Shrinkage of dispersion parameters in the double exponential family of distributions, with applications to genomic sequencing

by

Sean Matthew Ruddy

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Statistics

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Assistant Professor Elizabeth Purdom, Chair
Associate Professor Haiyan Huang
Professor John Ngai

Fall 2014

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

Shrinkage of dispersion parameters in the double exponential family of distributions, with applications to genomic sequencing

Abstract

Shrinkage of dispersion parameters in the double exponential family of distributions, with applications to genomic sequencing

by

Sean Matthew Ruddy

Doctor of Philosophy in Statistics

University of California, Berkeley

Assistant Professor Elizabeth Purdom, Chair

The prevalence of sequencing experiments in genomics has led to an increased use of methods for count data in analyzing high-throughput genomic data to perform analyses. The importance of shrinkage methods in improving the performance of statistical methods remains. A common example is that of gene expression data, where the counts per gene are often modeled as some form of an overdispersed Poisson. In this case, shrinkage estimates of the per-gene dispersion parameter have lead to improved estimation of dispersion in the case of a small number of samples. We address a different count setting introduced by the use of sequencing data: comparing differential proportional usage via an overdispersed binomial model. Such a model can be useful for testing differential exon inclusion in mRNA-Seq experiments in addition to the typical differential gene expression analysis. In this setting, there are fewer such shrinkage methods for the dispersion parameter. We introduce a novel method that is developed by modeling the dispersion based on the double exponential family of distributions proposed by Efron (1986), also known as the exponential dispersion model (Jorgensen, 1987). Our methods (WEB-Seq and DEB-Seq) are empirical bayes strategies for producing a shrunken estimate of dispersion that can be applied to any double exponential dispersion family, though we focus on the binomial and poisson. These methods effectively detect differential proportional usage, and have close ties to the weighted likelihood strategy of edgeR developed for gene expression data (Robinson and Smyth, 2007; Robinson *et al.*, 2010). We analyze their behavior on simulated data sets as well as real data for both differential exon usage and differential gene expression. In the exon usage case, we will demonstrate our methods' superior ability to control the FDR and detect truly different features compared to existing methods. In the gene expression setting, our methods fail to control the FDR; however, the rankings of the genes by p-value is among the top performers and proves to be robust to both changes in the probability distribution used to generate the counts and in low sample size situations. We provide implementation of our methods in the R package `DoubleExpSeq` available from the Comprehensive R Archive Network (CRAN).

Dedication

This dissertation is dedicated to Stephen Ruddy. I know him as my best friend, my mentor, my biggest fan, my source of unconditional love and support, but most importantly my dad. There is no greater title that I could ever achieve that would compare to that of being his son. He passed away unexpectedly in April of 2012 and not a single day has or ever will pass that I am not eternally thankful for all he has given me, has sacrificed, so that I could succeed.

Contents

List of Figures

List of Tables

Acknowledgments

I would like to start off by acknowledging my college mathematics professor, Kathy Perino. Her support, encouragement and belief in my abilities gave me the confidence and inspiration needed in my young adult life to continue in my academic endeavors.

I would like to also thank my advisor, Elizabeth Purdom, without whom my success would not have been realized. Her dedication to my work as well as her consistent positive attitude and endless suggestions at each road block in my research was essential for completing my Ph.D..

I would also like to thank another Ph.D. student of Elizabeth Purdom, Marla Johnson, for all the effort and work she put into obtaining and managing the real data sets that I have used throughout my research projects, in addition to fulfilling requests and fielding questions concerning these data.

I would also like to thank all the members of the Speed-Dudoit-Purdom Biostatistics Group whose helpful questions and insights over the years have been very useful at various stages of my work.

Lastly, I would like to thank my committee members Haiyan Huang and John Ngai for their support and helpful input.

Chapter 1

Introduction

"My soul is a hidden orchestra; I know not what instruments, what fiddlestrings and harps, drums and tamboura I sound and clash inside myself. All I hear is the symphony."
—Fernando Pessoa

Written in the earlier half of the 20^{th} century, it wasn't long after that Watson and Crick formally described the structure of DNA in the 1950's beginning a new era in biology dedicated to uncovering the function of the so called fiddlestrings, harps, drums, and tamboura that sound and clash inside the nucleus of our cells. After all, cells are the organic machines that give rise to the structure, order and diversity of life that we see every day. Studying how they work will shed light not only on normal cell function but also the means by which diseases, such as cancer, disrupt this function.

The single most important job that the components of a cell have is to make proteins which in turn allow the cell to carry out its purpose in the organism. Therefore, knowing which cells produce which proteins and in what amounts is a key step to understanding how proteins are directly related to its function and gives insight into how diseases change cell behavior. Direct protein detection technology exists and is constantly being refined and innovated upon but may not provide a complete picture of genetic activity. Alternatives, like that of microarrays or sequencing technologies, are capable of quantifying an intermediate, genetic product called mRNA which has a direct relationship to protein synthesis. These approaches, while being a proxy for protein production, can also yield insight into other important aspects of the cell machinery. Sequencing technologies in particular have done this on unprecedented levels, the data of which is the focus of the statistical methods developed here.

Though sequencing was developed in the late 1970s, it took almost three decades before the technology was advanced enough to be used in large scale, genomic studies. These newer technologies fall under the heading of next-generation sequencing (NGS) and are capable of producing vast amounts of data spanning the entire genome, quickly and affordably. A frequent use of NGS methods is to sequence the mRNA found in a cell with the final data being counts created from aggregating these sequenced molecules over specific regions of the genome from which they originated. A common type of question in this setting is to compare the counts of sequences measured under different experimental conditions called differential analysis. This usually requires replicates from multiple sources of genetic material, referred to as biological replicates, as opposed to repeated uses from the same source, or technical replicates, in order for the findings to have implications for a meaningful population.

One setting in which differential analysis is common is the case of detecting changes in the mRNA levels of a gene between conditions. In this case, the counts are the number of sequenced mRNA aggregated over genes and serve as a measure of their relative expression. There are different amounts of total sequenced mRNA collected across samples, so that the question of interest is more precisely whether the proportion of counts allocated to a given gene varies between conditions. In the gene setting, however, the total number of sequences in a sample is in the millions and are spread across thousands of genes, and so the proportions are quite small. For this reason it is common to use a Poisson distribution applied to each gene separately to model the counts, with an offset parameter equal to the total number of sequences, and has been shown to adequately model data from technical replicates (Marioni *et al.*, 2008). Biological replicates, however, are known to contain additional sources of variation that are not well explained by the poisson model due to the implicit model assumption that the variance equals the mean. Consequently, there is not enough flexibility for estimating the additional variance, and this causes an underestimate of variability resulting in a large amount of false positives. This extra variability is called overdispersion relative to the Poisson and requires direct attention. The prominent modeling technique for overdispersed count data has been to use a negative binomial (Robinson and Smyth, 2007), though some methods have incorporated overdispersed binomial distributions such as the beta-binomial and the extra-binomial variation of Williams (1982).

Complicating matters, there are generally few samples (sometimes on the order of 10 or less) with which to estimate the variance of each gene producing unstable and imprecise estimates. In such a paradigm, it has been found that shrinkage of the individual parameter estimates or test statistics greatly improves the reproducibility of the results. The setting for many of these shrinkage routines was initially in the context of the continuous, roughly log-normal intensity data from microarray experiments (Smyth, 2004). The growth of relatively cheap sequencing technologies has resulted in sequencing becoming preferred over the previous generation of microarray technologies and, as a result, there has been great interest in how to most effectively use discrete distributions for common tasks such as shrinkage that previously relied on normal data. Most of the focus of count based methods for sequencing data has been on creating shrinkage estimators of the parameters related to the variance such as the dispersion which has proven to enhance the performance of the methods in small sample sizes (Robinson and Smyth, 2007; Anders and Huber, 2010; Zhou *et al.*, 2011; Yang *et al.*, 2012; Wu *et al.*, 2013; Yu *et al.*, 2013; Leng *et al.*, 2013).

Though applicable, very few shrinkage methods have been developed for overdispersed binomial models, primarily because they have proved to be more difficult to work with. However, there are genomic analyses that require the direct use of binomial models and prohibit the use of shrinkage methods relying on the poisson approximation to the binomial. Our motivating example comes from the question of measuring alternative splicing—when the gene can produce multiple versions of mRNA that include different combinations of the exons of a gene. One simple approach to finding differences in alternative splicing across samples is to measure the number of sequences including the exon and compare it to the number excluding the exon (Shen *et al.*, 2012; Wu *et al.*, 2011), in which case differences in the exon usage appear as a question of comparing proportions across conditions. Another example can be found in resequencing of tumors where the mutations can be present at different proportions in a sample and the question is to compare proportions of mutation inclusion across conditions.

We present two novel and effective approaches for shrinkage within the binomial family using the double binomial distribution introduced by Efron (Efron, 1986), and has applications to several genomic analyses that will be discussed later. The theory developed here is in enough generality to apply to an entire family of double exponential distributions of which the double poisson is also a member. We show results of these methods on simulated and real data sets in both the settings of differential exon usage and differential gene expression. In the exon usage case, we will demonstrate our methods' superior ability to control the FDR and detect truly different features compared to existing methods. In the gene expression setting, our methods fail to control the FDR; however, the rankings of the genes by p-value is among the top performers and proves to be robust to both changes in the probability distribution used to generate the counts and in low sample size situations. Though we outperform all other methods in this regard for 2 out of 3 simulations we conclude that the `voom` method (Law *et al.*, 2014) is the clear top performer overall among 15 other differential gene expression methods, including our own, due to both its ability to similarly assign high ranks to truly differential genes as well as its superior control of the FDR. We provide implementation of our methods in the R package `DoubleExpSeq` available from the Comprehensive R Archive Network (CRAN).

The layout of this paper is as follows: chapter 2 gives a basic description of the necessary biology as well as in depth discussions about technological and statistical topics briefly introduced here; Chapter 3 develops the theory for our double exponential family shrinkage methods in full detail; and we demonstrate the performance of these methods for differential exon usage and differential gene expression analyses in Chapter 4 and Chapter 5, respectively.

Chapter 2

Background

This chapter gives a more in depth treatment of several of the topics introduced in the previous chapter to provide a better understanding of genomic data and the obstacles faced by statisticians. First, an overview of genetic biology will be given followed by a detailed description of microarray and sequencing technology. Next, differential analysis will be more formally introduced and expanded upon, in addition to covering both the challenges it presents in its current large scale format and approaches that have been developed to meet these challenges. This includes proper normalization of sequence count data, appropriate modeling of overdispersion, and the current existing techniques for reducing the effects of low sample size through shrinkage.

2.1 Protein Production

Proteins are made from smaller building blocks called amino acids of which 22 are naturally part of biological organisms. A protein is created by linking together a chain of any combination of the amino acids including repeated uses and can be comprised of several hundred to several thousand amino acids in total. Of course, protein synthesis requires regulation if it is to create the complexity and order seen in organisms. Within the nucleus of every cell in our body lies the blueprints for every protein. These blueprints are contained in a tiny molecule called DNA, short for deoxyribonucleic acid, that is comprised of two separate long chains of 4 smaller molecules called nucleotides. In a human, a single chain is made up of approximately three billion of these nucleotides. These chains, or strands, are wrapped together, linked position by position by hydrogen bonds, taking the form similar to a spiral staircase which creates the infamous double-helix shape we call DNA. To further clarify, the strands are connected together in a complimentary way at each nucleotide position such that the nucleotide adenine (A) in one strand always lies opposite the nucleotide thymine (T) in the other strand, and the nucleotide cytosine (C) always lies opposite to the nucloetide guanine (G). Specific, spatial groupings of nucleotides making up each strand of the DNA are called genes. The full categorization of the DNA into genes makes up what we call the genome and is a major area of focus in genomics. These genes yield the instruction for which amino acids to put together and in what order for constructing proteins. More specifically, it is each triplet of nucleotides, called a codon, that codes for a particular amino acid. For example, the nucleotide triplet ACG codes for the amino acid Threomine and CCG codes for the amino acid Proline and so on.

Transcription begins the process of protein synthesis by copying the nucleotides in the DNA of a single gene in its complimentary form (A↔T, C↔G) to produce a new molecule called messenger-RNA, or mRNA for short. It is simply a single, short strand of nucleotides linked together with a minor change that the nucleotide thymine is replaced with a 5^{th} nucleotide uracil, abbreviated as U. This mRNA molecule then goes through what is called splicing where particular parts of the mRNA are removed and discarded, and what remains is the final collection of codons that go on to make the protein. After splicing is complete, the mRNA travels out of the nucleus and into the cytoplasm of the cell where it finds its way to the ribosome. The ribosome is a cell organelle that takes the mRNA molecule and interprets and manufactures a protein by linking together the amino acids corresponding to the order that the codons are read in a process called translation. See Figure 2.1 for a visual representation.

2.2 Alternative Splicing

As mentioned in the previous section, after a gene has been transcribed creating a duplicate copy, the new molecule undergoes splicing to remove sections pertaining to the gene that do not play a role in coding for its corresponding protein. The parts that remain are joined together to create the final mRNA product that goes on to produce the protein via translation. The removed sections are referred to as introns and the protein coding sections are called exons. In more complex organisms, it is not always the case that every exon of a gene are joined, and often the final mRNA product will only contain a subset of the available exons. These different mRNAs are generally referred to as the *isoforms* of the gene. The process by which these different isoforms are synthesized is called alternative splicing, sometimes abbreviated as AS. It allows even a single gene to create fundamentally different protein and therefore is a major contributor to the diversity we see in higher organisms. Figure 2.2 provides a visual representation.

2.3 Technologies for measuring mRNA

2.3.1 Microarrays

The first innovation in genomic technology arriving in the 1990s that allowed the simultaneous measure of expression levels for thousands of genes was microarrays. Prior to this was the first generation of sequencing which at that time was not sophisticated enough to allow for such large scale experiments affordably. What microarrays accomplished was unprecendented and within a few years large genetic experiments became commonplace among research institutions. This also motivated the development of statistical methods that now serve as the foundation to the methods designed for sequencing data.

Microarray technology works by first creating probes each of which are a unique sequence of DNA that matches a section of an mRNA molecule of interest. Note, this means that the sequence of the genes must be known ahead of time which is a major limitation to

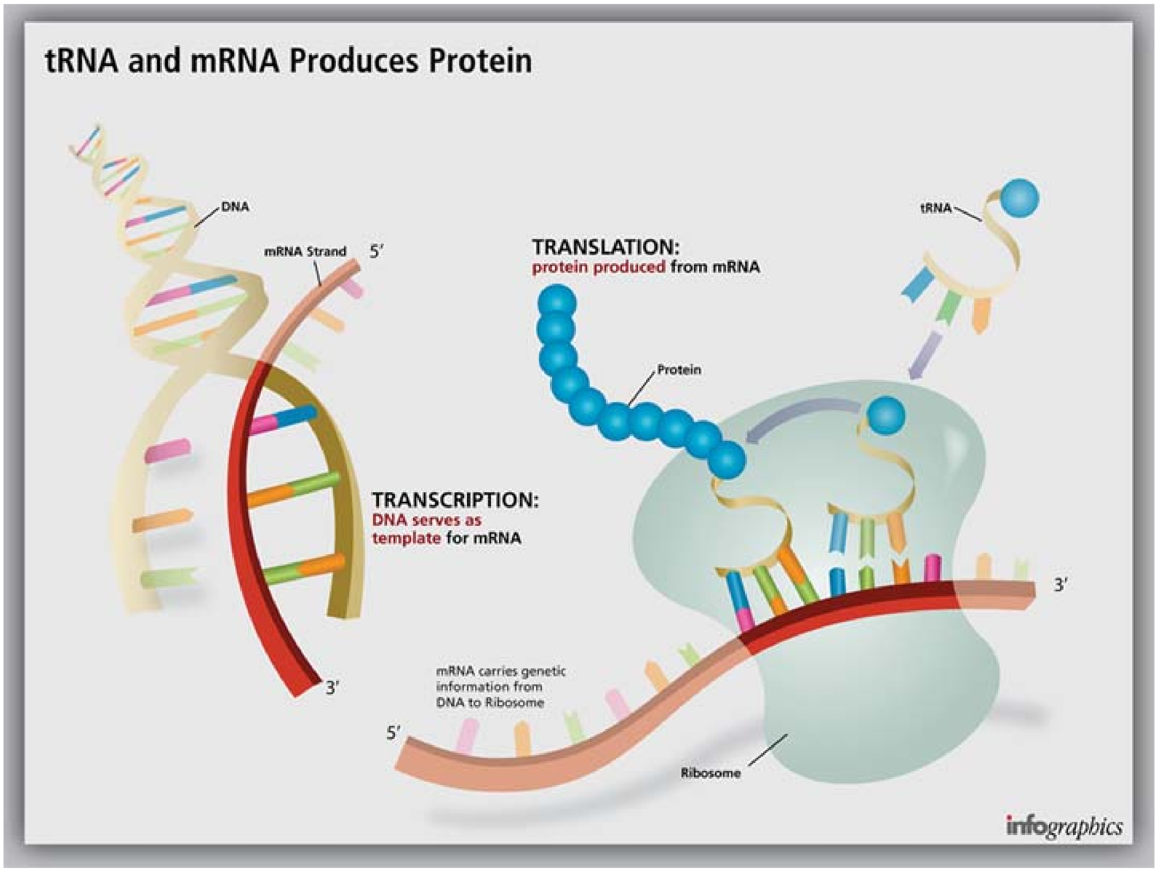

tRNA and mRNA Produces Protein
DNA
mRNA Strand
5′
TRANSLATION:
protein produced from mRNA
tRNA
Protein
TRANSCRIPTION:
DNA serves as template for mRNA
3′
3′
mRNA carries genetic information from DNA to Ribosome
5′
Ribosome
infographics

Figure 2.1: **Protein Synthesis.** Starting on the left where transcription occurs, the DNA is unraveled and with the help of the enzyme RNA polymerase, a section of a strand of DNA is copied from its $5'$ end toward the $3'$ by building together nucleotides complimentary to the DNA. The resulting mRNA molecule, after splicing, travels out of the nucleus to the ribosome where each triplet of nucleotides is used to obtain a specific amino acid. As the mRNA molecule continues, the ribosome connects the amino acids to produce the protein corresponding to the transcribed gene. This process, called translation, continues until the ribosome come across a specific codon that instructs the ribosome to stop building the protein, called a stop codon. Illustration by *tRNA and mRNA Produces Protein. Digital image. Infographics, Inc., 2013. Web. 18 July 2014. <http://www.ignyc.com/our-work/portfolio/transcription-translation/>.*

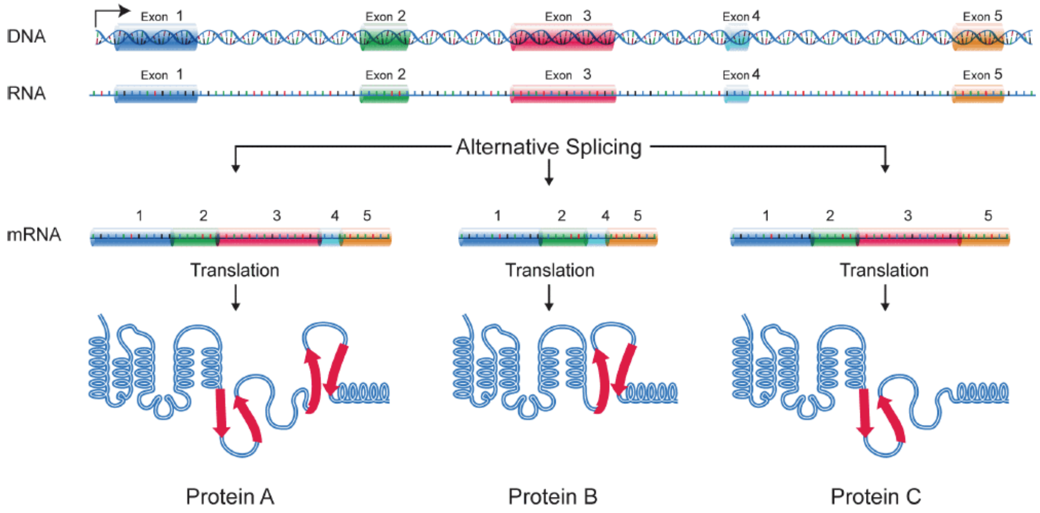


Figure 2.2: **Alternative Splicing.** A duplicate copy of a gene is created, producing a pre-mRNA molecule that contains all the gene's exons and introns (the space between the identified exons). This example shows the gene to have three corresponding isoforms, one of which uses all available exons and two other isoforms that only use a subset. The pre-mRNA then undergoes splicing: if the current cell environment wishes to synthesize the first isoform then splicing will remove only the introns and join together all exons; to create the second isoform, splicing will remove the third exon, in addition to all introns, and join the remaining four exons; and so on. The possibility of different outcomes indicates that this gene undergoes alternative splicing. Illustration by *Alternative Splicing. Digital image. National Human Genome Research Institute, n.d. Web. 18 July 2014.* $< http : //www.genome.gov/Images/EdKit/bio2j_{l}arge.gif >$

the technology. Tens of thousands of these probes are then aligned on a silicon chip or a glass surface in a grid format—the microarray. Each spot on the surface contains several thousand probes of the same sequence representing a unique gene or genomic region. The sample RNA (complimentary to the gene from which it was transcribed) is converted to a single stranded DNA molecule called cDNA and given a fluorescent tag. The cDNA is then placed over the microarray and they hybridize to their complimentary probes across all the spots on the surface. This is how the parallel analysis of thousands of genes occurs. The intensity of the fluorescent is measured for each spot on the microarray. In the case of comparing two groups of samples, for example, cancer cell mRNA versus normal cell mRNA the cDNA from one group is given the color red and the other group green. The intensity along the spectrum red-yellow-green shows the amount of differential expression of a gene in favor of one group over the other.

The resulting data is a continuous measure of intensity and theoretically has no upper bound. However, the machinery is limited in its range of detectable expression levels governed by the point at which the probes of a (highly expressed) gene become saturated and no further binding can occur. Conversely, there is also a lower bound of detectable expression where the signal cannot be differentiated from the background noise caused by the laser reflecting off the surface of the microarray. This lack of sensitivity can inhibit the discovery of interesting genes at both the higher and lower end of expression levels. Furthermore, because the cDNA hybridizes entire molecules to a probe complementary to the target sequence, microarrays suffer from a lack of specificity due to partial matches to the probe and even cross-hybridization where a cDNA sequence binds with the incorrect probe. Even with its limitations microarrays have been the main source for gene expression profiling in addition to other genomic experiments for nearly a decade.

2.3.2 First Generation Sequencing

Modern DNA sequencing methods began in 1977 with Sanger sequencing, named after its inventor Frederick Sanger. This method, in fact, played a major role in the completion of the Human Genome Project that set out to identify the 3 billion base pairs of the human genome for the first time. Sanger sequencing requires a single stranded fragment of DNA that has been amplified using polymerase chain reaction, or PCR, to create many copies of the exact same fragment of DNA. Primers—short sequences complimentary to a section of the DNA—are attached to a specific end of the DNA fragment called the $3'$ end which allow the further addition of more nucleotides. The DNA is then divided equally into 4 tubes labeled by the 4 nucleotides (A, T, C, G). All four free flowing nucleotides are added to each tube, along with a chemically altered, radioactively labeled version of the nucleotide that corresponds to the label on the tube. These altered nucleotides are the key ingredient to this method of sequencing. When hybridized to a single stranded DNA, the altered nucleotides terminate the sequencing process. Further, because each tube contains only one out of the 4 altered nuclotides, sequencing will always terminate on the nucleotide given by the tube label. The position at which an altered nucleotide binds with the DNA fragment is random and, therefore, we obtain sequences that, collectively across the tubes, terminate at every single nucleotide along the DNA sequence of interest.

The sequencing process begins with the addition of DNA polymerase which facilitates the addition of new nucleotides starting at the end of the primer and continuing along the DNA fragment until the binding of an altered nucleotide occurs. Once all the reactions have taken place within each of the 4 tubes, the now double stranded DNA fragments are denatured using heat into their single stranded form and the contents of each tube are transferred to 4 lanes, corresponding to the tube labels, at the negative end of an electrically charged, electrophoresis gel. A current is run through the gel causing the negatively charged DNA to move toward the positive end. The shorter fragments, being of lower molecular weight, travel faster and further than longer fragments along the lanes. They eventually stop and the gel is dried and an x-ray is taken. The radioactively labeled, altered nucleotides then appear as a band along the gel. These bands represent the last nucleotide that was sequenced from a fragment and is given by the lane assignments. Further, because every sequenced fragment began with the same primer, the order of the bands starting from the positive end moving across the lanes and toward the negative end reveal the complementary version of the DNA sequence of interest and can then be easily converted to the actual DNA sequence following the complimentary rule. Figure 2.3 is a visual representation of the sequencing results.

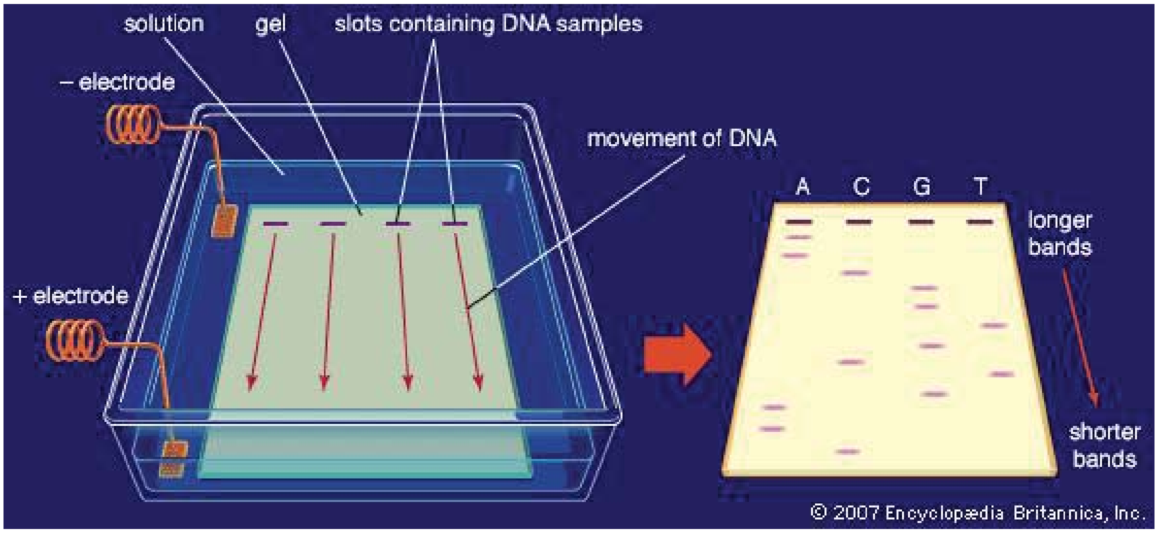


Figure 2.3: **DNA sequencing: gel electrophoresis.** After the DNA fragments have been sequenced and converted back to single stranded DNA, the contents of each of the 4 tubes are pipetted into the negative end of charged gel and in their respective lanes. The DNA moves toward the positive end with the shorter fragments moving farther. For each lane, bands appear at the positions of the DNA sequence where its corresponding nucleotide was encountered. The original DNA sequence is the complimentary version of the sequence read from the positive end to the negative end. In this example, the sequence read from the gel is CAAGTCGTGGCAA, and its complimentary form, GTTCAGCACCGTT, is the DNA sequence of interest. Illustration by *DNA sequencing: gel electrophoresis. Digital image. Encyclopædia Britannica, Inc., 2007. Web. 18 July 2014. <http://www.britannica.com/EBchecked/media/40224>*

2.3.3 Next Generation Sequencing

Sanger sequencing was the primary method for sequencing for over two decades until the advent of next generation sequencing (NGS) which all but replaced the Sanger approach, and certainly did so for large scale sequencing experiments. One major disadvantage to the Sanger method was the manual steps involved and though these eventually became automated it remained a slow, costly and low-throughput technology. In contrast, NGS methods provided new automated techniques of sequencing which significantly reduced the time and effort required to sequence DNA fragments producing large amounts of data. As a result, next generation sequencing is now the preferred method for DNA or RNA sequencing and has been steadily becoming the preferred method for genome analysis in general, owing to the rapid decline in price relative to microarrays, improvements in the accuracy, speed and coverage of the technology and pre-processing steps, the myriad of proven statistical methods that have been developed for various sequencing experiments, and the much improved sensitivity and specificity over microarray technologies. In the last decade since its introduction, NGS has made several strides in the area of personalized medicine, diagnostics, biomarker detection, disease classification and treatment, evolutionary biology, and genome assembly of new organisms to name a few, and is proving to be the a powerful tool for advancing our understanding of genomics on unprecedented levels.

Next generation sequencing methods, of which there are several, all use the common approach of massively parallel sequencing that yield vast amounts of data. Many of these methods differ in their approach by utilizing different biological and chemical techniques. The most widely used sequencing platfrom was designed by Illumina, Inc. and uses a method similar to Sanger sequencing in that they sequence nucleotide by nucleotide, also known as sequencing by sythesis, and add chemically altered nucleotides to halt the sequencing process. The altered nucleotides, however, are different from those used by Sanger and can be converted back to their natural state, giving the name cyclic reversible termination (CRT) to the method. “Cyclic” refers to the cycle of adding in terminating nucleotides, taking an image, removing their terminator component, and repeating.

Like all other NGS methods, the sample DNA or RNA must first be sheared to create fragments typically on the order of a couple hundred nucleotides in length, followed by PCR amplification which creates millions of copies of the fragments. In the case of Illumina sequencing, adapters are attached to the fragments of the sample, denatured to single stranded DNA, and then transferred to a solid surface called a flow cell where they bind and become immobilized. Free nucleotides are added along with a sequencing enzyme and copy the fragments to create double stranded DNA. These are denatured to create additional immobilized single stranded DNA and the process is repeated ultimately leaving clusters of copies of all original fragments. This results in millions of clusters each containing about 1000 copies that are now ready to be sequenced using the CRT method.

Each of the altered nucleotides are given the same fluorescent tag and because of this the nucleotides must be added in one at a time in order to be differentiated. At the start of the cycle, a large amount of a single termination nucleotide is added to the flow cell, along with primers and DNA polymerase, and begins sequencing at the $3'$ end of the fragment where the primer has been attached. Because these are chain terminating nucleotides, as soon as a single one has bound to a fragment no more of the terminating nucleotides can bind.

Once this is complete, an image is taken to detect the fluorescent tag from the fragments that binded with a terminating nucleotide. Since the added terminating nucleotide is known before addition, the image indicates the nucleotide of the first position of these fragments. The other 3 terminating nucleotide types are added in succession separated by an imaging step. After all types have been added, the first position of all fragments are now known. The final step in the cycle is to remove the terminating component placed on the nucleotides which will now open up the second position for sequencing. This cycle is repeated up to a pre-determined amount. Figure 2.4 is a visual representation.

These sequencing method produces short sequences of reads on the order of 100 nucleotides in length but does not contain any immediate information about where along the genome the read originated. As a result, the genome must be searched for locations matching the sequence of the read. This requires, of course, a known genome for reference which exists for several organisms including humans. Mapping of the reads to the genome must be done for every read produced by the sequencing method which roughly speaking is in the tens to hundreds of millions. In the case when a reference genome is not available de novo methods can be used that attempt to build the genome using only the reads produced in the experiment. It is worth noting that neither of these approaches are trivial and several mapping and de novo software are continually being refined to improve speed and accuracy of their algorithms. When completed the reads are aggregated over predefined regions of the genome, for example, within genes. This produces a count for each region and makes up the count data that will be analyzed to the purpose of the experiment.

