STATISTICAL METHODS FOR MASS SPECTROMETRY PROTEOMICS EXPERIMENTS

Jonathon O'Brien

A dissertation submitted to the faculty at the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Biostatistics.

Chapel Hill
2016

Approved by:

Bahjat Qaqish

Joseph Ibrahim

Wei Sun

Mengjie Chen

Nancy Thomas

ABSTRACT

Jonathon O'Brien: Statistical Methods for Mass Spectrometry Proteomics Experiments
(Under the direction of Bahjat Qaqish)

DNA makes RNA makes proteins is the central dogma of molecular biology. While the measurement of RNA has dominated the landscape of scientific inquiry for many years, often the true outcome of interest is the final protein product. Microarray and RNAseq studies do not tell researchers anything about what happens during and after translation. For this reason interest in directly measuring the proteome has flourished. Unfortunately the direct analysis of proteins often creates a complicated inferential situation. When scientists want to see the whole proteome (or at least a large unknown sample of the proteome) mass spectrometry is often the most powerful technology available. Mass spectrometers allow researchers to separate proteins from complex samples and obtain information about the relative abundance of around 10,000 proteins in a given experiment. However the analysis of mass spectrometry proteomics data involves a complicated statistical inference problem. Inference is made on relative protein abundance by examining protein fragments called peptides. This inference problem is complicated by the two intrinsic statistical difficulties of proteomics; matched pairs and non-ignorable missingness, which combine to create unexpected challenges for statisticians. Here I will discuss the complexities of modeling mass spectrometry proteomics and provide new methods to improve both the accuracy and depth of protein estimation. Beyond point estimation, great interest has developed in the proteomics community regarding the clustering of high throughput data. Although the strange nature of proteomics data likely causes unique problems for clustering algorithms, we found that work needed to be done regarding the statistical interpretation of clustering before any special cases could be considered. For this reason we have explored clustering from a statistical framework and used this foundation to establish new measures of clustering performance. These indices allow for the interpretation of a clustering problem in the commonly understood framework of

sensitivity and specificity.

ACKNOWLEDGMENTS

I would like to thank the National Cancer Institute for supporting all of my research through the training grant 'Biostatistics for Research in Genomics and Cancer', NCI grant 5T32CA106209-07 (T32). I would also like to gratefully acknowledge the Clinical Proteomic Tumor Analysis Consortium (CPTAC) and the Chen Biochemistry Lab for providing data and guidance in all matters related to mass spectrometry proteomics.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

My research is focused on statistical methods for discovery mass spectrometry proteomics (often referred to as shotgun proteomics). The purpose of these experiments is to study the abundance of proteins across of a large portion of the proteome. When scientists know what proteins they want to examine, many different approaches can be used to estimate protein abundance. But when the target is unknown or consists of a large class of proteins, then mass spectrometry proteomics is usually the best technology available. A quintessential example of how and why this technology is used can be found in the paper by Duncan et al. (2012). Their team was interested in discovering why a class of cancer drugs called MEK inhibitors were not particularly effective in fighting tumor growth. In theory, MEK inhibitors would block the activation of certain kinase proteins which were known to play an important role in out of control cell signaling processes found in cancer cells. Through first reducing their samples to activated kinase proteins they hoped to gain insight into the cell signaling process. By comparing mass spectrometry data on the untreated sample and the MEK inhibitor treated sample they discovered that the drug was achieving the desired effect. ERK and c-MYK proteins which were targeted by the MEK inhibitor were in fact inhibited in the treated sample. They were also able to observe that other kinases were stimulated. This suggested that the drug was not effective because the cancer cells were rerouting the cell signaling process. Had they simply looked for the proteins which they knew were involved in the cell signaling process they likely would not have discovered this behavior. This is precisely the advantage of a discovery proteomics experiment. Unfortunately the statistical modeling of this data is complicated. The technology presents two key features of paramount importance from a statistical perspective; matched pairs data and non-ignorable missingness. These two factors complicate everything from simple point estimation to downstream analyses such hypothesis testing, correlation analysis and clustering. I will begin by exploring the current

literature and explaining the stages of the experimental process pertinent to statistical analysis.

CHAPTER 2: LITERATURE REVIEW

2.1 Experimental Basics

It is useful to begin a literature review by researching the methods frequently used in the field. To this end I will start by examining the popular software packages MaxQuant and Inferno made by the Max Planck Institute of Biochemistry and the Pacific Northwestern National Laboratory respectively. Papers by Cox and Mann (2008) and Cox et al. (2014) along with a book on mass spectrometry by Eidhammer et al. (2008) describe the basic details of analyzing data from MS/MS experiments. MS/MS stand for Tandem Mass Spectrometry which implies that there are two implementations the mass spectrometer. First the machine is used to separate peptides according to their masses and count them. We will refer to this phase as ms1 and to the approximate counts as intensities. Then after an initial counting the particles are smashed into smaller pieces which are again measured by the mass spectrometer (the tandem step, ms2). This secondary reading is essential for the identification of peptides and the proteins to which they belong. Unfortunately, current technology does not allow us to select every particle which was measured during ms1 to be selected for ms2. To resolve this issue one may pre-specify masses to select (targeted MS), select as many as possible giving preference to the largest intensities from ms1 (data dependent acquisition, DDA) or something in between these extremes can be done (data independent acquisition, DIA). In this paper we will focus primarily on DDA. Within MS/MS with DDA, there are a number of technologies that present their own strengths and weaknesses. We will examine three of them; Stable isotope labeling by amino acids in cell culture (SILAC), Isobaric Tag for relative and absolute quantitation (iTRAQ) and Label Free Quantification (LFQ).

2.1.1 SILAC

As explained in Cox and Mann (2008), a SILAC experiment distinguishes two samples, for simplicity suppose we are comparing healthy lung tissue to cancerous lung tissue, by processing the samples so that certain atoms in each sample contain only one version of an specific atomic isotope. This isotopic labeling enables the tissues to be processed at the same time without losing our ability to identify our samples at the end of the experiment. The processing of multiple samples simultaneously is refereed to as multiplexing and it can provide a substantial reduction in experimental variation, when modeled correctly. After labeling and measuring out two samples the samples are processed into a solution of only proteins. The proteins are then digested with Trypsin into protein fragments called peptides. Once a solution of peptides has been obtained Liquid Chromatography (LC) will be used to begin separating peptides. This is essentially a way of slowly removing peptides from the solution according to their hydrophobicity. This separation of molecules through a tube is called elution. Similar particles, in terms of hydrophobicity, come through the column in the same time frame. At the end of the column there will be a mechanism to convert the peptide molecules into ions. With the technology used by the Chen Lab at UNC this is achieved via Electrospray Ionization (ESI). Essentially as peptides flow towards the end of the elution column they are charged with electricity until the solution obtains enough energy to evaporate. What is left is a mist of charged peptide ions. These ions are then drawn into a vacuum chamber leading into the mass spectrometer. From a statistical perspective it is essential to point out that not all of the peptides become ionized. In fact the probability that a peptide molecule becomes ionized will depend on the amino acid sequence of the peptide as well as the other peptides that happened to be nearby during ionization. The importance of this ionization will be discussed in depth at a later point but for now we will simply state that this necessitates the estimation of relative protein ratios instead of absolute abundances. The reason for all this ionization is that scientists have become exceptionally good at manipulating ions. Once inside the mass spectrometer ions can be contained and forced into deterministic patterns of movement. Then by forcing the ions to be ejected from a containment chamber at known levels of energy the mass spectrometer can count all the ions that were ejected at a known

mass to charge ratio. These counts are called intensities and the largest intensities are selected in real time for the tandem step. Once selected, the molecules are broken apart further and the component pieces are all measured to aid in identification of the peptide group. These mass to charge counts are processed for each peptide through time as the molecules continue to be processed through the elution tube. Eventually an entire peptide isotope group will be processed by the mass spectrometer and software such as MaxQuant will be used to add up the results for each grouping and identify, based on the mass to charge ratio, the peptide that the count applied to. A good deal of effort goes into processing and identifying peptide groups but we will not focus on that aspect.

2.1.2 LFQ

An LFQ experiment differs from SILAC in that no labeling of different samples ever occurs. In the absence of multiplexing all experimental variation that effects peptides differently in each run will be forced into the error term of a linear model. However, since LFQ methods do not account for this variability experimentally, some efforts have been made to correct for the variation mathematically. The MaxLFQ solution is a mathematical one as described by Cox et al. (2014). Essentially they utilize the assumption that most protein fold changes will not change from one condition to the next. Explicitly they state that the average protein fold change, on the log scale, should be zero. They then fit multiplicative parameters to every observation from each sample and fit those parameters in order to minimize the average squared protein level fold change. This raises a number of important statistical questions. First we might wonder if the assumption of no differential expression is at all reasonable. Similar assumptions were frequently used in the world of microarray analysis but it is not clear that the situation should be analogous. Even in the world of microarray studies the effects of normalization were hotly debated. For example the standard technique called quantile normalization was challenged by Qiu et al. (2013). Even if the assumption is valid it is not clear how well the technique of minimizing multiplicative parameters will work. The justifications for this technique by Cox et al. (2014) are typically based around high correlation levels between LFQ estimates and estimates from other techniques. This is not

very convincing, since it says nothing about sensitivity or specificity but the focus of our research will not be on the validity of LFQ techniques so this concern will not be elaborated upon. As it stands, the scientific community appreciates the flexibility and ease of use provided by LFQ methods and they seem to be growing in popularity despite the loss of precision relative to label based methods.

2.1.3 iTRAQ

iTRAQ experiments, as explained by Karp et al. (2010) and Breitwieser et al. (2011), are similar to SILAC in that different samples are processed together but varies considerably in that the quantification step now occurs during the tandem MS stage (recall that in the SILAC experiment the tandem stage was used purely for identification). Frequently iTRAQ experiments will also implement Matrix Assisted Laser Desorption/Ionization (MALDI) to convert peptides into ions. Basically the peptides are separated into chambers which will be blasted with lasers. Once again, this ionization process occurs in probability and the number of molecules that end up inside the mass spectrometer should not be assumed constant across techniques. Once in the mass spectrometer the iTRAQ theory utilizes the vastly different masses of isobaric tags relative to component peptides. Thus when a specific mass group is selected for tandem MS and broken apart into secondary fragments measurements will be made on masses far away from the other amino acid ions which represent the relative proportions of each tag. Frequently iTRAQ experiments will simultaneously tag and process 4 or 8 different samples. Modern SILAC experiments are also capable of processing multiple samples but the ability to handle multiple samples at once seems to have been a major contributor to the popularity of iTRAQ technologies.

2.2 Modeling Efforts

Statisticians have made many models to analyze proteomics data. However there seems to be a fairly large disconnect between the methods being made by statisticians and the methods being implemented the most popular software packages. MaxQuant described by Cox and Mann

(2008), Inferno by Polpitiya et al. (2008), ASAPRatio by Li et al. (2003) and ProteinPilot by Shilov et al. (2007) all estimate proteins with either the mean or median peptide ratio. Inferno is the only of those options that offers any alternatives based on average intensity. They have three estimation procedures that they refer to as Rollups. ZRollup converts intensities to Z scores and then takes the average Z score within a protein as the protein value. RRollup is essentially the method of median ratios generalized for many samples and QRollup takes the average of the upper 66% of peptide intensities as the average protein intensity. This last method aims to avoid the trouble caused by missing data by staying away from peptides that are likely to be missing. Somehow despite the overwhelming use of ratios in the scientific community statisticians seem to have exclusively decided to model proteins as the average of their intensities. Basic linear models were implemented by Oberg and Mahoney (2012) and Clough et al. (2012) which account for ratios by including a factor for peptides. In the absence of missing data, taking contrasts on the log scale, within statistical blocks based on peptides, will yield results very similar to an average ratio based method. Thus, including peptide as a factor in a linear model is very similar to a mean ratio method. However, this relationship breaks apart in the presence of non-ignorable missing data. The papers describing the use of linear models in proteomics have focused on bringing the advantages of traditional experimental design and statistical modeling to the field. This work has been done admirably however many complexities of the data have been ignored in these papers. One such complexity is the combination of both matched pairs and non-ignorable missingness that makes protein contrasts deviate from median ratio estimates. Many other statisticians have attempted to account for missing data bias. A review of missing data techniques in proteomics by Taylor et al. (2013) compared three methods for removing missing data bias; an accelerated failure time (AFT) model by Tekwe et al. (2012), a mixture model proposed by Karpievitch et al. (2009) and K-Nearest Neighbors (KNN) imputation as described by Troyanskaya et al. (2001). Notable additions to this group include three Bayesian methods. One method proposed by Luo et al. (2009) models missing values with a logit function. The Bayesian model of Lucas et al. (2012) attempts to account for missingness caused by misidentification and Koopmans et al. (2014) proposes a model which allows for a random detection limit. Each of these methods leaves something to be desired. Only the model by Luo et al. (2009) theoretically accounts for all the

sources of missingness described above. The AFT model and the mixture model both assume the existence of a fixed detection limit. But we should expect this detection limit to change throughout time depending on what other compounds are being processed in the background. Furthermore both of them assume that any missingness above this theoretical detection limit should be missing at random with the mixture model explicitly categorizing all missingness as either at random or due to a detection limit. Any technology using intensity dependent analysis, will falsify this assumption. The interesting effort by Koopmans et al. (2014) which attempts to model a random detection limit analyzes data at the protein level. In other words the estimation within an experiment has already occurred and the only bias left to correct is at the population level. Any missing data bias from the peptide level would inevitably be carried through to the population level even with proteins that are never missing. As for the KNN solution, it is difficult to imagine that there even could be a theoretical justification for imputing over 40% of a dataset and then proceeding with an analysis as though the observations were real. Nonetheless, this option has made its way into software packages such as Inferno (Formerly called Dante) described by Polpitiya et al. (2008), so we will examine its efficacy later. In summary, there are perfectly good methods for protein estimation that utilize the matched pairs nature of a proteomics experiment. There are also perfectly good efforts to account for the missing data bias intrinsic to all proteomics experiments. However, none of the models we found correctly model both the mean structure, the missing data mechanism and provide solutions that properly account for the variability caused by non-ignorable missingness. For this reason we will endeavor to improve on the current methodology.

2.3 Modeling more complex experiments

In the last section we described some of the fundamental difficulties that arise when modeling mass spectrometry discovery proteomics experiments. Unfortunately, many more complexities are found when considering more experiments that involve replicates and multiple samples. When dealing with the extended model, questions arise regarding which factors should be expected to interact with one another. Theoretically, there is a concern that variations in ionization efficiency

could result in very different intensities for a given peptide from run to run. In both SILAC and iTRAQ experiments we have good reason to believe that the ionization efficiency for a peptide will be the same across samples, since they are all ionized under the same conditions. However, the efficiency can vary dramatically from run to run. Richard Knochenmuss convincingly demonstrates that when using Matrix Assisted Laser Desorption/Ionization (MALDI), the most common ionization technology used in iTRAQ experiments, by simply altering the amount of sample processed, the rank order of the intensities measured can be completely reversed (Knochenmuss, 2012). He ascribes this behavior to the complicated nature of electrons transfers that occur in the ion plume. Essentially, the factors that determine ionization efficiency are highly multivariate and for the most part unobserved. Furthermore, these unobserved factors are bound to change when experiments are replicated because the background analytes (other chemical structures that are ionized at the same time as a target peptide) will always change due to variations in the elution process. The result is a large amount of variation in peptide level intensities that we would greatly like to remove from our estimation procedure. Schliekelman and Liu (2014) estimated that the effect of background peptides on the probability that a peptide would be observed in a sample exceeded the effect of the abundance of a peptide molecule. For these reasons we believe it is essential to model ionization efficiency as an outcome dependent on a peptide by run interaction. However this is only possible in multiplexed experiments, and even when possible it is not clear that statisticians agree on the importance of this term. The model by Oberg and Vitek (2009) uses many interactions but a peptide by run interaction is not even discussed. In addition to complications with the mean structure of a proteomics model, the extended experiment poses complicated questions about the very nature of a missing value in a mass spectrometry experiment. In a basic ANOVA model framework, as described by Scheffé (1999), the indices of each data point and factor are given a priori. So a missing data point is any value defined in the indices that does not have an observed value. In a discovery mass spectrometry experiment this is not the case. We do not know in advance what peptides will be observed, thus we do not know the range of our indices. In the simple case, comparing only two samples, it is natural to consider missing values to be peptides that were observed in one sample but not the other. With replicates and multiple samples it is no longer clear what we want to include in the set of

missing values. One might claim that since we know the complete amino acid sequence of each protein, that any part of the sequence for which we do not have a value is a missing data point. This sounds sensible, however nobody does this because many of those peptides might have had zero probability of being observed due to low ionization efficiency. If a potential outcome has probability zero of being observed does it make sense to call it missing? More pragmatically, can information on such missing data patterns be at all useful? These are the questions that need to be addressed before efforts can be made to model more complicated proteomics experiments.

2.4 Clustering

The statistical techniques commonly used in proteomics studies go beyond experimental modeling and point estimation. One technique common to proteomics studies is data clustering. The assignment of data into various clusters has been a multidisciplinary objective since at least 1956 when Steinhaus (1956) proposed the method of k-means clustering. Essentially this method groups samples into k somewhat evenly sized groups with members of each group having small Euclidean distance. In the world of cancer biology clustering became a very popular topic after a method called hierarchical clustering, described in detail by Hastie et al. (2009), was used to discover new subtypes of breast cancer by Sø rlie et al. (2001). Now clustering has become a commonly used tool throughout the omics fields and proteomics is no exception. A natural research topic would be to explore how intensity dependent missingness effects clustering algorithms. There does not appear to be a lot of work on this topic but a paper by de Brevern et al. (2004) demonstrated with microarray data that if as little as one percent of the data were missing at random it could greatly destabilize clustering results. Considering that mass spectrometry data could have upwards of 40% missing data this demonstrated a strong need for further research. However it also raised the much more fundamental question of how two sets of clustering assignments should be compared. de Brevern et al. (2004) used what they called the conserved pairs proportion (CPP). This is just the number of pairs that were linked together in both methods divided by the total number of possible pairs. It turns out the CPP is far from the only way to compare two clusters. The most common tool is called the Rand Index proposed

by Rand (1971). This index counts the number of elements which were grouped together in both plus the number of pairs that were not linked together in both methods all divided by the total number of possible pairs. The Rand index may be the most common index used for comparing two clustering assignments but the story does not end here. In a paper by Albatineh et al. (2006) 28 different indices for comparing cluster assignments were identified. Although researchers have been using such indices since at least 1971 very few arguments exist to convincingly demonstrate why any of these indices should be preferable to another. Furthermore it turns out that there are rarely any good reasons to prefer one clustering method over another, and there are many, many clustering algorithms. Aside from K-Means and Hierarchical Clustering, Mixture Models (McLachlan and Basford (1987)), Self Organizing Maps (Kohonen (1982)), Spectral Clustering (Hagen and Kahng (1992)), Fuzzy Clustering (Dunn (1973)) and Consensus Clustering (Monti et al. (2003)) are just a small subset of the clustering algorithms available. Jain (2010) report that there are actually thousands of clustering algorithms to choose from. They argue that the reason for this multitude of methodologies is that clustering is not a well defined problem. Any set of data can be split up length wise, width wise, in circles or any other creative way that a researcher might be inclined to try, and each one of those splits might be perfectly interesting for different reasons. Without the type of mathematical population model we routinely see in inference problems it becomes very difficult to argue which way someone should split up a set of data. Although different algorithms are likely to outperform each other depending on the specific setting, the indices appear to have a more well defined purpose. Albatineh (2010) show that almost all of the clustering indices belong to a family based on an underlying 2x2 table counting the number of pairs that were matched together in each combination of clusters. They also simplify the process of comparing these indices by deriving mathematical properties of this family that allow for easy computation of expectations and variances. We found their exploration to be highly useful but believe it did not go far enough into investigating the nature of this family. In Chapter 4 we argue that a subset of this family allows us to put cluster evaluation into a commonly accepted statistical framework. This framework is necessary to begin an exploration of the effects of proteomic missingness on clustering methods however that exploration will have to remain a topic for future research.

CHAPTER 3: PROTEOMIC MODELING

3.1 Introduction

At the highest level, proteomics is the large scale study of the structure and functions of proteins. An important class of studies within this field is shotgun, or discovery, proteomics. These experiments are designed to provide information on a large set of proteins that are not specified before conducting the experiment. Discovery proteomics experiments typically necessitate the use of a mass spectrometer which entails an inferential step between the readings of the mass spectrometer and the protein level outcomes of interest. Understanding the details of these experiments becomes essential in order to conduct a sensible analysis of proteomics data. A statistician might be fooled by the superficial similarities between microarray and proteomics data into simply adopting microarray methods to be used on protein data. Although the outcomes in the experiments share the same "intensity" designation, in reality the similarities end with the name. In a microarray experiment, an intensity refers to the observed brightness of a dye that will be present when a reaction occurs with some target molecule. The exact relationship between this intensity and the underlying molecular abundance is unclear but, as explained by Dabney and Storey (2007), a researcher at least believes the intensity to be a monotone increasing function of the analyte concentration. Nothing of the sort can be claimed for intensities from shotgun proteomics experiments. To justify this statement it will be necessary to provide a basic overview of the shotgun proteomics experiment, including a detailed explanation of the ionization process and the different sources of missing data. We will show how the experimental details create two statistical features characteristic of all shotgun proteomics experiments; matched pairs data and non ignorable missingness. After justifying these features we propose a model that accounts for them and test the performance of this model on both simulated and real data.

