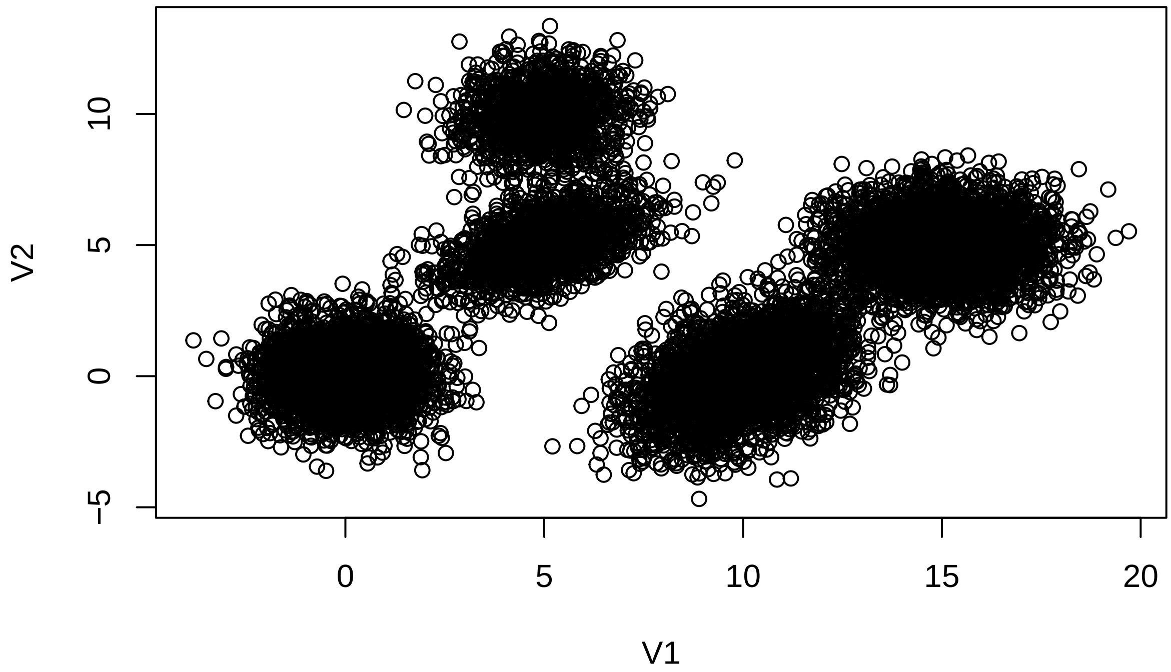
Genomics Homework 1

2023-02-04

First let's load in the data and look at the plot:

```
#loading in cluster data to predict, and actual cluster label
data.new <- as.matrix(read.table("data.txt",  header = FALSE));  #data we are interested in clustering
data.type <- read.table("true_clustering.txt",  header = FALSE)[[1]]; #correct label for each data poin
plot(data.new) #plotting the data
```