2.4 Differential Expression

2.4.1 Models for Sequencing Data

We start with an $n \times p$ matrix, Y, of data where each y_{ig} entry is a measure for feature g of sample i. The goal of differential analysis is to identify interesting features which amounts to comparing vectors of means between conditions:

$$\boldsymbol{\mu}_{c_1} = (\mu_{1c_1},\ \mu_{2c_1},\ ...,\ \mu_{pc_1}) \text{ versus } \boldsymbol{\mu}_{c_2} = (\mu_{1c_2},\ \mu_{2c_2},\ ...,\ \mu_{pc_2})$$

Given the high dimensional property of genomic data and other modeling challenges it presents, difficulties arise in designing suitable tests for such a comparison. To simplify the analysis, it is common to instead marginally examine the effect of each feature independently with a simple statistical test and results in a separate test for each genomic feature g:

$$H_0^g : \mu_{c_1,g} = \mu_{c_2,g}, \quad H_1^g : \mu_{c_1,g} \neq \mu_{c_2,g}$$

In this way, features can be modeled independently and under classical settings where a full column rank design matrix representing the conditions can be set up. Commonly, this is done using a GLM model for each feature separately across the n samples:

$$\eta(\mu_{ig}) = \beta_{0g} + \beta_{1g} \cdot 1_{\{\rho(i)=c_1\}} \ \text{(for i} = 1, ..., \text{n)},$$

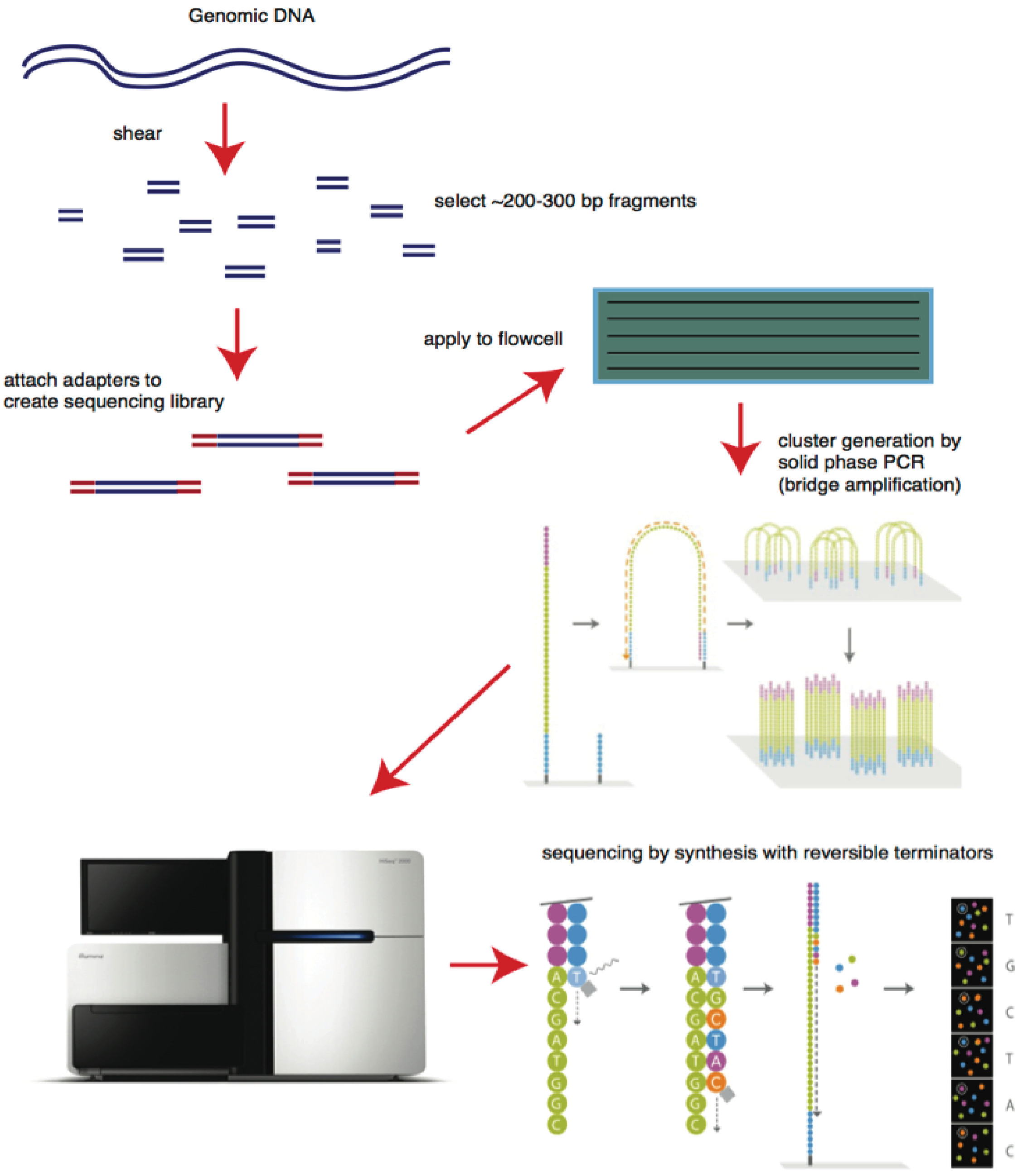


Figure 2.4: **Illumina Sequencing.** DNA is sheared to produce smaller fragments. Adapters are attached and the DNA is denatured using heat to produce single stranded DNA. These are then transferred to the flow cell lanes where the molecules attach and are immobilized. PCR amplification ensues creating millions of clusters consisting of copies of each fragment. Using the Cyclic Reversible Termination method (CRT), the fragments are sequenced position by position separated by 4 imaging steps, one for each nucleotide type, yielding tens of millions of reads. Illustration by *Brown, Stuart M. Sequencing-by-Synthesis: Explaining the Illumina Sequencing Technology. Digital image. BitesizeBio, 30 Aug. 2012. Web. 18 July 2014. <http://bitesizebio.com/13546>*

where η is the link function that relates the mean to the model parameters, β_{0g} is the overall strength of the signal across all samples, $\rho(\cdot)$ maps the sample subscript to its respective condition, and β_{1g} measures the effect that condition c_1 has on the mean, μ_{ig}, through η.

For RNA-Seq data, each y_{ig} entry of the data matrix is a count of the total number of sequenced fragments, or reads, originating from the genomic feature g. These counts can be viewed as following a binomial process in which the number of reads allocated to a feature is a proportion of the total number of reads, t_i, available to it in a sample:

$$y_{ig} \sim \mathrm{Bin}(t_i, \lambda_g), \quad \text{for i} = 1, ..., \text{n}.$$

Because the library sizes, t_i, are large and spread across thousands of features, the proportions, λ_g, are small. The counts can then be modeled more simply by a poisson distribution and is the most common assumption made in the gene expression setting:

$$y_{ig} \sim \mathrm{Poisson}(\mu_{ig} = t_i \lambda_g),$$

and a log-linear model is fit to the counts with an offset parameter equal to the library size:

$$\log \mu_{ig} = \beta_{0g} + \beta_{1g} \cdot 1_{\{\rho(i)=c_1\}} + \log(\mathrm{t_i}), \quad \text{for i} = 1, ..., \text{n}.$$

This simple model has shown to be suitable for describing data from technical replicates (Marioni *et al.*, 2008) but inadequate for more interesting experiments that utilize biological replicates that naturally introduce additional variation in the observed counts. Figure 2.5 shows such a situation for gene data where the raw variance of the counts for each gene is plotted against their raw mean. The purple line represents the intrinsic assumption in the poisson model that the variance equals the mean. The orange lines represent an estimate of the mean-variance relationship and we see that the poisson assumption greatly underestimates the observed trend, especially for highly expressed genes and leads to a high number of false positives. The lack of ability to describe the observed variance is referred to as overdispersion relative to the poisson.

In the setting of microarray experiments where the data are continuous measures of intensity for each gene, the prominent modeling technique is to assume the log of the intensities are normally distributed. This results in a mean and variance parameter that can be estimated independently of each other with the implication that any degree of variation in the observed data can be accurately estimated. For that reason, some RNA-Seq methods have proposed transformations of the counts to data that can be more suitably modeled using a normal distribution (Law *et al.*, 2014; Anders and Huber, 2010; Love *et al.*, 2014).

Modeling the counts directly is more advantageous in low sample sizes and for low expressed genes where procedures based on normality may not be appropriate. The most common approach for modeling overdispersion in count data is to use a negative binomial distribution. The negative binomial can be viewed as a hierarchical model by placing a gamma prior on the mean of the poisson distribution:

$$\begin{aligned} Y|\mu &\sim \mathrm{Poisson}(\text{mean} = \mu) \\ \mu|(r, \beta) &\sim \mathrm{Gamma}(\text{shape} = r, \text{rate} = \beta), \end{aligned}$$

Integrating over μ results in a negative binomial distribution for the marginal of Y:

$$Y \sim \text{ NB}\left(\mu = \frac{r}{\beta}, \phi = \frac{1}{r}\right)$$
$$\text{with } \text{E}(Y) = \mu \text{ and } \text{V}(Y) = \mu + \phi\mu^2.$$

In its usual parameterization, r is the number of failures until the experiment is stopped and $p = 1/(1 + \beta)$ is the probability of success in one cycle of the experiment. Its present parameterization in terms of μ and ϕ is more interpretable in the context of RNA-Seq data and shows the direct relationship of the variance to its mean in addition to the dispersion parameter, ϕ. This extra parameter allows the variance to be adjusted freely of the mean beyond that of the poisson in the presence of overdispersion. Continuing with the gene example, the full model would become,

$$y_{ig} \sim \text{ NB}\left(\mu_{ig} = t_i\lambda_g,\ \phi_g\right)$$

Other overdispersed models including the beta-binomial, the extra-binomial variation of Williams (1982), and quasi-likelihood methods have all been used as a basis for assessing differential expression for RNA-Seq data (Zhou *et al.*, 2011; Yang *et al.*, 2012; Auer and Doerge, 2011). All of these introduce a dispersion parameter with the same effect as in the negative binomial framework with the caveat that these can also be used in situations when the poisson approximation to the binomial cannot be made, unlike the negative binomial. In that respect, these have more universal application for RNA-Seq data.

2.4.2 Normalization in Differential Analysis

As was demonstrated previously, the library size of each sample is added to the model to offset the effect that it has on the observed counts. After all, it is reasonable to assume that if two samples equally express all genes and one is sequenced at half the depth of the other then the counts for that sample should be approximately cut in half. Therefore, not accounting for the library size will lead to erroneous differential expression results. This effect is referred to as a technical effect since it is an artifact of the sequencing procedure, not the biology. Another technical effect occurs when each mRNA is fragmented and subsequently amplified producing thousands of copies of each fragment. Since the technology fragments the mRNA independent of its length, the result is that longer mRNA transcripts produce more fragments and therefore account for more total copies available for sequencing. Longer transcripts will, on average, appear to be higher expressed than shorter transcripts of equal expression. Approaches such as the RPKM method of Mortazavi *et al.* (2008) was developed with this in mind. For the purpose of differential analysis, however, the assumption is typically made that this inherent length bias is the same across all samples of a feature and therefore can be ignored since the means being compared are from the same feature and therefore the effect cancels out. However, it has been noted that the length bias may impart a preferential selection in favor of differentially expressed genes of longer length compared to their shorter counterparts which is also not ideal (Oshlack and Wakefield, 2009). That said, it is still very common to ignore the length bias altogether.

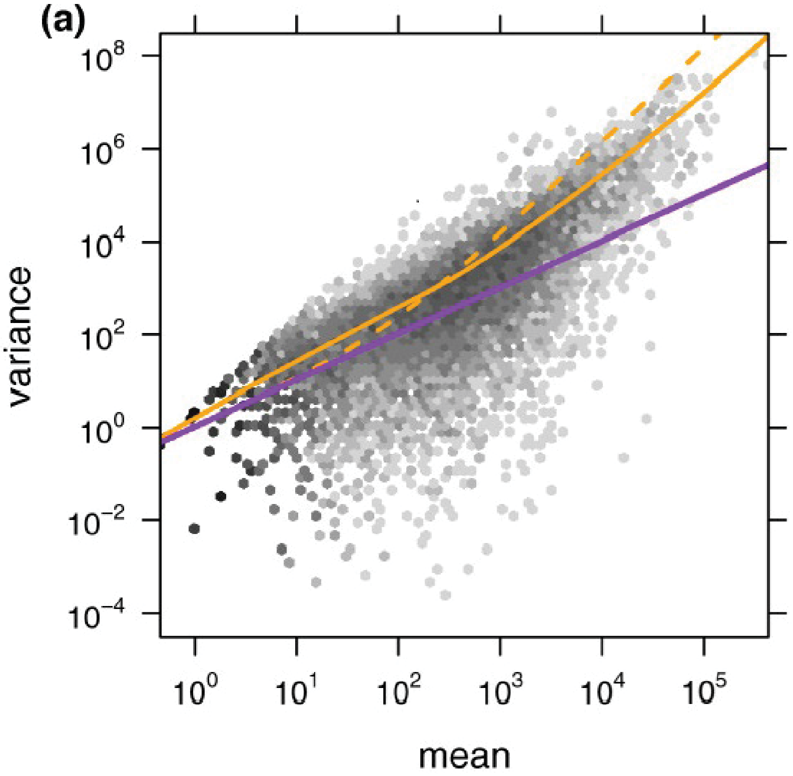


Figure 2.5: **Mean-Variance Relationship.** This is a mean-variance plot of an RNA-Seq data set consisting of biological replicates. Each dot represents the raw variance and mean of a particular gene. The purple line reflects the poisson model (variance=mean). The orange lines reflect estimates of the mean-variance trend assuming a negative binomial distribution. The poisson model clearly fails to recover the trend and is a sign that the data reflect overdispersion relative to the poisson. Illustration by Anders and Huber (2010)

Another artifact that has the potential to drive false positives among differential expression results is bit more subtle. The observed count for a gene not only depends on its own biological properties but also on those of the other genes. This is do to the fact that the genes are not being sampled independently, forcing them to share the same pool of sequenced reads from which to generate counts. More concretely, suppose a gene is truly differentially expressed where in one condition its expression is high relative to the other genes. Due to the high expression level, this gene soaks up a disproportionate amount of the available reads in each sample causing a down sampling of the other genes in that condition; in other words, they are effectively sequenced at a reduced depth. All else equal, if the gene is not highly expressed in the other condition then the down sampling effect is lessened resulting in an increase in sequencing depth for the other genes with the effect that they will now appear to be more differential. This sampling artifact is referred to as RNA composition and is more challenging to adjust for than simply the library size differences. In fact, many have demonstrated that adjustments based solely on library size, referred to as a total count normalization, perform poorly both for accurate estimation of relative expression level as well as in differential analyses, and several alternative approaches have been proposed (Bullard *et al.*, 2010; Robinson and Oshlack, 2010; Anders and Huber, 2010; Risso D and S, 2011; Hansen *et al.*, 2012), a few of which are presented below.

The approach used by the RNA-Seq differential gene expression method DESeq (Anders and Huber, 2010) attempts to correct for RNA composition as well as library size. It directly inputs the normalization factor into its statistical model and assumes that it is sample specific and linearly scales the mean of the counts and requires estimation:

$$\mu_{ig} = s_i \lambda_g$$

where s_i takes the place of the total count, t_i, as a more general normalization factor. The procedure is as follows:

1. For each gene g, find the geometric mean across the n samples:

$$m_g = \left(\prod_{k=1}^{n} y_{kg} \right)^{1/n}$$

2. For each sample i, calculate the ratio of each gene count, y_{ig}, to its respective geometric mean, m_g:

$$\mathrm{r}_i = (\mathrm{r}_{i1},\ \mathrm{r}_{i2},\ ...,\ \mathrm{r}_{ip}) = \left(\frac{y_{i1}}{m_g},\ \frac{y_{i2}}{m_g},\ ...,\ \frac{y_{ip}}{m_g} \right)$$

3. The per-sample normalization factor, s_i, is then estimated as the median of the sample's corresponding ratios:

$$\hat{s}_i = \underset{j=1,\ldots,p}{\mathrm{median}}\ \mathrm{r}_{ij} = \underset{j}{\mathrm{median}} \frac{y_{ig}}{\left(\prod_{k=1}^{n} y_{kj}\right)^{1/n}}, \quad \text{for } i = 1, ..., n.$$

Another approach introduced by Robinson and Oshlack (2010) used by the popular RNA-Seq differential gene expression method edgeR (Robinson and Smyth, 2007), is to use a weighted average of log-fold changes between samples after trimming the tails. The approach is given the abbreviation TMM which stands for Trimmed Mean of M values. Like DESeq, the goal of TMM is to also correct for RNA composition and library size. The log-fold change for a gene g between two samples i and j is defined as,

$$M_g = \log_2 \frac{y_{ig}}{N_i} - log_2 \frac{y_{jg}}{N_j}, \text{ where } N_k = \sum_{\ell} y_{k\ell}$$

The TMM method then proceeds as follows:

1. Choose a reference sample r.

2. Calculate M_{ig}^r for all genes g across all samples i to the reference sample r.

3. For each sample, identify the subset of genes, G_i^{mid}, whose corresponding log-fold change M_{ig}^r lies within the middle $(100 - x)\%$ (authors suggest setting $x = 30$).

4. Assign a weight to each gene for all samples:

$$w_{ig}^{r} = \begin{cases} \dfrac{N_i - y_{ig}}{N_i \cdot y_{ig}} + \dfrac{N_r - y_{rg}}{N_r \cdot y_{rg}}, & y_{ig},\ y_{rg} > 0 \text{ and } g \in G_i^{\text{mid}} \\ \\ 0, \text{ otherwise} \end{cases}$$

5. For each sample, calculate the normalization factor, s_i, using the trimmed weighted mean of M values:

$$\log_2(s_i) = \frac{\sum_g w_{ig}^r M_{ig}^r}{\sum_g w_{ig}^r}$$

Bullard et al. (Bullard *et al.*, 2010) propose two methods: upper-quartile normalization and quantile normalization the latter being an approach adapted from microarray data normalization (Bolstad *et al.*, 2003). Both of these also attempt to correct for RNA composition and sequencing depth and work by matching the distribution of the counts across lanes. The much simpler but effective upper-quartile method divides each count in a given sample by the count at the 75^{th} percentile of all non-zero counts in that sample. Once this is done for all samples the normalized data are re-scaled so that the sum of them across all samples equals the sum of the total counts from the original count matrix. The quantile normalization method equalizes the distributions of the counts in each sample to a reference distribution given by the median values of the counts with equal rank with respect to their samples. The procedure is as follows:

1. Start with a matrix of gene counts, y_{ig}, where each row i contains the counts for a sample across all p genes:

$$\mathbf{y}_i = (y_{ig_1},\ y_{ig_2},\ ...,\ y_{ig_p})$$

.

2. For each sample (or row) i, sort the genes and record their ranks:

$$\mathrm{y}_i^{\text{sort}} = (y_{ig_{(1)}},\ y_{ig_{(2)}},\ ...,\ y_{ig_{(p)}})$$
$$\mathrm{r}_i = (r_{ig_1},\ r_{ig_2},\ ...,\ r_{ig_p})$$

3. Find the p-vector of medians where each element is the median of the counts across the samples at a fixed rank:

$$\boldsymbol{\nu} = (\underset{i}{\text{med}}\ y_{i1}^{\text{sort}},\ \underset{i}{\text{med}}\ y_{i2}^{\text{sort}},\ ...,\ \underset{i}{\text{med}}\ y_{ip}^{\text{sort}}) = (\underset{i}{\text{med}}\ y_{ig_{(1)}},\ \underset{i}{\text{med}}\ y_{ig_{(2)}},\ ...,\ \underset{i}{\text{med}}\ y_{ig_{(p)}})$$

.

4. For each sample i create the p-vector of normalized counts, $y_{ig}^{'}$, by replacing the original count with the its corresponding median:

$$\mathbf{y}_i^{'} = (y_{ig_1}^{'},\ y_{ig_2}^{'},\ ...,\ y_{ig_p}^{'}) = (\boldsymbol{\nu}_{\mathrm{r}_{ig_1}},\ \boldsymbol{\nu}_{\mathrm{r}_{ig_2}},\ ...,\ \boldsymbol{\nu}_{\mathrm{r}_{ig_p}})$$

.

A recent paper comparing several additional normalization methods in a differential gene expression setting has found the above methods to have similar performance with the top performers being the DESeq method and TMM, and the upper-quartile method also doing quite well (Dillies *et al.*, 2013). This paper further supports the notion that RNA composition is a true problem in RNA-Seq experiments, and methods that only take into account library size corrections for between sample normalization, like total count normalization and RPKM, fail to address the issue adequately.

2.5 Shrinkage Techniques for Sequencing Data

In the marginal testing approach described in the previous section, each feature is modeled with its own set of parameters and can be estimated and tested independently. In the typical design, this includes both a mean and variance parameter. For many genomic experiments, it is common to expect a very large fraction of genes to not be differentially expressed. And due to the sheer size of these experiments, usually consisting of thousands to tens of thousands of features, it is reasonable to also expect that a fraction of them will be sampled in such a way that greatly under-represents their true variability, driving their test statistics upward and ultimately leading to false positives. The degree of the risk is largely a question of sample size which is frequently on the low end (10 or less) for genomic experiments. For RNA-Seq count data in particular, a lot of effort has been put toward finding the best approach for mitigating this risk in low sample sizes, yet they all work under the same paradigm: sharing information across the features. This concept was first introduced by Lönnstedt and Speed (2002) in the context of continuous log-intensities in microarray data and refined for practical implementation by Smyth (2004, 2005) in the popular limma package. It is worth demonstrating this microarray method for comparison to RNA-Seq based methods that work on the same principle.

2.5.1 Limma: Linear Models for Microarray Data

Limma first sets up a standard linear model for each gene:

$$\mathrm{E}(\mathbf{y}_g) = \mathbf{X}\boldsymbol{\beta}_g$$

and makes the following standard, normal linear model assumptions for the estimate of the β and variance parameter for each j:

$$\hat{\beta}_{gj}|\beta_{gj}, \sigma_g^2 \sim \mathrm{N}(\beta_{gj}, \nu_{gj}\sigma_g^2)$$

and,

$$s_g^2|\sigma_g^2 \sim \frac{\sigma_g^2}{d_g}\chi^2_{d_g}$$

where s_g^2 is the sample variance, d_g is the residual degrees of freedom and ν_{gj} is obtained from the j^{th} diagonal of the inverse of the covariance matrix. The following prior distributions are then assumed for β_{gj} and σ_g^2:

$$\beta_{gj}|(\sigma_g^2, \beta \neq 0) \sim \mathrm{N}(0, \nu_{0g}\sigma_g^2)$$
$$\sigma_g^2 \sim s_0^2(\chi_{d_0}^2/d_0)^{-1},$$

These priors are conjugate to the normal implying the posterior distribution of σ^2 is proportional to,

$$\chi^2 - \text{Inverse}\left(d_0 + d_g, \frac{d_0 s_0^2 + d_g s_g^2}{d_0 + d_g}\right)$$

yielding the following equation for the posterior mean:

$$\widetilde{s}_g^2 = \left[\mathrm{E}\left(\sigma_g^{-2}|s_g^2\right)\right]^{-1} = \frac{d_0 s_0^2 + d_g s_g^2}{d_0 + d_g}$$

An empirical bayes approach is used to estimate the prior parameters from the marginal distribution of s_g^2. This allows for new shrunk estimates of the sample variance toward that given by a global consensus. The estimate, $\tilde{s}_g$, is then substituted for the sample variance in the standard t-statistic producing a shrunken t-value for each gene.

2.5.2 Transformations

Given the strong performance, mathematical convenience, and the sheer amount of theoretical results provided by normal models, in addition to the existence of an already proven method, limma, that implements shrinkage of the variance parameter in the normal setting for microarray data (section 2.5.1), two RNA-Seq approaches transform the counts to obtain a new set of data to which limma can be applied. The R packages DESeq (Anders and Huber, 2010) and limma (Smyth, 2005) provide an option to carry out such transformations to the data. Note, DESeq merely provides a variance stabilizing transformation for the negative binomial and is not part of the DESeq method for testing differential expression. A variance stabilizing transformation combined with a normal linear model is a very commonly used approach to circumvent the analysis of count data directly. The corresponding method provided by limma is called voom (Law *et al.*, 2014) and is a more sophisticated approach in this setting.

DESeq transforms the counts by using a variance stabilizing transformation in the context of its own dispersion shrinkage approach. Specifically, it assumes the count data is distributed negative binomial and under this model finds shrinkage estimates of the dispersion parameter using the DESeq method. Then, the variance stabilizing transformation for the negative binomial distribution is either analytically or numerically applied depending on whether a parametric or nonparametric dispersion modeling technique was used. When this is done the transformed data is directly run through the limma pipeline to obtain differential expression results.

The voom method normalizes the counts for sequencing depth and then works with the log of these normalized counts. The transformed data is given the name log counts per million (log-cpm for short) which refers to the way in which the transformation is done and is given by,

$$r_{gi} = log_2\left(\frac{y_{gi} + 0.5}{R_i + 1} \times 10^6\right)$$

where R_i is the library size of sample i and y_{gi} is the count for gene g of sample i. Since this is not a variance stabilizing transformation across the full range of observed counts, there still remains a mean-variance trend of the log-cpm values, especially at lower count values, and therefore needs to be incorporated into the analysis. To do this, first a per-gene, ordinary least-squares fit of the log-cpm values across the samples to the experimental design is used to obtain fitted means, $\hat{\mu}_{ig}$, for each observation and a per-gene residual standard deviation $\hat{s}_g$. Next, the log-cpm values for each gene are converted to a log count via,

$$t_g = \bar{r}_g + log_2(S + 1) - log_2(10^6) \tag{2.1}$$

where R is the geometric mean of the library sizes. Each t_g is then paired with its corresponding square-root of the residual standard deviation, $\hat{s}_g^{0.5}$. The collection of t_g and $\hat{s}_g^{0.5}$ across all genes is modeled nonparametrically using a loess curve. However, instead of simply assigning a per-gene shrunken estimate, $\hat{s}_g^{0.5}$, given by the fitted curve evaluated at t_g, the voom method obtains per-observation predictions, $\hat{s}_{ig}^{0.5}$, by first transforming each *fitted* log-cpm value by replacing $\bar{y}_g$ with $\hat{\mu}_{gi}$ in 2.1 which yields fitted log counts, $\hat{t}_{ig}$. These are subsequently plugged into the loess curve to obtain the desired predicted values for each observation which are then used as weights within the normal linear model fit to the log-cpm values using limma. The justification for the per-observation predictions is that even though two observations of a gene may have the same log-cpm value, they may originate from completely different count sizes due to different sequencing depths across the samples, implying that each log-cpm has its own intrinsic variability.

2.5.3 Bayes & Empirical Bayes Methods

A very common approach to shrinkage for count data follows that of limma and places a prior distribution on the dispersion parameter of an overdispersed count distribution. Unlike in the case of limma, however, such distributions do not generally correspond to known conjugate priors and marginal distributions for the data, and numerical methods are necessary to estimate the hyperparameters and the posterior mean of the dispersion. Consequently, due to the large amounts of data common in RNA-Seq experiments, the run times of these methods are quite long and in some situations simplifying assumptions and/or ad hoc procedures are used to ease computation that would otherwise be avoided in the presence of a more mathematically convenient model. Furthermore, the posterior distribution is often a function of the mean parameter in addition to the data and therefore requires a plug-in estimate adding imprecision to the final dispersion estimate.

The method DSS (Wu *et al.*, 2013) assumes the usual negative binomial distribution for the count data and places a log-normal prior on the dispersion parameter, ϕ:

$$y_{gi} \sim NB(\mu_{ig}, \phi_g)$$
$$\phi_g \sim \log - \mathrm{normal}(\mathrm{m}_0, \tau^2).$$

The resulting posterior distribution is not a common one and therefore the posterior mean would need to be calculated numerically. However, doing this for each gene would be too computationally intensive and instead they use the posterior mode. An empirical bayes approach is used to estimate the hyperparameters, m_0 and τ. These parameters are assumed shared by all genes and therefore the entire set of data is used in the estimate. Specifically, they are estimated from the empirical distribution of an initial per-gene point estimate of ϕ for each gene combined with an ad hoc approach that simulates pseudo-data from a specific model and uses the data to arrive at final estimates of the prior parameters.

EBSeq, baySeq and ShrinkSeq are other empirical bayes approaches that both use the negative binomial model as a starting point and place a prior on the dispersion (Hardcastle and Kelly, 2010; Leng *et al.*, 2013; Van De Wiel *et al.*, 2013). In the case of baySeq, posterior probabilities for a differential expression model and a non-differential expression model are estimated for each gene. This is done by using numerical methods to estimate an empirical prior distribution using all the data. This is then used along with the negative binomial assumption to obtain the posterior probabilities. Note, this creates a ranking of genes based on their probability of differential expression as opposed to a p-value. Similarly, EBSeq calculates posterior probabilities for differential expression but places a Beta prior on the dispersion parameter of a negative binomial where the hyperparameters are estimated via the Expectation-Maximization algorithm. ShrinkSeq allows for fitting over various generalizations of the Poisson distribution that includes the negative binomial and places priors on both the mean and dispersion parameters, using the same prior on the mean as Lönnstedt and Speed (2002) and a nonparametric prior for the dispersion. The resulting posterior distributions require numerical methods to obtain estimates of the desired parameters.

2.5.4 Regression Based Approaches

Using regression to obtain shinkage estimates of the dispersion parameter is a simpler, less theory laden approach and is the basis of several differential gene expression methods for RNA-Seq data. Though the specific details of the regression technique used differs between methods, they all essentially model raw dispersion estimates as a function of other aspects of the data. In a parametric approach, additional parameters are used to relate a summary of the data for a gene to its dispersion. These parameters are assumed shared by all genes and are therefore estimated via regression by looking across all genes. Nonparametric approaches do not specify shared parameters but attempt to infer the relationship directly using data from all genes. Once the parametric or nonparametric form is estimated, the shrinkage of the initial dispersion estimates results by replacing the individual dispersion estimates with that given by the fitted curve. This can be considered total shrinkage to the estimate of the population's pattern of variance.

The first method to introduce this approach for RNA-Seq data was DESeq (Anders and Huber, 2010). They assume the counts come from a negative binomial distribution which implies the following mean-variance relationship:

$$E(y_{gi}) = \mu_{gi} = \lambda_{g\rho_i} s_i$$

$$V(y_{gi}) = \mu_{gi} + s_i^2 \phi_g \lambda_{g\rho_i}^2 = \mu_{gi} + s_i^2 \nu_{g,\rho_i}$$

where ρ_i is the condition of sample i and s_i is the normalization factor for sample i. DESeq makes the assumption that $\nu_{g,\rho}$ is a smooth function of of the mean parameter, $\lambda_{g\rho_i}$, and allows for both a parametric and a nonparamtric approach to estimating this function. In the nonparametric setting a localized regression is fit across the genes between raw estimates of ν and the mean of the normalized counts across the samples for each gene. In the parametric setting the dispersion and mean are related via,

$$\phi(\lambda) = \frac{a_1}{\lambda} + a_0$$

where a_1 and a_0 are parameters shared by all genes. Using initial estimates for ϕ and λ the parameters are estimated via regression. Once the parameters, a_0 and a_1, are estimated, the function is used to assign shrunken dispersion estimates per-gene using the raw mean of the normalized counts.

Another parametric regression approach is used by the method BBSeq. A beta-binomial model is assumed for the counts resulting in the following mean variance relationship:

$$E(y_{gi}) = \mu_{gi} = s_i p_g$$
$$V(y_{gi}) = s_i p_g (1 - p_g) \left(1 + \phi_g (s_i - 1)\right)$$

where s_i is the library size and ϕ is the dispersion parameter and lies between 0 and 1. A beta-binomial likelihood is maximized for each gene separately to obtain MLE estimates $\hat{\eta}_{pk} = \text{logit}(\hat{p}_{gk})$ and $\hat{\phi}_g$ for each condition k. Using these estimates, the coefficients for the cubic polynomial below are estimated via linear regression:

$$\psi = \text{logit}(\phi) = \gamma_0 + \gamma_1 \bar{\eta}_{g\cdot} + \gamma_2 \bar{\eta}_{g\cdot}^2 + \gamma_3 \bar{\eta}_{g\cdot}^3.$$

Once estimated, the cubic function is plugged into the beta-binomial likelihood function in place of the dispersion parameter and the resulting profile likelihood for p_{gk} for all groups k is maximized for each gene separately.

2.5.5 Conditional Likelihood

Bayesian models do not always result in a posterior distribution for the dispersion independent of the mean parameter, especially for count distributions, and require plug-in estimates for the mean to obtain dispersion estimates; see, for example, the methods DSS and DESeq2 (Wu *et al.*, 2013; Love *et al.*, 2014). This is not ideal as it induces further uncertainty. Conditional likelihood is conceptually a more desirable alternative since it can extract information about the dispersion parameter of a gene independently of its mean and therefore facilitates the combination of this information across genes with differing mean parameters without requiring estimates of them. The first use of conditional likelihood for RNA-Seq data was by Robinson and Smyth (2007) in their edgeR method and was applied to the dispersion parameter of the negative binomial and has been adapted to generalizations of the negative binomial distribution by Di *et al.* (2011) in their NBPSeq method.

A conditional likelihood is available when the joint likelihood of random variables can be written as a product of a conditional distribution that depends only on one of the parameters (the parameter of interest) and a marginal distribution that depends on all parameters. Therefore, in place of the full likelihood that depends on all parameters, we can use the conditional likelihood to estimate the parameter of interest alleviating the need to simultaneously estimate other nuisance parameters. For example, the conditional likelihood is theoretically available for the sufficient statistics of all exponential family distributions: Suppose (v, w) are the sufficient statistics for the canonical parameters (θ, η) in an exponential family distribution, then the joint likelihood of the sufficient statistics can be written as,

$$p_{\theta,\eta}(v, w) = p_\theta(v|w)p_{\theta,\eta}(w).$$

In other words, the distribution obtained from conditioning on the sufficient statistic for the nuisance parameter *only* depends on θ and works simply on the basis of sufficiency. Since the conditional distribution is truly a probability density, the estimate of θ can be estimated via maximum likelihood. Very few distributions are such that the conditional likelihood can be written down even if it theoretically exists since it relies on knowing the marginal distribution of the sufficient statistic for the nuisance parameter which is not generally available and is a major hurdle to implementing this method.

An example of its use is the method edgeR. It first assumes the counts for each gene follows a negative binomial distribution. Since we are focused on shrinking the dispersion parameter, the mean is considered the nuisance parameter in this step. A well known property of the negative binomial is that the sum of negative binomial random variables is also negative binomial provided the samples are i.i.d.. Since the sum is the sufficient statistic for the mean parameter, we are able to write down the exact conditional likelihood of the dispersion parameter. However, count data in RNA-Seq experiments are not i.i.d. since each sample is sequenced at different depths. To circumvent this issue, edgeR uses an ad hoc procedure to generate pseudo counts that are approximately i.i.d. and are used in place of the observed data. Then an exact conditional likelihood, $\ell_g(\phi_g)$, is formed for the dispersion parameter, ϕ_g, of each gene which results in a likelihood absent of the mean parameter.

2.5.6 Weighted Likelihood Method

Using the conditional likelihood theory presented in the previous section, to induce shrinkage edgeR combines the conditional likelihoods across all genes to form a global likelihood which is taken to be their sum and is parameterized by a single global dispersion:

$$\ell_{\text{global}}(\phi) = \sum_{g=1}^{G} \ell_g(\phi) \tag{2.2}$$

By maximizing the global likelihood with respect to the single parameter ϕ shared by all genes, a global estimate of the dispersion is obtained and serves as a shrunken estimate with respect to the individual dispersion parameters for each gene. Though this provides stability and greatly reduces sampling variability caused by low sample sizes, the complete shrinkage of the dispersion parameter to the global estimate may be an over correction leading to a large bias and therefore may not provide optimal performance. Furthermore, the sampling variability is reduced naturally through an increase in sample size and so it stands to reason that there is less need to rely on additional sources of information as more samples are made available and the amount of shrinkage should reflect that. For this purpose, edgeR implements what is called weighted likelihood that serves to control the degree of shrinkage.

The strategy of weighted likelihood is attributed to Wang (2006) and the authors of edgeR adapted it to the differential gene expression setting. A weighted likelihood is created for each gene and is specifically a weighted sum of the individual likelihood of the gene and the global likelihood:

$$WL_g(\phi_g) = \ell_g(\phi_g) + \delta \frac{1}{G} \sum_{j=1}^{G} \ell_j(\phi_g)$$

where $\delta >= 0$ controls the amount of shrinkage. A value of 0 corresponds to no shrinkage, a value of ∞ corresponds to full shrinkage to the global estimate, and a positive value is a compromise between the two. One caveat is that in general there is not a clear way of estimating the nuisance parameter δ, and so edgeR provides a rule of thumb that depends on the sample size and number of groups and is equal to $20/(n - K)$ where n is the sample size and K is the number of conditions.

