HackBio Bio-Data Science Task 3

In this stage, I will continue to improve on my R programming skills. I have two tasks. First, I will perform K-means clustering and Hierachical clustering on 'mtcars built in data set in R' and secondly, the biological data set 'microbial_stationary_phase”, that I performed pca on last week.

Consider the following explanations and plots.

1. Data set for this first analysis is 'mtcars'

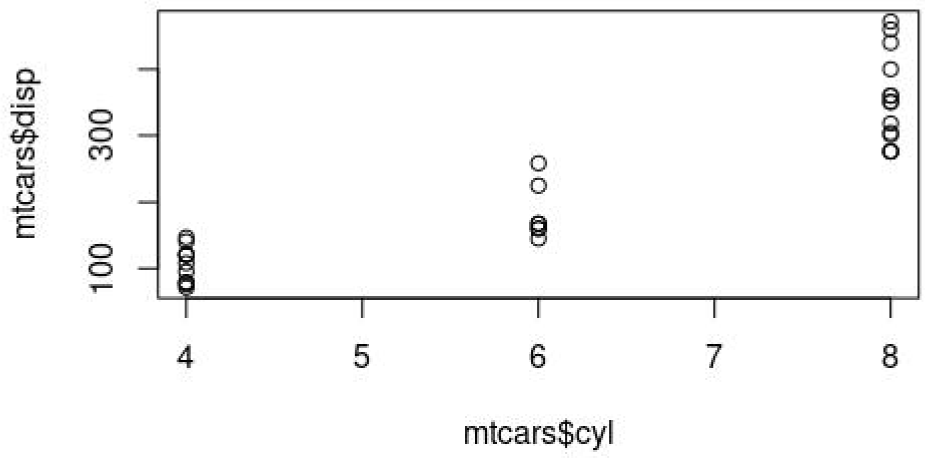
K-means clustering

2. Our data is mtcars, an inbuilt data in R

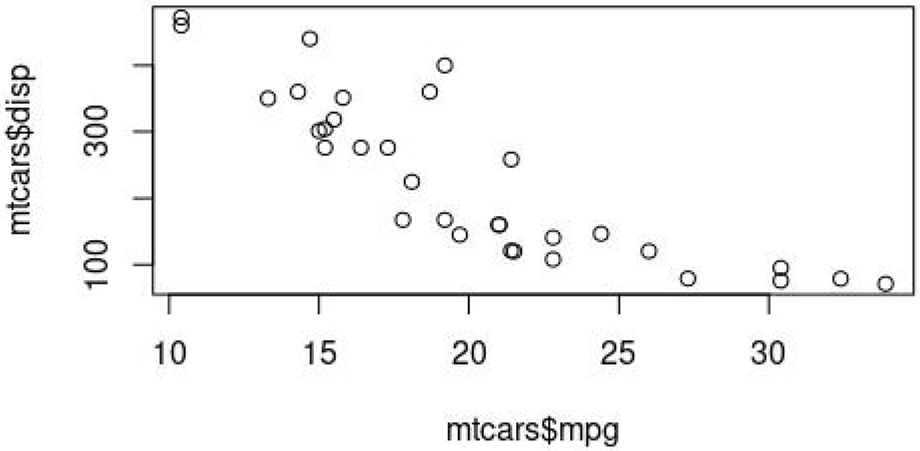
```
data()
data(mtcars)
View(mtcars)
```

3. For the purpose of visualization, I will like to use any two columns that correlate

```
cor(mtcars)
plot (mtcars$cyl, mtcars$disp)
```



```
plot (mtcars$mpg, mtcars$disp)
```



4. So we can have same results.I will set seed which is an arbitrary number. Since I want to get a consistent result from my K-Mean clustering (which is unsupervised). Picking '102' means that I have chosen to index to 102.

```
set.seed(102)
```