3.1.1 Bottom Up Relative Quantification Experiments

Discovery proteomics experiments are usually a type of relative quantification experiment (Eidhammer et al., 2008). The implication here is that the experiment can only be used to provide measures of protein abundance in one sample relative to another, without ever obtaining measures of the absolute abundance in either sample. Many experiments will compare samples from numerous conditions but the simplest scenario is a comparison of protein abundances between two samples: Sample A and Sample B. A number of relative quantification workflows are available for proteomics to achieve this goal. These include Label-Free Quantification (LFQ) (Cox et al., 2014), Stable Isotope Labeling by Amino Acids in Cell Culture (SILAC) (Cox and Mann, 2008) and Isobaric Tags for Relative and Absolute Quantification (iTRAQ) (Ross et al., 2004). The detailed workflows involved in these experiments are important and should determine the types of model that a statistician would consider. For instance the model proposed in this paper works for SILAC and LFQ data, but the missing data mechanism we employ is inappropriate for an iTRAQ experiment. Nonetheless a full explanation of each experimental workflow is not necessary for our purposes. The experiments described here are referred to as bottom-up proteomic methods, because proteins are too large to be identified by mass which forces us to make inference about relative protein abundance from measurements obtained on amino acid fragments called peptides. A typical bottom-up proteomic workflow involves the extraction of proteins from cells, tissues or biological secretions/fluids, followed by proteolysis which breaks proteins into peptides. Typically this is done by adding an enzyme called trypsin that specifically cleaves proteins at lysine and arginine amino acid residues. After this digestion occurs peptides from the sample are separated according to each peptides hydrophobicity in a process called elution, where the more hydrophobic peptides will be the last to separate. After elution, peptides will travel towards an ionization device which converts the peptides into ions so that they may enter and be manipulated by a mass spectrometer. The ionization is usually done with electricity in the form of electrospray ionization (ESI) or with lasers by matrix-assisted laser desorption/ionization (MALDI). It is important to emphasize that both of these ionization technologies affect large numbers of peptides all at the same time and not all of them will successfully ionize. The elution

process aims to completely separate each group of peptides but it does not work perfectly. When referring to the measurement made on a specific peptide we may refer to all of the other peptides that were ionized at the same time as co-eluting peptides. Co-eluting peptides play an important role in determining the probability of ionization. After ionization the newly formed peptide ions are manipulated by a mass spectrometer which is capable of separating the ions according to their mass and counting the number of ions corresponding to each mass. Peptides with a large counts will be selected for fragmentation and a second mass spectrometry reading of the fragments. The process of selecting peptides for a second mass spectrometry step based on the relative magnitude of the counts is called data-dependent analysis. The second mass spectrometry step is mostly used to identify the peptides that were just counted (quantification also happens during this step in an iTRAQ experiment). A summary of the ion counts for each now identified peptide, usually computed as the area under a curve from a plot of counts through time (Cox and Mann, 2008), is referred to as a peptide intensity. A more comprehensive description of the LFQ workflow can be found in a paper by Sandin et al. (2011). For the purposes of this paper we will focus on only the experimental details which motivated our statistical model.

3.1.2 Matched Pairs Data

Mass spectrometers work with ions because advances in technology have given us a tremendous ability to manipulate ions. For this reason ionization of peptide molecules is an indispensable aspect of a mass spectrometry proteomics experiment. Unfortunately, and this is a critical point, not all of the peptides from the sample will be ionized. Certain peptides tend to ionize more efficiently, while others will not ionize at all. The probability that a given peptide molecule will be ionized can be referred to as ionization efficiency. Ionization efficiency is believed to be a property of both the chemical structure of each peptide and the presence of other co-eluting peptides, sometimes referred to as matrix interferences. Schliekelman and Liu (2014) found that competition for charge between background peptides may actually be a more important factor than abundance in determining if a peptide will be detected. Regardless of what factors are most important, ionization efficiency can cause the proportion of peptides that enter into the

Table 3.1: This table shows the relationship between relative protein abundance and the intensities of a peptide belonging to that protein. p is the probability that the peptide ionizes and makes its way into the mass spectrometer. pW and pZ represent the expected intensities from samples A and B respectively.

	Protein Abundance	Peptide Abundance	Ion Abundance
Sample A	X	W	pW
Sample B	Y	Z	pZ
Ratio	$\frac{X}{Y} = \mu$	$\frac{W}{Z} = \frac{X}{Y} = \mu$	$\frac{pW}{pZ} = \mu$

mass spectrometer to be drastically altered. This is why we previously claimed that peptide intensities are not a monotone increasing function of peptide concentrations. One peptide might be far more abundant than another in the original sample but a lower ionization efficiency could reverse the relationship for peptide intensities. What we observe is the number of peptides in solution multiplied by that peptide's ionization efficiency. Fortunately, if the efficiency parameter is a property of the individual peptide, it will cancel out when put into a ratio with the same peptide from the other sample. This relationship is outlined in Table 3.1. In a SILAC experiment every peptide in both samples will be processed at the same time and thus will be exposed to the same conditions yielding identical ionization efficiencies. However, even in a SILAC replicate the efficiency may be altered due to variations in sample preparation and elution time resulting in a different profile for the background peptides. For a Label-Free experiment, run to run variation should always be expected. So unless the researchers have good reason to assume the ionization efficiencies will be equivalent, the outcomes may be confounded by run to run variation affecting each peptide differently. The incomplete and inconsistent ionization of analytes makes it impossible to accurately measure the abundance of proteins from the original sample. However in ratio form we can still make inference about the relative abundance of proteins in two samples. This is why proteomics experiments are often referred to as relative quantification experiments and it is why popular proteomics software packages, such as MaxQuant (Cox and Mann, 2008) estimate log protein ratios as the median of log peptide ratios. Similar methods dominate the techniques discussed by Eidhammer et al. (2008). Despite the ubiquity of median ratio estimates in proteomics software packages, none of the models we found in the literature make use of peptide

level ratios beyond including peptide as a covariate in a linear model. Instead, most statistical methods estimate the log protein ratio from Sample A to B by taking the average log intensity across all peptides within the protein from Sample A and then subtracting the average log intensity from all peptides within the protein from sample B. In the absence of missing data these methods are almost identical since $E(X - Y) = E(X) - E(Y)$. However, in the presence of wide-spread missing data many peptides are detected in only one of the two samples. This makes it difficult to interpret the results from a method based on average intensities. Unfortunately wide-spread missingness is unavoidable in a discovery mass spectrometry experiment as explained below.

3.1.3 Intensity-Dependent Missingness

Unlike microarray experiments in which missing values often comprise about 1-11% of the data (de Brevern et al., 2004), proteomics data sets almost always have a much higher percentage of missing data. In the dataset analyzed in this paper 25% of the peptides were missing and according to Karpievitch et al. (2009), 50% missing values are not uncommon. For this reason, the way we conceptualize and treat the missing data will take on huge importance. Here we will briefly discuss some of the largest causes of missing data.

Detection Limit

Mass spectrometers have both theoretical and a practical limits of detection (LOD). The theoretical LOD is the minimum number of ions a given instrument can capture and produce an ion current with adequate signal enhancement. Although any peptide exceeding this number of ions can theoretically be detected by the mass spectrometer every sample contains far more than one peptide which results in a considerable amount of noise. This noise results in a practical detection limit, which is dependent on the sample itself, whereby the software fails to distinguish peptide peaks from background noise. For this reason, sample-related factors that either result in a higher practical detection limit or a decreased intensity due to the nature of the sample can result in missing values. As previously discussed, a major driver in this setting is the peptide

ionization efficiency. If the ionization efficiency is low then the intensity will be low and may fall below the detection limit. This is clearly a form of non-ignorable missingness where the probability of being missing is directly related to the magnitude of the intensity.

Data-Dependent Tandem Mass Spectrometry

Most mass spectrometers performing data-dependent analysis (DDA) do not succeed in fragmenting every ionized peptide. Peptides are mass selected (isolated) for a tandem MS (MS/MS) step according to their intensity rank order. This tandem mass spectrometry step is where peptide identification occurs, so any peptide signal that is not selected for tandem mass spectrometry will not yield useful data. On a side note, in an iTRAQ experiment quantification also occurs in the tandem step whereas in LFQ and SILAC experiments quantification occurs during the first MS step. This is why our proposed missing data mechanism does not apply to iTRAQ experiments. The data that we analyze in this paper was generated from a Q Exactive mass spectrometer made by Thermo Scientific, which was only capable of capturing approximately 80% of the peak intensities above the LOD. Thus even above the practical LOD an intensity dependent process can result in missing values. The number of detectable (identifiable) features can be increased with advances in the operating speed of the device, and complete parallelization can be achieved in newer Orbitrap instruments (Lesur and Domon, 2015). However, this results in a trade off where the identification process becomes less certain. In most settings we will have to consider two sources of non ignorable missingness, one which occurs below a frequently changing detection limit and another above.

Misidentified/Unmatchable/Razor Peptides

A peptide might appear in one sample and not in another simply because it was misidentified. Matching algorithms are designed to minimize this problem but it undoubtedly still exists. A similar problem comes from razor peptides. These are peptides that are properly identified but that could belong to more than one protein. Many software programs assign razor peptides to

the protein that has the most peptide level evidence based on the Occam's Razor concept of protein parsimony (Cox and Mann, 2008). Yet this process could result in misidentification and consequently, missing values. It is also possible that a particular peptide will simply fail to be identified with any certainty which will also result in missing values. It is probably safe to classify missingness caused by classification errors as missing at random.

Modeling Missingness

Many efforts have been made to correct for missing data biases in mass spectrometry experiments. A review of missing data techniques in proteomics by Taylor et al. (2013) compared three methods for removing missing data bias: an accelerated failure time (AFT) model by Tekwe et al. (2012), a mixture model proposed by Karpievitch et al. (2009) and K-Nearest Neighbors (KNN) imputation as described by Troyanskaya et al. (2001). Notable additions to this group include three Bayesian methods. One method proposed by Luo et al. (2009) models missing probabilities with a logit function. The Bayesian model of Lucas et al. (2012) attempts to account for missingness caused by misidentification and Koopmans et al. (2014) proposes a model which allows for a random detection limit. However, none of these methods utilize peptide level ratios in their solutions which leaves room for improvement. On their merits as purely missing data techniques, only the model by Luo et al. (2009) theoretically accounts for all the sources of missingness described above. The AFT model and the mixture model both assume a fixed detection limit. But we should expect this detection limit to vary from peptide to peptide depending on what other compounds are being processed in the background. Furthermore, both of them assume that any missingness above this theoretical detection limit should be missing at random with the mixture model explicitly categorizing all missingness as either at random or due to a detection limit. The use of any technology which uses Data Dependent Analysis will make this assumption false since data dependent analysis is a form of intensity dependent missingness that occurs above the detection limit. The interesting effort by Koopmans et al. (2014) which attempts to model a random detection limit analyzes data at the protein level. In other words, the estimation within an experiment has already occurred and it is unclear what missing data

bias can be corrected. As for the KNN approach, it is difficult to imagine that there even could be a theoretical justification for imputing 40% of a dataset and then proceeding with an analysis as though the observations were real. Nonetheless, this option has made its way into software packages such as Inferno (Formerly called Dante) described by Polpitiya et al. (2008), so we will examine its efficacy later. The problem of ascertaining the cause of a missing value is probably intractable. However, we can say with some confidence that the probability of an intensity being missing should be a monotone increasing function of the intensity. Although the exact conditions and sources of missingness will vary from peptide to peptide and experiment to experiment a single monotone parametrized function of the probability of missingness could serve as a useful approximation for the conglomeration of missing data sources. For this reason, along with some mathematical niceties, we model a probit missing data mechanism such that for each peptide the probability of being observed is given by $\Phi(a + by)$ where Φ is the standard normal CDF, y is the peptide intensity, and a and b are missingness parameters to be estimated in our analysis.

3.2 Methods

In order to symmetrically model log peptide intensities, for the rest of the paper we will refer to intensities only on the log scale, for each peptide we frame the problem in terms of the protein fold change (the difference between the intensities) and the midpoint of the two peptides,

$$Y_{ijk} \sim N(\alpha_{j(i)} + (-1)^{k+1}\frac{\mu_i}{2}, \sigma)$$

where $Y_{i,j,k}$ is the intensity of the jth peptide within the ith protein from the kth sample, $k = 1, 2$, $i = 1, \ldots, n$ indexes the unique proteins in the samples and $j = 1, \ldots, m_i$ indexes the peptides within the ith protein, $\alpha_{j(i)}$ represents the midpoint of the two intensities of peptide j within protein i and μ_i represents the protein fold change. The notation $N(\beta, \sigma)$ denotes a normal random variable with mean β and variance σ. The mixed model definition is completed with $\alpha_{j(i)} \sim N(\beta_\alpha, \xi)$ independent from $\mu_i \sim N(\beta_\mu, \tau)$.

This midpoint mixed model (M3) provides a symmetric framework for analyzing a proteomics experiment in one statistical model while accounting for the matched pairs nature of the data. The model can easily be fit using standard software such as PROC MIXED in SAS or lme from the NLME package in R. We expect this model to provide similar results to standard ratio-based methods and improved downstream analysis by creating a single estimate of experimental variance. In fact, if we used fixed effects in place of random effects M3 is a reparameterization of a linear model with covariates for peptide within protein, sample, and a sample*protein interaction. Of course this model completely fails to account for missing data bias. We expand the M3 model into a selection model with a probit missingness mechanism. We refer to this new model as the midpoint mixed model with a missingness mechanism (M5). Let $I()$ be an indicator function so that $R_{ijk} = I(Y_{ijk}$ is observed). We assume $(R_{ijk}|Y_{ijk}) \sim Bernoulli(\Phi(a + bY_{ijk}))$.

Fitting this model is greatly complicated by the number of missing values in a proteomics experiment. In our dataset, there were certain proteins with over 200 missing values, and integrating the likelihood 200 times created computational difficulties. We resolved this issue by giving each parameter a non-informative prior and using a Gibbs Sampler. The Gibbs Sampler required three sampling steps that were non-standard: the distribution of a missing value given everything else $f_{(Y_{ijk}|\mu_i,\alpha_i,Y_{ijk'},\theta,\mathbf{R})}$, where $Y_{ijk'}$ is the matched pair corresponding to the missing value and θ is the vector of parameters $(a, b, \tau, \xi, \sigma, \beta_\alpha, \beta_\mu)$; the distribution of a protein fold change given everything else $f_{(\mu_i|\alpha_i,\mathbf{Y},\theta,\mathbf{R})}$; and the distribution of a midpoint given everything else $f_{(\alpha_{j(i)}|\mu_i,\mathbf{Y},\theta,\mathbf{R})}$. A bit of manipulation reveals that the distribution of a missing value follows the Extended Skew Normal distribution as described by Azzalini and Capitanio (2014).

$$f_{(Y_{ijk}|\mu_i,\alpha_{i,j},Y_{ijk'},\theta,\mathbf{R})}(x) = \frac{\phi(\frac{x-\mu_x}{\sqrt{\sigma}})\Phi(-a - bx)}{\sqrt{\sigma}\Phi(\omega)}$$

where

$$\mu_x = \alpha_{j(i)} + (-1)^{k+1}\frac{\mu_i}{2}, \quad \omega = \frac{-a - b\mu_x}{\sqrt{1 + \sigma b^2}}.$$

Table 3.2: The prior and posterior distributions used to complete the model. The parameters with non standard prior and posterior distributions are described in the text.

Parameter	Prior	Posterior
τ	$IG(.001, .001)$	$IG(.001 + n/2, .001 + \frac{\sum \mu_i - \beta_\mu}{2})$
ξ	$IG(.001, .001)$	$IG(.001 + \sum m_i/2, .001 + \frac{\sum \alpha_i - \beta_\alpha}{2})$
σ	$IG(.001, .001)$	$IG(.001 + \sum 2 * m_i/2, .001 + \frac{\sum \epsilon_i}{2})$
β_α	$N(0, 10000)$	$N(\sum \alpha_{i,j}/\xi, (\frac{1}{10000} + \frac{\sum m_i}{\xi}))$
β_μ	$N(0, 10000)$	$N(\sum \mu_i/\tau, (\frac{1}{10000} + \frac{\sum n}{\tau}))$
a	$N(0, 10000)$	Probit Regression Estimation
b	$N(0, 10000)$	Probit Regression Estimation

We also find that

$$(\mu_i|\alpha_i, \mathbf{Y}, \theta, \mathbf{R}) \sim N\left(\frac{\beta_\mu\sigma + \frac{\tau}{2}\sum_j (y_{ij1} - y_{ij2})}{\sigma + \frac{m_i\tau}{2}}, \frac{\sigma\tau}{\sigma + \frac{m_i\tau}{2}}\right),$$

and

$$(\alpha_{ij}|\mu_i, \mathbf{Y}, \theta, \mathbf{R}) \sim N\left(\frac{\beta_\alpha\sigma + \xi(y_{ij1} + y_{ij2})}{\sigma + 2\xi}, \frac{\xi\sigma}{\sigma + 2\xi}\right).$$

Proofs of these results can be found in Appendix 6. Although these formulas are complex the results are somewhat intuitive. Each missing value from a pair of points comes from a skew normal distribution where the skew is determined by the missing data mechanism. The fold change comes from a distribution centered around a weighted average of the mean protein fold change and the average of the observed differences in peptide intensities. The peptide midpoint is drawn from a distribution centered around a weighted average of the mean peptide midpoint and the observed midpoint of the pair of intensities.

The Bayesian model formulation is completed with the priors in Table 3.2. The posterior distribution of (a, b) is estimated by fitting the probit regression model

$$\Phi^{-1}(E[R_{ijk}|y_{ijk}]) = a + by_{ijk}$$

The posterior distribution is then approximated as

$$\begin{pmatrix} a \\ b \end{pmatrix} \sim N(\begin{pmatrix} \hat{a} \\ \hat{b} \end{pmatrix}, \hat{\Sigma})$$

Where $\hat{a}, \hat{b}$ and $\hat{\Sigma}$ are the parameter estimates from the probit regression and their corresponding covariance estimate, respectively. The bivariate normal distribution used here approximates the posterior distribution as a consequence of Bayesian large sample theory (Gelman et al., 2004, chap. 4).

3.2.1 Simulation Study

To explore the potential benefits of the M5 model, we conduct simulations to compare the accuracy of our estimates to six other estimation procedures. In addition to the M3 and M5 models, we analyze the commonly used method of median ratios, a one-way ANOVA for protein within sample, QRollup and QRollup performed after implementing a weighted K-Nearest Neighbors imputation (KNNQ). QRollup is a method of analyzing average intensities that seeks to avoid missing data bias by analyzing only the largest 66% of the intensities within each protein/sample combination. A software package called Inferno implements the QRollup method(Polpitiya et al., 2008). Similar approaches are discussed by Eidhammer et al. (2008). The Inferno software also offers an option to use Weighted K-Nearest Neighbors Imputation, which is why we coupled imputation with the QRollup method. The ANOVA model is intentionally simpler than that described by Oberg and Mahoney (2012), as we wanted to use an example of a model where proteins are estimated as the average intensity within groups regardless of the presence of a matched pair. Linear models that include peptide as a covariate should perform similarly to the M3 model. This set of methods gives us a look at the performance of three methods based on ratios and three methods based on average intensities. All methods, except for M3 and M5, are currently supported by proteomic software packages.

In our simulation the hyperparameter values were

$$\tau = 9, \xi = 4, \sigma = .3, a = -9, b = .5, \beta_\alpha = 18.5, \beta_\mu = 0.$$

We generated 500 protein fold changes from a $N(\beta_\mu, \tau)$ distribution and generated the number of constituent peptides within each protein by sampling with replacement from the set $\{1, ..., 12\}$. For each peptide we generated independent random midpoints from a $N(\beta_\alpha, \xi)$ distribution. We generated independent residual errors, ϵ_{ijk} as $N(0, \sigma)$ random variables. Then we created intensities $y_{ijk} = \alpha_{j(i)} + (-1)^{k+1}\frac{\mu_i}{2} + \epsilon_{ijk}$. Next we simulated missingness by computing probabilities of missingness for each intensity as $p_{ijk} = \Phi(a + by_{ijk})$, then we randomly drew Bernoulli random variables, (R_{ijk}), according to those probabilities, to identify which y_{ijk} are missing. We then fit all six models including 1,000 draws from the M5 Gibbs Sampler to create M5 estimates. Results were recorded and the whole process was repeated 100 times.

3.2.2 Protein Categories

Before comparing the six methods, some classification of observation patterns is needed since not all methods are capable of estimating the same proteins. To this end we classify each protein as "matched", "unmatched", "one-sided" or "missing". Figure 3.1 presents a visual depiction.

Missing proteins are proteins for which peptides were identified but no peptide intensities were observed. These are not interesting and even though they can be estimated with M3, M5 or KNNQ we recommend just removing them from the study. Matched proteins are proteins that have at least one matched peptide pair. With at least one shared peptide from each sample, all of the methods can be used for estimation. An unmatched protein has observed intensities from each sample but no peptides that are quantified in both samples. One-Sided proteins have intensities from peptides in only one sample and are completely missing in the other. This can be indicative of a large fold change difference. M5, M3 and KNNQ can be used to estimate all types of proteins. The ANOVA model and QRollup can be used for both matched and unmatched proteins while the method of median ratios can only be used on matched proteins.

