# Assignment 7 (quizz number 7-10)

## Iyarace Khampakdee

```
library(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, sd, var, xtabs
```

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

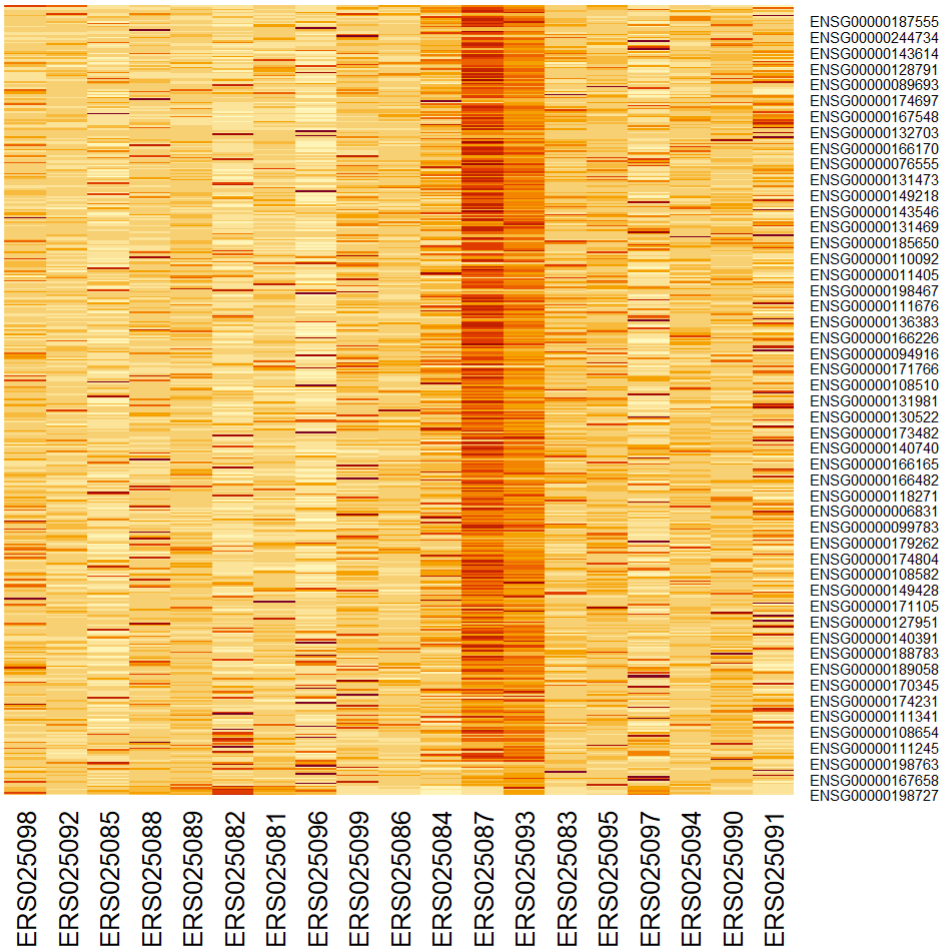
```
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
```

**7. Which of the following code chunks will make a heatmap of the 500 most highly expressed genes (as defined by total count), without re-ordering due to clustering? Are the highly expressed samples next to each other in sample order?**

```
con =url("http://bowtie-bio.sourceforge.net/recount/ExpressionSets/bodymap_eset.RData")
load(file=con)
close(con)
bm = bodymap.eset
edata = exprs(bm)
```

**Answer**

```
row_sums = rowSums(edata)
edata = edata[order(-row_sums),]
index = 1:500
heatmap(edata[index,],Rowv=NA,Colv=NA)
```

