# Lab7

Cluster Analysis

*Jessica Temporal 7547611*

*October 6, 2016*

## Contents

**1. Load the fibroEset library and data set (`library(fibroEset)`). Obtain the classifications for the samples.**

```
source("http://bioconductor.org/biocLite.R")
```

```
## Bioconductor version 3.2 (BiocInstaller 1.20.3), ?biocLite for help
```

```
## A new version of Bioconductor is available after installing the most
##   recent version of R; see http://bioconductor.org/install
```

```
biocLite("fibroEset")
```

```
## BioC_mirror: https://bioconductor.org
```

```
## Using Bioconductor 3.2 (BiocInstaller 1.20.3), R 3.2.2 (2015-08-14).
```

```
## Installing package(s) 'fibroEset'
```

```
##
## The downloaded source packages are in
##  '/tmp/RtmpXvMORv/downloaded_packages'
```

```
## Old packages: 'boot', 'class', 'cluster', 'codetools', 'foreign',
##   'lattice', 'MASS', 'Matrix', 'mgcv', 'nlme', 'nnet', 'spatial',
##   'survival'
```

```
library(fibroEset)
```

```
## Loading required package: Biobase
```

```
## Loading required package: BiocGenerics
```

```
## Loading required package: parallel
```

```
##
## Attaching package: 'BiocGenerics'
```

```
## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB
```

```
## The following objects are masked from 'package:stats':
##
##     IQR, mad, xtabs
```

```
## The following objects are masked from 'package:base':
##
##     anyDuplicated, append, as.data.frame, as.vector, cbind,
##     colnames, do.call, duplicated, eval, evalq, Filter, Find, get,
##     grep, grepl, intersect, is.unsorted, lapply, lengths, Map,
##     mapply, match, mget, order, paste, pmax, pmax.int, pmin,
##     pmin.int, Position, rank, rbind, Reduce, rownames, sapply,
##     setdiff, sort, table, tapply, union, unique, unlist, unsplit

## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
```

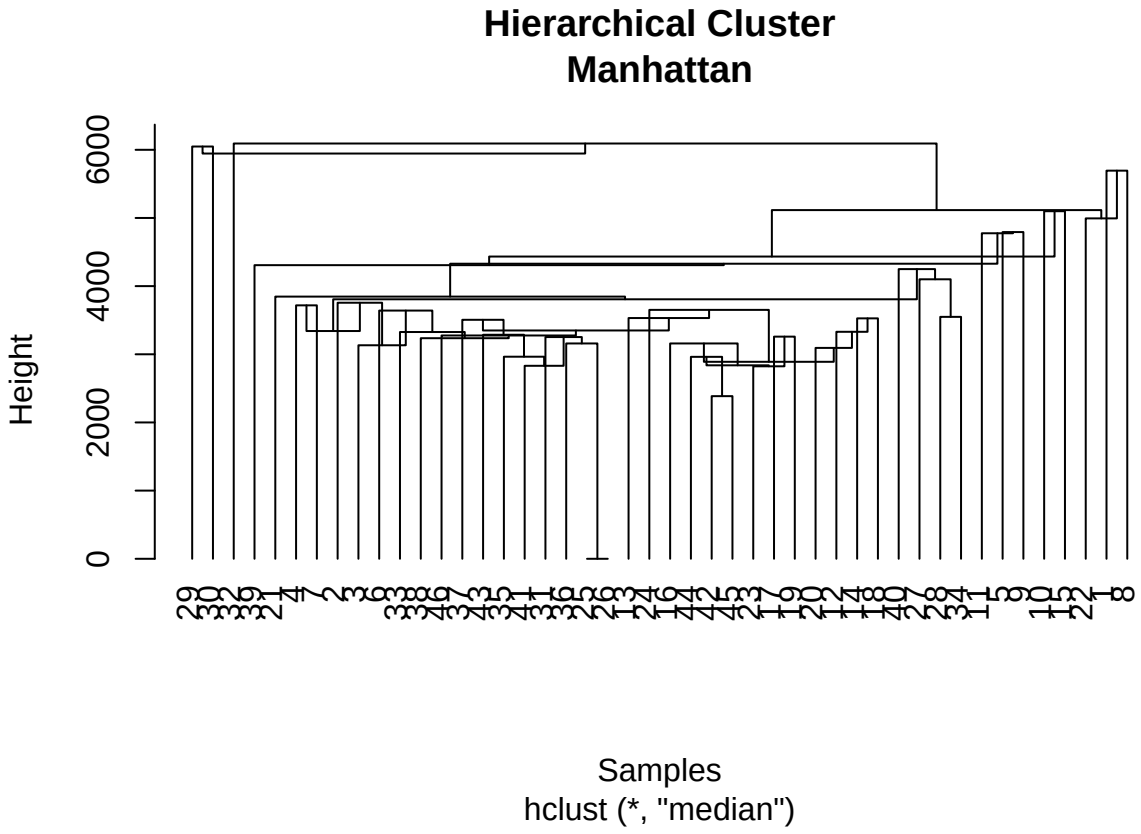
```
data("fibroEset")
fib <- exprs(fibroEset)
```

