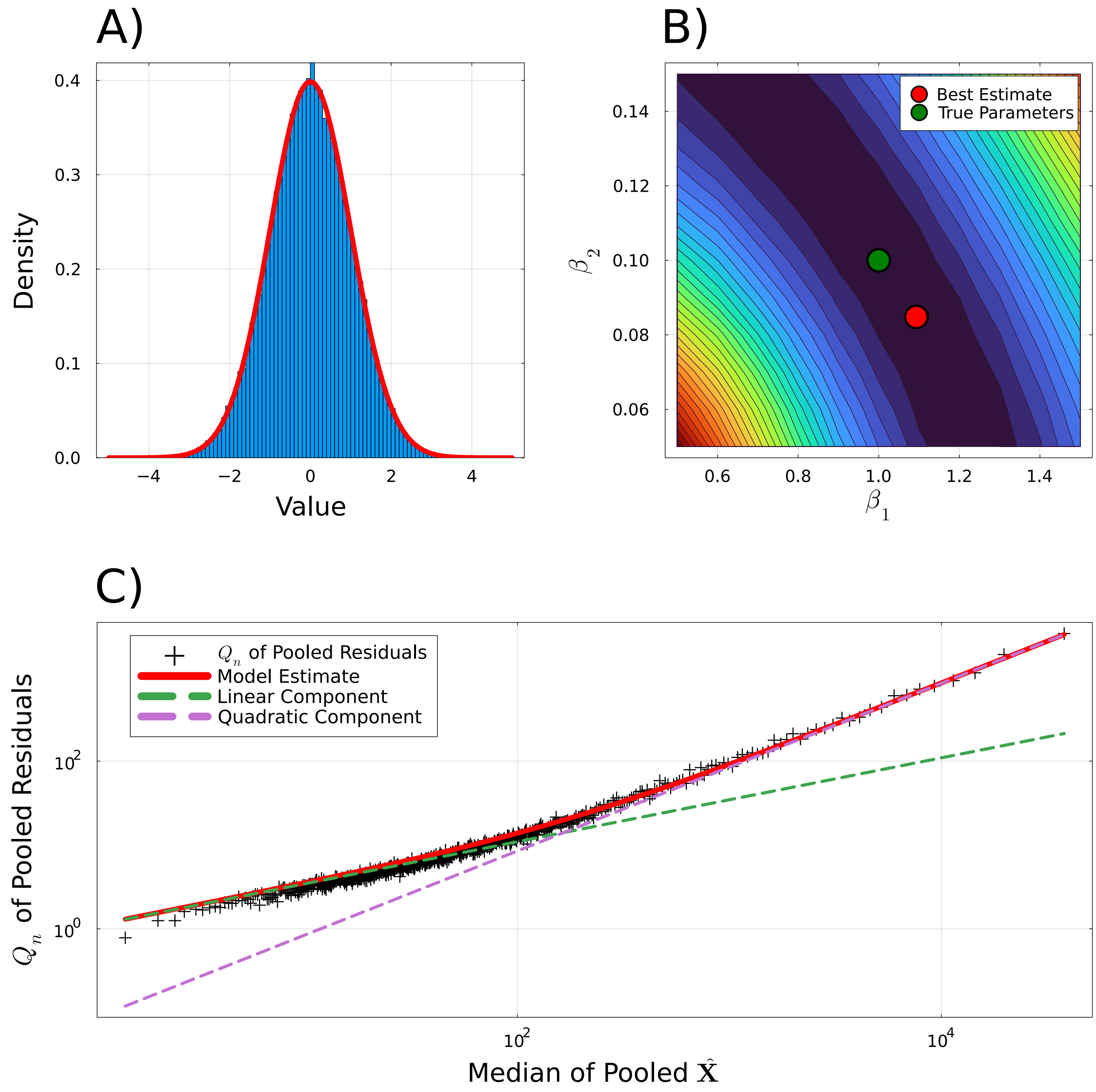

Figure 3.5: Diagnostic plots based on the error model fit to data with 5% of missing values following the introduced algorithm. **A)** A density histogram of the normalized residuals which should follow a standard normal distribution in a successful fit. The density of a standard normal distribution is overlaid in red. **B)** Contour plot showing the best estimate of the parameters, β_1 and β_2, as well as the true parameters (1 and 0.1, respectively). A dark blue corresponds to a low objective function. A bright red corresponds to a high objective function. **C)** The error model (red) superimposed on the robust standard deviation (Q_n) of the residuals. The axes are log-scaled. The quadratic and linear component are shown in purple and green, respectively.

assumption of normality of the remaining residuals and (2.) variances accounting for missing values with the adjustment factor, η, propagating into a low weight for affected points in MLPCA. However, if missing values are correlated (for example due to a faulty LC-MS device), the model may be unable to handle such cases due to the normal distribution being affected heavier in certain regions.

Figure 3.7B and 3.8B indicate a similar trend as observed in the previous simulation. β_1 is seemingly biased towards the high-end whereas β_2 is slightly biased towards the low-end, explained by the same trough seen in Figure 3.5B. While it may seem that the bias is lower than in the simulation of ideal data, this difference is most likely caused by a small sample size. The scaling factors are stable across different realizations of errors due to their low sensitivity to small changes in model parameters.

Outliers

Outliers may cause failure of the model as their effect potentially voids the assumption of normality of the standardized residuals. This concern necessitates an examination of the models robustness. To simulate outliers, 5% of data points are chosen at random and have their standard deviation inflated 10-fold. It is to note that this is a crude method of simulating outliers and does not account for more complex cases, such as correlated outliers. After fitting the model on a single error realization the resulting β_1 and β_2 are found to be 1.0208 and 0.1037, respectively. The scaling factors are $\mathbf{v} = (0.2575, 0.3435, 0.4294, 0.5162, 0.6040)$. Normalizing by 0.4294 recreates the true $\boldsymbol{\alpha} = (0.6, 0.8, 1.0, 1.2, 1.4)$.

Visual results are provided in Figure 3.6. Subplot A provides evidence that the algorithm handles outliers well as their presence does not heavily skew the resulting normalized residuals. This is expected due to the removal of values that would show up in the tail of the distribution. Future work may focus on simulating the impact of outliers that inflate the tails in regions that are not cutoff. Slightly heavier tails are observable (when looking closely) but can be considered meaningless. Subplot C provides a useful fit and does not show any signs of outliers due to choosing a robust

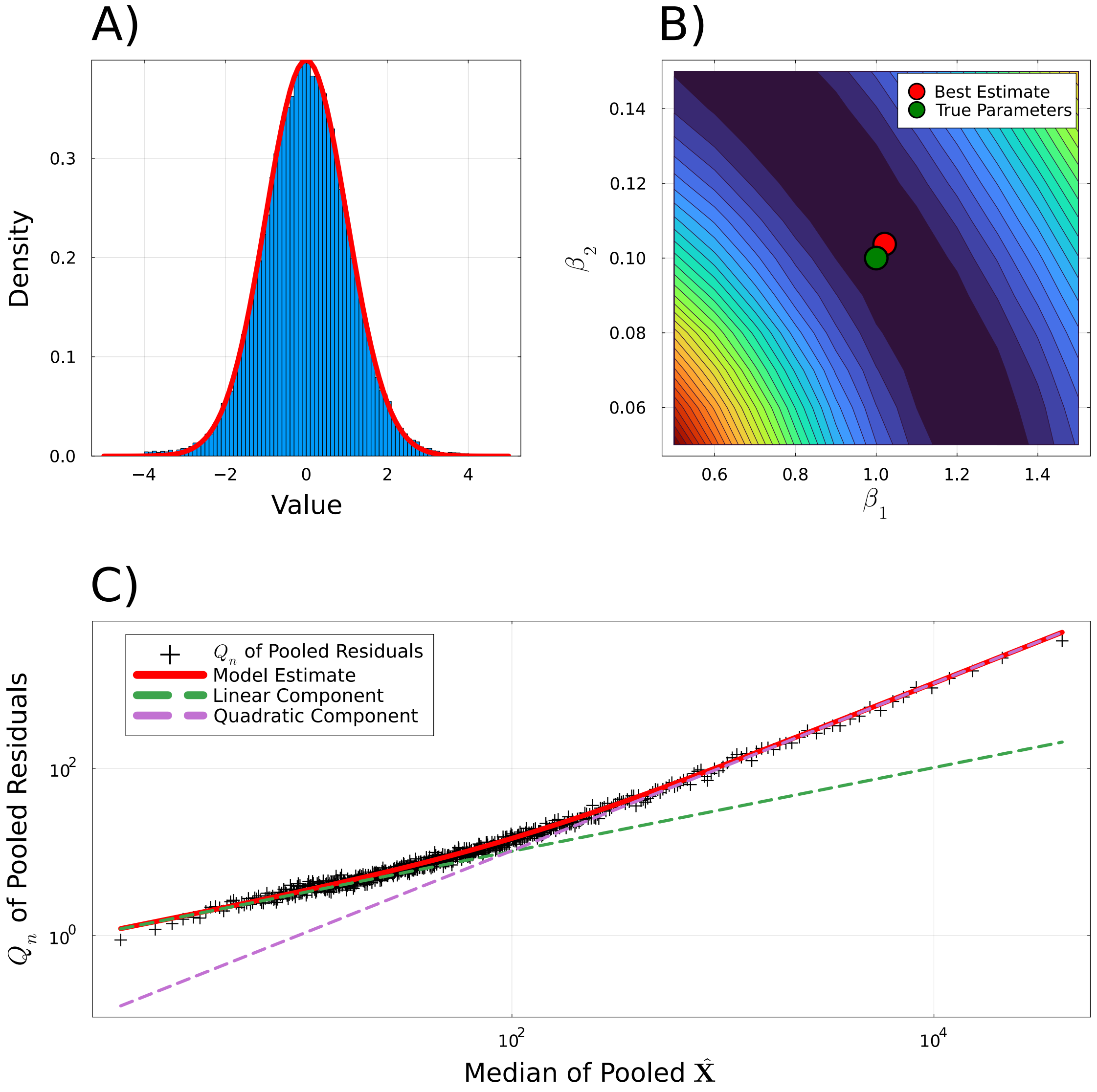

Figure 3.6: Diagnostic plots based on the error model fit to data with 5% of outlying values following the introduced algorithm. **A)** A density histogram of the normalized residuals which should follow a standard normal distribution in a successful fit. The density of a standard normal distribution is overlaid in red. **B)** Contour plot showing the best estimate of the parameters, β_1 and β_2, as well as the true parameters (1 and 0.1, respectively). A dark blue corresponds to a low objective function. A bright red corresponds to a high objective function. **C)** The error model (red) superimposed on the robust standard deviation (Q_n) of the residuals. The axes are log-scaled. The quadratic and linear component are shown in purple and green, respectively.

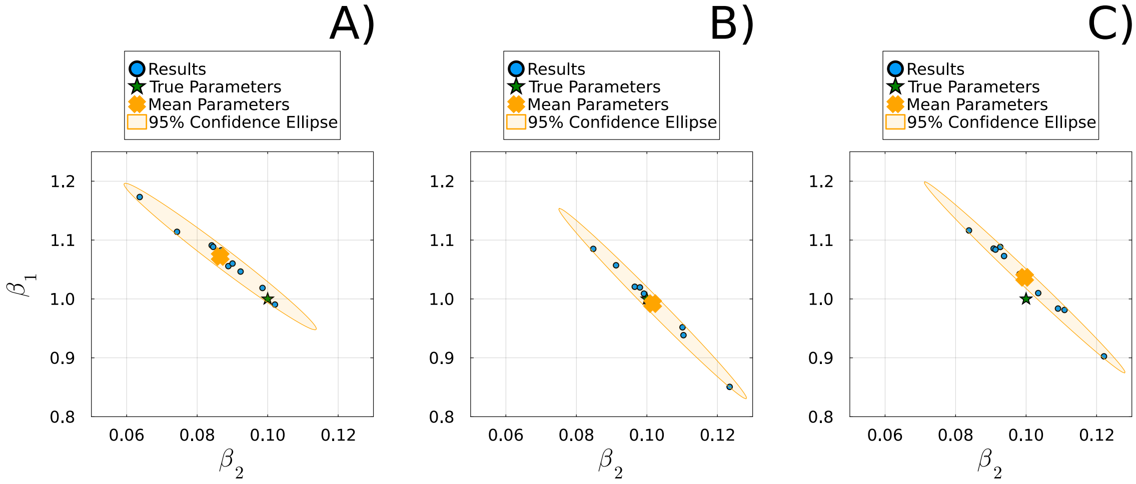

Figure 3.7: Estimate of the fit parameters, $\boldsymbol{\beta}$, for different realizations of error. **A)** The ideal case. **B)** 5% of data points are missing. **C)** 5% of data points are outliers. The orange ellipse indicates the 95% confidence interval around the mean estimate.

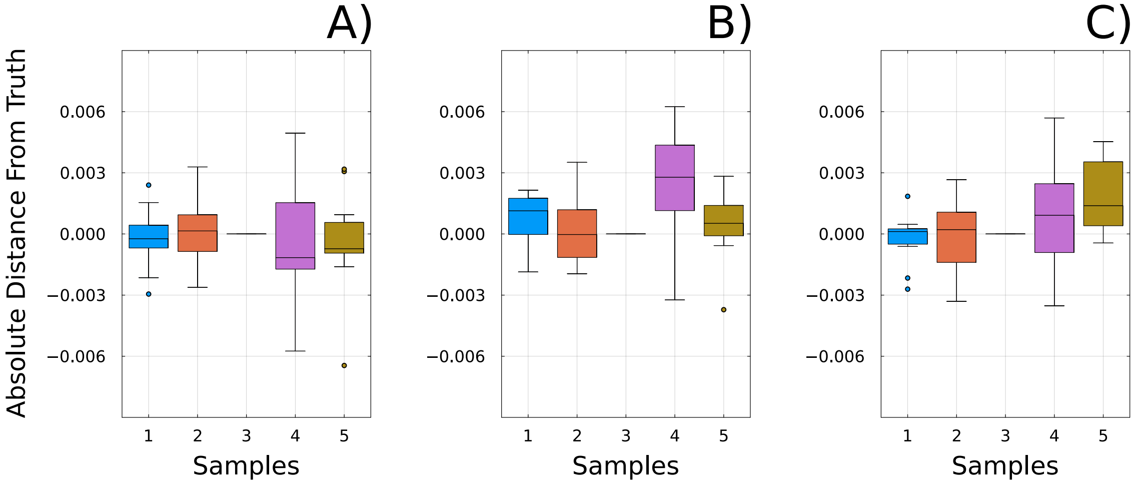

Figure 3.8: Absolute distance of the scaling factors, $\boldsymbol{\alpha}$, from their true value for different realizations of error. The scaling factors were normalized by the third sample; thus this sample has no derivation from the true value. **A)** The ideal case. **B)** 5% of data points are missing. **C)** 5% of data points are outliers.

estimate of the standard deviations for visualization. The trough seen in subplot B is similar to the previous simulations.

Results of repeated realizations of error are found in Figure 3.7C and 3.8C. The plots strongly mirror the observations from the previous simulations indicating robustness of the algorithm. This robustness can be attributed to the strong cutoff of non-normal residuals during the calculation of the objective function (see section 3.2.5) as well as the addition of outlier passes.

The simulation studies provide evidence for three key aspects of the model: Accuracy, Precision and Robustness. This underlines the models suitability to be applied to experimental, non-simulated data. However, future work may need to focus on optimizing the minimization algorithm to more reliably find the true minimum (improving accuracy, increasing precision, decreasing correlation in β_1 and β_2), exploring the effect of fitting a wrong error model (under-/over-estimating variances) and understanding different cases of outliers and missing values (correlated outliers/missing values, different magnitudes).

3.3.2 Experimental Data

After performing simulation studies, the model is tested on three experimental datasets. The selected data varies in quality (sample pre-processing, missing values) and size (how many data points). For each dataset the algorithm was run ten times. Each iteration was executed with 5 outlier passes. The run resulting in the minimum objective function was retained. The code was executed in Julia [12].

MSREPS

The *MSREPS* dataset has been collected by Nickerson et al. as part of their research on the impact of different sample preparation techniques on the observed variance in LC-MS experiments. *MSREPS* is used as a baseline standard with five collected, replicated, pooled and detergent-free samples. Detailed sample preparation procedures can be found in the publication [72]. Due to the marginal processing, this

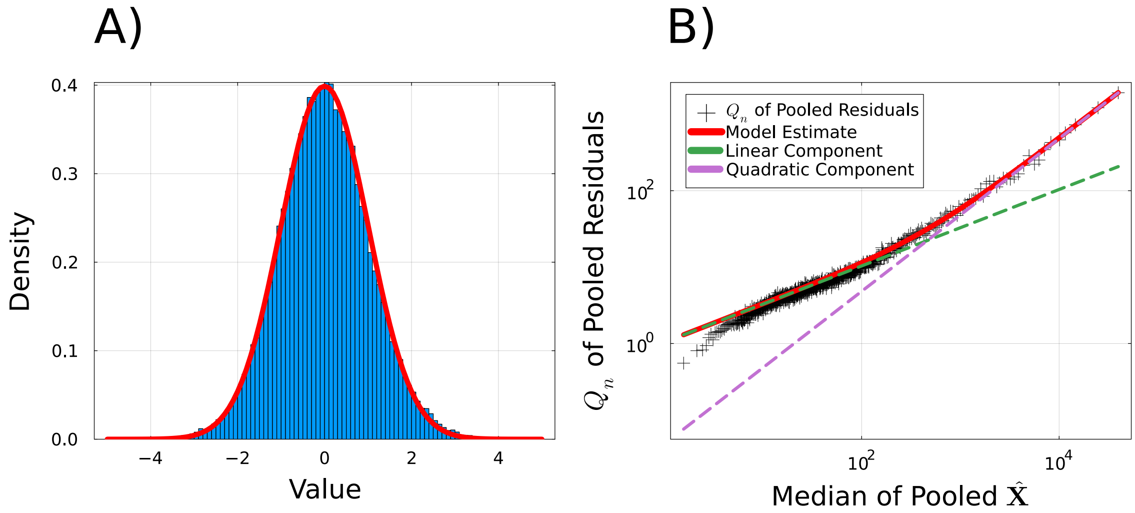

Figure 3.9: Diagnostic plots for the fit of the *MSREPS* dataset. **A)** A density histogram of the normalized residuals with the true standard normal distribution overlaid in red. **B)** The error model (red) superimposed on the pooled Q_n of the residuals plotted against the pooled median of $\hat{\mathbf{X}}$. The linear and quadratic component of the model are indicated using dashed lines. The axes are log-scaled.

dataset is expected to be the well-behaved. It consists of 19,255 measured channels and five samples such that $\mathbf{X}_{19255 \text{ x } 5}$ and, after removing channels (rows) with less than two available values, the dimension of $\mathbf{X}$ shrinks to 16,078 by 5 and still contains 10,499 missing values.

