1 Introduction

First, we layout the biological structure we are attempting to study. In each cell there are 23 pairs of chromosomes that include several genes. Each gene is "expressed" at varying levels. This expression is typically measured by something called *mRNA transcripts*. mRNA sequences are a key coding to protein creation. In particular RNA codes the DNA with respect to some reference strand (usually 5'). Cells transcribe these mRNA sequences and then discard introns that have no relevant coding information for protein creation. After this step, the mRNA exits the nucleus to code the creation of proteins. In high-throughput analysis we wish to measure gene expression or

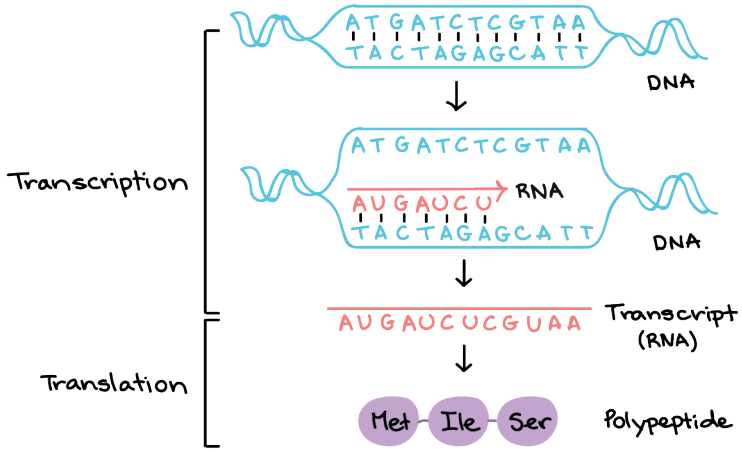

Figure 1: A visual representation of RNA transcription.

gene activity. One way in which we can do this is to count, at some point in time, the number or mRNA transcriptions the cell is producing from each gene. The thought process being that if more proteins are being coded by gene i then gene j then gene i is more active than gene j. These

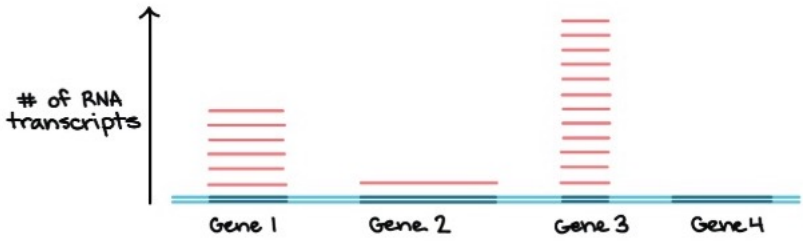

Figure 2: A visual representation of gene expression.

counts will serve as the basis for our analysis procedure. In that we will focus on the following data matrix Having established a biological motivation to the reads per gene datastructure we now turn to actually collecting this data.

Gene ID	Sample 1	Sample 2	...	Sample N
A1BG	24	30	...	17
ASM	24	500	...	343
⋮	⋮	⋮	⋮	⋮
B2BM	100	2	...	19

Figure 3: Counting "reads" per gene. The main data structure of our analysis.

2 RNA-Seq

In general, high throughput analysis tells us two things (i) which genes are active (ii) how much they are transcribed. In the early 2000s the most popular method was the microarray. RNA-Seq is beginning to replace microarrys. This is due to a couple of issues (i) microarrays are sensitive in the imagining is designed and (ii) RNA-Seq does not require designing probs (iii) RNA-seq allows for novel gene discovery. For this purpose we focus on RNA-Seq an approach to sequencing gene reads. This process can be broken into three different stages of analysis.

2.1 Library Preparation

This stage of the process is simply data preparation before using a sequencing machine. I won't discuss the molecular chemistry at play here but instead layout the linear process of this analysis.

1. Step 1: Isolate the RNA. This is done through the use of RNA polymerase that effectively "peels" the RNA off of the DNA double helix structure.
2. Step 2: These RNA sequences are usually quite long ($>$ 10k base pairs). Most sequencers are only accurate up to around 100 base pairs. So for this reason we break up the RNA into small fragments
3. Step 3: Express these small fragments back into double helix strand DNA. This is done for stability issues.
4. Step 4: Add sequence adapters to these fragments. This tells the machine where each strand starts and stops. Note: These adapters will not be added to all fragments and we will lose some information as a result.
5. Step 5: Amplify the fragments. (PCR Amplification). In essence this is making the fragments larger so the reads will be easier to make.
6. Step 6: Quality control e.g. Ensure that no fragments are just two adapters.

2.2 Sequencing

Once these fragments have been prepared we will have an incredibly large set of fragments of the form $(AGGCTCA, \ldots)$ that we need to sequence. One of the most popular current techniques is the illumnia chip or flow cell. One can thing of the cell as a large grid. From here, the fragments are aligned vertically above a single cell on the chip. The fragments are then passed *through* the cell