2.5.7 Shrinkage of the log-Fold Change

Though most of the effort has been put toward providing shrinkage of the dispersion estimate, it is not necessarily the most relevant parameter. In RNA-Seq data, it is common to see many genes with a low log-fold-change (LFC) among the top calls when using p-values as ranks and are generally considered uninteresting findings. Therefore, ranking by p-values in this setting does not necessarily correlate with a ranking based on biological significance. Conversely, using the raw estimates of LFC in order of largest to smallest would not provide a useful ranking in terms of statistical significance without taking into account the variability especially in low count genes. DESeq2 (Love *et al.*, 2014) proposes to shrink the LFC for each gene toward zero by taking into account other properties of the data, thereby creating a more stable and more relevant ranking of the genes. The empirical bayes method for shrinking the LFC's closely parallels the method DSS described above for shrinking the dispersion. A zero-centered normal prior is placed on the LFC's and the empirical distribution of the raw MLE estimates across all genes is used to estimate the variance parameter in the prior. Once this is done, the posterior mode of the LFC is found via numerical methods for each gene.

2.5.8 Other Differential Gene Expression Methods

Not all differential methods necessarily fit into the above categories. Nonparametric resampling strategies such as SAMSeq and NOISeq (Li and Tibshirani, 2013; Tarazona *et al.*, 2011) essentially permute the samples and test for differences to build up a distribution of the noise present in the data which can then be compared to the true grouping of the samples to determine how alike the observed grouping is to the noise distribution. These methods generally require a larger sample size than parametric approaches in order to perform adequately. There are also more ad hoc shrinkage procedures like sSeq (Yu *et al.*, 2013) that obtain a crude estimate of dispersion for each gene using the method of moments and shrink it toward a single, global constant that is found by minimizing the MSE across all genes.

Other approaches such as TSPM and PoissonSeq (Auer and Doerge, 2011; Li *et al.*, 2012) stick with the standard log-linear linear models but adapt them in some way to be more suitable for RNA-Seq data without applying shrinkage. In the case of Poissonseq, this includes a novel normalization technique as well as a power transformation to handle overdispersion in the data. The method TSPM which stands for a two stage poisson model first fits a random effects model to each gene separately and uses the theory behind random effect models to evaluate the hypothesis that the gene does not exhibit overdispersion relative to the Poisson. For genes that fail to reject the hypothesis according to some threshold, a Poisson log-linear model is fit, and for genes that reject the hypothesis a quasi-poisson likelihood is used; and under each of their respective models, each gene is tested separately for differential expression.

2.6 Differential Exon Usage

Though much effort has been put toward developing proper statistical techniques for differential gene expression analyses, only a few methods have attempted to carry over the same strategies to other differential testing settings such as alternative splicing in the form of exon usage or isoform expression. These analyses provide further insight into the transcriptional activity beyond that of a gene-level analysis but also present additional challenges in the statistical modeling.

Alternative splicing facilitates the production of different mRNA molecules, called isoforms, from a single gene, each consisting of a different combination of its exons (for a detailed description see section 2.2). Recall from Chapter 1, one approach to identifying differential splicing would be to measure the number of reads including the exon and compare it to the number explicitly excluding the exon (Shen *et al.*, 2012; Wu *et al.*, 2011), in which case differences in exon usage appear as a question of comparing proportions across conditions. In such a setting, the proportions potentially span the full 0-1 range invalidating the poisson approximation to the binomial widely implemented in the context of differential gene expression analysis, and therefore other approaches such as using overdispersed binomial models are needed. Unfortunately, overdispersed binomial distributions have not proven to be as popular primarily due to the fact that they are more difficult to work with than poisson based models and are therefore in short supply; however, the undeniable relevancy of genomic questions beyond that of gene expression have motivated their further development.

The methods developed here are specifically designed to handle such data and a complete description as well as results from their application to exon usage can be found in chapters 3 and 4, respectively. Competing methods like MATS and SpliceTrap (Shen *et al.*, 2012; Wu *et al.*, 2011) also use the inclusion/exclusion approach, while others like DEXSeq and DE-FPCA (Anders *et al.*, 2012; Xiong *et al.*, 2014) analyze the exon counts directly and fit within the same statistical framework as gene expression.

Within the specific setting of detecting differential alternative splicing, there are other approaches of detection in mRNA-Seq data besides even the two we explore here (exon counts and inclusion/exclusion counts). Exon inclusion counts may not be the most appropriate for every setting. In particular, there are many methods for estimating the expression levels of individual isoforms (Denoeud *et al.*, 2008; Jiang and Wong, 2009; Trapnell *et al.*, 2010; Richard *et al.*, 2010; Salzman *et al.*, 2010), and comparison of the isoform levels may give more insight into alternative splicing particularly when there is a great deal of information about the transcriptome that is being sequenced. However, in our experience there are still many cases where researchers find themselves without a well constructed annotation of the transcriptome, and often rely on de-novo methods to construct genes and/or transcripts (Trapnell *et al.*, 2010; Guttman *et al.*, 2010) in an effort to understand the use of alternative splicing as a means of cell regulation. This is an extremely complicated problem, and these de-novo methods can be unreliable and unstable if used on a single, small experiment or without significant depth. In contrast, inclusion/exclusion counts rely on detection of exons and splice sites, which are much simpler problems. Such inclusion/exclusion counts still provide useful, interpretable information about the undergoing of alternative splicing within the organism and our methods give a reliable technique for the statistical analysis of such data.

2.6.1 Statistical Methods for Exon Usage Analysis

MATS (Shen *et al.*, 2012) uses the same inclusion/exclusion approach laid out above. They assume a non-conventional hierarchical model for the observed proportions that serves to add additional variation to a standard binomial distribution:

$$y_{gi}|p_g \sim \mathrm{Bin}(n_{gi}, p_{gc_i})$$
$$(p_{gc_1}, p_{gc_2}) \sim \mathrm{MultiVarUniform}\left(0, 1, cor = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}\right)$$

where c_k correspond to group k and ρ governs the correlation between the proportions across the two groups and is considered shared across all exons. This global parameter is estimated from using all the data but is not associated with a per-exon analogue and therefore MATS is not technically performing shrinkage on a parameter. Nevertheless, due to being estimated from the full set of data the resulting estimate of ρ does not substantially increase variability into the method for low sample size settings.

Another approach to exon usage was developed by Anders *et al.* (2012) in their DEXSeq method. They set up a gene-level model that incorporates a parameter for an exon effect that can be tested for a difference from zero. Specifically, the counts within an exon are assumed to be distributed according to a negative binomial distribution:

$$y_{gil} \sim NB(s_j \mu_{gil}, \phi_{gl})$$

for sample i of exon l in gene g. The following generalized linear model for the mean parameter of each exon and sample is fit separately for each gene:

$$log(\mu_{gil}) = \beta_g^G + \beta_{gl}^E + \beta_{gi}^S + \beta_{g\rho_i l}^{EC}$$

where β_g^G accounts for the overall gene expression for gene g, β_{gl}^E accounts for the fraction of reads from gene g that overlap with exon l, β_{gi}^S accounts for differences between samples in overall gene expression, $\beta_{g\rho_i l}^{EC}$ is the effect that condition ρ_i has on the fraction of reads that fall into exon l. A nonzero value for $\beta_{g\rho_i l}^{EC}$ represents differential usage between conditions of exon l and therefore is the parameter of interest. In order to fit the model an estimate of the dispersion parameter in the negative binomial needs to be found for each exon. The authors take the same approach as edgeR and use the conditional likelihood method to obtain maximum conditional likelihood estimates of the dispersion, $\hat{\phi}_{gl}$, for each exon of each gene. For the purpose of shrinking the estimates of dispersion, the following parametric relationship between the dispersion and the mean parameter is assumed, the parameters of which are considered shared across all exons:

$$\phi(\mu) = a_0 + \frac{a_1}{\mu}.$$

To estimate the global parameters a_0 and a_1 a linear regression is performed on the graph given by each exon's normalized mean count and its corresponding estimate of dispersion, $\hat{\phi}_{gl}$. The normalized mean count is then plugged into back into the parametric function after substituting the estimates for a_0 and a_1 into the equation to obtain the final shrunken dispersion estimates.

The extra-binomial method, EB2, developed by Yang *et al.* (2012) was designed for testing differences in allele frequencies between conditions. Like MATS it also implements a binomial model but instead of assigning a true probability distribution to the proportion parameter, p_g, of the binomial, a quasi-likelihood is used and only specifies the first two moments:

$$y_{gi}|p_g \sim \text{Bin}(n_{gi}, p_g)$$
$$E(p_g) = \alpha_g, \quad V(p_g) = \phi_g \alpha_g (1 - \alpha_g)$$

This formulation results in a mean-variance relationship for the marginal of y_{gi} equal to that of the beta-binomial and is given by,

$$V\left(\frac{y_{gi}}{n_{gi}}\right) = \frac{\alpha_g(1-\alpha_g)}{n_{gi}}\left(1 + \phi_g(n_{gi} - 1)\right)$$

To obtain shrinkage estimates of the variance the following parametric form is assumed with global parameters, a and b:

$$V\left(\frac{y_{gi}}{n_{gi}}\right) = \alpha_g(1-\alpha_g)\left(\frac{a}{s} + \frac{b}{n_{gi}}\right)$$

where s is a known exact value depending on other aspects of the experiment. Raw estimates of the variance and α_g are obtained and linear regression is performed using all the data to obtain estimates of a and b and are substituted back into the equation to obtain predicted values of the variance evaluated at $\hat{\alpha}_g$ for each g.

Though the EB2 method was developed specifically for testing allele frequencies, because it is capable of handling proportions directly it can also be applied to differential exon usage, as can any binomial based model including the gene expression method BB-Seq which was presented in section 2.5.4.

Chapter 3

Shinkage Methods within the Double Exponential Family of Distributions

3.1 The Double Exponential Family

We focus our methods on the double exponential family (Efron, 1986), also known as the exponential dispersion model (Jorgensen, 1987), which is a probability model that results in estimates closely related to the quasi-likelihood method. This class of distributions, which we will describe in detail below, adds a dispersion parameter to any member of the exponential family. This distribution has the advantage of being closely related to the quasi-likelihood approach and yet still provides a likelihood platform for shrinkage methods. Furthermore, the distribution is itself in the two-parameter exponential family of distributions, making calculations and approximations straightforward.

In what follows, the data consists of two $n \times p$ count matrices, Y and $\mathcal{M}$. Each y_{ig} entry is the counts for inclusion of the event g for sample *i* and m_{ig} gives the total possible number of counts related to event g. For the setting of exon inclusion, y_{ig} would be the number of reads including or overlapping exon g and m_{ig} would be the total number of reads either expressing exon g or skipping exon g. In a gene expression analysis, y_{ig} would be the number of reads mapped to gene g and m_{ig} would be the effective library size for sample g and is the same value for all genes in the sample. The value y_{ig}/m_{ig} is the standard binomial estimate of the proportion of inclusion of the event for the sample. For the purpose of remaining general, we will refer to the features (i.e. exons, isoforms, genes) as “events”. The m_{ig} terms will often be referred to as the total count or offset. We will also provide examples of this family of dispersion models in the form of the double binomial in the following paragraphs to aid in understanding, though the results remain fully general and can be applied to any exponential family distribution. In this section, we will focus on the modeling of just a single event, and we will drop the subscript g when the meaning is clear.

Assume our initial exponential distribution is given in canonical form by

$$g_{m_i}(Z_i) = \exp(m_i(\eta Z_i - \psi(\mu)))dG_{m_i}(Z_i),$$

where $\mu = E(Z_i)$ and $\eta = \eta(\mu)$ is the link function relating the canonical parameter and the mean. For the case of binomial, the link function is the standard logit function,

$\eta(\mu) = \log\left(\frac{\mu}{1-\mu}\right)$ and the normalizing function ψ is given by $\psi(\mu) = -\log(1-\mu)$. Note that we follow the notation of Efron (1986) so that the parameterization is such that $E(Z_i) = \mu$, implying for the binomial distribution that Z_i is the proportion y_i/m_i.

Then the dispersed density, with dispersion parameter ϕ is given by

$$\frac{c(\mu, \phi, m_i)}{\sqrt{\phi}} \exp\left\{\frac{Z\eta_\mu - \psi(\mu)}{\phi}\right\} dF_{m_i}(Z_i), \tag{3.1}$$

where $c(\mu, \phi, m_i)$ is a normalizing constant. The role ϕ is reminiscent of the role of the variance parameter in a normal distribution where $\phi > 1$ implies over-dispersion and $\phi < 1$ implies under dispersion. The resulting variance of Z_i is approximately $\phi\frac{V(\mu)}{m_i}$, where $V(\mu)$ is the variance function for the corresponding (non-dispersed) exponential family ($V(\mu) = \mu(1-\mu)$ in the case of the binomial).

The double exponential distribution can be reparameterized to be a member of the two-dimensional exponential family, and we work with the distribution in its canonical exponential form,

$$\exp(m_i(\lambda Z_i + \theta U_i - A_i(\theta, \lambda)))dF_{m_i}(Z_i)$$

where $\theta = 1/\phi$, $\lambda = \eta/\phi$, and

$$A_i(\theta, \lambda) = \psi(\mu)\theta - \frac{1}{2m_i}\log\theta - \frac{1}{m_i}\log(c(\mu, \theta, m_i)).$$

The sufficient statistics of (λ, θ) are given by (Z_i, U_i) where $U_i = \psi(Z_i) - Z_i\eta(Z_i) = \rho(Z_i)$. The function $\rho(x) = \psi(x) - \eta(x)x$ is determined by the specific exponential distribution. In the case of the binomial,

$$\rho(x) = -\log(1-x) - x\log(\frac{x}{1-x}) = -(x\log x + (1-x)\log(1-x)).$$

For the binomial, $\rho(x)$ is defined on [0,1], with $\rho(0) = \rho(1) = 0$, so it is well defined for all values of Z_i.

3.1.1 Differential Expression Setup

We are interested in the case where there are covariates that results in different μ for different samples i, $\eta_i = x_i^T\beta$ with $\beta \in R^q$. We focus on the common case in genomic studies where x_i defines K separate groups. Then we have a separate η_k for each group k, and the joint likelihood can be written as

$$\exp\left\{M\left(\sum_k \lambda_k \frac{M_k}{M} T_k + \theta U - A(\theta, \lambda)\right)\right\} dF(Z_1, \ldots, Z_n) \tag{3.2}$$

where $M_k = \sum_{i\in k} m_i$, $T_k = \sum_{i\in k} \frac{m_i}{M_k} z_i$, the standard binomial estimate of the mean μ_k, and $U = \sum_i \frac{m_i}{M}\rho(Z_i)$. For convenience we can rewrite U in terms of the groups, $U = \sum_k \frac{M_k}{M} U_k$, where $U_k = \sum_{i\in k} \frac{m_i}{M_k}\rho(Z_i)$, and

$$\begin{aligned} A_k(\theta, \lambda_k) &= \sum \frac{m_i}{M_k} A_i(\theta, \lambda_k) \\ &= \psi(\mu_k)\theta - \frac{n_k}{2M_k} \log(\theta) - \frac{1}{M_k} \sum_{i \in k} \log(c_i(\mu_k, \theta)) \end{aligned}$$

Then the complete likelihood is given by

$$\exp\left(\sum_k M_k \lambda_k T_k + \theta \sum_k M_k U_k - \sum_k M_k A_k(\theta, \lambda_k) \right) dF(Z) \tag{3.3}$$

$$= \exp\left\{ M \left(\sum_k \frac{M_k}{M} \lambda_k T_k + \theta U - A(\theta, \lambda_1, \dots, \lambda_K) \right) \right\} dF(Z) \tag{3.4}$$

where

$$\begin{aligned} A(\theta, \lambda_1, \dots, \lambda_K) &= \theta \sum_k \frac{M_k}{M} \psi(\mu_k) - \frac{N}{2M} log\theta - \frac{1}{M} \sum_k \sum_{i \in k} \log(c_i(\mu_k, \theta)) \\ U &= \sum_k \frac{M_k}{M} U_k = \sum_i \frac{m_i}{M} \rho(Z_i) \end{aligned}$$

Now we have that

$$\begin{aligned} E(T) = \mu_T &= \frac{\partial A(\theta, \lambda)}{\partial \lambda} = \mu - \frac{1}{M} \sum_i \frac{\frac{\partial c_i(\mu, \lambda)}{\partial \lambda}}{c_i(\mu, \theta)} \\ E(U) = \mu_U &= \frac{\partial A(\theta, \lambda)}{\partial \theta} = \rho(\mu) - \frac{n}{2M\theta} - \frac{1}{M} \sum_i \frac{\frac{\partial c_i(\mu, \theta)}{\partial \theta}}{c_i(\mu, \theta)} \end{aligned}$$

and the MLE for λ and θ are given by the solution to this system of equations, when U, T are substituted for $E(T), E(U)$.

3.1.2 Approximating $c(m, \mu, \theta)$

The normalizing constant $c(m, \mu, \theta)$ can be computationally expensive to calculate, especially in the context of genomic studies where the maximization routines will need to be calculated for every event. Efron gives a general approach for calculating approximations to the normalization constant and specifically calculates them for the binomial and Poisson case. In addition, he also shows that the normalizing constant approaches 1 at any fixed values of μ and θ as $m \to \infty$. We note that the approximation of 1 is generally good for larger values of m but is dependent on the proximity of μ and θ to the boundary of their respective parameter spaces such that m needs to be even larger as the parameters move toward the boundary to obtain the same level of approximation. In differential gene expression m is generally in the tens of millions but both μ and θ can be in the range of 10^{-3} to 10^{-5}. We find, however, that the normalization constant is still near 1 in such a setting with a mean distance from 1 across an entire RNA-Seq data set of 0.036 and a standard deviation of 0.050. In the exon usage setting, the approximation is not as good since m can be as low 20. However, the mean and dispersion parameters are generally further from the boundary than for gene expression data which improves the approximation. The median difference from 1 is 0.09 and the mean and standard deviation are both 0.17. For the methods developed here, we will rely on the approximation $c(m, \mu, \theta) = 1$ since it is more convenient in terms of both the mathematics and computation as well as proving to be a close approximation to the true value.

Maximizing the approximate joint likelihood with $c(m, \mu, \theta) = 1$ gives MLE estimates for μ and ϕ, $\hat{\mu}_k = T_k$ and

$$\hat{\phi} = \frac{2M\left(\sum_k \frac{M_k}{M}(\rho(T_k) - U_k)\right)}{n} = \frac{1}{n}\sum_k \sum_{i \in k} D(Z_i, \hat{\mu}_k) \tag{3.5}$$

where $D(Z_i, \hat{\mu}_k)$ is the deviance of Z_i from its estimated group mean $\hat{\mu}_k$. Thus the approximation $c(m, \mu, \theta) = 1$ results in the standard quasi-likelihood estimates of the dispersion based on deviance residuals, giving a likelihood based method that echoes the quasi-likelihood method (Efron, 1986).

3.1.3 Double Binomial Distribution

For data that follow a binomial process, the double binomial distribution can be used to model overdispersion relative to the binomial. Overdispersed binomial models are not utilized as often for analyzing RNA-Seq count data, mainly because of statistical and computational roadblocks. However, recall from section 2.4 that for some genomic analyses binomial models are preferred, if not necessary, as in the case of differential exon usage.

As a reminder, the standard binomial distribution in its familiar form is,

$$\binom{m}{mz} p^{mz}(1-p)^{m(1-z)}$$

where $z = y/m$. In canonical form is,

$$\binom{m}{mz} \exp\left\{m\left[z\log\left(\frac{p}{1-p}\right) - (-\log(1-p))\right]\right\}$$

implying,

$$\eta(p) = \log\left(\frac{p}{1-p}\right)$$
$$\psi(p) = -\log(1-p)$$

both of which carry over to the double binomial model using Eq. 3.1 and yielding the density for the double binomial as,

$$\frac{c(p,\phi,m_i)}{\sqrt{\phi}}\exp\left\{\frac{z\mathrm{log}\left(\dfrac{p}{1-p}\right)-(-log(1-p))}{\phi}\right\}\mathrm{d}F_m(z)$$

where $c(m,p,\theta)$ is the normalizing constant that is approximately equal to 1. Another approximation given by Efron is,

$$c(m,p,\theta)\approx 1+\frac{1}{12m}\frac{1-\theta}{\theta}\left(1-\frac{1}{p(1-p)}\right)\xrightarrow{m\to\infty}1.$$

For completeness, the full density can be written as,

$$f(z|m;p,\theta)=\theta^{0.5}\,c(m,p,\theta)\binom{m}{mz}\frac{(mz)^{mz}(m-mz)^{m-mz}}{m^m}\frac{m^{\theta m}p^{\theta mz}(1-p)^{\theta(m-mz)}}{(mz)^{\theta mz}(m-mz)^{\theta(m-mz)}}. \tag{3.6}$$

3.1.4 Double Poisson Distribution

Similarly, the double Poisson can be used as a substitute for the Poisson model in situations where the data exhibit overdispersion relative to the Poisson. Overdispersed Poisson models are frequently used in the context of RNA-Seq count data, mainly in the form of the negative binomial as discussed in the background chapter.

The standard poisson distribution in its familiar form is,

$$\frac{\mu^y e^{-\mu}}{y!}$$

and in canonical form,

$$\frac{1}{y!}\exp\{y\mathrm{log}(\mu)-\mu\}$$

implying,

$$\eta(\mu)=\log(\mu)$$
$$\psi(\mu)=\mu$$

Using Eq. 3.1 the double poisson density is,

$$\frac{c(\mu,\phi,m_i)}{\sqrt{\phi}}\exp\left\{\frac{y\mathrm{log}(\mu)-\mu}{\phi}\right\}\mathrm{d}F(y)$$

where $c(\mu,\theta)$ is the normalizing constant that is approximately equal to 1. Note, m does not play a role in a poisson distribution so it is taken to be 1. Another approximation given by Efron is,

$$\frac{1}{c(\mu,\theta)}\approx 1+\frac{1-\theta}{12\mu\theta}\left(1+\frac{1}{\mu\theta}\right)\xrightarrow{m\to\infty}1.$$

For completeness, the full density can be written as,

$$f(y|\mu,\theta) = \theta^{0.5}\ c(\mu,\theta)\frac{e^{-(\mu\theta+y)}y^y(\mu e/y)^{\theta y}}{y!}. \tag{3.7}$$

3.1.4.1 Library Sizes in the Double Poisson

As discussed in section 2.4.1, in RNA-Seq experiments each sample is sequenced at different depths and so the raw counts are not identically distributed and require an extra modeling parameter if modeling the counts directly. In the case of the double poisson, this can be easily achieved by assuming the mean for each event g and sample can be written as $\mu_{ig} = s_i\omega_g$ where s_i is the, possibly adjusted, library size for sample i. The full density for the event includes these extra s_i but all results still hold with only slight adjustments to the calculations.

3.1.5 Double Binomial Convergence to Double Poisson when $c(m,\mu,\theta) = 1$

In the gene expression setting the library sizes are very large and the proportion of reads allocated to each gene is very small. In the case of standard binomial, by assuming $mp \rightarrow \mu$ as $m \rightarrow \infty$ where m and p are the number of trials and probability of success in a binomial, respectively, and μ is the mean of the counts in a Poisson, then we have that the binomial converges in distribution to a Poisson. Similarly, the double binomial converges in distribution to a double poisson. This falls out in much the same way as in the standard case by realizing that for large n,

$$\binom{m}{y} \approx \frac{m^y}{y!},$$
$$(1-p)^{m-y} = (1-\frac{mp}{m})^{m-y} \approx e^{-mp}$$
$$(m-y)^{m-y} \approx m^{m-y}e^{-y}$$

Applying these approximations and replacing mp with μ and mz with y in 3.6 we obtain the double poisson density given in 3.7:

$$\begin{aligned}
\lim_{m\rightarrow\infty} f_{DB}(y|m;p,\theta) &= \lim_{m\rightarrow\infty} \theta^{0.5}\binom{m}{y}\frac{(y)^y(m-y)^{m-y}}{m^m}\frac{m^{\theta m}p^{\theta y}(1-p)^{\theta(m-y)}}{(y)^{\theta y}(m-y)^{\theta(m-y)}} \\
&= \theta^{0.5}\frac{\cancel{n^y}y^y\cancel{n^{n-y}}e^{-y}}{y!\cancel{n^n}}\frac{\cancel{n^{n\theta}}\mu^{\theta y}e^{-\mu\theta}}{\cancel{n^{\theta(n-y)}}e^{-y\theta}y^{y\theta}\cancel{n^{\theta y}}} \\
&= \theta^{0.5}\ \frac{e^{-(\mu\theta+y)}y^y(\mu e/y)^{\theta y}}{y!} = f_{DP}(y|\mu,\theta)
\end{aligned}$$

3.2 Conditional Likelihood for the Double Exponential Family

For all of the methods of shrinkage that we develop, we rely on the conditional likelihood of θ per event for the purpose of combining likelihoods across events independently of the their proportion μ (Robinson and Smyth, 2007). This provides a likelihood of the data that depends solely on θ and allows us the opportunity to shrink the estimates of θ across events independently of our estimates of μ. Namely, let $\hat{\theta}$ and $\hat{\mu}$ be the joint MLEs of θ, μ; then the conditional distribution

$$P_{\theta,\lambda}(\hat{\theta}|\hat{\lambda}) = \ell(\theta)$$

defines a likelihood of θ that is independent of μ because our distribution is a member of the exponential family (see 2.5.5 for more details). The exact conditional distribution for the double exponential is not tractable; however, we can approximate the conditional distribution using the modified profile likelihood (see Pawitan (2001) Chapter 10 for a review). The approximate conditional likelihood $\hat{\theta}|\hat{\lambda}$, where $\hat{\theta}, \hat{\lambda}$ are the MLE of the joint likelihood in equation 3.2, is given by

$$\begin{aligned}
\ell_{AC}(\theta) =& M(\sum_k \hat{\lambda}_{\theta k} \frac{M_k}{M} T_k + \theta U - A(\theta, \hat{\lambda}_\theta)) + \frac{1}{2} \log \left[\prod_k \left(\frac{\partial^2}{\partial \lambda_k^2} MA(\theta, \lambda)|_{\lambda = \hat{\lambda}_\theta} \right) \right] \\
=& M\theta \left(U - \sum_k \frac{M_k}{M} (\hat{\psi}_{\theta k} - \hat{\eta}_{\theta k} T_k) \right) + \frac{N}{2} \log(\theta) + \sum_k \sum_{i \in k} \log(c_i(\hat{\mu}_{\theta k}, \theta)) + \\
& \frac{1}{2} \sum_k \log \left(\frac{1}{\theta^K} M_k V(\hat{\mu}_{\theta k}) - f_c(\hat{\mu}_{\theta k}, \theta) \right) \\
=& \ell(\hat{\lambda}_\theta, \theta) + \frac{1}{2} \sum_k \log \left[\frac{\partial^2}{\partial \lambda_k^2} MA(\theta, \lambda)|_{\lambda = \hat{\lambda}_\theta} \right] \qquad (3.8)
\end{aligned}$$

where

$$\hat{\lambda}_\theta = \text{arg} \max_\lambda \ell(\lambda, \theta)$$

and $f_c(\mu_k, \theta)$ is given by the portion of the second partial derivative of $A(\theta, \lambda)$ involving the normalizing constants,

$$\frac{\partial^2}{\partial \lambda_k^2} MA(\theta, \lambda_1, \ldots, \lambda_K) = M_k \frac{V(\mu_k)}{\theta} - \sum_{i \in k} \left\{ \frac{\frac{\partial^2 c_i(\mu_k, \theta)}{\partial \lambda_k^2}}{c_i(\mu_k, \theta)} + \left(\frac{\frac{\partial c_i(\mu_k, \theta)}{\partial \lambda_k^2}}{c_i(\mu_k, \theta)} \right)^2 \right\}.$$

3.2.1 Approximation of Conditional Likelihood when $c(m, \mu, \theta) = 1$

Using the approximation $c(m, \mu, \theta) = 1$, the double exponential distribution implies a simple approximate form for the conditional likelihood of the sum of the deviance residuals. Let,

$$S = \frac{1}{2}\sum_k \sum_{i \in k} D(z_i, \hat{\mu}_k) = M\left(\sum_k \frac{M_k}{M}(\rho(T_k) - U_k)\right) \tag{3.9}$$

be half the total sum of the deviance residuals. Then, the conditional log-likelihood of $\hat{\theta}|\hat{\mu}$ in 3.8 is simplified as

$$\ell_{AC}(\theta) = -\theta S + \frac{n-K}{2}\log(\theta). \tag{3.10}$$

and the resulting conditional likelihood estimate of θ is,

$$\hat{\theta}_{AC} = \frac{n-K}{2S} \tag{3.11}$$

and in terms of $\phi = 1/\theta$ this becomes,

$$\hat{\phi}_{AC} = \frac{2S}{n-K} = \frac{1}{n-K}\sum_k \sum_{i \in k} D(z_i, \hat{\mu}_k). \tag{3.12}$$

Note, the MLE estimate of ϕ given by equation 3.5 is identical to that here in equation 3.12 except that n is replaced with just $n - K$. The change in the degrees of freedom from n to $n - K$ is a common result of using conditional distributions, see for example REML methods in random effects models.

Examining the approximation in 3.10 we see this implies that the conditional distribution of $\hat{\theta}|\hat{\mu}$ is approximately proportional to $\theta^{\frac{n-K}{2}} e^{-\theta S}$. Recall that the density of the Gamma distribution with a rate parameter, θ, and known shape parameter $(n - K)/2$ is,

$$P_{\text{Gamma}}\left(X; \frac{n-K}{2}, \theta\right) = \frac{X^{\frac{n-k}{2}-1}}{\Gamma\left(\frac{n-K}{2}\right)} \; \theta^{\frac{n-K}{2}} e^{-\theta X}.$$

Therefore,

$$P(S|T_1, \ldots, T_K, \theta) \propto \text{Gamma}\left(\frac{n-K}{2}, \theta\right),$$

where θ is the rate parameter of the Gamma (the distribution can be equivalently expressed with ϕ as the scale parameter). This result does not depend on the form of the underlying exponential family and therefore applies to any double exponential family distribution.

3.3 Shrinkage Methods for the Dispersion Parameter in the Double Exponential Family

In section 2.5 we presented three general approaches to providing shrinkage of dispersion parameters that are common in the literature: 1) modeling of the dispersion values as a function of other aspects of the data, 2) weighted likelihood shrinkage, and 3) empirical bayes shrinkage by estimating prior parameters from the marginal distribution of the data. We developed methods for the double exponential family for all three of these approaches to shrinkage. In what follows, we only present the weighted likelihood and the empirical bayes approaches. The empirical bayes estimator results in our DEB-Seq method, while the fusion of the weighted likelihood and empirical bayes results in the WEB-Seq method.

The first approach, which we do not present, performs a parametric regression of the estimated dispersion per event to other parameters in the model. In our implementation of this approach taking the BBSeq approach (Zhou *et al.*, 2011) but using the double binomial model, we found that in the exon inclusion setting the resulting p-values grossly failed to control the false discovery rate, and so we did not consider it any further.

3.3.1 Weighted Likelihood

As discussed in section 2.5, Wang (2006) gives a general strategy adapted by edgeR (Robinson and Smyth, 2007) for combining likelihoods across multiple datasets that relies on a likelihood equation that is a weighted combination of the likelihoods of all the experiments. We follow the same strategy to create a weighted likelihood using the approximate conditional likelihood $\ell_{AC}(\theta)$ for the double exponential setting. Again, if we assume that $c(\mu, \theta, m) = 1$ we get an enormous simplification and can analytically solve for $\hat{\theta}^{WL}$. Note that though ignoring the normalization constant is not necessary to implement the weighted likelihood approach, the typical size of RNA-Seq data makes calculating the exact normalization constant too costly in practice.

Recall the definition of S_g given by equation 3.9 as half the total sum of the deviance residuals in a double exponential family distribution, then for a specific event $g*$, we define its weighted likelihood as

$$\begin{aligned}
\ell_{g*}^{WL}(\theta) &= \ell_{g*}^{AC}(\theta) + \delta\frac{1}{p}\sum_{j=1}^{p}\ell_{g}^{AC}(\theta) \\
&= -\theta(S_{g*} + \delta\frac{1}{p}\sum_{g} S_g) + \frac{n-K}{2}(1+\delta)\log(\theta)
\end{aligned}$$

which gives that

$$\hat{\theta}_{g*}^{WL} = \frac{\frac{n-K}{2}(1+\delta)}{S_{g*} + \delta\bar{S}},$$

where $\bar{S} = \frac{1}{p}\sum_g S_g$.

The weight δ given to the global likelihood $\ell_{\text{global}}(\theta)$ (see equation 2.2) is a tuning parameter that must be chosen, and McCarthy *et al.* (2012) suggest it be chosen so that it is proportional to sample size adjusted by degrees of freedom, with the edgeR package assigning a fixed value for δ by default equal to $\frac{20}{n-K}$.

3.3.2 Empirical Bayes (DEB-Seq)

Empirical bayes estimation of the dispersion parameters via an explicit likelihood formulation is a natural way to provide shrinkage estimators of the dispersion parameter. By this we mean, formulate a Bayesian model $Y|\theta \sim F$ and $\theta \sim G_\alpha$ to get estimates of θ equal to the posterior mean, $E_\alpha(\theta|Y)$, and then choose a parameter α by estimating it from the marginal distribution of Y. Many distributions, including the double binomial, do not have a prior that gives a tractable form for the marginal distribution of Y to permit easy estimation of α from the data. However, if we make the approximation that the normalizing constant $c(m, \mu, \theta)$ in the distribution is equal to 1, we showed that there is a simple conditional distribution for the statistic S (defined as half the sum of the deviance residuals), namely that $S|(T, \theta)$ is approximately Gamma distributed with known shape parameter $(n - K)/2$, and rate parameter equal to θ.

Critically, the approximate distribution of $S|(T, \theta)$ is independent of the individual total counts, m_{ig}, per event and sample. It depends only on the total sample size n, and therefore is comparable across events with different total counts m_{ig}. This suggests a simple bayesian estimation approach for θ_g of event g. With known shape parameter, a conjugate prior for the rate parameter of a gamma distribution is itself a gamma distribution, $\theta_g \sim \Gamma(\alpha_0, \beta_0)$, suggesting a prior distribution that is i.i.d. across all events g. Then the posterior distribution of $(\theta_g|S_g, T_{\cdot g}, \alpha_0, \beta_0)$ is $\Gamma(\frac{n-K}{2} + \alpha_0, S_g + \beta_0)$ and estimation of θ_g can be given by the mean of this gamma distribution. Specifically, the full hierarchical model is,

$$(S_g|T_{1g}, ..., T_{Kg}, \theta_g) \sim \text{Gamma}\left(\frac{n-K}{2}, \theta_g\right)$$
$$(\theta_g|\alpha_0, \beta_0) \sim \text{Gamma}(\alpha_0, \beta_0)$$

To give an empirical bayes solution, we estimate α_0 and β_0 from the marginal distribution of the $S_g|T_{\cdot g}$ across all events by using the fact that the conjugate Gamma prior for θ results in an analytical expression for the marginal distribution given by the generalized beta distribution (Raiffa and Schlaifer, 1961),

$$\begin{aligned} P(S_g|T_{1g},..., T_{Kg}, \alpha_0, \beta_0) &= \int_{\theta_g} P(S_g|T_{1g}, ..., T_{Kg}, \theta_g) P(\theta_g|\alpha_0, \beta_0)\, \mathrm{d}\theta_g \\ &= \frac{S_g^{(n-K)/2-1} \beta_0^{\alpha_0}}{(S_g + \beta_0)^{(n-K)/2+\alpha_0} B((n-K)/2, \alpha_0)} = \text{GB}\left(\frac{n-K}{2}, \alpha_0, 1, \beta_0\right). \end{aligned} \tag{3.13}$$

This means that the $S_g|T_{\cdot g}$ are marginally identically distributed. Assuming independence of both θ_g and S_g across all events we can define the joint likelihood of $\{S_g|T_{\cdot g}\}_1^p$ as the product of the marginals and use this to find estimates $\hat{\alpha}_0$ and $\hat{\beta}_0$.