After fitting the error model, the estimated values of β_1 and β_2 are 1.0369 and 0.0482, respectively, with scaling factors: $\mathbf{v} = (0.461, 0.4588, 0.4378, 0.4447, 0.4330)$. Dividing by an arbitrary number, say the maximum, results in the following (easier to digest) scaling factors: $\boldsymbol{\alpha} = (1.000, 0.9953, 0.9497, 0.9650, 0.9391)$. This indicates a small difference in sensitivity between samples. Diagnostic plots are shown in Figure 3.9. Subplot A visualizes an ideal fit of the normalized residuals to a standard normal distribution. This represents the most important metric, indicating a functional objective function and algorithm in the case of true experimental data. Subplot B visualizes the pooled robust standard deviation (Q_n) [83] against the pooled median of $\hat{\mathbf{X}}$. The figure provides an acceptable model fit for the majority of data. However, there are slight discrepancies to note at the lower end of the fit where the VF overestimates the true variances. A potential cause could be error introduced by sources other than shot noise or flicker noise. If that is the case, this result would call for an extension of the fitted VF. Additionally, this observation could be caused by signal processing anomalies for low intensity signals; a problem that does not have a simple remedy but can also be considered insignificant.

In comparison, the publication by Nickerson et al. reported a β_1 of 0.74 and a β_2 of 0.0673 [72]. The estimate of the proportional error (β_2) is comparable while the shot noise parameter (β_1) varies by a bigger margin. The difference in parameter estimates is most likely caused by the difference in modelling methods (Nickerson used locally pooled error models) and thus becomes hard to compare.

INGEL

The *INGEL* dataset was collected as part of the same research by Nickerson et al. [72]. Section 2.5 of their publication indicates the extraction procedure in detail; importantly containing the use of a detergent and extraction procedure. Therefore, the data is expected to be noiser than *MSREPS*. Four samples were collected in this manner [72]. The dataset contains 19,376 measurement channels of which only 11,322 remain after removing rows with less than two available data points. Of the total 45,288 data points 11135 are not available. Fitting the variance function results in β_1 to be 0.8915 and β_2 to be 0.1555. The scaling factors come out as $\mathbf{v} = (0.5112, 0.4808, 0.6160, 0.3579)$. Scaling them by the maximum provides more reader friendly provides $\boldsymbol{\alpha} = (0.8300, 0.7806, 1.000, 0.5810)$. Contrary to the *MSREPS* dataset, the sensitivity difference are larger with the fourth sample exhibiting almost half the sensitivity as the third sample. This is to be expected given the more intricate sample preparation procedure. Figure 3.10 visualizes the diagnostic plots with the same contents as Figure 3.9. Subplot A does not indicate any noticeable deviations from normality while subplot B represents a reasonable fit. At the low end, a similar observation regarding overestimation of the variance can be made.

Nickerson et al. reported β_1 as 0.6928 and β_2 as 0.1542 [72]. A similar trend as with the MSREPS dataset is observable; β_1 is lower and β_2 is comparable in Nickerson's findings.

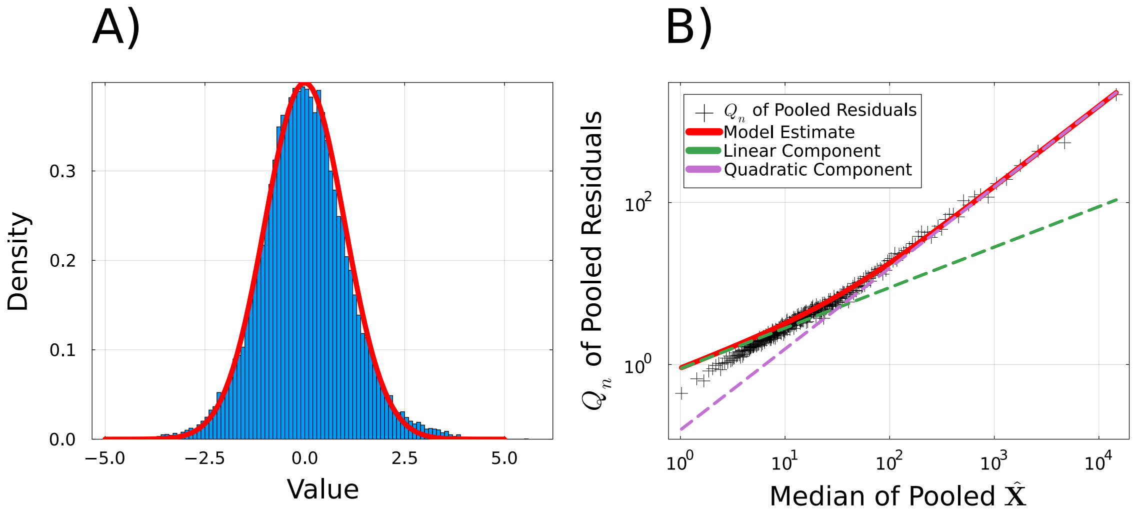

Figure 3.10: Diagnostic plots for the fit of the *INGEL* dataset. **A)** A density histogram of the normalized residuals with the true standard normal distribution overlaid in red. **B)** The error model (red) superimposed on the pooled Q_n of the residuals plotted against the pooled median of $\hat{\mathbf{X}}$. The linear and quadratic component of the model are indicated using dashed lines. The axes are log-scaled.

TNBC

The *TNBC* dataset was collected as part of a study of triple-negative breast cancer (TNBC) by Lixian Li et al. Blood serum from patients was passed through ultra-high-performance liquid chromatography-high resolution mass spectrometry [59]. The dataset consists of a total of eight replicated samples which were solely collected for quality control and should not contain any biological variability. Apart from enabling comparisons to different processing techniques, the fitted error model can further be applied to samples containing biological variation to assist in methods such as MLPCA. The data is of dimension $1,856$ by 8 and does not require any removal of rows. Of the 14,848 data points, 2903 are entered as NA. However, the overall sample size comparatively low for this data. After fitting the variance function β_1 results in 13.2147 and β_2 results in 0.0477. The scaling factors are $\mathbf{v} = (0.4123, 0.3771, 0.3662, 0.3526, 0.3363, 0.3281, 0.3211, 0.3247)$, and normalized by the maximum result in $\boldsymbol{\alpha} = (1.000, 0.9145, 0.8881, 0.8553, 0.8156, 0.7957, 0.7788, 0.7874)$. The sensitivity differences are acceptable. Figure 3.11 provides the resulting diagnostic plots. This dataset provides a case where it is clear that the normalized residuals do not follow a standard normal distribution, most noticeably in the tails (subplot

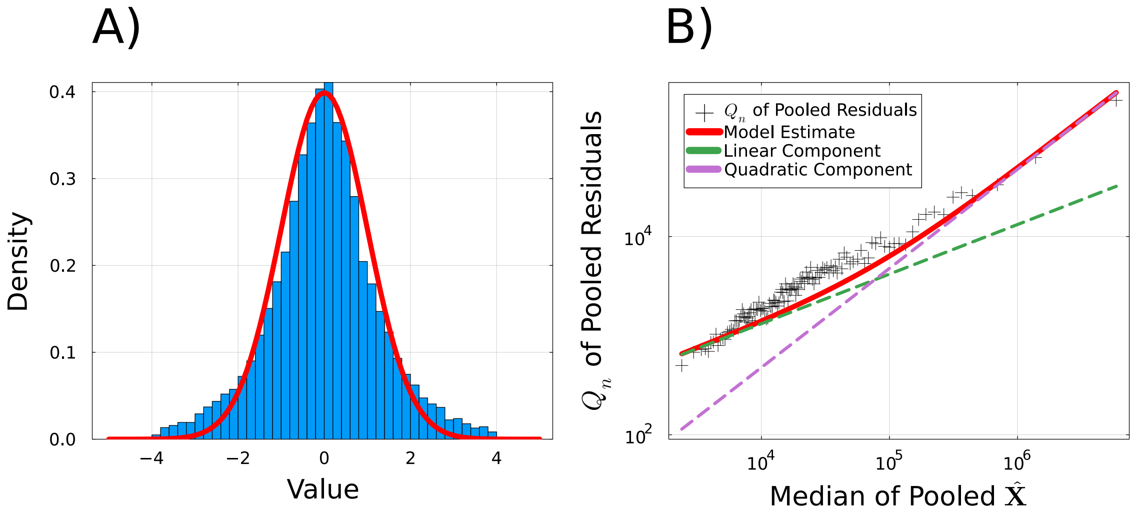

Figure 3.11: Diagnostic plots for the fit of the *TNBC* dataset. **A)** A density histogram of the normalized residuals with the true standard normal distribution overlaid in red. **B)** The error model (red) superimposed on the pooled Q_n of the residuals plotted against the pooled median of $\hat{\mathbf{X}}$. The linear and quadratic component of the model are indicated using dashed lines. The axes are log-scaled.

A). This observation explains the poor fit in subplot B. Most of the variances seem to be underestimated by the model despite using a robust estimate of the standard deviation.

Assuming the heavy tails in subplot A are caused by outliers, this dataset highlights the need for future work in this specific area. It is important to note that at this point it is difficult to argue whether the outliers are true outliers or simply heavily under-/over-estimated data points due to the model insufficiently accounting for the total variance. This could be tested by expanding the error model with addition parameters.

3.4 Conclusion

Chapter 3 has introduced a novel approach for estimating variance models through the means of an iterative algorithm based on dynamic MLPCA and fit parameter optimization. The main improvement relates to dynamic MLPCA as a means of normalization, allowing for more accurate modelling of the VF by using the normalized (and adjusted) residuals in connection with KL-Divergence to estimate the VF parameters. Simulation studies prove the robust, precise and accurate nature of the model, even in the presence of outliers and missing data, while application to experimental data provides evidence of the suitability of the model in more complex scenarios. However, the TNBC dataset does highlight the need of future work focused on outlier handling and model expansion.

Lastly, the results from each fitted variance function provide quantitative values to create comparisons of precision between different extraction methods (*MSREPS* vs *INGEL*). If such comparison is not required, the variance function also enables functional estimates of uncertainty for applications such as MLPCA or similar.

Chapter 4

A Bayesian Interpretation of Estimation of Measurement Error Models for LC-MS Data in Proteomics and Metabolomics

4.1 Introduction

4.1.1 The Basics of Bayesian Statistics

While the problem examined in the previous chapter (3) can be solved using traditional Frequentist approaches, it is interesting to examine the problem from a Bayesian perspective. Bayesian inference is rooted in the works of Thomas Bayes who discovered a novel framework for statistical reasoning and model fitting in 1763 [32]. The theorem provides a relationship between conditional probabilities, prior knowledge about the model of interest and observed data. Literature suggests that the Bayesian method may provide three major benefits in comparison to the more prevalent Frequentist approaches: (1.) it allows to estimate the probability that a given hypothesis is true, rather than estimating the probability of obtaining a dataset more extreme than the collected dataset (p-value). (2.) it allows the user to incorporate prior knowledge, proving especially helpful when data are sparse. (3.) it allows for continued model estimation based on prior results once new data becomes available, a process commonly referred to as Bayesian updating. Additionally, it is argued that the results of Bayesian analysis are easier to interpret for statistical experts and experimentalists alike [35].

Before introducing Bayes' theorem mathematically, a short review of probabilistic concepts is provided in this paragraph by introducing the following three expressions: $P(A)$, $P(A \cap B)$ and $P(A|B)$. The first expression describes the probability of observing A under any circumstances, the second describes the probability of observing

A and B at the same time, while the last expression describes the conditional probability of observing A given B. Generally, $P(A \cap B) = P(A)P(B) = P(B \cap A)$ and $P(A|B) = \frac{P(A \cap B)}{P(B)}$ but importantly $P(A|B) \neq P(B|A)$. A more detailed review of probabilistic concepts as well as non-abstract examples can be reviewed in Bauer's book *Probability theory* [9].

Combining these concepts, Bayes' theorem can be derived based on the two conditional probabilities shown in Equation 4.1 and 4.2.

$$P(A|B) = \frac{P(A \cap B)}{P(B)}, P(B) \neq 0 \tag{4.1}$$

$$P(B|A) = \frac{P(A \cap B)}{P(A)}, P(A) \neq 0 \tag{4.2}$$

Solving for $P(A \cap B)$ in Equation 4.2 and substituting into Equation 4.1 yields the most abstract form of Bayes' theorem shown in Equation 4.3 [62].

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, P(B) \neq 0 \tag{4.3}$$

This means that the conditional probability of A given B equals the product of the probability of B given A and the probability of A, divided by the probability of B. Clearly, the quantity is only defined in cases where $P(B) \neq 0$ [31]. For the remainder of the thesis, this condition is not continuously mentioned but implicitly still holds.

To understand this representation of Bayes' theorem in the context of estimating models, we adjust the Eqn. 4.3 with more meaningful variables such that:

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)} \tag{4.4}$$

Here, θ represents some model and D represents some data. Expressing this version of Bayes' theorem in natural language provides an easy-to-understand description of the concept. It states that the product of the probability of the data given the model ($P(\theta|D)$, data likelihood) and the probability of the model itself ($P(\theta)$, prior), normalized by the probability of the data ($P(D)$, evidence), can estimate the probability of the model given the data ($P(\theta|D)$, posterior). Thus, as long as likelihood, prior and evidence can be defined, it is possible to estimate the probability of any model given the currently observed data.

4.1.2 Introductory Examples

The following subsections will provide two examples based on the introduced Bayesian concepts to enable a deeper understanding of the relationships between prior, likelihood and posterior. Both examples make use of the grid approximation method. This approach discretizes the target posterior distribution, $P(D|\theta)$, by choosing a set of discrete values for θ, as it is impossible to evaluate the distribution at every possible θ [54]. Methods that can determine a continuous θ, and provide a better, non-discrete estimate of the posterior distribution, are introduced in later sections.

The Coin Flip

A simple application of Bayesian statistics focuses on estimating the true probability, p, of observing heads in an unbiased coin flip. As a starting point, the quantities on the right-hand side of Equation 4.4 need to be well defined. The model, θ, consists of p, where $0 \leq p \leq 1$ due to the nature of probabilities. Based on the grid approximation, p is discretized into eleven evenly spaced values (model options) within its bounds. The options are listed in row one of Table 4.1. The prior, $P(\theta)$, is set to be uniform. Conceptually this states that every grid option of θ has the same likelihood

θ (p)	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
$P(D\|\theta)$	0	0.0038	0.053	0.16	0.23	0.19	0.1	0.029	0.0033	0	0
$P(\theta)$	1	1	1	1	1	1	1	1	1	1	1
$P(D\|\theta)P(\theta)$	0	0.0038	0.053	0.16	0.23	0.19	0.1	0.029	0.0033	0.0	0
$P(\theta\|D)$	0	0.049	0.69	2.06	2.95	2.51	1.31	0.38	0.043	0	0

Table 4.1: Calculated quantities of the data likelihood, prior, proportional posterior and normalized posterior for each discrete model option of θ based 5 heads in 12 coinflips.

of being the truth. This concept is commonly referred to as an uninformative prior [1]. To calculate $P(D|\theta)$ correctly, a distribution best describing the observed data is required to calculate the data likelihoods. Coin flips are analogous to success-failure experiments; thus the binomial distribution which is parameterized by the number of trials, n, the number of successes, k, and the likelihood of success, p, is a natural choice. Lastly, $P(D)$ is often impossible to estimate as there are no suitable methods to calculate the probability of a single dataset. Therefore it is conveniently ignored or used as a normalization factor by marginalizing (integrating) over θ resulting in $P(D) = \int P(D, \theta)\ d\theta$ where the joint probability distribution of D and θ ($P(D, \theta)$) is the same as the numerator in Baye's Theroem ($P(D|\theta)P(\theta)$). This shortcut does not influence the information contained within the posterior distribution as it simply normalizes the area under the posterior to one, making the posterior a true probability density function. Table 4.1 lists the quantities for θ (which is equal to p), $P(D|\theta)$, $P(\theta)$, $P(D|\theta)P(\theta)$ and $P(\theta|D)$ given five heads in 12 coin flips. Figure 4.1 represents the data provided in Table 4.1 visually.

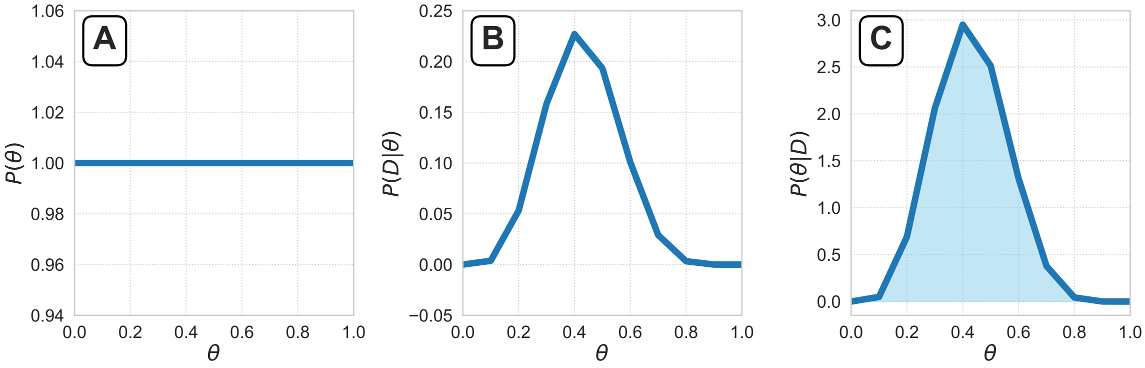

Figure 4.1: **A)** The prior distribution put on the model, θ, which consists of p. It assumes that there is no preference for any value of θ. **B)** The likelihood of observing the data, D, at differing values of the model, θ. The highest likelihood is observed at $\theta = \frac{5}{12}$. **C)** The normalized posterior distribution which represents a mixture of the prior distribution and data likelihood. The shaded area integrates to 1.

With the goal of explicitly helping the reader understand the origin of the quantities listed in Table 4.1 (and visualized in Figure 4.1), this paragraph provides the

[1] A popular discussion revolves around the meaning of *uninformative*. Using an *uniformative* prior expresses that nothing is known about the model but that in itself contains some information. There is no true *uninformative* prior.

exact equations needed to recreate the example. The data likelihood, $P(D|\theta)$ represents the probability of observing the data, 5 heads with 12 flips, given the current model (some value for p) and number of trials (which is constant). It can be calculated based on the probability mass function for binomial distributions provided in Equation 4.5, where n represents the number of trials, k is the number of heads and p is the probability of heads (θ).

$$P(k|p,n) = \binom{n}{k} p^k (1-p)^{n-k} \tag{4.5}$$

The probability of the prior, $P(\theta)$, is calculated given the currently selected model along with its prior distribution, answering the question of the likelihood of observing this model, θ, under the prior. Equation 4.6 provides the mathematical expression for this quantity under a uniform prior.

$$P(\theta) = \frac{1}{1-0}, \text{ for } 0 \leq \theta \leq 1 \tag{4.6}$$

In the last step, the posterior likelihood can be estimated using Equation 4.7. If posterior probabilities are required, the resulting data can be normalized to integrate to 1 as shown in the fifth row of Table 4.1.

$$P(\theta|D) = P(D|\theta)P(\theta) \tag{4.7}$$

This example is well suited to address the concept of Bayesian updating. A first step consists of collecting more data; flipping a coin 20 additional times yields 12 heads and 8 tails. To construct a new model, the previously estimated posterior becomes the new prior. In other words, the posterior resulting from the first dataset becomes the current belief about the model before infusing knowledge gained from a second dataset. Equations 4.5 through 4.7 equally apply in this new context. Figure 4.2 shows the prior (previous posterior), the new data likelihood as well as the normalized posterior. Visually, it is of value to note the posterior merging the distributions of the prior and data likelihood.