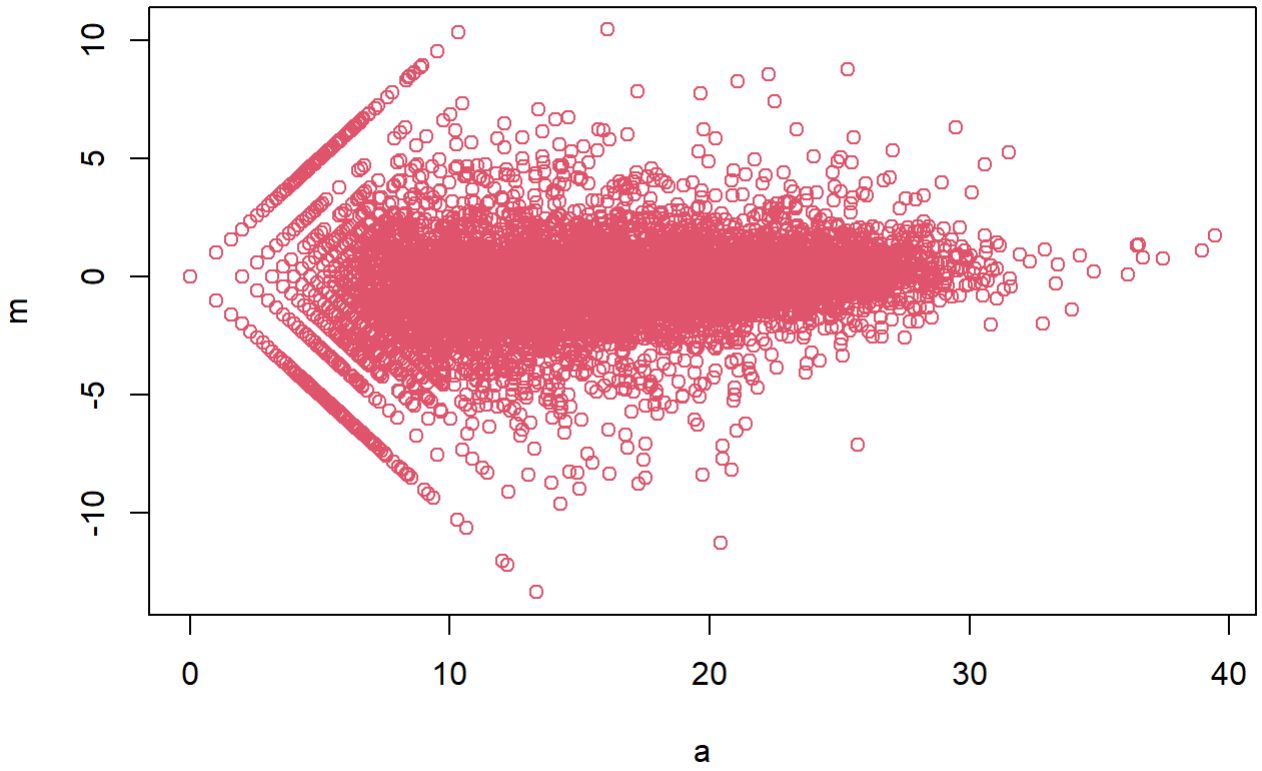



**8. Make an MA-plot of the first sample versus the second sample using the log2 transform (hint: you may have to add 1 first) and the rlog transform from the DESeq2 package. How are the two MA-plots different? Which kind of genes appear most different in each plot?**

```
con =url("http://bowtie-bio.sourceforge.net/recount/ExpressionSets/bodymap_eset.RData")
load(file=con)
close(con)
bm = bodymap.eset
pdata = pData(bm)
edata = exprs(bm)
```

**Answer**

```
m = log2(edata[,1]+1) - log2(edata[,2]+1)
a = log2(edata[,1]+1) + log2(edata[,2]+1)
plot(a,m,col=2)
```




The plots look pretty similar, but the rlog transform seems to shrink the low abundance genes more. In both cases, the genes in the middle of the expression distribution show the biggest differences.