Using these estimates, $\hat{\alpha}_0$ and $\hat{\beta}_0$, the empirical bayes estimate of θ_g for every event g is given as

$$\hat{\theta}_g^{DEB} = E(\theta_g|S_g, T_{\cdot g}, \hat{\alpha}_0, \hat{\beta}_0) = \frac{\frac{n-K}{2} + \hat{\alpha}_0}{S_g + \hat{\beta}_0}.$$

We call this Double exponential Empirical Bayes with application to Sequencing (DEB-Seq).

Estimation in DEB-Seq Since α_0 and β_0 are positive we first transform them to the (0,1) range by defining $\gamma_{\alpha_0} = \frac{\alpha_0}{1+\alpha_0}$ and $\gamma_{\beta_0} = \frac{\beta_0}{1+\beta_0}$, and maximize the likelihood over these parameters. We then transform these estimates back to the original parameter space to obtain maximum likelihood estimates of α_0 and β_0. The convexity of the negative log-likelihood as a function of γ_{α_0} and γ_{β_0} has not been proven. The second derivatives involve gamma functions and first and second derivatives of them making it difficult to prove that the hessian is positive semidefinite. However, using data from a real RNA-Seq experiment, we examined the eigenvalues of the hessian matrix for this parameterization over a $(0,1) \times (0,1)$ grid and found that in all situations both eigenvalues were positive implying the negative log-likelihood is convex at every point on the grid. A contour plot is shown in Figure 3.1 as further evidence. The optimization routine used is the "BFGS" method implemented in the `optim` function in R to which we supply the gradient of the negative log-likelihood to assist in optimization as well as the the method of moments estimate as the starting point for the algorithm. The method of moments estimates for α_0 and β_0 are,

$$\hat{\beta}_0^{MM} = \frac{\bar{W} \cdot \bar{S}}{\frac{n-K}{2} \cdot \bar{W} - [\bar{S}]^2 (1 + \frac{n-K}{2})}$$

$$\hat{\alpha}_0^{MM} = 1 + \frac{\hat{\beta}_0^{MM} \cdot \frac{n-K}{2}}{\bar{S}}$$

where the indicated averages are taken over all events g and $W_g = S_g^2$.

The DEB-Seq method is implemented in our `DoubleExpSeq` R package available from the Comprehensive R Archive Network (CRAN).

3.3.3 Empirical Bayes via Weighted Likelihood (WEB-Seq)

One major advantage of the empirical bayes method in estimating the dispersion is that the amount of shrinkage performed is entirely determined from the data unlike the weighted likelihood method which requires the user to choose a value. It is not known if a single default will perform adequately across a range of differing experiments.

It is clear from comparing the two estimators above that they take the same form, implying the weighted likelihood method can be written as an empirical bayes solution where the prior is parameterized by a single variable δ rather than the two parameters α_0 and β_0,

$$\alpha_0 = \delta \, \frac{n-K}{2}$$

$$\beta_0 = \delta \bar{S}.$$

We are implicitly treating $\bar{S}$ as a fixed value, rather than explicitly conditioning on it, but $\bar{S}$ will normally be the average of thousands if not tens-of-thousands of events. Note that in the weighted likelihood approach, δ is assumed to be strictly positive, and $\bar{S}$ will similarly be positive, therefore satisfying the assumptions for α_0 and β_0 to yield a true density.

This naturally suggests an estimator based on this alternative parameterization to fuse these two methods together, which we call a Weighted Empirical Bayes shrinkage with application to Sequencing, or WEB-Seq. This results in a reparameterization of the marginal density of $S_g | T_{\cdot g}$ (3.13) as,

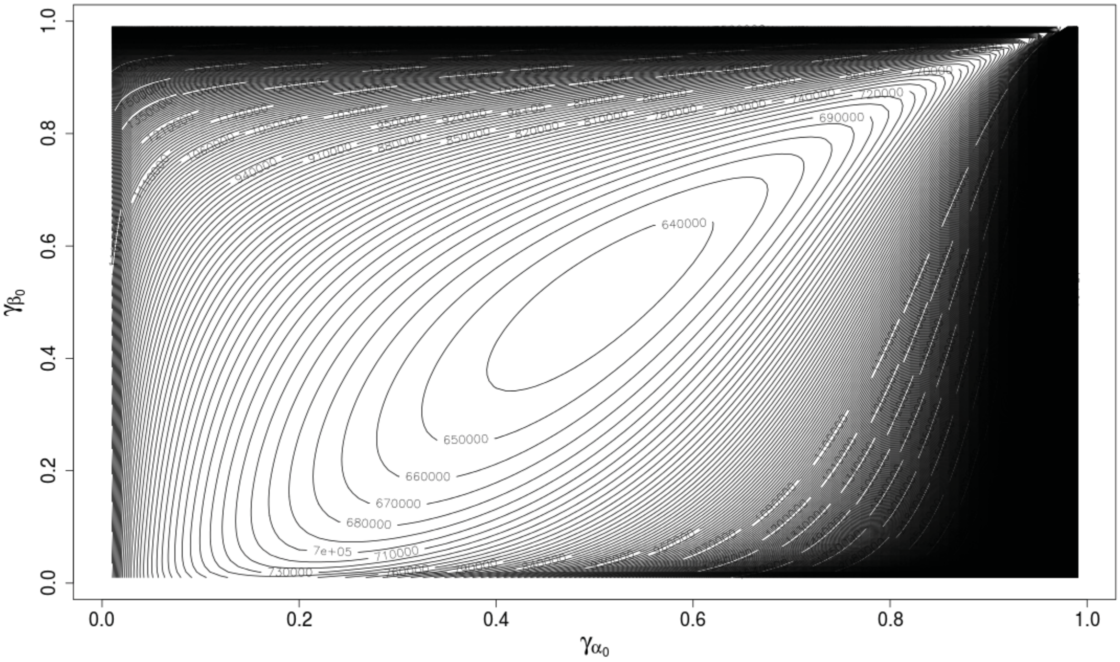


Figure 3.1: **Contour Plot of Generalized Beta Distribution** A contour plot of the negative log-likelihood of the generalized beta distribution corresponding to equation 3.13 and parameterized in terms of γ_{α_0} and γ_{β_0}. A real RNA-Seq data set was used to obtain the data, S_g for all events g. The plot seems to indicate a convex function.

$$p(S_g|T_g) = \frac{S_g^{(n-K)/2-1}(\delta\bar{S})^{\delta\ \frac{n-K}{2}}}{(S_g+\delta\bar{S})^{\frac{n-K}{2}(1+\delta)}B(\frac{n-K}{2},\delta\ \frac{n-K}{2})} = \text{GB}\left(\frac{n-K}{2},\delta\frac{n-K}{2},1,\delta\bar{S}\right)$$

Maximizing this density as described in section 3.3.2 for the empirical bayes method gives us an estimate $\hat{\delta}$ and represents the amount of shrinkage that is performed in the weighted likelihood method as determined by the data. The resulting dispersion estimates are then,

$$\hat{\theta}_g^{WEB} = \frac{\frac{n-K}{2}(1+\hat{\delta})}{S_g+\hat{\delta}\bar{S}}.$$

We will see that WEB-Seq has similar performance to the original empirical bayes approach, though it is more conservative and as a result slightly less powerful. Both methods perform well, and we choose to focus on this method largely because it appears to be more robust to violations of the model due to being more conservative.

Estimation in WEB-Seq To aid in optimization the dispersion parameter is transformed to the (0,1) range by defining $\gamma = \delta/(1+\delta)$ and optimizing over γ. Though the convexity of the negative log-likelihood in terms of γ has not been shown, the convexity and solution can be easily checked since it is a function of a single variable in a finite range. This is implemented in the `plot.optim` function provided in our own `DoubleExpSeq` package. In our application of this method, we have not found a situation in which the likelihood was not convex. The optimization routine used is the `optim` function provided in the base package of R using the 'Brent' method with the lower and upper bounds of 0 and 1, respectively.

3.3.4 General Estimation Details in Exon Usage

Missing Data In an exon usage analysis, it is common that some samples will contain no information for some exons. Recall that the offset or total count, m_{ig}, for exon g in sample i is defined as the sum of the reads that express the exon and the reads that explicitly skip the exon for that sample. In some cases, however, a sample may have no reads skipping nor expressing an exon; for instance, when a gene is not expressed in one of the samples then all exons from the gene will have no counts. This implies that $m_{ig} = 0$ and also that $y_{ig} = 0$, resulting in a missing value for the proportion, y_{ig}/m_{ig}. Recall the empirical bayes setup from section 3.3.2 that for each event g the marginal distribution of $S_g|T_{\cdot g}$ is,

$$P(S_g|T_{1g},...,T_{kg},\alpha_0,\beta_0) \sim \text{GB}\left(\frac{n-K}{2},\alpha_0,1,\beta_0\right)$$

where n is the number of samples, K is the number of groups, and S_g is calculated using all the data for event g.

When an exon is missing data for one or more samples, the number of informative samples for that exon is actually less than n and the marginal distribution of $S_g|T_{\cdot g}$ for that exon changes accordingly. To account for this, we define a new parameter, n_g^{eff}, which represents an exon specific sample size, or <u>eff</u>ective sample size, that is adjusted based on the available data for the exon. This results in a distribution of the marginal for each exon g as,

$$P(S_g|T_{1g},...,T_{kg},\alpha_0,\beta_0) \sim \text{GB}\left(\frac{n_g^{eff}-K}{2},\alpha_0,1,\beta_0\right)$$

and estimates for DEB-Seq and WEB-Seq,

$$\hat{\theta}_g^{DEB} = \frac{\frac{n_g^{eff}-K}{2}+\alpha_0}{S_g+\hat{\beta}_0}$$

$$\hat{\theta}_g^{WEB} = \frac{\frac{n_g^{eff}-K}{2}(1+\hat{\delta})}{S_g+\hat{\delta}\bar{S}}.$$

Note, it is required that $n_g^{eff} \in \{K+1, ..., n\}$ for the distribution and estimates to be valid and we further require that each group contain at least one sample with data.

Since the joint distribution of the marginals, $S_g|(T_{\cdot g}, \alpha_0, \beta_0)$, across all g is required to estimate the parameters α_0 and β_0 in the prior distribution of θ (again, refer to section 3.3.2 for the details) and the marginal distribution changes from exon to exon, we re-define the joint distribution as,

$$P(S_1, ..., S_p|T_{\cdot\cdot}, \alpha_0, \beta_0) = \prod_{r=K+1}^{n} \prod_{g:\ n_g^{eff}=r} \text{GB}\left(S_g;\ \frac{r-K}{2}, \alpha_0, 1, \beta_0\right)$$

Note, missing data is discussed specifically in the context of exon usage since such data is not possible in gene expression experiments. The total count, m_{ig}, for all genes g is defined as the total number of sequences across all genes of sample i which cannot be zero, barring an egregious error in the sequencing steps. In such a case, however, the sample would be removed from the experiment altogether.

Estimates of under-dispersion near the boundary We found in our initial simulations of exon usage analysis that estimated proportions lying on the boundary across a lot of samples for an exon (i.e. either exactly 1 or 0, corresponding to a sample that displays no skipping or only skipping) have a large and adverse effect on the false discovery rate (FDR) as the sample size increases (see Figure 4.8). This is because the method estimates underdispersion for such exons, leading to a large number of false discoveries. Moreover, the effect is worse with larger sample sizes: the effect becomes noticeable around 5-10 samples per group and for increased sample sizes the FDR grows without control (see section 4.1.2.1 and Figure 4.8). The reason for the increase with larger sample sizes is that exons with proportions *all* 1 or 0 across all samples get a p-value of one, which results in an implicit filter of the data. For exons whose true proportion is near the boundary, the observed data is more likely in low sample sizes to have estimated proportions entirely on the boundary and therefore assigned a p-value of one. In larger sample sizes, there is an increased chance that a non-boundary sample will be observed, allowing the exon to remain in the analysis and have an effect on the FDR results.

There are several ad hoc approaches to this issue. One is to filter exons with mean proportion close to the boundary. While a successful filtering procedure can result in a positive impact on the power of a test (Bourgon *et al.*, 2010b), in this case it is unsatisfactory to have a test-statistic that is so sensitive to the degree of filtering. Another approach is to not allow underdispersion by setting the dispersion in such cases to one, i.e. binomial variance (and see also the most recent version of DESeq called DESeq2 (Love *et al.*, 2014) that does not allow the dispersion estimate to be decreased via shrinkage).

We obtained better results by adding an additional adjustment to n_g^{eff} for each event g, where n_g^{eff} is further reduced by the amount of samples lying on the boundary for the event. This further reduction is in addition to the number of missing data points defined in the previous discussion. Also, recall that $n_g^{eff} \in \{K+1, ..., n\}$ must hold in order for the marginal distribution of $S_g|T_{\cdot g}$ to be valid. For this reason, n_g^{eff} is never allowed to be less than $K+1$. Letting a_g and b_g be the number of missing and boundary data points across the samples, respectively, n_g^{eff} is explicitly calculated as,

$$n_g^{eff} = \max\{n - (a_g + b_g),\ K + 1\}$$

The result is that it remains possible to estimate underdispersion for an exon, but this adjustment makes it more difficult to erroneously estimate underdispersion when an exon is on the boundary. With this adjustment no underdispersion is in fact estimated at any sample size for any exon in our simulated or real data sets, while previously exons with proportion parameters near the boundary were frequently estimated to be underdispersed. Further, a continuous range of dispersion values θ are estimated for these boundary exons which we find to be more natural than forcing them all to $\theta = 1$, i.e. no dispersion.

3.4 Inference for the Effect Size β_1

Once the dispersion has been estimated via one of the shrinkage methods, our attention returns to the original goal of testing for differential expression (or usage) in two group setting. Recall from section 2.4.1, a marginal testing approach is taken which results in a test for a difference between the mean parameters in each condition separately for every event g:

$$H_0^g : \mu_{c1,g} = \mu_{c2,g}, \quad H_1^g : \mu_{c1,g} \neq \mu_{c2,g}$$

where c_k represents condition k. To that end, we set up the following generalized linear model:

$$h(\mu_{ig}) = \beta_{0g} + \beta_{1g} \cdot 1_{\{\rho(i)=c_1\}} \tag{3.14}$$

where h is the link function, β_{0g} is the overall strength of the signal across all samples, $\rho(\cdot)$ maps the sample subscript to its respective condition, and β_{1g} measures the effect that condition c_1 has on the mean, μ_{ig}, through h.

At this point, we would have already chosen a double exponential family distribution to model the data. With it comes the canonical link, previously denoted as η_μ in section 3.1, which we use for h. Since our interest lies in detecting an effect of the condition on the mean parameter, this transforms our original hypothesis test to be in terms of β_{1g}:

$$H_0^g : \beta_{1g} = 0, \quad H_1^g : \beta_{1g} \neq 0$$

To test the above hypothesis we obtain an MLE estimate, $\hat{\beta}_{1g}$, by reparameterizing the likelihood for μ_{ig} given by our chosen distribution in terms of the GLM model parameters β_{0g} and β_{1g} and maximizing the likelihood with respect to these parameters. This also includes plugging in our shrunken estimate of the dispersion for θ into the likelihood which we treat as known constant.

3.4.1 Likelihood Ratio Statistics

A very popular test for the above hypothesis is a likelihood ratio test which compares the likelihood of the model corresponding to the null hypothesis and the likelihood of the full model given by Eq. 3.14 at their estimated parameters. For completeness, the null model is defined as,

$$h(\mu_{ig}) = \beta_{0g}$$

and a new estimate of β_{0g} is obtained from this model which will be denoted as $\beta_{0g}^{H_0}$. Note, from here on, the subscript g will dropped unless it is needed for clarity.

The standard quasi-likelihood approach defines the likelihood ratio statistic as,

$$W_{\hat{\theta}} = \log \frac{L(\hat{\beta}_0^{H_0};\ \beta_1 = 0, \hat{\theta})}{L(\hat{\beta}_0, \hat{\beta}_1;\ \hat{\theta})}$$

where $\hat{\theta}$ is estimated from the full model that includes β_1 and is considered known, and L is the likelihood function given by the distribution used to model the data. For calculating $L(\hat{\beta}_0, \hat{\beta}_1; \hat{\theta})$ we use $\hat{\beta}_1 = \hat{\eta}_{\hat{\theta},1} - \hat{\eta}_{\hat{\theta},2}$, where $\hat{\eta}_{\theta,c}$ is the maximum likelihood estimate of η in condition c at a *fixed* value of θ.

The dispersion estimate is dependent on the grouping structure placed on the data, and ideally, we would estimate a separate value for θ under the null hypothesis $H_0 : \beta_1 = 0$, in addition to the estimate under the full model. We denote the estimate under the null model as $\hat{\theta}_{H_0}$ and the estimate under the full model we keep as $\hat{\theta}$. Then, we define our modified likelihood ratio statistic as,

$$W_{\hat{\theta},\hat{\theta}_{H_0}} = \log \frac{L(\hat{\beta}_0^{H_0};\ \beta_1 = 0, \hat{\theta}_{H_0})}{L(\hat{\beta}_0, \hat{\beta}_1;\ \hat{\theta})}$$

We note that this implies the likelihoods are not strictly nested meaning that the test statistic can be both negative and positive which is not the case in the standard likelihood ratio test. Some work has been done for the testing of non-nested models but they create a test statistic based on a mean of the per-data point likelihood ratio that further requires an estimate of its standard error (Cox, 1961; Vuong, 1989). In small sample sizes, this standard error estimate of the mean induces additional variation to the test statistic and results in a large amount of false discoveries while also producing an unstable and inferior ranking of the events by p-value. Our modified likelihood ratio test statistic does not require further estimates that would be based on a small number of samples, and as we will demonstrate in the coming chapters we find it provides quality ranks compared to other methods and in some situations is superior to all other methods. For this reason, we choose to use our modified likelihood ratio statistic for testing and compare it to the chi-square distribution to provide a cutoff for significance.

3.4.1.1 Relationship Between $W_{\hat{\theta}}$ and $W_{\hat{\theta},\hat{\theta}_{H_0}}$

Since the asymptotic properties of $W_{\hat{\theta}}$ are well understood, it is important to draw connections between it and our modified version of the statistic. To that end, we use the definitions and results from sections 3.1 and 3.2 to get,

$$
\begin{aligned}
W_{\hat{\theta}} =& \hat{\theta}\left(\sum_k \left[M_k(\hat{\eta}_k T_k - \hat{\psi}_k)\right] - M(\hat{\eta}_{H_0} T - \hat{\psi}_{H_0})\right) = \hat{\theta}\left(\sum_k [M_k(-\rho(T_k))] \; - \; M(-\rho(T))\right) \\
=& \hat{\theta}\left(M\rho(T) - \sum_k M_k \rho(T_k)\right) = \hat{\theta}\left(M\rho(T) - \sum_k M_k\rho(T_k) - MU + \sum_k M_k U_k\right) \\
=& \hat{\theta}\left(M(\rho(T) - U) - \sum_k M_k(\rho(T_k) - U_k)\right) \\
=& \hat{\theta}\left(0.5\sum_i D_{H_0}(z_i, \hat{\mu}) - 0.5\sum_k \sum_{i \in k} D(z_i, \hat{\mu}_k)\right) \\
=& \hat{\theta}(S_{H_0} - S)
\end{aligned}
$$

and,

$$
\begin{aligned}
W_{\hat{\theta},\hat{\theta}_{H_0}} =& W_{\hat{\theta}} \; - \; M(\hat{\theta} - \hat{\theta}_{H_0})(\rho(T) - U) + \frac{n}{2}\log\left(\frac{\hat{\theta}}{\hat{\theta}_{H_0}}\right) \\
=& W_{\hat{\theta}} \; - \; 0.5(\hat{\theta} - \hat{\theta}_{H_0})\sum_i D_{H_0}(z_i, \hat{\mu}) + \frac{n}{2}\log\left(\frac{\hat{\theta}}{\hat{\theta}_{H_0}}\right) \\
=& W_{\hat{\theta}} \; - \; (\hat{\theta} - \hat{\theta}_{H_0})S_{H_0} + \frac{n}{2}\log\left(\frac{\hat{\theta}}{\hat{\theta}_{H_0}}\right) \\
=& \hat{\theta}(S_{H_0} - S) \; - \; (\hat{\theta} - \hat{\theta}_{H_0})S_{H_0} + \frac{n}{2}\log\left(\frac{\hat{\theta}}{\hat{\theta}_{H_0}}\right) \\
=& \hat{\theta}_{H_0}S_{H_0} - \hat{\theta}S + \frac{n}{2}\log\left(\frac{\hat{\theta}}{\hat{\theta}_{H_0}}\right)
\end{aligned}
$$

The difference between the two statistics becomes,

$$
D_{\hat{\theta},\hat{\theta}_{H_0}} = W_{\hat{\theta},\hat{\theta}_{H_0}} - W_{\hat{\theta}} = (\hat{\theta}_{H_0} - \hat{\theta})S_{H_0} + \frac{n}{2}\log\left(\frac{\hat{\theta}}{\hat{\theta}_{H_0}}\right),
$$

implying,

$$
W_{\hat{\theta},\hat{\theta}_{H_0}} = W_{\hat{\theta}} \; + \; D_{\hat{\theta},\hat{\theta}_{H_0}} \; = \; W_{\hat{\theta}} \; + \; (\hat{\theta}_{H_0} - \hat{\theta})S_{H_0} + \frac{n}{2}\log\left(\frac{\hat{\theta}}{\hat{\theta}_{H_0}}\right).
$$

S_{H_0} is nonnegative being the sum of deviance residuals (see section 3.2). The last two terms will always be opposite signs; therefore, the direction of change to $W_{\hat{\theta}}$ is dependent on which of these terms is larger in absolute value.

3.4.1.2 Test Statistics Under Shrinkage

If we now look at a shrinkage estimate, then we can only say that $W^s_{\hat{\theta}} = \hat{\theta}^s(S_{H_0} - S)$. To see how the MLE version $W^{\text{mle}}_{\hat{\theta}}$ is related to the shrinkage version $W^s_{\hat{\theta}}$ we can do the following:

$$W^s_{\hat{\theta}} = \hat{\theta}^s(S_{H_0} - S) = \hat{\theta}^s\left(\frac{n}{2\hat{\theta}^{\text{mle}}_{H_0}} - \frac{n}{2\hat{\theta}^{\text{mle}}}\right) = \frac{\hat{\theta}^s}{\hat{\theta}^{\text{mle}}}\frac{n}{2}\left(\frac{\hat{\theta}^{\text{mle}}}{\hat{\theta}^{\text{mle}}_{H_0}} - 1\right) = \frac{\hat{\theta}^s}{\hat{\theta}^{\text{mle}}}W^{\text{mle}}_{\hat{\theta}}$$

which does not always translate to shrinkage of the standard test statistic unless $\hat{\theta}^s \leq \hat{\theta}^{\text{mle}}$. If we assume $\hat{\theta}^s = \hat{\theta}^{WEB}$ then we can solve for the δ that would guarantee shrinkage. For a specific event $g*$,

$$\hat{\theta}^{\text{mle}}_{g*} = \frac{n}{2S_{g*}} \text{ and } \hat{\theta}^{WEB}_{g*} = \frac{\frac{n-K}{2}(1+\delta)}{S_{g*} + \delta\bar{S}} \text{ and } \bar{S} = \frac{1}{p}\sum_g S_g$$

Then,

$$\begin{aligned} &\hat{\theta}^{WEB}_{g*} < \hat{\theta}^{\text{mle}}_{g*} \\ \Longrightarrow &\frac{\frac{n-K}{2}(1+\delta)}{S_{g*} + \delta\bar{S}} < \frac{n}{2S_{g*}} \\ \Longrightarrow &\delta < \frac{K}{n - K - \frac{n}{S_{g*}}\bar{S}} \end{aligned}$$

Since this is difficult to interpret we can adjust the right side of the equation to arrive at an even stricter inequality:

$$\begin{aligned} \frac{\frac{n-K}{2}(1+\delta)}{S_{g*} + \delta\bar{S}} < \frac{n-k}{2S_{g*}} = \hat{\theta}^{\text{AC}}_{g*} < \frac{n}{2S_{g*}} = \hat{\theta}^{\text{mle}}_{g*} \\ \Longrightarrow \delta S_{g*} < \delta\bar{S} \\ \Longrightarrow S_{g*} < \bar{S}, \end{aligned}$$

where $\hat{\theta}^{AC}_{g*}$ is the estimate of θ_{g*} using the approximate conditional distribution of θ_{g*} (see section 3.2). This guarantees that all events g whose corresponding S_g is less than the mean across all events results in a shrunken test statistic for all $\delta > 0$, i.e. $0 < W^s_{\hat{\theta}} < W^{\text{mle}}_{\hat{\theta}}$. Note that if the distribution of S_g is roughly gamma this implies that most test statistics are shrunk since the gamma distribution is skewed right and therefore the median is less than the mean. A similar analysis for $\hat{\theta}^{DEB}_j$ would be based on two parameters and therefore more complicated to inspect.

For comparison, if we use the edgeR default for δ equal to $\frac{20}{n-K}$ (refer to section 3.3.1) it is easy to show that shrinkage of the test statistic is ensured if,

$$S_{g*} < 20\bar{S}\frac{n}{n-K}$$

which results in significantly more events having a shrunken test statistic. Note, with δ constructed in this way as $n \to \infty$ this converges to $20\bar{S}$ so that even with a large sample size the number of shrunken test statistics is only partially reduced. Comparing to the result of the inequality for δ, if $n \to \infty$ the right side goes to 0 and the requirement on δ to provide shrinkage of the test statistic also goes to 0 which is a more intuitive property, though it remains that $S_{g*} < \bar{S}$ still results in shrinkage.

Unfortunately, for our modified statistic, $W_{\hat{\theta},\hat{\theta}_{H_0}}$, comparisons between its shrinkage and MLE versions do not result in a form amenable to analytical interpretation. This includes the asymptotics of the shrinkage version as well as the effect of shrinkage relative to the MLE version.

3.4.2 Wald Statistics

Another common approach is to use Wald statistics. As stated previously, for a given estimate $\hat{\theta}$ of dispersion, we estimate η with $\hat{\eta} = \hat{\eta}_{\hat{\theta}}$, where $\hat{\eta}_\theta$ is the maximum likelihood estimate of η for a *fixed* θ. In the case of $c = 1$, the variance of $\hat{\eta}$ can be approximated by the inverse of the information matrix.

Specifically, we can write the joint information matrix as

$$I(\lambda, \theta) = M\nabla^2 A(\theta, \lambda) = \begin{pmatrix} I(\lambda) & v \\ v^T & I(\theta) \end{pmatrix}$$

where $I(\lambda)$ is a diagonal matrix with entries

$$I_{\lambda ii} = M_k \frac{V(\mu_k)}{\theta},$$

v is a vector with entries

$$v_k = -\eta_k I_{\lambda kk},$$

and

$$I(\theta) = \frac{n}{2\theta^2} + v^T 1.$$

For the calculation of these entries, note that

$$\begin{aligned} \frac{\partial^2 A}{\partial\lambda_i \lambda_k} &= \frac{M_k}{M}\frac{\partial\mu}{\partial\lambda} = \frac{M_k}{M}\frac{V(\mu_k)}{\theta} \qquad \text{if i=k; 0 otherwise} \\ \frac{\partial^2 A}{\partial\theta^2} &= \sum_k \frac{M_k}{M}\frac{\partial\rho(\mu_k)}{\partial\theta} + \frac{n}{2M\theta^2} \\ \frac{\partial^2 A}{\partial\theta\lambda_k} &= \frac{M_k}{M}\frac{\partial\mu_k}{\partial\theta} = -\frac{M_k}{M}\eta_k\frac{V(\mu_k)}{\theta} \end{aligned}$$

We are interested in contrasts of the vector η, so we can write the information matrix of (η, θ) as

$$I(\eta, \theta) = J^T I(\lambda, \theta) J = \begin{pmatrix} \theta^2 I(\lambda) & \theta v \\ \theta v^T & I(\theta) \end{pmatrix}$$

where J is a diagonal matrix,

$$\begin{pmatrix} \theta I_K & 0 \\ 0 & 1 \end{pmatrix}$$

Let

$$k = I(\theta) - v^T I(\lambda)^{-1} v = \frac{n}{2\theta^2}.$$

We have that

$$I(\eta,\theta)^{-1} = \frac{1}{k}\begin{pmatrix} k\frac{1}{\theta^2} I(\lambda)^{-1} + \frac{1}{\theta^2}\eta\eta^T & \frac{1}{\theta^2}\eta \\ \frac{1}{\theta^2}\eta^T & 1 \end{pmatrix}$$

For the two group case, let $\beta_1 = \eta_2 - \eta_1$ be the parameter of interest. We have that

$$var(\beta_1) \approx \frac{1}{\theta}\left(\frac{1}{M_1 V(\mu_1)} + \frac{1}{M_2 V(\mu_2)}\right) + \frac{2}{n}(\eta_1 + \eta_2)^2.$$

Since we use the estimate $\hat{\eta}_{\hat{\theta}}$, i.e. assume that θ was a known constant $\hat{\theta}$, we lose the second term in the equation of the variance of $var(\beta_1)$. This gives us the statistic,

$$t = \frac{\hat{\theta}\,(\eta_{\hat{\mu}_1} - \eta_{\hat{\mu}_2})}{\sqrt{\frac{1}{M_1 V(\hat{\mu}_1)} + \frac{1}{M_2 V(\hat{\mu}_2)}}}$$

If $\hat{\theta}$ is estimated from $c = 1$, it is in the form

$$\hat{\theta} = \frac{(n-K)/2 + \alpha_0}{S + \beta_0},$$

for some α_0, β_0 depending on whether the method was weighted likelihood or Empirical Bayes. Recall that $S = \sum_k M_k \rho(T_k) - MU$. Then the estimate can be written as

$$t = \frac{\sqrt{(n-K)/2 + \alpha_0}\,(\eta_{\hat{\mu}_1} - \eta_{\hat{\mu}_2})}{\sqrt{(S + \beta_0)\frac{1}{M_1 V(\hat{\mu}_1)} + \frac{1}{M_2 V(\hat{\mu}_2)}}}$$

Chapter 4

Application to Differential Exon Usage Analysis

As discussed in section 2.4, our motivating example comes from the question of measuring alternative splicing – when the gene can produce multiple versions of mRNA that include different combinations of the exons of a gene. One simple approach to finding differences in alternative splicing is to measure the number of sequences including the exon and compare it to the number that explicitly skip the exon, in which case the data being modeled are proportions that take on the full range of 0 to 1, necessitating the use of models in the binomial family.

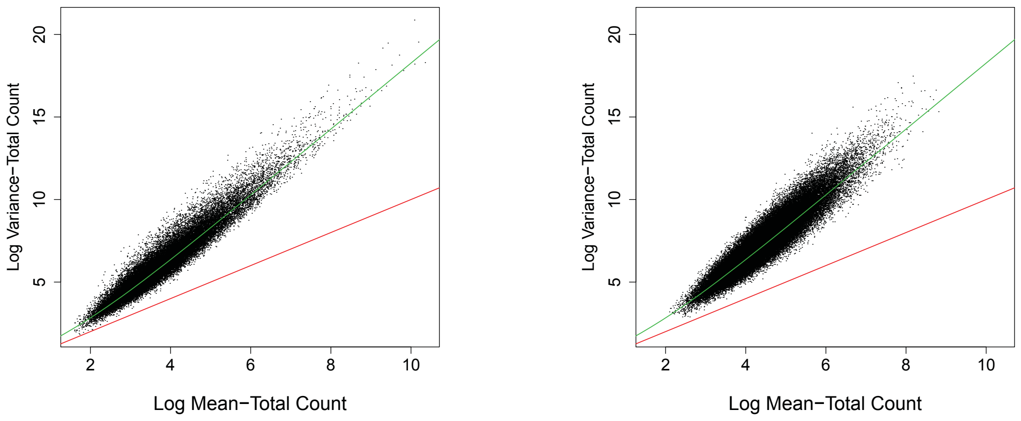
In chapter 3, we developed the double exponential framework and specifically the double binomial (section 3.1.3) which has the capability of analyzing data in this setting. Further, in sections 3.3.2 and 3.3.3, we developed two dispersion shrinkage estimators called WEB-Seq and DEB-Seq. In this chapter, we compare the performance of these estimators to other existing methods and demonstrate that in addition to providing a fully automated approach for shrinkage, our methods have superior performance on simulated data in the exon inclusion setting. We also apply these methods to mRNA-Seq data from real tumor samples generated by the Cancer Genome Atlas project (Cancer Genome Atlas Research Network, 2011) which suggests that it can similarly control the false discovery rate and find promising targets of splicing. Furthermore, there is very little computational overhead in our methods compared to existing methods.

4.1 Simulations

4.1.1 Simulation Details

We simulated data to mimic the exon inclusion setting. Specifically, for 85,373 exons, and a specified n samples per group, we simulated counts, per exon, of inclusion and exclusion based on either the double-binomial or the beta-binomial distribution. Each exon could be either null ($\beta_1 = 0$) or non-null, and the final measures of performance were the ability to control false discoveries and the power to detect non-null exons over 100 simulations.