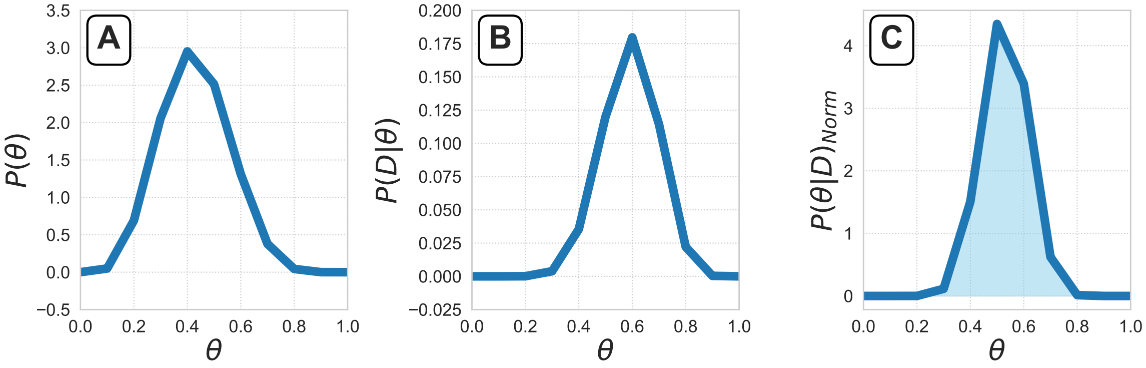

Figure 4.2: Results obtained after Bayesian updating. **A)** The new prior consists of the old, normalized posterior. **B)** The data likelihood for each option of p. It peaks at $p = \frac{12}{20}$. **C)** The posterior is a mixture of the prior and likelihood. It is shown in the normalized form. The shaded area integrates to 1.

Estimating Population Mean and Variance

The previous example is concerned with the estimation of a single-parameter model which is a rare occurrence in real-life problems. While building on the introduced processes, the following example aims to elaborate on the Bayesian approach given a more complex question. Figure 4.3 visualizes the density distribution of heights from people in the !Kung tribe collected by Nancy Howell [48]. The goal consists of estimating the population parameters under the assumption that the heights are normally distributed. Thus, our model, θ, consists of μ and σ, the population parameters for the assumed normal distribution.

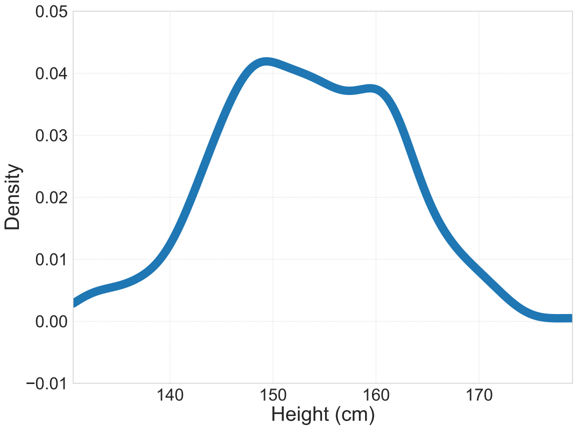

Figure 4.3: Distribution of heights from members of the !Kung tribe.

Using the grid approach, μ is discretized into steps of one given $100 \leq \mu \leq 260$ and σ is discretized in steps of 0.5 given $5 \leq \sigma \leq 15$. The grid results in 3381 model options. Each model parameter requires its own prior distribution. Since the world wide average height is estimated at 178 cm, it is reasonable to believe that $\mu \sim \mathcal{N}(178, 20)$ [2]. There is no prior belief for σ. However, it is zero-constrained and will, conceptually, not exceed 50 cm. Therefore, justifying $\sigma \sim U(0, 50)$. Calculation of $P(\theta)$ becomes trivial provided the question: What is the likelihood of observing one model combination (of μ and σ) in the grid given the respective priors? Mathematically, for $\mu = 100$ and $\sigma = 5$, this is expressed in Equation 4.8. Since likelihoods can become smaller than floating point accuracy, it is best practice to work with log likelihoods.

$$\begin{aligned}\log\left(P(\theta_{100,5})\right) = & \log\left(P\left(X = 100 \mid X \sim \mathcal{N}(178, 20)\right)\right) \\ & + \log(P(X = 5 \mid X \sim U(0, 50))) \\ = & \log\left(\frac{1}{20\sqrt{2\pi}} e^{-\frac{(100-178)^2}{2\times 20^2}}\right) + \log\left(\frac{1}{50-0}\right)\end{aligned} \tag{4.8}$$

The probability of the data given the model, $P(D|\theta)$, is calculated as the cumulative likelihood of observing the data given every model options in the grid. Equation 4.9 provides an example based on the same μ and σ from above.

$$\begin{aligned}P(D \mid \theta_{100,5}) = & \sum_{i=0}^{n_D} \log(P(X = \text{Data}_i \mid X \sim \mathcal{N}(100, 5))) \\ = & \sum_{i=0}^{n_D} \log\left(\frac{1}{5\sqrt{2\pi}} e^{-\frac{(D_i-100)^2}{2\cdot 5^2}}\right)\end{aligned} \tag{4.9}$$

In the last step, likelihood and prior surface are summed (log rules) and normalized to 1, resulting in $P(\theta|D)$, the posterior probability surface. All three surfaces are visualized in Figure 4.4. The surface exhibits a maximum probability of θ at $\sigma = 8.5$ and $\mu = 153$. The concept of Bayesian updating can be applied in a similar fashion to the coin flip example should new heights be collected in the future.

[2] The prior should not be influenced by looking at the data but should rather represent the current state of knowledge.

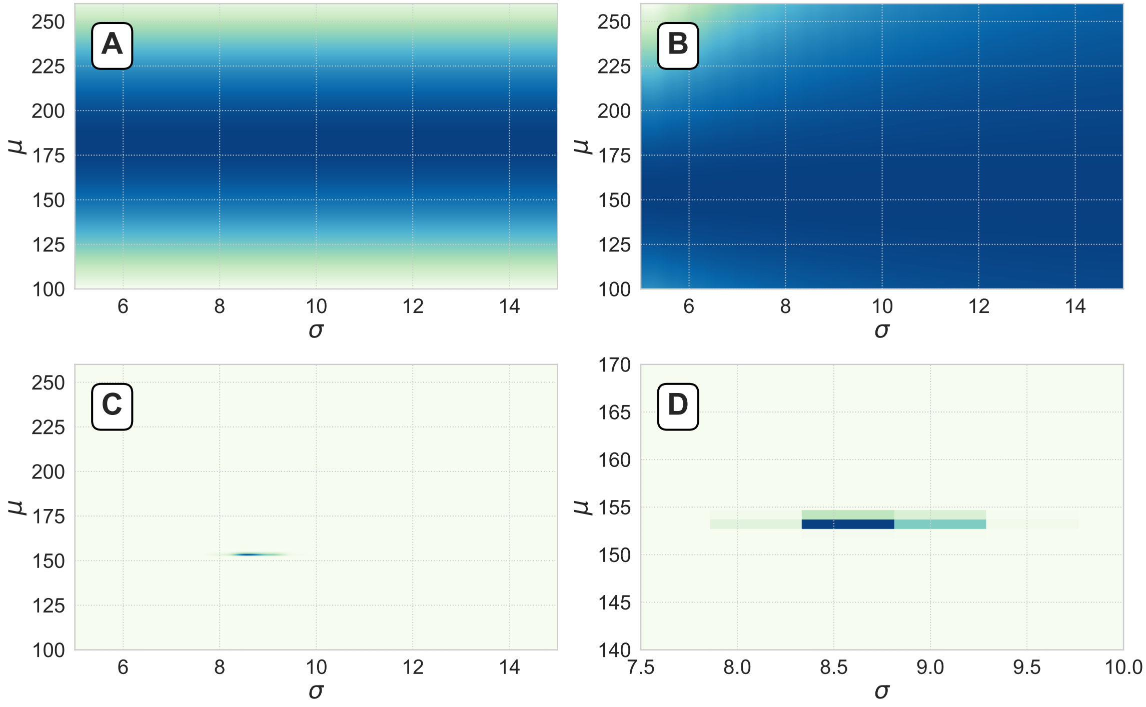

Figure 4.4: In each subplot a darker blue indicates a higher likelihood (probability). **A)** The log prior surface for every combination of σ and μ. **B)** The log likelihood surface for every σ and μ given the available data. **C)** The normalized posterior surface showing a single area of highest probability which corresponds to the estimated population mean and variance. **D)** The section of highest probability from subplot **C** enlarged.

4.2 Advanced Bayesian Concepts

The provided examples, solved with help of the grid approximation method, are well suited to gain an understanding of the basic concepts; mainly enabling an understanding of the interplay of prior and data likelihood. However, most problems in the scientific domain require a precise estimation of a continuous posterior distribution. While an *almost* continuous approximation would be possible if the grid is split into infinitesimal small intervals, this approach is not feasible in practice. Additionally, ignoring (or simplifying) the evidence term, $P(D)$, due to computational intractability is *good enough* for conceptual purposes but not suitable for scientific research. Therefore, a group of new methods that have the ability to provide precise posterior distributions for models with any number of parameters is required.

4.2.1 The Metropolis-Hastings Algorithm

In 1953, the paper *Equation of State Calculations by Fast Computing Machines* [67] introduced an algorithm revolutionizing the estimation of posterior distributions for latent variable models. Over the years, this algorithm has been coined after one of the authors: Nicholas Metropolis. The Metropolis Algorithm (M-Algorithm) aims to estimate some probability distribution, $P(\theta|D)$, given a function $f(\theta, D)$ which is: (1.) proportional to $P(\theta|D)$ (2.) defined and computable for every θ and D [80]. The initial version of the Metropolis algorithm ended up being replaced by a more general version proposed by W.K. Hastings in 1970, resulting in the Metropolis-Hastings Algorithm (MH-Algorithm) [80].

From prior discussion it is trivial to see that in a Bayesian context $f(\theta, D) \propto P(D|\theta)P(\theta) \propto P(\theta|D)$. With this knowledge, the first step in the algorithm consists of choosing of some arbitrary θ_i. This value is used to determine θ^* based on some proposal function, $g(\theta^*|\theta_i)$. The Metropolis algorithm requires the proposal function to be symmetric, $g(\theta^*|\theta_i) = g(\theta_i|\theta^*)$, whereas the Metropolis-Hastings algorithm does not impose such restrictions. A common symmetric proposal function is $g(\theta^*|\theta_i) = \theta^* \sim N(\theta_i, \sigma^2)$ for some σ. The generated θ^* is accepted as the new θ_i with a probability, $P_{acc}(\theta_i \rightarrow \theta^*)$, calculated as shown in equation 4.10. It is noteworthy that the quantity calculated in Equation 4.10 does not require the evidence term, $P(D)$ [54] [80].

$$P_{acc}(\theta_i \to \theta^*) = \min\left(1, \frac{P(D|\theta^*)P(\theta^*)}{P(D|\theta_i)P(\theta_i)} \frac{g(\theta_i|\theta^*)}{g(\theta^*|\theta_i)}\right) \tag{4.10}$$

Picking a new θ^* and accepting it based on $P_{acc}(\theta_i \to \theta^*)$ is repeated an arbitrary number of times. This results in a random walk of θ within the parameter space. The posterior distribution can be examined by using a histogram of all accepted θ normalized to a density of one; further called a density histogram. A plot of the values of θ against the number of iterations, called a trace plot, helps validate the algorithm. Ideally, the random walk generated by the algorithm should *walk* around the true value of the parameter to be estimated [54]. Navigating to this web page (https://chi-feng.github.io/mcmc-demo/app.html#EfficientNUTS,banana) allows to visual inspection of the MH-Algorithm one step at a time [105].

The MH-/M-Algorithm does come with a downside. The proposal function, $g(\theta^*|\theta_i)$, needs to be chosen wisely. Functions that result in small changes of θ will result in a high acceptance rate and inefficient exploration of the posterior distribution space as seen in Figure 4.5 A I and II. On the other hand, if the proposal function generates a new θ, in steps that are too large for the space to be explored, there is a risk of never exploring the posterior region of highest likelihood. This case is visualized in Figure 4.5 C I and II. Panel B (I and II) in the same figure provides a case in which the proposal function has been chosen properly for the problem at hand. This short-coming is important to keep in mind when estimating models of higher complexity as it may require a different method to estimate the posterior distribution. [54]

The Metropolis Algorithm and Coin Flip

Section 4.1.2 introduced an example with the goal of finding the posterior, $P(\theta|D)$, where θ is p, the probability of observing heads in an unbiased coin flip. Previously, the grid approximation method was used, resulting in a discretized, rough estimate of the posterior distribution. With the introduced background, Figure 4.6 shows the histogram and trace plot of all samples of θ resulting from the Metropolis Algorithm

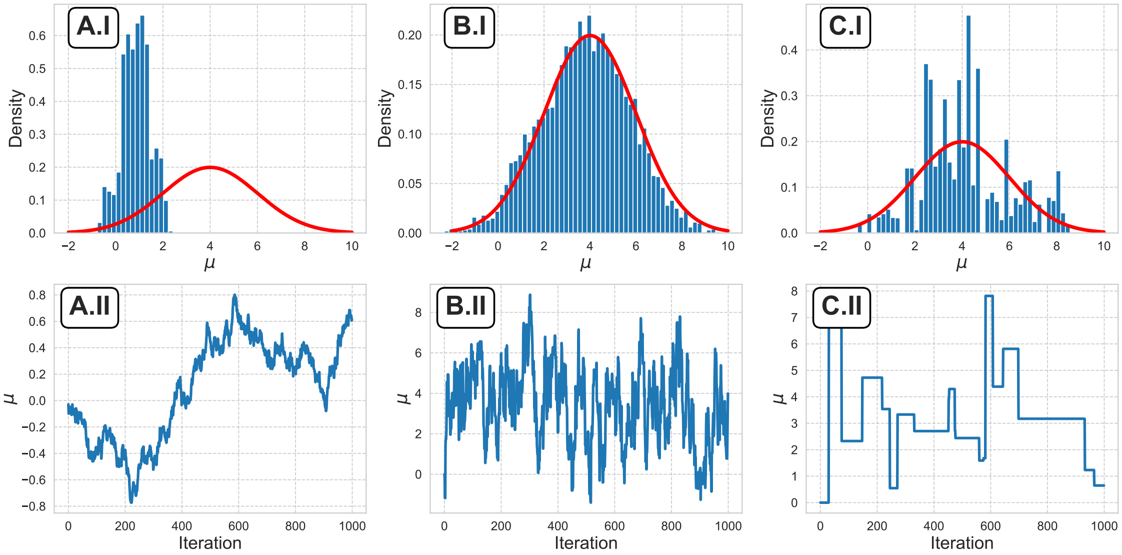

Figure 4.5: The first part of each sub figure (I) provides a density histogram of the collected samples. The red line represents the true probability distribution to be approximated. The second part of each sub figure (II) visualizes the trace plot of the samples across iterations. **A)** The chosen proposal function provides too small of steps leading to high acceptance rates but slow exploration. **B)** The proposal function is chosen ideally. **C)** The proposal function generates new proposals at too large of steps leading to a low acceptance rate and inadequate approximation of the distribution of interest.

(symmetric proposal function) implemented in Python's PYMC library [2]. It is clear to see that the posterior distribution matches up closely withe the grid approximation albeit the Metropolis algorithm providing a continuous approximation. The trace plot does not exhibit any concerning deviations mentioned in Figure 4.5.

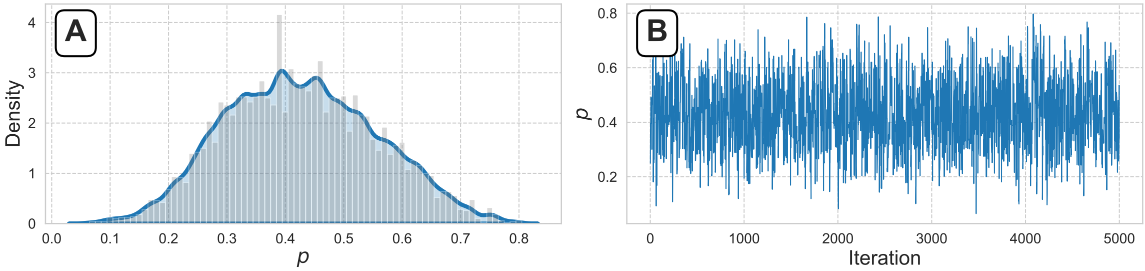

Figure 4.6: **A)** Posterior distribution as well as a histogram of p after 5000 samples were generated using the Metropolis-Algorithm. **B)** All 5000 samples of p (θ) visualized as a trace plot.

4.2.2 Hamilton Monte Carlo

As discussed previously, the short-coming of the MH-/M-Algorithm is the choice of the proposal function and step size. Once models become more complex, resulting in a high dimensional parameter space that is to be explored, the MH-/M-Algorithm breaks down by achieving an acceptance rate too small or too large with highly auto-correlated proposals, struggling to explore the posterior distribution. The Hamilton Monte Carlo Algorithm (HMC) is another method for obtaining a random walk sequence while avoiding the inherent issues of the MH-/M-Algorithm by creating a pseudo-physical system and applying Hamiltonian concepts; thus more effectively exploring the posterior space with lower auto-correlation. The algorithm first appeared in 1987 when it was used in lattice quantum chromodynamics. It remained unpopular until the 2010s when Radford M. Neal published their paper *MCMC using Hamilton dynamics* resulting in the algorithm becoming more widely used [71].

As a starting point to understand the HMC algorithm, it is most trivial to imagine a ball rolling on a surface. Given some posterior distribution $P(\theta|D)$, inverting it will result in some minimum equivalent to the optimum model parameters (conceptually refer to Figure 4.4). The HMC algorithm places a ball on the surface causing the ball to naturally move due to potential and kinetic energy. At arbitrary time steps the algorithm provides new momentum to the ball. Recording all positions at which new momentum is transferred creates a random walk sequence which can be visualized in a histogram, normalized to a density of one, resulting in an approximation of the posterior distribution. Mathematically, these exact movements can be expressed with Hamiltonian Mechanics.[71]

Hamilton mechanics is used to describe the motion and energy of a physical system using kinetic and potential energies expressed as functions of position and momentum. The following paragraphs aim to familiarize the reader with the basic ideas involved in HMC sampling. A more elaborate explanation can be found in Neal's Paper: *MCMC using Hamiltonian dynamics* [71]. An extensive, bottom-up, review of Lagrangian and Hamiltonian concepts is available in *A Student's Guide to Lagrangians and Hamiltonians* [44].