5. I will now perform my KMEAN Clustering for now. I want to pick 3 centers (clusters)

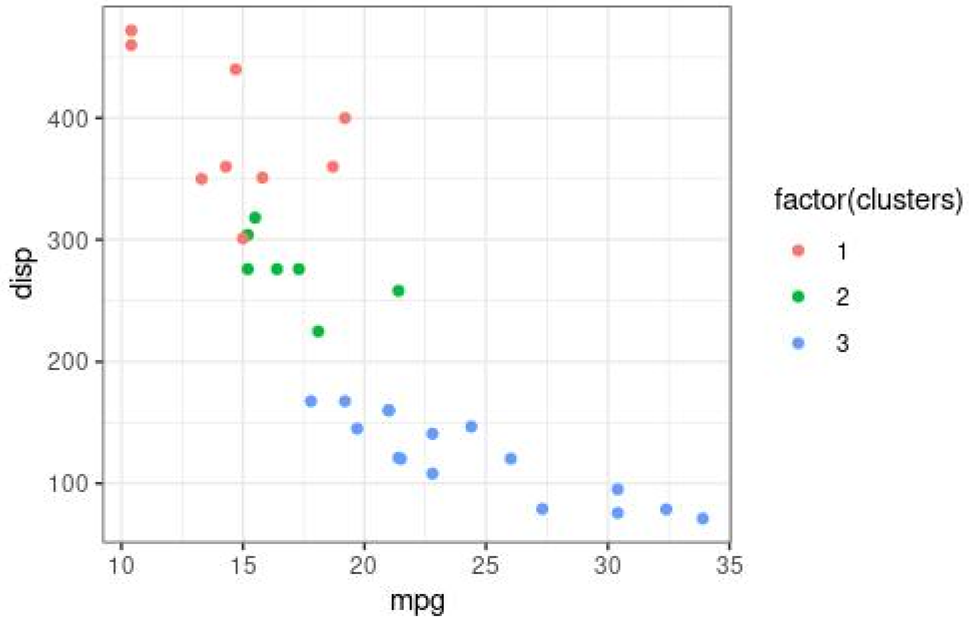
```
mtcarsK3 <- kmeans (x = mtcars, centers = 3)
```

6. Well, I can also add this cluster information to my dataset

```
mtcars$clusters <- c(mtcarsK3$cluster)
```

7. I want to visualize this cluster information for each car

```
ggplot(mtcars, aes(x = mpg, y = disp, color = factor(clusters))) + geom_point() +
theme_bw()
```



8. I need to install "useful" for my K-Means Clustering

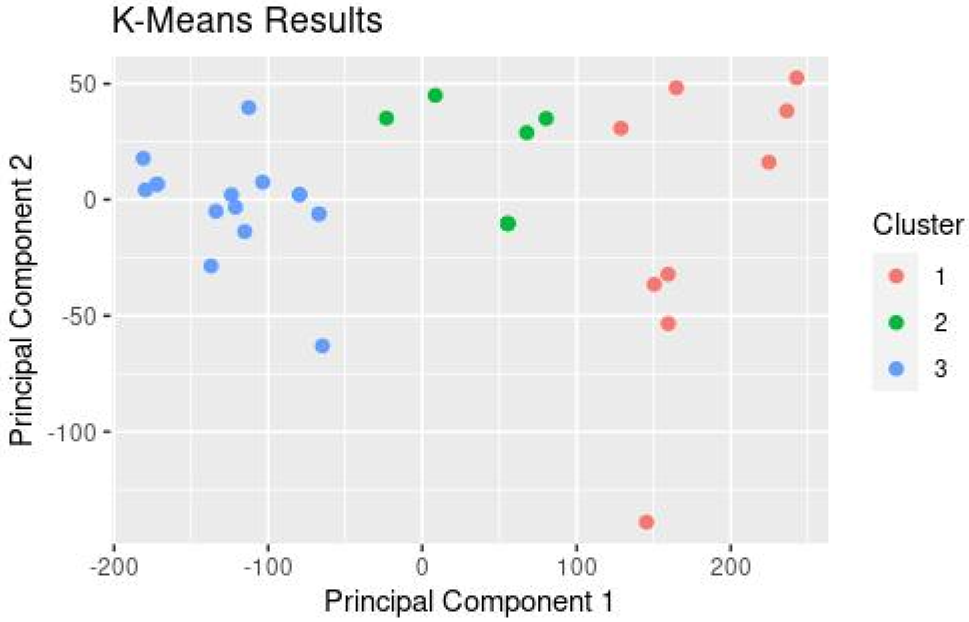
```
install.packages('useful')
library ('useful')
```