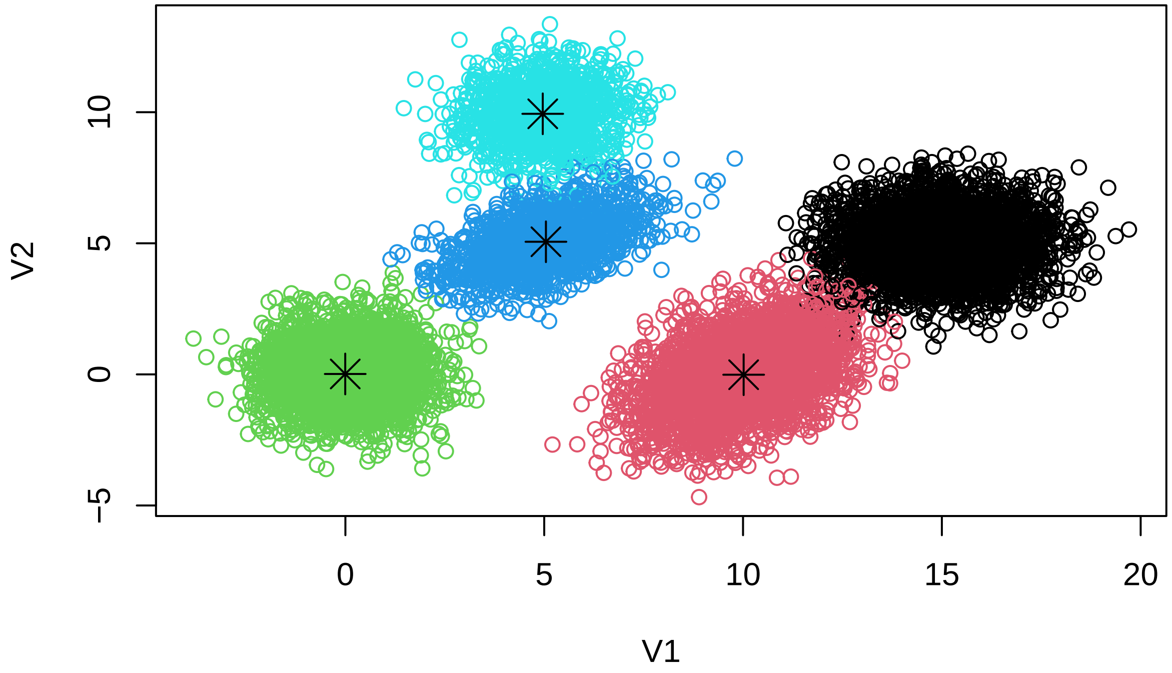
Here, we can see that there seems to be an obvious amount of 5 clusters here. Also, looking at "true_clustering.txt" data, we can also see it varies from 1-5 for the labels. We can confidently assume here, that we want to find five clusters. now below, we will use K-means, PAM, and Hierarchical clustering.

K-means Clustering:

```
# K-means with 5 clusters using the data
fit.kmeans <- kmeans(data.new, centers = 5)

cluster.kmeans <- fit.kmeans$cluster #collecting cluster information

table(cluster.kmeans, data.type) #comparing the cluster prediction, versus the actual label
```

```
##               data.type
## cluster.kmeans    1    2    3    4    5
##              1    0    0    0   25 3592
##              2    0    0    0 3037    0
##              3 3506   13    0    0    0
##              4    0 1334    0    0    0
##              5    0    1 1272    0    0
```

```
plot(data.new, col = fit.kmeans$cluster) #plotting same, data but colored with our prediction
points(fit.kmeans$centers, col = 1 , pch = 8, cex = 2) #center points chosen based on the algorithm
```