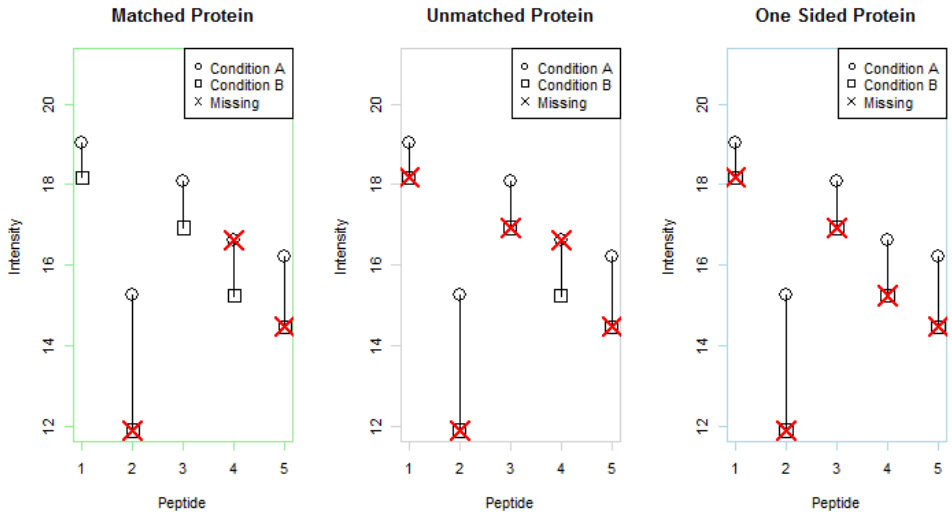

Figure 3.1: The three categories of proteins. Matched proteins contain at least one matched peptide pair. Unmatched proteins contain data from both conditions but no matched pairs. One-sided proteins contain peptide measurements from only one sample.

3.2.3 Simulation Results

The sampling chains all appeared to achieve stationary distributions after about 300 draws (in the real data this was achieved within 50). For this reason, our estimates were based on the posterior mean after a burn in length of 500 draws.

Figure 3.2 shows the distribution of mean squared errors across simulations. The most obvious result here is that the methods based on ratios are far outperforming methods based on average intensities. M5 demonstrates the best performance with an average MSE of 0.26. The method of medians was the second most accurate with an MSE of 0.35, which represents an increase in error of 35%. This is a fairly large increase but it is hardly noticeable relative to the error coming from the average intensity methods. The best of these was QRollup with an average MSE of 1, which represents an increase of 285%. It should be noted that the commonly used validation tool of correlation does not do a very good job of assessing algorithmic weaknesses here. Figure 3.3 shows that even though some of these methods more than sextuple mean squared error, the lowest correlation coefficient is still above 0.9. It should also be noted that the use of Weighted

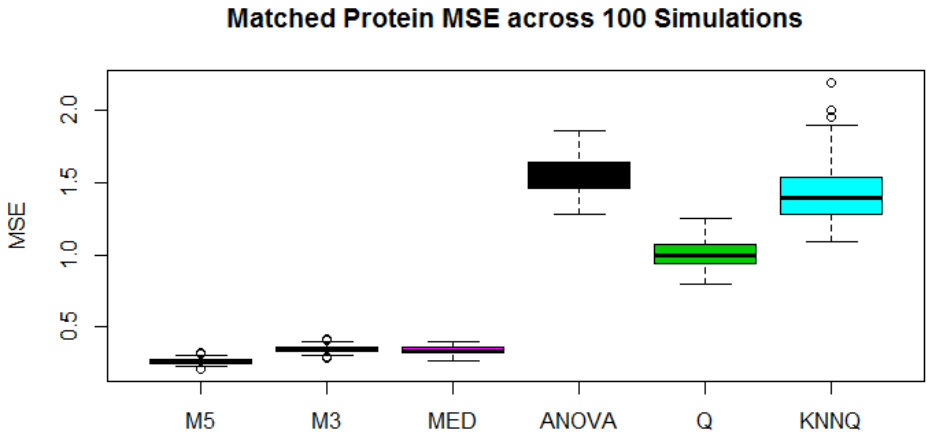

Figure 3.2: MSE for each method computed across matched Proteins and within each simulation.

K-Nearest Neighbors appears to be detrimental to the accuracy of QRollup estimates.

These relationships can be further explored by categorizing proteins according to the percentage of peptides which are missing as shown in Figure 3.4. This plot shows that the error for the KNNQ method increases substantially once more than 50% of the data requires imputation. In this chart we can see that, as missingness increases, the ANOVA estimation also loses accuracy at a much faster rate than the other methods. This is likely because the ANOVA model simply reports average intensities within each sample regardless of the amount of missing data.

In the case of one-sided and unmatched Proteins the method of medians is obviously not applicable. Among the other methods the rank ordering based on average MSE remains the same. (see Figure 3.5).

In this case the average MSE for M5 is 1.5 and the second best is the M3 model at 2.4. The best average intensity method was the ANOVA model with an MSE of 3. Correlation coefficients are much weaker in this category as pictured in Figure 3.6.

Arguably the greatest advantage to using the M5 model comes from the ability to estimate one-sided proteins. These proteins are difficult to estimate since one of the samples provides no

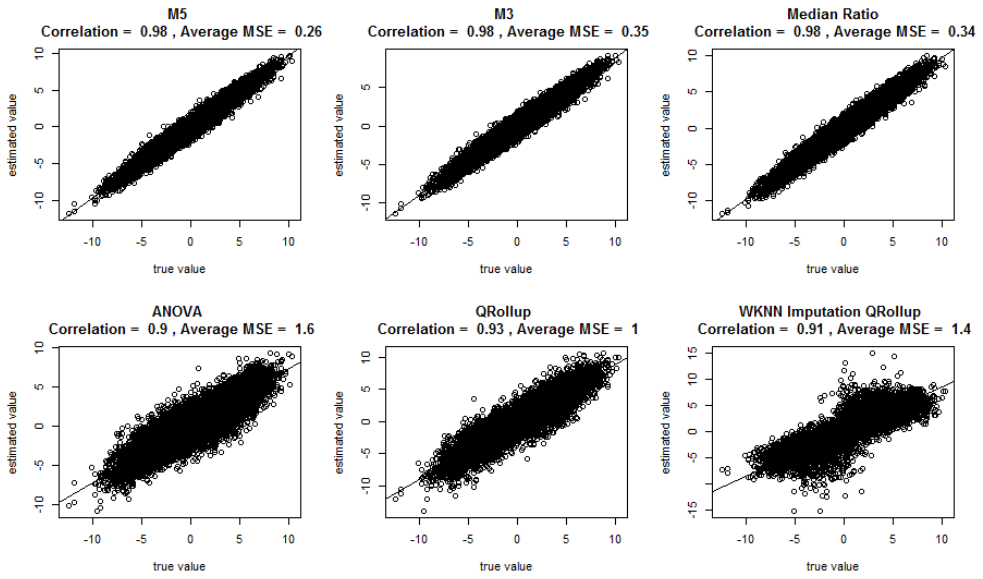

Figure 3.3: Scatterplot of true simulated fold changes for matched Proteins vs their estimates across all simulations. Correlation coefficients are also computed across all simulations.

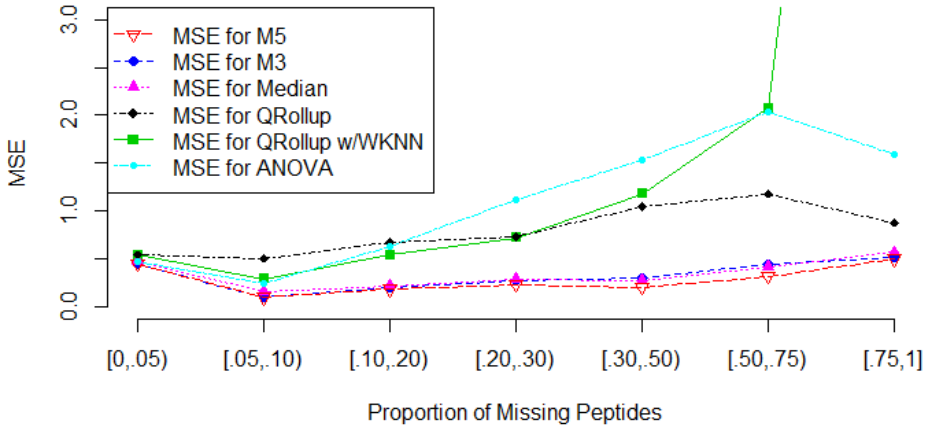

Figure 3.4: MSE for each method according to categories of the percentage of missing peptides. The MSE for KNNQ at 75% missing data is 12.81 which was too extreme to be plotted along with the other methods.

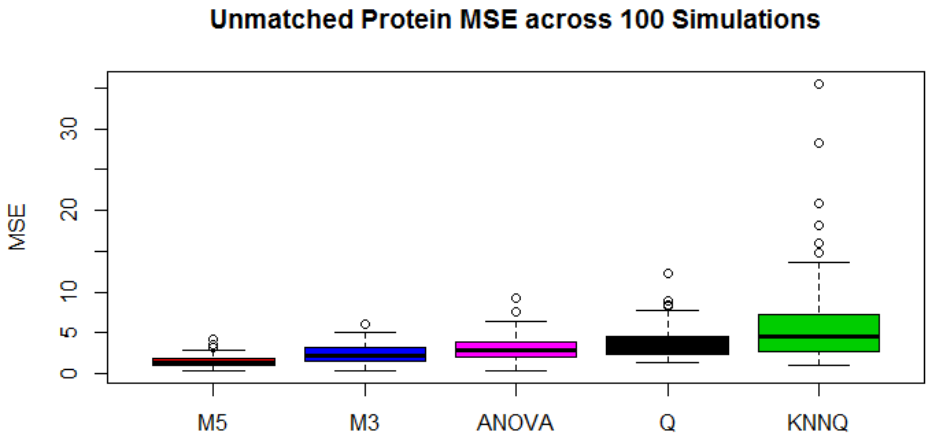

Figure 3.5: MSE for each method computed across unmatched proteins and within each simulation. The method of medians, MED, is not applicable to unmatched proteins.

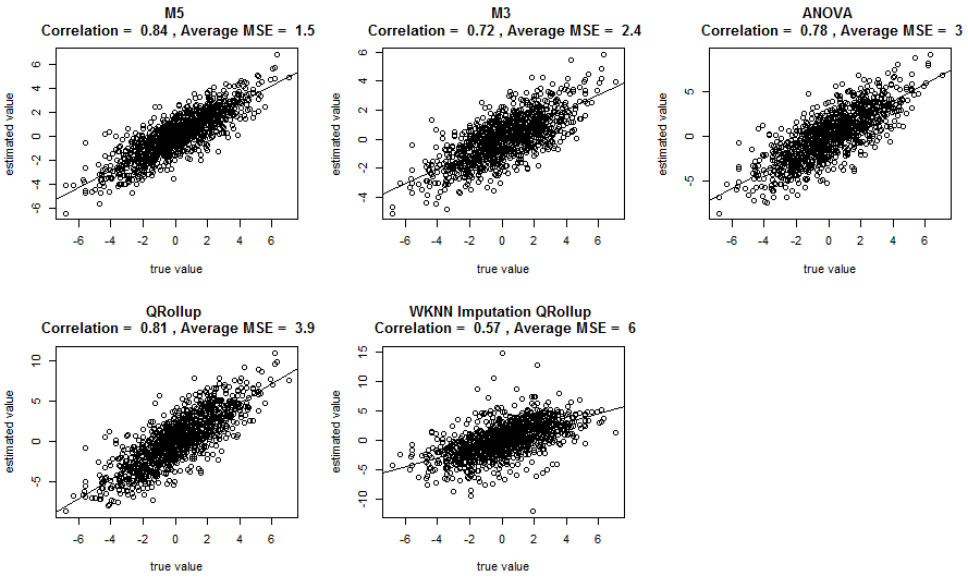

Figure 3.6: Scatterplot of true simulated fold changes for unmatched proteins vs their estimates across all simulations. Correlation coefficients are computed across all simulations.

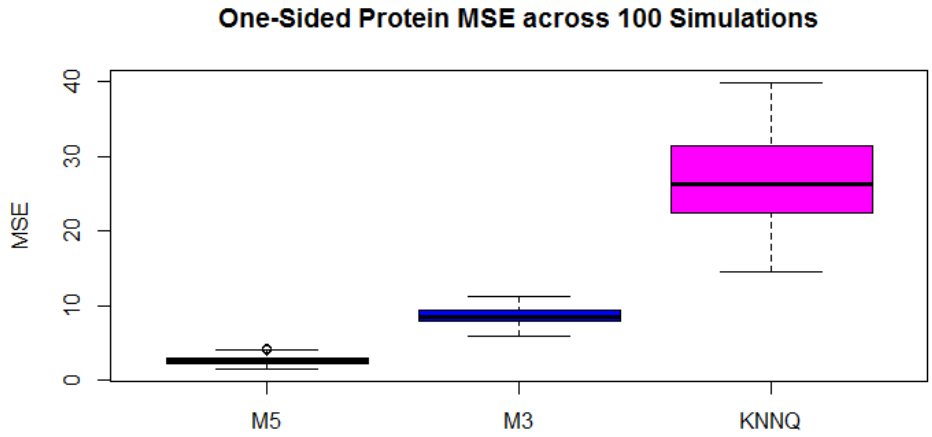

Figure 3.7: MSE for each method computed across One-Sided proteins and within each simulation. The method of medians (MED), ANOVA, and QRollup (Q) are not applicable to One-Sided proteins

observed values. Keep in mind for a One-Sided protein that we could assume that the abundance of the missing value is between zero and the detection limit. However, the upper bound on an abundance ratio is infinite. Nonetheless, M5 does provide decent estimates in this situation as can be seen in Figure 3.7. Only three methods were capable of estimating one-sided proteins and only one of them could be considered useful. The MSE's for M5, M3 and KNNQ were respectively 2.7, 8.6 and 27. The range of the log-scale fold changes in this simulation were roughly -10 to 10. So an average MSE for M5 of 2.7 is certainly small enough for the estimates to be of interest. The scatter plot in Figure 3.8 strongly highlights the advantages of the M5 model.

3.3 Breast Cancer Data

In order to make sure the results of our simulation study are not artifacts of the data generation procedure, we also analyzed the effect of non-informative missingness on a real data set. The data, generated by the Chen Biochemistry Lab, contains peptide level LFQ measurements from two samples of breast cancer tissue (one Basal and one Luminal). This dataset can be found in

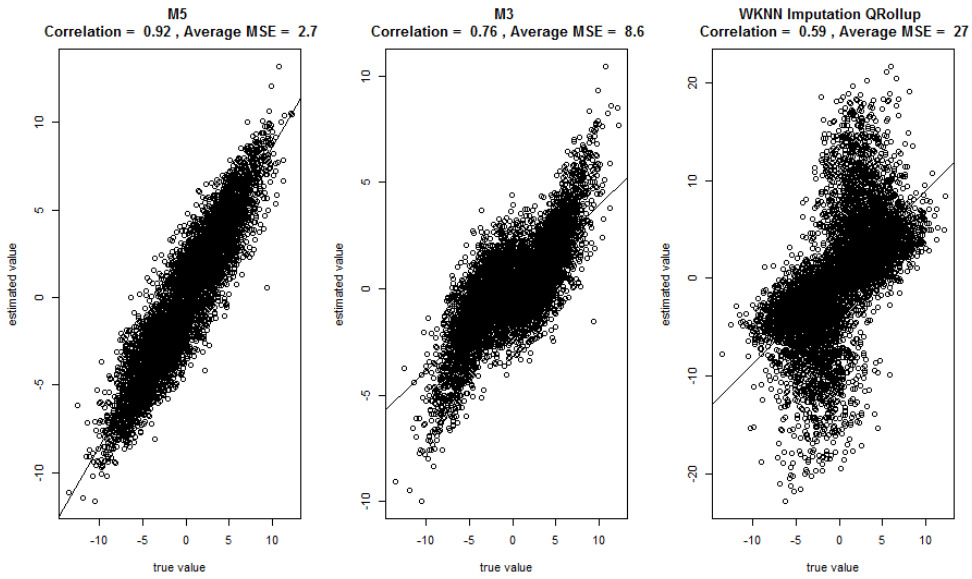

Figure 3.8: Scatter plot of true simulated fold changes for one-sided proteins vs their estimates across all simulations. Correlation coefficients are computed across all simulations.

the supplementary material. 11,866 unique proteins were identified in the data, of these 594 were Missing, 9,265 had at least one peptide pair, 1,810 were one-sided and 197 had intensities in both samples but no matched pairs. This breakdown is pictured in Figure 3.9.

Not considering missing proteins, we can see that before we even do an analysis the M5 model is capable of estimating an additional 2,007 (22%) proteins compared with the method of medians. This would be a substantial gain if our method is capable of estimating those proteins

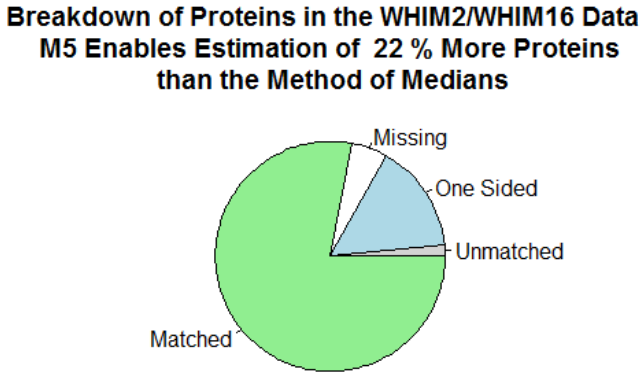

Figure 3.9: Distribution of proteins in the tumor data.

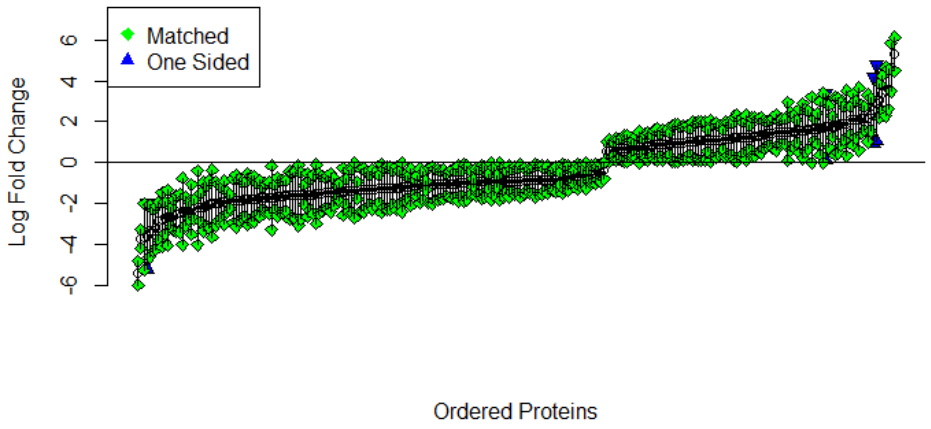

Figure 3.10: M5 estimates of the log fold change between proteins found in Basal and Luminal breast cancer tissues. Only proteins with 95% credible intervals that do not contain zero are pictured.

with a decent level of accuracy as our simulations suggest. Furthermore, the entire data set contains information on 248,342 peptides, 61,418 (about 25%) of which are missing. There is a tremendous amount of information in the patterns of those 61,418 missing values, and in theory the M5 model takes full advantage of them. M5 model estimates were computed on a random set of 1,000 proteins and 95% credible regions were computed. 1,000 draws from the Gibbs sampler were used with a burn in length of 500. We reduced the data size purely for computational simplicity. From the 1,000 proteins, 252 matched Proteins and 4 one-sided Proteins did not contain zero in their credible intervals (Figure 3.10). Among the 4 one-sided Proteins is protein O75363 which is better known as the gene product for the Novel Amplified in Breast Cancer-1 gene (NABC1). This gene is known to be involved in cancer typically being up-regulated in breast cancers and down-regulated in colon cancer (Beardsley et al., 2003). Since this protein was one-sided in our dataset no useful information regarding NABC1 would have been found without the M5 model. After estimation, we computed the square of the M5 estimates divided by the posterior standard deviation. The proteins were then ranked in descending order and are presented in Table 3.3. We also used the real data to study the effects of non-ignorable missingness

Table 3.3: Twenty proteins ordered by the highest squared ratio of posterior mean to posterior standard error. At the bottom of the table is the complete set of one-sided proteins for which the credible region did not include zero.

Protein	Estimate	Posterior SD	Category
P48681	-5.39	0.300	Matched
O95425	-3.72	0.230	Matched
O76070	5.31	0.420	Matched
Q13557	-2.64	0.231	Matched
F8VTL3	-1.89	0.169	Matched
Q07065	-2.33	0.211	Matched
Q13813	-0.97	0.096	Matched
P12270	-1.12	0.112	Matched
G3XAI2	-2.26	0.241	Matched
Q9Y4L1	-1.42	0.163	Matched
P97457	4.653	0.597	Matched
Q86SF2	3.39	0.436	Matched
O60231	-1.93	0.25	Matched
Q13363	3.58	0.501	Matched
Q9NX62	2.16	0.302	Matched
Q5T6V5	3.02	0.426	Matched
Q86WJ1	-2.70	0.386	Matched
P23786	1.77	0.261	Matched
Q6BCY4	-2.55	0.376	Matched
Q9NP74	-2.37	0.357	Matched
P12109	-3.59	0.813	One-sided
Q16666	1.76	0.823	One-sided
B4E1Z4	2.55	0.817	One-sided
O75363	2.90	0.944	One-sided

on each of the six estimation methods. To accomplish this goal we first reduce the data to allow a complete case analysis, so that only peptides with observed intensities from both samples are included in the reduced dataset. From this complete-case data, 500 proteins were randomly selected for a sensitivity analysis. The mean peptide ratio within each protein was calculated and considered to be the reference value. We explored what happens to the estimates from each model as higher levels of intensity-dependent missingness are introduced. Appropriate values of the missingness parameter b were discovered by trial and error to provide overall missingness levels of 1, 5, 10, 20, 30, 40 and 50 percent. Mean squared error was then calculated for each of the six methods on all 7 datasets.