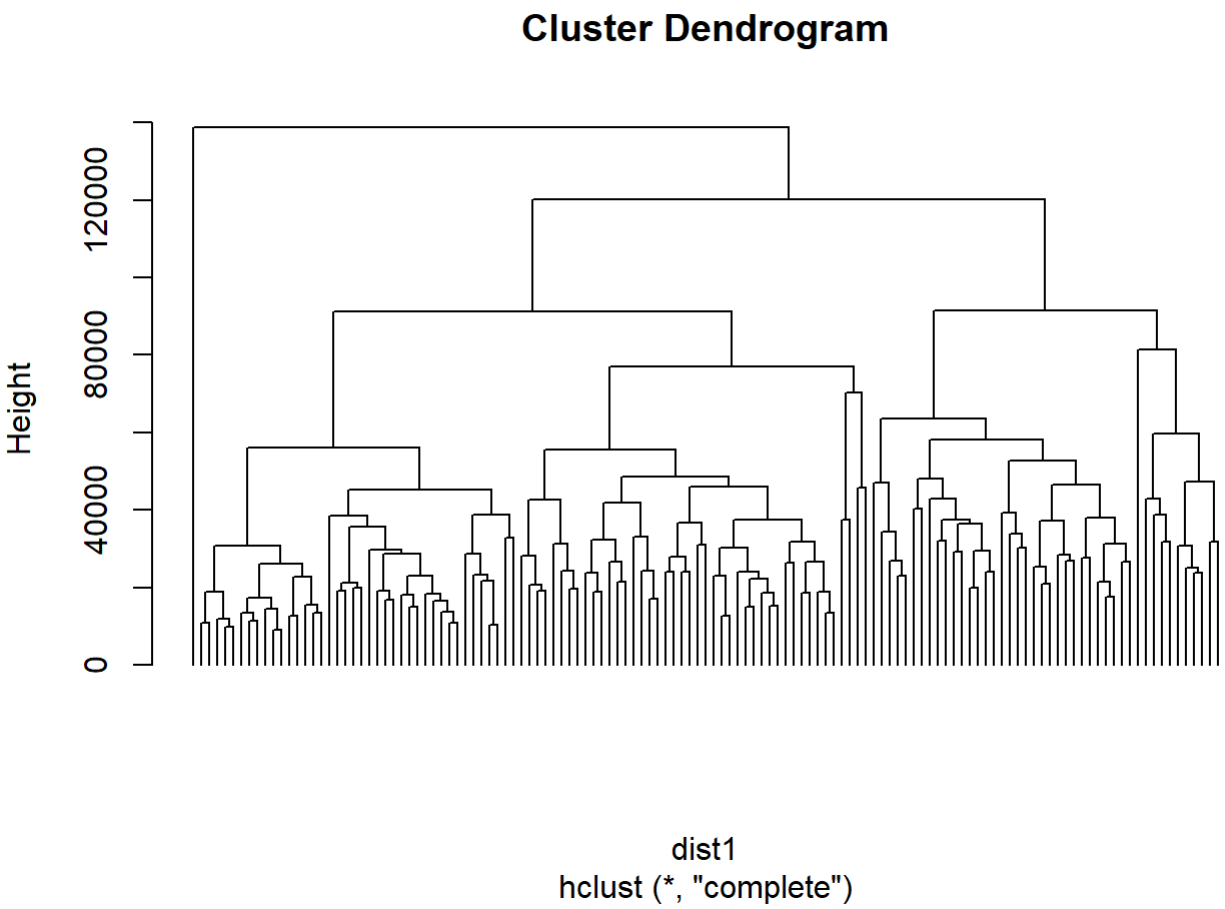
**9. Cluster the data in three ways: 1. With no changes to the data 2. After filtering all genes with rowMeans less than 100 3. After taking the log2 transform of the data without filtering Color the samples by which study they came from (Hint: consider using the function myplclust.R in the package rafalib available from CRAN and looking at the argument lab.col.) How do the methods compare in terms of how well they cluster the data by study? Why do you think that is?**

```
con =url("http://bowtie-bio.sourceforge.net/recount/ExpressionSets/montpick_eset.RData")
load(file=con)
close(con)
mp = montpick.eset
pdata=pData(mp)
edata=as.data.frame(exprs(mp))
fdata = fData(mp)
```

