# Programming Assignment 5 - RNA-Seq

BIOINFO M260

Due: March 24th at 11:59 pm

This programming assignment is designed to teach you about RNA Sequencing.

## Overview

In this assignment, you are given single-end reads generated from RNA sequences. Your job will be to quantify the isoforms in a sample of RNA.

## Files

In the outer folder, you have a file called `ref_hw5.txt`, which will be the reference sequence for this project; it is 1 million base-pairs long, and is used as the reference sequence for all of the files. Each dataset is generated from its own unique donor sequence, which includes a small number of SNPs versus the reference.

Looking at the `hw5_W_0`, there are five files:

`reads.txt` contains 50-bp single-end reads, taken from the transcribed RNA. There are rare (about .5 percent) read errors within the reads.

`exons.txt` lists the start and end positions (in the reference genome) of the transcribed exons in the dataset. The exons are listed in their order along the genome, within each gene.

```
>hw5_W_0
GENE_1_EXON_1:257900,261358
GENE_1_EXON_2:261364,265839
GENE_1_EXON_3:265928,267072
```

This indicates that the first exon of `GENE_1` starts at position 257900 in the genome and ends at 261358; the second exon starts at 261364 and ends at 265839; the third exon of gene 1 starts at 265928 and ends at 267072; etc. You will need this data to compute the length of each exon.

`isoforms.txt` lists the exons that compose each isoform of the transcript.

```
>hw5_W_0
GENE_1_ISO_1:EXON_1,EXON_2,EXON_3,EXON_4,EXON_5,EXON_6
GENE_1_ISO_2:EXON_1,EXON_2,EXON_5,EXON_6
GENE_1_ISO_3:EXON_2,EXON_3,EXON_4
```

This indicates that isoform 1 of Gene 1 has exons 1,2,3,4,5,6 (in that order); isoform 2 has exons 1,2,5,6; isoform 3 of gene 1 has exons 2,3,4; etc.

`exon_counts.txt` is a count of which exons are included; undergrads will use this file to compute the frequency of each isoform; grads can use this file to check the accuracy of their alignment.

```
>hw5_W_0
GENE_1_EXON_1:850
GENE_1_EXON_2:1135
GENE_1_EXON_3:253
```

This indicates that exon 1 was found 850 times in the dataset, exon 2 was found 1135 times, and exon 3 was found 235 times in the dataset.

`ans.txt` are the relative frequencies of each isoform used to generate this data. Your answers should be submitted in this format, namely:

```
>dataset_name
GENE_A_ISO_B:relative frequency
GENE_A_ISO_C:relative frequency
GENE_X_ISO_Y:relative frequency
```

The order of the frequencies does not matter.

You will be graded by how close your answers are to these answers. A grader script is also included; it should work properly if your answers are correctly formatted.

# Assignment

For **undergrads**, your job will be to use the `exon_counts.txt` data, the data on the lengths of each exon in `exons.txt`, and the data on which exon is found in which isoform in `isoforms.txt` to determine the frequency of each isoform. You should submit to CCLE an answer file for `hw5_E_1` data.

For **grads**, you will modify your hasher to count the instances of each exon, count the instances of the exons, and use these counts to compute an answer file for the `hw5_M_2` data.

For extra credit, everyone is invited to try to solve `hw5_G_3`, where some of the exons and isoforms have been hidden. More details on this dataset are below.

## Algorithm

Computationally, RNA-Seq is a similar problem to viral quasispecies quantification via DNA sequencing. The key difference is that in viral quasispecies quantification, the frequency of each read is directly proportional to the frequency of the viral sequence that generated it; in RNA-Seq, the frequency of each exon is proportional to the frequency of the sequences that contain that exon, and to the length of the exon.

This can still be solved using least-squares, however, your design matrix will look slightly difference.

Consider a single RNA transcript T with 3 exons, E1, E2, and E3.

There are three isoforms of T: isoform 1 has all 3 exons (E1E2E3); isoform 2 has the first 2 exons (E1E2), isoform 3 has the last two exons (E2E3)

Since the count of each exon in our sample is proportional to its length, we need to know the lengths of each exon to solve for the frequencies.. Let's assume E1 has length 300 bp, E2 has length 1000 bp, and E3 has length 700 bp. We can encode this in a matrix as follows:

$$E = \begin{bmatrix} 300 & 300 & 0 \\ 1000 & 1000 & 1000 \\ 700 & 0 & 700 \end{bmatrix}$$

The rows of $E$ represent exons, and the columns represent the different isoforms. Suppose we run an experiment and find the counts of each exon to be as follows:

$$f = \begin{bmatrix} 240 \\ 900 \\ 490 \end{bmatrix}$$

We'd like to solve for the frequency of each isoform, which we'll call $b$, via the equation $Eb = f$.

Like viral quasispecies estimation, we can solve this using least squares estimation:

```
import numpy as np
data = [[300, 300, 0], [1000, 1000, 1000], [700, 0, 700]]
exon_counts = [240, 900, 490]
np.linalg.lstsq(data, frequencies)[0] # array([ 0.6,  0.2,  0.1])
```

This gives us relative frequencies of the three isoforms as 0.6:0.2:0.1.

## Grading

You will be graded on the quality of your relative quantifications using the cosine similarity; between your answers and the parameters used to generate the model.

For example, if the true isoform frequencies used in the above were proportional to (.5, .4, 0), your score would be:

$$\frac{.5 * .6 + .4 * .2 + 0 * .1}{\sqrt{(.5^2 + .4^2 + 0^2)(.6^2 + .2^2 + .1^2)}} = \frac{38}{41}$$

Note that the scale of your relative frequenices doesn't matter; returning frequencies of (.6, .2, .1) is no different from scores of (60, 20, 10).