3.3.1 Results of the Sensitivity Analysis

We explored the effect of intensity dependent missingness on complete case estimates. The performance demonstrated similar patterns to what we found in the simulation analysis with ratio based methods having far more stable estimates than the average intensity based methods, shown in Figure 3.11.

These plots paint a picture consistent with the results from the simulation study. The M5 model outperforms all other methods. The method of median ratios and the M3 model have very similar performance. The ANOVA model and QRollup methods perform comparably to the ratio-based methods until the missingness is increased to around 10%. Once missingness hits 40%, the difference in frameworks becomes substantial, and at 50% the average MSE from KNNQ is about eleven times higher than that from the M5 model.

3.4 Misspecification of the Missing Data Mechanism

An obvious artificial strength of the simulation study is the use of the same missing data mechanism in both the simulation and the analysis. The scientific process supports the use of a missing data mechanism in which the probability of a peptide being observed is a monotone increasing function of the intensity. Beyond this basic structure very little evidence exists to

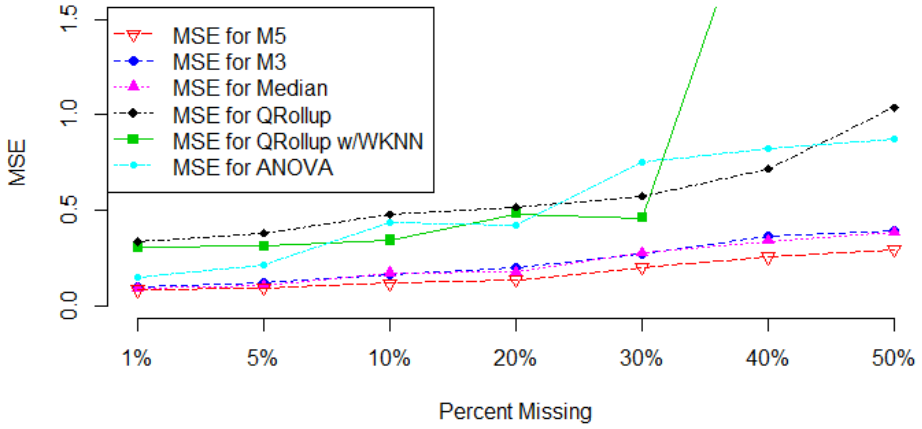

Figure 3.11: MSE computed as the average squared difference between the estimate from a complete case analysis and the estimate from a dataset with simulated intensity dependent missingness. MSE for QRollup with weighted KNN imputation takes the values 2.34 and 3.31 at 40 and 50 percent missingness respectively. These values were too extreme to be plotted with the other methods.

suggest the proper shape of this curve. Our probit model fits the monotonicity requirement however it is not unique in doing so. To examine the robustness to misspecification of the missing data mechanism we compare estimation results from three different missing data mechanisms; a linear model within a probit function, a quadratic model within a probit function and a linear model within a logit function. Data was simulated 100 times and for each data set a different set of missing values was simulated according to the three different models. Simulation parameters were selected so that the overall percentage of missing values would be near 33%. As pictured in Figure 3.12, the results suggest that the M5 model is fairly robust to misspecification of the missing data mechanism. Amongst matched proteins the worst case scenario occurred when the real mechanism was a logit model. In this case the average MSE increased by 8% from .26 to .28, which is still 20% lower than the MSE for the method of medians found in the simulation study. Misspecification from a quadratic model actually reduced the average MSE by 10%. These results seem to suggest that sharper the increase in the probability of observing a peptide the better our model will perform. For one-sided and unmatched proteins the effects of misspecification

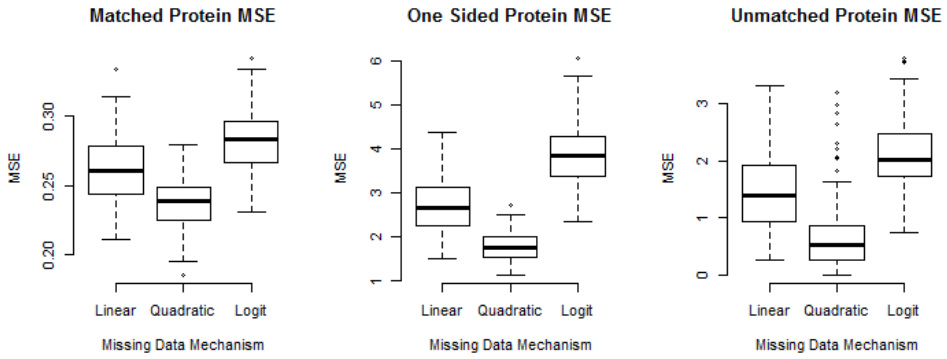

Figure 3.12: MSE computed from 100 simulations utilizing 3 different missing data mechanisms.

were more pronounced. For one-sided proteins we observed a 42% increase in average MSE with a probit misspecification and a 34% decrease for the quadratic model. For unmatched proteins these changes were 43% and -53% respectively. The increased effect of the missing data mechanism for these proteins should not be surprising since the mechanism plays a larger role in the estimation when no matched pairs are observed. In the worst case scenario the average MSE for a one-sided Protein from a logit missing data model was 3.87. With a range of fold changes in the data from roughly -10 to 10 an MSE of 3.87 is highly encouraging as it suggests that even with misspecification the M5 model provides a legitimate way to detect one-sided proteins with large fold changes.

3.5 Conclusion

We have identified the two fundamental statistical features of mass spectrometry proteomics as matched pairs data and non-ignorable missingness. Of the two features, ignoring matched pairs appears to be far more detrimental than ignoring the missing data bias. Not only is the average intensity across peptides difficult to interpret, but the simple method of taking the median ratio greatly outperforms methods based on average intensities in terms of mean squared error. In turn, relative to the method of medians, our M5 model is capable of improving both the depth and accuracy of the mass spectrometry experiment. To the best of our knowledge, this is the first

model that provides reliable estimation of one-sided proteins and our experiments and sensitivity analysis suggests that these estimates could be valuable even when the missing data mechanism has been misspecified. A great deal of work can be done to extend this model. Here we have only demonstrated the importance of peptide level matching with a model that compares two samples from either a SILAC or LFQ experiment. A different missing data mechanism would be required to fit iTRAQ data and extensions for multiple sample comparisons and sample fractionation are not immediately obvious. In whatever way these more complicated problems might be solved, the lessons from this study should be incorporated into the solution. The joint presence of matched pairs data and non-ignorable missingness can greatly inflate the error in estimation procedures that do not explicitly account for both features.

CHAPTER 4: EXTENDING BEYOND THE TWO SAMPLE EXPERIMENT

4.1 Introduction

Mass spectrometry proteomics experiments are often referred to as relative quantification experiments. This is because the technology only enables inference to the ratio of protein abundances without ever obtaining measures of absolute abundance. Even in the most basic case, where two samples are being compared, variations in ionization efficiency and widespread missing data can substantially complicate the task of creating a formal statistical model. In the previous chapters potential solutions to these complications were discussed. However, modeling is further complicated by experiments that contain multiple samples and replicate data. If there are K samples to be compared then there are K choose 2 ratios that can be estimated which can be a source of confusion for statisticians attempting to model proteomics data. In this chapter we propose models for iTRAQ and SILAC experiments which account for ionization efficiency, missing data and which apply to arbitrary numbers of samples and replicates. Understanding the motivation for our model requires explanations of ionization efficiency and the nature of missing data in a mass spectrometry experiment.

4.1.1 Model construction

The factors in our experiment are protein, peptide within protein, sample and run. Suppose that we have $i = 1, ..., I$ proteins, $j(i) = 1, ..., J_i$ peptides within each protein, $k = 1, ..., K$ samples to compare and $l = 1, ..., L$ runs. Then for a given peptide the number of molecules in a sample should depend on the sample the peptide and possibly some systematic experimental deviations in the form of a run effect. The obvious model for mean peptide molecule abundance is given by

$$E(a_{ijkl}) = \beta_0 + \alpha_i + \beta_{j(i)} + \gamma_k + \delta_l + \alpha\gamma_{ik}.$$

However these variables are not the same as those that affect ionization efficiency. In both SILAC and iTRAQ experiments we have good reason to believe that the ionization efficiency for a peptide will be the same across samples, since they are all ionized under the same conditions. However, the efficiency can vary dramatically from run to run. Richard Knochenmuss convincingly demonstrates that when using Matrix Assisted Laser Desorption/Ionization (MALDI), the most common ionization technology used in iTRAQ experiments, by simply altering the amount of sample processed the rank order of the intensities measured can be completely reversed (Knochenmuss, 2012). He ascribes this behavior to the complicated nature of electrons transfers that occur in the ion plume. Essentially, the factors that determine ionization efficiency are highly multivariate and for the most part unobserved. Furthermore, these unobserved factors are bound to change when experiments are replicated because the background analytes (other chemical structures that are ionized at the same time as a target peptide) will always change due to variations in the elution process. The result is a large amount of variation in peptide level intensities that we would greatly like to remove from our estimation procedure. Schliekelman and Liu (2014) estimated that the effect of background peptides on the probability that a peptide would be observed in a sample exceeded the effect of the abundance of a peptide molecule. For these reasons we have addressed the modeling problem assuming that an interaction term will be needed. If for a particular sample a researcher believes that no background interferences will occur then it will be simple to remove the interaction from our model. This leads us to the following model for the expected ionization efficiency.

$$E(\pi_{ijkl}) = \beta_0^* + \beta_{j(i)}^* + \delta_l^* + \beta\delta_{j(i)l}.$$

Thus a model for the intensities we actually can observe would be given by

$$E(y_{ijkl}) = E(a_{ijkl} + \pi_{ijkl}) = (\beta_0 + \beta_0^*) + \alpha_i + (\beta_{j(i)} + \beta_{j(i)}^*) + \gamma_k + (\delta_l + \delta_l^*) + \alpha\gamma_{ik} + \beta\delta_{j(i)l}.$$

Notice that in this model, though many parameters can only be interpreted as a combination of ionization and abundance effects, the contrast between two samples k and k' with all other factors fixed, $(\gamma_k + \alpha\gamma_{ik} - \gamma_{k'} + \alpha\gamma_{ik'})$, only contains parameters from the abundance model. Thus, even though we only make observations on the number of ions that enter a mass spectrometer, estimating between sample contrasts still provides a way to make inference on the original sample. For this reason we consider the contrasts to be the parameters of interest.

Without explicitly modeling ionization efficiency other researchers have created similar models for proteomics data (Oberg and Vitek, 2009). There are two substantial features that set apart our approach. The first obvious difference is the inclusion of an interaction between peptide and run. Without this all of the variations in ionization efficiency will greatly inflate the error term. The importance of variations in ionization efficiency across runs has not been well studied and may very well depend on the sample being examined. Of course, if a researcher feels certain that no peptide by run interaction will be present, removing the term from the model causes no difficulties. The second major difference is that our model provides a clear explanation of the special role contrasts play in a mass spectrometry proteomics experiment. Without our explicit modeling of ionization efficiency there would be nothing in the model to suggest that the experiment was only intended to estimate absolute abundance. With this model established as a general framework for mass spectrometry proteomics experiments we can now consider the effects of missing data.

4.1.2 Categories of Missing Data

One of many characteristics of a proteomics experiment, which makes it difficult to apply standard statistical methods, is the lack of a priori knowledge about what data will be observed in the experiment. In the terminology of our model, we don't know the values of I or J_i. This

creates some ambiguity when discussing missing data. If we have one row in our data set for each observed peptide there are many ways we could expand the number of rows to contain missing values. The simplest solution is to use a complete case analysis and not expand the rows at all. However as we showed in chapter 3, substantial gains in accuracy and depth of discovery can be made by making use of missing data patterns. However, only two samples were being compared and there were no replicates so it seemed fairly straightforward to establish the definition of a missing value. We now consider more possibilities. At the opposite extreme from a complete case analysis we can consider the all possible peptides model. Since we know how proteins should fragment into peptides we can create lists of all possible peptides based on the complete amino acid sequence of the identified proteins. Thus for every protein observed, rows would be added to the dataset for all possible peptides in every sample and run. This would be a complete expansion of row space to all possible peptide outcomes and consequently if there is any useful information in the missing data patterns, the full expansion would contain it. It is at least conceivable that information on the proportion of an amino acid sequence that manifested into observed peptide intensities could be useful in assessing the variability of our estimates. However, to the best of our knowledge nobody has attempted to analyze all possible peptide expansions. This is likely due to the difficulty of the task as it would require a good deal of database searching and computation just to figure out what rows of missing data should be included in the model. Once included, it is also not clear how much would be gained by the effort. Many of the peptides that were not observed might have had chemical properties such that ionization could not possibly have occurred. Should a researcher consider a data point that could not have conceivably appeared to be missing? If there is a reason to do so it is not immediately obvious. At least three possibilities exist in between the complete case analysis and an all possible peptides model. We seek to clarify the problem of including missing values by classifying peptides into five different levels, shown in table 4.1.

Level zero is the complete case analysis where only peptides that are observed are included in the statistical model and level four is the all possible peptide model. For level one we introduce the concept of verifiably sufficient ionization efficiency. A peptide has sufficient ionization efficiency

Table 4.1: Different levels of missing data that could be accounted for in a statistical model.

Missingness Level	Intensities included in model
Level 4	All possible intensities
Level 3	Intensities for all peptides ever shown to be ionizable
Level 2	Intensities for all peptides shown to be ionizable in the data
Level 1	Intensities with verifiably sufficient ionization efficiency
Level 0	All intensities observed in the data

if the probability of ionization is high enough to conceivably enable the observation of a peptide intensity. Recall that ionization efficiency is a property of not only the peptide but also the background analytes and ultimately the experimental run. We say that a peptide has verifiably sufficient ionization efficiency if the dataset contains any instances of a peptide being observed within a specific run. In other words we know that the ionization efficiency, created by the amino acid sequence and the specific experimental conditions, was not so low as to prevent all observations of the peptide intensity. In terms of the parameter in our model this means that we only include midpoints where at least one of the samples was observed. Level 1 is a subset of level 2, which contains all intensities belonging to peptides which we know could have ionized due to their presence somewhere in the dataset. To clarify the difference, suppose that in an experiment with 4 samples and 2 replicates for peptide $j(i)$ we observe only $y_{j(i),1,1}$ and $y_{j(i),2,1}$. Then using level 1 missingness our model would include peptides $y_{j(i),1,1}, y_{j(i),2,1}, y_{j(i),3,1}, y_{j(i),4,1}$ where the peptides from samples 3 and 4 are missing. Using level 2 missingness our model would include $y_{j(i),1,1}, y_{j(i),2,1}, y_{j(i),3,1}, y_{j(i),2,1}, y_{j(i),1,2}, y_{j(i),2,2}, y_{j(i),3,2}, y_{j(i),4,2}$ adding missing values for all of the peptides that theoretically could have appeared in run 2. In terms of our model, level 1 can be understood as only using midpoints for which at least one data point exists. Level 2 includes all peptides that we know, without considering variations in ionization efficiency, could have been ionized because they were observed at some point in our experiment. If a researcher thinks this is reasonable it is difficult to see why we would stop here. Level 3 is defined to include all intensities belonging to peptides that we have knowledge of being ionizable. This would require information outside of the experiment and it is not immediately obvious how to compile such information but in theory the proteomics literature already contains a tremendous amount of information regarding what amino acid sequences are capable of being ionized with various technologies. The

final all inclusive step is level 4 which contains all of the possible intensities from every possible peptide sequence within each identified protein. We cannot rule out the possibility that methods which utilize level 4 missingness would be optimal, however there is a tradeoff because as the missingness level increases so do the potential sources of missingness.

4.2 Missing Data Mechanisms

Previously we gave a detailed explanation of the causes of missing data in a mass spectrometry proteomics experiment. The causes are numerous and identifying the direct cause for any particular peptide is a hopeless endeavor. However, we have good reason to believe that the probability of observing an intensity will be a monotone increasing function of the intensity itself. For this reason we fit a probit missingness model such that the probability of observing an intensity, y, is given by $\Phi(a+by)$ where Φ is the standard normal CDF, and a and b are missingness parameters to be estimated in our analysis. This missingness mechanism is really an approximation of what we expect to happen when the many different sources of missing data are consider in aggregate. For multi-sample SILAC experiments the same missing data mechanism should be useful. However, the justifications for this mechanism do not apply to iTRAQ experiments. We previously identified an ever changing detection limit, and data dependent analysis, as the two major sources of non-ignorable missingness. Essentially an intensity needs to be sufficiently large to be selected by the operational software for the tandem step of a mass spectrometry experiment. In either a SILAC or a LFQ experiment this tandem step is solely used to identify the peptide. In an iTRAQ experiment all of the quantification also occurs during the tandem step. Critically, isobaric tags do not alter the mass of the peptides from different samples. Thus the intensity from the first MS step is actually a reading of the sum of the intensities across samples. Consequently, we should not expect the probability of missingness to be dependent on an individual peptide y, rather it should be dependent on the sum of intensities across samples. Furthermore, if a given peptide is not selected for tandem mass spectrometry in an iTRAQ experiment then that peptide will be missing in all of the samples. If we only consider Level 1 missing data then intensities that are missing because they were not selected for tandem mass spectrometer will not appear in our

data, not even as missing values. This begs for a re-evaluation of what might cause values to appear as missing in a Level 1 missing data iTRAQ experiment. In this case we have observed at least one peptide intensity while the same peptide from the same run was not observed in at least one of the other samples. At this stage of the experiment we know exactly where the mass of the isobaric tag should be located. We also know that the identification went well since the peptide was identified in another sample. It seems likely that any missing values in this situation are legitimately missing because they fell below the instruments detection limit. In this case, we prefer to take the conservative approach and impute the minimum intensity found in the data set.

To extend our previous work we would like to create extended versions of M3 and M5 for both ms1 and ms2 types of data. Taking the model framework described in this chapter and fitting it with a mixed model will serve as an M3 type estimation for both ms1 and ms2 data. This model will essentially ignore all missing values, but since we are obtaining random effects predictors even the missing proteins will have estimates taken from the overall parameter distributions. In order to extend M5 the full conditional distributions must be derived in order to fit the Gibbs sampler. In particular we find that the distribution for any of the mean parameters in our model is given by

$$(\alpha|\cdot) \sim N\left(\frac{\sigma\beta_\alpha + \tau\sum_{j=i}^{I}(y_j - \mathbf{X}\theta^*_{[j]})}{\sigma + \tau I}, \frac{\tau\sigma}{\sigma + \tau I}\right)$$

and that the distribution of a missing value given everything else, for an ms2 technology follows the skew normal distribution

$$\begin{aligned} f_{(Y_m|\cdot)}(x) &= \frac{\phi(\frac{x-\mu_x}{\sqrt{\sigma}})\Phi(\omega\sqrt{1+(-b\sqrt{\sigma})^2} - b\sqrt{\sigma}(\frac{x-\mu_x}{\sqrt{\sigma}}))}{\sqrt{\sigma}\Phi(\omega)} \\ &= \frac{\phi(\frac{x-\mu_x}{\sqrt{\sigma}})\Phi(-a - b\sum_{j=i}^{n} y_j - bx)}{\sqrt{\sigma}\Phi(\omega)}. \end{aligned}$$

Derivations of these distributions can be found in Appendix A.1. With these derivations completed, all of the machinery is in place to analyze data from more complicated proteomics

experiments.

4.3 Discussion

We have established new categories for missing data, and created extensions to M3 and M5 for both ms1 and ms2 data. However, for every step we take a dozen questions appear. The most obvious question is whether or not there is any value in utilizing Level 2 missing data. We have already demonstrated that Level 1 missing data can be used to improve the accuracy of point estimation and increase the depth of discovery so we expect to find similar results when comparing results from the extensions of M3 and M5 that utilize ms1 missing data. However, it is unclear if the same benefits will be found when using level 2 missing data. This is of particular importance when analyzing ms2 data, where no level 1 missing data exists. This also suggests an obvious way to explore the potential benefit. We construct a simulation experiment where for every data set simulated a set of ms1 and a set of ms2 missing data vectors are generated using either a probit on an individual intensity or on the sum of intensities across samples, respectively. We will then fit the m3 and m5 extensions for both ms1 and ms2 data (four models are fit for each simulated dataset). Boxplots of the simulated mean squared errors are shown in Fig 4.1.

The average MSE across 100 simulations for the M3 model with ms1 missingness was 1.46. The MSE for the M5 model on the same data was 64% lower at .64. Thus the simulations confirm that accuracy can be improved by utilizing missing data patterns in the extended ms1 model. However, the same gains are not seen with the ms2 data. In fact using the ms2 M5 model we obtained an average MSE of .69 which is actually an increase of 23% over the average MSE for M3 of .56. This suggests that using level 2 missing data is not very helpful in regards to point estimation, at least not in the ms2 data. Such a finding might be considered disheartening but there are benefits to knowing that we can safely ignore level 2 missing data in iTRAQ experiments. For very large studies, keeping track of level 2 missing data results in enormous and sparse data sets. Being confident that we are not losing anything by removing this category of missing value from the data could save a lot of computational effort. This one simulation study should not

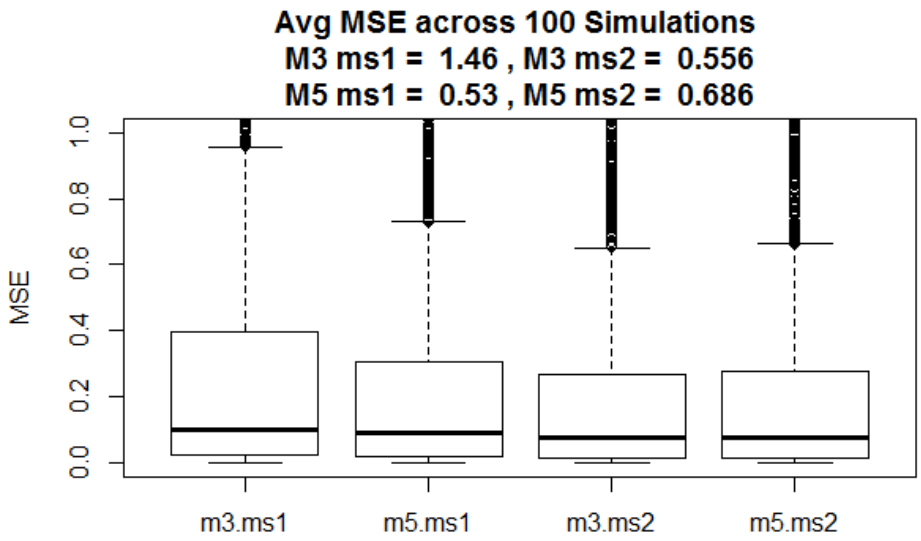

Figure 4.1: Boxplots of the MSE from M3 and M5 models applied to both ms1 and ms2 datasets. The y axis was cutoff at 1 to preserve resolution of the boxplots.

be considered conclusive regarding the usefulness of level 2 missing data. It is possible that the missing values while being unhelpful in point estimation could improve interval estimation. Studies on real data and further simulations regarding the point estimation result would also be nice. However, these studies will be topics for future research.

