Senior Thesis in Mathematics

The Expectation Maximization Algorithm in RNA-Sequencing Read Alignment

Author:
Amy Watt

Advisor:
Dr. Jo Hardin

Submitted to Pomona College in Partial Fulfillment
of the Degree of Bachelor of Arts

May 16, 2020

Abstract

High throughput RNA Sequencing is a powerful tool for studying gene expression. Here, we elucidate the details of the EM algorithm as it is used to align RNA Sequencing reads to a reference with the overarching goal to estimate expression.

Contents

Chapter 1

Introduction

Measuring the expression levels of genes is a useful tool in biological research. One application is analysis of the heat shock response. Most organisms exhibit a heat shock response involving the synthesis of heat shock proteins under conditions of elevated temperatures. As global temperatures rise year after year, many organisms such as corals are dying because their environment is exceeding their maximum tolerable temperatures. Thus, it is important to study heat shock response and the ways that organisms respond to stressors. One approach is to analyze the expression of genes when organisms are under heat shock, and compare expression to control conditions. Identifying over- and under- expressed genes will provide a stronger understanding of the mechanisms behind the heat shock response. RNA-sequencing technology is a novel tool that can be employed to estimate the expression of genes and isoforms. The steps in RNA-sequencing are:

1. RNA is isolated from a sample
2. RNA is converted to cDNA fragments through reverse transcription and fragmentation
3. A high-throughput sequencer generates reads (sequence of A, C, T, G nucleotides) from the fragments
4. Reads are aligned (or mapped) to a reference or de novo transcriptome with an alignment tool
5. Counts of reads mapped to each transcript are used to estimate expression levels (Li, Ruotti, et al. 2010)

Prior to the development of RNA-sequencing, the main technology available to study RNA expression was microarrays. In microarrays, a plate is set up with a different probe in each well, where each probe is specific to an RNA sequence. RNA is extracted from a sample and converted to cDNA (as in RNA-sequencing) but with the inclusion of tracker nucleotides. The cDNA from the sample is applied to the microarray and RNA expression is quantified by the amount of probes that are attached to corresponding fragments from the cDNA that was applied (Pereira et al. 2015). RNA-sequencing has proven to be reproducible and more accurate, in addition to other advantages such as large dynamic range, low background noise, and the ability to discover new genes (Li, Ruotti, et al. 2010). The large amount of

data produced by RNA-sequencing, though, requires the development of new computational methods. One aspect of RNA-sequencing that requires computation is read mapping. When mapping reads to a transcriptome, there are often multireads defined as reads which map to more than one position in the transcriptome. There are several reasons why multireads exist. First, due to alternative splicing, there are several different possible forms one transcript can take on after introns (portions of RNA that do not code for proteins) are removed and exons (portions of RNA that do code for proteins) are spliced back together (Figure 1.1).

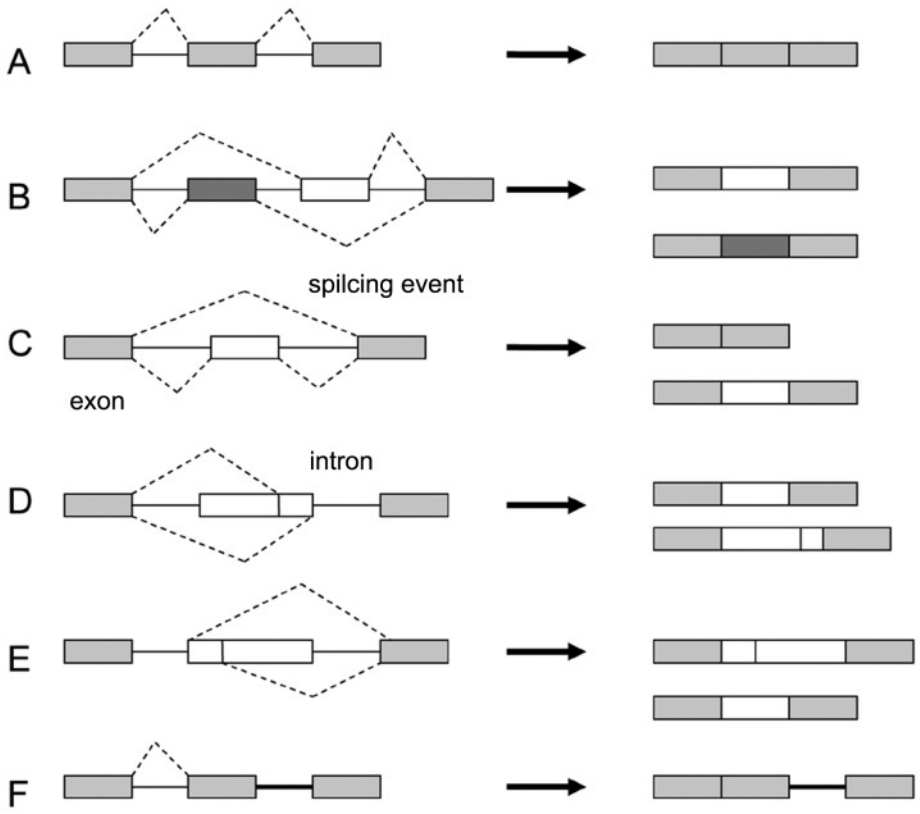

Figure 1.1: Types of Alternative Splicing (Wang et al. 2015)

Second, many organisms have paralogous genes, which are similar genes with the same origin but have evolved to serve different purposes. These genes have conserved region of high similarity. Lastly, low complexity sequences are repeats of short sequence motifs that are common in some genes (Li, Ruotti, et al. 2010). All three play a role in mapping uncertainty.

There have been several methods developed to process multireads. The first approach was to discard and ignore multireads. This introduces biases against genes with several isoforms, paralogous genes, and genes with low sequence complexity by underestimating the expression of these genes. Another approach allocated fractions of multireads to genes in the same proportions as the uniquely mapping genes, which improved upon the first model and gave estimates that more closely match results from cDNA microarrays. The most recent model uses statistics to estimate expression while incorporating multireads using the Expectation-Maximization (EM) algorithm with the goal to estimate the relative number of transcripts of each isoform present in the sample, given the reads and known isoforms (Li,

Ruotti, et al. 2010).

Chapter 2

The EM Algorithm

The Expectation Maximization algorithm is a widely used and general procedure that iteratively finds the values of an unknown parameter θ that maximizes the likelihood $P(r|\theta)$ of the observed data $\{r_1, r_2, ..., r_n\}$ using unobserved latent variables z. It involves two sequential steps after initializing the starting estimates of the parameter. The first step (expectation) finds the values of the missing information under the current estimate of the parameter and the data by using expected values. The second step (maximization) finds the maximum likelihood estimate values of the parameter given the missing information. It has been proven that each successive iteration of the algorithm never decreases the likelihood of the data under the estimated parameters, so that the algorithm will converge to a local optimum (Dempster, Laird, and Rubin 1977).

E Step: Compute values of missing information z given an estimate of θ at time t.

$$q^t(z) = P(z|r, \theta^{(t)})$$

M Step: Update the estimates of θ based on new z values that maximize the data likelihood.

$$Q(\theta|\theta^{(t)}) = \sum_z q^t(z) \log[P(r, \hat{z}|\theta)]$$

$$\hat{\theta}^{(t+1)} = argmax_\theta Q(\theta|\theta^{(t)})$$

2.1 Derivation of the EM algorithm

The log data likelihood is $\log(P(r|\theta))$. Using the law of total probability, z is introduced into the data likelihood. We will define q as a distribution over z such that $\sum_z q(z) = 1$ and $q(z) \geq 0$. This is introduced into the data likelihood by multiplying and dividing by the

same value (Andrew 2018).

$$\begin{aligned}
\log[P(r|\theta)] &= \log\left(\sum_z P(r,z|\theta)\right) \\
&= \log\left[\sum_z q(z)\cdot\frac{P(r,z|\theta)}{q(z)}\right] = \log\left[E_{qZ}\left(\frac{P(r,z|\theta)}{q(z)}\right)\right] \qquad (2.1)\\
&\geq \sum_z q(z)\log\left[\frac{P(r,z|\theta)}{q(z)}\right] = E_{qZ}\left[\log\left(\frac{P(r,z|\theta)}{q(z)}\right)\right] \qquad (2.2)
\end{aligned}$$

The penultimate step (from Equation 2.1 to 2.2) uses Jensen's inequality to get a lower bound on the log data likelihood, which is the quantity we want to maximize.

Definition 2.1 (Jenson's Inequality) *Given a convex function f and random variable X,*

$$E[f(X)] \geq f(EX).$$

If f is strictly convex, then

$$E[f(X)] = f(EX) \iff X \text{ is constant.}$$

Jensen's inequality also holds for concave function f with the inequality reversed:

$$f(EX) \geq E[f(X)]$$

(Andrew 2018).

