Quantitative Understanding in Biology Module I: Statistics Laboratory I: RNA-Seq Analysis

In this laboratory you will be working with RNA-seq data obtained from six pediatric patients diagnosed with acute lymphoblastic leukemia (ALL). For each patient in the study, RNA-seq runs from undifferentiated lymphoblasts were performed upon diagnosis of ALL. All of the cases in this study relapsed after treatment, and RNA-seq was run on undifferentiated lymphoblasts from each patient after relapse.

The primary goal of this study is to identify the molecular mechanisms of relapse. As a first step, towards this goal, you need to identify differences in the RNA expression profiles of the two disease states (diagnosis and relapse). This is a complex problem with no clear consensus among current practitioners about how such an analysis should be done. There is therefore no single correct recipe or answer for this exercise. Any particular analysis strategy will likely involve some compromises and tradeoffs, and a key part of this exercise is your ability to identify the advantages and limitations of your chosen analyses, and justify why you chose the path you did. Using methods not specifically covered in class is encouraged, but be sure to clearly explain what those methods do and what their limitations are; i.e., you may not simply running a cookie-cutter RNA-seq analysis package (e.g. from bioconductor) without understanding and articulating all of the underlying analysis in detail.

You will receive two input files for this laboratory. The first, `df_all.Rdata`, contains an R data frame containing the results of primary analysis of the RNA-seq data. For each combination of patient and disease status (**D**iagnosis or **R**elapse), up to eight lanes of high-throughput sequencing data were run using Illumina's Solexa platform. Data from each of the lanes were aligned to a reference human genome (hg18), and reads were mapped to Ensembl transcripts using the Bioconductor package for R. The data you'll be receiving for this lab will be the number of reads that mapped to each Ensembl gene for each RNA-seq lane. The second file, `genes.Rdata`, is reference data containing the name of each Ensembl gene and the genomic loci of the transcript that comprise that gene.

The data that you are being given is unpublished, so please do not share it outside of this course.

Consider a single patient and determine which genes show evidence of differential expression. How meaningful do you think these results are? Explain.

Considering all of the patients together, determine which genes show evidence of differential expression between relapse and diagnosis. What assumptions were made in your analysis, and how do you justify them? What are the limitations and advantages of your chosen analysis?

Based on the data you see here, how might you suggest changing the experimental plan for a similar future study. For example, would you have been better off with fewer patients but deeper sequencing? Can you draw any conclusions about the need for technical and/or biological replicates when using Illumina sequencing technology?

Hints on Working with Large Datasets

An important goal of this laboratory is to gain experience in dealing with large, real-world data sets typically found in high-throughput experiments (the raw sequencing data for this experiment is over 200GB). When using R, this involves using data frames, and you should be comfortable with the basics of manipulating data frames before you start working on this lab; see any decent tutorial or book on R. In particular, functions such as `aggregate`, `apply`, `merge`, `rbind` and `subset` are your friends (but not your only friends!).

Working at this scale has practical implications that you should keep in mind. There are roughly 37,000 genes in this dataset. If an analysis takes 100ms per gene on your computer, you'll still need over an hour to complete a run. You may find it useful to work on a subset of the data while you develop your analysis methods.

Experienced programmers may naturally gravitate toward the use of `for` loops as part of their solution. While these loops work, it is often *much* more efficient to use R's vectorized functions such as `apply` and friends (`mapply`, `lapply`, `sapply`). See the article on vectorization in R News Vol 8/1 (http://cran.r-project.org/doc/Rnews/). Hint: Spend some time thinking about the structure of your data frame. Simple reorganization of your data before doing statistical analysis can often improve the efficiency of the analysis.