CHAPTER 5: CLUSTERING INDICES

The task of assessing the quality of a clustering algorithm cannot be addressed in a sensible manner without first understanding the objective of clustering. Unfortunately, there is no clear consensus on how this problem should be defined. A review paper by Jain (2010) described clustering as an ill-posed problem. The basic idea is to partition a set of points into "similar" groups. However, the notion of "similar" is completely subjective. This has led to the development of literally thousands of clustering algorithms, and even minor alterations to these algorithms can result in substantially different clusterings of the same dataset (Jain, 2010). From a statistical perspective, one which includes randomness, it makes sense to define clustering in terms of model parameters. To this end, a large literature has been created on the subject of model based clustering (McLachlan and Basford, 1987). However, we find that model based clustering has largely oversimplified the task of data clustering by reducing the objective to a classification problem. We will demonstrate that even when given the true model for data generation; classification, clustering and linkage detection pose three distinct estimation problems whose solutions do not always overlap. After clarifying these concepts we propose new indices of clustering performance which assess algorithmic performance in terms of sensitivity and specificity. Finally we will demonstrate the advantages of our indices on a variety of simulated and real world situations.

5.1 The Model

We assume that $(X_i, A_i), i = 1, \cdots, n$ are iid random vectors where X_i is a p dimensional vector and A_i takes values in $\{1, \cdots, G\}$. The variate A_i is the unobserved subgroup identifier, and there are G subgroups in the population. The subgroup probabilities are $\pi_g := P(A_i = g), g = 1, \cdots, G$, where $0 < \pi_g < 1, \sum_{g=1}^{G} \pi_g = 1$. Let $f_g(\cdot)$ be the pdf of X_i conditional on

$\{A_i = g\}, g = 1, \cdots, G$. Although it is possible to consider discrete and mixed-type distributions, we assume for simplicity that the cdf's are all continuous with corresponding densities $\{f_g\}$. To avoid trivialities we assume that the G cdf's are all distinct. This is clearly a mixture formulation (McLachlan and Basford, 1987), but it is not restrictive since any distribution can be expressed as a mixture. We note that the supports of the densities $f_1, \cdots, f_G$ need not be distinct and generally will overlap or be the same (as in normal mixtures).

The model parameters are: $(\pi_1, \cdots, \pi_G, f_1, \cdots, f_G)$. The densities can be parametric or non-parametric.

It is worth noting that without specifying the form of the mixture densities, the value of G is completely arbitrary. For example, with $p = 1$, and $X_1, \cdots, X_n$ iid normal(0,1), it may seem that $G = 1$. That is a reasonable choice, but other choices are possible. We can define G to be 2 by expressing the normal(0,1) as a 50:50 mixture of a half normal over $x > 0$ and another (mirror image) half-normal over $x < 0$. A half-normal is obtained from a normal by restricting the range of x to be above the mean or below the mean. Indeed, we can define G to be any positive integer we wish by partitioning the real line into G disjoint sets, and defining the jth mixture component to be a normal(0,1) restricted to the jth set in the partition. However, in this example, specifying that the mixture components are normal, forces the choice $G = 1$. This shows that without adequate specification of the mixture components, the very definition of G can be problematic.

With the clustering model in place we can now investigate the nature of the clustering problem from a statistical perspective. Given the true model there are various possibilities for what a researcher might be interested in studying.

Possibilities include:

1) The grouping of the n sample observations into mutually-exclusive sets. This is essentially the definition of clustering.

2) Estimation of G, if it is (considered) unknown.

3) The estimation of π and $f_1, \cdots, f_g$.

4) Classifying a future observation X_{n+1} into one of the detected clusters. (Without repeating the whole clustering procedure on the $n+1$ observations).

We focus on defining indices of clustering performance for 1 and 4 above.

5.1.1 Classification

With a known or hypothesized model for the data, the grouping of the n sample observations into mutually-exclusive sets is usually reduced to the task of assigning labels $1, \cdots, G$ to each of the observations. Since in practice, the f_i's are unknown, there will be no clear 1:1 correspondence between population subgroups and label assignments identified in the sample. This has important implications regarding the assessment of clustering performance as we cannot compute a misclassification rate without knowing to which subgroups our assigned labels corresponds. Nonetheless, we can make use of some powerful results from the classification literature. In particular, when the f_i's are known (or hypothesized), the strategy which minimizes the misclassification rate is to calculate the posterior probabilities of each subgroup and assign a label, to whichever subgroup has the highest posterior probability (Seber, 2009, chap. 6). This strategy is the foundation for almost all of the model based clustering methods currently in use and we will refer to this method as the "optimal classifier". When all model parameters are known, the optimal classifier is the one that assigns an observation x_i to label k with the largest posterior probability, i.e. $\hat{g}_i = \hat{g}(x_i) = k$. if $P(A_i = k|x_i) > P(A_i = j|x_i), j \neq k$. We assume no ties, which is realistic for continuous variables. It should be noted that this classification procedure is distinct from a clustering procedure, since a clustering procedure does not necessarily provide meaningful labels. Clustering is defined as the formation of mutually-exclusive sets of observations, and the estimation of subgroup labels is not required to achieve this goal.

5.1.2 Clustering

We define a clustering as a partition of a dataset into mutually-exclusive groups, and we refer to each mutually-exclusive group as a cluster. It should be noted that any classification will induce a clustering, but clusterings do not imply classifications. For example in a model with 4 subgroups, the classification of $X_1, \cdots, X_5$, given by $\hat{\mathbf{A}} = (1, 1, 3, 4, 3)$, implies the clustering $\{\{X_1, X_2\}\{X_3, X_5\}\{X_4\}\}$. Yet, the converse does not hold as the clustering $\{\{X_1, X_2\}\{X_3, X_5\}\{X_4\}\}$ could have been generated from $4 * 3 * 2 = 24$ different sets of labels. It should be observed that since any set of labels induces a clustering the true labels also induce a true clustering. The partition P induced by $A_1, \cdots, A_n$ is obviously a random event, so the clustering of the n observations is a prediction problem. The aim is to predict P, not $A_1, \cdots, A_n$. This is a discrete problem of very high dimension. The number of possible partitions of a set of n elements grows very quickly with n. For $n = 5$, there are 52 possible partitions (the Bell number $B_5 = 52$) of a set of 5 items. Though the differences between clustering and classification are clear, we can re-frame the clustering problem into a classification problem. While initially we sought to assign labels $1, \cdots, G$ to each of the observations we could alternatively assign a label, which looks like a set theoretical partition, to the entire vector of observations. The only reason to conceptualize the problem in this way is to make use of the previously mentioned theory regarding the minimization of the misclassification rate. For $n = 5$, each of the 52 possible partitions has its own posterior probability and selecting the partition with the highest posterior probability will minimize the probability of misclassifying the clustering. We will refer to this as the "optimal clustering". For the case when G=2 there are always two label sets that would induce the same partition. For example with 5 variables $x1, ..., x5$ and cluster assignments $k1, ..., k5$ where $k \in [1, 2]$. The classifications $(1, 1, 1, 2, 2)$ and $(2, 2, 2, 1, 1)$ are the only two classifications that induce the partition $\{\{x1, x2, x3\}\{x4, x5\}\}$.

Let $\pi_{i,j}$ be the posterior probability that the ith data point, $i \in [1 : n]$, came from the jth component.

Table 5.1: The posterior probabilities for a counterexample to the claim that optimal clustering is equivalent to the clustering induced by the optimal classifier.

	G1	G2	G3
x1	0.4	0.3	0.3
x2	0.4	0.3	0.3

Then, in this example the probability of partition $\{\{x1, x2, x3\}\{x4, x5\}\}$ is given by

$$\pi_{1,1}\pi_{2,1}\pi_{3,1}\pi_{4,2}\pi_{5,2} + \pi_{1,2}\pi_{2,2}\pi_{3,2}\pi_{4,1}\pi_{5,1}$$

A proof that the optimal clustering is equivalent to the optimal classifier when $G = 2$ can be found in appendixB.2. However a simple counterexample demonstrates that this relationship does not necessarily hold for $G > 2$ Let G=3 and let the posterior probabilities be described in Table 5.1. Then the optimal classification is $(1, 1)$ which induces the partition $\{\{x1, x2\}\}$. However three different classifications could have created this partition so the probability of the clustering is given by $p(\{\{x1, x2\}\}) = .4*.4+.3*.3+.3*.3 = .34$. The only other possible partition is the singleton partition, $\{\{x1\}, \{x2\}\}$, which could have come from any of six clusters, resulting in a posterior partition probability $p(\{\{x1, x2\}\}) = .4*.3+.4*.3+.3*.4+.3*.3+.3*.4+.3*.3 = .66$. Thus, in general, we have shown that the clustering induced by the optimal classifier does not equal the optimal clustering. This suggests that researchers should give careful consideration to whether their primary objective is truly clustering or classification, or something else entirely. In any case, when $f_1, \cdots, f_g$ are unknown neither the optimal classifier or the optimal clustering can be computed. Furthermore, the misclassification rate, regardless of what is being classified, also remains beyond our reach. In the absence of a 1-1 correspondence between subgroups and clusters the only way to analyze the efficacy of a clustering algorithm is to study the comparative patterns of which variables were and were not linked together. This is typically done by comparing pairwise linkage.

5.2 The optimal linkage detector

First we will define optimal link detection and then we will discuss its relationship to the optimal classifier and to clustering.

Suppose that we try to predict linkage between X_i and $X_j, i \neq j$, defined as $L_{ij} := I(A_i = A_j)$, based on observed values x_i and x_j, and we denote the 0/1 predictor by $\hat{L}_{ij}$, implicitly a function of x_i, x_j and model parameters. Declaring linkage ($\hat{L}_{ij} = 1$) if the classifiers are equivalent, $\hat{g}_i = \hat{g}_i$, is a possible approach, but it may not be the best approach. Considering 0/1 to be a label we could once again utilize posterior probabilities to minimize the misclassification error. The optimal linkage detector is $\hat{L}_{ij} = 1$ if $P(A_i = A_j|x_i, x_j) \geq 0.5$, and $\hat{L}_{ij} = 0$ if $P(A_i = A_j|x_i, x_j) < 0.5$. That is, it assigns to $\hat{L}_{ij}$ the value with the largest posterior probability. Note that $P(A_i = A_j|x_i, x_j) = \sum_{g=1}^{G} P(A_i = A_j = g|x_i, x_j) = \sum_{g=1}^{G} P(A_i = g|x_i)P(A_j = g|x_j)$, since (X_i, A_i) is independent of (A_j, X_j) for $i \neq j$.

It is worth observing that every classification implies a clustering and every clustering implies a linkage matrix $\mathbf{L}$. This suggests that there might be some relationship between the optimal classification, the optimal clustering and the optimal linkage matrix. We have already shown that the optimal classification is not necessarily equivalent to the optimal clustering. We now show that the optimal linkage detector and the optimal classifier are not the same and produce different $\hat{L}_{ij}$, except in the case $G = 2$.

To see that the optimal linkage detector and the optimal classifier are not the same, consider the case $G = 3$, and suppose $P(A_i = 1|x_i) = P(A_j = 1|x_j) = 1/2$, $P(A_i = k|x_i) = P(A_j = k|x_j) = 1/4$ for $k = 2, 3$. Now, $P(A_i = A_j|x_i, x_j) = 3/8$ and the optimal detector assigns $\hat{L}_{ij} = 0$. However, the optimal classifier puts both observations in cluster 1 and hence assigns $\hat{L}_{ij} = 1$.

In the case $G = 2$, let $a = P(A_i = 1|x_i)$ and $b = P(A_j = 1|x_j)$. If both $a, b \geq 1/2$ then $\hat{g}_i = \hat{g}_j = 1$. If both $a, b < 1/2$ then $\hat{g}_i = \hat{g}_j = 2$. In both cases, $P(A_i = A_j|x_i, x_j) = ab + (1 - a)(1 - b) > 1/2$, and the optimal linkage detector will assign $\hat{L}_{ij} = 1$. It is also easy to verify that if $a < 1/2 < b$ or $b < 1/2 < a$, then $\hat{g}_i \neq \hat{g}_j$ and $\hat{L}_{ij} = 0$. Hence, it is only for $G = 2$

Table 5.2: The posterior probabilities for a counterexample to the claim that the optimal linkage matrix, when it forms a partition, is equivalent to the linkage matrix induced by the optimal clustering.

	G1	G2	G3
x1	0.40	0.31	0.29
x2	0.67	0.15	0.18
x3	0.15	0.70	0.15
x4	0.18	0.70	0.12
x5	0.75	0.13	0.12

that the optimal classifier is also the optimal linkage detector.

A classifier can be used for clustering since it creates a partition. A linkage detector by itself is not sufficient since the $\binom{n}{2}$ links, $\hat{L}_{ij}$, generally do not define a partition (consider $\hat{L}_{12} = 1, \hat{L}_{23} = 1, \hat{L}_{13} = 0$). In this setting another procedure would be needed to convert the linkage matrix into a partition. Furthermore, even when a linkage matrix does correspond to a partition, the optimal clustering still does not have to be equivalent to the optimal linkage matrix. For a counter example consider the case where $G = 3$, $n = 5$ and the posterior probability matrix is given in Table 5.2. In this case it can be seen, with a bit of help from a computer, that the linkage matrix induced by the optimal clustering is defined by 1001001100, where each number from left to right represents the strictly lower triangular entries of the matrix. However, the optimal linkage matrix is defined by 0000001100.

Thus we have established that clustering, classification, and linkage identification are three inherently different tasks, so even when the model is known and optimal procedures are available it might not be clear which strategy to use. On the other hand, it is conceivable that just being aware of these differences could help a researcher to determine an appropriate strategy for their particular application. For clustering, the distinction between pairwise linkage optimization and clustering optimization is particularly interesting because the standard methods for evaluating cluster performance are all based on pairwise linkage.

5.3 Clustering Indices

Substantial efforts have been made to assess the comparative strength of clustering algorithms by proposing indices that measure the similarity of two clusterings. Such indices take on a special meaning when one of the clusterings represents the true subgroups. Often clustering algorithms will be tested in simulations or other settings where the true subgroups are known and an index will be used to compare the results. The most commonly used is the Rand index (Rand, 1971) which is calculated as the number of pairs that were linked together in both clusterings, plus the pairs that were not grouped together in either clustering, all divided by the total number of possible pairs. The result of this computation is a number between zero and one, where a one implies perfect pairwise alignment between the clusterings and a zero represents complete disagreement. Intuitively this index seems to be capturing some form of agreement. However, the Rand index does not stand alone. In a paper by Albatineh et al. (2006) 28 different indices, all based on pairwise comparisons, for comparing two cluster assignments were identified. Although researchers have been using such indices since at least 1971 very few arguments exist to convincingly demonstrate why any one of these indices should be preferable to another. An effort made to evaluate these indices by Arbelaitz et al. (2013) attempted to find the top performing cluster index through extensive simulations. They analyzed 30 cluster indices under 6,480 configurations of clustering methods and data generation combinations. They decided to measure the success of a validity index by the frequency with which the index achieved its highest value, when the selected number of clusters coincided with the true number of groups from the simulations. Rather unsurprisingly, they found that the rank order of performance changed depending on the settings in a complicated multivariate way. The notion that optimal performance should occur when the number of specified clusters corresponds to the number of underlying subgroups is what we call the $K = G$ conjecture and it is something we will explore throughout the examples in this paper. Rather than assume that optimal performance will occur when $K = G$ we prefer to create a mathematical framework for the indices with a desirable probabilistic interpretation.

5.4 Clustering Sensitivity and Specificity

Here we define indices of clustering performance. For reasons previously discussed performance can't be based on knowing the true cluster status. The reason is not only that the true status is unknown, but also, the lack of 1:1 correspondence mentioned above. Performance measures should not require knowing G, since in practice G is either fixed at some value, or estimated from the data. We define indices that satisfy the above criteria. Consider two independent random vectors (X_i, A_i) and (X_j, A_j), and define $\hat{g}_i$ and $\hat{g}_j$ to be their respective cluster id assignments given by a clustering procedure. If the ith and jth observations belong to the same subgroup, $A_i = A_j$, we say that they are linked. If we view the clustering problem as a problem of detecting links between observations, we can use measures from the diagnostic testing literature to describe the performance of diagnostic tests. That is, we view a clustering procedure as a diagnostic test; the test for linkage between the ith and jth observations is positive if $\hat{g}_i = \hat{g}_j$ and negative if $\hat{g}_i \neq \hat{g}_j$. The true linkage status between the two observations is positive if $A_i = A_j$ and negative if $A_i \neq A_j$. Of course, in diagnostic testing studies, the true status is known, but in clustering applications it is unknown except in simulation experiments.

Now the proposed criteria are defined.

Clustering sensitivity (CSENS):

$$\kappa_1 := P(\hat{g}_i = \hat{g}_j | A_i = A_j).$$

Clustering specificity (CSPEC):

$$\kappa_2 := P(\hat{g}_i \neq \hat{g}_j | A_i \neq A_j).$$

Clustering positive predictive value (CPPV):

$$\kappa_3 := P(A_i = A_j | \hat{g}_i = \hat{g}_j).$$

Clustering negative predictive value (CNPV):

$$\kappa_4 := P(A_i \neq A_j | \hat{g}_i \neq \hat{g}_j).$$

The CSENS measures how well observations from the same cluster are clustered together by a second procedure. The CSPEC measures how well observations from different clusters are actually assigned to different clusters in a second procedure. CSENS and CSPEC measure the ability of the procedure to detect linkage and non-linkage. On the other hand, the CPPV and CNPV measure the strength of the detected linkage or non-linkage.

The indices $\{\kappa_j : j = 1, 2, 3, 4\}$ must be taken together as a measure of performance; one index alone is not sufficient. We can easily get CSENS = 1 by placing all observations into one cluster. But that would lead to CSPEC=0. This situation is analogous to sensitivity and specificity of diagnostic tests; we can't improve one without compromising the other.

A useful property of these indices is that they do not require the true number of clusters G to be known, and do not require the chosen number K, whether fixed or estimated, to be equal to G.

Another point to note is that these indices can (in principle) be computed for any clustering procedure and given model when all the parameters are known. In comparing various clustering methods we start with a statistical model for the problem of interest. Then we can assess the difficulty of the problem under various assumptions and parameter values. The indices can also be used to compare the performance of different clustering methods under one given model or under various models. We give two simple examples where the optimal index values can be evaluated theoretically. For more general models and arbitrary clustering algorithms, a numerical approximation can be obtained via simulation.

The optimal values, the indices that occur when using either the optimal clustering or the optimal classifier, are an upper bound on the precision of any clustering algorithm and hence measure the "difficulty" of the clustering problem. An example of a difficult clustering task is

a mixture of two very similar distributions, e.g. a 50:50 mixture of a $N(0,1)$ and a $N(10^{-6},1)$. Under this model, no clustering procedure is expected to do well with any realistic amount of data. On the other hand the task of clustering a 50:50 mixture of a N(0, 1) and a N(10, 1) is an easy problem. Closeness of the mixture components to each other and very small mixing probabilities both tend to make the problem more difficult. However, there are situations where the picture is not so clear, and where indices can help assess the problem. The proposed indices help in quantifying the difficulty level inherent in a given model with given parameter values.

The above examples show that the number of groupings that we wish to look at varies almost completely with the underlying population model. However in those examples each grouping was at least compact to some extent. We can also give examples where the optimal clustering does not involve compact sets. Thinking about the underlying structure of the data at a population level will clearly alter the interpretation of a clustering algorithm, but given an assumed or known structure we need to determine how well a clustering algorithm is working.

Now we give general expressions for the indices. Let $\pi_g = P(A_i = g), g = 1, \cdots, G$, and let π denote the column vector $(\pi_1, \cdots, \pi_G)^\top$. Of course, $\sum_{g=1}^{G} \pi_g = 1$. Next, we compute the probabilities of relevant events.

For two independent observations, say observations 1 and 2, $P(A_1 = A_2) = \sum_{g=1}^{G} \pi_g^2 = \pi^\top \pi = ssq(\pi)$, ssq denotes the sum of squares.

Define $b_{gj} = P(\hat{g}_i = j | A_i = g)$ for $g = 1, \cdots, G; j = 1, \cdots, k$. The b_{gj}'s are collected into the $G \times K$ matrix B.