A strictly concave function is a function f where $f''(x) \leq 0$. In the terms of the EM algorithm, we conclude that $\log\left[E_{qZ}\left(\frac{P(r,z|\theta)}{q(z)}\right)\right] \geq E_{qZ}\left[\log(\frac{P(r|\theta)}{q(z)})\right]$ from Equations 2.1 and 2.2. Here, $f(x) = \log(x)$, where $x = \frac{P(r,z|\theta)}{q(z)}$, is a strictly concave function because $f''(x) = -x^{-2} \leq 0$. Thus, $f(EX) \geq E[f(X)]$. Note that the inequality becomes an equality when $\frac{P(r,z|\theta)}{q(z)}$ is equal to a constant.

$$\begin{aligned}
c &= \frac{P(r,z|\theta)}{q(z)} \\
q(z) &= \frac{P(r,z|\theta)}{c} \\
&= \frac{P(r,z|\theta)}{\sum_z P(r,z|\theta)} \text{ because } \sum_z q(z) = 1 \\
&= \frac{P(r,z|\theta)}{P(r|\theta)} \\
&= P(z|r,\theta)
\end{aligned}$$

By setting $q(z) = P(z|r, \theta)$ (the posterior distribution of z over the data r and the current estimate of θ), we get an equality for the the log likelihood of the data, $\log(P(r|\theta))$. For the true value of θ (unknown), we can say:

$$\log(P(r|\theta)) = \sum_z q(z) \log\left(\frac{P(r, z|\theta)}{q(z)}\right) = \sum_z P(z|r, \theta) \log\left(\frac{P(r, z|\theta)}{P(z|r, \theta)}\right)$$

For the given estimate of θ at time t (θ^t), and the associated $q^t(z) = P(z|r, \theta^t)$ we can say:

$$\log(P(r|\theta^t)) = \sum_z q^t(z) \log\left(\frac{P(r, z|\theta^t)}{q^t(z)}\right) = \sum_z P(z|r, \theta^t) \log\left(\frac{P(r, z|\theta^t)}{P(z|r, \theta^t)}\right)$$

Here, we can get the parameter $\theta^{(t+1)}$ by maximizing the likelihood involving the previous estimate of the parameter. Because in this specific case, $q^t(z)$ is not defined as $P(z|r, \theta)$, but rather $P(z|r, \theta^t)$, the following is true due to Jenson's inequality (note there is no equality due to $q(z) \neq q^t(z)$):

$$\log(P(r|\theta)) \geq \sum_z q^t(z) \log\left(\frac{P(r, z|\theta^t)}{q^t(z^t)}\right) = \sum_z P(z|r, \theta^t) \log\left(\frac{P(r, z|\theta^t)}{P(z|r, \theta^t)}\right)$$

Further simplifying the log likelihood at a given estimate at time t gives us:

$$\begin{aligned} \log(P(r|\theta^t)) &= \sum_z q^t(z) \log\left(\frac{P(r, z|\theta^t)}{q^t(z)}\right) \\ &= \sum_z q^t(z) \log[P(r, z|\theta^t)] - \sum_z q^t(z) \log(q^t(z)) \end{aligned}$$

Note that the second term is independent of θ. Because we want to maximize the likelihood with respect to θ, we only need to maximize

$$Q(\theta|\theta^{(t)}) = E_{q^t_{(Z)}}[\log(P(r, z|\theta^t)] = \sum_z q^t(z) \log(P(r, z|\theta^t)$$

where $\theta^{(t)}$ is the current estimate of θ. We now iterate between the following two steps of the algorithm, where each iteration never decreases the likelihood of the data under the estimated parameters (Dempster, Laird, and Rubin 1977).

1. E step: $Q(\theta|\theta^{(t)})$ is the expected value of the log likelihood function of θ with respect to the current conditional distribution of z given an estimate of θ and data r.

$$q(z) = P(z|r, \theta^{(t)})$$

$$Q(\theta|\theta^{(t)}) = \sum_z q(z) \log(P(r, z|\theta)$$

$$Q(\theta|\theta^{(t)}) = \sum_z P(z|r, \theta^{(t)}) \log(P(r, z|\theta)$$

2. M-step: Update the estimates of θ based on new z values that maximize the data likelihood.

$$\hat{\theta}^{(t+1)} = argmax_{\theta} Q(\theta|\theta^{(t)})$$

2.2 Example of an Application of the EM Algorithm

Suppose there are two coins, coin 0 and coin 1, with different probabilities of landing on heads (θ). The parameters we want to estimate are θ_0 and θ_1. We run an experiment by selecting a coin at random, flipping it ten times, and recording the number of heads to get one observation. Repeat this process 5 times for 5 sets of total observations and record data in the vector r. We could estimate these values by setting θ to the number of heads over all flips for each coin, but we do not know which coin was flipped in each observation. The missing information is contained in the latent variable $z_n \in \{0, 1\}$, which represents the identity of the coin from observation r_n. The EM algorithm will be used to estimate θ using the missing information (Do and Batzoglou 2008).

First, we will get equations for the data likelihood. The observed data follow a binomial distribution.

$$P(r|\theta, z) = \binom{10}{r}(\theta_z)^r(1-\theta_z)^{10-r} \tag{2.3}$$

In this example, it is a priori equally likely to pick either coin, so

$$P(z|\theta) = \frac{1}{2} \tag{2.4}$$

By conditional probability and the law of total probability,

$$\begin{aligned} P(r, z|\theta) &= P(z|\theta)P(r|z, \theta) \\ &= \frac{1}{2}\binom{10}{r}(\theta_z)^r(1-\theta_z)^{10-r} \qquad (2.5) \\ P(r|\theta) &= \frac{1}{2}\binom{10}{r}(\theta_1)^r(1-\theta_1)^{10-r} + \frac{1}{2}\binom{10}{r}(\theta_0)^r(1-\theta_0)^{10-r} \qquad (2.6) \end{aligned}$$

We want to maximize the probability of the observed data, given that $q^t(z) = P(z|r, \theta^t)$, so the inequality from Jenson's Inequality holds as an equality:

$$
\begin{aligned}
P(r|\theta^t) &= \prod_n P(r_n|\theta^t) \\
P(r,z|\theta^t) &= \prod_n \sum_{z_n\in\{0,1\}} P(r_n,z_n|\theta^t) \\
&= \prod_n \sum_{z_n\in\{0,1\}} q^t(z_n)\cdot\frac{P(r_n,z_n|\theta^t)}{q^t(z_n)} \\
\log P(r,z|\theta^t) &= \log\left(\prod_n \sum_{z_n\in\{0,1\}} q^t(z_n)\cdot\frac{P(r_n,z_n|\theta^t)}{q^t(z_n)}\right) \\
&= \sum_n \log\left(\sum_{z_n\in\{0,1\}} q^t(z_n)\cdot\frac{P(r_n,z_n|\theta^t)}{q^t(z_n)}\right) \\
&= \sum_n \log\left(E_{q_Z}\frac{P(r_n,z_n|\theta^t)}{q^t(z_n)}\right) \\
&\geq \sum_n \left(E_{q_Z}\log(\frac{P(r_n,z_n|\theta^t)}{q^t(z_n)})\right) \text{ (by Jenson's)} \\
&= \sum_n \sum_{z_n\in\{0,1\}} q^t(z_n)\cdot\log\left(\frac{P(r_n,z_n|\theta^t)}{q^t(z_n)}\right) \\
&= \sum_n \sum_{z_n\in\{0,1\}} q^t(z_n)\cdot\log P(r_n,z_n|\theta^t) - \sum_n \sum_{z_n\in\{0,1\}} q^t(z_n)\cdot\log q^t(z_n)
\end{aligned}
$$

We want to maximize the data likelihood with respect to θ, so we can maximize $Q(\theta|\theta^{(t)})$ to get an estimate of $\theta^{(t+1)}$, and each iteration converges closer to θ

$$
\begin{aligned}
Q(\theta|\theta^{(t)}) &= \sum_n \sum_{z_n\in\{0,1\}} q^t(z_n)\cdot\log P(x_n,z_n|\theta^{(t)}) \\
Q(\theta|\theta^{(t)}) &= \sum_n \sum_{z_n\in\{0,1\}} P(z_n|x_n,\theta^{(t)})\cdot\log P(x_n,z_n|\theta^{(t)})
\end{aligned}
$$

Now, we summarize the use of the EM algorithm:

1. E step: find missing information z given the current estimates of θ. $\hat{z}_n^t = q^t(z)$ will be a value between 0 and 1 that represents the probability that the observation is from coin 1. Here, the expected value of z is the likelihood of the data with coin 1, over the likelihood of the data with coin 1 plus the likelihood of the data with coin 0.

$$
\begin{aligned}
\hat{z}_n^t &= P(z_n = 1 | r_n, \theta^{(t)}) \\
&= \frac{P(r_n | z_n = 1, \theta^{(t)}) \cdot P(z_n = 1 | \theta^{(t)})}{P(r_n | \theta^{(t)})} \\
&= \frac{\binom{10}{r_n} (\theta_1^{(t)})^{r_n} (1 - \theta_1^{(t)})^{10 - r_n} \cdot \frac{1}{2}}{\frac{1}{2}\binom{10}{r_n} (\theta_1^{(t)})^{r_n} (1 - \theta_1^{(t)})^{10 - r_n} + \frac{1}{2}\binom{10}{r_n} (\theta_0^{(t)})^{r_n} (1 - \theta_0^{(t)})^{10 - r_n}} \quad (Eqns\ 2.3, 2.4, 2.6) \\
&= \frac{(\theta_1)^{r_n} (1 - \theta_1^{(t)})^{10 - r_n}}{(\theta_1^{(t)})^{r_n} (1 - \theta_1^{(t)})^{10 - r_n} + (\theta_0^{(t)})^{r_n} (1 - \theta_0^{(t)})^{10 - r_n}}
\end{aligned}
$$

Note that the probability that the observation is from coin 0 is $1 - \hat{z}_n^t = P(z_n = 0 | r_n, \theta^{(t)})$.

2. M step: find the estimates of θ that maximize the data, r and the newly estimated $\hat{z}_n^t$

$$
\begin{aligned}
Q(\theta | \theta^{(t)}) &= \sum_n \sum_{z_n \in \{0,1\}} P(z_n | r_n, \theta^{(t)}) \cdot \log P(r_n, z_n | \theta^{(t)}) \\
&= \sum_n \sum_{z_n \in \{0,1\}} P(z_n | r_n, \theta^{(t)}) \cdot \log \frac{1}{2} \binom{10}{r_n} (\theta_{z_n})^{r_n} (1 - \theta_{z_n})^{10 - r_n} \quad (Eqn\ 2.5) \\
&= \sum_n P(z_n = 0 | r_n, \theta^{(t)}) \cdot [\log \frac{1}{2} \binom{10}{r_n} (\theta_0)^{r_n} (1 - \theta_0)^{10 - r_n}] \\
&\quad + \sum_n P(z_n = 1 | r_n, \theta^{(t)}) \cdot [\log \frac{1}{2} \binom{10}{r_n} (\theta_1)^{r_n} (1 - \theta_1)^{10 - r_n}] \\
&= \sum_n \hat{z}_n^t \cdot [\log \frac{1}{2} \binom{10}{r_n} (\theta_0)^{r_n} (1 - \theta_0)^{10 - r_n}] \\
&\quad + \sum_n (1 - \hat{z}_n^t) \cdot [\log \frac{1}{2} \binom{10}{r_n} (\theta_1)^{r_n} (1 - \theta_1)^{10 - r_n}]
\end{aligned}
$$

To find the MLE of θ_0, take the partial derivative with respect to θ_0 and set it equal to 0.