For the application of Hamiltonian concepts to Bayesian model estimation, the position of the ball is expressed as a vector of θ, the parameters of interest [3]. The momenta are expressed as some arbitrary same length vector, r. It can be shown that the Hamiltonian in this pseudo-physical system is equal to Equation 4.11. Here, M is a symmetric and positive definite mass matrix and $f(\theta)$ represents the non-normalized target density; from the previous section it is clear that $f(\theta) = P(D|\theta)P(\theta)$ [71].

$$\begin{aligned} H(r,\theta) &= -\log[f(\theta)] + \frac{1}{2}r^T M^{-1} r \\ &= -\log[P(\theta)P(D|\theta)] + \frac{1}{2}r^T M^{-1} r \\ &= U(\theta) + K(r) \end{aligned} \tag{4.11}$$

Equation 4.11 easily shows the parts of the Hamiltonian that represent the potential, $U(\theta)$, and kinetic energy, $K(r)$. With knowledge of the Hamiltonian, the HMC algorithm uses Hamilton's Equations from Equation 4.12 and 4.13 to calculate the

[3] To stay consistent with references and general literature in the Bayesian domain, vectors and matrices are not bold.

path of the rolling ball [71].

$$\frac{d\theta_i}{dt} = \frac{\partial H}{\partial r_i} \tag{4.12}$$

$$\frac{dr_i}{dt} = \frac{\partial H}{\partial \theta_i} \tag{4.13}$$

Since the differential equations are dependent on each other, it is required to choose a method for discretization. The most common choice is the leapfrog method, which requires two tune-able parameters: the number of steps, L, and the step size, Δt. Very generally speaking, the smaller Δt and the larger L, the better the simulation of the Hamiltonian system, leading to better exploration of posterior space but at the cost of computational requirements. This method also requires the derivative of $U(\theta)$ with respect to θ, making it intractable should such gradient be impossible to calculate. A more in-depth review of the leapfrog method can be found here [52].

After discretization, a new θ^* is generated based on the Hamiltonian system. A new momentum vector, r^*, is sampled at random and independent from the previous iteration. The new values are part of a modified Metropolis-Hastings acceptance probability, calculated according to Equation 4.14. By theory, the HMC algorithm will always yield an acceptance probability of 1 if the system is simulated exactly (no error due to simulation). However, even with error introduced due to simulation, the HMC algorithm yields high acceptance probabilities [71]. Repeatedly generating and accepting new samples of θ results in a random walk (similar to the MH-/M-Algorithm) which can be visualized to provide an estimate of the posterior density. The web tool suggested above also allows for visualization of the HMC algorithm. It can be found here (https://chi-feng.github.io/mcmc-demo/app.html#EfficientNUTS,banana)[105].

$$P_{acc}(\theta_i \rightarrow \theta^*, r_i \rightarrow r^*) = \min\left[1, \exp(-U(\theta^*) + U(\theta_i) - K(r^*) + K(r_i))\right] \tag{4.14}$$

Over the years the general idea of HMC sampling has been extended to improve performance. Some examples include the No U-Turn Sampler (NUTS), partial momentum refreshment or the Langevin method. However, exploration of these modifications is outside the scope of this thesis and can be researched as needed. Some references are available here [71] [23].

The HMC and the Coin Flip Example

Applying HMC sampling to the coin flip example from before becomes a simple task. Conceptually, it does not stray far away from rolling a ball within a 1-Dimensional trough. By intuition, the ball will visit the lowest points of the trough, the most probable p, the most often. Figure 4.7 shows the resulting posterior distribution and trace plot obtained through HMC sampling with PYMC in Python [2]. Through a direct comparison to the results obtained by the Metropolis-Algorithm, it is visible that HMC sampling has a higher density of samples in the area of highest likelihood. The trace plot exhibits excpected behaviour of varying around the true value.

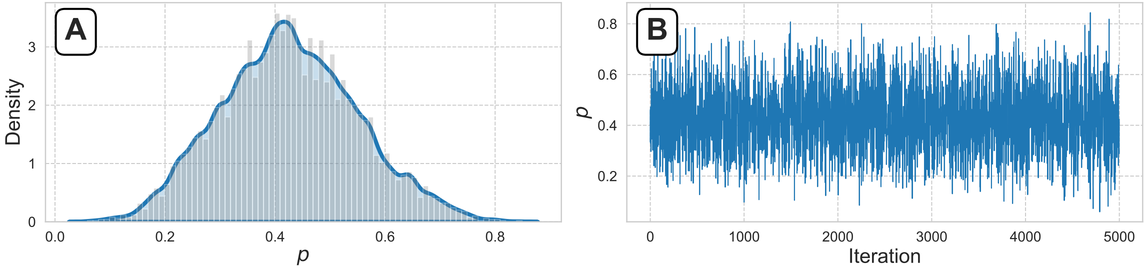

Figure 4.7: **A)** Posterior distribution as well as a histogram of p after 5000 samples generated using HMC sampling. **B)** All 5000 samples of p visualized as a trace plot.

4.2.3 Slice Sampling

The last sampling approach to be discussed is slice sampling. This method aims to estimate the posterior under the condition that a gradient, specifically $\Delta U(\theta)$, cannot be calculated for HMC sampling and the parameter space is too high-dimensional for the MH-/H-Algorithm. It provides an alternative should none of the previously discussed methods be suitable. The first mention of slice sampling applied in the Bayesian context was in the 2003 paper *Slice sampling* written by Radford M. Neal

[70]. Surprisingly, slice sampling is considered to be one of the easier sampling methods but is usually computationally slower than the MH-/H- or HMC algorithm.

As seen previously, the distribution to be estimated and sampled from is the non-normalized posterior, $P(D|\theta)P(\theta)$. In a first step, θ_i is randomly (or arbitrarily) initialized. The posterior is evaluated at θ_i, resulting in some value y. A random height, u, is sampled uniformly between 0 and the current height, y, of the posterior. In a next step, the algorithm's goal is to find an interval, $[L, R]$, within which the posterior is greater than u. The interval is initially centered around θ_i by using an arbitrary width w, resulting in $[\theta_i - w, \theta_i + w]$. By means of looping, L is extended leftwards until $P(D|L)P(L) \leq u$ and R is extended rightwards until $P(D|R)P(R) \leq u$. The resulting interval represents a slice of the posterior distribution that is above u. During the last step, a new θ^* is randomly generated from the specified interval. If $P(D|\theta^*)P(\theta^*) \geq u$, then $\theta_i \rightarrow \theta^*$ else the interval is adjusted to exclude the rejected θ^* and a new θ^* is sampled. Once a θ^* is accepted, the height y is recalculated, a new interval is created around θ^* and the algorithm is repeated until sufficient samples have been generated. As before, visualizing the random walk in a density histogram provides an estimate of the posterior distribution. [70].

The previously mentioned web tool does not offer a simple visualization of the slice algorithm. This deficit promted me to create my own interactive web tool to understand slice sampling. It can be found here (https://slice-sampling.fabianbong.me).

Slice Sampling and Coin Flips

The only condition required to apply slice sampling is a prior and data likelihood; thus making the method easily applied to the coin flip example. Figure 4.8 shows the resulting posterior and trace plot obtained through PYMC in Python [2]. The approximated posterior distribution is highly similar to the distribution obtained by HMC sampling. The trace plot exhibits the expected random walk around the true value.

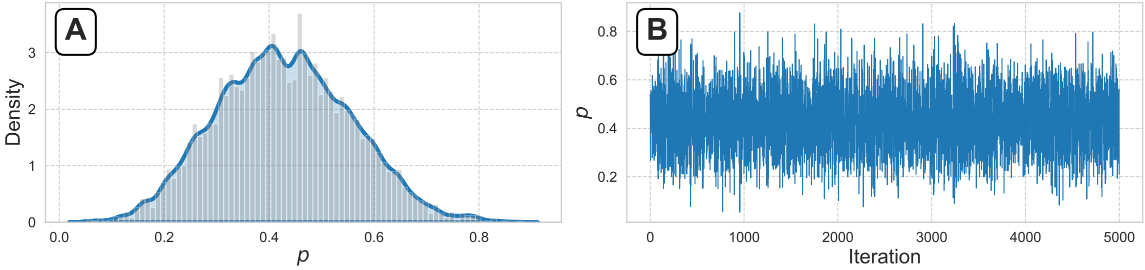

Figure 4.8: **A)** Posterior distribution as well as a histogram of p after 5000 samples generated using slice sampling. **B)** All 5000 samples of p visualized as a trace plot.

4.2.4 Diagnostic Quantities

As with any fitting algorithm, there exists a suite of more detailed concepts useful to understand and improve fit quality. Examples include best practices, diagnostic quantities and many more. In the case of Bayesian statistics, these are usually independent of the sampling method. The following paragraphs aim to provide a broken-down overview of the most prevalent concepts.

As a starting point, a good practice to follow consists of collecting more than one random walk at a time, usually in parallel, to confirm that a single random walk is not stuck in a local minimum. This is comparable to running multiple iterations for the algorithm in Chapter 3. Each random walk should start from different initial values and is referred to as a *chain* of samples. Additionally, an arbitrary number of samples are discarded at the beginning of each chain to allow the random walk to travel to the area of highest likelihood without affecting the estimate of the posterior distribution. This is referred to as burn-in [53].

When visualizing multiple chains on a trace plot (excluding the burn-in phase), ideally the chains overlap, indicating that they found the same minimum. This concept is captured in the traditional $\hat{R}$ statistic. It is calculated as the square root of the variance between chains divided by the variance within chains. The closer $\hat{R}$ is to one, the higher the agreement between chains. However, this statistic does not necessarily indicate a good fit. $\hat{R}$ can still result in 1 if all chains find the same local (wrong) minimum. Generally, any $\hat{R} \leq 1.05$ is considered acceptable. Lastly, there are a number of shortcomings of the traditional $\hat{R}$ calculation and recent research suggests the use of more sophisticated methods to calculate $\hat{R}$. The concepts can be explored in the reference [106].

Another quantity of interest is the effective sample size (ESS, N_{eff}). This value represents how many non-auto-correlated (independent) samples are needed to capture the same information as the collected, auto-correlated samples. Simply speaking, the higher the ESS (N_{eff}) (in relation to the actual number of collected samples) the better the observed samples are to gain information about the true posterior distribution. Usually, N_{eff} is expressed as the minimum of the 5% and 95% quantiles of the posterior distribution. [91] [56].

4.3 Error Model Estimation with Bayesian Statistics

With the established background knowledge follows the application of Bayesian statistics to the problem introduced in Chapter 3.

4.3.1 Model Construction

Given the data matrix $\mathbf{X}$ of size I by J, where I describes the number of channels (features) and J describes the number of samples, it has been established that the standardized residuals follow a standard normal distribution (see section 3.2.4). This directly translates to $x_{i,j} \sim N(\mu_{i,j}, \sigma_{i,j})$, where $\mu_{i,j}$ is the mean response and $\sigma_{i,j}$ the adjusted standard deviation for the corresponding replicate and channel. The values can be estimated as shown in Equation 4.15 and 4.16, assuming that MLPCA is applied in one dimension and returns a vector $\mathbf{u}$ of length I, a constant singular value, S, and a loadings vector $\mathbf{v}$ of length J.

$$\mu_{i,j} = u_i S v_j \tag{4.15}$$

$$\sigma_{i,j} = \sqrt{(\beta_1^2 \mu_{i,j} + \beta_2^2 \mu_{i,j}^2)\eta_{i,j}} \tag{4.16}$$

In the equations above, β_1 and β_2 refer to to the parameters to be estimated. As such, they are given the following priors based on the expected range of the variable.

Conceptually, the parameters are constrained to be above zero but do not have an upper limit. Values closer to zero are more likely but values as high as 10 are possible. Thus, the exponential distribution becomes a reasonable choice for the parameters. Explicitly, $\beta_1 \sim E(0.5)$ and $\beta_2 \sim E(5)$ which have a mean of 2 and 0.2, respectively.

The above model posterior for $\boldsymbol{\beta}$ can be estimated using slice sampling. Other sampling methods are not suitable. The problem is too complex for the MH-/H-Algorithm and it is intractible to define a gradient for MLPCA to utilize HMC sampling.

4.3.2 Missing Data and Outliers

Missing Data

Missing data points are defined as observations that do not have a numeric value associated with them. A collection of samples has a high likelihood of containing missing data points across single features (channels); therefore it is required to determine a strategy to handle such cases. In the context of mass spectrometry, the observance of missing data can confidently be attributed to randomness inherent to the world, γ. This is mathematically described by $P(M = 1|X, \gamma) = P(M = 1|\gamma)$, where the random variable M describes a data point that is missing ($M = 1$) or present ($M = 0$). Thus, there is no need to consider any dependence structure between observed and missing data. If any measurement channel has at least one non-missing value, MLPCA can be used to estimate a mean projection allowing calculation of a likelihood despite other missing data points. However, if less than two values are present for any channel, that channel needs to be discarded.

Outliers

Outliers are more difficult to define than missing values as they, by definition, do not follow a known distribution. The Frequentist approach handles outliers by identifying them based on the number of standard deviations a value deviates from the mean and then discarding them. A Bayesian approach to the same problem consists of increasing the likelihood of outlying values and thus only decreasing their effect on the model but not removing them completely [59]. This can be achieved by substituting the normal distribution with a three-parameter-form of the student-t-distribution which exhibits increased density in the tails with lower degrees of freedom. A modified model (from section 4.3.1) assumes $x_{i,j} \sim \mathrm{StudentT}(u_{i,j}, \sigma_{i,j}, \nu)$ where $\sigma_{i,j}$, $\mu_{i,j}$ and ν represent the scale, location and degrees of freedom of the distribution, respectively. This adjustment adds ν as another parameter to be estimated; thus requiring a prior. Since $\nu \geq 0$ (by definition) and $\nu \gg 100$ very closely mirrors a normal distribution, a prior consisting of $\nu \sim U(0, 100)$ is reasonable.

The change in distribution is accompanied by some difficulties. Mainly, $\sigma_{i,j}$ does not represent the standard deviation but rather the scale of the t-distribution. Thus, the model itself fits pseudo-variances (or pseudo-standard-deviations). These are linearly proportional to the real variances by a factor of $\frac{\nu}{\nu-2}$. Since each step in the model (most importantly MLPCA) is independent of scale, the model estimated for pseudo-variances is equivalent to the model used for real variances and does not require adjustment.

Lastly, while providing a method to better fit the data, the newly introduced parameter, ν, also has conceptual meaning. The lower ν, the higher the non-normality of the standardized residuals and thus the worse the quality of the model for the specific dataset. A large ν on the other hand indicates that the model is well suited for the data; the normalized residuals are normal.

4.4 Results and Discussion

As in the previous chapter the results section is split into two parts. The first section focuses on fitting variance models to simulated data with the goal of validating the algorithm. In the second section follows the application of the algorithm to experimental data introduced before.

4.4.1 Simulation Studies

The data used in this section has been simulated according to the specifications mentioned in section 3.3.1 of the previous chapter. Each run is considered a single execution of the Bayesian model using slice sampling with two chains of which each chain consists of 30 burn-in samples and 100 retained samples. The low number of chains and samples is ideal to simply test the algorithm. The model was executed in Python's PYMC version 5.15.1 [2].

The Ideal Case

For this simulation, the model has been fit to the simulated data, X_{ref} without any added irregularities. This case assumes full data coverage and no outliers. While such case is rare to come across in real datasets, the results verify the validity of the Bayesian method. After fitting the model, β_1 results in 1.005 ± 0.005 and β_2 results in 0.096 ± 0.001 while $\mathbf{v} = (0.258, 0.345, 0.430, 0.516, 0.603)$ or normalized by 0.403, $\boldsymbol{\alpha} = (0.6, 0.8, 1.0, 1.2, 1.4)$. The degrees of freedom of the student t-distribution are optimal at $\nu = 85.474 \pm 7.955$; which corresponds to a high degree of normality [4]. This is expected as the data is specifically simulated so that the normalized residuals follow a normal distribution. The high variability in the estimate of ν is not of concern since the shape of the t-distribution does not change significantly within a 2 standard deviation range (85.474 ± 15.91). $\hat{R}$ is observed as 1.01, 1.00 and 1.05 for β_1, β_2 and ν, respectively, indicating convergence. This is further underlined by acceptably high effective sample sizes of 188, 155 and 109, respectively.

Figure 4.9 provides the resulting fit for a single realization of the error. More elaborate diagnostic plots including posterior distributions and trace plots can be found in the supplemental material (Figure A.1). The diagnostics underline the accuracy and viability as indicated by the parameter estimates above. Figure 4.9A confirms that the residuals correctly fit to a t distribution (or normal distribution) with 85.763 degrees of freedom. In the same figure, subplot B shows a highly acceptable model fit to the robust estimate of the standard deviation (Q_n of Rousseeuw and Croux [83]) across the pooled residuals against the median of the pooled $\hat{\mathbf{X}}$. The graphs provided in Figure 4.12A and 4.13A show accuracy across multiple error realizations and ability of the model to accurately recall scaling parameters. While the variability of the scaling parameters does not differ much in comparison to the Frequentist approach,

[4] Any $\nu > 30$ can be considered to be highly similar to a normal distribution

the Bayesian approach provides greater precision and improved accuracy compared to the Frequentist approach in the parameter estimates. For this simulation, the Bayesian approach does also not exhibit a correlation between β_1 and β_2 as observed for the Frequentist approach. This comes at the expense of computational time with a single chain of 100 samples requiring upwards of 30 minutes.

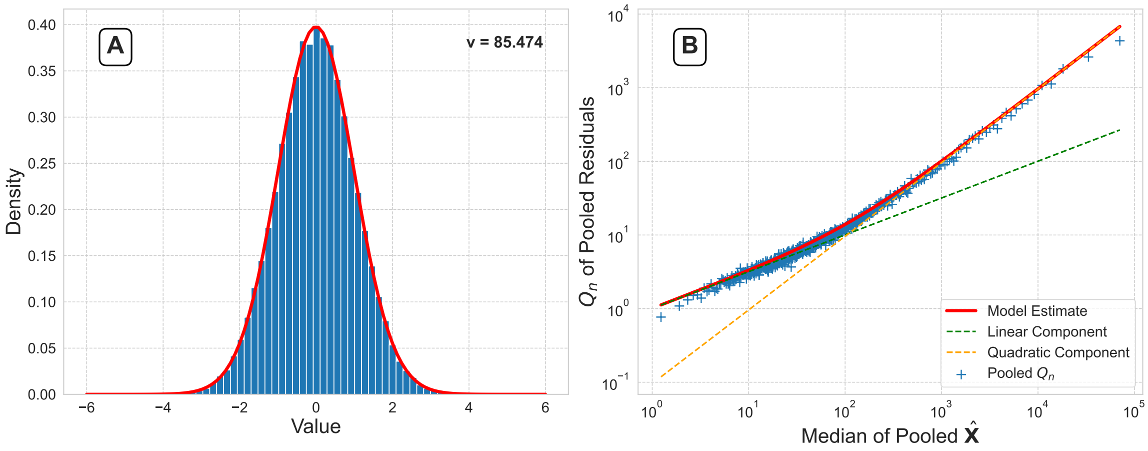

Figure 4.9: Diagnostic plots for an ideal dataset. **A)** Histogram of normalized residuals (by pseudo-variance) with a student t distribution of 85.474 degrees of freedom as reference (red). **B)** Pooled Q_n suggested by Rousseeuw and Croux [83] of non-normalized residuals against pooled median of $\hat{\mathbf{X}}$ with a bin size of 100. The red line visualizes the error model fit based on β_1 and β_2. The green and orange line show the linear and quadratic component, respectively. The axes are log-scaled.