**Answer**

1. With no changes to the data

```
dist1 = dist(t(edata))
hclust1 = hclust(dist1)
plot(hclust1, hang = -1, labels=FALSE)
```

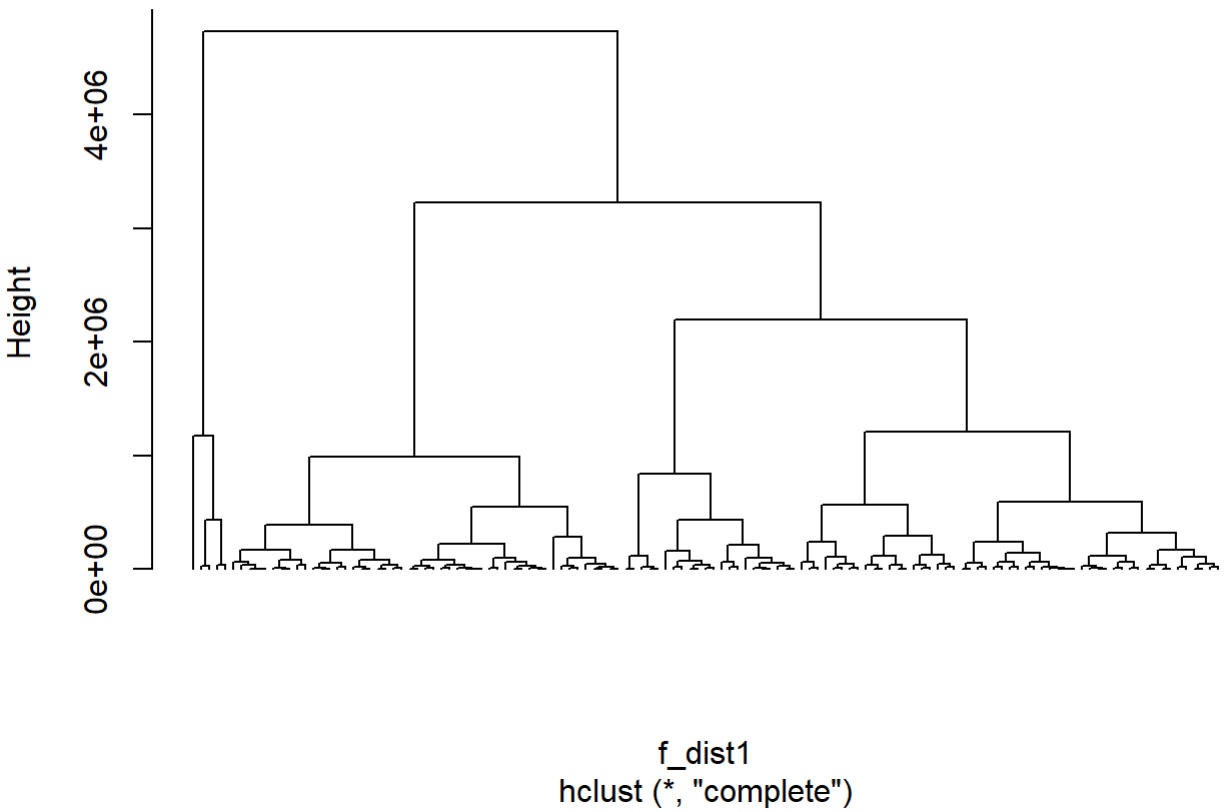



2. After filtering all genes with rowMeans less than 100

```
low_genes = rowMeans(edata) < 100
table(low_genes)
```

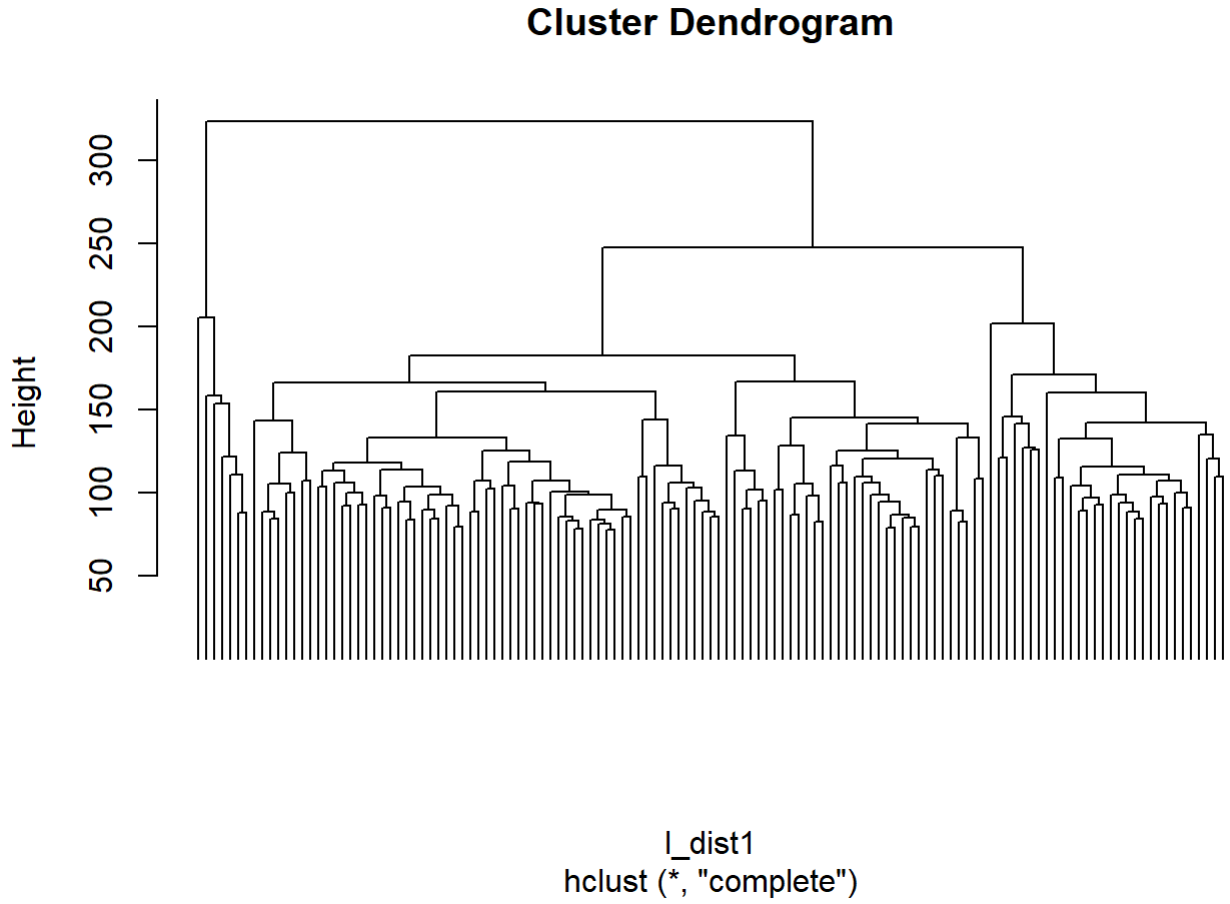
```
## low_genes
## FALSE  TRUE
##  3072 49508
```

```
filtered_edata = filter(edata, !low_genes)
f_dist1 = dist(t(filtered_edata))
f_hclust1 = hclust(f_dist1)
plot(f_hclust1, hang = -1, labels=FALSE)
```




3. After taking the log2 transform of the data without filtering Color the samples by which study they came from

```
log_edata = log2(edata + 1)
l_dist1 = dist(t(log_edata))
l_hclust1 = hclust(l_dist1)
plot(l_hclust1, hang = -1, labels=FALSE)
```