$$\begin{aligned}
\frac{\partial}{\partial\theta_0}Q(\theta|\theta^{(t)}) &= \frac{\partial}{\partial\theta_0}\left(\sum_n \hat{z}_n^t \cdot [\log\frac{1}{2}\binom{10}{r_n}(\theta_0)^{r_n}(1-\theta_0)^{10-r_n}]\right)\\
0 &= \sum_n \frac{\partial}{\partial\theta_0}\left(\hat{z}_n^t r_n \log(\theta_0) + \hat{z}_n^t(10-r_n)\log(1-\theta_0)\right)\\
0 &= \sum_n \left(\frac{\hat{z}_n^t r_n}{\theta_0} - \frac{\hat{z}_n^t(10-r_n)}{1-\theta_0}\right)\\
\sum_n \frac{\hat{z}_n^t r_n}{\theta_0} &= \sum_n \frac{\hat{z}_n^t(10-r_n)}{1-\theta_0}\\
(1-\theta_0)\sum_n \hat{z}_n^t r_n &= \theta_0 \sum_n \hat{z}_n^t(10-r_n)\\
\sum_n \hat{z}_n^t r_n &= \theta_0 \sum_n \hat{z}_n^t(10-r_n) + \theta_0 \sum_n \hat{z}_n^t r_n\\
\sum_n \hat{z}_n^t r_n &= \theta_0 \sum_n \hat{z}_n^t 10\\
\hat{\theta}_0^{(t+1)} &= \frac{\sum_n \hat{z}_n^t r_n}{\sum_n \hat{z}_n^t 10}
\end{aligned}$$

To find the MLE of θ_1, take the partial derivative with respect to θ_1 and set it equal to 0. By a similar process,

$$\begin{aligned}
\frac{\partial}{\partial\theta_1}Q(\theta|\theta^{(t)}) &= \frac{\partial}{\partial\theta_1}\left(\sum_n (1-\hat{z}_n^t) \cdot [\log\frac{1}{2}\binom{10}{r_n}(\theta_1)^{r_n}(1-\theta_1)^{10-r_n}]\right)\\
0 &= \sum_n \frac{\partial}{\partial\theta_1}\left((1-\hat{z}_n^t) r_n \log(\theta_1) + (1-\hat{z}_n^t)(10-r_n)\log(1-\theta_1)\right)\\
0 &= \sum_n \left(\frac{(1-\hat{z}_n^t) r_n}{\theta_1} - \frac{(1-\hat{z}_n^t)(10-r_n)}{1-\theta_1}\right)\\
\sum_n \frac{(1-\hat{z}_n^t) r_n}{\theta_1} &= \sum_n \frac{(1-\hat{z}_n^t)(10-r_n)}{1-\theta_1}\\
(1-\theta_1)\sum_n (1-\hat{z}_n^t) r_n &= \theta_1 \sum_n (1-\hat{z}_n^t)(10-r_n)\\
\sum_n (1-\hat{z}_n^t) r_n &= \theta_1 \sum_n (1-\hat{z}_n^t)(10)\\
\hat{\theta}_1^{(t+1)} &= \frac{\sum_n (1-\hat{z}_n) r_n}{\sum_n 10(1-\hat{z}_n)}
\end{aligned}$$

Based on the z values, these estimators of θ are calculated by allocating portions of

each observation to coin 0 and coin 1. If z_n is large, a larger portion of r_n is used in the calculation of the estimator of coin 1's bias.

We will now iterate through the initial iteration of the EM algorithm, with starting values of $\hat{\theta}_0^0 = 0.5$ and $\hat{\theta}_1^0 = 0.6$ (arbitrarily chosen to be different). Suppose we have 5 data points and $r = \{5, 9, 8, 4, 7\}$.

1. E step: $\hat{z}^t = \frac{(\theta_1)^{r_n}(1-\theta_1^{(t)})^{10-r_n}}{(\theta_1^{(t)})^{r_n}(1-\theta_1^{(t)})^{10-r_n}+(\theta_0^{(t)})^{r_n}(1-\theta_0^{(t)})^{10-r_n}}$
   $\hat{z}_1^1 = \frac{(0.6)^5(0.4)^5}{(0.6)^5(0.4)^5(0.5)^5(0.5)^5} = 0.449$
   $\hat{z}_2^1 = \frac{(0.6)^9(0.4)^1}{(0.6)^9(0.4)^1(0.5)^9(0.5)^1} = 0.805$
   $\hat{z}_3^1 = \frac{(0.6)^8(0.4)^2}{(0.6)^8(0.4)^2(0.5)^8(0.5)^2} = 0.733$
   $\hat{z}_4^1 = \frac{(0.6)^4(0.4)^6}{(0.6)^4(0.4)^6(0.5)^4(0.5)^6} = 0.352$
   $\hat{z}_5^1 = \frac{(0.6)^4(0.4)^6}{(0.6)^4(0.4)^6(0.5)^4(0.5)^6} = 0.647$

2. M step: $\hat{\theta}_0^{(t+1)} = \frac{\sum_n \hat{z}_n^t r_n}{\sum_n \hat{z}_n^t 10}$
   $\hat{\theta}_1^1 = \frac{.449 \cdot 5 + .806 \cdot 9 + .733 \cdot 8 + .352 \cdot 4 + .647 \cdot 7}{10(.449 + .805 + .733 + .352 + .647} = 0.713$
   $\hat{\theta}_0^1 = \frac{.551 \cdot 5 + .195 \cdot 9 + .267 \cdot 8 + .648 \cdot 4 + .353 \cdot 7}{10(.551 + .195 + .267 + .648 + .353} = 0.581$

Chapter 3

RNA Sequencing and Read Mapping

3.1 Notation and Variables

Variable	Definition
L	read length
N	total number of reads (library size)
M	number of known isoforms
l_i	length of isoform $i \in \{1, ..., M\}$
R_n	sequence of read $n \in \{1, ..., N\}$
$S_n \in [1, l_i]$	start position of read n
$O_n \in \{1, 0\}$	orientation of read n
$G_n \in [1, M]$	isoform mapped to read n
τ_i	fraction of all transcripts from isoform $i = \frac{\nu_i}{l_i}(\sum_{j=1}^{l_i} \frac{\nu_j}{l_j})$
ν_i	fraction of all nucleotides from isoform $i = \frac{\tau_i l_i}{\sum_{j=1}^{l_i} \tau_j l_j}$
θ_i	prior probability that a read is from isoform $i = [P(G_n = i)]$

3.2 Motivation Behind EM Algorithm in RNA-Sequencing

RNA-sequencing data is able to provide the relative abundances of transcripts within one sample. Many algorithms have been developed to achieve the same outcome. TIGAR2, for example utilizes Baysian inference, while others (RSEM, Cufflinks, eXpress) utilize variations of the EM algorithm (Zhang et al. 2017). We will focus on RSEM and delineate its application of the EM algorithm to RNA-Sequencing, as originally reported by the developers Li and Dewey (Li, Ruotti, et al. 2010; Li and Dewey 2011).

There are two ways to measure relative expression for isoform i: τ_i (fraction of all transcripts in a biological sample from isoform i) or ν_i (fraction of all nucleotides in a biological sample from isoform i). We want to use RNA-sequencing data to estimate τ and ν, and we assume the number of reads from any isoform is a function of its length. We introduce another measure of expression, θ_i (the probability that a read is from isoform i). Because longer isoforms are assumed to produce more reads, there is a proportional relationship be-

tween τ_i and ν_i. There is also a relationship between ν_i and θ_i where ν_i is equal to the probability that isoform i produces a read scaled to consider only the reads that do align (θ_0 is defined as the proportion of reads that do not map to any isoform):

$$\nu_i = \frac{\theta_i}{1 - \theta_0}$$

$$\nu_i = \frac{\tau_i \cdot l_i}{\sum_{k=1}^{M} \tau_k \cdot l_k}$$

$$\tau_i = \frac{\frac{\nu_i}{l_i}}{\sum_{k=1}^{M} \frac{\nu_k}{l_k}}$$

We need to create a data likelihood for the RNA-Sequencing process to estimate these measures of expression, motivating the construction of a mathematical model.

3.3 Building a Model of the Sequencing Process

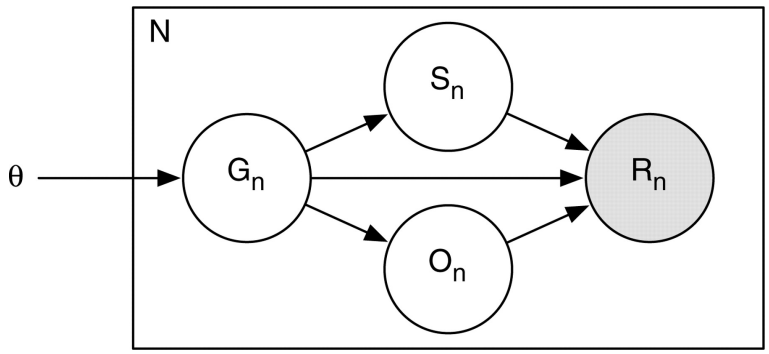

Figure 3.1: Bayesian network (graphical model) of RNA-sequencing process built to generate N reads of length L. The parameter is θ. The latent random variables are G, S, O. The data are R, and will be referred to as r because they are observed. (Li, Ruotti, et al. 2010)

.

The graphical model (Bayesian network) of the RNA sequencing process is shown in Figure 3.1. This model generates N reads of length L with parameter θ (the probability that an isoform produces a read) which corresponds to expression levels (Li, Ruotti, et al. 2010). This graph displays the relationship between variables, where each node in the graph represents a random variable. Within Bayesian networks, certain independence assumptions hold that determine what information is required to specify the probability distribution among the variables. By formalizing independence assumptions, we can simplify the joint data likelihood, which requires a prior on the roots (θ) and conditionals on the non-roots (G, S, O, R). This requires more information regarding theory on networks and graphs.