Missing Data

The dataset in this simulation has missing data introduced completely at random. 5% of the data are missing. Results are expected to not deviate vastly from the previous simulation due to random removal of data not influencing the expected distribution of pseudo-normalized residuals. The estimates of β_1, β_2 and ν are observed at 1.000 ± 0.006, 0.095 ± 0.001 and 82.957 ± 12.957, respectively. The scaling factors are found to be $\mathbf{v} = (0.258, 0.345, 0.430, 0.516, 0.603)$ or normalized by 0.430, $\boldsymbol{\alpha} = (0.6, 0.8, 1.0, 1.2, 1.4)$. The parameters of interest show an $\hat{R}$ of 1.04, 1.02 and 1.20 as well as an effective sample sizes of 84 , 119 and 12, respectively. The diagnostic values indicate acceptable convergence despite some lower N_{eff} and a higher $\hat{R}$. This can mostly be attributed to a decrease in available samples in the dataset, $\mathbf{X}_{\text{sim}}$, but may be corrected by increasing the number of samples collected.

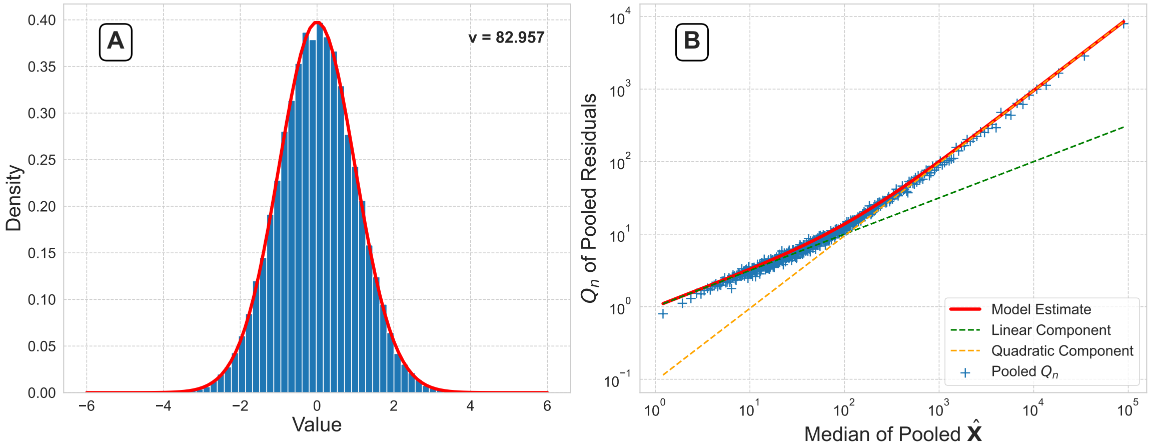

Figure 4.10: Diagnostic plots for data with missing values. **A)** Histogram of normalized residuals (by pseudo-variance) with a student t distribution of 82.957 degrees of freedom as reference (red). **B)** Pooled Q_n suggested by Rousseeuw and Croux [83] of non-normalized residuals against pooled median of $\hat{\mathbf{X}}$ with a bin size of 100. The red line visualizes the error model fit based on β_1 and β_2. The green and orange line show the linear and quadratic component, respectively. The axes are log-scaled.

Figure 4.10 depicts the resulting fit for a single realization of errors and missing values. Subplots A and B provide results that do not differ from expectations. Additional diagnostic plots (trace plot and posterior distribution) are available in the supplemental material (Figure A.2). There is no concern that the method does not hold for data with missing values. Figure 4.12B and 4.13B shows repeated parameter and scaling factor estimates on different realizations of the error (and removed values), again, proving reproducibility and precision. As before, these results indicate higher accuracy and precision (without correlation in $\boldsymbol{\beta}$) of the Bayesian method in comparison to the Frequentist approach.

Outliers

The introduction of outliers is completely at random. For 5% of data points the standard deviation has been inflated 10 fold. This simulation is of interest to understand the behaviour of the degrees of freedom. Fitting the model results in β_1, β_2 and ν as 0.932 ± 0.007, 0.107 ± 0.001 and 2.948 ± 0.032, respectively. In the same

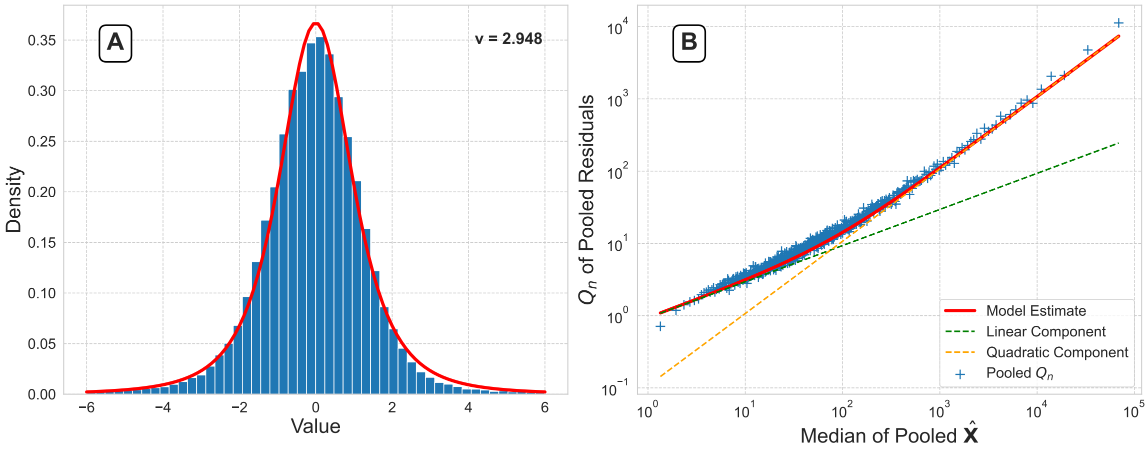

Figure 4.11: Diagnostic plots for data with outlying values. **A)** Histogram of normalized residuals (by pseudo-variance) with a student t distribution of 2.948 degrees of freedom as reference (red). **B)** Pooled Q_n suggested by Rousseeuw and Croux [83] of non-normalized residuals against pooled median of $\hat{\mathbf{X}}$ with a bin size of 100. The red line visualizes the error model fit based on β_1 and β_2. The green and orange line show the linear and quadratic component, respectively. The axes are log-scaled.

order, $\hat{R}$ is observed to be 1.02, 1.03 and 1.05 while N_{eff} is 133, 96 and 68; both diagnostic values indicate acceptable convergence. Scaling factors are found to be $\mathbf{v} = (0.259, 0.345, 0.430, 0.518, 0.601)$ or normalized by 0.430, $\boldsymbol{\alpha} = (0.6, 0.8, 1.0, 1.2, 1.4)$. The low value of ν indicates a higher degree of non-normality caused by the outliers. In contrast to the previous simulations, the standard deviation for ν is small as changes in such a low range correspond to large changes in the shape of the distribution.

Figure 4.11 contains the necessary diagnostic plots with additional visualization provided in the supplemental material (Figure A.3). As with previous fits, the diagnostic plots do not show noteworthy deviations from expectations. The low value of ν does indicate that this model may not be optimal given the amount of outliers. In connection with this finding, Figure 4.11B does indicate that even the robust standard deviations in the pooled visualization show the model to (very) slightly underestimate the true variance. Additionally, Figure 4.12C does highlight that parameters tend to have a bias towards the low end in multiple simulations with different error realizations. This bias is not reflected in the estimates of the scaling factors (Figure 4.13C).

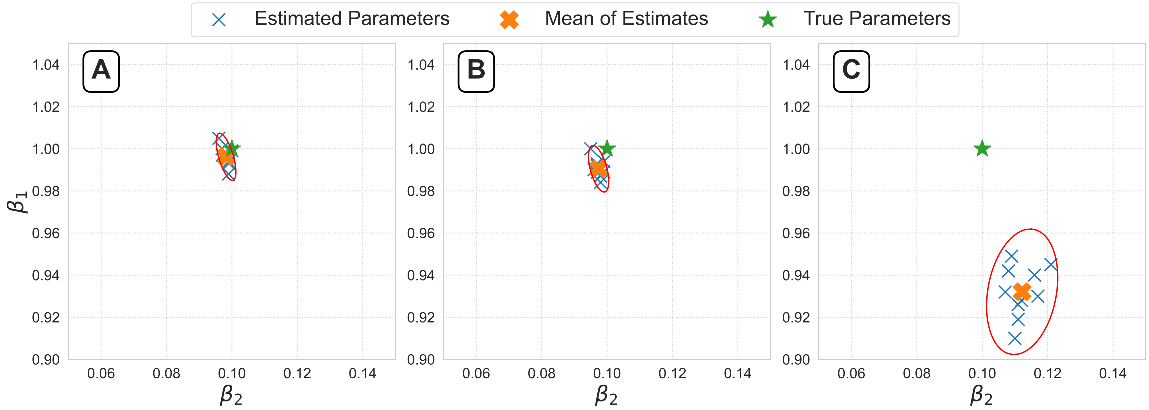

Figure 4.12: Estimates of β_1 and β_2 for different realizations of error. **A)** The data does not contain any outliers or missing values. **B)** 5% of the available data is missing. **C)** 5% of the available data are outliers but present. The red ellipse indicates the 95% confidence interval around the mean estimate.

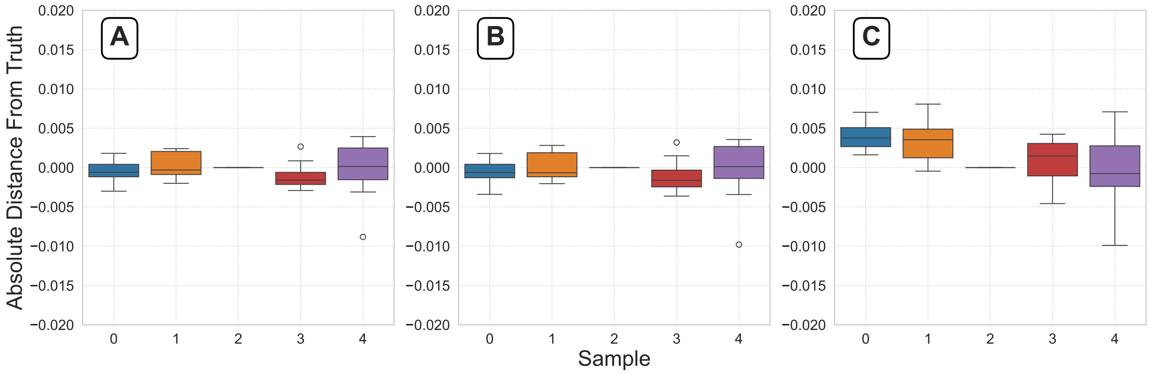

Figure 4.13: Estimates of the scaling factors, $\boldsymbol{\alpha}$, for different realizations of error centered around the true value for each sample. **A)** The data does not contain any outliers or missing values. **B)** 5% of the available data is missing. **C)** 5% of the available data are outliers but present.

The bias towards a lower β_1 is not observed in the Frequentist approach. However, the precision of the biased estimate is still higher in the Bayesian model which, depending on the application, may be of higher importance than accuracy (or bias). Future work may focus on methods to avoid such bias but retain precision; keeping the best parts of the Frequentist and Bayesian approach.

4.4.2 Experimental Data

The following datasets are equivalent to the datasets used in the previous section. For a more thorough explanation of the origin and sample preparation refer to Chapter 3 (section 3.3.2). To achieve accurate results, the model has been fit using slice sampling and five chains with 100 burn-in samples followed by 200 retained samples. The model was executed in Python's PYMC version 5.15.1 [2].

MSREPS

The parameter estimates for the MSREPS dataset come out as $\beta_1 = 0.924 \pm 0.004$, $\beta_2 = 0.055 \pm 0.001$, $\nu = 8.431 \pm 0.227$. The scaling factor result in $\mathbf{v} = (0.461, 0.459, 0.438, 0.444, 0.433)$. Division by the maximum scaling factor provides $\boldsymbol{\alpha} = (1.000, 0.995, 0.950, 0.963, 0.940)$. The convergence statistics are 1.01, 1.02 and 1.03 for $\hat{R}$ for β_1, β_2 and ν, respectively. In the same order, N_{eff} results in 596, 670 and 231. The quantities provide evidence that the algorithm did successfully converge on the highest likelihood solution.

Figure 4.14 depicts the diagnostic plots. Density histograms and trace plots are provided in the supplementary material (Figure A.4). Subplot A shows an acceptable fit of the normalized residuals to a student t distribution with 8.431 degrees of freedom. The high degrees of freedom indicate that the normalized residuals are close to normal which correlates with the observations made in Chapter 3. Subplot B visualizes the pooled robust standard deviation (Q_n) against the pooled median of $\hat{\mathbf{X}}$; depicting an acceptable fit.

Generally, due to its normality, this dataset does not show a strong deviation from the Frequentist results, except the estimate of β_1 being slightly lower. In comparison to Nickerson's findings, β_1 is higher in this approach while β_2 is comparable. The scaling factors ($\mathbf{v}$) between the two approaches introduced here are highly similar.

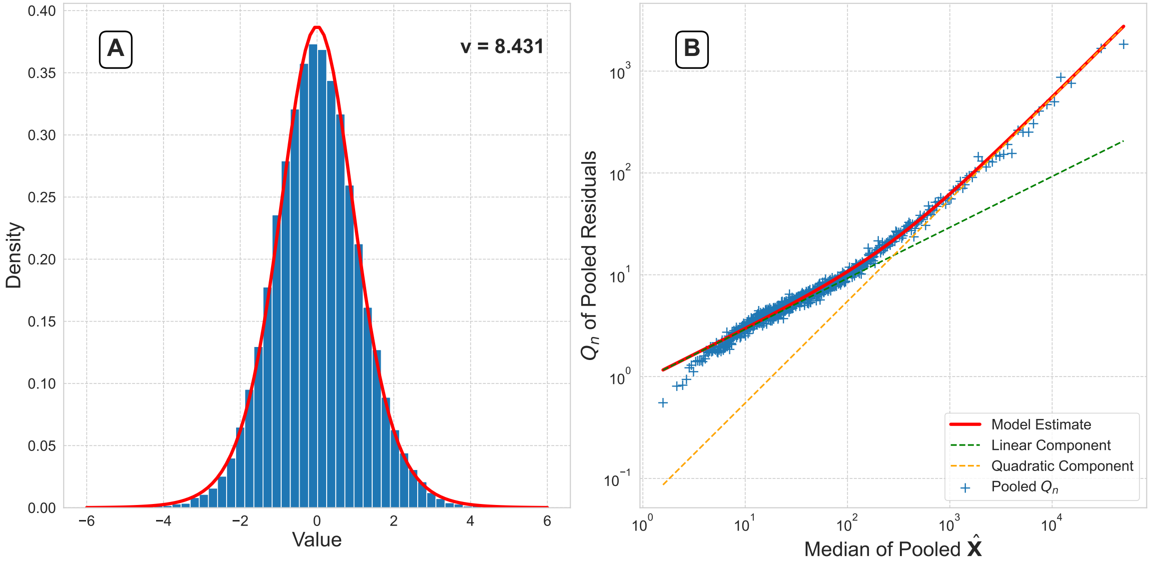

Figure 4.14: Diagnostic plots resulting from the Bayesian error model fit to the *MSREPS* dataset. **A)** Histogram of normalized residuals (by pseudo-variance) with a student t distribution of 8.431 degrees of freedom as reference (red). **B)** Pooled Q_n suggested by Rousseeuw and Croux [83] of non-normalized residuals against pooled median of $\hat{\mathbf{X}}$ with a bin size of 100. The red line visualizes the error model fit based on β_1 and β_2. The green and orange line show the linear and quadratic component, respectively.

INGEL

Fitting the *INGEL* dataset using the Bayesian model results in $\beta_1 = 0.565 \pm 0.010$, $\beta_2 = 0.214 \pm 0.003$ and $\nu = 4.950 \pm 0.135$. $\hat{R}$ (and N_{eff}) are found to be 1.02 (473), 1.02 (265) and 1.02 (334), respectively. The scaling factors are $\mathbf{v} = (0.499, 0.471, 0.621, 0.378)$. Division by the maximum provides normalized scaling factors: $\boldsymbol{\alpha} = (0.804, 0.758, 1.000, 0.609)$. The fit statics provide evidence of sufficient convergence.

Figure 4.15 shows the diagnostic plots while the posterior distribution and trace plot can be found in the supplemental material (Figure A.5). Subplot A shows an acceptable fit of the normalized residuals to a t distribution of 4.950 degrees of freedom.

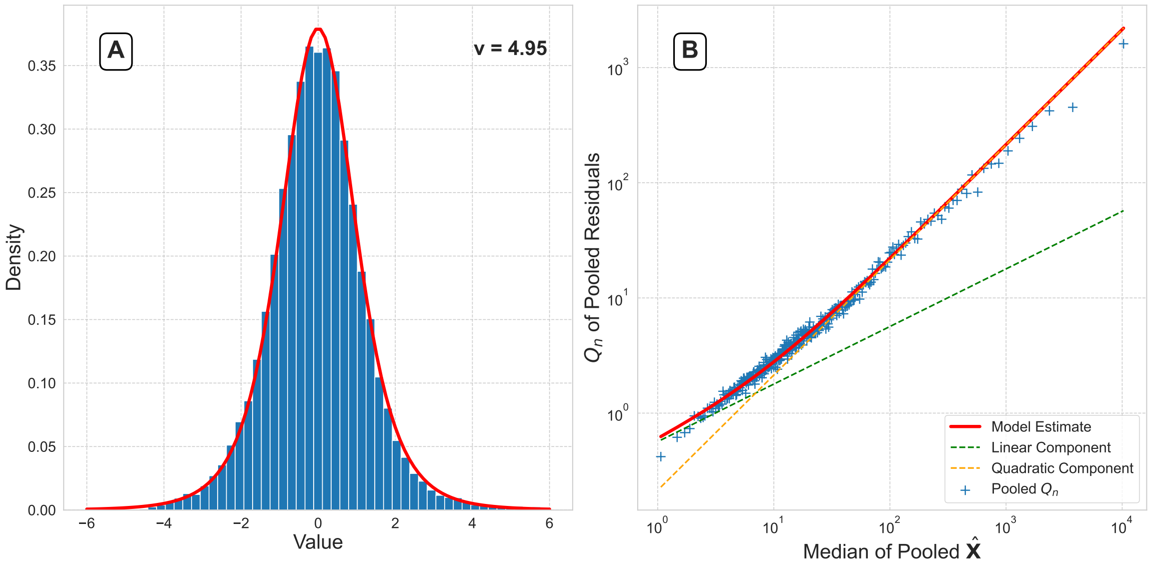

Figure 4.15: Diagnostic plots resulting from the Bayesian error model fit to the *INGEL* dataset. **A)** Histogram of normalized residuals (by pseudo-variance) with a student t distribution of 4.95 degrees of freedom as reference (red). **B)** Pooled Q_n suggested by Rousseeuw and Croux [83] of non-normalized residuals against pooled median of $\hat{\mathbf{X}}$ with a bin size of 100. The red line visualizes the error model fit based on β_1 and β_2. The green and orange line show the linear and quadratic component, respectively.