**2. Select a random set of 50 genes from the data frame, and subset the data frame.**

```
fib.genes <- rownames(fib)
sample.genes <- sample(fib.genes, 50)
fib.sub <- fib[sample.genes, ]
```

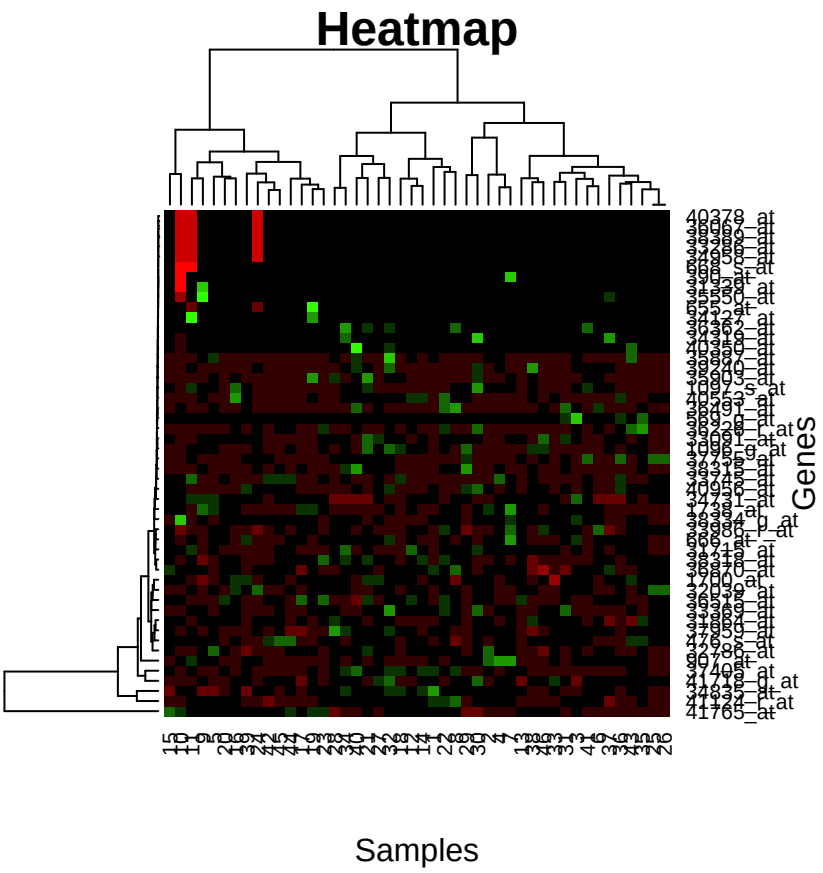
**3. Run and plot hierarchical clustering of the samples using manhattan distance metric and median linkage method. Make sure that the sample classification labels are along the x-axis. Title the plot.**

```
fib.man.samples <- dist(t(fib.sub), method = "manhattan")
fib.hclust.samples <- hclust(fib.man.samples, method = "median")
plot(fib.hclust.samples,
     main = "Hierarchical Cluster\nManhattan",
     xlab = "Samples",
     hang = -1)
```