The simulation required additional parameters other than β_1 for each exon. For each



(a) Real Data

(b) Simulated Data

Figure 4.1: **Total Count Mean-Variance Relationship** Plotted is the variance of the total count (y-axis) against the mean of the total count (x-axis) per exon on the log scale. The red line signifies the poisson fit to the data (variance=mean) and the green line represents the negative binomial fit to the data. The data is comprised of all 170 samples. (a) is plotted using the real AML data. (b) is plotted from a randomly selected simulated data set out of the 100 total simulated data sets that were created from the AML data.

of the two distributions, the dispersion parameter θ, the probability of success μ, and the total number of counts m_i for each sample i, need to be determined for each exon. In order to select the parameters, we used a real set of 170 RNA-Seq samples sequenced from Acute Myeloid Leukemia samples generated by the Cancer Genome Atlas project (TCGA) (Cancer Genome Atlas Research Network, 2011) and subsequently downloaded and processed by Marla Johnson of the Purdom Lab at UC Berkeley. Then the simulation parameters for each exon were determined by fitting a standard GLM separately to each exon of the real data set with 170 samples (i.e. no shrinkage was used) using either the double-binomial or beta-binomial likelihood, as relevant.

Specifically, the distribution of the total counts m_{ig} within a given exon g closely followed a negative binomial (see Figure 4.1). Across exons, the log of the mean of the total counts of an exon, $\bar{m}_{\cdot g}$, closely followed a log-normal distribution (see Figure 4.2). The variance of the m_{ig} per exon was observed to be linear in $\bar{m}_{\cdot g}$. These observations resulted in the following method for picking individual m_{ig}. For a given exon g and group, mean total count $\bar{m}_{\cdot g}$ was chosen from the log-normal distribution matching that of the data and the variance of the m_{ig} was then chosen according to the linear relationship with $\bar{m}_{\cdot g}$ observed in the data. These two values defined the parameters of the negative binomial distribution from which the individual (per sample) total counts m_{ig} were selected.

The mean of the total counts $\bar{m}_{\cdot g}$ was observed to be related to the dispersion θ and the proportion μ (see Figures 4.3 and 4.4). To emulate this relationship, the observed $\hat{\theta}_g$ and $\hat{\mu_g}$ estimated from the data were binned into 50 groups based on their observed $\bar{m}_{\cdot g}$. For each simulated exon with a value of $\bar{m}_{\cdot g}$ assigned as described above, a value of θ and μ were then assigned by uniformly sampling from the bin of $\hat{\theta}$ and $\hat{\mu}_g$ that corresponded to

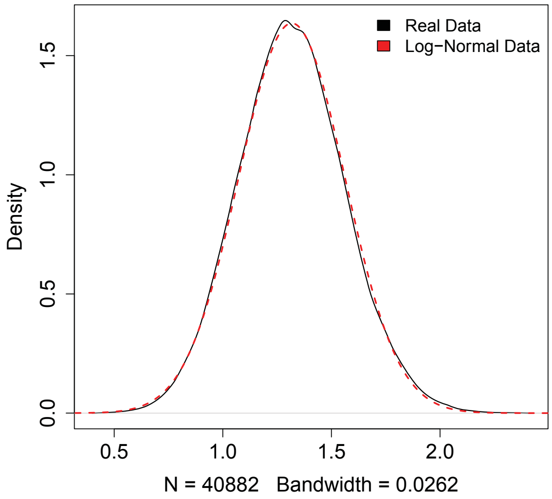


Figure 4.2: **log-Normal fit to Total Counts** Plotted are the densities of a normal curve (red) and the double log of the mean total counts per exon (black). Note, this shows that the single log of the mean total counts is roughly log-normal since the double log is roughly normal.

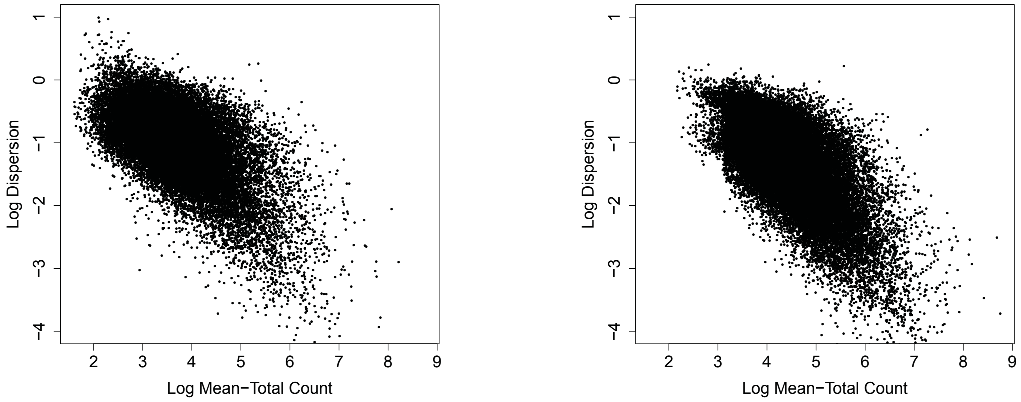
the simulated value of $\bar{m}_{\cdot g}$.

The effect size β_1 for non-null exons was uniformly chosen between 0.5 and 3 and then randomly chosen to be up or down regulated.

One-hundred simulated data sets of the same size and design of conditions as the real RNA-Seq data were generated under each of the two distributions. From these, smaller data sets were subsampled ranging from a 2 versus 2 scenario to a 75 versus 75. To be able to still control the amount of null and non-null exons in our simulation, each exon was simulated under a null and non-null simulation and then filtered. After filtering, 10% of the remaining exons were assigned their non-null version and the remaining were assigned their null version.

We used the simulated data to evaluate the methods developed above: 1) the empirical bayes with a single parameter prior (WEB-Seq) 2) the general two-parameter empirical bayes method for the prior parameters (DEB-Seq), and 3) the weighted likelihood method with δ fixed to be equal to the default value implemented in edgeR ($\frac{20}{n-K}$) (Robinson *et al.*, 2010)). In addition to our dispersion shrinkage methods, we implemented the shrinkage method of BBSeq and EB2. The MATS method described in section 2.6.1 does not take as input inclusion and exclusion count matrices, but rather creates it own from BAM alignment files, and thus could not be compared on the simulated data.

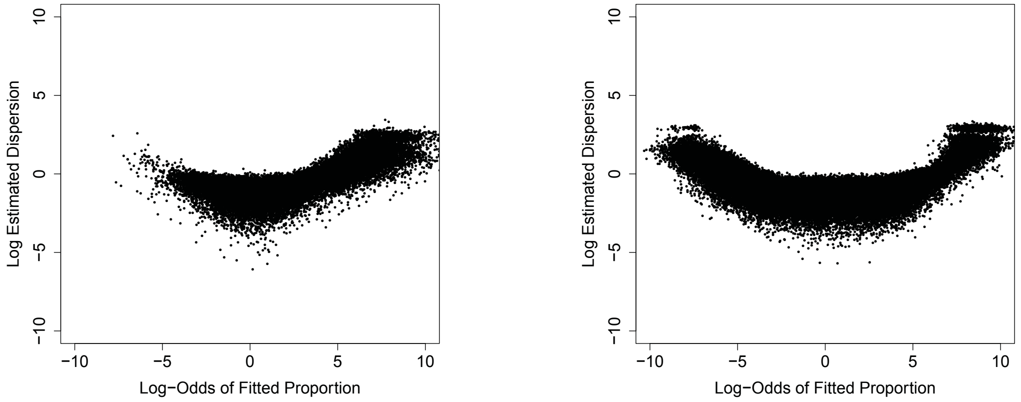
We also implemented three methods that fit a dispersion parameter per exon but with no shrinkage across exons: quasi-binomial GLM estimation as implemented in the `glm` function in R (R Core Team, 2013), maximum likelihood estimation based on a beta-binomial distribution, and maximum likelihood estimation based on an approximate double binomial distribution where the normalizing constant is set to 1 (see Section 3.1). The quasi-binomial



(a) Real Data

(b) Simulated Data

Figure 4.3: **Relationship between Dispersion and Total Count** Plotted is the log of estimated dispersions (y-axis) against the log of the mean total count per exon. (a) is plotted using the real AML data. (b) is plotted from a randomly selected simulated data set out of the 100 total simulated data sets that were created from the AML data.



(a) Real Data

(b) Simulated Data

Figure 4.4: **Relationship between Dispersion and Proportion** Plotted is the log of the estimated dispersion (y-axis) against the log odds of the fitted proportion for a single group (x-axis). (a) is plotted using the real AML data. (b) is plotted from a randomly selected simulated data set out of the 100 total simulated data sets that were created from the AML data.

GLM and the double binomial MLE are closely connected, as described in Section 3.1, and are both non-shrinkage counterparts to our methods. However, the quasi-binomial estimation by default uses Pearson residuals to estimate the dispersion, rather than deviance residuals. The beta-binomial maximum likelihood method is the non-shrinkage counterpart of the BBSeq method.

For each procedure, the estimation procedures were performed and the p-values were adjusted to control the FDR to a 0.05 level using the standard Benjamini-Hochberg FDR procedure (Benjamini and Hochberg, 1995) implemented in the `p.adjust` function in R (R Core Team, 2013). The final measures of performance were the methods' ability to control false discoveries and their power to detect non-null exons over the 100 simulations.

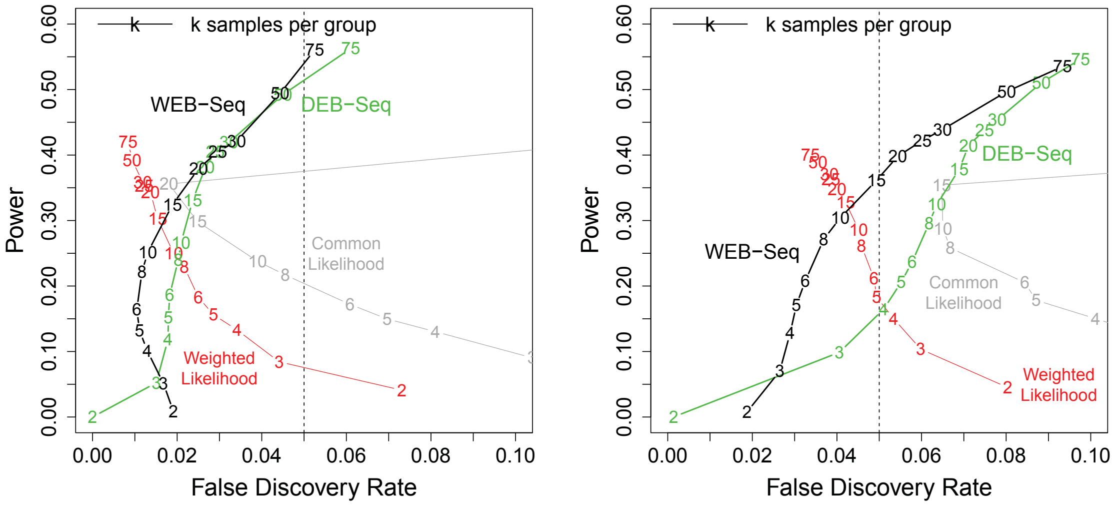
4.1.2 Simulation Results

We show the true false discovery rate plotted against power for different sample sizes for our shrinkage methods in Figure 4.5. For data distributed as double binomial, WEB-Seq and DEB-Seq control the FDR at all sample sizes converging on the expected 0.05 FDR for very large sample sizes, with WEB-Seq being more conservative and with slightly less power as a result. Weighted likelihood with a pre-determined tuning parameter (based on edgeR recommendation) is slightly erratic in its control of FDR for extremely small sample sizes, but then adequately controls FDR. However, the pre-determined tuning method becomes over-conservative for large sample sizes and the result is a large drop in power for large sample sizes.

To evaluate the robustness of the methods, we consider data not following the given model, but rather the beta-binomial distribution. Here there appears to be an underlying bias due to the p-values being calculated under the wrong model, and for large sample sizes both WEB-Seq and DEB-Seq converge to around an FDR of 0.10. However for moderate sample sizes (less than 20 per group) the more conservative WEB-Seq still manages to control the FDR; DEB-Seq still has greater power, but has a slight increase of FDR to about 0.07 for moderate sample sizes.

In Figure 4.6 we compare to other existing methods. WEB-Seq shows great improvement in controlling the FDR compared to all of the other methods – both those that use shrinkage and those that do not. The methods that do not utilize shrinkage have large false discovery rates for small to moderate samples sizes. The beta-binomial MLE with no shrinkage performs the best of the alternative methods under both simulations, but still has an FDR larger than the target 5% for less than 10-15 per group. Quasi-binomial does not come close to controlling the FDR even with double binomial distributed data until more than 50 samples per group (with an FDR of 9.7%). In contrast, WEB-Seq controls the FDR at the desired level across the full range of sample sizes for the double binomial data, only showing increased FDR in the beta-binomial data for large sample sizes.

The EB2 and BBSeq methods both implement shrinkage and rely on beta-binomial dispersion models. Both methods do not even minimally control the FDR in our simulated setting, even for beta-binomial data with large sample sizes, with FDR values ranging from 0.73 and 0.85 ($n = 2$) to 0.092 and 0.14 ($n = 75$) for BBSeq and EB2, respectively. Similarly, the double binomial GLM (not plotted) fails to control the FDR at any sample size and converges to an FDR at around 40%.



(a) Double-Binomial Simulation

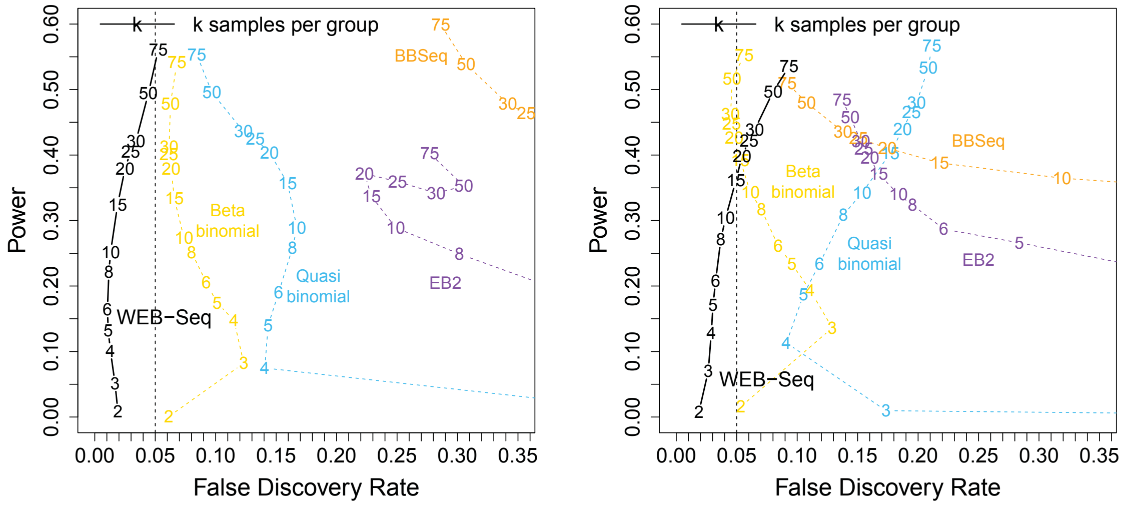
(b) Beta-Binomial Simulation

Figure 4.5: **Double Binomial Methods** Plotted is the average Power (y-axis) against FDR (x-axis) over various sample sizes across 100 double binomial simulations, based on p-values adjusted to provide a 5% FDR level. The results for a single method across different sample sizes are connected by a line. The numbers that overlay a method denote the power and FDR for that specific sample size (*per group*) in a 2 group comparison. The 5% FDR boundary is given by the dotted vertical line. The data are simulated under (a) a double binomial distribution and (b) a beta-binomial distribution. The methods shown are all based on a double binomial to account for over-dispersion: 1-parameter empirical bayes (WEB-Seq); 2-parameter empirical bayes (DEB-Seq); edgeR default weighted likelihood; and estimation of a single dispersion parameter θ for all exons (common likelihood). The double binomial MLE is not shown because it's FDR values were beyond the limits of the plot.

Many biological studies focus on the top performing exons for validation and followup analysis, especially when there are large numbers of significant results. We find that the WEB-Seq method not only provides better global performance, but also gives p-values that better prioritize the truly non-null exons, i.e. the ranks of the exons based on the p-values. In Figure 4.7, we plot the average proportion of false discoveries in the top-ranked exons for simulations with five samples per group. We see that the alternative methods have a much higher proportion of false positives in the top-ranked exons compared to WEB-Seq. We see similar behavior for beta-binomial distributed data (Figure 4.7). This demonstrates that the difference in the global FDR and power we see in Figure 4.6 is due to the actual choice of statistic, not merely a problem in the distributional assumptions for creating p-values.

4.1.2.1 Boundary Data

In Table 4.1 we compare the proportion of exons that are affected by our adjustment to the degrees of freedom (Section 3.3.4), for both the simulated and real data. We see that



(a) Double-Binomial Simulation

(b) Beta-Binomial Simulation

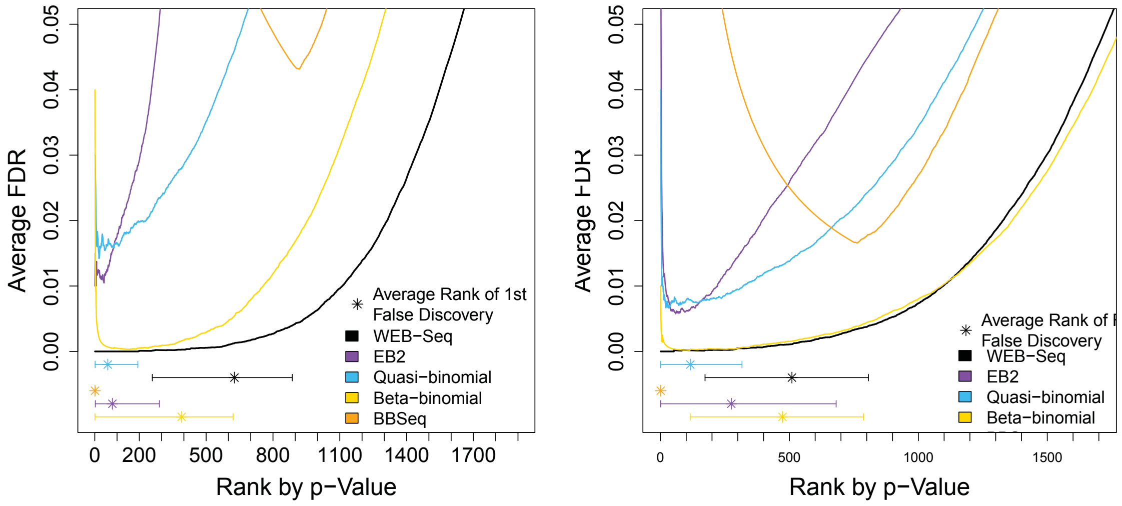
Figure 4.6: **Comparison to Alternative Methods:** Plotted is the average Power (y-axis) against FDR (x-axis) over various sample sizes across 100 simulations based on p-values adjusted to provide a 5% FDR level (see Figure 4.5 for details). The data are simulated under (a) a double binomial distribution and (b) a beta-binomial distribution. The alternative methods, both those that perform shrinkage and those that do not, are compared to WEB-Seq: Quasi-binomial (no shrinkage), BBSeq, EB2, and Beta-binomial MLE estimates (no shrinkage).

78% of the exons are affected by these changes, with 43% having a large reduction of 5 or more. For comparison, we also include similar results for the Tissue data (see 4.2.1), and interestingly we see that the real data more closely follows the double binomial simulation in this respect, rather than the beta-binomial.

A major source of the large levels of false discovery in several of the methods are also boundary exons, where the true proportions are close to 0 or 1. In Figures 4.8 and 4.9, we show similar performance plots for each of the methods. Figure 4.8 shows the improvement for the double binomial based methods on the FDR from applying the boundary correction described in Section 3.3.4. All of the methods' FDR values appear to grow uncontrollably with increasing sample size. Figure 4.9 show the results after removing the exons whose mean proportion across all samples is different from 0 or 1 by less than 0.05. We see that EB2 and quasi-binomial with no shrinkage have much better levels of FDR when these exons are filtered and are similar to WEB-Seq. However, they have a much reduced power as a result of the filtering. The beta-binomial method, on the other hand, is not affected by the filtering.

4.1.2.2 Likelihood-Ratio and Wald Tests

We compared the performance of WEB-Seq to the Wald and standard Likelihood-Ratio tests in terms of FDR and power (see Section 3.4 for derivations). The shrunken dispersion



(a) Double-Binomial Simulation

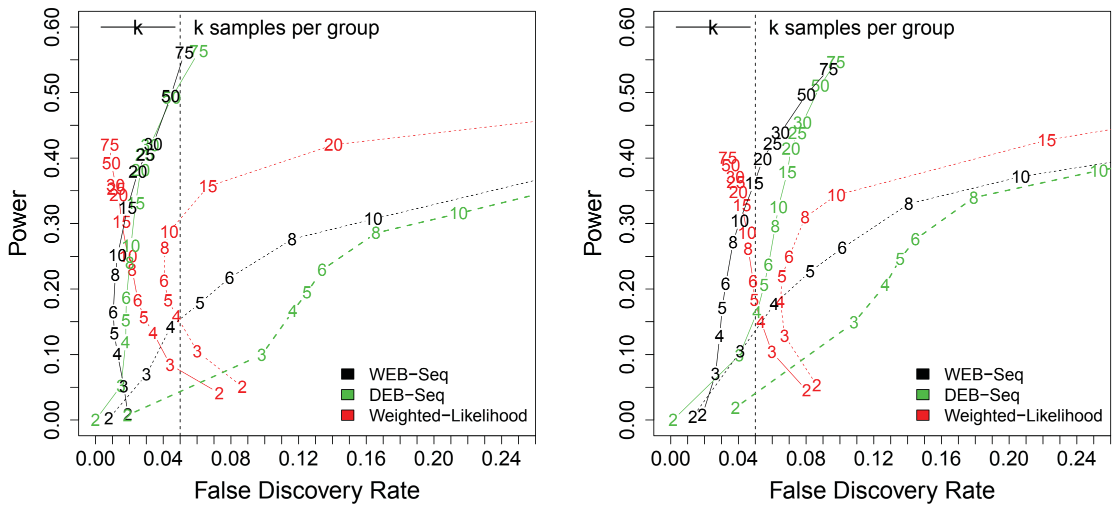
(b) Beta-Binomial Simulation

Figure 4.7: **False Discoveries by Rank.** Plotted is the average proportion of false discoveries (y-axis) in the top x exons (x-axis) for a 5 versus 5 comparison. For each method, the inner 95% range at which the FDR first becomes positive across the 100 simulations is given by the horizontal lines with the average marked by an asterisk. The data are simulated under (a) a double binomial distribution and (b) a beta-binomial distribution.

Table 4.1: **Percent of Exons Across the Possible Values for the Number of Non-boundary Samples in a 5 versus 5 setting.** Maximum value for each exon is 10 corresponding to no samples lying on the boundaries. Minimum value is 0 corresponding to an exon where all samples lie on the boundaries $\{0, 1\}$. n_{eff} is calculated as the maximum of the number of non-boundary samples and $K + 1$, where K is the number of groups. In a two group setting, K=2 corresponding to a minimum value of 3 for n_{eff}, given to exons with only 0, 1 or 2 non-boundary samples.

Data Set	Possible Values for Number of Non-boundary Samples (5 versus 5)										
Type	0	1	2	3	4	5	6	7	8	9	10
Double Bin. Simulation	0.02	10.81	9.51	8.43	7.70	7.45	7.10	7.47	7.97	10.43	23.12
Beta Bin. Simulation	0.00†	15.92	7.36	4.65	3.63	3.24	3.17	3.31	4.53	9.07	45.12
Tissue Data	1.13	10.95	8.60	7.78	7.50	7.44	7.19	7.62	8.67	10.58	22.52

†Percentage is exactly zero.



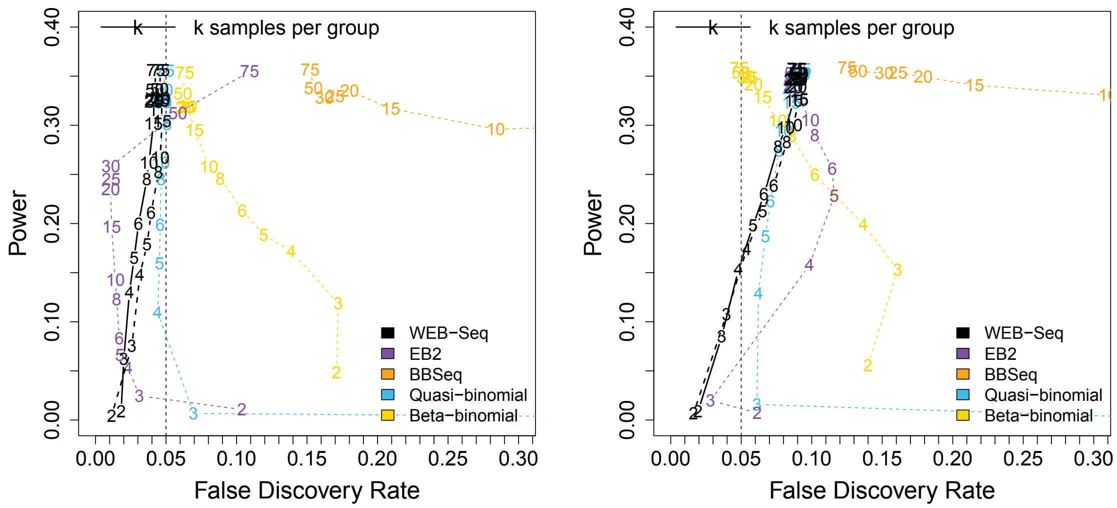
(a) Double-Binomial Simulation

(b) Beta-Binomial Simulation

Figure 4.8: **Effect of Boundary conditions** Plotted is the average proportion of false discoveries in the top x exons up to an FDR of 5% for a 5 versus 5 setting. The data are simulated under (a) a double binomial distribution and (b) a beta-binomial distribution. Each double-binomial method, with and without the boundary correction described in Section 3.3.4 are both plotted, with the dashed line corresponding to methods without the boundary correction. See the legend of Figure 4.7 of the main text for more general details about the plot.

estimates are the same between all tests. The LR test statistic as well as the WEB-Seq test statistic differ only in the calculation of the test statistic; both use the χ^2 distribution to calculate p-values. The Wald statistic is compared to the normal distribution for p-value calculation.

The likelihood-ratio procedure that tests for a difference in the proportion parameter compares the log-likelihood values between the experimental model fit that includes the experimental condition as a covariate and the null model, or intercept-only model, fit that excludes this covariate. The standard approach when a dispersion parameter is present is to estimate a single dispersion value that will be used in the likelihoods of both models but estimated under the experimental model. We implement a modification to this test statistic for WEB-Seq and DEB-Seq that estimates a dispersion under both the experimental and intercept-only model and uses these estimates within their corresponding likelihood function. In either approach, the resulting test statistic would then be tested for significance against a χ^2 or F distribution depending on whether or not the uncertainty in the estimate of dispersion is taken into account. In the case of shrinkage, the dispersion parameter is estimated using thousands if not tens of thousands of data points and so the uncertainty in the estimate is considered negligible and we therefore treat the dispersion parameter as a known constant. This allows us to use the χ^2 distribution for testing significance (see 3.4 for derivations of of the Wald and the standard and modified LR-statistics).



(a) Double-Binomial Simulation

(b) Beta-Binomial Simulation

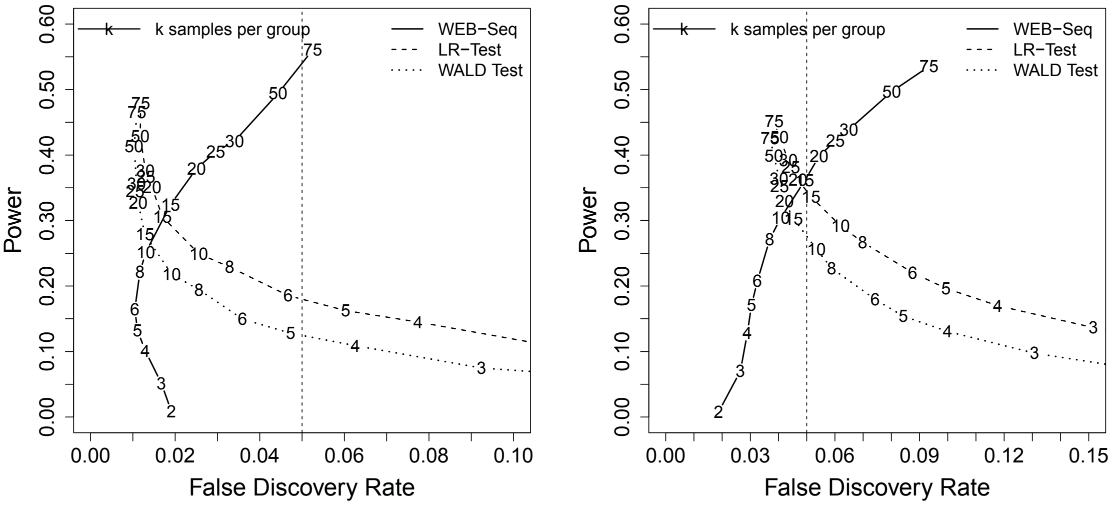
Figure 4.9: **Power vs. FDR under Filtering.** Plotted is the average Power (y-axis) against FDR (x-axis) over various sample sizes across 100 Double-Binomial simulations, based on p-values adjusted to provide a 5% FDR level. Exons with mean proportion across all samples that were in the top 5% and bottom 5% were removed from consideration. The data are simulated under (a) a double binomial distribution and (b) a beta-binomial distribution. WEB-Seq with and without the boundary correction described in Section are both plotted after filtering. See the legend of Figure 4.7 of the main text for more details.

Figure 4.10 compares these two test statistics and shows the advantage for estimating a dispersion separately for each model. In the double binomial simulation, WEB-Seq controls FDR for all sample sizes while the Wald and standard LR-test do not begin to control FDR until 5 versus 5 and 6 versus 6, respectively. Further, the Wald and LR-test become over-conservative at the cost of power and as a result WEB-Seq becomes more powerful for moderate to large sample sizes. In the beta-binomial simulation, the Wald and LR-test do not control FDR until 15 versus 15 whereas WEB-Seq controls FDR for all sample sizes up through 15 versus 15. In addition, WEB-Seq becomes more powerful at approximately 6 versus 6 despite it still controlling FDR while the other tests do not. Note, the more conservative F and t tests were also analyzed and found to control FDR but were highly conservative and had no power for small to moderate sample sizes and therefore were not deemed adequate.

4.2 Application to Real Data

4.2.1 Real Data Details

In order to have a reasonable setting for detecting differential alternative splicing, RNA-Seq data was downloaded from two different tumor types sequenced by the TCGA: Stomach



(a) Double Binomial Simulation

(b) Beta-Binomial Simulation

Figure 4.10: **Wald and LR tests.** Plotted are the FDR and Power results when using the standard Likelihood-Ratio test (dashed line) and Wald test (dotted line). The WEB-Seq method is also added for comparison. The data are simulated under (a) a double binomial distribution and (b) a beta-binomial distribution. The WEB-Seq method for shrinkage of the dispersion parameter is used for all tests. The only difference is in the test statistics. The standard LR-test only utilizes the dispersion estimate obtained under the experimental model for calculating the test statistic, as does the Wald. The WEB-Seq method is a LR test but utilizes both the experimental and intercept-only model dispersion estimates for calculating its test statistic. The Wald and LR tests fail to control FDR in low to moderate sample sizes as well as suffer from a lack of power in larger sample sizes compared to WEB-Seq.

and Ovarian. For comparisons between these two sets of tumors, we expect that there should be differences in alternative splicing due to the simple fact that the tumors originated from two different tissue types, and tissue-specific alternative splicing is well documented (Pan *et al.*, 2008).

In what follows, all data processing steps for the real data were carried out by Marla Johnson of the Purdom Lab at UC Berkeley.

RNA sequencing data was analyzed from 15 ovarian serous cystadenocarcinoma samples and 15 stomach-esophageal cancer samples (see Table 4.2 for sample ID's), which had been sequenced as part of The Cancer Genome Atlas (TCGA) project (Cancer Genome Atlas Research Network, 2011). The BAM files generated by TCGA were downloaded and converted to fastq files using the Picard tool SamToFastq. The fastq files were aligned to the human genome version GRCh37 using TopHat 1.4.1 (Trapnell *et al.*, 2009), and exon inclusion and exclusion counts were calculated for exons annotated by ENSEMBL version 66 (Flicek *et al.*, 2013). Isoform expressions for known transcripts were estimated using the reference annotation GRCh37.66 in Cufflinks 2.0.2 (Trapnell *et al.*, 2010).

Using the Ensembl annotation GRCh37.66, a set of unique, non-overlapping regions were generated across all transcripts of a gene, specifically regions of contiguous bases, all of which are annotated to be in the same transcripts by Ensembl. The UTR regions were separated from coding regions, so that the resulting non-overlapping regions would be either entirely UTR or entirely coding in each transcript; however, a single region defined as such could be UTR in one transcript and coding in another transcript.

By utilizing the pysam 0.6 module in Python 2.7.2, reads which fall within certain chromosomal positions could be found. For each properly paired, unique mapped read pair, this then allowed the region or set of regions for which each mate of the read pair aligned to be determined. For reads which spanned multiple regions, all junctions that occurred and disregarded the read if it contained an unknown junction were checked. Then the remaining right and left reads by pair were combined to generate the set of regions to which the read pairs mapped and calculated the number of occurrences of each possible set.

These sets of regions were then used to calculate the number of reads which aligned to each region allowing the generation of counts for each gene by summing across all regions in a gene. Also by using these alignments as well as the expected order of the regions given by the annotation, skipped regions and the frequency of these skipping events could be calculated. In addition, it was also possible to use these sets of regions to calculate the occurrences in which fragments spanned the borders of each region.

4.2.2 Real Data Results

We create a ‘null’ situation to compare the methods, where the two groups that are compared are both of the same tissue type. We note that these are tumor samples, so there may be differential alternative splicing in the different tumors even though they are the same tissue type, but since the two groups of samples are randomly chosen this is unlikely to be a significant factor. We compare the proportions of exons called significant in the null setting across methods, though we note that if we believe there are no significant exons this is not a measure of FDR directly, since *any* discoveries in an all-null setting would imply that the rate of false discoveries is 1.

We demonstrate the performance of the double binomial based shrinkage methods in Table 4.3. We see that all methods call almost no exons significant. We also see that like in the simulations, WEB-Seq has less ‘power’, i.e. the least significant calls under the real setting. This shows a possible weakness of the WEB-Seq method, where in the 2 versus 2 real comparison setting WEB-Seq lacks power and makes 0 calls. In the simulation we similarly saw that the WEB-Seq method was conservative, holding the FDR lower than necessary.