The tails are visibly thicker indicating either a high number of outliers or an insufficient variance model. Subplot B shows a reasonable fit of the model to the robust standard deviation of the pooled residuals. Lastly, a comparison to the Frequentist approach reveals a stark contrast of β_1 (Frequentist: 0.8915; Bayesian: 0.565) while β_2 is reasonably comparable (Frequentist: 0.1555; Bayesian: 0.214). These difference were expected due to the low ν. Interestingly, when contrasting the fit between the Bayesian and Frequentist approach there is visual evidence of a better fit when using the Bayesian approach, especially at the low end (refer Figure 3.10B and Figure 4.15B). In terms of Nickerson's findings, β_1 is different (Nickerson: 0.6982; Bayesian 0.565) while β_2 is comparable (Nickerson: 0.1542; Bayesian: 0.214). The scaling factors are highly similar between the introduced approaches.

TNBC

Fitting the *TNBC* data results in $\beta_1 = 9.269 \pm 0.522$, $\beta_2 = 0.117 \pm 0.004$ and $\nu = 1.683 \pm 0.04$. $\hat{R}$ (and N_{eff}) are found to be 1.05 (378), 1.08 (238) and 1.03 (344), respectively. The scaling factors are found to be: $\mathbf{v} = (0.413, 0.383, 0.366, 0.353, 0.328, 0.326, 0.322, 0.327)$. The scaling factors, normalized by the maximum, are: $\boldsymbol{\alpha} = (1.000, 0.927, 0.886, 0.855, 0.794, 0.789, 0.780, 0.792)$. The convergence statistics indicate that convergence may not be achieved as some parameters provide a higher than usual $\hat{R}$. Apart from the number of chains and samples collected, this issue may also be due to the small sample size with only 1860 channels (rows) per sample including missing values.

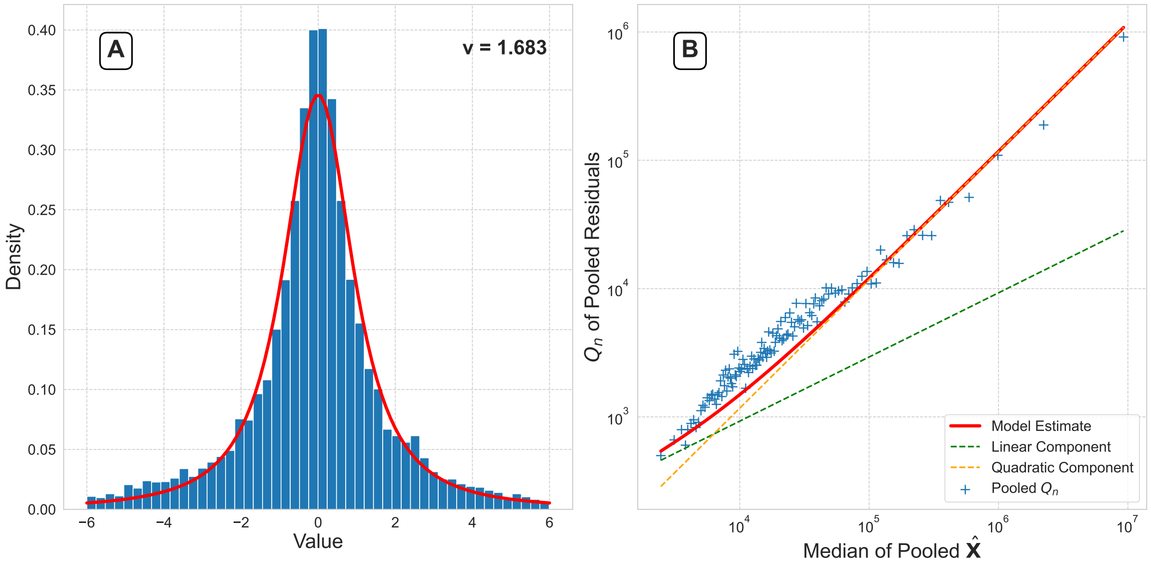

Figure 4.16: Diagnostic plots resulting from the Bayesian error model fit to the *TNBC* dataset. **A)** Histogram of normalized residuals (by pseudo-variance) with a student t distribution of 1.683 degrees of freedom as reference (red). **B)** Pooled Q_n suggested by Rousseeuw and Croux [83] of non-normalized residuals against pooled median of $\hat{\mathbf{X}}$ with a bin size of 100. The red line visualizes the error model fit based on β_1 and β_2. The green and orange line show the linear and quadratic component, respectively.

Figure 4.16 shows the diagnostic plots; posterior distribution and trace plot are found in the supplementary material (Figure A.6). The low ν indicates that the model may not be a good fit for the data and that the pseudo-normalized residuals are highly non-normal. This observations correlates with the deviation from normality observed in Chapter 3 (refer Figure 3.11B). Subplot A, the normalized residuals, shows a good fit to the appropriate t distribution of 1.683 degrees of freedom. The heavy tails are important to notice. The center of the distribution seems to be containing a larger than expected number of residuals. Subplot B shows a non-ideal fit in which the error model underestimates most of the variances. This, in combination with a low ν and unacceptable fit diagnostics, provides more evidence for an insufficiently chosen error model or unreliable data. Additionally, the fit parameters are highly different in between the two introduced methods ($\beta_1 = 13.21$ (Frequentist) and 9.27 (Bayesian), $\beta_2 = 0.048$ (Frequentist) and 0.12 (Bayesian)). However, despite this, none of the methods shows a reasonable fit. The scaling factors remain highly similar as in previous applications.

4.5 Conclusion

In conclusion, this chapter has demonstrated the potential of Bayesian statistics improving the process of fitting measurement error models to LC-MS data. If the most important model assumption of normally distributed residuals is met, the Bayesian approach provides VF estimates with less correlation in $\boldsymbol{\beta}$ and higher accuracy and precision than the equivalent Frequentist model. As a summary, Table 4.2 and 4.3 collect the model estimates for the simulated as well as the experimental datasets for each available method (including Nickerson's paper), respectively. Lastly, the findings in this chapter suggest a broader integration of Bayesian methods in the field; potentially enabling more precise and accurate model development.

		Ideal Data	Missing Data	Outlying Data
Frequentist	β_1	1.072 ± 0.051	0.992 ± 0.066	1.037 ± 0.066
	β_2	0.087 ± 0.011	0.102 ± 0.011	0.099 ± 0.012
Bayesian	β_1	0.996 ± 0.004	0.991 ± 0.004	0.932 ± 0.012
	β_2	0.098 ± 0.001	0.097 ± 0.001	0.112 ± 0.004
	ν	89.13 ± 2.50	88.28 ± 2.400	3.965 ± 1.507
Nickerson	β_1	N/A	N/A	N/A
	β_2	N/A	N/A	N/A

Table 4.2: Parameter estimates for the simulated datasets discussed in Chapter 3 and 4.

		MSREPS	*INGEL*	*TNBC*
Frequentist	β_1	1.037	0.892	13.22
	β_2	0.0482	0.156	0.048
Bayesian	β_1	0.924 ± 0.004	0.565 ± 0.010	9.269 ± 0.522
	β_2	0.055 ± 0.001	0.214 ± 0.003	0.117 ± 0.004
	ν	8.431 ± 0.227	4.950 ± 0.135	1.683 ± 0.040
Nickerson	β_1	0.747	0.693	N/A
	β_2	0.067	0.146	N/A

Table 4.3: Parameter estimates for the experimental datasets discussed in Chapter 3 and 4.

Chapter 5

Conclusions

This thesis set out to suggest two solutions to prevalent problems associated with modeling high throughput -omics data: (a) an algorithm for unsupervised, robust sample classification via projection pursuit using kurtosis as a projection index (kPPA) and coupling it to classification and regression trees (CART), and (b) a framework for fitting variance models based on replicate measurements devoid of biological significance to recapitulate systematic variation introduced in -omics data by the analytical measurement procedures.

In Chapter 2, I have shown that kPPA-CART provides superior classification for datasets with small effect sizes; enabling high-risk, high-reward studies. Further, I validated the algorithm with two experimental datasets consisting of the Takemon (proteomics and transcriptomics) and TCGA Breast Cancer dataset (transcriptomics). While kPPA-CART allowed for superior classification of the integrated Takemon data, I further provided evidence that kPPA-CART is capable of discovering novel genes for the classification of PAM50 in the TCGA dataset and that these genes are biologically significant.

In Chapters 3 and 4, I propose a novel approach for error modeling of LC-MS experiments based on MLPCA as a normalization approach; accounting for sensitivity differences in-between replicate samples. The variance function parameters are then fit by using KL-Divergence, optimizing the parameters so that the normalized (and adjusted) residuals follow a standard normal distribution. By use of simulations, I provide evidence that both (Frequentist and Bayesian) models accurately recapitulate the pre-determined scaling factors and error parameters with high accuracy and precision. Application of the models to experimental data provides reasonable diagnostics for the *INGEL* and *MSREPS* dataset. However, the same diagnostics for the

TNBC dataset suggest that either the error model does not account for most of the variance contained within the data or that the data contains too much noise; creating avenues for future research.

Bibliography

[1] SMART Servier Medical Art. Servier Medical Art is your global source for over 3000 free, up-to-date medical images designed to meet your needs.

[2] Oriol Abril-Pla, Virgile Andreani, Colin Carroll, Larry Dong, Christopher J. Fonnesbeck, Maxim Kochurov, Ravin Kumar, Junpeng Lao, Christian C. Luhmann, Osvaldo A. Martin, Michael Osthege, Ricardo Vieira, Thomas Wiecki, and Robert Zinkov. PyMC: a modern, and comprehensive probabilistic programming framework in Python. *PeerJ Computer Science*, 9:e1516, September 2023.

[3] Bruce Alberts, editor. *Molecular biology of the cell.* Garland Science, New York, 4th ed edition, 2002.

[4] Gustav Alfelt. *Modeling the covariance matrix of financial asset returns.* Department of Mathematics, Stockholm University, Stockholm, 2021.

[5] Markus Anderle, Sushmita Roy, Hua Lin, Christopher Becker, and Keith Joho. Quantifying reproducibility for differential proteomics: noise analysis for protein liquid chromatography-mass spectrometry of human serum. *Bioinformatics*, 20(18):3575–3582, December 2004.

[6] Ricard Argelaguet, Britta Velten, Damien Arnol, Sascha Dietrich, Thorsten Zenz, John C Marioni, Florian Buettner, Wolfgang Huber, and Oliver Stegle. Multi-Omics Factor Analysis—a framework for unsupervised integration of multi-omics data sets. *Molecular Systems Biology*, 14(6):e8124, June 2018.

[7] Pierre Bady, Sylvain Dolédec, Bernard Dumont, and Jean-François Fruget. Multiple co-inertia analysis: a tool for assessing synchrony in the temporal variability of aquatic communities. *Comptes Rendus. Biologies*, 327(1):29–36, January 2004.

[8] Jasmine Barra, Federico Taverna, Fabian Bong, Ibrahim Ahmed, and Tobias K. Karakach. Error modelled gene expression analysis (EMOGEA) provides a superior overview of time course RNA-seq measurements and low count gene expression. *Briefings in Bioinformatics*, 25(3):bbae233, March 2024.

[9] Heinz Bauer. *Probability theory.* Number 23 in De Gruyter studies in mathematics. Walter de Gruyter, Berlin ; New York, 1996.

[10] Hoss Belyadi and Alireza Haghighat. Unsupervised machine learning: clustering algorithms. In *Machine Learning Guide for Oil and Gas Using Python*, pages 125–168. Elsevier, 2021.

[11] Vance W. Berger and YanYan Zhou. Kolmogorov–Smirnov Test: Overview. In Ron S. Kenett, Nicholas T. Longford, Walter W. Piegorsch, and Fabrizio Ruggeri, editors, *Wiley StatsRef: Statistics Reference Online*. Wiley, 1 edition, September 2014.

[12] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017. Publisher: SIAM.

[13] Leo Breiman. Machine Learning: Random Forests. *Machine Learning*, 45(1):5–32, 2001.

[14] Richard G. Brereton. The Mahalanobis distance and its relationship to principal component scores. *Journal of Chemometrics*, 29(3):143–145, March 2015.

[15] Jamie O. Brett, Laura M. Spring, Aditya Bardia, and Seth A. Wander. ESR1 mutation as an emerging clinical biomarker in metastatic hormone receptor-positive breast cancer. *Breast Cancer Research*, 23(1):85, December 2021.

[16] Stephen J. Callister, Richard C. Barry, Joshua N. Adkins, Ethan T. Johnson, Wei-jun Qian, Bobbie-Jo M. Webb-Robertson, Richard D. Smith, and Mary S. Lipton. Normalization Approaches for Removing Systematic Biases Associated with Mass Spectrometry and Label-Free Proteomics. *Journal of Proteome Research*, 5(2):277–286, February 2006.

[17] Cancer Genome Atlas Network. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61–70, October 2012.

[18] Laura Cantini, Pooya Zakeri, Celine Hernandez, Aurelien Naldi, Denis Thieffry, Elisabeth Remy, and Anaïs Baudot. Benchmarking joint multi-omics dimensionality reduction approaches for the study of cancer. *Nature Communications*, 12(1):124, January 2021.

[19] Marc Chadeau-Hyam, Gianluca Campanella, Thibaut Jombart, Leonardo Bottolo, Lutzen Portengen, Paolo Vineis, Benoit Liquet, and Roel C.H. Vermeulen. Deciphering the complex: Methodological overview of statistical models to derive OMICS-based biomarkers. *Environmental and Molecular Mutagenesis*, 54(7):542–557, August 2013.

[20] Prabhakar Chalise and Brooke L. Fridley. Integrative clustering of multi-level 'omic data based on non-negative matrix factorization algorithm. *PLOS ONE*, 12(5):e0176278, May 2017.

[21] Travers Ching, Xun Zhu, and Lana X. Garmire. Cox-nnet: An artificial neural network method for prognosis prediction of high-throughput omics data. *PLOS Computational Biology*, 14(4):e1006076, April 2018.

[22] D. N. Chirilă, O. Bălăcescu, R. Popp, A. Oprea, N. A. Constantea, S. Vesa, and C. Ciuce. GSTM1, GSTT1 and GSTP1 in patients with multiple breast cancers and breast cancer in association with another type of cancer. *Chirurgia (Bucharest, Romania: 1990)*, 109(5):626–633, 2014.

[23] Kyungmee Choi. A review of the Bayesian approach with the MCMC and the HMC as a competitor of classical likelihood statistics for pharmacometricians. *Translational and Clinical Pharmacology*, 31(2):69, 2023.

[24] Robert J. Cotter and American Chemical Society, editors. *Time-of-flight mass spectrometry.* Number 549 in ACS symposium series. American Chemical Society, Washington, DC, 1994.

[25] Francis Crick. Central Dogma of Molecular Biology. *Nature*, 227(5258):561–563, August 1970.

[26] C. Croux, P. Filzmoser, and M.R. Oliveira. Algorithms for Projection–Pursuit robust principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 87(2):218–225, June 2007.

[27] Christophe Croux and Anne Ruiz-Gazen. High breakdown estimators for principal components: the projection-pursuit approach revisited. *Journal of Multivariate Analysis*, 95(1):206–226, July 2005.

[28] M. Daszykowski, I. Stanimirova, B. Walczak, and D. Coomans. Explaining a presence of groups in analytical data in terms of original variables. *Chemometrics and Intelligent Laboratory Systems*, 78(1-2):19–29, July 2005.

[29] M Daszykowski, B Walczak, and D.L Massart. Projection methods in chemistry. *Chemometrics and Intelligent Laboratory Systems*, 65(1):97–112, January 2003.

[30] Stephen P. Driscoll, Yannick S. MacMillan, and Peter D. Wentzell. Sparse Projection Pursuit Analysis: An Alternative for Exploring Multivariate Chemical Data. *Analytical Chemistry*, 92(2):1755–1762, January 2020.

[31] E. Eells. Review: Bayes's Theorem. *Mind*, 113(451):591–596, July 2004.

[32] Stephen E. Fienberg. When did Bayesian inference become "Bayesian"? *Bayesian Analysis*, 1(1), March 2006.

[33] Philippe Flajolet, Danièle Gardy, and Loÿs Thimonier. Birthday paradox, coupon collectors, caching algorithms and self-organizing search. *Discrete Applied Mathematics*, 39(3):207–229, November 1992.

[34] Paul Flowers, William R. Robinson, Richard Langley, and Klaus Theopold. *Chemistry.* OpenStax, Houston, Texas, March 2015. https://openstax.org/books/chemistry/pages/1-introduction.

[35] Isabella Fornacon-Wood, Hitesh Mistry, Corinne Johnson-Hart, Corinne Faivre-Finn, James P.B. O'Connor, and Gareth J. Price. Understanding the Differences Between Bayesian and Frequentist Statistics. *International Journal of Radiation Oncology*Biology*Physics*, 112(5):1076–1082, April 2022.

[36] Fuchang Gao and Lixing Han. Implementing the Nelder-Mead simplex algorithm with adaptive parameters. *Computational Optimization and Applications*, 51(1):259–277, January 2012.

[37] Susana Garcia-Recio, Toshinori Hinoue, Gregory L. Wheeler, Benjamin J. Kelly, Ana C. Garrido-Castro, Tomas Pascual, Aguirre A. De Cubas, Youli Xia, Brooke M. Felsheim, Marni B. McClure, Andrei Rajkovic, Ezgi Karaesmen, Markia A. Smith, Cheng Fan, Paula I. Gonzalez Ericsson, Melinda E. Sanders, Chad J. Creighton, Jay Bowen, Kristen Leraas, Robyn T. Burns, Sara Coppens, Amy Wheless, Salma Rezk, Amy L. Garrett, Joel S. Parker, Kelly K. Foy, Hui Shen, Ben H. Park, Ian Krop, Carey Anders, Julie Gastier-Foster, Mothaffar F. Rimawi, Rita Nanda, Nancy U. Lin, Claudine Isaacs, P. Kelly Marcom, Anna Maria Storniolo, Fergus J. Couch, Uma Chandran, Michael Davis, Jonathan Silverstein, Alexander Ropelewski, Minetta C. Liu, Susan G. Hilsenbeck, Larry Norton, Andrea L. Richardson, W. Fraser Symmans, Antonio C. Wolff, Nancy E. Davidson, Lisa A. Carey, Adrian V. Lee, Justin M. Balko, Katherine A. Hoadley, Peter W. Laird, Elaine R. Mardis, Tari A. King, AURORA US Network, Aguirre A. De Cubas, and Charles M. Perou. Multi-omics in primary and metastatic breast tumors from the AURORA US network finds microenvironment and epigenetic drivers of metastasis. *Nature Cancer*, December 2022.