In defining any network, there are three types of connections from a to c through a node b. If we have a path from a to c through b (the path does not have to be directed), then b can be referred to as linear, converging, or diverging with respect to its location in the path

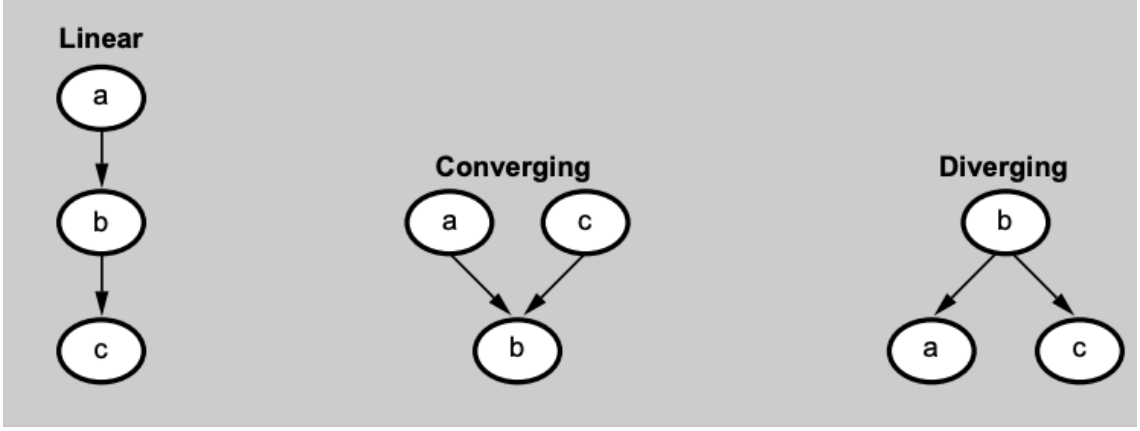

Figure 3.2: Connection types (Charniak 1991)

from a to c (Figure 3.2) (Charniak 1991).

Definition 3.1 (d-connecting) *A path from node a to node c is d-connecting with respect to evidence $E = \{e_1, ..., e_m\}$ if every interior node b in the path has one of the following properties:*

1. *b is linear or diverging and not in E*
2. *b is converging and either b or one of its descendants (node that is a child of n or is recursively a descendent of a child of b) is in E*

If there is no d-connecting path, then a and c are d-separated (Charniak 1991).

Theorem 3.2 (Probabilistic Implications of d-Separation) *If Variable A is d-separated from Variable B given some evidence E in a directed acyclic graph, then A is independent of B given E. (Pearl 2009).*

We will use Definition 3.1 and Theorem 3.2 to determine which variables in our Bayesian network are independent in the complete data likelihood.

In order to compute the complete data likelihood based on the model, we need to use the definition of conditional probability to derive a simplification of a probability of the form $P(x_1, x_2, x_3, x_4, x_5)$.

Definition 3.3 (Conditional Probability) *$P(B|A)$ is the probability of B occurring, conditional on the fact that A has occurred.*

$$\frac{P(x_1, x_2)}{P(x_2)} = P(x_1|x_2)$$

$$P(x_1, x_2) = P(x_1|x_2)P(x_2)$$

Using the definition of conditional probability, we can expand to the joint distribution on all variables of interest:

$$\begin{aligned} P(r, S, O, G, \theta) &= P(r|S, O, G, \theta)P(S, O, G, \theta) \\ &= P(r|S, O, G, \theta)P(S|O, G, \theta)P(O, G, \theta) \\ &= P(r|S, O, G, \theta)P(S|O, G, \theta)P(O|G, \theta)P(r, \theta) \\ &= P(r|S, O, G, \theta)P(S|O, G, \theta)P(O|G, \theta)P(G|\theta)P(\theta) \end{aligned} \tag{3.1}$$

Based on the model (Figure 3.1), the complete data likelihood for the RNA Sequencing data is:

$$\begin{aligned} P(G, S, O, r|\theta) &= \frac{P(G, S, O, r, \theta)}{P(\theta)} \\ &= \frac{P(r, S, O, G, \theta)}{P(\theta)} \\ &= \frac{P(r|S, O, G, \theta) \cdot P(S|O, G, \theta) \cdot P(O|G, \theta) \cdot P(G|\theta) \cdot P(\theta)}{P(\theta)} \ (Eqn\ 3.1) \\ &= P(r|S, O, G, \theta) \cdot P(S|O, G, \theta) \cdot P(O|G, \theta) \cdot P(G|\theta) \\ &= \Pi_{n=1}^{N} P(r_n|S_n, O_n, G_n, \theta_n) \cdot P(S_n|O_n, G_n, \theta_n) \cdot P(O_n|G_n, \theta_n) \cdot P(G_n|\theta_n) \end{aligned} \tag{3.2}$$

We will make several simplifications to the likelihood in Equation 3.2 by applying our knowledge of Bayesian networks and independence due to d-separation to our RNA-sequencing model (Figure 3.1) as it is a Bayesian network:

1. r is independent of θ given S, O, G
   G is a linear interior node in the paths $(\theta \to G \to R)$, $(\theta \to G \to S \to r)$ (Figure 3.3), and $(\theta \to G \to O \to r)$ and is part of $E = \{S, O, G\}$. There is no d-connecting path between r and θ with respect to evidence E.

2. S is independent of O given G
   G is a diverging node in the path $(O \to G \to S)$ (Figure 3.4) and is part of $E = \{G\}$ so this is not a d-connecting path between G and O. r is a converging node on the path $(S \to r \to O)$ and is not in E so this is not a d-connecting path between G and O. Neither of the two paths between G and O are d-connecting.

3. S is independent of θ given G
   Any path from S to θ includes G (linear) which is part of $E = \{G\}$. There is no d-connecting path between S and θ.

4. O is independent of θ given G
   Any path from O to θ includes G (linear) which is part of $E = \{G\}$. There is no d-connecting path between O and θ.

The data likelihood can be simplified to:

$$P(G, S, O, r|\theta) = \Pi_{n=1}^{N} P(r_n|S_n, O_n, G_n) \cdot P(S_n|G_n) \cdot P(O_n|G_n) \cdot P(G_n|\theta_n) \tag{3.3}$$

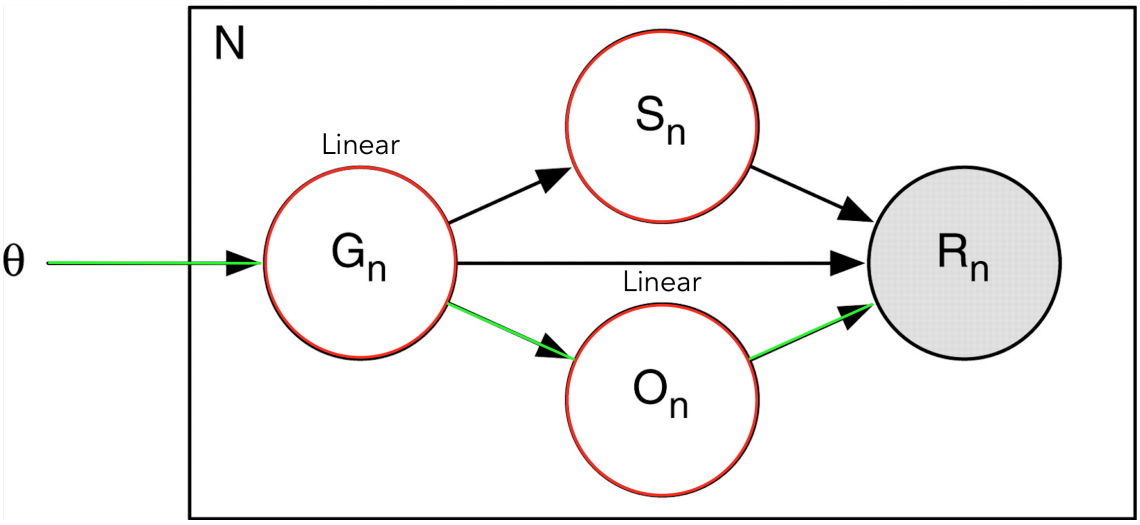

Figure 3.3: Path $\theta \to G \to S \to r$ (green) is d-separated with respect to evidence $E = \{S, O, G\}$ (red) because interior nodes G, O are both linear and in E.

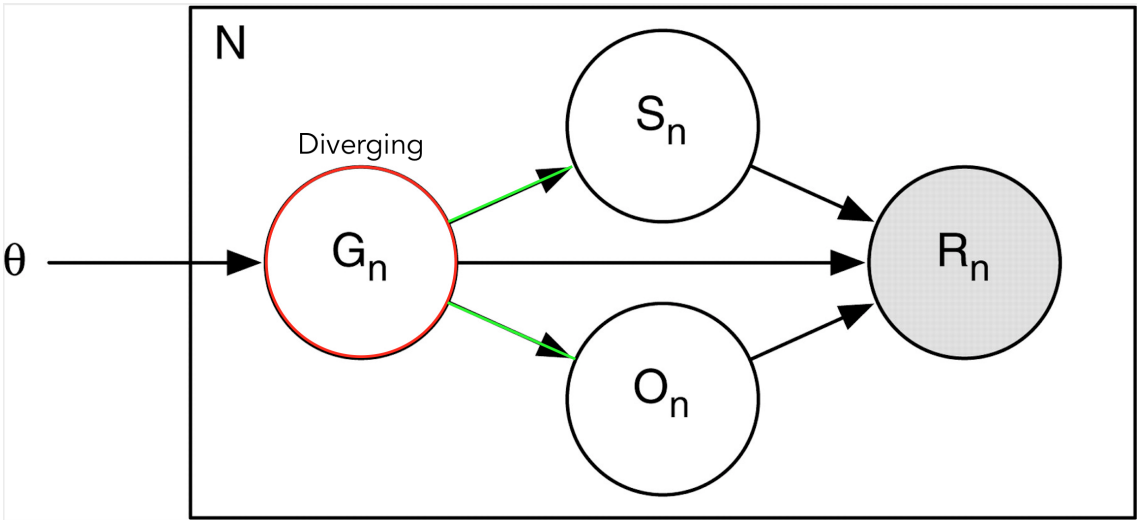

Figure 3.4: Path $O \to G \to S$ (green) is d-separated with respect to evidence $E = \{G\}$ (red) because interior node G is diverging and in E.

3.4 Conditional Probabilities from the Model

We will assume we are given all M isoforms, and will assume a uniform distribution read start position along the length of the isoform. To generate reads following the model:

1. Draw $G_n : [0, M]$, where G_0 represents the noise isoform (reads that do not map to any of the known isoforms).
   (a) $P(G_n = i|\theta) = \theta_i$
2. Given that the n^{th} read comes from the i^{th} isoform, choose where on an isoform a read starts: $S_n : [1, l_i]$ (with poly-A tailing) or $S_n : [1, l_i - L + 1]$ (without poly-A tailing).
   (a) $P(S_n = j|G_n = i) = \frac{1}{l_i}$ (with poly-A tailing)
   (b) $P(S_n = j|G_n = i) = \frac{1}{l_i - L + 1}$ (without poly-A tailing)
3. Choose the orientation of the read: $O_n : \{0, 1\}$ where 0 represent a read in the same orientation as the parent isoform, and 1 represents a read in the same direction as the parent reverse complement.
   (a) $P(O_n = 0|G_n \neq 0) = 0.5$
   (b) $P(O_n = 1|G_n \neq 0) = 0.5$
4. R_n is generated by sequencing G_n from position S_n in the orientation of O_n.