We also ran EB2, BBSeq and MATS on the TCGA data sets (Table 4.4). MATS could only be run in the null setting, as the stomach and ovarian samples were of different types and the MATS software could not handle this setting. BBSeq and EB2’s poor control of the FDR in the simulated data appears to be echoed in the real data. As demonstrated in Table 4.4, EB2 finds roughly 7% of the exons significant and BBSeq finds 3-14% significant. For comparison with MATS, we applied WEB-Seq to the inclusion/exclusion count matrices produced by MATS. In the null setting, MATS appears to have a call rate between 1.8%

Table 4.2: TCGA Samples used in the analysis of ovarian and stomach.

TCGA Barcode	UUID	Tissue Type
TCGA-09-0367-01A-01R-1564-13	047e04aa-6a91-4535-b279-50430098e638	Ovarian
TCGA-10-0936-01A-01R-1564-13	04112b9c-6ef4-49cd-bed8-43151b5bbe1f	Ovarian
TCGA-13-1405-01A-01R-1565-13	044d949f-0eb4-44c9-8327-90226936fc7c	Ovarian
TCGA-13-1411-01A-01R-1565-13	0069c63d-f699-41d4-9798-9a109d1c09df	Ovarian
TCGA-23-1030-01A-02R-1564-13	0cdd0255-c7b2-4cbc-8306-37e74d295324	Ovarian
TCGA-24-1564-01A-01R-1566-13	03b8480f-d19e-43e4-8a09-f9230c525c1e	Ovarian
TCGA-25-1312-01A-01R-1565-13	0d7561dc-7069-41df-ba49-8fcd4df52efb	Ovarian
TCGA-25-1321-01A-01R-1565-13	74ad9d3e-97a9-419d-8842-3fb37d3f099a	Ovarian
TCGA-25-1633-01A-01R-1566-13	7517c446-401d-4501-bf27-916e0835deac	Ovarian
TCGA-25-2404-01A-01R-1569-13	05ab6bb9-0cb1-47f1-8dd2-92a48239e82e	Ovarian
TCGA-31-1946-01A-01R-1568-13	05abb2f5-07c3-42da-92d9-0b4560fb6f09	Ovarian
TCGA-57-1585-01A-01R-1566-13	06df1324-bc1b-4bc3-bb01-4ef55a5aeef5	Ovarian
TCGA-61-1914-01A-01R-1567-13	0d75483f-eab3-41c8-b2cc-453af29b6f44	Ovarian
TCGA-61-1919-01A-01R-1568-13	056cca38-92df-42ff-b064-b0c243f8a82f	Ovarian
TCGA-61-1998-01A-01R-1568-13	0814c20e-fe77-4ff9-ab5e-a8ae2f74069d	Ovarian
TCGA-BR-4191-01A-02R-1131-13	439b3b31-fb3d-4373-a39b-384e30039217	Stomach
TCGA-BR-4255-01A-01R-1131-13	1e048ef6-c4e5-4d65-86d5-6d744703eab3	Stomach
TCGA-BR-4280-01A-01R-1131-13	1b245b59-ee33-45d3-b36b-976d4069c8e7	Stomach
TCGA-BR-4292-01A-01R-1131-13	27375592-f624-4f14-ad1e-01a443f6ea4a	Stomach
TCGA-BR-4361-01A-01R-1157-13	29942ecb-05d0-46d0-b4f6-0252ea19b8a2	Stomach
TCGA-BR-4369-01A-01R-1157-13	0c392e19-322b-4e2c-896f-198871324295	Stomach
TCGA-CG-4301-01A-01R-1157-13	7af442fb-3c5b-456a-bcfc-9c6c4fbc4053	Stomach
TCGA-CG-4304-01A-01R-1157-13	b2b49a63-b59d-4dce-91e2-0d85c2720560	Stomach
TCGA-CG-4436-01A-01R-1157-13	1af0a9d5-1d19-4666-948c-9aecc7555368	Stomach
TCGA-CG-4438-01A-01R-1157-13	7f2b32fa-a74e-491f-b43a-a0672098d949	Stomach
TCGA-CG-4442-01A-01R-1157-13	ba717f17-8371-4860-a332-29b17eea8470	Stomach
TCGA-CG-4462-01A-01R-1157-13	03ece93c-67f7-42b9-942f-653583ca3bee	Stomach
TCGA-CG-4469-01A-01R-1157-13	7e3bd01f-c9b9-4867-9315-0a5cddbb9dd0	Stomach
TCGA-CG-4475-01A-01R-1157-13	0d6f061b-119b-4bcf-9b1b-df7d7061af0d	Stomach
TCGA-CG-4477-01A-01R-1157-13	be12a5a1-933e-4d0a-86e5-050466cc2f9e	Stomach

and 3.9% (646 to 1,557 exons called significant), while WEB-Seq makes at most one call for any given sample size. These false positive rates do not directly compare with the FDR rates from the simulations, since FDR depends on the total number found significant. For comparison, if 10% of the exons were found significant and the method had 100% power, the false positive rate would have to be 0.6% to get an FDR of 5%, and in practice would need to be even lower since not all of the truly significant exons will be detected. A 3-7% false positive rate would then mean a minimum FDR of 21-38% and likely much higher. This indicates that the large rates of FDR shown in our simulation appear to be supported by implementation on the real data.

For the comparison of two different tissue types, we see many more calls made by BBSeq. The EB2 method, despite it's high false positive rate on the null setting, does not give many more calls than WEB-Seq, except in small samples sizes. Given the high false positive rate on the null set and our simulation results, it is likely that the additional calls of these methods represent a much higher level of false discoveries than the reported FDR control. Table 4.6 shows the propensity for the methods to make calls for exons whose log-fold-change is estimated to be infinite due to all samples of at least one group lying entirely on the boundary. For such exons, the WEB-Seq method will reduce their degrees of freedom via the n_{eff} parameter making it more difficult to make calls in these situations. The EB2 and quasi-binomial calls are highly influenced by these exons across all sample sizes, whereas DEXSeq, BBSeq, beta-binomial, and WEB-Seq are much less affected and display similar behavior as the sample size increases. Though these exons are not necessarily false positives since they can indicate the usage of an exon going from an off state to an on state, an over sensitivity to them may not be desirable, especially in low sample sizes when boundary values are more likely to be sampling artifacts as opposed to a true signal; again this corresponds well to the behavior we saw in the simulated results.

4.2.2.1 Alternative approach of DEXSeq

We made a further comparison of the performance of our method to another popular method of finding differential alternative splicing in exons, DEXSeq (Anders *et al.*, 2012). The DEXSeq framework is quite different than the inclusion/exclusion framework. This method assumes knowledge of the identification of exons to genes and fit a linear model per gene to the counts per exon, allowing for an individual exon effect i.e. how different an exon is from the overall mean gene expression. Then they find alternatively spliced exons by detecting exons who have different exon effects in the two groups. In fitting this model, they use a negative binomial model for the exon counts with shrinkage of the dispersion in the same manner as DESeq for gene expression.

We emphasize that DEXSeq is not just an alternative statistical method for exon counts, but uses significantly different aspects of the mRNA-Seq data, compared to the inclusion/exclusion setting, and the starting input data for the two approaches is entirely distinct: all counts overlapping an exon for DEXSeq, and the counts both overlapping and skipping an exon for inclusion/exclusion. DEXSeq does not make use of the information of junctions skipping the exons, except in their contribution to reads overlapping an exon. Further, it requires a gene model and would not be applicable in a setting where the gene models are not available, unlike the inclusion/exclusion approach. However, DEXSeq can in principle

Table 4.3: **Comparison of Double Binomial based Methods.** Shown in the table below are the percentage of and total exons called significant from the Tissue Data under the null and real scenarios described above for the methods we developed based on the double binomial distribution. The total number of exons is $412,002$. The rates are percentages out of only those exons that had at least one skipping event, a number which varies with sample size but is roughly 1/4 of all exons.

Percentage of Exon Calls

Sample	WEB-Seq		DEB-Seq		Wt-Likelihood	
Size	Real	Null	Real	Null	Real	Null
2 vs 2	0.00†	0.00†	0.00†	0.00†	0.12	0.00†
3 vs 3	0.48	0.00†	1.38	0.00†	1.39	0.00†
4 vs 4	3.67	0.00†	4.45	0.00†	3.55	0.00†
5 vs 5	2.94	0.00†	3.92	0.00†	3.18	0.00†
6 vs 6	5.35	0.00†	6.00	0.00†	4.74	0.00†
7 vs 7	6.97	0.00†	7.38	0.00†	5.91	0.00†

†Percentage is exactly zero.

Total Calls

Sample	WEB-Seq		DEB-Seq		Wt-Likelihood		Total #
Size	Real	Null	Real	Null	Real	Null	of Exons
2 vs 2	0	0	7	0	421	0	88,117
3 vs 3	1,728	0	4,942	0	4,977	0	98,900
4 vs 4	13,190	0	15,968	0	12,749	0	106,208
5 vs 5	10,562	0	14,092	0	11,429	0	109,684
6 vs 6	19,219	0	21,545	0	17,033	0	113,002
7 vs 7	25,026	0	26,500	0	21,221	0	115,354

find differential usage of exons that do not have inclusion/exclusion data resulting from their alternative usage, for example exons that are removed upstream from the beginning of transcripts and/or downstream of the end of transcripts. For these reasons, it is not clear that you can make a reasonable comparison between WEB-Seq and DEXSeq. However, DEXSeq is a popular method for detecting alternative splicing with just exon counts, so we attempt some basic comparisons.

When we compare our methods to DEXSeq, we note that the paradigm of inclusion, exclusion offers one possibly significant advantage regardless of the statistical method. In the inclusion, exclusion paradigm, those exons that show no reads skipping the exon in any sample of any group are naturally excluded (by getting a p-value of 1 by definition). Because of the difference in exons evaluated between the methods, we first concentrate on comparing the performance of DEXSeq for just the same set of exons that are used in

Table 4.4: **Comparison to Competing Methods.** Shown in the table below are the percentage of and total exons called significant from the Tissue Data under the null and real scenarios described above. DEXSeq was post-filtered to have the same set of exons as the inclusion/exclusion setting. For all the results shown below, except for MATS, the total number of starting exons is $412,002$ but the rates are percentages out of only those exons that had at least one skipping event, a number which varies with sample size but is roughly 1/4 of all exons. The results from MATS are based on a different set of exon data produced internally by MATS, roughly 35,000 exons; WEB-Seq results are not shown on this set of exons, but WEB-Seq makes at most one significant call on the MATS set of exons (for sample sizes 3, 5 & 7) and zero for other sample sizes.

Percentage of Exon Calls

Sample	DEXSeq		EB2		BBSeq		MATS	WEB-Seq	
Size	Real	Null	Real	Null	Real	Null	Null	Real	Null
2 vs 2	2.62	0.00	3.99	6.88	7.18	3.63	3.45	0.00[†]	0.00[†]
3 vs 3	11.29	0.01	4.47	7.15	9.38	8.07	1.77	0.48	0.00[†]
4 vs 4	20.86	0.00	4.64	6.68	10.72	5.60	2.45	3.67	0.00[†]
5 vs 5	16.56	0.00	4.43	6.47	11.59	7.14	2.74	2.94	0.00[†]
6 vs 6	22.11	0.02	4.79	6.48	11.98	14.87	3.93	5.35	0.00[†]
7 vs 7	26.99	0.01	4.87	6.18	12.44	14.27	3.39	6.97	0.00[†]

[†] Percentage is exactly zero.

Total Calls

Sample	DEXSeq		EB2		BBSeq		MATS	WEB-Seq	
Size	Real	Null	Real	Null	Real	Null	Null	Real	Null
2 vs 2	3,291	3	14,312	8,624	25,773	4,587	1,168	0	0
3 vs 3	14,163	28	16,049	8,970	33,694	10,117	646	1,728	0
4 vs 4	26,163	11	16,671	8,377	38,507	7,065	936	13,190	0
5 vs 5	20,772	6	15,911	8,110	38,649	7,642	1,070	10,562	0
6 vs 6	27,728	86	17,185	8,130	41,605	17,729	1,557	19,219	0
7 vs 7	33,846	34	17,504	7,754	43,035	17,792	1,363	25,026	0

	Percentage of Exon Calls				Total Calls			
Sample	Quasi-bin.		Beta-bin.		Quasi-bin.		Beta-bin.	
Size	Real	Null	Real	Null	Real	Null	Real	Null
2 vs 2	0.07	0.02	0.00	0.00[†]	282	95	2	0
3 vs 3	0.20	0.02	2.85	0.00[†]	804	98	11,738	0
4 vs 4	3.40	0.02	5.34	0.00[†]	14,006	67	22,007	0
5 vs 5	3.82	0.01	3.74	0.00[†]	15,751	58	15,401	0
6 vs 6	6.29	0.03	5.56	0.00[†]	25,895	125	22,887	0
7 vs 7	7.64	0.04	6.75	0.00[†]	31,497	165	27,798	0

[†] Percentage is exactly zero.

Table 4.5: **DEXSeq Analysis** Shown in the table below are the percentage of and total exons called significant for the Tissue data under the null and real scenarios described above and with the proper application of DEXSeq. For all the results shown below the rates are percentages out of the total number of exon, $412,002$.

DEXSeq

Sample	% of Exon Calls		Total Calls	
Size	Real	Null	Real	Null
2 vs 2	3.94	0.00	16,229	19
3 vs 3	14.53	0.02	59,871	93
4 vs 4	25.48	0.01	104,966	23
5 vs 5	21.15	0.00	87,144	9
6 vs 6	25.95	0.03	106,920	112
7 vs 7	31.43	0.01	129,498	40

Table 4.6: **Percent of Calls Out of Total Significant for Exons with an Infinite Log-Fold-Change.** Shown below are the percentage of significant calls for which the exon has a log-Fold-Change of the odds between the groups that is infinite, corresponding to a situation where all samples of at least one of the two groups lie on the boundary, either all 1 or all 0. For DEXSeq, the percentage is based on the significance results after post-filtering to the DEXSeq results to the same set of exons as the inclusions/exclusion counts, i.e. those exons with non-zero skipping in some sample. For 2 vs 2, WEB-Seq has no significant calls, so the percentage is not defined.

Sample Size	DEXSeq	EB2	BBSeq	Quasi-bin.	Beta-bin.	WEB-Seq
2 vs 2	0.43	12.42	0.18	11.70	0.00†	-
3 vs 3	0.78	12.78	0.55	14.68	1.21	0.23
4 vs 4	0.87	12.64	0.71	4.58	1.61	0.42
5 vs 5	1.74	18.33	0.00†	9.38	3.36	0.76
6 vs 6	1.77	11.16	1.11	6.85	2.87	1.25
7 vs 7	1.39	7.57	1.66	4.66	2.24	1.27
10 vs 10	1.91	4.55	1.93	5.14	2.77	2.28
15 vs 15	3.06	1.88	2.03	5.64	2.96	3.92

†Percentage is exactly zero.

WEB-Seq; namely, we run DEXSeq on all exons, as required by the algorithm, and then filter out those not found to have any skipping events. For this set of exons the performance of DEXSeq to WEB-Seq in the null setting is roughly equivalent, while DEXSeq calls many more exons significant in the real setting (Table 4.4). Similar relative performance is seen in DEXSeq in its proper application using all exons (see Table 4.5).

To get a sense of the value of observing reads that skip an exon, we can compare to the annotation used, here Ensembl version 66, where some exons are annotated as constitutive (i.e. should not be skipped in any of the transcripts if the annotation is completely accurate) and others as alternative. Of the constitutive exons (12.7% of the exons in the data), only 1% (529 exons) show *any* reads skipping the exon in *any* of the 30 tissue samples, while 35.0% of those annotated as alternatively spliced show skipping. This strongly suggests that the implicit removal of exons with no skipping junctions is preferentially removing null exons, which ultimately can increase the power (Bourgon *et al.*, 2010a). There is no natural way to exclude such exons in the DEXSeq model since the constitutive exons are actually important in building the gene model, though the post-analysis filtering we described above could be implemented to eliminate exons that were not skipped.

Clearly this implicit filtering can be a disadvantage if many alternatively spliced exons are excluded because of a lack of sufficient reads to detect the skipping event. We view this as less of a practical disadvantage because we find that in practice practitioners are likely to want evidence in the form of junction reads skipping an exon to have faith in calling an exon alternatively spliced. But more generally, because the inclusion/exclusion paradigm relies heavily on reads that span the junctions of exons, which are a small percentage of all reads, a criticism of the inclusion/exclusion paradigm is that it relies on a lower number of reads and could have lower power. It is clear in the real data comparison that DEXSeq makes more significant calls than WEB-Seq even when limited to the same set of exons.

It is difficult to directly evaluate whether the additional calls made by DEXSeq are on average finding more true discoveries than false ones. Comparing exon calls to the annotation is one way of roughly assessing the performance for calling differential exon usage: about 12% of constitutive exons are called significant by DEXSeq (Table 4.7). This is roughly their total representation in the data so DEXSeq does not appear to be preferentially finding exons annotated to be alternatively spliced. However, directly comparing the exons found by DEXSeq with the annotation has the problem that the method is designed to detect only differential usage as compared to the average usage of all exons in the gene as opposed to the actual exon that is alternatively spliced; these could be different, for example, if many of the exons in a gene are alternatively spliced so that relative to the mean the unusual exon are the few that are not alternatively spliced, a point the authors of DEXSeq make as well (Anders *et al.*, 2012). Using this logic, we instead compare only exons that are the sole exon called significant in their gene; when these “single-exon” significance calls are compared to the annotation, even a larger percentage are annotated as constitutive *and* furthermore have no reads skipping them in the data for any of the 30 samples (18%-46%, Table 4.7). In comparison, in WEB-Seq, 0.2% of the significant exons (or 76 exons) are annotated as constitutive and all of them, by definition, have reads skipping them to at least justify the call of significance (this calculation is based on all exons WEB-Seq analyzes since it is reasonable to directly compare all the calls made by WEB-Seq to the annotation, not just “single-exon” calls).

Table 4.7: **Percent of single-exon calls made by DEXSeq, by annotation and skipping event.** 'AS', 'NSkC' and 'SkC' stand for an 'Alternatively Spliced' exon, a 'Non-Skipped Constitutive' exon, and a 'Skipped Constitutive' exon, respectively, where the designation of an alternatively spliced and constitutive exon is made using the Ensembl GRCh37.66 annotation.

	% of Total Single-Exon Calls				% of Total Significant Calls			
Sample Size	2 vs 2	3 vs 3	5 vs 5	15 vs 15	2 vs 2	3 vs 3	5 vs 5	15 vs 15
AS	81.75	70.74	67.70	54.26	85.89	87.20	87.01	88.11
NSkC	18.00	29.06	32.12	45.74	13.88	12.58	12.79	11.72
SkC	0.25	0.19	0.18	0.00†	0.23	0.22	0.20	0.17
Total Calls	2,356	2,085	1,681	916	16,229	59,871	87,144	186,570

†Percentage is exactly zero.

We can also evaluate the data properties of the significant exons to evaluate whether they demonstrate data characteristics that would lead us to trust the call. In Figure 4.13 we compare the density of the log-fold-change between the groups of the odds-ratio of skipping an exon for the significant calls made by both methods. WEB-Seq clearly has a much stronger tendency to find exons with large differences in the skipping proportion, which is not surprising given that that is the basis of its test statistic, unlike DEXSeq. More striking is that for DEXSeq there are significant peaks at 0, indicating many of the exons found significant by DEXSeq do not show evidence of differential exon usage in the form of a difference in the proportion of skipping counts. The constitutive exons, in particular, are completely centered at zero. This could be because of the lack of identification of the correct exon, explained above; when we examine the "single-exon" genes which are presumed to target the appropriate exon, these exons show slightly greater propensity to be removed from zero (Figure 4.11c).

Ultimately, we find the inclusion/exclusion paradigm, as implemented with our methods, concentrates the analysis on those exons with tangible evidence of alternative splicing as well as directly highlighting the specific exons of interest. We suspect this will also be an effective way of preventing a large source of false discoveries as well as being robust to the behavior of the other exons in the gene.

4.2.2.2 Computational Constraints

In an exon usage analysis, the method used needs to potentially be able to handle all exons, a number which for the human genome can be in the hundreds of thousands. Under the inclusion/exclusion paradigm there are natural filters that significantly reduce the set of exons under analysis and we have seen ranges between 40K and 200K exons in this scenario for real RNA-Seq experiments when looking at just protein coding exons. In the gene model based, exon usage analysis carried out by DEXSeq such filters are not appropriate and therefore the number of exons being analyzed can range between 300K and 400K

Table 4.8: **Computation Times (Hours): Inclusion\Exclusion Setting** Computation times for the methods WEB-Seq and DEXSeq in <u>*hours*</u> where exons in the DEXSeq analysis were pre-filtered to contain the same exons as WEB-Seq. The amount of exons analyzed for each sample size are given in the last line.

Method	Sample Size							
	2v2	3v3	4v4	5v5	6v6	8v8	10v10	15v15
WEB-Seq	0.001	0.004	0.005	0.007	0.007	0.009	0.009	0.012
DEXSeq	2.44	3.07	3.70	4.01	4.44	5.16	5.70	7.12
# of Exons	88,117	98,900	106,208	109,684	113,002	117,785	119,926	125,398

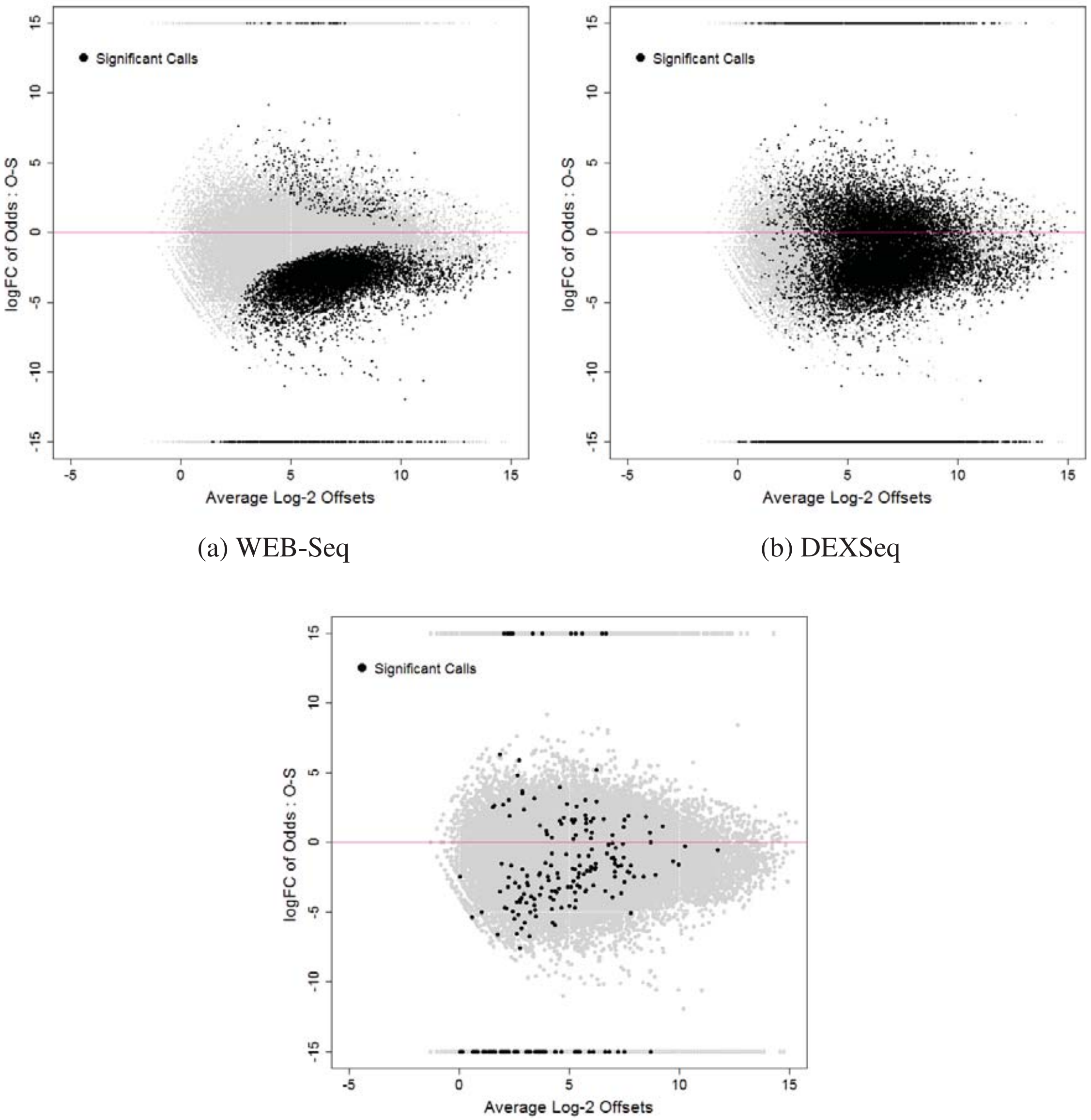
Table 4.9: **Computation Times (Hours): DEXSeq** Computation times in <u>*hours*</u> for a proper implementation of DEXSeq. The amount of exons analyzed for each sample size are given in the last line.

Method	Sample Size							
	2v2	3v3	4v4	5v5	6v6	8v8	10v10	15v15
DEXSeq	26.82	29.27	34.79	38.40	39.09	39.28	40.49	42.34
# of Exons	328,750	348,326	360,892	367,675	376,334	385,542	389,127	399,766

exons. A comparison of computation times in hours was done using a single core computer running a AMD Opteron 6272, 2.1 GHz processor. For proper comparison purposes, we present two tables. Table 4.8 shows computational times for DEXSeq and WEB-Seq using the inclusion/exclusion filters and we see that what would take DEXSeq several hours to process only takes a fraction of a minute for WEB-Seq. Since in practice these filters are not valid for DEXSeq, Table 4.9 shows computational times for a proper implementation of DEXSeq using their own default filters and the times are significantly increased by a factor of 10. We note that DEXSeq allows for the use of multiple cores to speed up computation. As for MATS, since they start with the raw BAM files the full analysis takes anywhere from 20-40 hours on a single core depending on sample size, though multiple cores can be used. The extra-binomial (EB2) method is relatively fast taking only 2-5 minutes to complete.

4.2.2.3 Implementation

The WEB-Seq and DEB-Seq methods are available for download on `CRAN` under the title `DoubleExpSeq`. DEXSeq was run under version 1.8.0 of the `DEXSeq` R package using default settings. The EB2 method was run under version 2.1 of the `extraBinomial` R package using default settings. MATS version 3.0.8 was used with standard parameters and is available on their website. BBSeq was run under version 1.0 of the `BBSeq` R package using default settings. The beta-binomial model was estimated via the `free.estimate` function provided in the `BBSeq` package. The quasi-binomial model was estimated via the `glm` function available in the `base` package of R. R version 3.0.2 was used for all analyses.

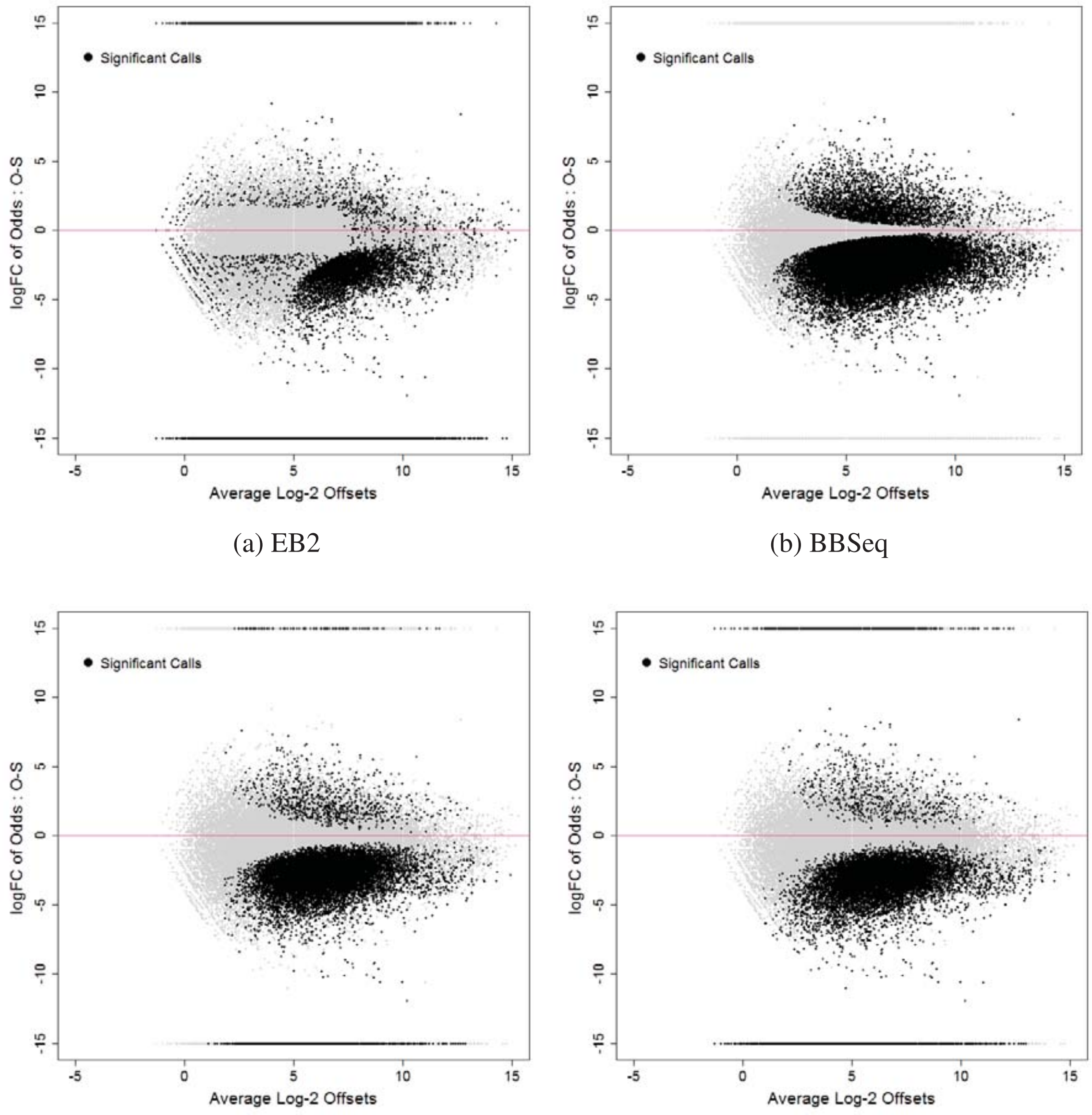


(a) WEB-Seq

(b) DEXSeq

(c) DEXSeq, only calls of single-exon per gene

Figure 4.11: **MA Plots for WEB-Seq, DEXSeq:** Plotted is log of the odds-ratio (y-axis) against the average of the total counts per exon, on the log-scale (x-axis). Data is from a comparison of 5 vs. 5. Points in grey represent all exons, while those in black are those found significant by the method. (a) WEB-Seq, (b) DEXSeq, all significant exons, (c) DEXSeq, only calls from exons which are the single significant exon in their gene.



(a) EB2

(b) BBSeq

(c) Beta-binomial

(d) Quasi-binomial

Figure 4.12: **MA Plots for EB2, BBSeq, Beta-binomial and Quasi-binomial:** Plotted is log of the odds-ratio (y-axis) against the average of the total counts per exon, on the log-scale (x-axis). Data is from a comparison of 5 vs. 5. Points in grey represent all exons, while those in black are those found significant by the method.

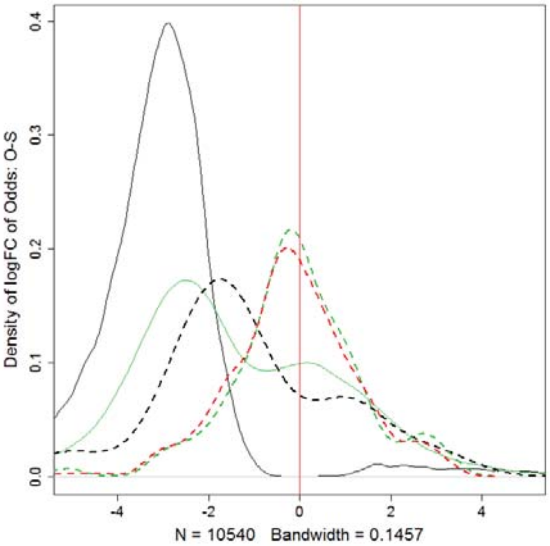


Figure 4.13: **Density of odds-ratio for WEB-Seq and DEXSeq**. A smoothed density plot of the log of the odds-ratio is plotted for different subsets of exons, where DEXSeq is shown in green and WEB-Seq is shown in black. The dashed lines are exons annotated to be constitutive, while solid lines are annotated to be alternatively spliced. The red density line is all 431 constitutive exons that have skipping in the 5v5 samples.

Chapter 5

Application to Differential Gene Expression

Differential gene expression has played an integral part in advancing the field of genomics and has provided the motivation for the development of several different statistical techniques for RNA-Seq data. Though our `WEB-Seq` and `DEB-Seq` methods discussed in Chapter 3 were developed and tailored specifically for detecting differential exon usage, we find it natural to compare their performance among the many alternatives in the gene setting. The results show our methods do not do as well as many others in controlling the FDR which was not the case in the exon usage analysis. However, similar to the exon usage setting, the quality of the gene ranks by p-value rivals the competitors and for some simulations is superior to all other methods. We find this to be true even in very low sample sizes where other methods fail to maintain high quality calls in their top ranked genes.

In addition to benchmarking our `WEB-Seq` method, this chapter also serves to be a comparison of RNA-Seq gene expression methods in general, and we find `voom` (Law *et al.*, 2014) which was outlined in section 2.5.2 to be the clear top performer overall compared to 14 other methods including `WEB-Seq` in both simulated and real data tests. We note that this is also the conclusion drawn by another, more in-depth comparison paper by Soneson and Delorenzi (2013). However, they further claim that `voom` has low to medium power relative to other methods which we do not see evidence of in our simulations or real data results.

Along with simulated data sets, we also apply these methods to mRNA-Seq gene counts from the same tumor samples generated by the Cancer Genome Atlas project (Cancer Genome Atlas Research Network, 2011) that we refer to as the Tissue data in the exon usage analysis (see section 4.2.1). For the simulated data we measure each methods' ability to control the FDR, to make calls for truly different genes (power), and their ability to rank these genes above non-differential genes. For the real data, we simulate the same "null" and "real" setting as we did for the exon usage analysis and draw comparisons between these results and those seen in the simulations.