[38] Jeff Gauthier, Antony T Vincent, Steve J Charette, and Nicolas Derome. A brief history of bioinformatics. *Briefings in Bioinformatics*, 20(6):1981–1996, November 2019.

[39] David M. Glover and Philip K. Hopke. Exploration of multivariate chemical data by projection pursuit. *Chemometrics and Intelligent Laboratory Systems*, 16(1):45–59, September 1992.

[40] ManishKumar Goel, Pardeep Khanna, and Jugal Kishore. Understanding survival analysis: Kaplan-Meier estimate. *International Journal of Ayurveda Research*, 1(4):274, 2010.

[41] Eric Green. Shotgun Sequencing, July 2024. Courtesy: National Human Genome Research Institute.

[42] Balázs Győrffy. Integrated analysis of public datasets for the discovery and validation of survival-associated genes in solid tumors. *Innovation (Cambridge (Mass.))*, 5(3):100625, May 2024.

[43] Benjamin Haibe-Kains, Christine Desmedt, Sherene Loi, Aedin C. Culhane, Gianluca Bontempi, John Quackenbush, and Christos Sotiriou. A three-gene model to robustly identify breast cancer molecular subtypes. *Journal of the National Cancer Institute*, 104(4):311–325, February 2012.

[44] Patrick Hamill. *A student's guide to Lagrangians and Hamiltonians*. Cambridge University Press, New York, 2014.

[45] Yong Jin Heo, Chanwoong Hwa, Gang-Hee Lee, Jae-Min Park, and Joon-Yong An. Integrative Multi-Omics Approaches in Cancer Research: From Biological Networks to Clinical Subtypes. *Molecules and Cells*, 44(7):433–443, July 2021.

[46] Robin Holliday. Epigenetics: A Historical Overview. *Epigenetics*, 1(2):76–80, April 2006.

[47] S. Hou and P.D. Wentzell. Fast and simple methods for the optimization of kurtosis used as a projection pursuit index. *Analytica Chimica Acta*, 704(1-2):1–15, October 2011.

[48] Nancy Howell. Dobe Height + Weight Observations., May 2008. https://hdl.handle.net/1807/10398.

[49] Yechen Hu, Zhongcheng Wang, Liang Liu, Jianhua Zhu, Dongxue Zhang, Mengying Xu, Yuanyuan Zhang, Feifei Xu, and Yun Chen. Mass spectrometry-based chemical mapping and profiling toward molecular understanding of diseases in precision medicine. *Chemical Science*, 12(23):7993–8009, 2021.

[50] Zhiyuan Hu, Cheng Fan, Daniel S. Oh, J. S. Marron, Xiaping He, Bahjat F. Qaqish, Chad Livasy, Lisa A. Carey, Evangeline Reynolds, Lynn Dressler, Andrew Nobel, Joel Parker, Matthew G. Ewend, Lynda R. Sawyer, Junyuan Wu, Yudong Liu, Rita Nanda, Maria Tretiakova, Alejandra Ruiz Orrico, Donna Dreher, Juan P. Palazzo, Laurent Perreard, Edward Nelson, Mary Mone, Heidi Hansen, Michael Mullins, John F. Quackenbush, Matthew J. Ellis, Olufunmilayo I. Olopade, Philip S. Bernard, and Charles M. Perou. The molecular portraits of breast tumors are conserved across microarray platforms. *BMC genomics*, 7:96, April 2006.

[51] Huei-Chung Huang and Li-Xuan Qin. Empirical evaluation of data normalization methods for molecular classification. *PeerJ*, 6:e4584, April 2018.

[52] A. Iserles. Generalized Leapfrog Methods. *IMA Journal of Numerical Analysis*, 6(4):381–392, 1986.

[53] A.M. Johansen. Markov Chain Monte Carlo. In *International Encyclopedia of Education*, pages 245–252. Elsevier, 2010.

[54] Alicia A. Johnson, Miles Q. Ott, and Mine Dogucu. *Bayes rules! an introduction to applied Bayesian modeling.* Texts in statistical science. CRC Press Taylor & Francis Group, Boca Raton, Fla London New York NY, first edition edition, 2022.

[55] Sunghwan Kim, Steffi Oesterreich, Seyoung Kim, Yongseok Park, and George C. Tseng. Integrative clustering of multi-level omics data for disease subtype discovery using sequential double regularization. *Biostatistics*, 18(1):165–179, January 2017.

[56] John K. Kruschke. *Doing Bayesian data analysis: a tutorial with R, JAGS, and Stan.* Academic Press, Boston, edition 2 edition, 2015.

[57] Dongmei Li, Zidian Xie, Marc Le Pape, and Timothy Dye. An evaluation of statistical methods for DNA methylation microarray data analysis. *BMC Bioinformatics*, 16(1):217, December 2015.

[58] Jing Li, Wen Xu, Fang Liu, Silin Huang, and Meirong He. GSTM1 polymorphism contribute to colorectal cancer in Asian populations: a prospective meta-analysis. *Scientific Reports*, 5(1):12514, July 2015.

[59] Zhenning Li, Haicheng Liao, Ruru Tang, Guofa Li, Yunjian Li, and Chengzhong Xu. Mitigating the impact of outliers in traffic crash analysis: A robust Bayesian regression approach with application to tunnel crash data. *Accident Analysis & Prevention*, 185:107019, June 2023.

[60] Eric F. Lock, Katherine A. Hoadley, J. S. Marron, and Andrew B. Nobel. Joint and individual variation explained (JIVE) for integrated analysis of multiple data types. *The Annals of Applied Statistics*, 7(1), March 2013.

[61] Miodrag Lovric. *International encyclopedia of statistical science.* Mathematics and Statistics (Springer-11649). Springer Berlin Heidelberg Springer e-books Imprint Springer, Berlin, Heidelberg, 2011.

[62] Scott Michael Lynch. *Introduction to applied Bayesian statistics and estimation for social scientists.* Statistics for social science and public policy. Springer, New York, 2007.

[63] Fergal J Martin, M Ridwan Amode, Alisha Aneja, Olanrewaju Austine-Orimoloye, Andrey G Azov, If Barnes, Arne Becker, Ruth Bennett, Andrew Berry, Jyothish Bhai, Simarpreet Kaur Bhurji, Alexandra Bignell, Sanjay Boddu, Paulo R Branco Lins, Lucy Brooks, Shashank Budhanuru Ramaraju, Mehrnaz Charkhchi, Alexander Cockburn, Luca Da Rin Fiorretto, Claire Davidson, Kamalkumar Dodiya, Sarah Donaldson, Bilal El Houdaigui, Tamara El Naboulsi, Reham Fatima, Carlos Garcia Giron, Thiago Genez, Gurpreet S Ghattaoraya, Jose Gonzalez Martinez, Cristi Guijarro, Matthew Hardy, Zoe Hollis, Thibaut Hourlier, Toby Hunt, Mike Kay, Vinay Kaykala, Tuan Le, Diana Lemos, Diego Marques-Coelho, José Carlos Marugán, Gabriela Alejandra Merino, Louisse Paola Mirabueno, Aleena Mushtaq, Syed Nakib Hossain, Denye N Ogeh, Manoj Pandian Sakthivel, Anne Parker, Malcolm Perry, Ivana Piližota, Irina Prosovetskaia, José G Pérez-Silva, Ahamed Imran Abdul Salam, Nuno Saraiva-Agostinho, Helen Schuilenburg, Dan Sheppard, Swati Sinha, Botond Sipos, William Stark, Emily Steed, Ranjit Sukumaran, Dulika Sumathipala, Marie-Marthe Suner, Likhitha Surapaneni, Kyösti Sutinen, Michal Szpak, Francesca Floriana Tricomi, David Urbina-Gómez, Andres Veidenberg, Thomas A Walsh, Brandon Walts, Elizabeth Wass, Natalie Willhoft, Jamie Allen, Jorge Alvarez-Jarreta, Marc Chakiachvili, Bethany Flint, Stefano Giorgetti, Leanne Haggerty, Garth R Ilsley, Jane E Loveland, Benjamin Moore, Jonathan M Mudge, John Tate, David Thybert, Stephen J Trevanion, Andrea Winterbottom, Adam Frankish, Sarah E Hunt, Magali Ruffier, Fiona Cunningham, Sarah Dyer, Robert D Finn, Kevin L Howe, Peter W Harrison, Andrew D Yates, and Paul Flicek. Ensembl 2023. *Nucleic Acids Research*, 51(D1):D933–D941, January 2023.

[64] Chen Meng, Dominic Helm, Martin Frejno, and Bernhard Kuster. moCluster: Identifying Joint Patterns Across Multiple Omics Data Sets. *Journal of Proteome Research*, 15(3):755–765, March 2016.

[65] Otília Menyhárt and Balázs Győrffy. Multi-omics approaches in cancer research with applications in tumor subtyping, prognosis, and diagnosis. *Computational and Structural Biotechnology Journal*, 19:949–960, 2021.

[66] METABRIC Group, Christina Curtis, Sohrab P. Shah, Suet-Feung Chin, Gulisa Turashvili, Oscar M. Rueda, Mark J. Dunning, Doug Speed, Andy G. Lynch, Shamith Samarajiwa, Yinyin Yuan, Stefan Gräf, Gavin Ha, Gholamreza Haffari, Ali Bashashati, Roslin Russell, Steven McKinney, Anita Langerød, Andrew Green, Elena Provenzano, Gordon Wishart, Sarah Pinder, Peter Watson, Florian Markowetz, Leigh Murphy, Ian Ellis, Arnie Purushotham, Anne-Lise Børresen-Dale, James D. Brenton, Simon Tavaré, Carlos Caldas, and Samuel Aparicio. The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature*, 486(7403):346–352, June 2012.

[67] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6):1087–1092, June 1953.

[68] Bilal Mirza, Wei Wang, Jie Wang, Howard Choi, Neo Christopher Chung, and Peipei Ping. Machine Learning and Integrative Analysis of Biomedical Big Data. *Genes*, 10(2):87, January 2019.

[69] Hojung Nam, Bong Chul Chung, Younghoon Kim, KiYoung Lee, and Doheon Lee. Combining tissue transcriptomics and urine metabolomics for breast cancer biomarker identification. *Bioinformatics*, 25(23):3151–3157, December 2009.

[70] Radford M. Neal. Slice sampling. *The Annals of Statistics*, 31(3), June 2003.

[71] Radford M. Neal. *MCMC using Hamiltonian dynamics.* May 2011. arXiv:1206.1901 [physics, stat].

[72] Jessica L. Nickerson, Hugo Gagnon, Peter D. Wentzell, and Alan A. Doucette. Assessing the precision of a detergent-assisted cartridge precipitation workflow for non-targeted quantitative proteomics. *PROTEOMICS*, 24(10):2300339, May 2024.

[73] Pentti Paatero and Unto Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, June 1994.

[74] Joel S. Parker, Michael Mullins, Maggie C. U. Cheang, Samuel Leung, David Voduc, Tammi Vickery, Sherri Davies, Christiane Fauron, Xiaping He, Zhiyuan Hu, John F. Quackenbush, Inge J. Stijleman, Juan Palazzo, J. S. Marron, Andrew B. Nobel, Elaine Mardis, Torsten O. Nielsen, Matthew J. Ellis, Charles M. Perou, and Philip S. Bernard. Supervised risk predictor of breast cancer based on intrinsic subtypes. *Journal of Clinical Oncology: Official Journal of the American Society of Clinical Oncology*, 27(8):1160–1167, March 2009.

[75] R Patel, M Roy, and Dutta Goutam. Mass spectrometry - A review. *Veterinary World*, page 185, 2012.

[76] C. M. Perou, T. Sørlie, M. B. Eisen, M. van de Rijn, S. S. Jeffrey, C. A. Rees, J. R. Pollack, D. T. Ross, H. Johnsen, L. A. Akslen, O. Fluge, A. Pergamenschikov, C. Williams, S. X. Zhu, P. E. Lønning, A. L. Børresen-Dale, P. O. Brown, and D. Botstein. Molecular portraits of human breast tumours. *Nature*, 406(6797):747–752, August 2000.

[77] Bastian Pfeifer, Hubert Baniecki, Anna Saranti, Przemyslaw Biecek, and Andreas Holzinger. Multi-omics disease module detection with an explainable Greedy Decision Forest. *Scientific Reports*, 12(1):16857, October 2022.

[78] Milan Picard, Marie-Pier Scott-Boyer, Antoine Bodein, Olivier Périn, and Arnaud Droit. Integration strategies of multi-omics data for machine learning analysis. *Computational and Structural Biotechnology Journal*, 19:3735–3746, 2021.

[79] Jüri Reimand, Meelis Kull, Hedi Peterson, Jaanus Hansen, and Jaak Vilo. g:Profiler–a web-based toolset for functional profiling of gene lists from large-scale experiments. *Nucleic Acids Research*, 35(Web Server issue):W193–200, July 2007.

[80] Christian P. Robert. The Metropolis-Hastings algorithm, January 2016. arXiv:1504.01896 [stat].

[81] Seong Woon Roh, Guy C.J. Abell, Kyoung-Ho Kim, Young-Do Nam, and Jin-Woo Bae. Comparing microarrays and next-generation sequencing technologies for microbial ecology research. *Trends in Biotechnology*, 28(6):291–299, June 2010.

[82] Florian Rohart, Benoît Gautier, Amrit Singh, and Kim-Anh Lê Cao. mixOmics: An R package for ‘omics feature selection and multiple data integration. *PLOS Computational Biology*, 13(11):e1005752, November 2017.

[83] Peter J. Rousseeuw and Christophe Croux. Alternatives to the Median Absolute Deviation. *Journal of the American Statistical Association*, 88(424):1273–1283, December 1993.

[84] Frederick Sanger. Determination of Nucleotide Sequences in DNA. *Science*, 214(4526):1205–1210, December 1981.

[85] Heena Satam, Kandarp Joshi, Upasana Mangrolia, Sanober Waghoo, Gulnaz Zaidi, Shravani Rawool, Ritesh P. Thakare, Shahid Banday, Alok K. Mishra, Gautam Das, and Sunil K. Malonia. Next-Generation Sequencing Technology: Current Trends and Advancements. *Biology*, 12(7):997, July 2023.

[86] Jay Shendure and Joshua M. Akey. The origins, determinants, and consequences of human mutations. *Science*, 349(6255):1478–1483, September 2015.

[87] Amrit Singh, Casey P Shannon, Benoît Gautier, Florian Rohart, Michaël Vacher, Scott J Tebbutt, and Kim-Anh Lê Cao. DIABLO: an integrative approach for identifying key molecular drivers from multi-omics assays. *Bioinformatics*, 35(17):3055–3062, September 2019.

[88] Age K. Smilde, Mariët J. Van Der Werf, Sabina Bijlsma, Bianca J. C. Van Der Werff-van Der Vat, and Renger H. Jellema. Fusion of Mass Spectrometry-Based Metabolomics Data. *Analytical Chemistry*, 77(20):6729–6736, October 2005.

[89] Kelly H. Soanes, John C. Achenbach, Ian W. Burton, Joseph P. M. Hui, Susanne L. Penny, and Tobias K. Karakach. Molecular Characterization of Zebrafish Embryogenesis via DNA Microarrays and Multiplatform Time Course Metabolomics Studies. *Journal of Proteome Research*, 10(11):5102–5117, November 2011.

[90] Meng Song, Jonathan Greenbaum, Joseph Luttrell, Weihua Zhou, Chong Wu, Hui Shen, Ping Gong, Chaoyang Zhang, and Hong-Wen Deng. A Review of Integrative Imputation for Multi-Omics Datasets. *Frontiers in Genetics*, 11:570255, October 2020.

[91] Stan Development Team. Stan Modeling Language Users Guide and Reference Manual, July 2024. Version 2.35.

[92] Indhupriya Subramanian, Srikant Verma, Shiva Kumar, Abhay Jere, and Krishanpal Anamika. Multi-omics Data Integration, Interpretation, and Its Application. *Bioinformatics and Biology Insights*, 14:117793221989905, January 2020.

[93] Torsten Söderström. Errors-in-variables methods in system identification. *Automatica*, 43(6):939–958, June 2007.

[94] T. Sørlie, C. M. Perou, R. Tibshirani, T. Aas, S. Geisler, H. Johnsen, T. Hastie, M. B. Eisen, M. van de Rijn, S. S. Jeffrey, T. Thorsen, H. Quist, J. C. Matese, P. O. Brown, D. Botstein, P. E. Lønning, and A. L. Børresen-Dale. Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences of the United States of America*, 98(19):10869–10874, September 2001.

[95] Motoki Takaku, Sara A. Grimm, John D. Roberts, Kaliopi Chrysovergis, Brian D. Bennett, Page Myers, Lalith Perera, Charles J. Tucker, Charles M. Perou, and Paul A. Wade. GATA3 zinc finger 2 mutations reprogram the breast cancer transcriptional network. *Nature Communications*, 9(1):1059, March 2018.

[96] Motoki Takaku, Sara A. Grimm, and Paul A. Wade. GATA3 in Breast Cancer: Tumor Suppressor or Oncogene? *Gene Expression*, 16(4):163–168, December 2015.

[97] Yuka Takemon, Joel M Chick, Isabela Gerdes Gyuricza, Daniel A Skelly, Olivier Devuyst, Steven P Gygi, Gary A Churchill, and Ron Korstanje. Proteomic and transcriptomic profiling reveal different aspects of aging in the kidney. *eLife*, 10:e62585, March 2021.

[98] Joel Tellinghuisen. Variance function estimation by replicate analysis and generalized least squares: A Monte Carlo comparison. *Chemometrics and Intelligent Laboratory Systems*, 99(2):138–149, December 2009.

[99] A. Tenenhaus, C. Philippe, V. Guillemot, K.-A. Le Cao, J. Grill, and V. Frouin. Variable selection for generalized canonical correlation analysis. *Biostatistics*, 15(3):569–583, July 2014.

[100] Giulia Tini, Luca Marchetti, Corrado Priami, and Marie-Pier Scott-Boyer. Multi-omics integration—a comparison of unsupervised clustering methodologies. *Briefings in Bioinformatics*, 20(4):1269–1279, July 2019.

[101] Pawel L. Urban. Quantitative mass spectrometry: an overview. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2079):20150382, October 2016.