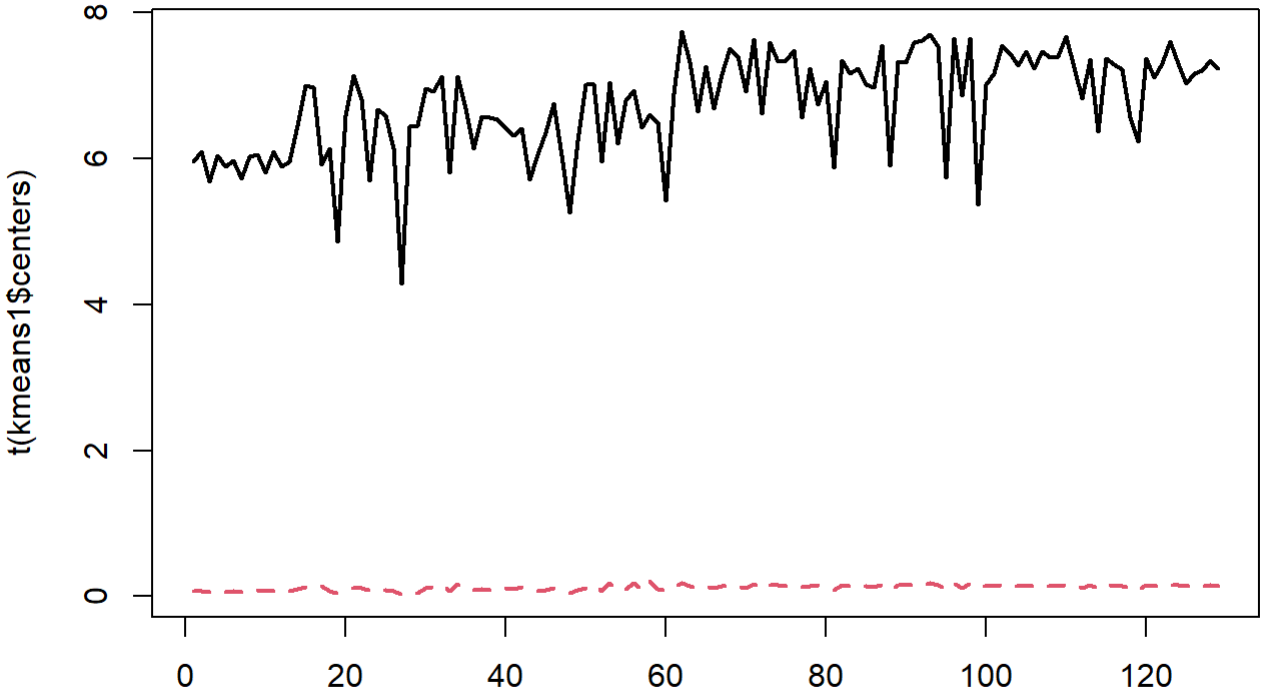

Clustering with or without filtering is about the same. Clustering after the log2 transform shows better clustering with respect to the study variable. The likely reason is that the highly skewed distribution doesn't match the Euclidean distance metric being used in the clustering example.

**10. Cluster the samples using k-means clustering after applying the log2 transform (be sure to add 1). Set a seed for reproducible results (use set.seed(1235)). If you choose two clusters, do you get the same two clusters as you get if you use the cutree function to cluster the samples into two groups? Which cluster matches most closely to the study labels?**