The marginal distribution of $\hat{g}_i$ is given by

$$\begin{aligned} P(\hat{g}_1 = j) &= \sum_{g=1}^{G} P(A_1 = g) P(\hat{g}_1 = j | A_1 = g) \\ &= \sum_{g=1}^{G} \pi_g b_{gj} = (B^\top \pi)_j. \end{aligned}$$

Table 5.3: The 2×2 table of probabilities used to compute the clustering indices.

	$\hat{g}_1 \neq \hat{g}_2$	$\hat{g}_1 = \hat{g}_2$	Total
$A_1 \neq A_2$	γ_1	γ_2	$1 - ssq(\pi)$
$A_1 = A_2$	γ_3	$\gamma_4 = trace(C^\top C)$	$ssq(\pi)$
Total	$1 - ssq(B^\top \pi)$	$ssq(B^\top \pi)$	1

That is, the K-vector $B^\top \pi$ is the pmf of $\hat{g}_i$, and

$$P(\hat{g}_1 = \hat{g}_2) = \pi^\top B B^\top \pi = ssq(B^\top \pi).$$

The joint probability of true linkage and its detection in a sample is

$$\begin{aligned}
P(\hat{g}_1 = \hat{g}_2, A_1 = A_2) &= \sum_{g=1}^{G} \sum_{j=1}^{K} P(\hat{g}_1 = \hat{g}_2 = j, A_1 = A_2 = g) \\
&= \sum_{g=1}^{G} \sum_{j=1}^{K} P(A_1 = A_2 = g) P(\hat{g}_1 = \hat{g}_2 = j | A_1 = A_2 = g) \\
&= \sum_{g=1}^{G} \pi_g \sum_{j=1}^{K} b_{gj}^2 \\
&= trace(B^\top diag(\pi_g^2) B) \\
&= trace(C^\top C),
\end{aligned}$$

where $C = diag(\pi_g) B$.

The relevant probabilities can be displayed in a 2×2 table as shown in Table 5.3. Now we easily obtain

$$\begin{aligned}
\gamma_2 &= ssq(B^\top \pi) - trace(C^\top C), \\
\gamma_3 &= ssq(\pi) - trace(C^\top C), \\
\gamma_1 &= 1 - ssq(\pi) - ssq(B^\top \pi) + trace(C^\top C), \\
\kappa_1 &= \gamma_4 / (\gamma_3 + \gamma_4), \\
\kappa_2 &= \gamma_1 / (\gamma_1 + \gamma_2),
\end{aligned}$$

$$\begin{aligned}\kappa_3 &= \gamma_4/(\gamma_2+\gamma_4),\\ \kappa_4 &= \gamma_1/(\gamma_1+\gamma_3).\end{aligned}$$

5.5 Examples

Example 1:

A (π_1, π_2) mixture of a uniform(0,1) and a uniform$(\delta, \delta+1)$, with $\delta \in (0,1)$. By symmetry, it suffices to consider $\pi_1 \in (0, 0.5]$. The optimal classifier is $\hat{g}(x_i) = 1$ if $x_i \in (0, \delta)$ and $\hat{g}(x_i) = 2$ otherwise. The indices are computed as follows.

$$\begin{aligned}B &= \begin{bmatrix} \delta & 1-\delta \\ 0 & 1 \end{bmatrix},\\ ssq(B^\top \pi) &= 1 - 2\pi_1\delta(1-\pi_1\delta),\\ \pi^\top\pi &= ssq(\pi) = 1 - 2\pi_1(1-\pi_1),\\ trace(C^\top C) &= \pi_1^2\{1-2\delta(1-\delta)\} + \pi_2^2,\end{aligned}$$

and the indices are

$$\begin{aligned}\kappa_1 &= \frac{\pi_1^2\{1-2\delta(1-\delta)\}+\pi_2^2}{1-2\pi_1(1-\pi_1)},\\ \kappa_2 &= \delta,\\ \kappa_3 &= \frac{\pi_1^2\{1-2\delta(1-\delta)\}+\pi_2^2}{1-2\pi_1\delta(1-\pi_1\delta)},\\ \kappa_4 &= \frac{1-\pi_1}{1-\pi_1\delta}.\end{aligned}$$

Example: $\pi_1 = 0.3, \delta = 0.5$ gives $\kappa_1 = 0.922, \kappa_2 = 0.5, \kappa_3 = 0.718, \kappa_4 = 0.824$.

Tables demonstrating the relationship between the indices and changes to π_1 and δ can be found in Appendix C.2. For this example, sensitivity decreases as both π_1 and δ increase, while

specificity appears to only be a function of δ.

Example 2:

Consider a mixture of two normals; a $\mathrm{N}(0,1)$ with probability π_1 and a $\mathrm{N}(\mu,\sigma^2)$ with probability $\pi_2 = 1-\pi_1$. The parameters are: (π_1, μ, σ^2). Since a linear transformation leaves the problem essentially unchanged, it suffices to consider $\mu > 0, \sigma^2 \geq 1$ and $\pi_1 \in (0,1)$.

Now we derive the optimal classifier, the one which assigns $\hat{g}_i = \hat{g}(x_i) = 1$ if $\pi_1 f_1(x_i) > \pi_2 f_2(x_i)$, and $\hat{g}_i = 2$ otherwise.

We will use $\phi(.,a,b)$ to denote the pdf of the normal distribution with mean a and variance b, and $\Phi()$ to denote the cdf of the standard normal distribution. We start with the case of $\sigma^2 = 1$. Here, $\hat{g}(x) = 1$ if $\mu x < \log(\pi_1/\pi_2) + \mu^2/2$, which (for $\mu > 0$) gives

$$b_{11} = \Phi\left(\frac{1}{\mu}\log\frac{\pi_1}{\pi_2} + \frac{\mu}{2}\right), \qquad b_{21} = \Phi\left(\frac{1}{\mu}\log\frac{\pi_1}{\pi_2} - \frac{\mu}{2}\right).$$

That is, x is assigned to cluster 1 if it is below $\frac{\mu}{2} + \frac{1}{\mu}\log\frac{\pi_1}{\pi_2}$ and to cluster 2 otherwise. The optimal classifier divides the real line into two disjoint sets.

Now we deal with the case $\sigma^2 > 1$. The ratio

$$\frac{f_1(x)}{f_2(x)} = \sigma \exp\left(\frac{\mu^2}{2(\sigma^2-1)}\right)\sqrt{2\pi\tau^2}\phi\left(x,\theta,\tau^2\right)$$

has a maximum of

$$M := \sigma \exp\left(\frac{\mu^2}{2(\sigma^2-1)}\right).$$

In the above,

$$\theta := \frac{-\mu}{\sigma^2-1}, \quad \tau^2 := \frac{\sigma^2}{\sigma^2-1}.$$

Note that $M > 1$ since $\sigma^2 > 1$. So if $M < \pi_2/\pi_1$ then $\pi_1 f_1(x) < \pi_2 f_2(x)$ and $\hat{g}(x) = 2$ for all x. This can arise only if $\pi_2 > 1/2$. This is an interesting case in which the optimal procedure assigns all observations to a single cluster, even though the model with two clusters and all its parameters are completely known. It shows that accurate estimation of G is not a requirement for

optimal performance of a clustering procedure. A numerical example shows that this is possible: $\pi_1 = 0.25, \mu = 1.5, \sigma^2 = 4, M = 2.91 < \pi_2/\pi_1 = 3$.

This model with any combination of parameter values that leads to $M < \pi_2/\pi_1$ has an optimal procedure with $\kappa_1 = 1, \kappa_2 = 0$, $\kappa_3 = ssq(\pi)$ and κ_4 is undefined. The sum $\kappa_1 + \kappa_2$ is 1, which is exactly what is expected from a diagnostic test that is of no value. Hence, such parameter values define what is, in a sense, a most-difficult problem. We emphasize that this applies to only a subset of the parameter space.

Another example is: $\pi_1 = 0.15, \mu = 3, \sigma^2 = 16, M = 5.40 < \pi_2/\pi_1 = 5.67$.

Another example is: $\pi_1 = 0.35, \mu = 0.1, \sigma^2 = 1.01, M = 1.66 < \pi_2/\pi_1 = 1.86$.

Another example is: $\pi_1 = 0.2, \mu = 1, \sigma^2 = 1.5, M = \sqrt{1.5}e \approx 3.329 < \pi_2/\pi_1 = 4$.

If $M \geq \pi_2/\pi_1$ then $\hat{g}(x) = 1$ if

$$|x - \theta| < \sqrt{2\tau^2 \log \frac{M\pi_1}{\pi_2}},$$

Otherwise, we assign $\hat{g}(x) = 2$. That is, x values within $\sqrt{2\tau^2 \log \frac{M\pi_1}{\pi_2}}$ from θ are assigned to cluster 1. More extreme values, above and below θ are assigned to cluster 2. Hence the region assigned to cluster 2 is a union of two disjoint sets. This shows that the notion that observations close together should be placed in the same cluster is generally false. This also highlights the need to define precisely what is meant by a "cluster", and to not confuse true population subgroups with estimated sample clusters. Here we define the subgroup of x_i to be the mixture component that generated it, not what "region" the observation is in. Indeed an observation from cluster 1 may fall inside the region assigned by the optimal classifier to cluster 2. This may be a subtle point, but it bears keeping in mind.

5.5.1 Computing the indices in other models and clustering procedures

The only practical way to compute the indices for more complex models and clustering procedures is to use simulation. Briefly, we simulate a dataset of n iid observations, apply any

given clustering algorithm, compute estimates of the indices, repeat a large number of times and average the estimates.

From each sample we obtain a cross tabulation of the true cluster id versus the assigned cluster id. This will be a $G \times K$ table with entries n_{ij} and total $n_{\cdot\cdot} = n$. From this table we obtain a 2x2 table that tabulates the true linkage status versus the assigned linkage status.

For each combination, we compute $\kappa_j, j = 1, 2, 3, 4$.

We examine K-means clustering (MacQueen, 1967), Hierarchical Clustering with Average linkage (Tibshirani et al., 2004, ch. 14) and Model Based Clustering (McLachlan and Basford, 1987) under two sets of parameters. All clustering algorithms were implemented in R 2.15.1. K-means and Hierarchical Clustering were computed using the base functions kmeans() and hclust() respectively. Model based clustering was performed with the function normalmixEM() in the 'mixtools' package. In all the settings that we will test these algorithms, the true model is a mixture of two Gaussian random variables where the first component is a $N(0, 1)$ random variable. Results for a $N(0, 1)$ is mixed evenly with a $N(3, 1)$ are given in Table 5.4.

Table 5.4: The clustering indices are computed for Kmeans, Hierarchical Clustering with Average Linkage, and model based clustering with different specifications of K the number of clusters. The true model is a Gaussian mixture model with a $N(0, 1)$ is mixed evenly with a $N(3, 1)$.

	K=2	K=3	K=5	K=10
CSENS:Kmeans	0.872	0.584	0.378	0.200
CSENS:HC	0.865	0.729	0.48	0.234
CSENS:MB	0.848	0.815	0.82	0.854
CSPEC:Kmeans	0.871	0.900	0.948	0.977
CSPEC:HC	0.754	0.863	0.928	0.972
CSPEC:MB	0.756	0.443	0.304	0.228
CSENS+SPEC:Kmeans	1.74	1.48	1.33	1.18
CSENS+CSPEC:HC	1.62	1.59	1.41	1.21
CSENS+CSPEC:MB	1.6	1.26	1.12	1.08
CPPV:Kmeans	0.871	0.854	0.879	0.899
CPPV:HC	0.779	0.842	0.87	0.894
CPPV:MB	0.776	0.594	0.541	0.525
CNPV	0.872	0.684	0.604	0.55
CNPV:HC	0.848	0.761	0.641	0.559
CNPV:MB	0.832	0.705	0.628	0.611

In this setting all of the clustering algorithms perform the best, in terms of sensitivity plus

specificity, when the number of clusters is correctly specified. However, it is worth noting that we are looking at only a univariate setting where the components are two standard deviations apart and even correctly setting $K = G$ sensitivity can be as low as .85 and specificity as low as .75. With other clustering indices, we can only use these values for comparative purposes. However, with our indices we can take advantage of the probabilistic interpretation to observe that 15% of the time pairs that should be linked together will not be and 25% of the time pairs that should not be linked together will be. This means that with a model where statisticians usually expect to be able to easily detect group differences, with a decent sample size, the problem of assigning group labels fails fairly often. We will now consider a more difficult setting. We set the second component equal to a $N(1.5, 4)$ mixed in at a proportion of $\pi_2 = .75$. These are the parameter settings described above that yield an optimal classifier when every sample is assigned to the same cluster. Unsurprisingly, Table 5.5 none of the methods do well in this configuration. From the standpoint of correct classification, sensitivity plus specificity is equal to one when simply guessing at random.

Table 5.5: The clustering indices are computed for Kmeans, Hierarchical Clustering with Average Linkage, and model based clustering with different specifications of K the number of clusters. The true model is a Gaussian mixture model where 25% comes from a $N(0, 1)$ and the rest comes from a $N(1.5, 4)$.

	K=2	K=3	K=5	K=10
CSENS:Kmeans	0.54	0.379	0.245	0.121
CSENS:HC	0.788	0.582	0.345	0.16
CSENS:MB	0.719	0.746	0.781	0.82
CSPEC:Kmeans	0.509	0.659	0.787	9×10^{-1}
CSPEC:HC	0.187	0.403	0.669	0.851
CSPEC:MB	0.299	0.289	0.272	0.229
CSENS+SPEC:Kmeans	1.05	1.04	1.03	1.02
CSENS+CSPEC:HC	0.975	0.985	1.01	1.01
CSENS+CSPEC:MB	1.02	1.03	1.05	1.05
CPPV:Kmeans	0.647	0.649	0.657	0.669
CPPV:HC	0.618	0.619	0.635	0.642
CPPV:MB	0.631	0.636	0.641	0.639
CNPV	0.399	0.389	0.385	0.381
CNPV:HC	0.346	0.366	0.38	0.378
CNPV:MB	0.39	0.406	0.427	0.433

The next simulation is from a mixture model with 5 equally proportioned components. Vari-

ance of each component is one and the means are evenly spaced between 2 and 10. The results shown in Table 5.6 show that, in terms of sensitivity plus specificity, the optimal classifier provides the best performance, all methods perform the best when $K = G$, and even the best performance still yields a 20% chance of failing to group pairs together that should be and a 5% chance of grouping together pairs that should not be together.

Table 5.6: The clustering indices are computed for Kmeans, Hierarchical Clustering with Average Linkage, model based clustering, and the optimal classifier (when applicable), with different specifications of K the number of clusters. The true model is a 5 component Gaussian mixture model, with each component evenly proportioned and two standard deviations apart.

	K=2	K=3	K=5	K=7	K=10
CSENS:Kmeans	0.911	0.845	0.774	0.595	.439
CSENS:HC	0.944	0.885	0.777	0.646	.485
CSENS:MB	0.941	0.871	0.762	0.672	.548
CSENS:Opt			0.808		
CSPEC:Kmeans	0.601	0.791	0.94	0.96	.974
CSPEC:HC	0.585	0.769	0.921	0.951	.971
CSPEC:MB	0.507	0.69	0.832	0.903	.941
CSPEC:Opt			0.952		
CSENS+SPEC:Kmeans	1.51	1.64	1.71	1.55	1.41
CSENS+CSPEC:HC	1.53	1.65	1.7	1.6	1.46
CSENS+CSPEC:MB	1.45	1.56	1.59	1.58	1.49
CSENS+CSPEC:Opt			1.76		
CPPV:Kmeans	0.363	0.503	0.762	0.788	.809
CPPV:HC	0.362	0.49	0.712	0.766	.805
CPPV:MB	0.323	0.413	0.531	0.634	.699
CPPV:Opt			0.808		
CNPV:Kmeans	0.964	0.953	0.943	0.905	.874
CNPV:HC	0.977	0.964	0.943	0.915	.883
CNPV:MB	0.972	0.955	0.933	0.917	.893
CNPV:Opt			0.952		
CPPV+CNPV:Kmeans	1.33	1.46	1.71	1.69	1.68
CPPV+CNPV:HC	1.34	1.45	1.65	1.68	1.69
CPPV+CNPV:MB	1.29	1.37	1.46	1.55	1.59
CPPV+CNPV:Opt			1.76		

All of these examples are of mostly academic interest to the researchers who use clustering algorithms since univariate clustering very rarely takes place. We now calculate the indices on a pair of real data sets. The first dataset contains gene expression data for 5565 genes from 103 samples. These samples are made up of 26 breast cancer, 23 colon cancer, 28 lung cancer and 26

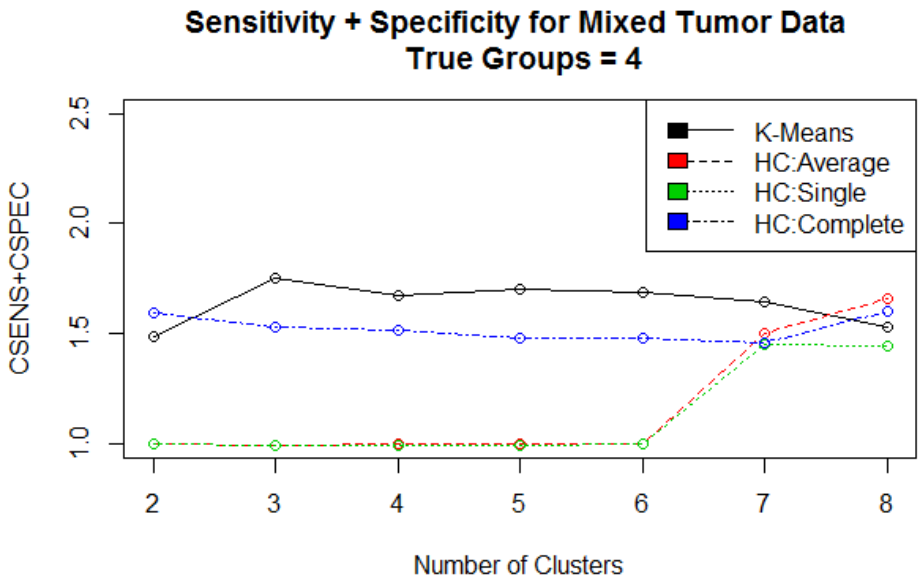

Figure 5.1: Cluster Sensitivity plus Cluster Specificity for the mixed tumor dataset.

prostate cancer samples. The data collection is described in a paper by (Hoshida et al., 2007) and the data can be obtained at http://www.broadinstitute.org/cgi-bin/cancer/datasets.cgi . This should be representative of a relatively easy clustering problem since the samples being compared are from completely distinct cancer types. To clarify, we expect this problem to be easy relative to the task of discovering tumor subtypes. The complete tables for the indices can be found in Appendix C.2. Figure 5.1 shows sensitivity plus specificity for various methods and specifications of K.

There are three obvious lessons in this plot. First, the performance bump that we saw when $K = G$ in the univariate setting does not appear to exist with this real data as none of the methods show a convincing improvement in performance at the correct number of subgroups. The second clear message is that the selection of the linkage function in a hierarchical clustering algorithm can have a substantial impact on the algorithms performance. Finally the sudden jump in performance at $K = 7$, for HC with Average and Single Linkage, is worth examining. The distribution of clustering assignments for $K = 6, 7$ and 8 are shown in Figure 5.7, Figure 5.8 and Figure 5.9 respectively. These tables suggest that in the high dimensional setting of real data,

Table 5.7: Contingency table of cluster assignments for the Mixed Tumor Data when $K = 6$

Pathology/Cluster	1	2	3	4	5	6
Breast	23	1	1	1	0	0
Colon	23	0	0	0	0	0
Lung	22	0	0	0	1	5
Prostate	25	1	0	0	0	0

Table 5.8: Contingency table of cluster assignments for the Mixed Tumor Data when $K = 7$

Pathology/Cluster	1	2	3	4	5	6	7
Breast	21	1	1	2	1	0	0
Colon	0	0	0	23	0	0	0
Lung	22	0	0	0	0	1	5
Prostate	0	1	0	25	0	0	0

Hierarchical clustering methods are very sensitive to outliers, often grouping them into distinct clusters. For this reason we see a pathological behavior where Hierarchical Clustering performs best when K is grossly misspecified, forcing the outliers into larger groups.

