Nate Evans
Christina Zheng & Shannon McWeeney
BMI650
Dec. 4 2018

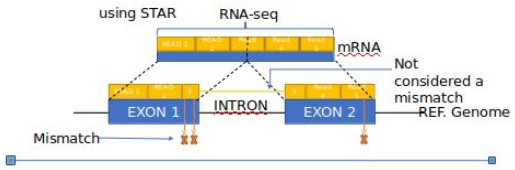
Final Project

Background

The goal of this project is to select an optimal parameter of allowable sequence mismatches, denoted T, for an alignment software such as **STAR** or TOPHAT to best align RNA-seq 100-bp reads from the wild type PWK of mus musculus to the reference genome B6. We are given the following data:

chr1_PWK_PhJ.mgp.v5.snps.dbSNP142.vcf	This shows common SNPs (variation) in chromosome 1 of B6. (Assembly 38)
Mus_musculus.GRCm38.dna.chromosome.1.fa	This is the nucleotide sequence for chromosome 1 of B6 (Assembly 38)
PWK_R1.fastq	These are single paired RNA-seq reads from an Illumina Hi-Seq2500
Mus_musculus.GRCm38.86.gtf	Full B6 reference genome (Assembly 38)

Problem

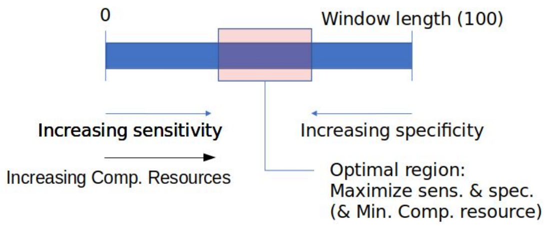


There are two approaches to be considered in this paper:

1. What is a naive approach for predicting an optimal T to align the PWK RNA-seq data?
2. How can we estimate an optimal T to align the PWK RNA-seq data from only the RNA-seq data?

The first question will be the majority of this paper, and the latter will be discussed in the Extension section.

To begin, we know that T can be any value contained within our read length, so for read lengths of 100, our T value can be [0,100]. There are three major trends to keep in mind while developing our algorithm and choosing a T. First, as T increases from zero, the sensitivity of our aligner will increase, but the computational resources necessary will also increase. As T decreases from 100, the specificity of our aligner will increase. From this, it is easy to see that between these two extremes, there is an optimal value of T. This is illustrated in the graphic below.



Method

Part 1:
The brute force method I employ to predict a value of T relies on several key assumptions:

1. The variation in the exonic regions of chromosome 1 of B6 is similar to the variation within the rest of the genome.
2. The variation in the exonic regions of the B6 chromosome is similar to the variation in exonic regions of PWK genome.

To predict a value of T, this approach looks at the variation in B6 chromosome 1, as provided by the .vcf file, and sets a threshold of T based a percentage of the data that will be properly aligned given a threshold T. To build on this, the algorithm pulls the B6 chromosome 1 genes from entrez and calculates variation for the exonic (gene) regions only, for better external validity when applied to RNA-seq data.

Step 1: create SNPs mask

Create a boolean array marking the location of SNPs on chromosome 1; Array index matches nucleotide index.

Step 2: create exon mask

Pull the genes located on chromosome 1 of B6 from entrez, parse and create a boolean mask marking the exonic regions, as before, array index matches nucleotide index.

MISSING STEP: Splice out introns by using STAR's canonical splice sites (GT/AG, GC/AG, AT/AC) or by pulling refSeq data from entrez. Because this is missing the 'exonic' regions are really gene-regions as they include both introns and exons.

Step 3: calculate variation per 100-mer window over B6 chromosome 1

Iterate through each 100-mer window of the exonic regions of B6 chromosome 1 and count the number of SNPs present.

Step 4: Plot variation and generate T outputs

For each given T, calculate the number of 100-mers that would be rejected, thus counting the False Negatives (FN) at each proposed T value.

Step 5: Choose T

Based on the computational resources available and how much falsely rejected data is acceptable, choose your T. For my algorithm, a threshold T that included 99.9% of the data was chosen.

Part 2:

The second part of this algorithm focuses on trying to set an upper bound on T by calculating theoretical False Positives and True Negatives. This is done by making assumptions on the statistically random variation in probability. Due to lacking validation on any models, I chose a simple naive/worst-case model:

$$P_{FalsePositive} = \frac{permutations\ sub - seq\ with\ T\ mismatches}{permutations\ of\ window\ K} = \frac{4^T + k}{4^k}$$