5.1 Simulations

5.1.1 Simulation Details

To reflect RNA-Seq gene expression data, counts were simulated using the gene counts from the Tissue data described in section 4.2.1 as a starting point. For each simulation, data for $20,000$ genes was generated: $18,000$ under a null setting ($\beta_1 = 0$) and $2,000$ exhibiting differential expression for a two group comparison (non-null setting). This was done under a negative binomial distribution, a double poisson distribution, and a beta binomial distribution, corresponding to two overdispersed poissons and an overdispersed binomial, respectively. For each of these distributions, a two group simulated data set was created with 100 samples per group. This data set was then subsampled across the samples to mimic data at different sample sizes, ranging from a 2 versus 2 scenario to 30 versus 30.

Unless otherwise specified, each step laid out here is done for each of the 3 distributions to create 3 separate simulated data sets. To generate count data, the mean and dispersion need to be determined for each gene and a library size for each sample. For the differentially expressed genes a separate mean is needed for each condition, though it is assumed the conditions share the same dispersion parameter. To obtain values for the mean and dispersion parameters, MLE estimates for each gene were calculated within tissue type taking into account library sizes and then were randomly sampled as a pair with replacement. Two-hundred library size parameters were randomly sampled from the actual library sizes, one for each simulated sample, and were used across all 3 distributions. For the null setting, the same normalized mean parameter that was sampled for each gene is used across their respective 200 samples. For the non-null setting, a log linear model was fit to the Tissue data to estimate the effect size of the group for each gene but only for the negative binomial distribution. Two-thousand effect sizes were then randomly sampled independent of the mean and dispersion parameters and only from the set lying in $[-5, -0.5]$ or $[0.5, 5]$, and the resulting sample of effect sizes was used for the non-null genes across all 3 distributions. The second group mean for the non-null genes was calculated using their corresponding effect sizes in accordance to the relationship implied by each distribution's typical link function: the $\log$ function in the case of the negative binomial and double poisson, and the logit function in the case of the beta-binomial. The 200 sampled library sizes were then used to re-scale the mean parameter values across the 200 samples. At this point, all the parameters have been assigned values for all 20,000 genes across the 200 samples for each distribution and are used to separately generate negative binomial, double poisson, and beta-binomial data.

We used these three simulated data sets to evaluate our `WEB-Seq` and `DEB-Seq` methods (refer to section 3.3) applied to both the double poisson model and double binomial model (refer to sections 3.1.4 and 3.1.3) since overdispersed poissons and overdispersed binomials are equally valid in this context. We also implemented a myriad of other RNA-Seq methods for comparison: `baySeq`, `DESeq`, `DESeq2`, `DSS`, `EBSeq`, `edgeR`, `NBPSeq`, `PoissonSeq`, `SAMseq`, `sSeq`, `TSPM`, `voom`, `vst+limma` (Hardcastle and Kelly, 2010; Anders and Huber, 2010; Love *et al.*, 2014; Wu *et al.*, 2013; Leng *et al.*, 2013; Robinson and Smyth, 2007; Di *et al.*, 2011; Li *et al.*, 2012; Li and Tibshirani, 2013; Yu *et al.*, 2013; Auer and Doerge, 2011; Law *et al.*, 2014; Smyth, 2004). For a brief description of these methods refer to section 2.5. We also implement the standard quasi-binomial using the R function `glm` provided in the `stats` package and apply it to each gene independently without any shrinkage.

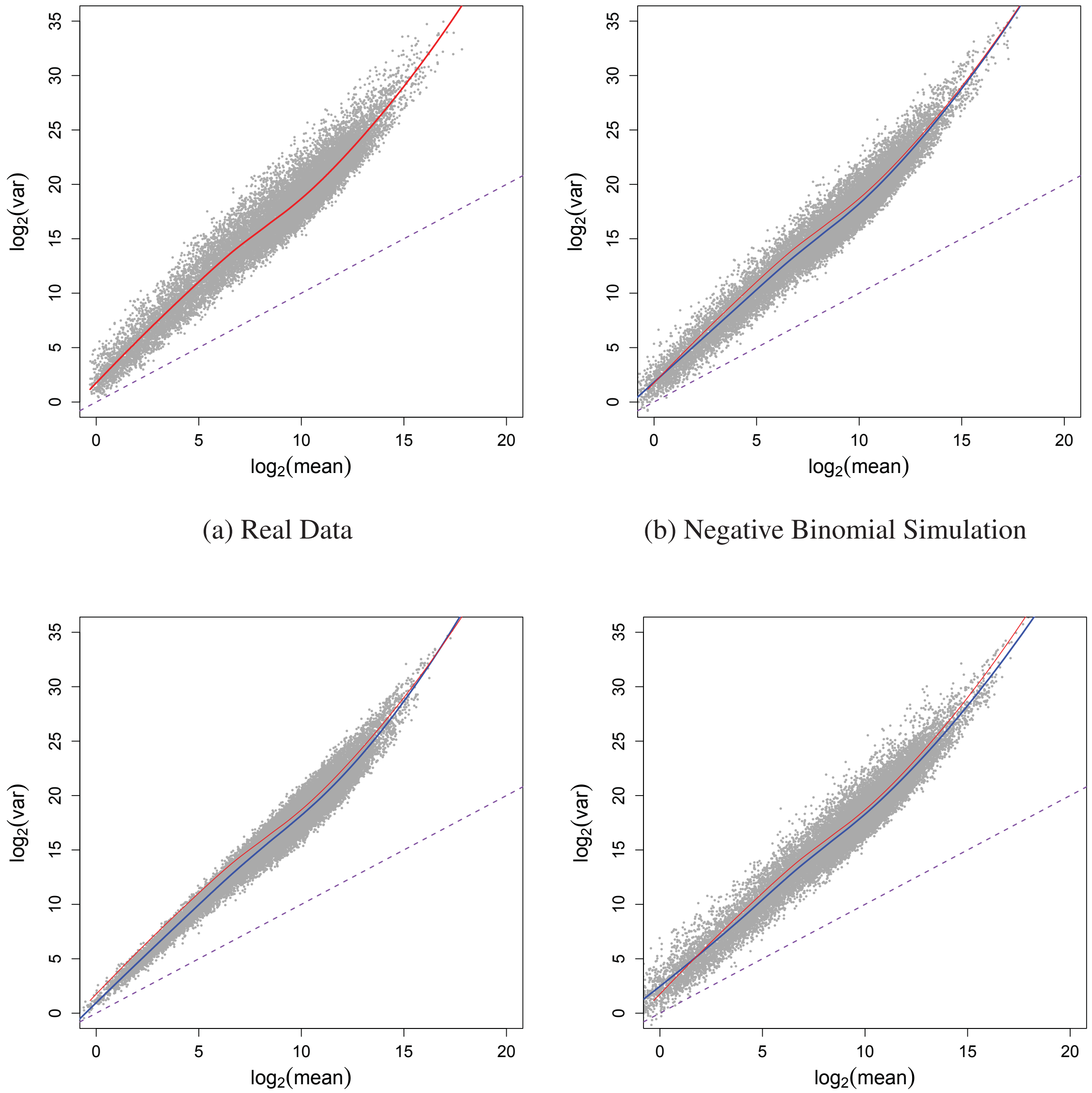
For each method, their own default internal normalization procedure was used since for some of the methods their default performed better than alternatives. For our methods we obtained the best results using the TMM (section 2.4.2) method also used by `edgeR` (Robinson and Oshlack, 2010). For methods that generated p-values, these were adjusted to control the FDR to a 0.05 level using the standard Benjamini-Hochberg FDR procedure (Benjamini and Hochberg, 1995) implemented in the `p.adjust` function in R (R Core Team, 2013).

5.1.2 Results of Simulation for Double Exponential Shrinkage

Figure 5.2 shows the false discovery rate plotted against power for different sample sizes and simulations for our shrinkage methods. Comparing the double binomial based methods to the double poisson we see that there is practically no difference between the two methods in terms of FDR and rankings. As shown in section 3.1.5, the double binomial converges to the double poisson when the number of trials is large and the probability of success is small and as expected we see this holds for RNA-Seq data.

One very noticeable difference is in the behavior of `WEB-Seq` compared to `DEB-Seq` (green versus black). The FDR for `WEB-Seq` starts off well below the 5% level and climbs as the sample size increases, whereas `DEB-Seq` is the opposite and starts off around 25%, decreasing as the sample size increases; and this is regardless of the simulation and the assumed double exponential distribution. Investigating the cause, we find that the test statistic for `DEB-Seq` is almost always larger on a per-gene basis than `WEB-Seq` resulting in more calls and therefore a higher chance for false discoveries. Further we note that both methods rank the genes very similarly (Figure 5.3) and quite effectively suggesting the issue is due to a higher cutoff point for `DEB-Seq` as opposed to a worse ordering of the genes by the test statistic. In fact, `DEB-Seq` consistently makes several more calls, and when these additional calls beyond the amount made by `WEB-Seq` are removed the FDR results become nearly identical.

`WEB-Seq` is clearly better than `DEB-Seq` but it still fails to control the FDR at the desired level except for very small sample sizes, ≤ 3. Our moderated likelihood ratio test statistic (see section 3.4.1), though still superior to the standard version (not shown), is still too liberal in the amount of calls it makes. Using an F or t distribution which is commonly used to correct for the estimation of a variance parameter results in an major over correction and therefore a severe drop in power making these tests useless. Filtering the data set to remove low count genes only results in a slight drop in FDR, and using various normalization techniques did not have a marked effect. We do see, however, that we have significantly better control of FDR and better convergence in the double poisson data than the other distributions. This is not surprising since the model assumes the correct distribution up to the normalization constant being equal to 1. All this suggests that though `WEB-Seq` does well under its own model, the method in the context of gene expression analysis is not robust to departures from that model in terms of FDR. However, as we will show in the next section, the quality of the ranks by p-value compared to other methods is



(a) Real Data

(b) Negative Binomial Simulation

(c) Double Poisson Simulation

(d) Beta-binomial Simulation

Figure 5.1: **Mean-Variance Relationship** Plotted is the variance of the total count (y-axis) against the mean of the total count (x-axis) per gene on the log scale with a loess line drawn for each plot to reflect the mean-variance relationship in a 15 versus 15 setting. The red line in both plots represents the fit to the *real* Tissue data. The purple line signifies the poisson fit to the data (variance=mean). (a) is plotted using the real Tissue data set. (b) is plotted from the Negative Binomial simulation. (c) is plotted from the Double Poisson simulation. (d) is plotted from the Beta-binomial simulation.

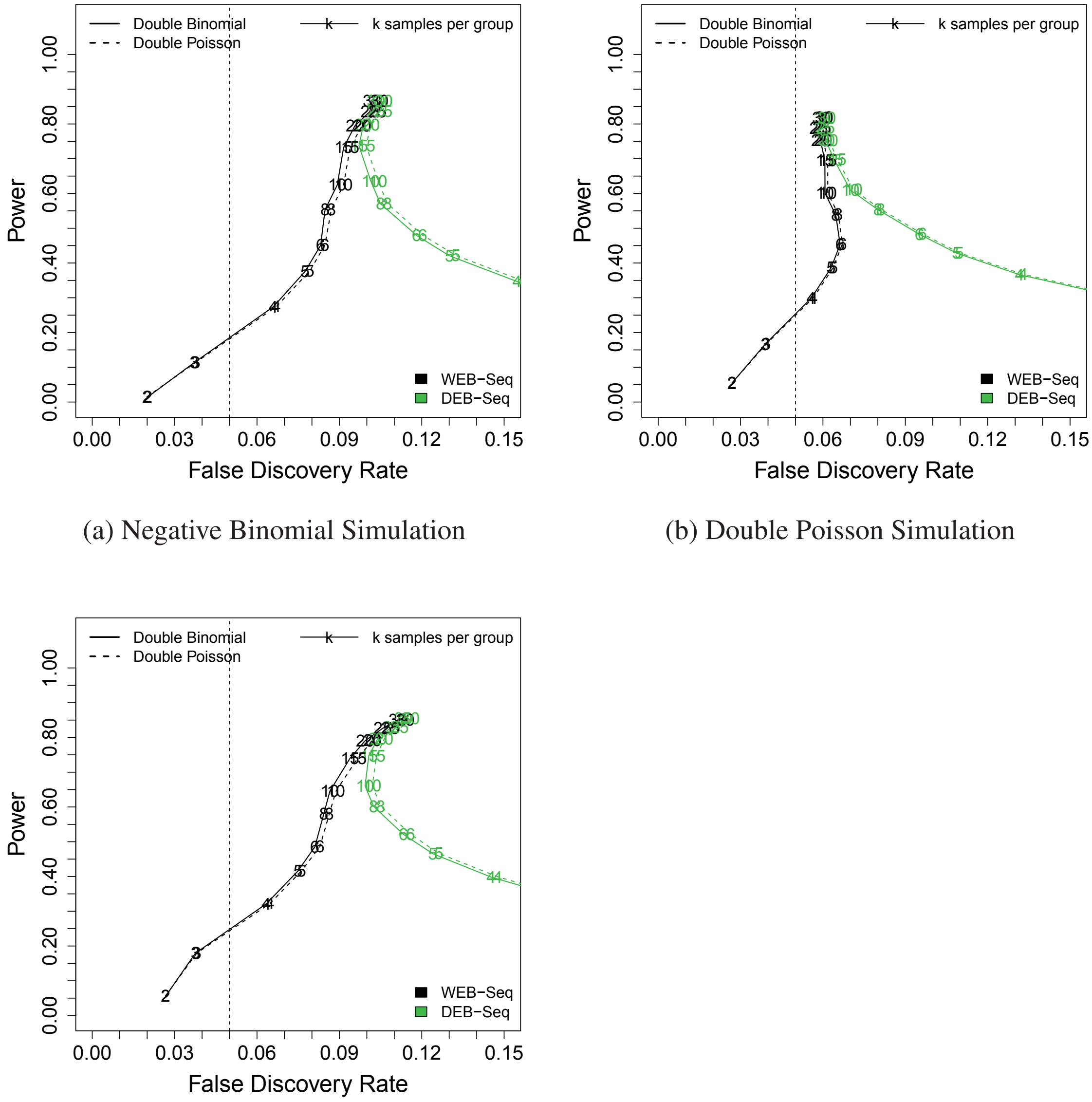
very robust to both small sample sizes and count data that is different from the assumed distribution.

5.1.3 Results of Simulation for Alternative Methods

Here we present various other methods for differential gene expression in RNA-Seq data, all of which are discussed in sections 2.5 and 2.5.8. We also carry over our top performer from the previous section, `WEB-Seq`, for comparison. Figure 5.4 shows the average FDR and power across 100 simulations for each sample size for all of the methods mentioned in section 5.1.1. The best method in terms of FDR control and power is clearly `voom` across all simulations. The combination of a variance stabilizing transform and limma, `vst+limma`, is a close second but does have an FDR slightly above 5% in the beta-binomial simulation for sample sizes ≥ 3, and even then shows less power than `voom`. `SAMseq` and `DESeq` also do well in terms of both FDR and power but only starting around 6 samples per group and still have less power than `voom`. `baySeq` does well in the negative binomial simulation for sample sizes ≥ 4 but shows erratic behavior in the other simulations and fails to control the FDR for several sample sizes. The only method that rivals `voom` in terms of consistent FDR control is the standard quasi-binomial that is fit to each gene separately and does not use any form of shrinkage or any additional steps beyond its typical use. Its performance agrees with the general thought that quasi-likelihood methods are robust and applicable in many settings. It does however lack power overall compared to `voom`.

The methods `edgeR`, `DESeq2` and `EBSeq` do very poorly and never control the FDR in the negative binomial and beta-binomial simulations until very large sample sizes. They do, however, show much better performance in the double poisson simulation and control FDR starting at a sample size of 5, 15 and 6, respectively. `PoissonSeq`, `TSPM`, `DSS`, `sSeq`, and `NBPSeq` never control the FDR. Out of the four, `TSPM` performs the best and also outperforms our method `WEB-Seq` starting at a sample size of 5 per group. `DSS` and `sSeq` have minimum FDRs across all simulations and sample sizes of 25% and 13%, respectively and therefore do not fit within the plot regions. `NBPSeq` only fits within the plot regions for the double-poisson simulation.

The rank plots given in Figures 5.5 and 5.6 confirm the poor performances of `TSPM`, `PoissonSeq`, `DSS`, `sSeq` and `NBPSeq` as they all tend to make false calls in the top genes. `edgeR` which had trouble in terms of FDR control actually does relatively well in terms of ranks but still falls short of being a top performer. `baySeq` which showed good control of the FDR in the negative binomial simulation appears to make false discoveries among its top ranked genes in the same simulation for both the 5 versus 5 and 8 versus 8 sample sizes. `baySeq` actually does significantly better in terms of ranks for the other simulations and has competitive performance with the other top methods. `SAMseq` appears to make false discoveries among its top ranked genes in both 5 versus 5 and 8 versus 8 scenarios. This can be clearly seen in the horizontal bars in the rank plots which indicate the positions of the first false discovery across the 100 simulations. In the 5 versus 5 scenario it fails to make it onto the plot altogether due to having several false discoveries in its highest ranks; though this is not unexpected given that `SAMseq` is a nonparametric method and needs a moderate amount of samples before it begins to perform well. In the 8 versus 8

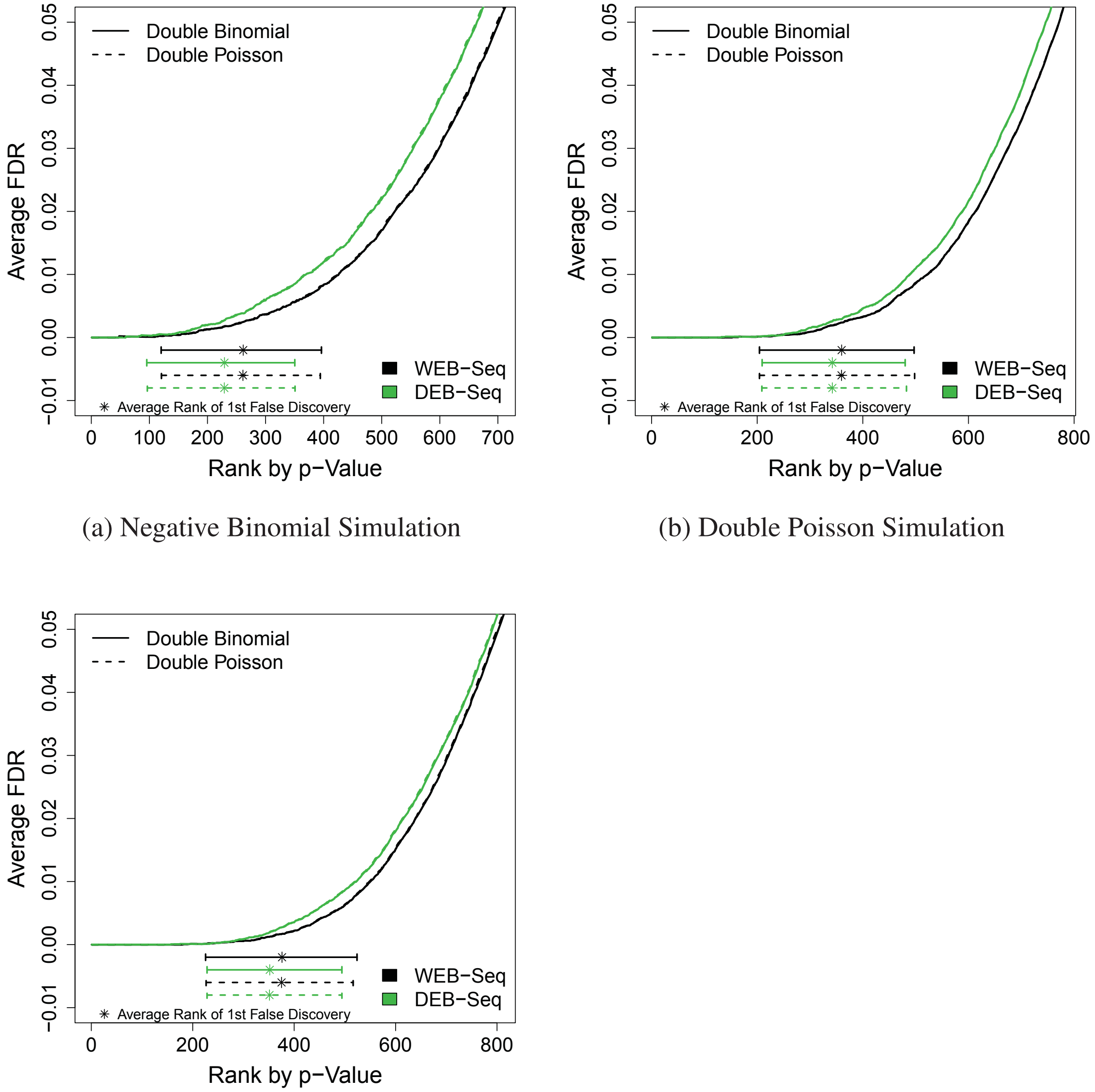


(a) Negative Binomial Simulation

(b) Double Poisson Simulation

(c) Beta-binomial Simulation

Figure 5.2: **Double Exponential Methods: FDR and Power** Plotted is the average Power (y-axis) against FDR (x-axis) over various sample sizes across 100 simulations, based on p-values adjusted to provide a 5% FDR level. The results for a single method across different sample sizes are connected by a line. The numbers that overlay a method denote the power and FDR for that specific sample size (*per group*) in a 2 group comparison. The 5% FDR boundary is given by the dotted vertical line. The data are simulated under (a) a Negative Binomial distribution, (b) a Double Poisson distribution, and (c) a Beta-binomial distribution. The shrinkage methods shown are all based on a double exponential distribution, either a double binomial denoted by a solid line or a double poisson denoted by a dashed line to account for over-dispersion, and using either a 1-parameter empirical bayes (`WEB−Seq`) or a 2-parameter empirical bayes (`DEB−Seq`).



(a) Negative Binomial Simulation

(b) Double Poisson Simulation

(c) Beta-binomial Simulation

Figure 5.3: **Double Exponential Methods: False Discoveries by Rank** Plotted is the average proportion of false discoveries (y-axis) in the top x exons (x-axis) for a 5 versus 5 comparison. For each method, the inner 95% range at which the FDR first becomes positive across the 100 simulations is given by the horizontal lines with the average marked by an asterisk. The data are simulated under (a) a Negative Binomial distribution, (b) a Double Poisson, and (c) a Beta-binomial distribution.

scenario, though it performs much better, SAMseq appears to still maintain a propensity to make false calls in the very top ranks, yet quickly corrects itself in the lower ranks. The quasi-binomial method, though proving to have good control of the FDR, shows a lack in ability to rank truly differential genes among its top calls. The lack of power seen from the quasi-binomial method therefore suggests it achieves FDR control despite the rankings by being conservative in the number of total calls made compared to other methods.

Our method WEB-Seq clearly outperforms all other methods in the double poisson simulation as well as in the very top ranks for the beta-binomial simulation for the 5 versus 5 and 8 versus 8 simulation as demonstrated by the average rank of the first false discovery. The voom method does best in the beta-binomial simulations in terms of power at the 5% FDR cutoff, though as just mentioned WEB-Seq performs better in about the first 600 calls for both sample sizes. In the negative binomial simulation, DESeq2, vst+limma and voom all perform roughly equally with DESeq2 edging out the others. Though the status of top performer in terms of the quality of the ranks is shared across the simulations by WEB-Seq, voom and DESeq2, they all exhibit robust performance in every simulation and sample sizes shown here. Recall, however, that voom is the only method that also performs robustly in terms of FDR control.

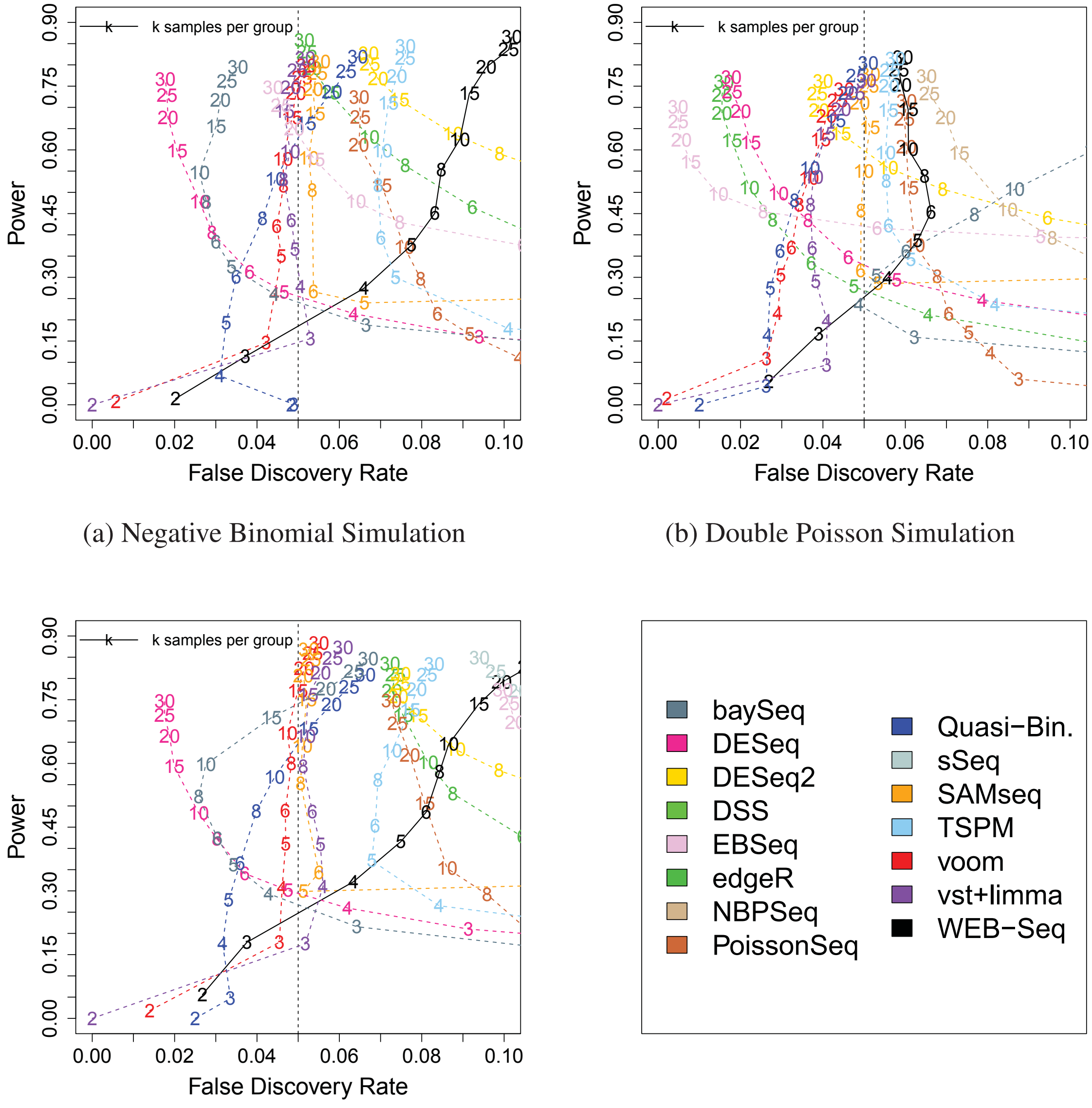
5.1.3.1 Very Low Sample Size

Though very low sample sizes is becoming more and more rare as the cost of sequencing technologies decreases, they are not entirely absent from the research community. For that reason, we show the ranking results for the smallest sample size of 2 versus 2 in Figure 5.7. In addition, these plots show the limits of effectiveness for each methods' shrinkage approach. We see that WEB-Seq and voom stand out above the others in their ability to maintain high quality calls relative to the other methods as well as control the FDR (Figure 5.4). In particular, they both show a marked distance from 0 for their average first false discovery as well as maintaining a relatively high level of power at the 5% FDR cutoff. DESeq2 and vst+limma though demonstrating similar ranking abilities in moderate sample sizes fail to maintain that ability in low sample sizes.

5.2 Application to Real Data

5.2.1 Tissue Data

We create a 'null' situation to compare the methods, where the two groups that are compared are both of the same tissue type. We remind the reader that these are tumor samples, so there may be differential expression in the different tumors even though they are the same tissue type, but since the two groups of samples are randomly assigned this is unlikely to be a significant factor. We compare the average proportions of genes called significant in the null setting across the methods for 100 simulations at different sample sizes. We note that if we believe there are no significant genes this is not a measure of FDR directly, since *any* discoveries in an all-null setting would imply that the rate of false discoveries is 1. This is rather a measure of the false positive rate (FPR). We also show the average percentage of genes called significant across 100 simulations in the real setting that compares the true



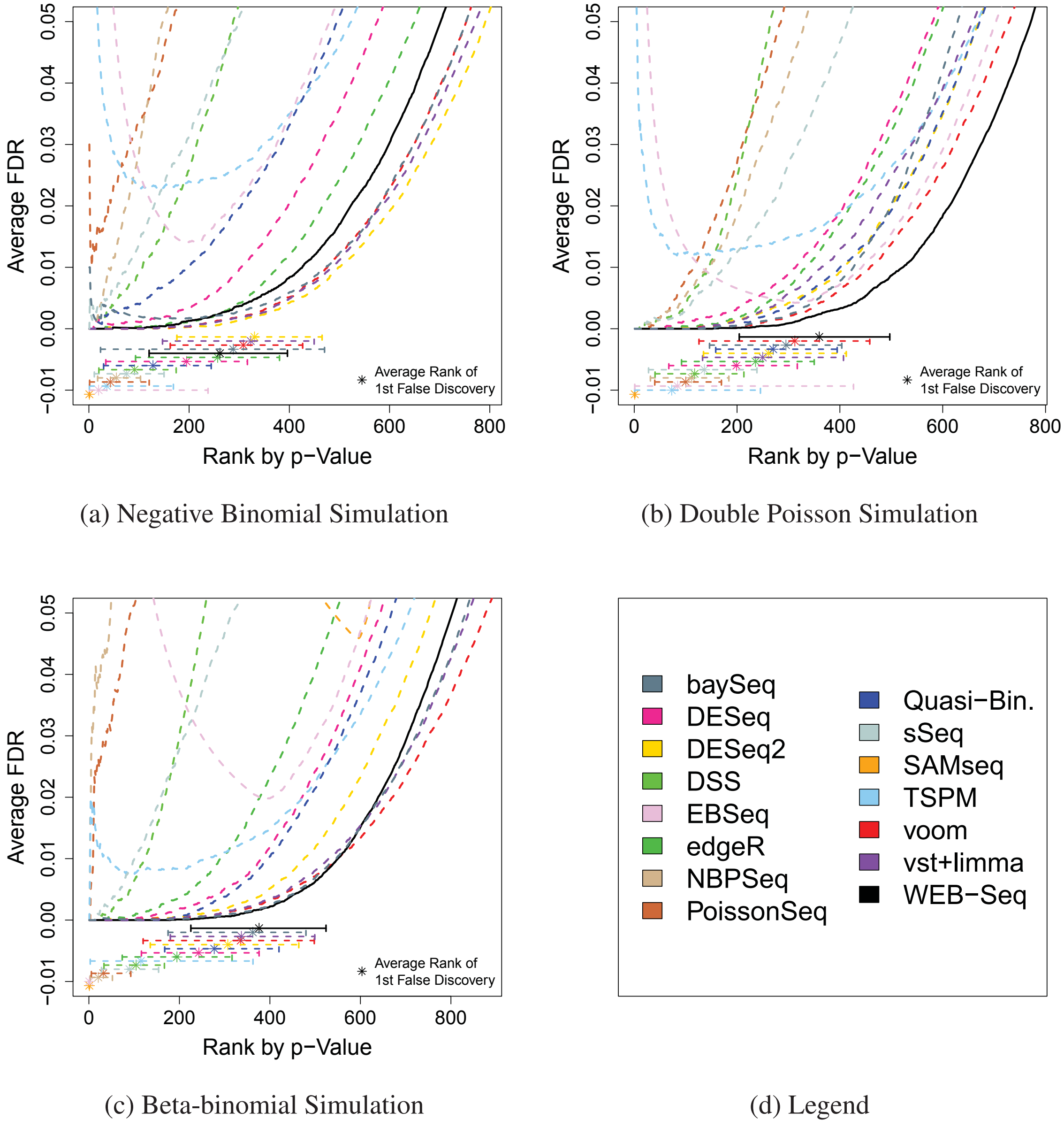
(a) Negative Binomial Simulation

(b) Double Poisson Simulation

(c) Beta-binomial Simulation

(d) Legend

Figure 5.4: **Alternative Methods: FDR and Power** Plotted is the average Power (y-axis) against FDR (x-axis) over various sample sizes across 100 simulations based on p-values adjusted to provide a 5% FDR level (see Figure 5.2 for details). The data are simulated under (a) a Negative Binomial distribution, (b) a Double Poisson distribution, and (c) a beta-binomial distribution; and a legend is provided in (d). The following alternative methods, both those that perform shrinkage and those that do not, are compared to `WEB-Seq`: `baySeq`, `DESeq`, `DESeq2`, `DSS`, `EBSeq`, `edgeR`, `NBPSeq`, `PoissonSeq`, `Quasi-Bin.`, `SAMseq`, `sSeq`, `TSPM`, `voom`, `vst+limma`



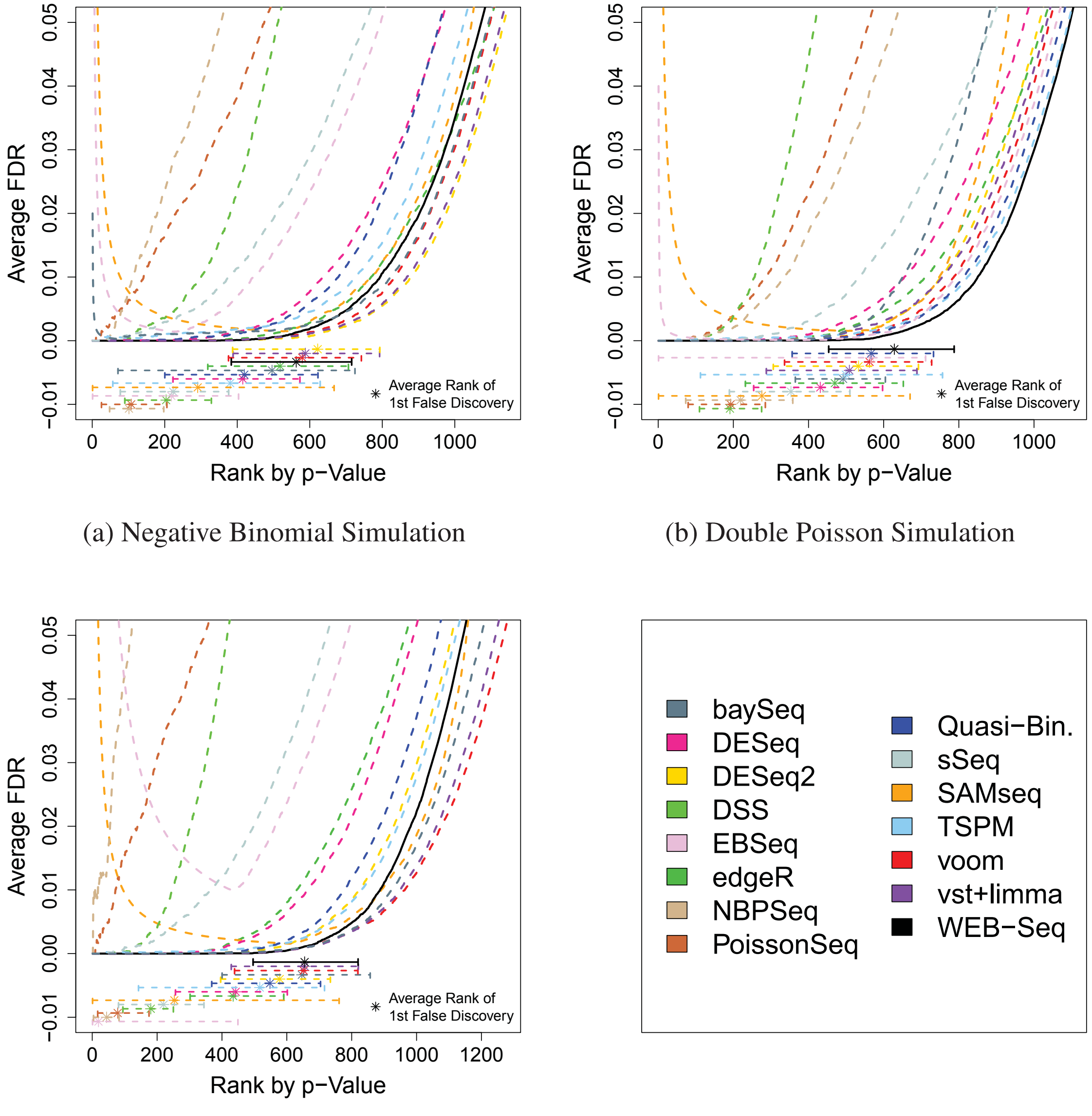
(a) Negative Binomial Simulation

(b) Double Poisson Simulation

(c) Beta-binomial Simulation

(d) Legend

Figure 5.5: **Alternative Methods: False Discoveries by Ranks** Plotted is the average proportion of false discoveries (y-axis) in the top x exons (x-axis) for a 5 versus 5 comparison. For each method, the inner 95% range at which the FDR first becomes positive across the 100 simulations is given by the horizontal lines with the average marked by an asterisk. The data are simulated under (a) a Negative Binomial distribution, (b) a Double Poisson, and (c) a Beta-binomial distribution; and a legend is provided in (d).