We now turn to the more difficult problem of using clustering algorithms to identify subtypes of lung cancer. The next dataset is described as dataset A in a paper by Bhattacharjee et al. (2001), and it was also analyzed in a paper by Thalamuthu et al. (2006). The data can be found at http://www.pnas.org/content/98/24/13790/suppl/DC1 The data contains 203 lung samples, (127 adenocarcinomas, 21 squamous cell carcinomas, 20 pulmonary carcinoids, 6 small cell lung cancers and 17 normal lung specimens) categorized by histology. The full table of clustering indices are provided in Appendix C.2. As expected, Figure 5.2, shows an overall drop in performance relative to the mixed tumor example. We also once again see that the selection of a linkage function has a powerful impact on algorithm performance and once again no boost in performance is observed when $K = G$. It should also be noted that the order of algorithmic performance has changed. When analyzing the mixed tumor data, Kmeans was clearly the best algorithm while

Table 5.9: Contingency table of cluster assignments for the Mixed Tumor Data when $K = 8$

Pathology/Cluster	1	2	3	4	5	6	7	8
Breast	21	1	1	2	1	0	0	0
Colon	0	0	0	23	0	0	0	0
Lung	22	0	0	0	0	0	1	5
Prostate	0	1	0	0	0	25	0	0

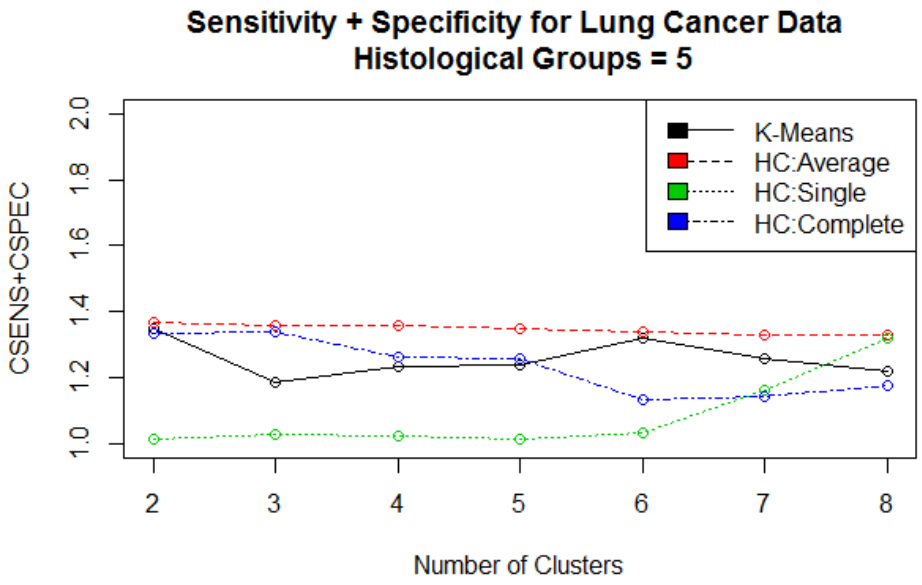

Figure 5.2: Cluster Sensitivity plus Cluster Specificity for the Lung Cancer dataset.

hierarchical clustering with single and average linkage performed terribly. For the lung cancer data, average linkage hierarchical clustering demonstrated the best performance at all values of K. Unfortunately, we can offer no theory that would allow a researcher to anticipate the relative performance of these methods a priori.

5.6 Discussion

We have demonstrated the differences between clustering, classification and linkage detection. We have introduced new indices of clustering performance, which put the problem of clustering validation into the well understood framework of sensitivity and specificity. Our theoretical analysis along with the examination of index performance on real and simulated data has provided a number of important lessons for researchers interested in both using and developing clustering algorithms. For researchers who seek to improve clustering algorithms it is important to realize the differences between clustering classification and linkage detection. Many iterative clustering algorithms that utilize optimal classification rules could conceivably be improved by instead im-

plementing optimal clustering rules. For users of clustering algorithms the use of clustering indices with clear probabilistic interpretations can be used to improve the interpretation of clustering results. For example, we showed that even in some relatively simple settings pairs of samples were being grouped together when they should not be about 25% of the time. This suggests that, in the biomedical field, it would be unwise to think of clustering labels as some sort of newly discovered 'true' classification. This limitation can be seen even in the absence of problems with algorithm selection, variable reduction and estimation of G, all of which undoubtedly make clustering results more difficult to interpret. Researchers should be aware that the selection of different clustering algorithms can have powerful impacts on the results observed. Furthermore, we have seen that optimal algorithm performance can theoretically occur when $K \neq G$ and in practice performance may not improve the slightest bit when $K = G$. Thus, validation methods that rely on this principle are likely to be misguided. Our indices, and all of the others we are familiar with, are based on pairwise linkage and we have shown that optimal pairwise linkage does not necessarily correspond to optimal clustering. This suggests that there could be circumstances under which indices based on pairwise comparisons would not be ideal. Nonetheless, for most purposes we have proposed an effective way to not only compare clustering methods but also to improve the way researchers interpret clustering results. Testing an algorithm on similar supervised settings with our indices can help researchers find an appropriate level of confidence to ascribe the results of their clustering algorithms. For the settings that we examined, it would be foolish to think of cluster assignments as an approximation for subtype discovery. The reality is that different algorithms yield vastly different partitions and even in the best case scenarios elements from different subtypes will frequently be grouped with elements from different components and vice versa. This is not to say that the clustering algorithms are useless. We are grouping together elements according to some sort of defined distance metric which could conceivably be valuable for many different reasons. We simply observe that it is easy to construct examples where important subtypes exist and clustering algorithms should not be expected to discover those subtypes with any sort of reliability. It is our hope that model based approach that we have provided, along with our probabilistic indices, will help researchers better understand and interpret the results of clustering algorithms on their data.

CHAPTER 6: FUTURE WORK

Mass spectrometry proteomics experiments pose interesting and unique problems for statistical modeling. In this thesis, progress has been made by establishing formal models that account for missing data biases which are for the most part currently being ignored by the scientific community. The most obvious need for further work would be the development software tools to implement these methods. To this regard some theoretical work will need to be done to obtain model estimates in a practical amount of time. In all of our work we have used small sets of data to establish principles but in the real world scientists are working with ever larger datasets. A lot of work would need to be done to make the Bayesian solutions proposed feasible for real world studies. In addition to these pragmatic developments, much work still needs to be done to explore the complexities posed by the general discovery mass spectrometry experiment. We have established new ways to categorize missing data, but it remains unclear which levels of missing data under what circumstances can actually be useful. There is also work that needs to be done regarding reference selection in large experiments. It is not known a priori which contrasts in these experiments will be estimable. In the Bayesian model this will be reflected in the variance, however in order for this to be discovered contrasts need to be taken directly from the Gibbs samples. Attempting to combine summary statistics will not work. In addition to the difficulties posed by large complex experiments it would be natural to combine the work done on clustering evaluation and proteomics modeling. It would be very surprising if the missing data biases and relative nature of mass spectrometry data didn't have unusual effects on clustering algorithms. The nature of these effects has yet to be explored.

APPENDIX A: CHAPTER 3 DETAILS

A.1 Deriving the full conditionals for the M5 model

$$
\begin{aligned}
f_{(Y_{ijk}|\cdot)}(y_n) &= \frac{\prod_{k=n_1+1}^{N}(1-\Phi(a+by_k))\prod_{k=1}^{n_1}\Phi(a+by_k)f_{\mathbf{Y_i}}}{\int\prod_{k=n_1+1}^{N}(1-\Phi(a+by_k))\prod_{k=1}^{n_1}\Phi(a+by_k)f_{\mathbf{Y_i}}dy} \\
&= \frac{(1-\Phi(a+by_n))f_{\mathbf{Y_i}}}{\int(1-\Phi(a+by_n))f_{\mathbf{Y_i}}dy} \\
&= \frac{\Phi(-a-by_n)\exp(-\frac{1}{2\sigma}(y_n-\alpha_{n(i)}\pm\frac{\mu_i}{2})^2)}{\int\Phi(a-by_n)\exp(-\frac{1}{2\sigma}(y_n-\alpha_{n(i)}\pm\frac{\mu_i}{2})^2)dy} \\
&\propto \Phi(-a-by_n)\exp(-\frac{1}{2\sigma}(y_n-\alpha_{n(i)}\pm\frac{\mu_i}{2})^2)
\end{aligned}
$$

Which is the kernel of an extended skew normal distribution defined as

$$
f_{skew}(x) = \frac{\phi(\frac{x-\mu_x}{\sigma_x})\Phi(\omega\sqrt{1+c^2}+c(\frac{x-\mu_x}{\sigma_x}))}{\sigma_x\Phi(\omega)}
$$

where

$$
\mu_x = \alpha_{n(i)} \pm \frac{\mu_i}{2}, \quad \sigma_x = \sqrt{\sigma}
$$

and

$$
\Phi(-a-by_n) = \Phi(-a-\frac{\sigma_x}{\sigma_x}b(y_n-\mu_x+\mu_x)) = \Phi(-a-\sigma_x b(\frac{y_n-\mu_x}{\sigma_x})-b\mu_x)
$$

Thus,

$$
-b\sigma_x = c, \quad \omega\sqrt{1+c^2} = -a-b\mu_x
$$

$$
\Rightarrow \omega = \frac{-a-b\mu_x}{\sqrt{1+\sigma b^2}}
$$

Therefore,

$$f_{(Y_{ijk}|\mu_i,\alpha_{j(i)},Y_{ijk'},\theta,\mathbf{R})}(x) = \frac{\phi(\frac{x-\mu_x}{\sqrt{\sigma}})\Phi(\omega\sqrt{1+(-b\sqrt{\sigma})^2} - b\sqrt{\sigma}(\frac{x-\mu_x}{\sqrt{\sigma}}))}{\sqrt{\sigma}\Phi(\omega)}$$
$$= \frac{\phi(\frac{x-\mu_x}{\sqrt{\sigma}})\Phi(-a-bx)}{\sqrt{\sigma}\Phi(\omega)}.$$

Similarly

$$f_{(\mu_i|\cdot)} \propto f_{\mathbf{Y}|\mu_i,\alpha} f_{\mu_i}$$
$$\propto \exp(-\frac{1}{2\tau}(\mu_i-\beta_\mu)^2)\prod_{j=1}^{m_i}\exp\left(-\frac{1}{2\sigma}\left((y_{ij1}-\alpha_{j(i)}-\frac{\mu_i}{2})^2+(y_{ij2}-\alpha_{j(i)}+\frac{\mu_i}{2})^2\right)\right)$$
$$= \exp\left(-\frac{1}{2\tau}(\mu_i-\beta_\mu)^2+\sum_{j=1}^{m_i}-\frac{1}{2\sigma}\left((y_{ij1}-\alpha_{j(i)}-\frac{\mu_i}{2})^2+(y_{ij2}-\alpha_{j(i)}+\frac{\mu_i}{2})^2\right)\right)$$
$$= \exp\left(-\frac{1}{2}\left(\frac{1}{\tau}\mu_i^2-\frac{2\beta_\mu}{\tau}\mu_i+\frac{1}{\sigma}\sum_{j=1}^{m_i}\frac{1}{2}\mu_i^2-\mu_i(y_{ij1}-\alpha_{j(i)})+\mu_i(y_{ij2}-\alpha_{j(i)})\right)+C\right)$$

for some constant C.

$$\propto \exp\left(-\frac{1}{2}\left(\mu_i^2(\frac{1}{\tau}+\frac{m_i}{2\sigma})-\mu_i\left(\frac{2\beta_\mu}{\tau}+\frac{1}{\sigma}\sum(y_{ij1}-y_{ij2})\right)\right)\right)$$
$$= \exp\left(-\frac{1}{2(\frac{2\sigma\tau}{2\sigma+m_i\tau})}\left(\mu_i^2-\mu_i\frac{2\sigma\tau}{2\sigma+m_i\tau}\left(\frac{2\beta_\mu}{\tau}+\frac{1}{\sigma}\sum(y_{ij1}-y_{ij2})\right)\right)\right)$$
$$= \exp\left(-\frac{1}{2(\frac{2\sigma\tau}{2\sigma+m_i\tau})}\left(\mu_i^2-\mu_i(\frac{2\beta_\mu\sigma+\tau\sum(y_{ij1}-y_{ij2})}{\sigma+\frac{m_i\tau}{2}})\right)\right)$$
$$\propto \exp\left(-\frac{1}{2(\frac{2\sigma\tau}{2\sigma+m_i\tau})}\left(\mu_i-(\frac{\beta_\mu\sigma+\frac{\tau}{2}\sum(y_{ij1}-y_{ij2})}{\sigma+\frac{m_i\tau}{2}})\right)^2\right)$$

Therefore,

$$(\mu_i|\alpha_{\mathbf{i}}, \mathbf{Y}, \theta, \mathbf{R}) \sim N\left(\frac{\beta_\mu \sigma + \frac{\tau}{2}\sum_j (y_{ij1} - y_{ij2})}{\sigma + \frac{m_i \tau}{2}}, \frac{\sigma\tau}{\sigma + \frac{m_i\tau}{2}}\right).$$

The distribution for α given everything else can be derived in a similar manner.

$$\begin{aligned}
f_{(\alpha_{j(i)}|\cdot)} &\propto f_{\mathbf{Y}|\mu_i,\alpha} f_{\alpha_{j(i)}} \\
&= \exp(-\frac{1}{2\xi}(\mu_i - \beta_\alpha)^2) \exp\left(-\frac{1}{2\sigma}\left((y_{i,j,1} - \alpha_{j(i)} - \frac{\mu_i}{2})^2 + (y_{ij2} - \alpha_{j(i)} + \frac{\mu_i}{2})^2\right)\right) \\
&= \exp(-\frac{1}{2}(\frac{1}{\xi}(\alpha_{j(i)}^2 - 2\beta_\alpha \alpha_{j(i)}) + \\
&\frac{1}{\sigma}\left(2\alpha_{j(i)}^2 - 2\alpha_{j(i)}(y_{ij1} - \frac{\mu_i}{2}) - 2\alpha_{j(i)}(y_{ij2} + \frac{\mu_i}{2})\right)) + C)
\end{aligned}$$

for some constant C

$$\begin{aligned}
&\propto \exp(-\frac{1}{2}(\frac{1}{\xi}(\alpha_{j(i)}^2 - 2\beta_\alpha \alpha_{j(i)}) + \frac{2}{\sigma}\left(\alpha_{j(i)}^2 - \alpha_{j(i)}(y_{ij1} + y_{ij2})\right))) \\
&= \exp(-\frac{1}{2}(\alpha_{j(i)}^2(\frac{1}{\xi} + \frac{2}{\sigma}) - \alpha_{j(i)}(\frac{2\beta_\alpha}{\xi} + 2\frac{y_{ij1} + y_{ij2}}{\sigma}))) \\
&= \exp(-\frac{1}{2\frac{\xi\sigma}{\sigma+2\xi}}(\alpha_{j(i)}^2 - \alpha_{j(i)}\frac{\xi\sigma}{\sigma+2\xi}(\frac{2\beta_\alpha}{\xi} + 2\frac{y_{ij1} + y_{ij2}}{\sigma}))) \\
&= \exp(-\frac{1}{2\frac{\xi\sigma}{\sigma+2\xi}}(\alpha_{j(i)}^2 - \alpha_{j(i)}(\frac{2\beta_\alpha\sigma + 2\xi(y_{ij1} + y_{ij2})}{\sigma+2\xi}))) \\
&= \exp(-\frac{1}{2\frac{\xi\sigma}{\sigma+2\xi}}(\alpha_{j(i)} - (\frac{\beta_\alpha\sigma + \xi(y_{ij1} + y_{ij2})}{\sigma+2\xi}))^2)
\end{aligned}$$

Therefore,

$$(\alpha_{j(i)}|\cdot) \sim N\left(\frac{\beta_\alpha\sigma + \xi(y_{ij1} + y_{ij2})}{\sigma + 2\xi}, \frac{\xi\sigma}{\sigma+2\xi}\right)$$

APPENDIX B: CHAPTER 4 DETAILS

B.1 Deriving a general full conditional distribution for the normal parameters

Each of the parameters relating to the mean model have a similar structure and the full conditional distribution is given by one generalized formula.

Let α represent any of the mean parameters with a Gaussian prior. Let β_α, τ be the mean and variance of α and let $\mathbf{X}\theta^*$ be the product of the design matrix and parameter vector with α removed from θ and the column $\mathbf{X}_{[\cdot,\alpha]}$ removed from $\mathbf{X}$. Here the subscript $[\cdot, \alpha]$ is used to reference a sub-matrix of X. α indicates a single column, the one corresponding to the α parameter, and the $\cdot$ indicates that all of the rows are included. Finally, let $i, \cdots, I$ represent the row indices for which $\mathbf{X}_{[\cdot,\alpha]} = 1$. Then

$$
\begin{aligned}
f_{(\alpha|\cdot)} &\propto f_{(\mathbf{Y}|\cdot)} f_\alpha \\
&\propto \exp\left(-\frac{1}{2\tau}(\alpha - \beta_\alpha)^2 - \frac{1}{2\sigma}\left((y_i - \alpha - \mathbf{X}\theta^*_{[i]})^2\right) - \cdots - \frac{1}{2\sigma}(y_I - \alpha - \mathbf{X}\theta^*_{[I]})^2\right) \\
&\propto \exp\left(-\frac{1}{2\tau}(\alpha^2 - 2\alpha\beta_\alpha) - \frac{1}{2\sigma}\sum_{j=i}^{I}\left(\alpha^2 - 2\alpha(y_j - \mathbf{X}\theta^*_{[j]})\right)\right) \\
&= \exp\left(-\frac{1}{2}\left(\alpha^2\left(\frac{1}{\tau} + \frac{I}{\sigma}\right) - \alpha\left(\frac{2\beta_\alpha}{\tau} + \frac{2\sum_{j=i}^{I}(y_j - \mathbf{X}\theta^*_{[j]})}{\sigma}\right)\right)\right) \\
&= \exp\left(-\frac{1}{2\tau\sigma/(\sigma + \tau I)}\left(\alpha^2 - \frac{2\alpha}{\sigma + \tau I}\left(\sigma\beta_\alpha + \tau\sum_{j=i}^{I}(y_j - \mathbf{X}\theta^*_{[j]})\right)\right)\right) \\
&\propto \exp\left(-\frac{1}{2\tau\sigma/(\sigma + \tau I)}\left(\alpha - \frac{\sigma\beta_\alpha + \tau\sum_{j=i}^{I}(y_j - \mathbf{X}\theta^*_{[j]})}{\sigma + \tau I}\right)^2\right)
\end{aligned}
$$

Therefore,

$$(\alpha|\cdot) \sim N\left(\frac{\sigma\beta_\alpha + \tau\sum_{j=i}^{I}(y_j - \mathbf{X}\theta^*_{[j]})}{\sigma + \tau I}, \frac{\tau\sigma}{\sigma + \tau I}\right)$$

B.2 Deriving the full conditional distribution for a level 2 missing value in an iTRAQ experiment

In a K-plex iTRAQ experiment we expect that either a peptide will be observed in K samples or none for a given run. If Y_m is a missing intensity then Let $Y_i, ..., Y_n$ represent the intensities from the same run for the same peptide as Y_m. Then,

$$\begin{aligned} f_{(Y_m|\cdot)}(y_m) &\propto (1 - \Phi\left(a + b(y_m + \sum_{j=i}^{n} y_j)\right)\exp(-\frac{1}{2\sigma}(y_m - \mathbf{X}\theta_{[m]})^2) \\ &= \Phi\left(y_m(-b) - a - b\sum_{j=i}^{n} y_j\right)\exp(-\frac{1}{2\sigma}(y_m - \mathbf{X}\theta_{[m]})^2) \end{aligned}$$

Which is the kernel of an extended skew normal distribution defined as

$$f_{skew}(x) = \frac{\phi(\frac{x-\mu_x}{\sigma_x})\Phi(\omega\sqrt{1+c^2} + c(\frac{x-\mu_x}{\sigma_x}))}{\sigma_x\Phi(\omega)}$$

Where

$$\mu_x = \mathbf{X}\theta_{[m]}, \quad \sigma_x = \sqrt{\sigma}$$

and

$$\begin{aligned} \Phi\left(y_m(-b) - a - b\sum_{j=i}^{n} y_j\right) &= \Phi\left(\frac{-b\sigma_x}{\sigma_x}(y_m - \mu_x + \mu_x) - a - b\sum_{j=i}^{n} y_j\right) \\ &= \Phi\left(-b\sigma_x\frac{(y_m - \mu_x)}{\sigma_x} - b\mu_x - a - b\sum_{j=i}^{n} y_j\right) \end{aligned}$$

Thus,

$$-b\sigma_x = c, \quad \omega\sqrt{1+c^2} = -b\mu_x - a - b\sum_{j=i}^{n} y_j$$

$$\Rightarrow \omega = \frac{-a - b\sum_{j=i}^{n} y_j - b\mu_x}{\sqrt{1+\sigma b^2}}$$

Therefore,

$$\begin{aligned} f_{(Y_m|\cdot)}(x) &= \frac{\phi(\frac{x-\mu_x}{\sqrt{\sigma}})\Phi(\omega\sqrt{1+(-b\sqrt{\sigma})^2} - b\sqrt{\sigma}(\frac{x-\mu_x}{\sqrt{\sigma}}))}{\sqrt{\sigma}\Phi(\omega)} \\ &= \frac{\phi(\frac{x-\mu_x}{\sqrt{\sigma}})\Phi(-a - b\sum_{j=i}^{n} y_j - bx)}{\sqrt{\sigma}\Phi(\omega)}. \end{aligned}$$

APPENDIX C: CHAPTER 5 DETAILS

C.1 Proof that optimal classification and clustering are equal when G=2

Claim: For G=2 the partition induced from the optimal classification is equivalent to the optimal partition.

Proof by induction.

Atomic Case: $n = 2$. There are two partitions. $\{1, 2\}$ and $\{\{1\}, \{2\}\}$

$$p(1, 2) = \pi_{1,1}\pi_{2,1} + \pi_{1,2}\pi_{2,2}$$

$$p(1, 2) = \pi_{1,1}\pi_{2,2} + \pi_{1,2}\pi_{2,1}$$

Let $\pi_{i,U} = max(\pi_{i,1}, \pi_{i,2})$ and $\pi_{i,L} = min(\pi_{i,1}, \pi_{i,2})$

Then

$$\pi_{1,U}\pi_{2,U} + \pi_{1,L}\pi_{2,L} - (\pi_{1,U}\pi_{2,L} + \pi_{1,L}\pi_{2,U})$$

$$= \pi_{1,U}(\pi_{2,U} - \pi_{2,L}) + \pi_{1,L}(\pi_{2,L} - \pi_{2,U})$$

$$= (\pi_{1,U} - \pi_{1,L})(\pi_{2,U} - \pi_{2,L}) > 0$$

Thus $\pi_{1,U}\pi_{2,U} + \pi_{1,L}\pi_{2,L} > \pi_{1,U}\pi_{2,L} + \pi_{1,L}\pi_{2,U}$ and whichever partition corresponds to the probability $\pi_{1,U}\pi_{2,U} + \pi_{1,L}\pi_{2,L}$ is the optimal partition. But $\pi_{1,U}\pi_{2,U}$ is the probability of the optimal cluster assignment. Thus when G=2 the partition induced by the optimal classification is the optimal partition.

