**Biostatistics Midterm**

*The datasets in these projects are all in the DATA folder*

*Three data files from two datasets, we will do both the mid-term project and the final project based on these datasets. Extra points (10% of total) for any insights outside of the questions asked. All code submitted must contain*
*annotations throughout, so if the code were stripped away, it will tell "a story" of what the program does and how it is designed. Should not be big essays, short comments, interspersed in the code) and the answers should be separately presented in a document.*

**Dataset 1: train.csv.gz**
This is hand-written digits (0-9) from many people.
It contains 785 columns, first columns is digit and the 784 remaining columns are pixels 0-783. These are essentially 28x28 squares, so you can map the 784 columns to entries in the squares. Each entry is 0-255, 0 is for black and 255 stands for white, numbers in between are shades of gray

**1) Read in the data and convert the pixel data into pictures using plot show a few examples in your report along with the code**

- *See r markdown document for the code*

**2) Separate the data by digits, and calculate average value of each pixel,**

**a) Plot the average values, does it still resemble the digit label ?**

- Yes, when all the averages are plotted they resemble the digit label. 1 is the only exception where it is a bit distorted and more like a slanted line instead of a straight vertical line.

**b) which digits fare the best under this operation ?**

- Visually, it appears that the digit zero maintains its appearance the best when averaged..

**3) Find which columns have the most variance and which have the least.**

**a) Over the whole dataset and then separately, for each digit (0-9)**

- *See r markdown document for the code*

**b) Can you connect the variance to the results in 2b ?**

- We selected 0 as the digit that looks the best when each pixel is averaged. However, the digit 1 is the digit with the lowest average variance. Comparatively, the visual representation of the average 1 is fairly blurry.

**c) Does replacing the columns with the lowest variability by their average value have an effect on the digits?**

- Replacing the lowest variability pixels with their average value has a minimal impact on the visualization, because columns with low variability had values all close to the average anyway, so the change is fairly negligible.

**d) How many columns have average values close to 255 or 0 and why ?**

- Columns with averages close to 0 tend to be near the edges of the image and are white because there is usually nothing drawn in that space. Columns with average close to 255 are closer to the center, and where the average digit almost always has some part of the digit included in that pixel.

**4) Write the digits (0-9) in these squares and "digitize" them, essentially add lines corresponding to your own handwriting to this set. You should present a program that prints out digits in your handwriting.**

- *See r markdown document for the code*

**Dataset 2: a) Mnemiopsis_col_data.csv b) Mnemiopsis_count_data.csv**

*This is gene expression data, The columns represent samples, whose information is in the col_data file. The count_data file contains counts for each gene (rows).*
*The file, info_gene.txt contains information about the organism and some links to look up gene functions. It will be a good experience to learn to use the genome resources, as this is the kind of struggles most researchers go through when they start looking at genes.*

**1) What are the top 5 genes with the highest average expression (across experiments) in the set? What is their function ?**

- The top 5 genes with the highest average expression across experiments are: ML20395a, ML26358a, ML46651a, ML020045a, and ML00017a.
- Their functions are:
- ML20395a: Elongation factor 1-alpha (translation)
- ML26358a: Actin (major protein constituent of cytoskeleton-->microfilaments, and for thin filaments in muscle fibrils)
- ML46651a: Membrane attack complex? (according to Argot2: no other results)
- ML020045a: Tubulin beta chain (second protein component of microtubule)
- ML00017a: Elongation factor 2 (translation)

**2) Are the top 5 genes different if they are done on a per column basis ?**

ORIGINAL TOP 5 GENES ARE: ML20395a, ML26358a, ML46651a, ML020045a, and ML00017a

(S) = same; (D) = different
- When sorted on a per-column basis, the top 5 genes differ as follows:
- aboral1: ML46651a(S), ML20395a(S), ML020045a(S), ML174731a(D),ML26358a(S)
- aboral2: ML20395a(S),ML46651a(S),ML26358a(S),ML01482a(D),ML034334a(D)
- aboral3: ML20395a(S),ML01482a(D),ML26358a(S),ML46651a(S),ML034334a(D)
- aboral4: ML01482a(D),ML20395a(S),ML034334a(D),ML46651a(S),ML034336a(D)
- oral1: ML20395a(S),ML020045a(S),ML04011a(D),ML26358a(S),ML00017a(S)
- oral2: ML20395a(S),ML020045a(S),ML04011a(D),ML00017a(S),ML26358a(S)
- oral3: ML20395a(S),ML004510a(D),ML26358a(S),ML00017a(S),ML04011a(D)
- oral4: ML20395a(S),ML004510a(D),ML46651a(S),ML020045a(S),ML00017a(S)

-- Yes, the top 5 genes vary depending if it is done on a per-column basis. Many of the original top 5 genes reappear in these newly generated "top 5" gene sets, but each column has 1-3 different genes in its "top 5" listing.