In order to determine the conditional probability of R_n, we need to define a new indicator variable that summarizes the hidden random variables.

$$z_{nijk} = 1 \text{ if } (G_n, S_n, O_n) = (i, j, k)$$

Then, the conditional probability of R_n is

$$P(R_n = \rho|z_{nijk} = 1) = \begin{cases} \Pi_{t=1}^{L} w_t(\rho_t, \gamma^i_{j+t-1}) & k = 0 \\ \Pi_{t=1}^{L} w_t(\rho_t, \gamma^{-i}_{j+t-1}) & k = 1 \end{cases} \tag{3.4}$$

where ρ_t is the t^{th} base of a particular sequence ρ of length L, γ^i is the sequence of isoform i, γ^{-i} is the sequence of the reverse complement of isoform i, and $w_t(a, b)$ is a matrix whose values are the probability of seeing an a at position t given that b is present in the reference at position t. Thus, w_t models a read position and base-dependent substitution process. Due to the nature of RNA-sequencing technology, the further a base is from the start of a read, the more likely there will be a sequencing error. Thus, we expect $w_L(a, a)$ to be smaller than $w_1(a, a)$.

Using the conditional probabilities in Equation 3.3, we get (with poly-A tailing):

$$P(G, S, O, r|\theta) = \Pi_{n=1}^{N} P(r_n|z_{nijk} = 1) \cdot \frac{\theta_i}{2 \cdot l_i} \tag{3.5}$$

The mathematical model of RNA sequencing (Figure 3.1 and Equation 3.5) do pose some potential inaccuracies. One, it assumes the fragment size is constant and is equal to read length. Two, the RNA-sequencing model starts with an isoform to generate a fragment to generate read, while in reality RNA-sequencing starts with a read to generate a fragment to generate an isoform. Despite this, Equation 3.5 will be used to help determine the values of θ (Li, Ruotti, et al. 2010; Cheplyaka 2017).

3.5 Setting a Prior Distribution on θ

θ represents the probability that any single read is from a given isoform. If θ has a uniform distribution, there is equal probability that a read comes from any isoform. This assumes that short isoforms produce more reads because for two isoforms with equal expression, we expect fewer reads to be produced by a shorter isoform. τ is the proportion of transcripts from a given isoform. If τ has a uniform distribution, there are equal proportions of transcripts coming from all isoforms, meaning that there is equal expression of all isoforms (Cheplyaka 2017). The relationship between θ and τ is as follows (assuming that all reads align so $\theta_0 = 0$):

$$\nu_i = \frac{\tau_i \cdot l_i}{\sum_{k=1}^{M} \tau_k \cdot l_k}$$

$$\nu_i = \frac{\theta_i}{1 - \theta_0} = \theta_i$$

$$\tau_i = \frac{\frac{\nu_i}{l_i}}{\sum_{i=1}^{M} \frac{\nu_k}{l_k}} = \frac{\frac{\theta_i}{l_i}}{\sum_{k=1}^{M} \frac{\theta_k}{l_k}}$$

When formulating a prior probability density of θ, we assume that there is no information on isoform expression levels. So, we formulate our prior based on the assumption that all isoforms are expressed equally, namely that τ is uniform under the constraint that $\sum_{i=1}^{M} \tau_i = 1$ (i.e. τ is uniformly distributed on the unit M-simplex).

Definition 3.4 (Unit K-simplex) *A set of points $x \in \mathbb{R}^{K+1}$ such that for $0 \leq k \leq K$, $x_k \geq 0$ and $\sum_{k=0}^{K} x_k = 1$ (Malygin and Postnikov 2011). The unit K-simplex is k dimensional in $\mathbb{R}^{K+1}$.*

The unit 1-simplex is a 1 dimensional line segment in $\mathbb{R}^2$ with extreme points $(0, 1)$ and $(1, 0)$. Some other points in this unit 1-simplex are $(0.3, 0.7)$ since $x_1 + x_2 = 0.3 + 0.7 = 1$, and $(0.94, 0.06)$ since $x_1 + x_2 = 0.94 + 0.06 = 1$. The unit 2-simplex (in $\mathbb{R}^3$) is a triangle with vertices $(1, 0, 0)$, $(0, 1, 0)$, $(0, 0, 1)$. To define a pdf f on $\tau_1, ..., \tau_M$, we will use the fact that the volume of a unit simplex in $\mathbb{R}^M$ is $\frac{1}{M!}$, and that the distribution is uniform.

$$
\begin{aligned}
f(\tau_1, ..., \tau_M) &= c \\
\int f(\tau_1, ..., \tau_M) d\tau_1 ... d\tau_M &= 1 \\
\int c d\tau_1 ... d\tau_M &= 1 \\
c\frac{1}{M!} &= 1 \\
c &= M!
\end{aligned}
$$

Thus, the pdf $f(\tau_1, ..., \tau_M) = M!$.

Theorem 3.5 (Multivariate Transformation) *τ has a continuous distribution and a pdf. Define $\theta_1, ..., \theta_M$ as:*

$$\theta_i = r_i(\tau_1, ..., \tau_M) = \frac{\tau_i \cdot l_i}{\sum_{k=1}^{M} \tau_k \cdot l_k}$$

where r_i is a one-to-one differentiable function from the unit simplex where τ is uniform, into a different space. s_i is the inverse

$$\tau_i = s_i(\theta_1, ..., \theta_M) = \frac{\frac{\theta_i}{l_i}}{\sum_{k=1}^{M} \frac{\theta_k}{l_k}}$$

Then, the joint pdf of θ is:

$$f_\theta(\theta_1, ..., \theta_n) = f_s(s_1, ..., s_n)|J|$$

where $|J| = \frac{\partial s}{\partial \theta}$ is the Jacobian determinant.

The pdf $f_\theta(\theta_1, ..., \theta_M)$ is obtained by starting with the pdf $f(\tau_1, ..., \tau_M)$ with τ_i expressed as $s_i(\theta_1, ..., \theta_M)$ (we know this to be $f(\tau_1, ..., \tau_M) = M!$) and multiplying by the determinant. Thus,

$$f_\theta(\theta_1, ..., \theta_M) = M!|J|$$

The intuition behind this transformation relies on the relation between $\tau = s(\theta)$. As measures of expression, the change in one must be the same as the change in the other. We can equate changes in $dF(\tau)$ and $dF(\theta)$ as:

$$
\begin{aligned}
dF(\theta) &= dF(\tau) \\
f(\theta)d\theta &= f(\tau)d\tau \\
f(\theta) &= f(\tau)\frac{d\tau}{d\theta}
\end{aligned}
$$

where $\frac{d\tau}{d\theta}$ is the Jacobian determinant (DeGroot and M.J. 2011).

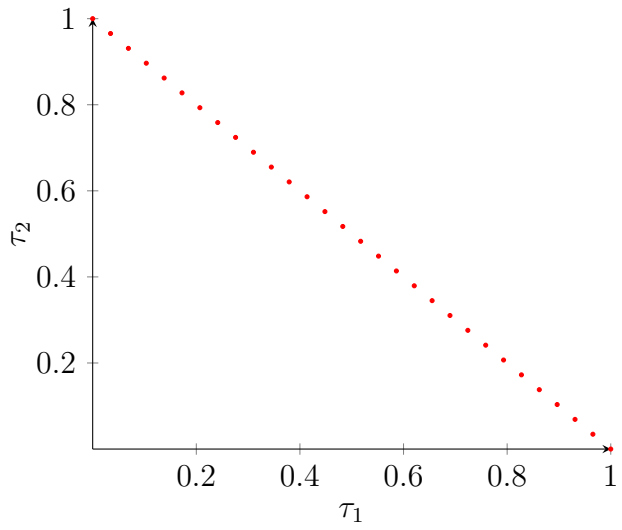

Figure 3.5: Uniform distribution on the unit 1-simplex

Starting with a simple example, suppose we only have two isoforms. θ_1 represents the probability a read is from isoform 1, and θ_2 represents the probability a read is from isoform 2. τ_1 represents the proportion of isoform 1 in the sample, and θ_2 represents the proportion of isoform 2 in the sample. Below is a visual of the possible values of (τ_1, τ_2) (the 1-unit simplex in $\mathbb{R}^2$).

The joint probability density function $f(\tau_1, \tau_2) = 2! = 2$. We want to do a change of variable from (τ_1, τ_2) to (θ_1, θ_2).

$$\tau_1 = s_1(\theta_1, ..., \theta_M) = \frac{\frac{\theta_1}{l_1}}{\frac{\theta_1}{l_1} + \frac{\theta_2}{l_2}}$$

$$\tau_2 = s_2(\theta_1, ..., \theta_M) = \frac{\frac{\theta_2}{l_2}}{\frac{\theta_1}{l_1} + \frac{\theta_2}{l_2}}$$

$$\frac{d\tau_1}{d\theta_1} = \frac{\frac{1}{l_1} \cdot (\frac{\theta_1}{l_1} + \frac{\theta_2}{l_2} - \frac{\theta_1}{l_1})}{\frac{\theta_1}{l_1} + \frac{\theta_2}{l_2}}$$

$$\frac{d\tau_1}{d\theta_2} = \frac{\frac{1}{l_2} \cdot (\frac{\theta_1}{l_1} + \frac{\theta_2}{l_2} - \frac{\theta_1}{l_1})}{\frac{\theta_1}{l_1} + \frac{\theta_2}{l_2}}$$

$$\frac{d\tau_2}{d\theta_1} = \frac{\frac{1}{l_1} \cdot (\frac{\theta_1}{l_1} + \frac{\theta_2}{l_2} - \frac{\theta_2}{l_2})}{\frac{\theta_1}{l_1} + \frac{\theta_2}{l_2}}$$

$$\frac{d\tau_2}{d\theta_2} = \frac{\frac{1}{l_2} \cdot (\frac{\theta_1}{l_1} + \frac{\theta_2}{l_2} - \frac{\theta_2}{l_2})}{\frac{\theta_1}{l_1} + \frac{\theta_2}{l_2}}$$

$$\mathbf{J} = \begin{bmatrix} \frac{\partial \tau_1}{\partial \theta_1} & \frac{\partial \tau_1}{\partial \theta_2} \\ \frac{\partial \tau_2}{\partial \theta_1} & \frac{\partial \tau_2}{\partial \theta_2} \end{bmatrix}$$