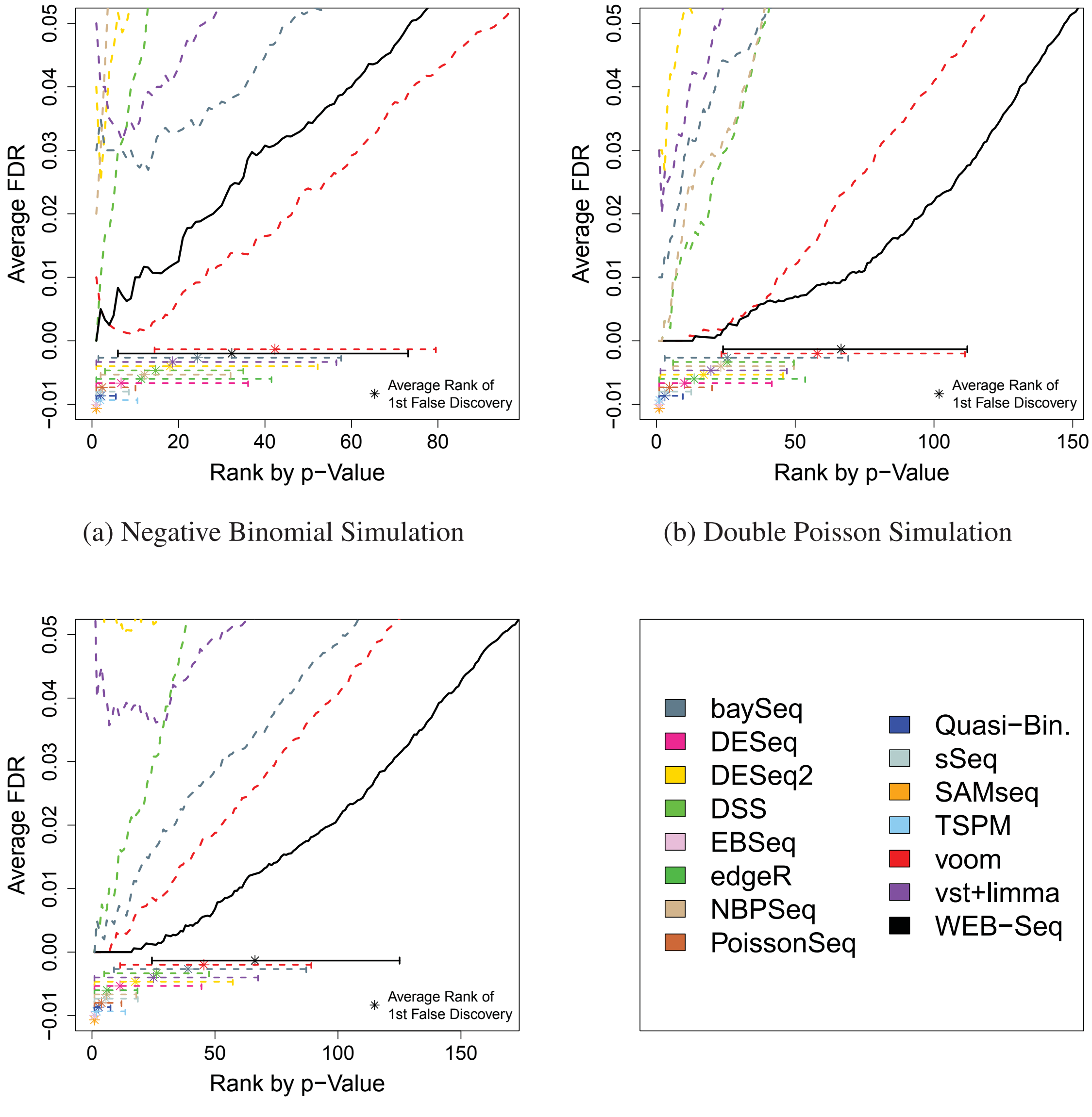
(a) Negative Binomial Simulation

(b) Double Poisson Simulation

(c) Beta-binomial Simulation

(d) Legend

Figure 5.6: **Alternative Methods: False Discoveries by Ranks** Plotted is the average proportion of false discoveries (y-axis) in the top x exons (x-axis) for a 8 versus 8 comparison. For each method, the inner 95% range at which the FDR first becomes positive across the 100 simulations is given by the horizontal lines with the average marked by an asterisk. The data are simulated under (a) a Negative Binomial distribution, (b) a Double Poisson, and (c) a Beta-binomial distribution; and a legend is provided in (d).



(a) Negative Binomial Simulation

(b) Double Poisson Simulation

(c) Beta-binomial Simulation

(d) Legend

Figure 5.7: **Very Low Sample Size: False Discoveries by Ranks** Plotted is the average proportion of false discoveries (y-axis) in the top x exons (x-axis) for a 2 versus 2 comparison. For each method, the inner 95% range at which the FDR first becomes positive across the 100 simulations is given by the horizontal lines with the average marked by an asterisk. The data are simulated under (a) a Negative Binomial distribution, (b) a Double Poisson, and (c) a Beta-binomial distribution; and a legend is provided in (d).

stomach samples to the ovarian at different samples sizes. Though this is not directly a measure of power it serves as a relative comparison to the null setting as well as across methods.

The methods that clearly do best are our method WEB-Seq, voom, vst+limma, DESeq, sSeq, and quasi-binomial. Except for WEB-Seq and sSeq, all of them performed satisfactorily in the simulations. WEB-Seq, however, does not appear to be making a higher number of false calls as the simulation results would suggest. In fact, out of all shrinkage methods which excludes quasi-binomial, WEB-Seq performs the best in the null setting and for some sample sizes does better than the quasi-binomial while simultaneously having a higher number of calls.

It is difficult to analyze the quality of the rankings or calls since there is no gold standard, however, we can compare the overlap in genes called significant in the real setting between our top performers in the simulations in terms of rank. Figure 5.8 is a venn diagram showing the amount of overlap between WEB-Seq, vst+limma and voom in a real setting for a randomly chosen 5 versus 5 data set. voom makes the most number of calls followed by vst+limma and then WEB-Seq, and this agrees with what we observe in the tables. Compared to voom, WEB-Seq only makes 319 unique calls which represents 13% of the total calls made by WEB-Seq and of these only 21 genes lie inside the top 50% of calls ranked by p-value. vst+limma also has a high agreement with voom and similary only finds 12% unique compared to voom. This result agrees with their similar abilities to rank differentially expressed genes as seen in the simulated data sets.

Many methods that performed badly in the simulations also tend to perform badly in the null setting. For example, DESeq2 did well as far as rankings but failed to control the FDR in the simulations. This would suggest that its cutoff point for making calls is too liberal which agrees with what is seen in the null and real setting here. edgeR and NBPSeq which overall could not control the FDR also appear to have relatively high false positive rates. DESeq which showed control of the FDR but with relatively low power also appears to be true here though to a larger degree.

Some methods performed quite differently from their results in the simulations. The most striking is that of SAMseq which performed well in simulations starting around 5 samples per group but in the null setting it has the highest false positive rates even in larger sample sizes. In addition, it also calls the most genes significant in the real setting which in combination with behavior in the null setting suggests it is too liberal in its calls. This surprising result may suggest that SAMseq requires even larger sample sizes in a real setting compared to simulations to perform as expected for nonparametric approaches. PoissonSeq and TSPM also appear to have slightly different behavior relative to each other compared to their simulation results. In the simulations, TSPM controlled the FDR better and had higher power than PoissonSeq. However, PoissonSeq makes many more calls suggesting that PoissonSeq should have a much greater false positive rate than TSPM in the null setting which it does not. Nevertheless, both of these methods still only show mediocre performance as was the case in the simulations.

Another important factor when considering which method to implement is computation time. Table 5.3 shows computation times in minutes for each method across various sample sizes for 19,742 genes. Some methods appear to be only slightly affected by large increases in sample sizes while others are greatly affected like baySeq, DESeq, NBPSeq, sSeq,

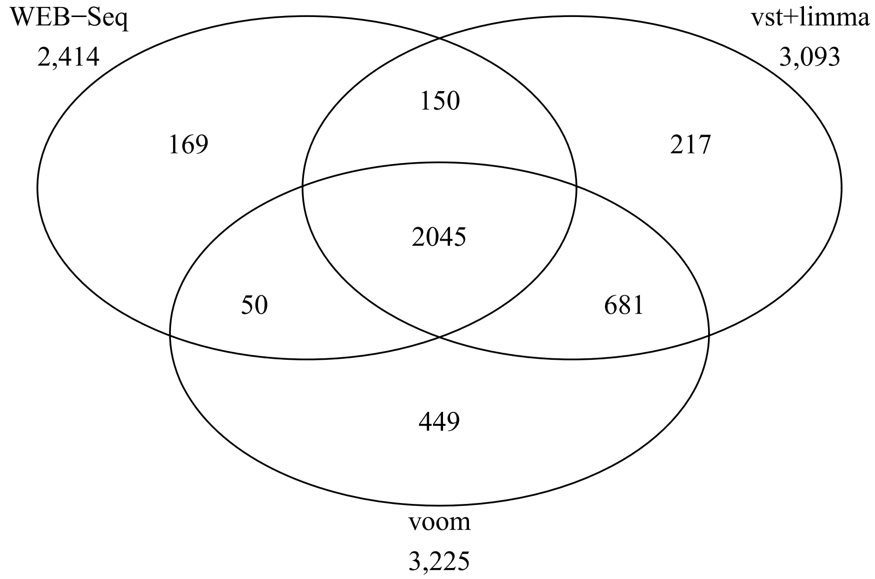


Figure 5.8: **Venn Diagram of Significant Genes** A venn diagram of the overlap of genes called significant between WEB-Seq, vst+limma and voom methods under the real scenario for a 5 versus 5 setting.

and `EBSeq`. The top performing methods `voom`, `vst+limma`, and `WEB-Seq`, fortunately require a negligible amount of time but we note the fastest methods are `WEB-Seq` and `voom`.

5.3 Conclusion

There are many methods to choose from in the differential gene expression arena. A method that demonstrates high power; control of FDR; is quick, easy and interpretable; robust; and preferentially ranks truly differential genes in the top calls are all qualities that biologists and statisticians look for—`voom` clearly meets all of these criteria. In addition, `voom` offers the flexibility and theoretical foundation that only normal linear models can offer which gives the experimenter readily accessible options in both the experimental design and analysis of the data, and we find this to be a huge advantage over the other methods presented here. `DESeq2` and `vst+limma` also perform well but both are clearly inferior to `voom` and so offer no advantage. Our method `WEB-Seq` performs well in many regards including in very low sample size situations. It is, however, too liberal in the amount of calls made in larger sample sizes which results in a lack of FDR control. We note that the WEB-Seq method has been tailored specifically to exon usage and not to the unique properties of gene expression. With further development, our WEB-Seq method may show better results. All other competing methods show a lack of ability to rank truly different genes in their top calls and we find this to be the worst quality in a method since biologists tend to select the top ranked genes for further study.

Table 5.1: **Null and Real Analysis: Percent Significant** Shown in the table below is the average percentage of genes called significant across 100 sub-sampled data sets from the Tissue data under the null and real scenarios described above. For all the results shown below the total number of genes is $19,742$.

Percentage of Gene Calls

Sample	WEB-Seq		voom		vst+limma		edgeR		DESeq	
Size	Real	Null	Real	Null	Real	Null	Real	Null	Real	Null
2 vs 2	0.01	0.00	0.44	0.00†	0.00†	0.00†	7.88	0.60	1.89	0.13
3 vs 3	1.24	0.00†	6.48	0.00	6.00	0.00	13.68	0.35	2.74	0.02
4 vs 4	10.55	0.01	15.17	0.01	15.03	0.00	19.58	0.36	3.87	0.01
5 vs 5	19.36	0.00	22.89	0.00	22.85	0.02	25.46	0.20	5.45	0.00
6 vs 6	26.62	0.00	28.97	0.00	28.68	0.00	30.72	0.25	7.38	0.00
7 vs 7	32.44	0.00	33.79	0.03	33.93	0.00	35.03	0.32	9.67	0.00

†Percentage is exactly zero.

Percentage of Gene Calls

Sample	DESeq2		baySeq		DSS		EBSeq		NBPSeq	
Size	Real	Null	Real	Null	Real	Null	Real	Null	Real	Null
2 vs 2	14.69	1.38	4.58	0.09	16.54	1.87	23.01	6.01	9.27	1.73
3 vs 3	19.53	0.54	7.24	0.07	13.76	0.25	17.62	1.61	14.76	1.91
4 vs 4	26.04	0.45	11.64	0.04	17.60	0.14	17.92	0.61	19.27	2.09
5 vs 5	31.77	0.17	16.92	0.06	23.01	0.04	19.33	0.47	23.67	2.30
6 vs 6	36.41	0.18	22.61	0.07	28.46	0.03	22.14	0.47	27.57	2.63
7 vs 7	40.07	0.21	26.65	0.10	34.12	0.04	23.77	0.49	31.20	2.73

†Percentage is exactly zero.

Percentage of Gene Calls

Sample	sSeq		SAMseq		PoissonSeq		TSPM		Quasi-Bin.	
Size	Real	Null	Real	Null	Real	Null	Real	Null	Real	Null
2 vs 2	1.43	0.01	21.90	21.46	8.55	0.07	3.03	2.04	0.00	0.00
3 vs 3	3.71	0.01	14.03	26.64	13.28	0.06	1.87	0.22	0.01	0.00
4 vs 4	7.22	0.01	23.75	18.79	18.64	0.05	6.33	0.05	1.13	0.00
5 vs 5	12.53	0.01	35.16	22.41	22.52	0.01	15.09	0.03	6.44	0.00
6 vs 6	17.42	0.00	40.77	17.78	27.15	0.01	22.73	0.01	13.08	0.00
7 vs 7	21.02	0.01	46.34	14.79	31.32	0.02	29.43	0.01	19.52	0.00

†Percentage is exactly zero.

Table 5.2: **Null and Real Analysis: Total Significant** Shown in the table below is the average number of genes called significant across 100 sub-sampled data sets from the Tissue data under the null and real scenarios described above. For all the results shown below the total number of genes is $19,742$.

Total Gene Calls

Sample	WEB-Seq		voom		vst+limma		edgeR		DESeq	
Size	Real	Null	Real	Null	Real	Null	Real	Null	Real	Null
2 vs 2	1	0	87	0	0	0	1,557	118	358	24
3 vs 3	245	0	1,280	0	1,184	0	2,701	70	524	3
4 vs 4	2,082	2	2,996	3	2,968	0	3,865	71	744	2
5 vs 5	3,821	0	4,519	0	4,512	4	5,026	39	1,052	0
6 vs 6	5,255	0	5,719	0	5,662	0	6,065	50	1,428	0
7 vs 7	6,405	1	6,670	6	6,699	0	6,916	64	1,876	0

Total Gene Calls

Sample	DESeq2		baySeq		DSS		EBSeq		NBPSeq	
Size	Real	Null	Real	Null	Real	Null	Real	Null	Real	Null
2 vs 2	2,775	258	904	17	3,265	370	4,344	1,126	1,714	318
3 vs 3	3,450	95	1,430	14	2,716	50	3,362	305	2,768	355
4 vs 4	4,525	78	2,297	9	3,475	27	3,443	116	3,642	391
5 vs 5	5,468	29	3,340	12	4,542	7	3,731	90	4,501	433
6 vs 6	6,235	31	4,465	14	5,619	5	4,282	91	5,259	497
7 vs 7	6,840	37	5,260	20	6,736	8	4,607	96	5,969	517

Total Gene Calls

Sample	sSeq		SAMseq		PoissonSeq		TSPM		Quasi-Bin.	
Size	Real	Null	Real	Null	Real	Null	Real	Null	Real	Null
2 vs 2	281	2	4,323	4,236	1,530	12	598	404	0	0
3 vs 3	732	3	2,769	5,259	2,430	10	369	44	2	0
4 vs 4	1,426	1	4,690	3,709	3,447	9	1,250	9	222	0
5 vs 5	2,473	3	6,940	4,423	4,200	2	2,979	6	1,271	0
6 vs 6	3,439	1	8,050	3,509	5,071	2	4,486	2	2,582	0
7 vs 7	4,150	2	9,149	2,921	5,859	3	5,811	1	3,853	0

Table 5.3: **Computation Times (Minutes)** Computation times in minutes for all gene methods explored in this chapter are given at various sample sizes.

Method	Sample Size				
	2 vs 2	5 vs 5	10 vs 10	15 vs 15	30 vs 30
WEB-Seq	0.0	0.1	0.1	0.1	0.1
voom	0.1	0.1	0.1	0.2	0.2
edgeR	0.1	0.1	0.2	0.4	0.6
PoissonSeq	0.1	0.2	0.2	0.4	0.6
DSS	0.3	0.3	0.3	0.4	0.5
SAMseq	0.1	0.5	1.2	1.7	2.7
vst+limma	0.6	0.8	1.1	1.4	2.9
DESeq2	0.7	0.9	1.1	1.4	2.7
EBSeq	0.9	2.0	3.7	5.1	9.4
TSPM	4.5	4.7	5.2	5.1	5.6
NBPSeq	2.0	4.5	6.8	9.4	18.5
DESeq	3.3	6.9	10.8	15.7	29.3
sSeq	3.7	6.8	11.2	16.2	30.7
baySeq	18.7	23.1	23.4	24.8	46.0

Table 5.4: **Method Information** For each method analyzed in this chapter the version numbers are given along with any additional important information.

Method	R Package	Version	Notes
baySeq	baySeq	1.16.0	default settings
DESeq	DESeq	1.14.0	default settings
DESeq2	DESeq2	1.2.10	default settings
DSS	DSS	1.8.0	default settings
EBSeq	EBSeq	1.2.0	default settings
edgeR	edgeR	3.4.2	“tagwise” setting
NBPSeq	NBPSeq	0.1.8	default settings
PoissonSeq	PoissonSeq	1.1.2	”twoclass” setting
sSeq	sSeq	1.0.0	default settings
SAMseq	samr	2.0	“Two class unpaired” setting
TSPM	No R package	no version #	default settings
voom	limma	3.18.13	default settings
vst+limma	limma & DESeq2	–	DESeq2 provided the transform

5.4 Implementation

Here we present a table of documenting the version numbers for each of the methods as well as any important processing steps. R version 3.0.2 was used for all analyses.

Bibliography

Anders, S. and Huber, W. (2010). Differential expression analysis for sequence count data. *Nature Precedings*. doi:10.1038/npre.2010.4282.2. URL `http://precedings.nature.com/documents/4282/version/2`.

Anders, S., Reyes, A., and Huber, W. (2012). Detecting differential usage of exons from RNA-seq data. *Genome Research*, **22**(10), 2008–2017. ISSN 1549-5469. doi:10.1101/gr.133744.111. URL `http://dx.doi.org/10.1101/gr.133744.111`.

Auer, P. L. and Doerge, R. W. (2011). A Two-Stage Poisson Model for Testing RNA-Seq Data. *Statistical Applications in Genetics and Molecular Biology*, **10**(1), 1–26. ISSN 1544-6115.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, **57**(1), 289–300. URL `http://www.jstor.org/stable/2346101`.

Bolstad, B., *et al.* (2003). A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, **19**(2), 185. doi:10.1093/bioinformatics/19.2.185. URL `http://bioinformatics.oxfordjournals.org/cgi/content/abstract/19/2/185`.

Bourgon, R., Gentleman, R., and Huber, W. (2010a). Independent filtering increases detection power for high-throughput experiments. *Proc Natl Acad Sci USA*, **107**(21), 9546–51. doi:10.1073/pnas.0914005107.

Bourgon, R., Gentleman, R., and Huber, W. (2010b). Reply to talloen et al.: Independent filtering is a generic approach that needs domain specific adaptation — pnas. *Proceedings of the National Academy of Sciences of the United States of America*. URL `http://www.pnas.org/content/early/2010/11/05/1011698107.full.pdf+html?etoc`.

Bullard, J. H., *et al.* (2010). Evaluation of statistical methods for normalization and differential expression in mrna-seq experiments. *BMC Bioinformatics*, **11**, 94. doi:10.1186/1471-2105-11-94. URL `http://www.biomedcentral.com/1471-2105/11/94`.

Cancer Genome Atlas Research Network (2011). Integrated genomic analyses of ovarian carcinoma. *Nature*, **474**(7353), 609–615.

Cox, D. R. (1961). Tests of separate families of hypotheses. URL `http://projecteuclid.org/euclid.bsmsp/1200512162`.

Denoeud, F., *et al.* (2008). Annotating genomes with massive-scale RNA sequencing. *Genome Biol*, **9**(12), R175. doi:10.1186/gb-2008-9-12-r175.

Di, Y., *et al.* (2011). The NBP Negative Binomial Model for Assessing Differential Gene Expression from RNA-Seq. *Statistical Applications in Genetics and Molecular Biology*, **10**(1), 1–28. ISSN 1544-6115. doi:10.2202/1544-6115.1637. URL `http://dx.doi.org/10.2202/1544-6115.1637`.

Dillies, M.-A., *et al.* (2013). A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. *Briefings in Bioinformatics*, **14**(6), 671–683. ISSN 1477-4054. doi:10.1093/bib/bbs046. URL `http://dx.doi.org/10.1093/bib/bbs046`.

Efron, B. (1986). Double Exponential Families and Their Use in Generalized Linear Regression. *Journal of the American Statistical Association*, **81**(395), 709–721.

Flicek, P., *et al.* (2013). Ensembl 2013. *Nucleic Acids Research*, **41**(D1), D48–D55. ISSN 1362-4962. doi:10.1093/nar/gks1236. URL `http://dx.doi.org/10.1093/nar/gks1236`.

Guttman, M., *et al.* (2010). Ab initio reconstruction of cell type-specific transcriptomes in mouse reveals the conserved multi-exonic structure of lincrnas. *Nature Biotechnology*, **28**(5), 503–10. doi:10.1038/nbt.1633.

Hansen, K. D., Irizarry, R. A., and Wu, Z. (2012). Removing technical variability in RNA-seq data using conditional quantile normalization. *Biostatistics (Oxford, England)*, **13**(2), 204–216. ISSN 1468-4357. doi:10.1093/biostatistics/kxr054. URL `http://dx.doi.org/10.1093/biostatistics/kxr054`.

Hardcastle, T. and Kelly, K. (2010). baySeq: Empirical Bayesian methods for identifying differential expression in sequence count data. *BMC Bioinformatics*, **11**(1), 422+. ISSN 1471-2105. doi:10.1186/1471-2105-11-422. URL `http://dx.doi.org/10.1186/1471-2105-11-422`.

Jiang, H. and Wong, W. H. (2009). Statistical inferences for isoform expression in RNA-seq. *Bioinformatics*, **25**(8), 1026–32. doi:10.1093/bioinformatics/btp113.

Jorgensen, B. (1987). Exponential dispersion models. *Journal of the Royal Statistical Society. Series B (Methodological)*, **49**(2), pp. 127–162. ISSN 00359246. URL `http://www.jstor.org/stable/2345415`.

Law, C., *et al.* (2014). voom: precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biology*, **15**(2), R29+. ISSN 1465-6906. doi:10.1186/gb-2014-15-2-r29. URL `http://dx.doi.org/10.1186/gb-2014-15-2-r29`.

Leng, N., *et al.* (2013). EBSeq: an empirical Bayes hierarchical model for inference in RNA-seq experiments. *Bioinformatics*, **29**(8), 1035–1043. ISSN 1460-2059. doi:10.1093/bioinformatics/btt087. URL `http://dx.doi.org/10.1093/bioinformatics/btt087`.

Li, J. and Tibshirani, R. (2013). Finding consistent patterns: A nonparametric approach for identifying differential expression in RNA-Seq data. *Statistical Methods in Medical Research*, **22**(5), 519–536. ISSN 1477-0334. doi:10.1177/0962280211428386. URL `http://dx.doi.org/10.1177/0962280211428386`.

Li, J., *et al.* (2012). Normalization, testing, and false discovery rate estimation for RNA-sequencing data. *Biostatistics*, **13**(3), 523–538. ISSN 1468-4357. doi:10.1093/biostatistics/kxr031. URL `http://dx.doi.org/10.1093/biostatistics/kxr031`.

Lönnstedt, I. and Speed, T. (2002). Replicated microarray data. *Statistica sinica*, **12**(1), 31–46. URL `http://www.ams.org/mathscinet/search/publications.html?pg1=MR&s1=MR1894187`.

Love, M. I., Huber, W., and Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-Seq data with DESeq2. *bioRxiv*. doi:10.1101/002832. URL `http://dx.doi.org/10.1101/002832`.

Marioni, J. C., *et al.* (2008). Rna-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Research*, **18**(9), 1509–17. doi:10.1101/gr.079558.108. URL `http://genome.cshlp.org/content/18/9/1509.long`.

McCarthy, D. J., Chen, Y., and Smyth, G. K. (2012). Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic acids research*, **40**(10), 4288–4297. ISSN 1362-4962. doi:10.1093/nar/gks042. URL `http://dx.doi.org/10.1093/nar/gks042`.

Mortazavi, A., *et al.* (2008). Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature Methods*, **5**(7), 621. doi:doi:10.1038/nmeth.1226. URL `http://www.nature.com/nmeth/journal/v5/n7/abs/nmeth.1226.html`.

Oshlack, A. and Wakefield, M. J. (2009). Transcript length bias in RNA-seq data confounds systems biology. *Biol Direct*, **4**, 14. doi:10.1186/1745-6150-4-14.

Pan, Q., *et al.* (2008). Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing. *Nat Genet*, **40**(12), 1413–5. doi:10.1038/ng.259.

Pawitan, Y. (2001). *In All Likelihood: Statistical Modelling and Inference Using Likelihood*. Oxford Univ Press.

R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL `http://www.R-project.org/`.

Raiffa, H. and Schlaifer, R. (1961). *Applied statistical decision theory*. Studies in managerial economics. Division of Research, Graduate School of Business Adminitration, Harvard University, Boston. URL `http://opac.inria.fr/record=b1082847`.

Richard, H., *et al.* (2010). Prediction of alternative isoforms from exon expression levels in RNA-seq experiments. *Nucleic Acids Research*, **38**(10), e112. doi:10.1093/nar/gkq041. URL `http://nar.oxfordjournals.org/cgi/content/full/38/10/e112`.

Risso D, S. G., Schwartz K and S, D. (2011). Gc-content normalization for rna-seq data. *BMC Bioinformatics*, **12(1)**, 480.

Robinson, M. and Oshlack, A. (2010). A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biology*, **11**(3), R25–R25. doi:http://genomebiology.com/2010/11/3/R25. URL `http://genomebiology.com/2010/11/3/R25/abstract`.

Robinson, M. and Smyth, G. (2007). Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, **23**(21), 2881. doi:10.1093/bioinformatics/btm453. URL `http://bioinformatics.oxfordjournals.org/cgi/content/full/23/21/2881`.

Robinson, M. D., Mccarthy, D. J., and Smyth, G. K. (2010). edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics (Oxford, England)*, **26**(1), 139–140.

Salzman, J., Jiang, H., and Wong, W. H. (2010). Statistical modeling of RNA-SEQ data. Technical Report BIO-252, Division of Biostatistics, Stanford University, Palo Alto. URL `http://statistics.stanford.edu/~ckirby/techreports/BIO/BIO%20252.pdf`.

Shen, S., *et al.* (2012). MATS: a Bayesian framework for flexible detection of differential alternative splicing from RNA-Seq data. *Nucleic Acids Research*, **40**(8), e61. ISSN 1362-4962. doi:10.1093/nar/gkr1291. URL `http://dx.doi.org/10.1093/nar/gkr1291`.

Smyth, G. (2004). Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Statistical applications in genetics and molecular biology*, **3**(1), 1–25.

Smyth, G. K. (2005). Limma: linear models for microarray data. In R. Gentleman, V. Carey, S. Dudoit, and W. H. R. Irizarry, editors, *Bioinformatics and Computational Biology Solutions using R and Bioconductor*, pages 397–420. Springer, New York.

Soneson, C. and Delorenzi, M. (2013). A comparison of methods for differential expression analysis of RNA-seq data.

Tarazona, S., *et al.* (2011). Differential expression in RNA-seq: A matter of depth. *Genome Research*, **21**(12), 2213–2223. ISSN 1549-5469. doi:10.1101/gr.124321.111. URL `http://dx.doi.org/10.1101/gr.124321.111`.

Trapnell, C., Pachter, L., and Salzberg, S. (2009). Tophat: discovering splice junctions with RNA-seq. *Bioinformatics*, **25**(9), 1105. doi:10.1093/bioinformatics/btp120. URL `http://bioinformatics.oxfordjournals.org/cgi/content/full/25/9/1105`.

Trapnell, C., *et al.* (2010). Transcript assembly and quantification by RNA-seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature Biotechnology*, **28**(5), 511. doi:doi:10.1038/nbt.1621. URL `http://www.nature.com/nbt/journal/v28/n5/abs/nbt.1621.html`.

Van De Wiel, M. A., *et al.* (2013). Bayesian analysis of RNA sequencing data by estimating multiple shrinkage priors. *Biostatistics*, **14**(1), 113–128. ISSN 1468-4357. doi:10.1093/biostatistics/kxs031. URL `http://dx.doi.org/10.1093/biostatistics/kxs031`.

Vuong, Q. H. (1989). Likelihood Ratio Tests for Model Selection and Non-Nested Hypotheses. *Econometrica*, **57**(2), 307–333. ISSN 00129682. doi:10.2307/1912557. URL `http://dx.doi.org/10.2307/1912557`.

Wang, X. (2006). Approximating Bayesian inference by weighted likelihood. *Canadian Journal of Statistics*, **34**(2), 279–298.

Williams, D. A. (1982). Extra-binomial variation in logistic linear models. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, **31**(2), pp. 144–148. ISSN 00359254. URL `http://www.jstor.org/stable/2347977`.

Wu, H., Wang, C., and Wu, Z. (2013). A new shrinkage estimator for dispersion improves differential expression detection in RNA-seq data. *Biostatistics*, **14**(2), 232–243. ISSN 1468-4357. doi:10.1093/biostatistics/kxs033. URL `http://dx.doi.org/10.1093/biostatistics/kxs033`.

Wu, J., *et al.* (2011). Splicetrap: a method to quantify alternative splicing under single cellular conditions. *Bioinformatics*, **27**(21), 3010–3016. doi:10.1093/bioinformatics/btr508. URL `http://bioinformatics.oxfordjournals.org/content/27/21/3010.abstract`.

Xiong, H., *et al.* (2014). De-fpca: Testing gene differential expression and exon usage through functional principal component analysis. In *Statistical Analysis of Next Generation Sequencing Data*, pages 129–143. Springer International Publishing.

Yang, X., *et al.* (2012). Extra-binomial variation approach for analysis of pooled DNA sequencing data. *Bioinformatics*, **28**(22), 2898–2904. ISSN 1460-2059. doi:10.1093/bioinformatics/bts553. URL `http://dx.doi.org/10.1093/bioinformatics/bts553`.

Yu, D., Huber, W., and Vitek, O. (2013). Shrinkage estimation of dispersion in Negative Binomial models for RNA-seq experiments with small sample size. *Bioinformatics*, **29**(10), 1275–1282. ISSN 1367-4811. doi:10.1093/bioinformatics/btt143. URL `http://dx.doi.org/10.1093/bioinformatics/btt143`.

Zhou, Y. H., Xia, K., and Wright, F. A. (2011). A powerful and flexible approach to the analysis of RNA sequence count data. *Bioinformatics (Oxford, England)*, **27**(19), 2672–2678.