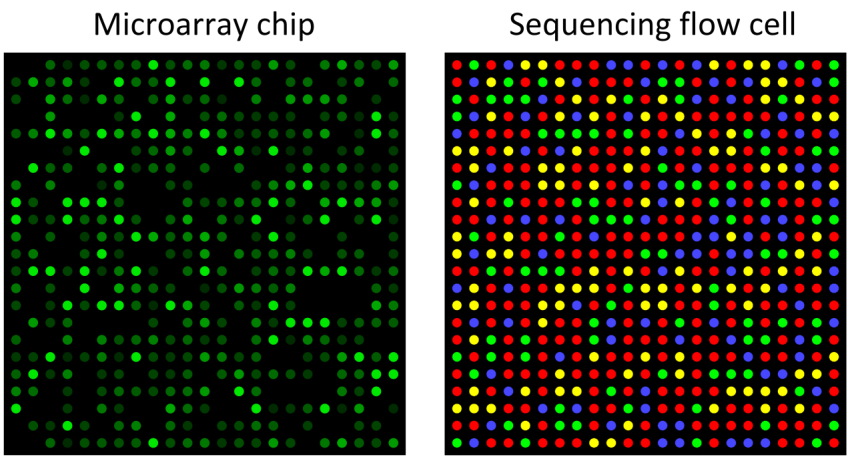

Figure 4: A single read or an illumni flow cell. This will be repeated for the entire length of the fragment.

as it reads which base in each region. A figure of one read is given below. This method introduces a few different QC issues. Namely for different length segments, we will see low diversity in certain regions of the grid. Issues surrounding this sequencing tool has been well studied and solutions to these problems exist and won't be discussed here.

Once we have sequenced each fragment, the fragments need to be realigned to the genome. This alignment is the focus of many bioinformatists. TopHat and STAR offer computational solutions of fuzzy matching fragments to a reference genome. From here, we can attribute each read to each gene and begin constructing the reads per gene matrix.

2.3 Analysis

Once we have arrived at our final read per gene matrix, a few normalization measures need to be taken. In particular the reads depend on the total library size as well as the gene length. For this reason we define the following normalized quantities.

$$RPKM = \frac{\#\text{Reads}}{\text{gene length}/1000 * \text{total no. reads}/1,000,000}$$
$$CPM = \frac{\text{Reads Mapped to Gene}}{\text{total no. reads} * 1,000,000}$$
$$FPKM = \text{fragments per kilo base per million mapped fragments}$$

FPKM is used for paired end reads. While these transformations are rather naive in practice, we can also develop count models such as Poission or Negative Binomial models. These two models are popularized by the software packages edgeR and DESeq2.

2.4 edgeR

Suppose we have n RNA-Seq sample that is sequenced, mapped to the genome, and the reads are counted by each gene. The number of reads for sample i mapped to gene g is given by y_{gi}. The set

of genewise counts for sample i makes up the *library* for that sample. We then model the expected size of each count as the relative abundance times the library size (m_i). In this model, we model the entries of the reads per count matrix as a negative binomial. In particular, if we take the model

$$\begin{aligned} y_{gi} &\sim NB(m_i\lambda, \phi) \\ f(y;\mu\phi) &= \frac{\Gamma(y+\phi^{-1})}{\Gamma(\phi^{-1})\Gamma(y+1)}\left(\frac{1}{1+\mu\phi}\right)^{\phi^{-1}}\left(\frac{\mu}{\phi^{-1}+\mu}\right)^{y} \\ \mathbb{E}(Y) &= \mu, \mathrm{Var}(Y) = \mu + \phi\mu^2 \end{aligned}$$

If we suppose that the library size for each sample is equal across samples, we have

$$Z_g \equiv \sum_{i=1}^{n} y_{gi} \sim NB(mn\lambda, \phi)$$

From here we could just use conditional maximum likelihood conditioned on the number of reads for individual i

$$\ell_{y_{gi}|Z=z}(\phi) = \sum_{i=1}^{n} \log\Gamma(y_{gi}+\phi^{-1}) + \log\Gamma(n\phi^{-1}) - \log\Gamma(z+n\phi^{-1}) - n\log\Gamma(\phi^{-1}) \tag{1}$$

In unequal library settings, we simply interpolate these counts as if they were from $NB(m^*\lambda, \phi)$ for $m^* = (\prod_{i=1}^{n} m_i)^{1/n}$. This interpolation follows the following procedure.

1. Initialize ϕ
2. Estimate λ given ϕ
3. Find the proportion of p_i of y_i as follows $p_i = \mathbb{P}(Y \le y_i; m_i\lambda, \phi) + \frac{1}{2}\mathbb{P}(Y = y_i; m_i\lambda, \phi)$
4. Find the p_ith percentile assuming $NB(m^*\lambda, \phi)$. We call this the psuedo-data.
5. Calculate ϕ using CML on the psuedo data
6. Repeat until ϕ converges

Another method that does not rely on assuming this equal library size is based on weighted likelihood function.

$$WL(\phi_g) = \mathcal{L}_g(\phi_g) + \alpha\mathcal{L}_c(\phi_g) \tag{2}$$

Here the problem reduces to estimating α. It does so by matching moments and normal approximations.

2.5 DESeq2

A similar model to edgeR, DESeq2 uses a negative binomial model given by the following

$$\begin{aligned} y_{ij} &\sim NB(\mu_{ij}, \alpha_i) \\ \mu_{ij} &= s_i q_{ij} \\ s_i &= \mathrm{median}_{y_i^R \neq 0} \frac{y_{ij}}{y_i^R} \quad y_i^r = \left(\prod_{j=1}^{m} y_{ij}\right)^{1/m} \\ \log q_{ij} &= \beta^T x_j \end{aligned}$$

From here we estimate the dispersions using the Bayesian model

$$\log \alpha_i \sim N(\log \alpha_{tr}(\overline{\mu}_i), \sigma_d^2)$$
$$\overline{\mu}_i = \frac{1}{m} \frac{Y_{ij}}{s_j}$$

For details see the lecture slides. In essence this is an extension of edgeR to a Bayesian framework.

2.6 Voom + limma

Perhaps the most simple model, we just use a linear model with $\log(CPM)$ where we precision is adjusted by using the weights of the mean-variance relation.