Then, $P(\theta) = f(\theta_1, \theta_2) = f(\tau_1, \tau_2)|J|$, where $|J|$ is defined above. Generalizing (parameterizing on M-1 isoforms because we view $\theta_M = 1 - \sum_{k=0}^{M-1} \theta_k$),

$$P(\theta) = (M-1)!|J|$$

3.6 Application of the EM Algorithm to Read Mapping

To estimate the expression of genes and isoforms in a sample, we seek to estimate the values of θ, which correspond to expression levels as such:

$$\nu_i = \frac{\theta_i}{1 - \theta_0}$$

$$\tau_i = \frac{\frac{\nu_i}{l_i}}{\sum_{i=1}^{M} \frac{\nu_k}{l_k}}$$

We will use the EM algorithm to find values of θ that maximize the likelihood of the observed data. The complete log data likelihood is:

$$\begin{aligned}
\log[P(r|\theta)] &= \log[\prod_{n=1}^{N} P(r_n|\theta)] \\
&= \sum_n \log[P(r_n|\theta)] \\
&= \sum_n \log\left[\sum_i P(r_n, G_n = i|\theta)\right] \\
&= \sum_n \log\left[\sum_i P(r_n|G_n = i, \theta) \cdot P(G_n = i|\theta)\right] \\
&= \sum_n \log\left[\sum_i P(r_n|G_n = i, \theta) \cdot \theta_i\right] \qquad (3.6)
\end{aligned}$$

z_{nijk} was previously defined as an indicator variable for read n that equals 1 if $G_n = i, S_n = j, O_n = k$. To simplify the problem, we will assume a strand-specific protocol and a uniform read start position distribution. The new indicator is z_{nij} equals 1 if $G_n = i, S_n = j$. Thus,

$$\begin{aligned}
P(r_n|z_{nij} = 1) &= P(r_n|G_n = i, S_n = j) \\
P(r_n|G_n = i) &= \sum_j P(r_n|G_n = i, S_n = j) \cdot P(S_n = j) \text{ (by conditional probability)} \\
&= \frac{1}{l_i} \sum_j P(r_n|z_{nij} = 1) \text{ (since } P(S_n = j) = \frac{1}{l_i} \text{ with poly-A tailing)} \qquad (3.7)
\end{aligned}$$

By conditional probability, and using the fact that for a given read n, $z_{nij} = 1$ for one value of i, j and 0 for all others for the step in Equation 3.8,

$$\begin{aligned}
P(r|z_{nij} = 1) &= P(r_n|z_{nij} = 1, G_n = i, S_n = j) \\
&= \frac{P(r_n, z_{nij} = 1|G_n = i, S_n = j)}{P(z_{nij} = 1|G_n = i, S_n = j)} \\
&= P(r_n, z_{nij} = 1|G_n = i, S_n = j) \\
&= z_{nij} \cdot P(r_n|G_n = i, S_n = j) \qquad (3.8) \\
&= z_{nij} \cdot P(r_n|z_{nij} = 1) \qquad (3.9)
\end{aligned}$$

Combining Equations 3.7 and 3.9,

$$P(r_n|G_n = i) = \frac{1}{l_i} \sum_j z_{nij} \cdot P(r_n|z_{nij} = 1)$$

Using this result and the data likelihood in Equation 3.6:

$$\log[P(r|\theta)] = \sum_n \log\left[\sum_i P(r_n|G_n = i, \theta) \cdot \theta_i\right]$$

$$\begin{aligned}
\log[P(r, z|\theta)] &= \sum_n \log\left[\sum_i \sum_j z_{nij} \cdot \frac{\theta_i}{l_i} P(r_n|z_{nij} = 1)\right] \\
&= \sum_n \log\left[\prod_i \prod_j [\frac{\theta_i}{l_i} P(r_n|z_{nij} = 1)]^{z_{nij}}\right] \text{ (z_{nij} binary and is 1 for only one n, i, j)} \\
&= \sum_{n,i,j} z_{nij} \cdot \log[\frac{\theta_i}{l_i} P(r_n|z_{nij} = 1)]
\end{aligned}$$

We now iterate between the E and M steps of the algorithm. In the E step, we compute the probabilities of z given an estimate of θ and the data r.

$$\hat{z}_{nij} = q(z_{nij}) = P(z_{nij}|r, \theta^{(t)})$$

$$\begin{aligned}
\hat{z}_{nij} &= P(z_{nij}|r_n, \theta^{(t)}) \\
&= \frac{P(r_n|z_{nij}, \theta^{(t)}) \cdot P(z_{nij} = 1|\theta^{(t)})}{P(r_n|\theta^{(t)})} \\
&= \frac{1 \cdot P(r_n, G_n = i, S_n = j|\theta^{(t)})}{P(r_n|\theta^{(t)})} \text{ (since } P(r_n|z_{nij}, \theta^{(t)}) = P(r_n|r_n, s_n, g_n, \theta^{(t)}) = 1) \\
&= \frac{P(r_n|S_n = j, G_n = i) \cdot P(S_n = j|G_n = i) \cdot P(G_n = i|\theta^{(t)})}{P(r_n|\theta^{(t)})} \text{ from Eqn 3.3} \\
&= \frac{P(r_n|z_{nij}) \cdot \frac{\theta_i}{l_i}}{P(r_n|\theta^{(t)})} \text{ from Section 3.4} \\
&= \frac{P(r_n|z_{nij}) \cdot \frac{\theta_i}{l_i}}{\sum_{i',j'} P(r_n, G_n = i', S_n = j'|\theta^{(t)})} \text{ law of total probability} \\
&= \frac{P(r_n|z_{nij}) \cdot \frac{\theta_i}{l_i}}{\sum_{i',j'} P(r_n|z_{ni'j'}) \cdot \frac{\theta'_i}{l'_i}} \text{ from Eqn 3.3 and Section 3.4}
\end{aligned}$$

where $P(r_n|z_{nij})$ is given in Equation 3.4.

In the M step, we update the estimates of θ to increase the data likelihood based on new z values.

$$
\begin{aligned}
Q(\theta|\theta^{(t)}) &= \sum_{n,i,j}\left[q(z_{nij})\cdot\log(P(r,z_{nij}|\theta^{(t)})\right] \\
&= \sum_{n,i,j}\left[\hat{z}_{nij}\cdot\log[\frac{\theta_i}{l_i}P(r_n|z_{nij}=1)]\right]
\end{aligned}
$$

We need to maximize $Q(\theta|\theta^{(t)})$ under the constraint that $\sum_i \theta_i = 1$ (the probability that any isoform produces a read is 1). This requires the use of The Method of Lagrange Multipliers (Trench 2013). The Method of Lagrange Multipliers is a technique to solve for the maximum/minimum of a multivariate function $f(\theta_1, \theta_2, ...)$ under some constraint function $g(\theta_1, \theta_2, ...) = c$. In our application of the method to the EM algorithm in RNA sequencing, the multivariate function $f(\theta_1, \theta_2, ...)$ is $Q(\theta|\theta^{(t)})$ and the constraint $g(\theta_1, \theta_2, ...) = c$ is $\sum_i \theta_i = 1$. Using the Method of Lagrange Multipliers,

$$\nabla Q(\theta|\theta^{(t)}) = \lambda \nabla g(\theta)$$

and we solve for the values of θ_i in terms of λ. These values are used in the constraint function to solve for the value of the Lagrange Multiplier λ. The values of θ that maximize $Q(\theta|\theta^{(t)})$ can be found using the value of λ in the $L(\theta, \lambda)$, an equation that represents the likelihood of the data. Take the derivative and set it to 0:

$$L(\theta, \lambda) = Q(\theta|\theta^{(t)}) - \lambda g(\theta) = 0$$

First, we solve for θ in terms of λ.

$$
\begin{aligned}
\frac{\partial}{\partial\theta_i}Q(\theta|\theta^{(t)}) &= \lambda\frac{\partial}{\partial\theta_i}g(\theta) \\
\frac{\partial}{\partial\theta_i}\sum_{n,i,j}\left[\hat{z}_{nij}\cdot\log[\frac{\theta_i}{l_i}P(r_n|z_{nij}=1)]\right] &= \lambda\frac{\partial}{\partial\theta_i}\sum_i\theta_i \\
\sum_{n,j}\frac{\partial}{\partial\theta_i}\left[\hat{z}_{nij}\cdot\log(\theta_i)\right] &= \lambda \\
\sum_{n,j}\left[(\hat{z}_{nij}\cdot\frac{1}{\theta_i})\right] &= \lambda \\
\frac{1}{\theta_i}\sum_{n,j}\hat{z}_{nij} &= \lambda \\
\theta_i &= \frac{\sum_{n,j}\hat{z}_{nij}}{\lambda}
\end{aligned}
$$

Using these values in the constraint $g(\theta) = \sum_i \theta_i = 1$:

$$\begin{aligned} \sum_i \theta_i &= 1 \\ \frac{\sum_{n,i,j} \hat{z}_{nij}}{\lambda} &= 1 \\ \lambda &= \sum_{n,i,j} \hat{z}_{nij} \end{aligned}$$

Now we can solve

$$\begin{aligned} \frac{\partial L(\theta, \lambda)}{\partial \theta_i} &= \frac{\partial}{\partial \theta_i}\left(Q(\theta|\theta^{(t)}) - \lambda g(\theta)\right) \\ 0 &= \frac{1}{\theta_i}\sum_{n,j} \hat{z}_{nij} - \sum_{n,i,j} \hat{z}_{nij} \cdot 1 \\ \theta_i &= \frac{\sum_{n,j} \hat{z}_{nij}}{\sum_{n,i,j} \hat{z}_{nij}} \\ \hat{\theta}_i^{t+1} &= \frac{\sum_{n,j} \hat{z}_{nij}}{N} \end{aligned}$$

Given the n^{th} read, $\sum_{i,j} \hat{z}_{nij} = 1$ because each read must come from some start position within the isoforms with probability 1. Thus, given all n reads $\in \{1, ..., N\}$, $\sum_n \sum_{i,j} \hat{z}_{nij} = \sum_n 1 = N$.