Induction Step: For n data points the optimal partition equals the clustering induced by the optimal classifier.

Once again the probability of any partition can be written as the sum of the two classification probabilities which generate the partition. So for n+1 points let the optimal partition be

$\pi_{1,x_1} \dots \pi_{n,x_n} \pi_{n+1,U} + \pi_{1,x'_1} \dots \pi_{n,x'_n} \pi_{n+1,L}$, where x' is always the opposite assignment of x.

Consider

$$\pi_{1,U} \dots \pi_{n,U} \pi_{n+1,U} + \pi_{1,L} \dots \pi_{n,L} \pi_{n+1,L} - (\pi_{1,x_1} \dots \pi_{n,x_n} \pi_{n+1,U} + \pi_{1,x'_1} \dots \pi_{n,x'_n} \pi_{n+1,L})$$

$$= \pi_{n+1,U}(\pi_{1,U} \dots \pi_{n,U} - \pi_{1,x_1} \dots \pi_{n,x_n}) - \pi_{n+1,L}(\pi_{1,x'_1} \dots \pi_{n,x'_n} - \pi_{1,L} \dots \pi_{n,L})$$

By the induction assumption we know that

$$\pi_{1,U} \dots \pi_{n,U} + \pi_{1,L} \dots \pi_{n,L} \geq \pi_{1,x_1} \dots \pi_{n,x_n} + \pi_{1,x'_1} \dots \pi_{n,x'_n}$$

$$\Rightarrow \pi_{1,U} \dots \pi_{n,U} - \pi_{1,x_1} \dots \pi_{n,x_n} \geq \pi_{1,x'_1} \dots \pi_{n,x'_n} - \pi_{1,L} \dots \pi_{n,L}$$

So

$$= \pi_{n+1,U}(\pi_{1,U} \dots \pi_{n,U} - \pi_{1,x_1} \dots \pi_{n,x_n}) - \pi_{n+1,L}(\pi_{1,x'_1} \dots \pi_{n,x'_n} - \pi_{1,L} \dots \pi_{n,L})$$

$$> \pi_{n+1,U}(\pi_{1,x'_1} \dots \pi_{n,x'_n} - \pi_{1,L} \dots \pi_{n,L}) - \pi_{n+1,L}(\pi_{1,x'_1} \dots \pi_{n,x'_n} - \pi_{1,L} \dots \pi_{n,L})$$

$$= (\pi_{n+1,U} - \pi_{n+1,L})(\pi_{1,x'_1} \dots \pi_{n,x'_n} - \pi_{1,L} \dots \pi_{n,L}) > 0$$

Thus the probability of the optimal partition for n+1 points is given by

$$\pi_{1,U} \dots \pi_{n,U} \pi_{n+1,U} + \pi_{1,L} \dots \pi_{n,L} \pi_{n+1,L}$$

which is the probability for the partition generated by the optimal cluster since is contains the term $\pi_{1,U} \dots \pi_{n,U} \pi_{n+1,U}$.

By induction, when G=2 the optimal partition is equivalent to the partition induced by the optimal cluster.

C.2 Tables

C.2.1 Example 1

```
> kappa_table (1, pi1, delta)
    0   0.1   0.2   0.3   0.4   0.5   0.6   0.7   0.8   0.9 1
0   1 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1
0.1 1 0.998 0.996 0.995 0.994 0.994 0.994 0.995 0.996 0.998 1
0.2 1 0.989 0.981 0.975 0.972 0.971 0.972 0.975 0.981 0.989 1
0.3 1 0.972 0.950 0.935 0.926 0.922 0.926 0.935 0.950 0.972 1
0.4 1 0.945 0.902 0.871 0.852 0.846 0.852 0.871 0.902 0.945 1
0.5 1 0.910 0.840 0.790 0.760 0.750 0.760 0.790 0.840 0.910 1

> kappa_table (2, pi1, delta)
    0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
0   0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
0.1 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
0.2 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
0.3 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
0.4 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
0.5 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

> kappa_table (3, pi1, delta)
       0   0.1   0.2   0.3   0.4   0.5   0.6   0.7   0.8   0.9 1
0   1.00 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1
0.1 0.82 0.835 0.850 0.866 0.883 0.901 0.919 0.938 0.958 0.978 1
0.2 0.68 0.700 0.723 0.748 0.775 0.805 0.838 0.874 0.912 0.955 1
0.3 0.58 0.599 0.621 0.648 0.681 0.718 0.762 0.811 0.868 0.931 1
0.4 0.52 0.532 0.550 0.574 0.606 0.647 0.698 0.759 0.830 0.911 1
```

Table 6.1: CSENS for Mixed Tumor Data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	0.98	0.943	0.802	0.749	0.724	0.646	0.625
Average Linkage HC	0.98	0.959	0.874	0.835	0.817	0.785	0.785
Single Linkage HC	0.98	0.959	0.941	0.884	0.85	0.85	0.834
Complete Linkage HC	0.946	0.843	0.826	0.756	0.752	0.653	0.653

Table 6.2: CSPEC for Mixed Tumor Data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	0.51	0.813	0.873	0.951	0.963	0.977	0.988
Average Linkage HC	0.019	0.038	0.131	0.166	0.184	0.715	0.872
Single Linkage HC	0.019	0.038	0.057	0.112	0.149	0.601	0.613
Complete Linkage HC	0.65	0.685	0.691	0.727	0.727	0.808	0.947

```
0.5 0.50 0.503 0.512 0.530 0.559 0.600 0.655 0.725 0.808 0.901 1
```

```
> kappa_table (4, pi1, delta)
      0   0.1   0.2   0.3   0.4   0.5   0.6   0.7   0.8   0.9   1
0   1.0 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000   1
0.1 0.9 0.909 0.918 0.928 0.938 0.947 0.957 0.968 0.978 0.989   1
0.2 0.8 0.816 0.833 0.851 0.870 0.889 0.909 0.930 0.952 0.976   1
0.3 0.7 0.722 0.745 0.769 0.795 0.824 0.854 0.886 0.921 0.959   1
0.4 0.6 0.625 0.652 0.682 0.714 0.750 0.789 0.833 0.882 0.938   1
0.5 0.5 0.526 0.556 0.588 0.625 0.667 0.714 0.769 0.833 0.909   1
```

Table 6.3: CSENS+CSPEC for mixed tumor data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	1.49	1.76	1.68	1.7	1.69	1.62	1.61
Average Linkage HC	1	0.997	1	1	1	1.5	1.66
Single Linkage HC	1	0.997	0.998	0.997	0.999	1.45	1.45
Complete Linkage HC	1.6	1.53	1.52	1.48	1.48	1.46	1.6

Table 6.4: CPPV for mixed tumor data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	0.402	0.628	0.68	0.837	0.867	0.903	0.947
Average Linkage HC	0.251	0.251	0.252	0.252	0.251	0.48	0.673
Single Linkage HC	0.251	0.251	0.251	0.251	0.251	0.417	0.419
Complete Linkage HC	0.476	0.473	0.472	0.482	0.48	0.533	0.806

Table 6.5: CNPV for mixed tumor data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	0.987	0.977	0.929	0.919	0.912	0.892	0.887
Average Linkage HC	0.748	0.736	0.756	0.75	0.75	0.908	0.923
Single Linkage HC	0.748	0.736	0.742	0.743	0.748	0.923	0.917
Complete Linkage HC	0.973	0.929	0.922	0.899	0.897	0.874	0.891

C.2.2 Mixed Tumor Data

C.2.3 Lung Cancer Subtype Data

Table 6.6: CPPV+CNPV for mixed tumor data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	1.39	1.61	1.61	1.76	1.78	1.8	1.83
Average Linkage HC	0.999	0.987	1.01	1	1	1.39	1.6
Single Linkage HC	0.999	0.987	0.992	0.994	0.999	1.34	1.34
Complete Linkage HC	1.45	1.4	1.39	1.38	1.38	1.41	1.7

Table 6.7: CSENS for lung cancer data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	0.998	0.511	0.425	0.365	0.379	0.37	0.263
Average Linkage HC	1	0.986	0.984	0.975	0.962	0.948	0.947
Single Linkage HC	0.998	0.996	0.983	0.969	0.968	0.96	0.96
Complete Linkage HC	0.93	0.651	0.424	0.414	0.264	0.264	0.259

Table 6.8: CSPEC for lung cancer data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	0.351	0.682	0.81	0.876	0.94	0.941	0.957
Average Linkage HC	0.368	0.373	0.373	0.373	0.378	0.382	0.382
Single Linkage HC	0.018	0.035	0.041	0.047	0.065	0.205	0.362
Complete Linkage HC	0.404	0.688	0.839	0.845	0.87	0.878	0.92

Table 6.9: CSENS+CSPEC for lung cancer data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	1.35	1.19	1.23	1.24	1.32	1.31	1.22
Average Linkage HC	1.37	1.36	1.36	1.35	1.34	1.33	1.33
Single Linkage HC	1.02	1.03	1.02	1.02	1.03	1.16	1.32
Complete Linkage HC	1.33	1.34	1.26	1.26	1.13	1.14	1.18

Table 6.10: CPPV for lung cancer data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	0.603	0.614	0.688	0.744	0.862	0.862	0.858
Average Linkage HC	0.61	0.608	0.608	0.606	0.604	0.603	0.602
Single Linkage HC	0.501	0.505	0.503	0.501	0.506	0.544	0.598
Complete Linkage HC	0.607	0.673	0.722	0.726	0.667	0.682	0.761

Table 6.11: CNPV for lung cancer data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	0.993	0.585	0.587	0.582	0.605	0.602	0.568
Average Linkage HC	0.999	0.964	0.959	0.939	0.909	0.882	0.879
Single Linkage HC	0.905	0.907	0.708	0.609	0.669	0.84	0.902
Complete Linkage HC	0.853	0.666	0.595	0.593	0.544	0.547	0.557

Table 6.12: CPPV+CNPV for lung cancer data

	K=2	K=3	K=4	K=5	K=6	K=7	k=8
K-Means	1.6	1.2	1.28	1.33	1.47	1.46	1.43
Average Linkage HC	1.61	1.57	1.57	1.54	1.51	1.48	1.48
Single Linkage HC	1.41	1.41	1.21	1.11	1.17	1.38	1.5
Complete Linkage HC	1.46	1.34	1.32	1.32	1.21	1.23	1.32

BIBLIOGRAPHY

Albatineh, A. N. (2010), "Means and variances for a family of similarity indices used in cluster analysis," *Journal of Statistical Planning and Inference*, 140, 2828–2838.

Albatineh, A. N., Niewiadomska-Bugaj, M., and Mihalko, D. (2006), "On similarity indices and correction for chance agreement," *Journal of Classification*, 23, 301–313.

Arbelaitz, O., Gurrutxaga, I., Muguerza, J., Pérez, J. M., and Perona, I. n. (2013), "An extensive comparative study of cluster validity indices," *Pattern Recognition*, 46, 243–256.

Azzalini, A. and Capitanio, A. (2014), *The skew-normal and related families*, Cambridge: Cambridge University Press.

Beardsley, D. I., Kowbel, D., Lataxes, T. A., Mannino, J. M., Xin, H., Kim, W.-J., Collins, C., and Brown, K. D. (2003), "Characterization of the novel amplified in breast cancer-1 (NABC1) gene product." *Experimental cell research*, 290, 402–13.

Bhattacharjee, A., Richards, W. G., Staunton, J., Li, C., Monti, S., Vasa, P., Ladd, C., Beheshti, J., Bueno, R., Gillette, M., Loda, M., Weber, G., Mark, E. J., Lander, E. S., Wong, W., Johnson, B. E., Golub, T. R., Sugarbaker, D. J., and Meyerson, M. (2001), "Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses." *Proceedings of the National Academy of Sciences of the United States of America*, 98, 13790–13795.

Breitwieser, F. P., Müller, A., Dayon, L., Köcher, T., Hainard, A., Pichler, P., Schmidt-Erfurth, U., Superti-Furga, G., Sanchez, J. C., Mechtler, K., Bennett, K. L., and Colinge, J. (2011), "General statistical modeling of data from protein relative expression isobaric tags," *Journal of Proteome Research*, 10, 2758–2766.

Clough, T., Thaminy, S., Ragg, S., Aebersold, R., and Vitek, O. (2012), "Statistical protein quantification and significance analysis in label-free LC-MS experiments with complex designs." *BMC bioinformatics*, 13 Suppl 1, S6.

Cox, J., Hein, M. Y., Luber, C. A., Paron, I., Nagaraj, N., and Mann, M. (2014), "MaxLFQ allows accurate proteome-wide label-free quantification by delayed normalization and maximal peptide ratio extraction." *Molecular & cellular proteomics : MCP*, 13, 2513–2526.

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

Dabney, A. R. and Storey, J. D. (2007), "A new approach to intensity-dependent normalization of two-channel microarrays," *Biostatistics*, 8, 128–139.

de Brevern, A. G., Hazout, S., and Malpertuy, A. (2004), "Influence of microarrays experiments missing values on the stability of gene groups by hierarchical clustering." *BMC bioinformatics*, 5, 114.

Duncan, J. S., Whittle, M. C., Nakamura, K., Abell, A. N., Midland, A. a., Zawistowski, J. S., Johnson, N. L., Granger, D. a., Jordan, N. V., Darr, D. B., Usary, J., Kuan, P. F., Smalley, D. M., Major, B., He, X., Hoadley, K. a., Zhou, B., Sharpless, N. E., Perou, C. M., Kim, W. Y., Gomez, S. M., Chen, X., Jin, J., Frye, S. V., Earp, H. S., Graves, L. M., and Johnson, G. L. (2012), "Dynamic reprogramming of the kinome in response to targeted MEK inhibition in triple-negative breast cancer," *Cell*, 149, 307–321.

Dunn, J. C. (1973), "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters," .

Eidhammer, I., Flikka, K., Martens, L., and Mikalsen, S.-O. (2008), *Computational Methods for Mass Spectrometry Proteomics*, John Wiley & Sons.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2004), *Bayesian Data Analysis Second Edition*, vol. 1, Chapman & Hall/CRC.

Hagen, L. and Kahng, A. B. (1992), "New spectral methods for ratio cut partitioning and clustering," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 11, 1074–1085.

Hastie, T., Tibshirani, R., and Friedman, J. H. (2009), *The elements of statistical learning : data mining, inference, and prediction*, Springer; 2nd ed. 2009.

Hoshida, Y., Brunet, J. P., Tamayo, P., Golub, T. R., and Mesirov, J. P. (2007), "Subclass mapping: Identifying common subtypes in independent disease data sets," *PLoS ONE*, 2.

Jain, A. K. (2010), "Data clustering: 50 years beyond K-means," *Pattern Recognition Letters*, 31, 651–666.

Karp, N. a., Huber, W., Sadowski, P. G., Charles, P. D., Hester, S. V., and Lilley, K. S. (2010), "Addressing accuracy and precision issues in iTRAQ quantitation." *Molecular & cellular proteomics : MCP*, 9, 1885–1897.

Karpievitch, Y., Stanley, J., Taverner, T., Huang, J., Adkins, J. N., Ansong, C., Heffron, F., Metz, T. O., Qian, W. J., Yoon, H., Smith, R. D., and Dabney, A. R. (2009), "A statistical framework for protein quantitation in bottom-up MS-based proteomics," *Bioinformatics*, 25, 2028–2034.

Knochenmuss, R. (2012), "MALDI Ionization Mechanisms: An Overview," in *Electrospray and MALDI Mass Spectrometry: Fundamentals, Instrumentation, Practicalities, and Biological Applications: Second Edition*, pp. 147–183.

Kohonen, T. (1982), "Self-organized formation of topologically correct feature maps," *Biological Cybernetics*, 43, 59–69.

Koopmans, F., Cornelisse, L. N., Heskes, T., and Dijkstra, T. M. H. (2014), "Empirical bayesian random censoring threshold model improves detection of differentially abundant proteins." *Journal of proteome research*, 13, 3871–80.

Lesur, A. and Domon, B. (2015), "Advances in high-resolution accurate mass spectrometry application to targeted proteomics." *Proteomics*, 15, 880–90.

Li, X.-J., Zhang, H., Ranish, J. A., and Aebersold, R. (2003), "Automated statistical analysis of protein abundance ratios from data generated by stable-isotope dilution and tandem mass spectrometry." *Analytical chemistry*, 75, 6648–57.

Lucas, J. E., Thompson, J., Dubois, L. G., McCarthy, J., Tillmann, H., Thompson, A., Shire, N., Hendrickson, R., Dieguez, F., Goldman, P., Schwarz, K., Patel, K., McHutchison, J., and Moseley, M. (2012), "Metaprotein expression modeling for label-free quantitative proteomics," *BMC Bioinformatics*, 13, 74.

Luo, R., Colangelo, C. M., Sessa, W. C., and Zhao, H. (2009), "Bayesian Analysis of iTRAQ Data with Nonrandom Missingness: Identification of Differentially Expressed Proteins." *Statistics in biosciences*, 1, 228–245.

MacQueen, J. (1967), "Some methods for classification and analysis of multivariate observations," in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, The Regents of the University of California.

McLachlan, G. J. and Basford, K. E. (1987), *Mixture Models: Inference and Applications to Clustering*, Taylor & Francis.

Monti, S., Tamayo, P., Mesirov, J., and Golub, T. (2003), "Consensus clustering: A resampling based method for class discovery and visualization of gene expression microarray data," *Machine Learning*, 52, 91–118.

Oberg, A. L. and Mahoney, D. W. (2012), "Statistical methods for quantitative mass spectrometry proteomic experiments with labeling." *BMC bioinformatics*, 13 Suppl 1, S7.

Oberg, A. L. and Vitek, O. (2009), "Statistical design of quantitative mass spectrometry-based proteomic experiments." *Journal of proteome research*, 8, 2144–56.

Polpitiya, A. D., Qian, W.-J., Jaitly, N., Petyuk, V. A., Adkins, J. N., Camp, D. G., Anderson, G. A., and Smith, R. D. (2008), "DAnTE: a statistical tool for quantitative analysis of -omics data." *Bioinformatics (Oxford, England)*, 24, 1556–8.

Qiu, X., Wu, H., and Hu, R. (2013), "The impact of quantile and rank normalization procedures on the testing power of gene differential expression analysis." *BMC bioinformatics*, 14, 124.

Rand, W. M. (1971), "Objective Criteria for the Evaluation of Clustering Methods," *Journal of the American Statistical Association*, 66, 846–850.

Ross, P. L., Huang, Y. N., Marchese, J. N., Williamson, B., Parker, K., Hattan, S., Khainovski, N., Pillai, S., Dey, S., Daniels, S., Purkayastha, S., Juhasz, P., Martin, S., Bartlet-Jones, M., He, F., Jacobson, A., and Pappin, D. J. (2004), "Multiplexed protein quantitation in Saccharomyces cerevisiae using amine-reactive isobaric tagging reagents." *Molecular & cellular proteomics : MCP*, 3, 1154–69.

Sandin, M., Krogh, M., Hansson, K., and Levander, F. (2011), "Generic workflow for quality assessment of quantitative label-free LC-MS analysis." *Proteomics*, 11, 1114–24.

Scheffé, H. (1999), *The Analysis of Variance*, John Wiley & Sons.

Schliekelman, P. and Liu, S. (2014), "Quantifying the effect of competition for detection between coeluting peptides on detection probabilities in mass-spectrometry-based proteomics." *Journal of proteome research*, 13, 348–61.

Seber, G. A. F. (2009), *Multivariate Observations*, John Wiley & Sons.

Shilov, I. V., Seymour, S. L., Patel, A. A., Loboda, A., Tang, W. H., Keating, S. P., Hunter, C. L., Nuwaysir, L. M., and Schaeffer, D. A. (2007), "The Paragon Algorithm, a next generation search engine that uses sequence temperature values and feature probabilities to identify peptides from tandem mass spectra." *Molecular & cellular proteomics : MCP*, 6, 1638–55.

Sø rlie, T., Perou, C. M., Tibshirani, R., Aas, T., Geisler, S., Johnsen, H., Hastie, T., Eisen, M. B., van de Rijn, M., Jeffrey, S. S., Thorsen, T., Quist, H., Matese, J. C., Brown, P. O., Botstein, D., Lø nning, P. E., and Bø rresen Dale, A. L. (2001), "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, 10869–74.

Steinhaus, H. (1956), "Sur la division des corp materiels en parties," *Bull. Acad. Polon. Sci*, 1, 801 – 804.

Taylor, S. L., Leiserowitz, G. S., and Kim, K. (2013), "Accounting for undetected compounds in statistical analyses of mass spectrometry 'omic studies." *Statistical applications in genetics and molecular biology*, 12, 703–22.

Tekwe, C. D., Carroll, R. J., and Dabney, A. R. (2012), "Application of survival analysis methodology to the quantitative analysis of LC-MS proteomics data," *Bioinformatics*, 28, 1998–2003.

Thalamuthu, A., Mukhopadhyay, I., Zheng, X., and Tseng, G. C. (2006), "Evaluation and comparison of gene clustering methods in microarray analysis." *Bioinformatics (Oxford, England)*, 22, 2405–12.

Tibshirani, R., Hastie, T., Narasimhan, B., Soltys, S., Shi, G., Koong, A., and Le, Q. T. (2004), "Sample classification from protein mass spectrometry, by 'peak probability contrasts'," *Bioinformatics*, 20, 3034–3044.

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., and Altman, R. B. (2001), "Missing value estimation methods for DNA microarrays," *Bioinformatics*, 17, 520–525.