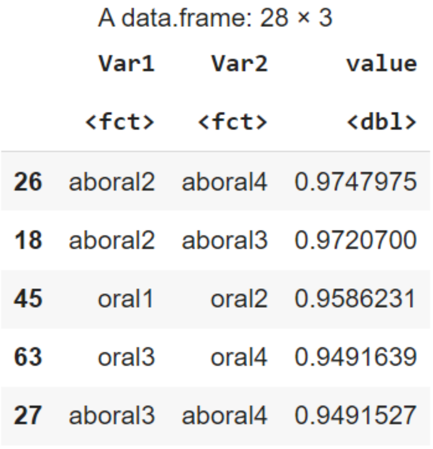
**3) Calculate mean and standard deviation of each column. If the mean is different, then scale the columns so that they all have the same mean for the subsequent questions**

- *See r markdown document for the code*

- Aboral1 mean: 524.097897026831
- Aboral2 mean: 580.521936185642
- Aboral3 mean: 581.273567802756
- Aboral4 mean: 560.089678510998
- Oral1 mean: 551.640258641528
- Oral2 mean: 428.993352671018
- Oral3 mean: 419.606719845299
- Oral4 mean: 457.431713802272
- Aboral1 std. dev.: 2281.93650477954
- Aboral2 std. dev.: 2665.17920400535
- Aboral3 std. dev.: 2451.04047674849
- Aboral4 std. dev.: 2687.42881888043
- Oral1 std. dev.: 2362.58414475044
- Oral2 std. dev.: 1631.39235737646
- Oral3 std. dev.: 1726.88911041491
- Oral4 std. dev.: 1912.52250585408

**3) Use correlations between columns to find the samples that are closely related. Is this concordant with the column labels ?**

- Top 5 correlated columns.



A data.frame: 28 × 3

| | Var1<br>&lt;fct&gt; | Var2<br>&lt;fct&gt; | value<br>&lt;dbl&gt; |
|---|---|---|---|
| **26** | aboral2 | aboral4 | 0.9747975 |
| **18** | aboral2 | aboral3 | 0.9720700 |
| **45** | oral1 | oral2 | 0.9586231 |
| **63** | oral3 | oral4 | 0.9491639 |
| **27** | aboral3 | aboral4 | 0.9491527 |

- For correlation values above 0.9, these samples that are closely correlated with each other are concordant with the column labels. However, we also do see high aboral v. oral correlation values at 0.85 and below.

**4) Use correlations between rows to find the closest pairs (top 5)    Are these close because they vary a lot between the groups or are they close because they don't vary much ?**