To summarize:

E Step: Compute values of missing information z given an estimate of θ at time t.

$$\hat{z}_{nij} = \frac{P(r_n|z_{nij}) \cdot \frac{\theta_i}{l_i}}{\sum_{i',j'} P(r_n|z_{ni'j'}) \cdot \frac{\theta'_i}{l'_i}} \text{ with poly-A tails}$$

$$\hat{z}_{nij} = \frac{P(r_n|z_{nij}) \cdot \frac{\theta_i}{l_i - L + 1}}{\sum_{i',j'} P(r_n|z_{ni'j'}) \cdot \frac{\theta'_i}{l'_i - L + 1}} \text{ without poly-A tails}$$

M Step: Update the estimates of θ based on new z values that maximize the data likelihood.

$$Q(\theta|\theta^{(t)}) = \sum_{n,i,j} \left[\hat{z}_{nij} \cdot \log[\frac{\theta_i}{l_i} P(r_n|z_{nij} = 1)]\right]$$

$$\hat{\theta}_i^{t+1} = \frac{\sum_{n,j} \hat{z}_{nij}}{N}$$

Chapter 4

Simulations of the EM Algorithm in RNA Sequencing

Simulations were run using the derived EM Algorithm for RNA Sequencing:

E Step: Compute values of missing information z given an estimate of θ at time t.

$$\hat{z}_{nij} = \frac{P(r_n|z_{nij}) \cdot \frac{\theta_i}{l_i}}{\sum_{i',j'} P(r_n|z_{ni'j'}) \cdot \frac{\theta'_i}{l'_i}} \text{ with poly-A tails}$$

$$\hat{z}_{nij} = \frac{P(r_n|z_{nij}) \cdot \frac{\theta_i}{l_i - L + 1}}{\sum_{i',j'} P(r_n|z_{ni'j'}) \cdot \frac{\theta'_i}{l'_i - L + 1}} \text{ without poly-A tails}$$

M Step: Update the estimates of θ based on new z values that maximize the data likelihood.

$$Q(\theta|\theta^{(t)}) = \sum_{n,i,j} \left[\hat{z}_{nij} \cdot \log[\frac{\theta_i}{l_i} P(r_n|z_{nij} = 1)] \right]$$

$$\hat{\theta}_i^{t+1} = \frac{\sum_{n,j} \hat{z}_{nij}}{N}$$

A few simplifying assumptions were made in the simulation.

1. Sequencing is strand specific.
2. There is no poly-A tailing.
3. The error rate in sequencing as given in Equation 3.4 is constant (i.e. the probability of an error is independent of the position within the read and the identity of the nucleotide).
4. The initial estimates of θ were set to values that correspond to equal expression of all isoforms. In Li, Ruotti, et al. 2010, initial estimates were set to the values of θ obtained from looking at the uniquely mapping reads.

5. Convergence is defined as having happened when the maximum change between θ^t and θ^{t-1} is less than 0.0001.

Example 4.1 (RSEM with 10 reads) *In a simple simulation, suppose we have the following isoforms and values of θ to produce a set of 10 reads of length 3.*

Name	θ	**Sequence**	**Generated Reads**
isoform 1	0.5	BBBBBBBBB	BBB, BBB, BBB, BBB, BBB
isoform 2	0.4	AAAAAAAAA	AAA, AAA, AAA, AAA
isoform 3	0.1	BBAAABB	BBA

Notice that the reads AAA are multireads: they could have originated from isoform 2 and isoform 3. Because we generated the reads, we know that they are from isoform 2 but in real RNA sequencing, in such circumstances we do not know which isoform originally produced the read. This is the reason why we need to implement the EM algorithm to align reads and estimate expression.

First, we initiate values of $\hat{\theta}^0$ that represent equal expression. This means that

$$\tau_i = \frac{1}{3} = \frac{\frac{\theta_i}{l_i}}{\sum_{k=1}^{3} \frac{\theta_k}{l_k}}$$

$$\hat{\theta}_1^0 = 0.384...$$
$$\hat{\theta}_2^0 = 0.384...$$
$$\hat{\theta}_3^0 = 0.263...$$

Next, we start iterating through the EM algorithm until convergence. This process is illustrated in Figure 4.1. We start with the initial values of $\hat{\theta}^0$. We calculate the $\hat{z}$ values next in the E step. Figure 4.1 displays the results from this step for $r_1 = AAA$. For example, the probability that r_1 came from a given start position in isoform 1 is $z_{1,1,j} = 0.000000000125$ because all start positions in isoform 1 would result in the same BBB read sequence. The probability that r_1 came from isoform 1 is then $\sum_j z_{1,1,j} = 7 \cdot 0.000000000125 = 0.000000000877$. We update the estimates of $\hat{\theta}$ in the M step. Iterations between the E step and M step continue until convergence.

In Example 4.1, we stop after 6 iterations and have final estimates of $\hat{\theta}_1 = 0.502$, $\hat{\theta}_2 = 0.375$, and $\hat{\theta}_3 = 0.122$. Notice that in each successive iteration, $\hat{\theta}$ tends to get closer and closer to θ (Figure 4.2).

Iteration	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\theta}_3$
0	0.3684	0.3684	0.2632
1	0.5006	0.3499	0.1495
2	0.5017	0.3684	0.1300
3	0.5018	0.3735	0.1246
4	0.5019	0.3749	0.1231
5	0.5019	0.3753	0.1227
6	0.5019	0.3754	0.1226