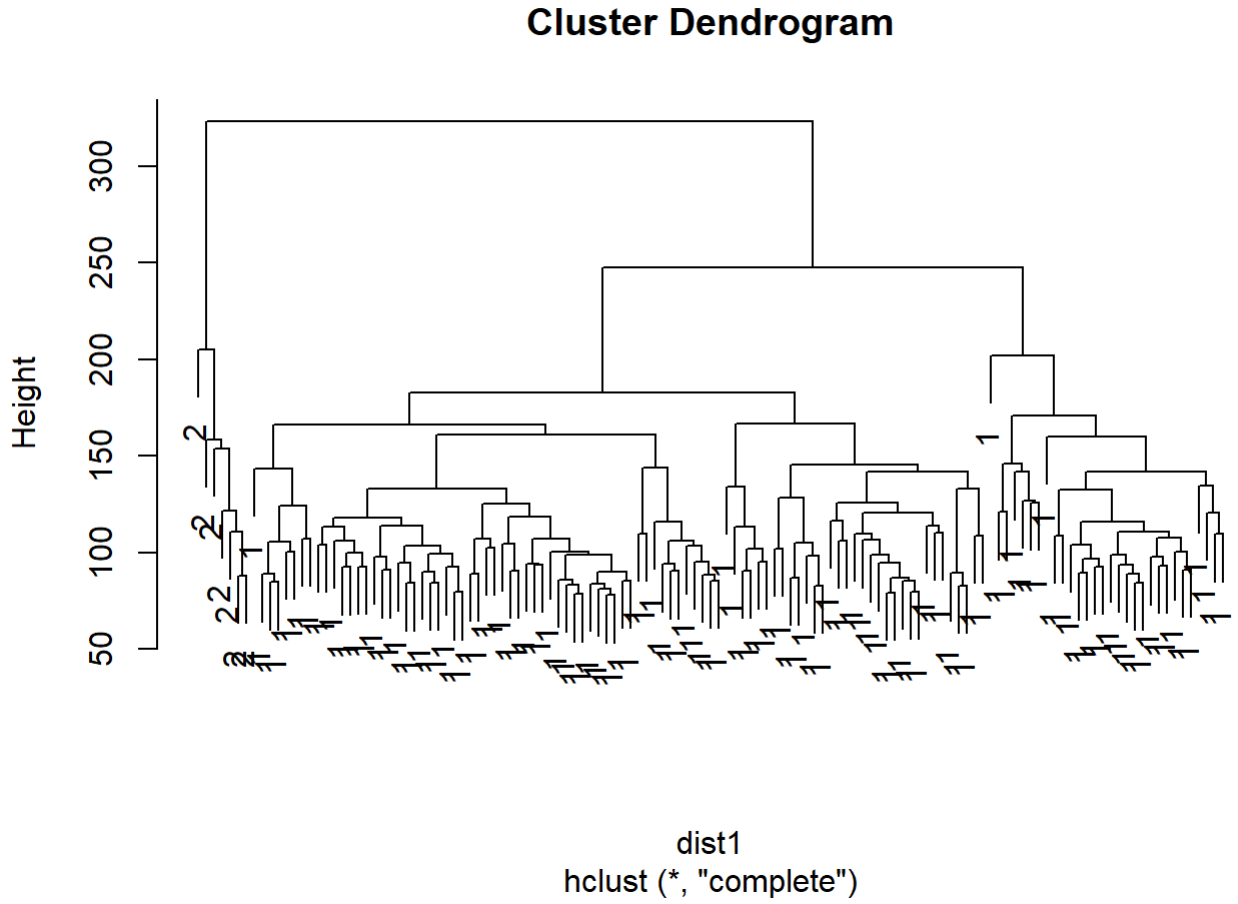
```
con =url("http://bowtie-bio.sourceforge.net/recount/ExpressionSets/montpick_eset.RData")
load(file=con)
close(con)
mp = montpick.eset
pdata=pData(mp)
edata=as.data.frame(exprs(mp))
fdata = fData(mp)
```

**Answer**

```
edata = log2(edata + 1)
set.seed(1235)
kmeans1 = kmeans(edata, centers = 2)
matplot(t(kmeans1$centers), col = 1:2, type = "l", lwd = 2)
```




```
dist1 = dist(t(edata))
hclust1 = hclust(dist1)
tree = cutree(hclust1, 2)
plot(hclust1, tree)
```




They produce different answers, with hierarchical clustering giving a much more unbalanced clustering. The k-means clustering matches study better.