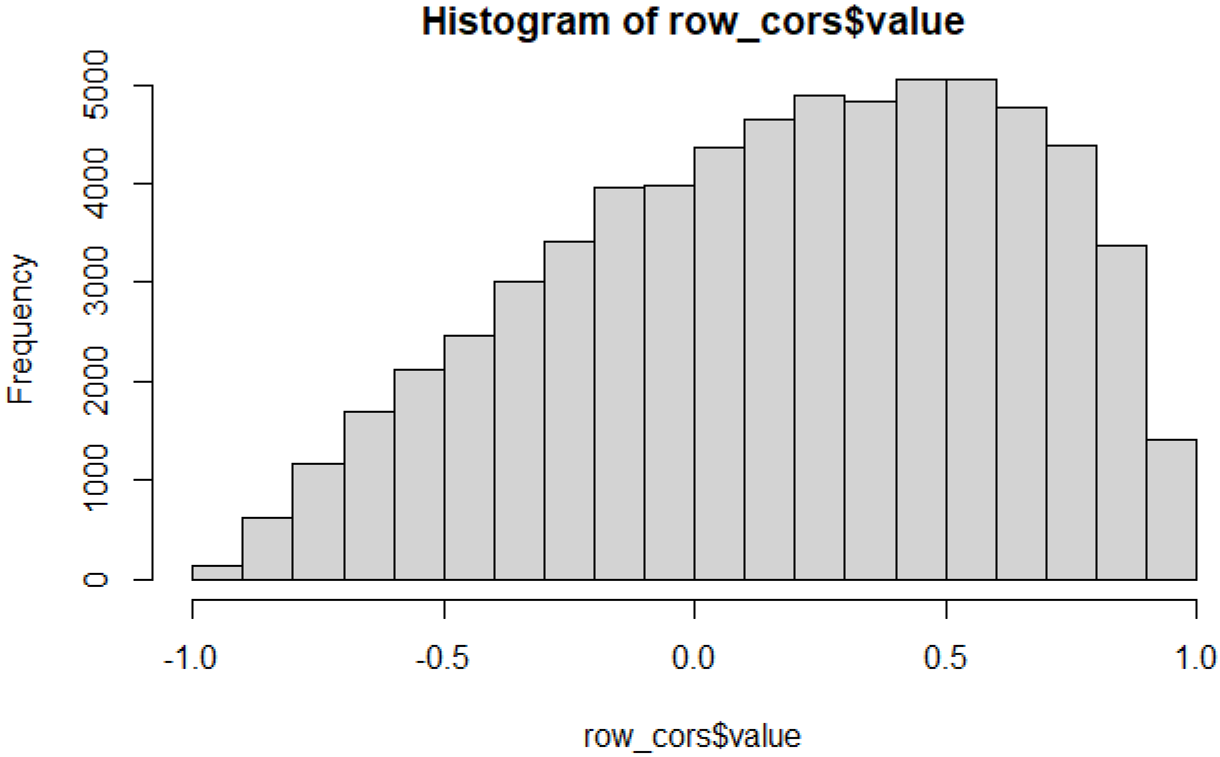
The problem with computing row-wise correlations for every gene is that the output would be 16548^2 calculations. To reduce this, we attempted to only calculate correlations for rows which would have a high correlation. To estimate the correlation strength, we used a method which first calculated a PCA value for each row.

First, we calculated the PCA value for each row, forcing the number of principal components to 1. This should provide an approximation of similarity, so that rows likely to have high correlations have similar PCA values. By sorting by this value we place rows likely to be highly correlated near each other in the data frame. After this, we test the correlation of each row versus the five following rows and record the correlation. This sorting after dimensionality reduction is performed to reduce the computational load for calculating n^2 correlations. The method is not perfect, and may omit some highly correlated pairs, but should provide decent coverage of highly correlated genes.

The top 5 correlations:

| Index | Gene1 | Gene2 | Correlation |
|---|---|---|---|
| 64464 | ML45843a | ML073030a | 0.9993858 |
| 64913 | ML00365a | ML193210a | 0.9985403 |
| 65316 | ML034336a | ML034334a | 0.9981401 |
| 65311 | ML034337a | ML034336a | 0.9969169 |
| 65204 | ML148538a | ML148534a | 0.9960324 |

The histogram below shows the distribution of correlations, which does skew towards higher correlations, though maybe not as strongly as we would have liked. Increasing the number of PC parameters may improve this model.