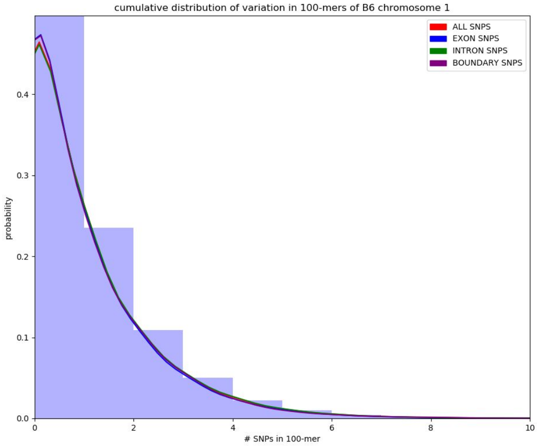
$$\alpha = 0.05 -> \frac{0.05}{l - 101} = \frac{4^T + k}{4^k} -> T = 84.05$$

$$l = chr1\ length$$

Using a bonferroni adjustment for an alpha value of 0.05, I calculated the threshold T which gives this probability. That is, in a random sequence of nucleotides, what number of mismatches will commonly accept random alignments? The answer is: greater than 84 mismatches. Granted, this upper bound is so high that it's not terribly helpful, and the computational resource argument will weight the T value heavily toward the lower bound. However, with a better model for the Probability of a false positive, this may be able to push the upper bound down significantly.

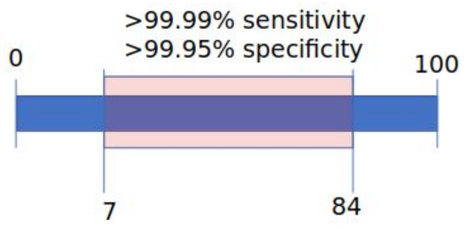
Results

After calculating the SNPs distribution for exonic, intronic, boundary and all regions, the plot below was produced. As you can see, the majority of windows had less than 4 SNPs, and the maximum exonic region SNPs per 100-mer was 20.



Additionally, the sensitivity was calculated for each threshold T applied to the exonic windows. If a window had greater variation (more SNPs) than T, it was considered a False Negative (FN). Alternatively, if a window had less variation (fewer SNPs) than T, it was considered a True Positive (TP). From this we can calculate the sensitivity as: TPR = TP / (TP + FN).

Region-Chr1	Mean	Std. Dev.	Max	Sensitivity (T)			
				99% FN: 2e6	99.9 % FN: 2e5	99.99 % FN: 3e3	99.999 % FN: 534
ALL	0.834	1.259	26				
GENE REGION	0.801	1.221	20	4	7	13	16
NON-GENE REGION	0.848	1.273	26				
BOUNDARY	0.817	1.269	20				

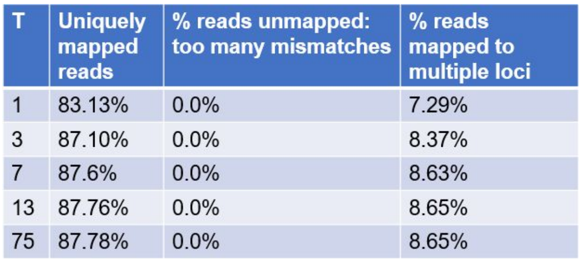


My final conceptual bounds of T. I chose the lower limit (7).

Sequence Alignment (STAR)

Using STAR and the command:

```
evansna@state:~/aligners/STAR-2.6.1d/bin/Linux_x86_64$ ./STAR
--runMode alignReads --genomeDir
/home/courses/BMI550/FinalProject/genomeIndex/STAR/GRCm38
--outFileNamePrefix
/home/users/evansna/aligners/STAR_OUTPUTS/output.txt
--readFilesIn
/home/courses/BMI550/FinalProject/small_PWK_R1.fastq
--outFilterMismatchNmax T
```



T	Uniquely mapped reads	% reads unmapped: too many mismatches	% reads mapped to multiple loci
1	83.13%	0.0%	7.29%
3	87.10%	0.0%	8.37%
7	87.6%	0.0%	8.63%
13	87.76%	0.0%	8.65%
75	87.78%	0.0%	8.65%

From this data, it appears that STAR is far less sensitive to the T threshold then I had expected. In fact, it didn't affect the "% reads unmapped: too many mismatches" at all.

Extension

To extend this technique for general use to on any strain, I propose the following algorithm. In essence it still relies on much of the algorithm discussed above, but provides variation data specific to the RNA-seq data we have collected. Using a small subset of our provided RNA-seq data, perform a global optimal alignment to the reference genome using an algorithm such as shortest path. From this alignment data, calculate the variation distribution of the aligned RNA-seq and predict a threshold T by the algorithm described above. The advantage of this method is that the variation data comes directly from the RNA-seq data, rather than a reference genome such as B6. This is especially beneficial when aligning PWK since we know that the B6 genome is inbred and thus variation is far lower than in naturally occurring mice, such as strain PWK. One major drawback of this method is that to get a viable amount of variation data requires optimal global pairwise alignment of many RNA-seq reads to large reference sequences, which will take significant computational resources. For a pseudocode describing this approach, please see evans_extension_pseudocode.py

Discussion

The major step that this algorithm missed was splicing out the introns from genes, as discussed in the methods section. As is, the algorithm does not analyze variation in exonic regions but rather, gene regions and because of this is less applicable to RNA-seq. Additionally, when pulling genes from entrez, it was difficult to provide computational QC to verify that I had pulled the right gene. I suspect that my entrez search term needs some refinement and/or additional QC methods should be developed to verify the right gene indices are included. Lastly, the model for probability of a false positive rate is almost certainly incorrect, but the conceptual two pronged approach still holds validity and with refinement I think could be a useful approach.