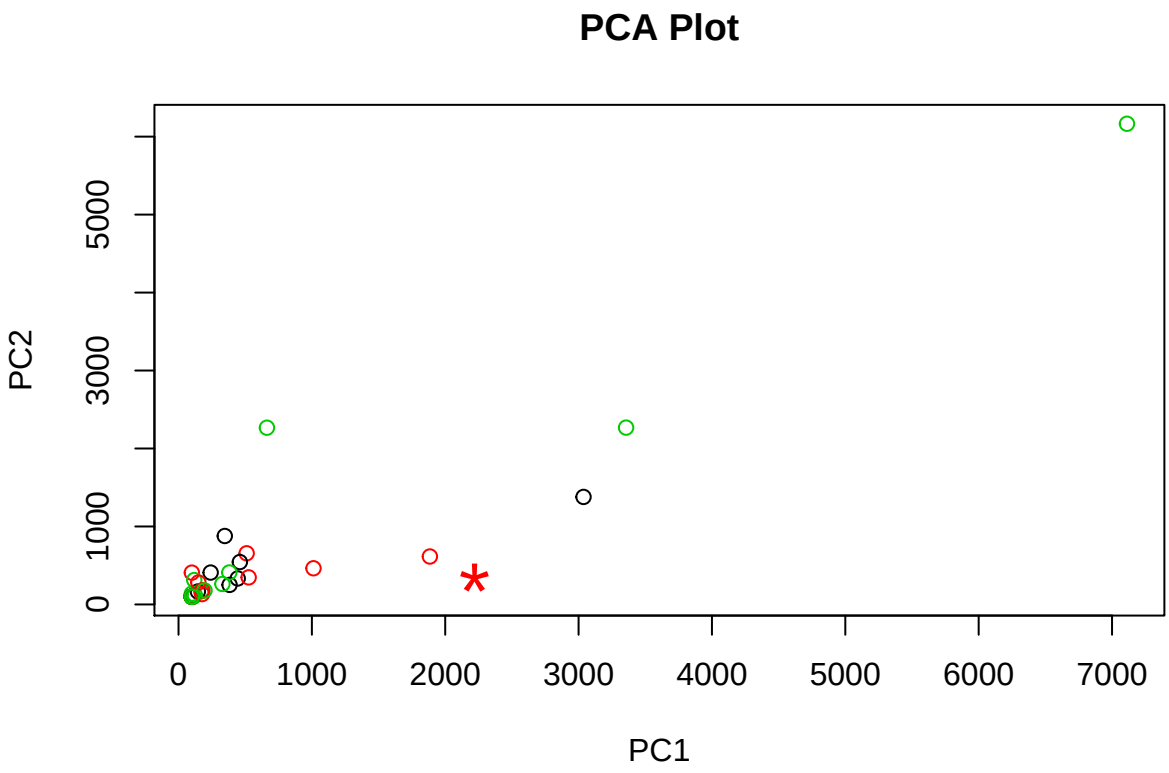
**4. Now both run hierarchical clustering and plot the results in two dimensions (on samples and genes). Plot a heatmap with the genes on the y-axis and samples on the x-axis. Once again, make sure that the sample and genes labels are present. Title the plot.**

```
hm.col <- c("#FF0000", "#CC0000", "#990000", "#660000", "#330000", "#000000",
            "#000000", "#0A3300", "#146600", "#1F9900", "#29CC00", "#33FF00")
heatmap(fib.sub, main = "Heatmap", xlab = "Samples", ylab = "Genes", col = hm.col)
```



**5. Calculate PCA on the samples and retain the first two components vectors (eigenfunctions). Calculate k-means clustering on these first two components with k=3.**

```
fib.pca <- prcomp(x = t(fib.sub))
fib.kmeans <- kmeans(fib.pca$x[,1:2], centers = 3)
plot(fib.sub, col = fib.kmeans$cluster, cex=1, xlab = "PC1", ylab = "PC2",
     main = "PCA Plot")
points(fib.kmeans$centers, col = 1:4, pch = "*", cex = 2.5)
```



**6. Plot a two-dimensional scatter plot of the sample classification labels, embedded with the first two eigenfunctions (from PCA). Color the labels with the color that corresponds to the predicted cluster membership. Make sure to label the axes and title the plot. Color based on kmeans cluster. Put the different species and identify them and then color them based on kmeans cluster to see which species didn't cluster correctly.**

```
fib.species <- fibroEset$species
plot(fib.sub, col = fib.kmeans$cluster, cex = 1, xlab = "PC1", ylab = "PC2",
     main = "PCA Plot")
text(fib.sub, labels = fib.species, cex = 1, pos = 1)
```