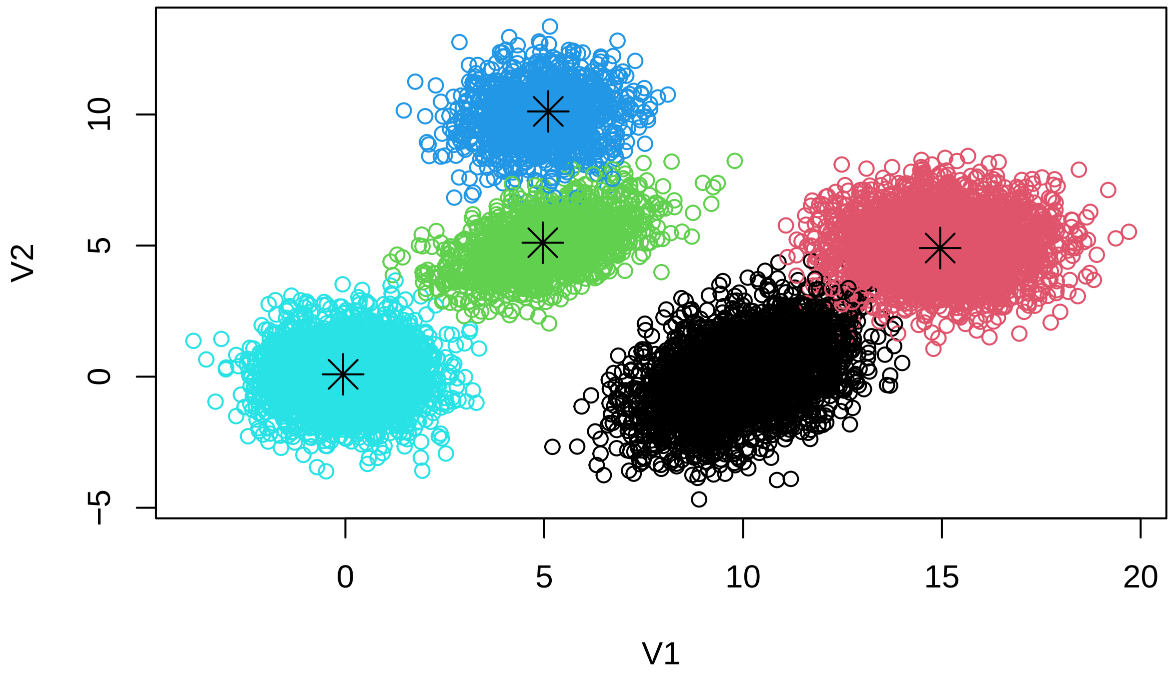


PAM Clustering:

```
library(cluster)

# Clustering by PAM with 5 clusters
fit.pam <- pam(data.new, k = 5, metric = "euclidean")

cluster.pam <- fit.pam$clustering #collecting cluster information

table(cluster.pam, data.type) #comparing the cluster prediction, versus the actual label
```

```
##            data.type
## cluster.pam    1    2    3    4    5
##           1    0    0    0 3038    0
##           2    0    0    0   24 3592
##           3    0 1334    0    0    0
##           4    0    2 1272    0    0
##           5 3506   12    0    0    0
```

```
plot(data.new, col = fit.pam$cluster) #plotting same, data but colored with our prediction
points(fit.pam$medoids, col = 1 , pch = 8, cex = 2) #center points chosen based on the algorithm
```