9. I will create a new column for country

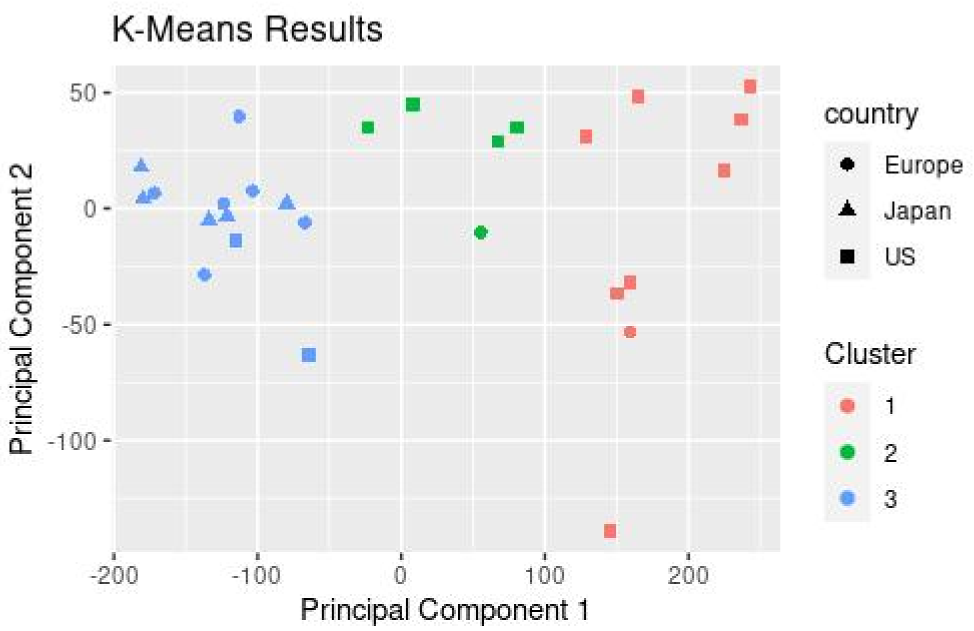
```
mtcars.country <- c(rep("Japan", 3), rep("US", 4), rep("Europe", 7), rep("US", 3),
"Europe", rep("Japan", 3), rep("US", 4), rep("Europe", 4), rep("US", 3))
mtcars$country <- c(mtcars.country)
```

10. Let's now plot

```
plot(mtcarsK3, data = mtcars)
```



```
plot(mtcarsK3, data = mtcars, class = 'country')
```

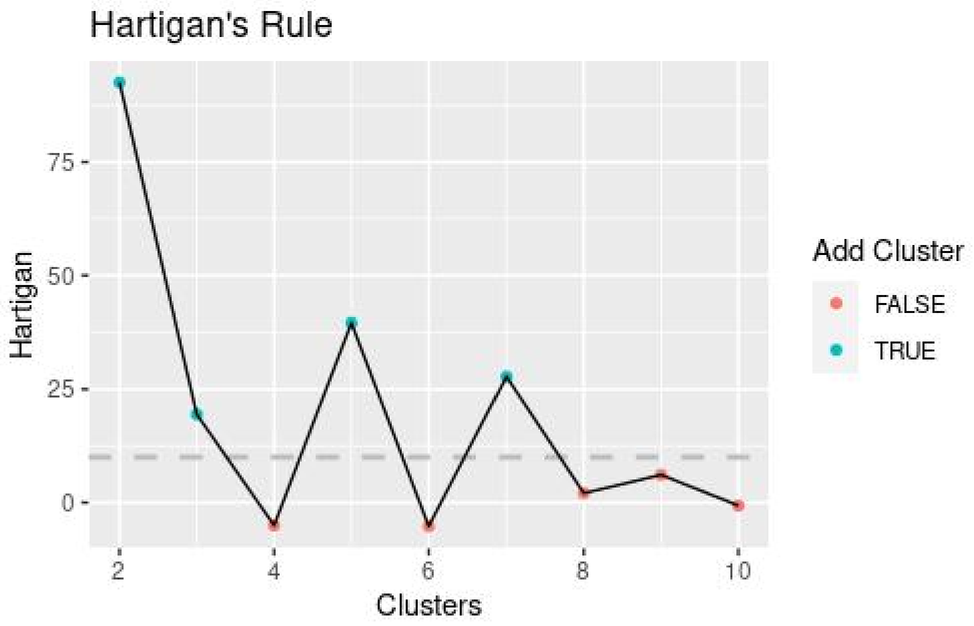


11. Choosing the right number of clusters: I will set my maximum cluster to 10.
NB: I need to re-run my data set since the columns of 'country' and 'cluster' which I previously added will not all our code to run because it doesn't require alphabets.

```
mtcarsBEST <- FitKMeans(mtcars, max.clusters=10)
mtcarsBEST
```