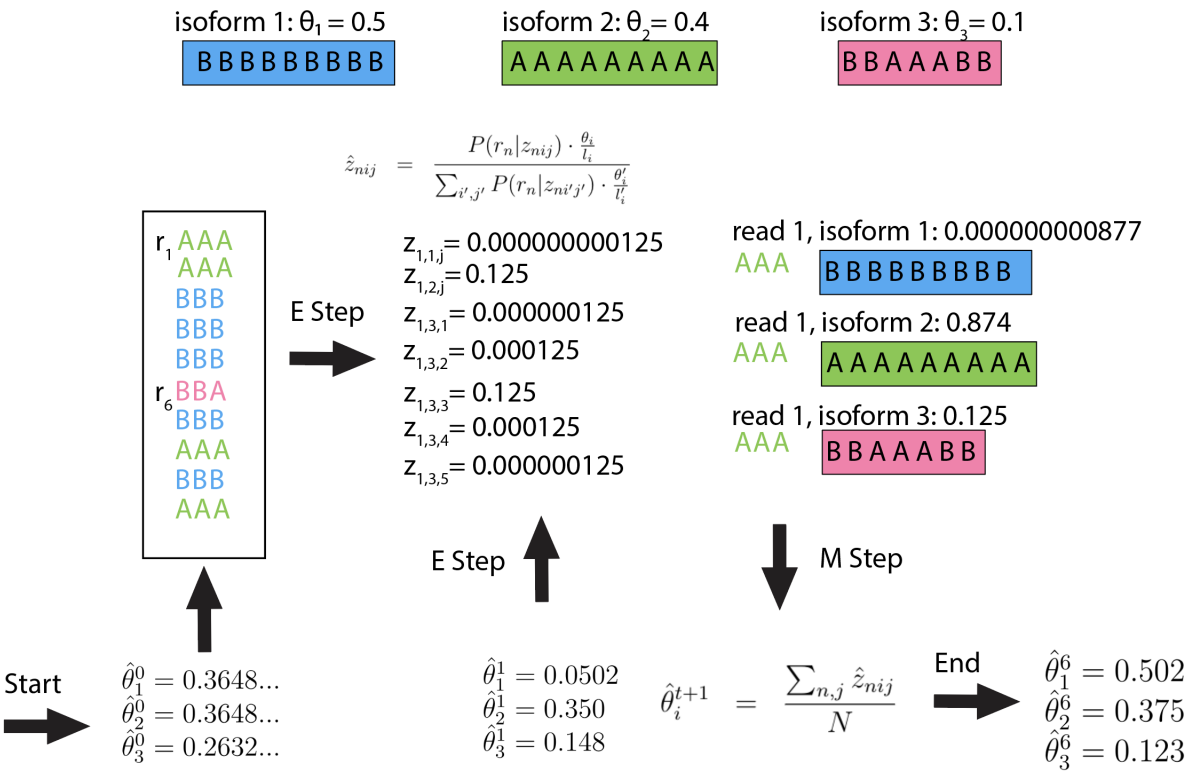

Figure 4.1: Parameter estimation for RNA Sequencing with the EM Algorithm

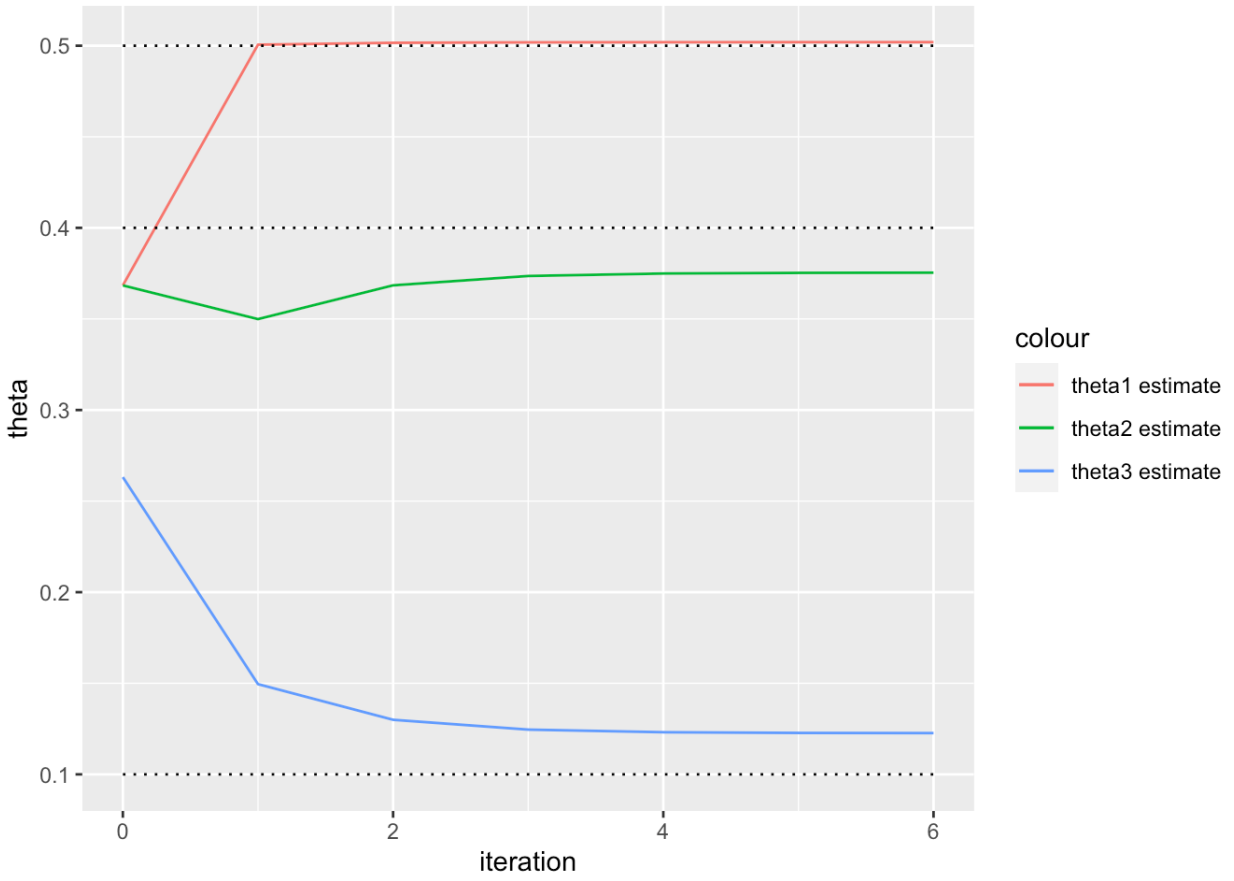

Figure 4.2: Theta estimates from the EM Algorithm as a function of the iteration number. The dotted lines represent the true values of theta, and in this specific instance, the sample values of theta as well.

Example 4.2 (RSEM with variable number of reads) *In a simple simulation, suppose we have the following isoforms and values of θ to produce sets of reads of length 3, with the number of reads in each set ranging from 5 to 1000.*

Name	θ	**Sequence**
isoform 1	0.5	BBBBBBBBB
isoform 2	0.4	AAAAAAAAA
isoform 3	0.1	BBAAABB

Example 4.2 follows the same process as Example 4.1 but extends it beyond 10 reads. From Example 4.2, we see that as the number of reads increases, there is a decrease in the error compared to both the true values of θ and the values of θ in our simulated samples (known due to the method of simulating reads) (Figure 4.3). When multireads exist, the $\hat{\theta}$ error for the isoforms that can produce multireads is higher than for those that don't. Isoform 1 produces unique reads while isoforms 2 and 3 can produce multireads. In Figure 4.3, the error for isoform 1 ($\hat{\theta_1}$ compared to true and sample θ) is lower than that for isoforms 2 and 3, especially when the number of reads is less than 200.

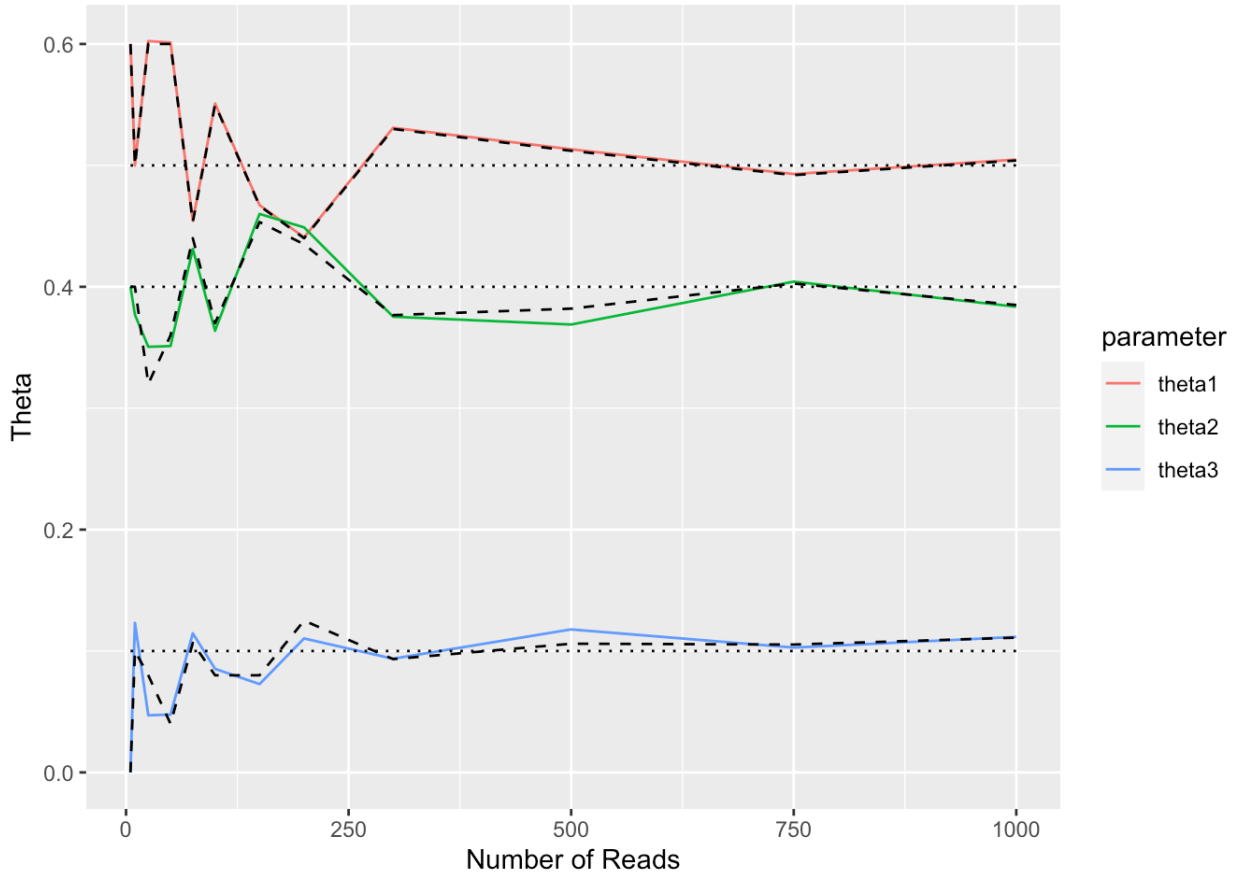

Figure 4.3: Estimates from the EM Algorithm as a function of the number of reads. The dashed line represents the sample values of θ, and the dotted line represents the true values of θ

Chapter 5

Conclusion

The RSEM program developed by Li and Dewey provides a method of aligning reads produced by RNA Sequencing in order to estimate gene expression. In this paper, each step in RSEM is detailed to provide a better understanding of the mathematical foundations of RSEM. Through simple simulations, we have shown that the use of the EM algorithm does provide expression estimates that converge to the true values, the accuracy tends to improve with larger numbers of reads, and estimates are more accurate for isoforms that cannot produce multireads.

In practice, many other factors such as variable read length, sequencing quality scores, and pair-end versus single-end reads are included into RSEM. The inclusion of more factors increases its versatility and accuracy of expression estimates (Li and Dewey 2011). The oversimplified model presented here, while not used in practice, is the base upon which the current RSEM program is built.

Bibliography

Andrew, Ng (Nov. 2018). {*CS*}*229 Lecture Notes*. URL: http://cs229.stanford.edu/notes/cs229-notes8.pdf.

Charniak, E (1991). "Bayesian Networks Without Tears". In: *AI Magazine*, pp. 50–63.

Cheplyaka, Roman (Jan. 2017). *Theory behind RSEM*. URL: https://ro-che.info/articles/2017-01-29-rsem.

DeGroot, M.H. and Schervish M.J. (2011). *Probability and Statistics*. Vol. 4. Pearson.

Dempster, A, N Laird, and D Rubin (1977). "Maximum Likelihood from Incomplete Data via the EM Algorithm". In: *Journal of the Royal Statistical Society* 39.1, pp. 1–38.

Do, Chuong and Serafim Batzoglou (Sept. 2008). "What is the Expectation Maximization Algorithm?" In: *Nature Biotechnology* 26, pp. 897–899.

Li, Bo and Colin N Dewey (Apr. 2011). "RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome". In: *BMC Bioinformatics* 12, p. 323. DOI: 10.1093/bioinformatics/btp692.

Li, Bo, Victor Ruotti, et al. (Feb. 2010). "RNA-Seq gene expression estimation with read mapping uncertainty". In: *Bioinformatics* 26.4, pp. 493–500. DOI: 10.1093/bioinformatics/btp692.

Malygin, S.N. and M.M. Postnikov (2011). *Standard Simplex*. In: *Encyclopedia of Mathematics*. URL: http://www.encyclopediaofmath.org/index.php?title=Standard_simplex&oldid=17060.

Pearl, J. (2009). *Causality*. Cambridge University Press.

Pereira, David M. et al. (2015). "Chapter 2.1.2 - "Omics" Technologies: Promises and Benefits for Molecular Medicine". In: *Principles of Translational Science in Medicine (Second Edition)*. Ed. by Martin Wehling. Second Edition. Boston: Academic Press, pp. 25–39. ISBN: 978-0-12-800687-0. DOI: https://doi.org/10.1016/B978-0-12-800687-0.00003-7. URL: http://www.sciencedirect.com/science/article/pii/B9780128006870000037.

Trench, W. F. (2013). *The Method of Lagrange Multipliers*. URL: http://ramanujan.math.trinity.edu/wtrench/texts/TRENCH_LAGRANGE_METHOD.PDF.

Wang, Yan et al. (Jan. 2015). "Mechanism of alternative splicing and its regulation". In: *Biomedical reports* 3.2, pp. 152–158. DOI: 10.3892/br.2014.407.

Zhang, Chi et al. (July 2017). "Evaluation and comparison of computational tools for RNA-seq isoform quantification". In: *BMC Genomics* 18, p. 583. DOI: 10.1186/s12864-017-4002-1.