[102] Nasim Vahabi and George Michailidis. Unsupervised Multi-Omics Data Integration Methods: A Comprehensive Review. *Frontiers in Genetics*, 13:854752, March 2022.

[103] Robert A Van Den Berg, Iven Van Mechelen, Tom F Wilderjans, Katrijn Van Deun, Henk Al Kiers, and Age K Smilde. Integrating functional genomics data using maximum likelihood based simultaneous component analysis. *BMC Bioinformatics*, 10(1):340, December 2009.

[104] Irene Van Den Broek, Fred P.H.T.M. Romijn, Nico P.M. Smit, Arnoud Van Der Laarse, Jan W. Drijfhout, Yuri E.M. Van Der Burgt, and Christa M. Cobbaert. Quantifying Protein Measurands by Peptide Measurements: Where Do Errors Arise? *Journal of Proteome Research*, 14(2):928–942, February 2015.

[105] Aki Vehtari. Interactive dashboard visualizing various MCMC sampling algorithms.

[106] Aki Vehtari, Andrew Gelman, Daniel Simpson, Bob Carpenter, and Paul-Christian Bürkner. Rank-normalization, folding, and localization: An improved R for assessing convergence of MCMC. 2019. Publisher: arXiv Version Number: 5.

[107] Yongcui Wang, Yingxi Yang, Shilong Chen, and Jiguang Wang. DeepDRK: a deep learning framework for drug repurposing through kernel-based multi-omics integration. *Briefings in Bioinformatics*, 22(5):bbab048, September 2021.

[108] J. D. Watson and F. H. C. Crick. Molecular Structure of Nucleic Acids: A Structure for Deoxyribose Nucleic Acid. *Nature*, 171(4356):737–738, April 1953.

[109] Ziwei Wei, Dunsheng Han, Cong Zhang, Shiyu Wang, Jinke Liu, Fan Chao, Zhenyu Song, and Gang Chen. Deep Learning-Based Multi-Omics Integration Robustly Predicts Relapse in Prostate Cancer. *Frontiers in Oncology*, 12:893424, June 2022.

[110] P.D. Wentzell. Other Topics in Soft-Modeling: Maximum Likelihood-Based Soft-Modeling Methods. In *Comprehensive Chemometrics*, pages 399–439. Elsevier, 2009.

[111] Peter D. Wentzell, Thays R. Gonçalves, Makoto Matsushita, and Patrícia Valderrama. Combinatorial projection pursuit analysis for exploring multivariate chemical data. *Analytica Chimica Acta*, 1174:338716, August 2021.

[112] Johan A. Westerhuis, Huub C. J. Hoefsloot, Suzanne Smit, Daniel J. Vis, Age K. Smilde, Ewoud J. J. Van Velzen, John P. M. Van Duijnhoven, and Ferdi A. Van Dorsten. Assessment of PLSDA cross validation. *Metabolomics*, 4(1):81–89, March 2008.

[113] Gangcai Xie, Chengliang Dong, Yinfei Kong, Jiang F. Zhong, Mingyao Li, and Kai Wang. Group Lasso Regularized Deep Learning for Cancer Prognosis from Multi-Omics and Clinical Features. *Genes*, 10(3):240, March 2019.

[114] Fangzhou Yao, Jeff Coquery, and Kim-Anh Lê Cao. Independent Principal Component Analysis for biologically meaningful dimension reduction of large biological data sets. *BMC Bioinformatics*, 13(1):24, December 2012.

[115] Clement G. Yedjou, Jennifer N. Sims, Lucio Miele, Felicite Noubissi, Leroy Lowe, Duber D. Fonseca, Richard A. Alo, Marinelle Payton, and Paul B. Tchounwou. Health and Racial Disparity in Breast Cancer. In Aamir Ahmad, editor, *Breast Cancer Metastasis and Drug Resistance*, volume 1152, pages 31–49. Springer International Publishing, Cham, 2019.

[116] Aaron Lun<Alun@Wehi Edu Au> Yunshun Chen <Yuchen@Wehi. Edu.Au>. edgeR, 2017.

[117] Yumeng Zhu, Xiaochao Wang, Yanqing Xu, Lu Chen, Peipei Ding, Jianfeng Chen, and Weiguo Hu. An Integrated Analysis of C5AR2 Related to Malignant Properties and Immune Infiltration of Breast Cancer. *Frontiers in Oncology*, 11:736725, September 2021.

Appendix A

Supplementary Material

A.0.1 Simulation Studies of Bayesian Error Modelling

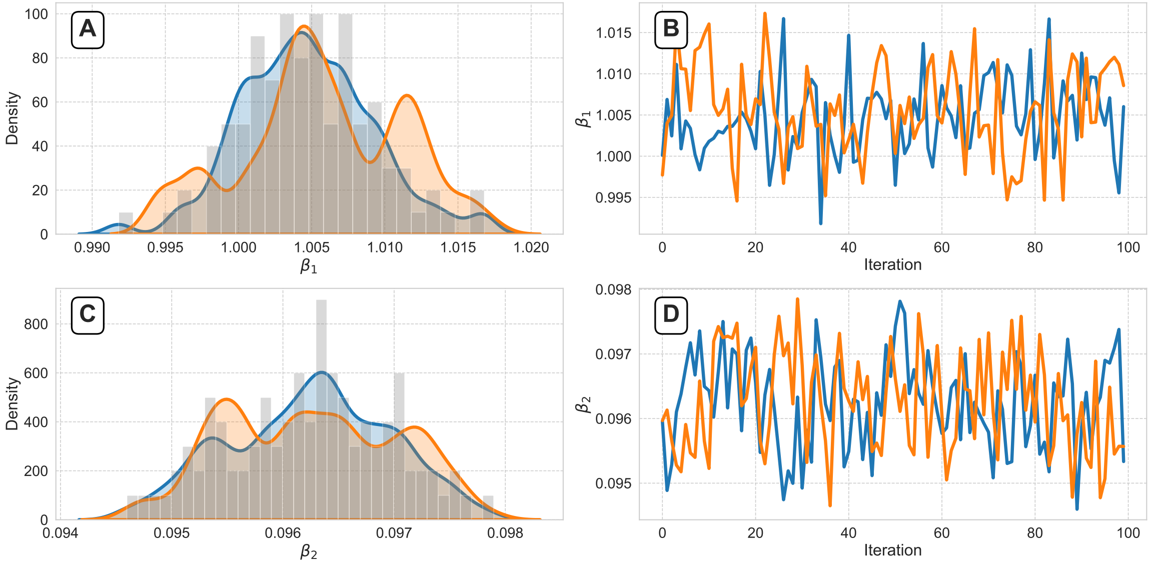

Figure A.1: The plots refer to the ideal case discussed in the simulation studies of chapter 4. **A)**, **C)** Posterior distribution of β_1 and β_2, respectively. Different colors represent different chains. **B)**, **D)** Trace plot for β_1 and β_2, respectively. Different colors represent different chains.

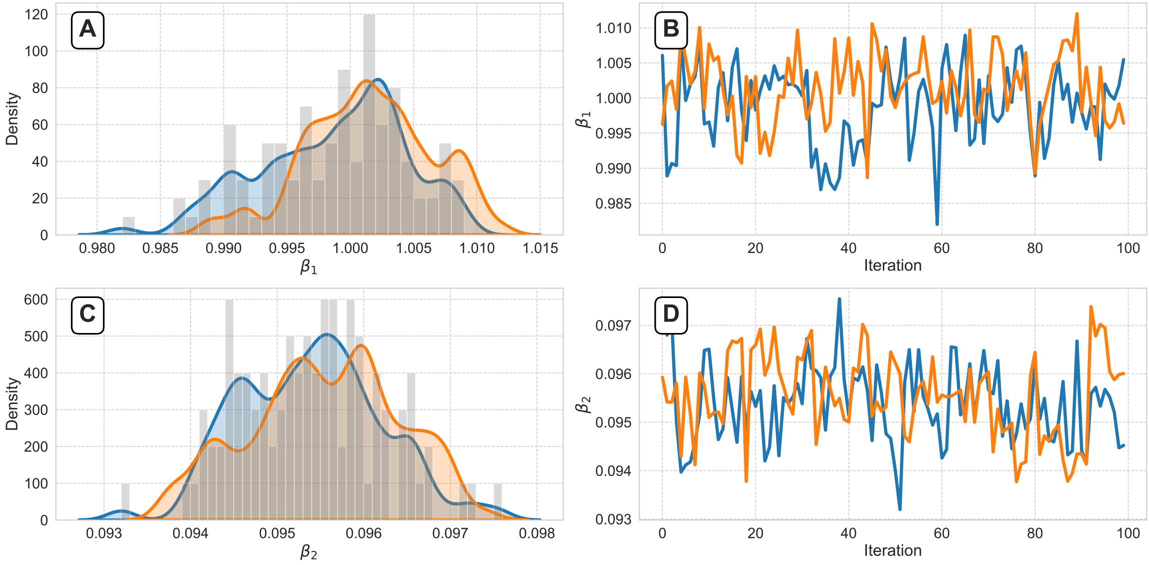

Figure A.2: The plots refer to the missing data case discussed in the simulation studies of chapter 4. **A), C)** Posterior distribution of β_1 and β_2, respectively. Different colors represent different chains. **B), D)** Trace plot for β_1 and β_2, respectively. Different colors represent different chains.

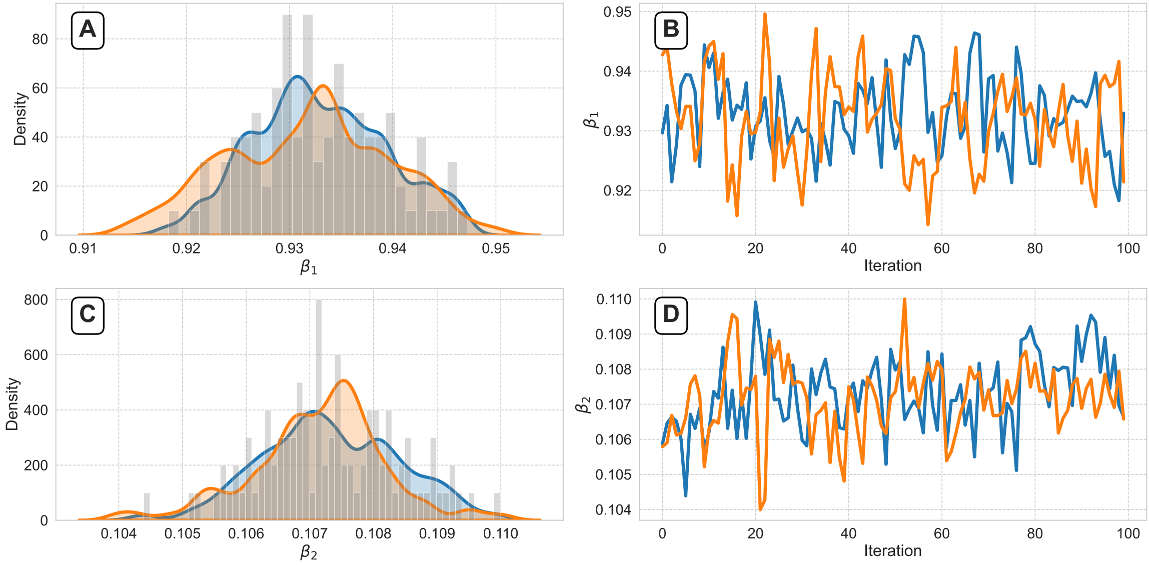

Figure A.3: The plots refer to the outlying data case discussed in the simulation studies of chapter 4. **A), C)** Posterior distribution of β_1 and β_2, respectively. Different colors represent different chains. **B), D)** Trace plot for β_1 and β_2, respectively. Different colors represent different chains.

A.0.2 Experimental Results of Bayesian Error Modelling

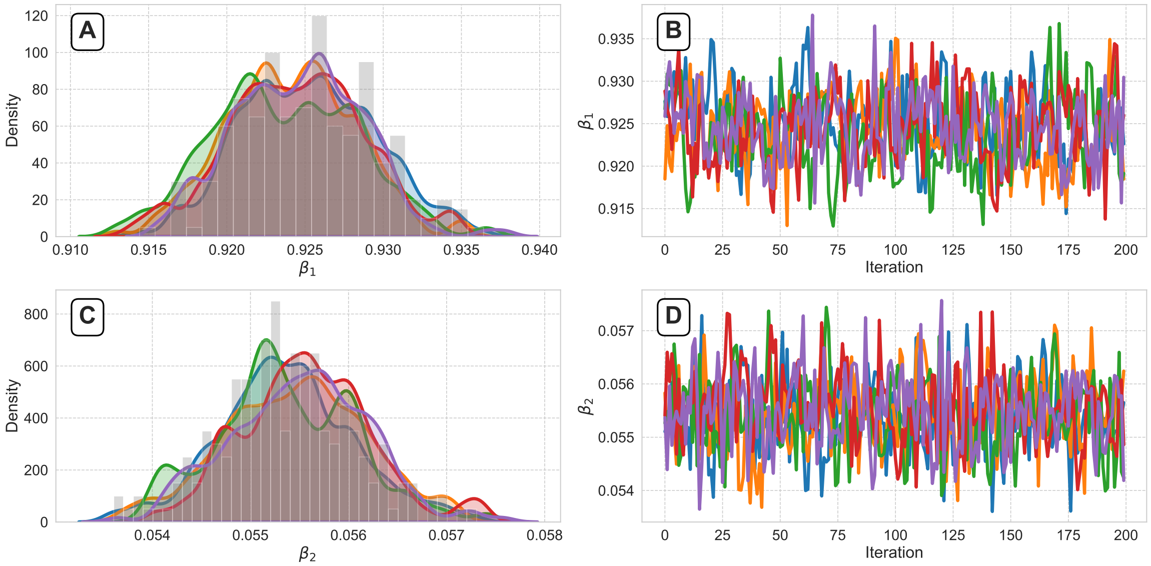

Figure A.4: The plots refer to the *MSREPS* dataset discussed in chapter 4 (and 3). **A)**, **C)** Posterior distribution of β_1 and β_2, respectively. Different colors represent different chains. **B)**, **D)** Trace plot for β_1 and β_2, respectively. Different colors represent different chains.

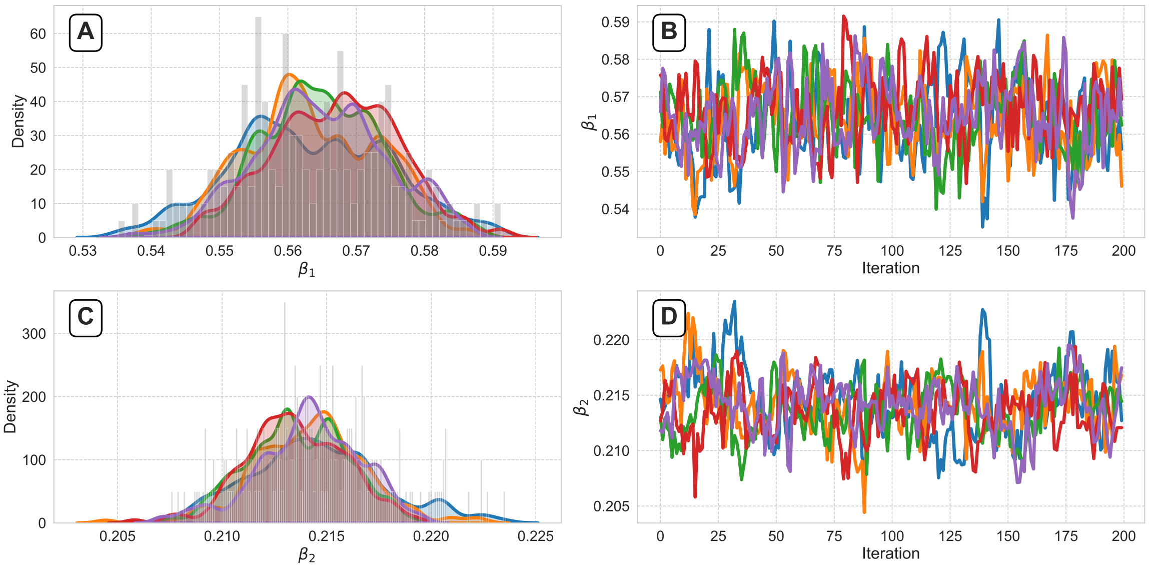

Figure A.5: The plots refer to the *INGEL* dataset discussed in chapter 4 (and 3). **A)**, **C)** Posterior distribution of β_1 and β_2, respectively. Different colors represent different chains. **B)**, **D)** Trace plot for β_1 and β_2, respectively. Different colors represent different chains.

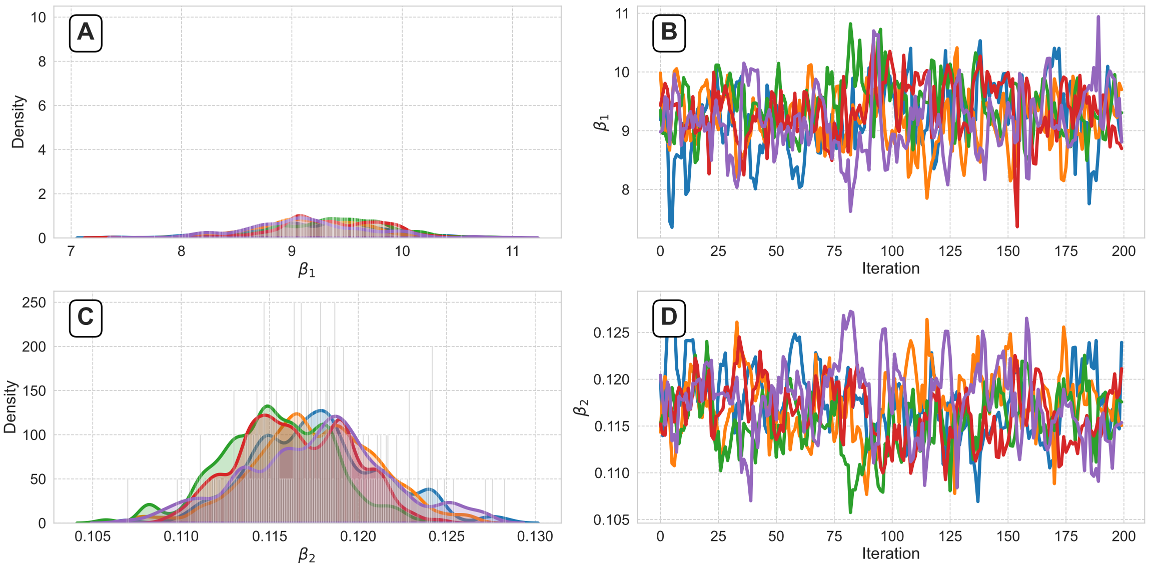

Figure A.6: The plots refer to the *TNBC* dataset discussed in chapter 4 (and 3). **A)**, **C)** Posterior distribution of β_1 and β_2, respectively. Different colors represent different chains. **B)**, **D)** Trace plot for β_1 and β_2, respectively. Different colors represent different chains.