Programming for Biologists (MMG1002H)

Assignment 6
10 Points

Assignment goals:
Become familiar with reading in and plotting data as well as significance tests

The data from this lab are taken from the following study, which you can read about by clicking http://www.nature.com/nature/journal/v415/n6871/abs/415530a.html:

"Gene expression profiling predicts clinical outcome of breast cancer." Nature 415, 530-536 (31 January 2002). van 't Veer LJ, Dai H, van de Vijver MJ, He YD, Hart AA, Mao M, Peterse HL, van der Kooy K, Marton MJ, Witteveen AT, Schreiber GJ, Kerkhoven RM, Roberts C, Linsley PS, Bernards R, Friend SH.

We will focus on understanding the data, loading the data into R, and doing some simple manipulations of the data. The main goal of this assignment is to find genes that have significantly different expression in metastatic and non-metastatic samples, as these could be important for cancer progression.

Part I: Loading and manipulating gene expression data in R (5 points)
(1) We have provided two data files for this lab: The actual expression data (24481 genes X 78 samples) "assignment6_data.txt" and a simple file that contains the sample assignments of non-metastatic [1] and metastatic samples [2] "assignment6_sample_labels.txt" (one number per line, corresponding to columns in the data file i.e. the 5th line contains the label of the sample measured in the 5th data file column). First load all the data into R using the "read.table" command. You may want to transfer everything over into a matrix using a command like "cancerdata_matrix = data.matrix(cancerdata)".

(2) Write a series of commands that will do each of the following.

(a) Print the data for the BRCA1 gene (gene index is 12361)
(b) Compute the mean and standard deviation of the expression data for the BRCA1 gene across all samples
(c) Compute the mean and standard deviation of the expression of all genes in the first metastatic sample (column #45).
(d) Create a vector called met_samples that consists of the column indices of all metastatic samples
(e) Create a vector called nonmet_samples that consists of the column indices of all non-metastatic samples
(f) Use the vectors created in parts (d) and (e) to compute a new vector expr_difference, which contains the difference in the mean expression for each gene between the metastatic and non-metastatic sample groups. (Hint: For each gene, this will require you to compute the mean expression for that gene for each of the non-metastatic and metastatic groups and measure the difference between these two means. You can use the "apply" function to do this, but as usual, that is not the only possible solution).
(g) Using the expr_difference vector you just created, print the gene that has the

largest positive difference (i.e. metastatic < non-metastatic) and the largest negative difference (i.e. metastatic > non-metastatic) in mean expression between the metastatic and non-metastatic groups.

Part II: Statistical analysis of gene expression data (5 points)

We will find a subset of the genes that is differentially expressed using the T-test. Remember, the purpose of the t-test is to statistically quantify the difference in expression for a single gene between two groups of samples (in this case, metastatic and non-metastatic), and answer the yes/no question "is this difference significant?"

(1) Our goal is to answer the yes/no question of whether or not each gene is significant. To do this, we will need to compute a p-value that reflects the significance of the t-statistic. Small p-values reflect cases where the gene is very likely different between the two sets of samples, and typically a cutoff of 0.05 is used— genes with less than a 0.05 p-value are called significantly differentially expressed. (In a real world situation, we would have to apply multiple testing correction, such as FDR, but we will ignore this here). Use the "t.test" function in R to calculate the p-value of the t-statistic for every gene (for the difference in expression between metastatic and non-metastatic samples). You can either use a simple "for" loop, or you can use the "apply" function, but this is a bit complicated here.

(2) Answer the following questions by writing short sections of R code:
a. Print a list of significantly differentially expressed genes (p-value < 0.05), sorted in order of their significance (most to least). Include the gene name, the t-statistic, and the corresponding p-value in your list.
b. How many genes are significantly differentially expressed at a p-value < 0.05?
c. How many genes are significantly under-expressed in metastatic tumors vs. non-metastatic tumors?
d. How many genes are significantly over-expressed in metastatic tumors vs. non-metastatic tumors?

These days, RNA-Seq and single cell RNA-seq has mostly replaced mRNA profiling using microarrays. However, once the RNA-Seq data is normalized and expression level computed using standard tools (see e.g. http://bioconductor.org/packages/release/bioc/html/edgeR.html), the above workflow is still applicable.

Submitting your assignment

When you're finished with the assignment, make a report of any questions you answered plus any requested output, and gather the scripts that you modified. Place them all in a folder, and create a single archive file called assignment6.tar.gz. Remember, you'll need to use a command like this:

```
tar –czf assignment6.tar.gz yourFolder
```

When you're finished, upload the `assignment6.tar.gz` file to the course website. You only need to upload your single `.tar.gz` file that contains all your answers and output for this assignment.

You're finished with Assignment 6!