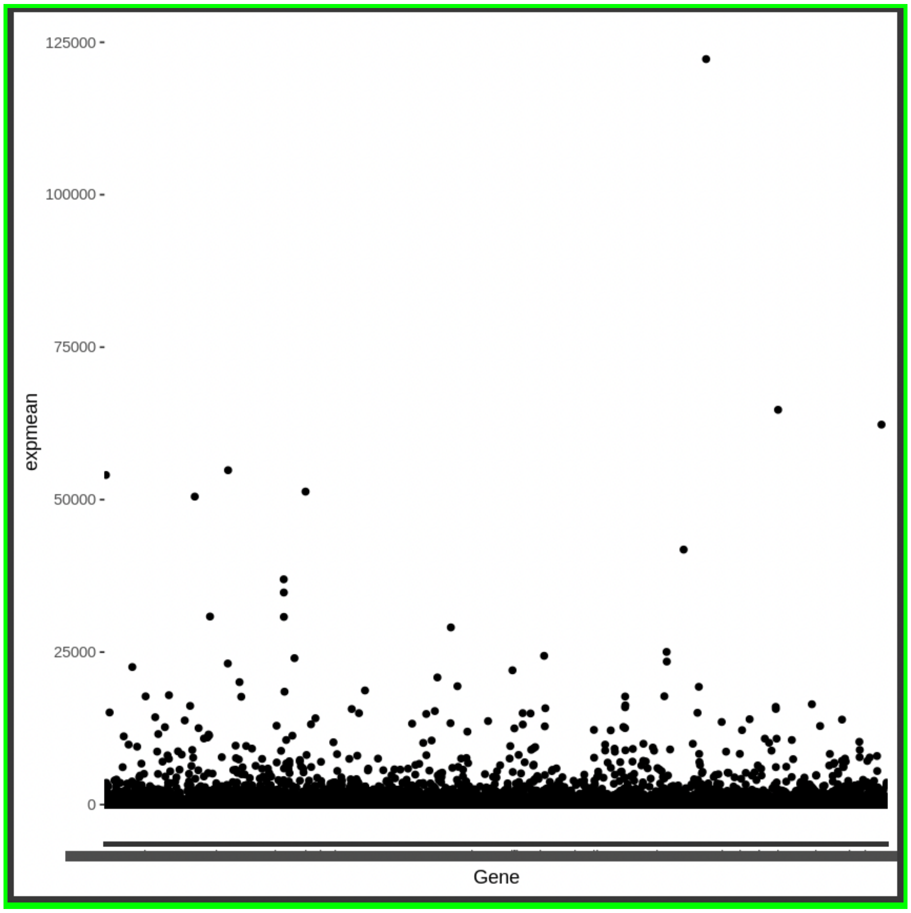


These top 5 correlated genes are close because they do NOT vary much between groups in expression level. For example, the expression pattern of the highest correlated pair is shown below.

| Gene | aboral1 | aboral2 | aboral3 | aboral4 | oral1 | oral2 | oral3 | oral4 |
|---|---|---|---|---|---|---|---|---|
| ML073030a | 4256 | 10042 | 9666 | 12496 | 1069 | 1335 | 3394 | 2379 |
| ML45843a | 4182 | 10115 | 9354 | 12679 | 799 | 1155 | 3322 | 2453 |

**5) If you were forced to divide the genes in each column into high, medium and low count genes, how would you do this based on the data that you have? Make a graph to see what the distribution looks like. if it is fairly evenly distributed, you can just divide into 3 equal parts. if not, then you would have to skew the division towards the side with more members.**

Most of the data is between an expmean of 0 and 25000. Therefore, to divide the data into three groups, you must decide on a cutoff for low, medium and high. the genes cannot be equally divided into three groups.



**6) make a list of the top 5 genes with most variability and top 5 genes with least variability (exclude genes that have low expression values)**

Perform differential expression analysis before filtering by variance so normalized counts are used.
Must do between-sample normalization, which is needed to account for technical effects, (differences not because of the biological conditions of interest), that prevent read count data from accurately reflecting differences in expression.
For this, DESeq differential expression was used.

Highest variability genes from greatest to least:
ML20395a
ML26358a
ML46651a
ML020045a
ML00017a

Lowest variability genes from least to greatest:
ML32095a
ML29351a
ML16594a

ML11345a
ML25222a

**7) Using the labels of columns provided, find the top variable genes between the two groups using a t-test list the 5 most up regulated and 5 most down regulated genes. What happens if you rank by p-value of the t-test ? would you exclude some of these genes for having low expression ?**

Check for the most positive and negative non-zero values. The most positive are the most upregulated, and the most negative are the most downregulated (for a compared to b). Using simply the log method, the genes we are interested in are as follows. (Excluding positive and negative infinity log values).
format: gene name(log of aboral vs oral ratio)

- Most upregulated aboral vs oral: ML327424a(6.169369), ML343422a(5.351331), ML14971a(5.258369), ML27982a(4.941642), and ML311627a(4.862107)

- Most downregulated aboral vs oral: ML34341a(-9.785023), ML090812a(-9.394743), ML087114a(-8.896168), ML034332a(-8.767921), and ML319815a(-8.266678)

—-----------------------------------------------------------------------------------------------------

We can also rank by p-value of the t-test, which will tell us which genes have the most highly differential gene expression between the mean aboral vs oral values.

The top 10 genes with the lowest t-test p-values are shown below:

ML050913a, ML263524a, ML01833a, ML329912a, ML070258a, ML005114a, ML204423a, ML282521a, ML15096a, ML102911a

- Based on the values seen for these selected genes, I woud not exclude some of these genes for having low expression.