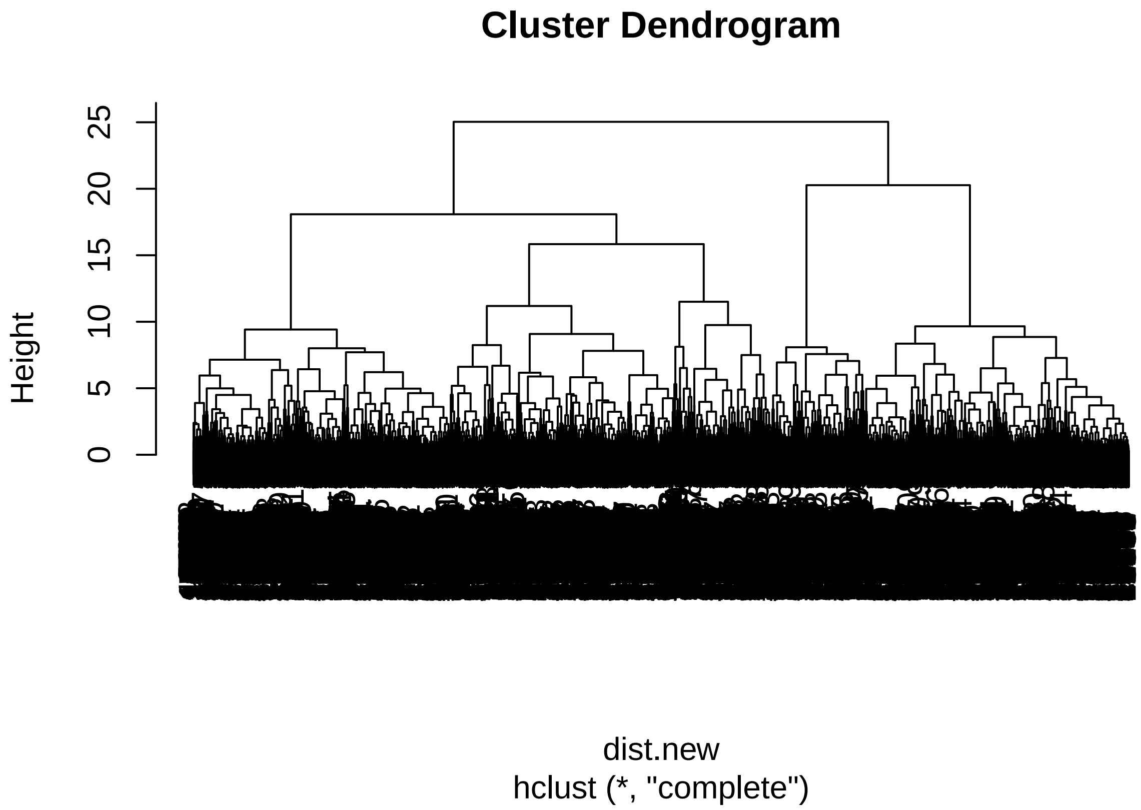


Hierarchical Clustering:

```
#Specifying distance function we will be using, which is euclidean, and collecting distances
dist.new <- dist(data.new, method = "euclidean")

# Apply hierachical clustering to this distance matrix
fit.hclust <- hclust(dist.new, method = "complete")

# Based on the clustering result, find two clusters
cluster.hclust <- cutree(fit.hclust, k = 5)

# Number of correctly and incorrectly clustered objects
table(cluster.hclust, data.type)
```

```
##               data.type
## cluster.hclust    1    2    3    4    5
##              1    0    0    0 3051    1
##              2    0    0    0   11 3591
##              3    1 1338    2    0    0
##              4    0    0 1270    0    0
##              5 3505   10    0    0    0
```

```
plot(fit.hclust) #plotting fitted hierarchical cluster dendogram
```



Error Rates for each respective algorithm:

```
library(mclust)
```

```
## Package 'mclust' version 6.0.0
## Type 'citation("mclust")' for citing this R package in publications.
```

```
classError(fit.kmeans$cluster, data.type) #error rat2e for k means clustering
```

```
## $misclassified
##  [1]   476   949  1352  1376  1631  2446  2456  2549  2684  2824  2901  3927
## [13]  4009  4089  4721  4787  5099  5142  5520  5549  6370  6393  6459  6759
## [25]  6904  7060  7268  7293  7404  7625  7810  9720 10020 10813 11009 11339
## [37] 12164 12233 12331
##
## $errorRate
## [1] 0.003051643
```

```
classError(fit.pam$clustering, data.type) #error rate for pam clustering
```

```
## $misclassified
##  [1]   476   949  1352  1376  1631  1828  2446  2456  2549  2684  2824  2901
## [13]  3927  4009  4089  4721  4787  5099  5520  5549  6370  6393  6459  6759
## [25]  6904  7060  7268  7293  7404  7625  7810  9720 10020 11009 11339 12164
## [37] 12233 12331
##
## $errorRate
## [1] 0.002973396
```

```
classError(cluster.hclust, data.type) #error rate for hierarchical clustering
```

```
## $misclassified
##  [1]   949  1192  2446  2824  2901  3927  4009  4089  4721  4787  6370  6440
## [13]  6459  6759  7268  7404  7625  7810  8858  9684  9720 10020 11024 11112
## [25] 12331
##
## $errorRate
## [1] 0.001956182
```

Summary

Finally, after looking at our graphs and error rates for each algorithm we can see that they all perform fairly well, and classify correctly over 99% of the time. If we explicitly look at error rate, the lowest error rate would be hierarchical clustering.

Looking at the plots too, K-means and PAM seem to classify correctly from an eye test point of view. The exact clusters we saw in the beginning, seemed to be correctly labeled in both clusters. The dendrogram that the hierarchical clustering produced also seems to have a clear 5 branches although that contain all the data points, although there are many small branches. In fact, K-means and PAM had near exact same results with some slight deviation in certain clusters.

Overall, all the results seem to be fairly close to eachother. Choosing one algorithm over another might depend on the context of the question.