Programming Assignment 4 - An exercise in isoform quantification

CS/BIOINFO M122/M222

Due: Sunday, June 6th, 11:59pm

Introduction

This RNA Sequencing assignment was designed to help you understand the basics of isoform quantification, which is critical for deciphering gene function and regulation.
In this assignment, your input is a 100-million base genome, reads generated from that genome, and a data file that gives the locations of the genes and their exons as well as the isoforms for those genes.
Your task is to align reads to the exons and determine the abundance of the isoforms.
To do this, you will apply the least squares method that you learned about in class.
As you work on this small assignment, think of how you would scale up the process for hundreds of genes.

The files you need to complete this assignment:
* Genome file - full_genome.txt
* File containing single-ended reads of length 50 base pairs - shuffled_reads.txt
* Gene annotation file - DATA_PA_1000_0

File Formats
* full_genome.txt file has the entire genome in one string.
* shuffled_reads.txt file contains one read of length 50 bases in each line.Only a small percentage of these reads will map for this project.
* DATA_PA_1000_0 contains the following:
  - N -- the first line in the file, indicating the number of genes in the file
  - Then for each gene i:
    - e_i -- number of exons for gene i
    - s_i_1 s_i_2 ... s_i_M -- the starting index of each of the exons
    - e_i_1 e_i_2 ... e_i_M -- the ending index of each of the exons
    - l_i -- number of isoforms for gene i
    - then l_i lines containing combinations that show which exons are in each isoform, e.g.:
      - 2 3
      - 1 2

Note that the genome is indexed starting at 0 and the coordinates of the exons are inclusive. E.g: An exon with start coordinate 4 and end coordinate 10 contains bases 4, 5, 6, 7, 8, 9 and 10.

These are examples that might appear as the first two genes in the file DATA_PA_1000_0.txt:

```
5
3
90792 91139 91314
90872 91233 91412
2
0 1 2
1 2
3
571929 572247 572400
571995 572328 572497
1
0 1 2
```

From this data, we see that N=5, and for the first gene i=1, we have:

e_1 = 3,

s_1_1 = 90792, s_1_2 = 91139, s_1_3 = 91314

e_1_1 = 90872, e_1_2 = 91233, e_1_3 = 91412

l_1 = 2, with exons 0, 1, 2 for the first isoform and 1, 2 for the second isoform.

Here is the translation of that to plain English. For the first gene in the file, there are three exons. Exon 1 starts at position 90792 and ends at position 90872. Exon 2 starts at position 91139 and ends at position 91233. Exon 3 starts at position 91314 and ends at position 91412. There are two gene isoforms. The first isoform contains all three exons, while the second isoform contains only the second and third exons (which are called 1 and 2 due to zero-indexing).

For the second gene (i=2), we have e_2 = 3, s_1_1 = 571929, and so on.

For quantification, you need to align the reads to the exon regions listed in DATA_PA_1100_0.txt and count the number of reads that map onto each exon.
Then use the method of least squares in the procedure outlined in class to solve for the isoform frequencies.

Starter Code
Starter code for the project is available at https://github.com/rosie068/CM122_starter_code. Use `git pull` in the CM122_starter_code repository you cloned to get the updated code, or you can just redownload the files directly from the link.

We are providing you with the skeleton for one script:
1. `quantify_isoforms.py` takes in a genome, a set of reads, an annotation file, an output file name, and an output header and outputs the contigs generated by assembly.

Running the above scripts with the `-h` option should be self explanatory, but here is an example of running them to create a file that can be submitted on the website.

1. Download the data from CCLE Week 9 into the HP4 folder and unzip it. The commands below assume that you have a folder named hw4_r_4/ in the HP4 folder. If you download and save things in a different place you'll have to adjust the file paths below.

2. Use `quantify_isoforms.py` as shown below.

```
python quantify_isoforms.py -g hw4_r_4/full_genome.txt -r hw4_r_4/shuffled_reads.txt -a hw4_r_4/DATA_PA_1100_0 -o test_output.txt -t hw4_r_4_chr_1
```

This will generate a file of isoforms and their abundances in test_output.txt and a zipped version of that file formatted correctly for submission.

You can submit your results as many times as you want to achieve a passing score.

Output File Format

The provided starter code will output the correctly-formatted file. In case you do not want to use the starter code, the format is described below.

The first two lines of your file should be:

```
>hw4_r_4_chr_1
>RNA
```

The solution should then be formatted with the transcript sequence and abundance in each line separated by one space.
The following is an example:

```
>hw4_r_4_chr_1
>RNA
agcttcaaaa….. .7
```

gggtcaattttg…. .3
cattggaaac…. .1
tttggaccaac… .1
ggggggcct... .8
```

### Grading

The score will be based on the accuracy of the transcript sequence and the abundance estimate.
For full credit, you should be able to get 100 both for the transcript accuracy and the abundance.
Your grade will be the average of the two scores.
Maximum grade is 100 and lowest is 50.
```