NB: Once I got a FALSE, I stopped counting for clusters

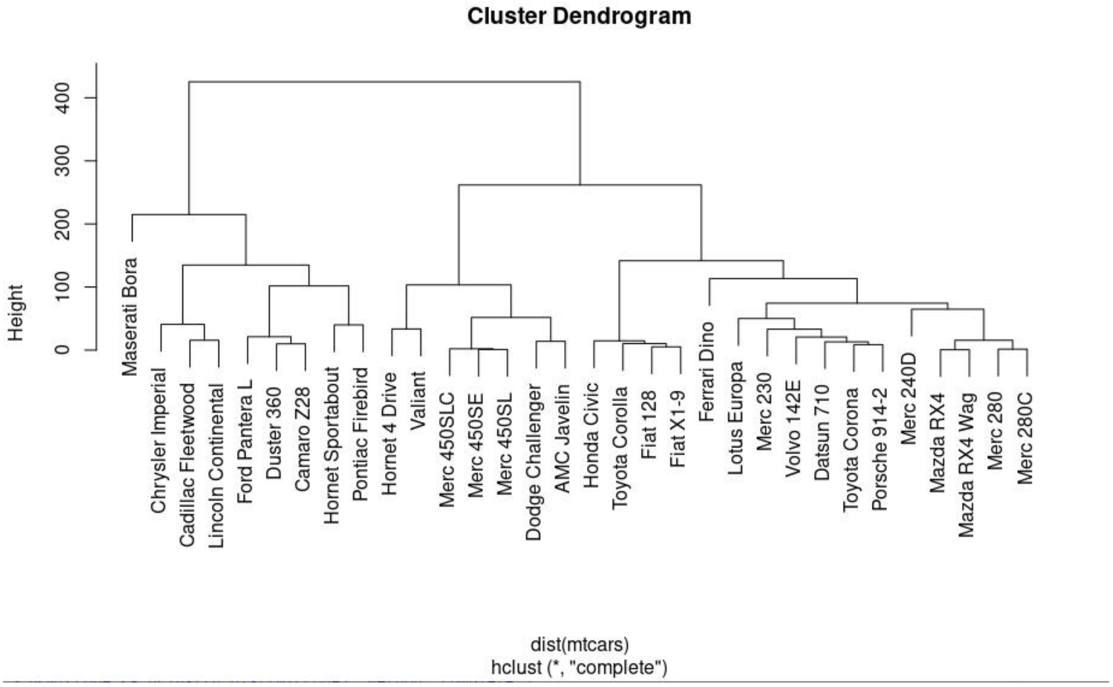
```
PlotHartigan(mtcarsBEST)
```



12. I will now plot again, this time using the number of clusters that I determined. Since I determined 3 clusters just like previously, there is no need to repeat all over.

13. Hierachical Clustering is used to cluster clusters into clusters. Let's see how to implement it: I will start by calculating the distance between the rows. Next, I will pick a method. In this case I will be using the "complete method". "hclust and dist" are inbuilt functions in R.

```
hcmtcars <- hclust(d=dist(mtcars), method='complete')
plot(hcmtcars)
```



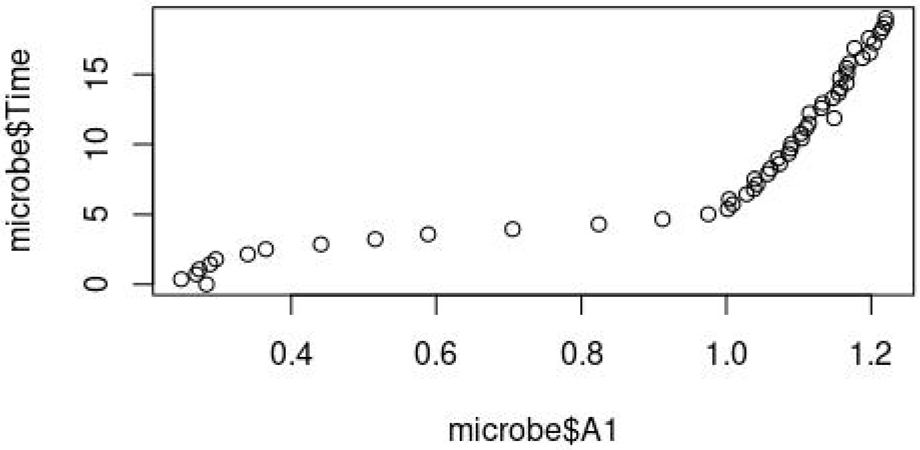
Second K-means clustering

14. Our second data set is 'microbial_stationary_phase.csv' from last week's task. Import .csv file in R

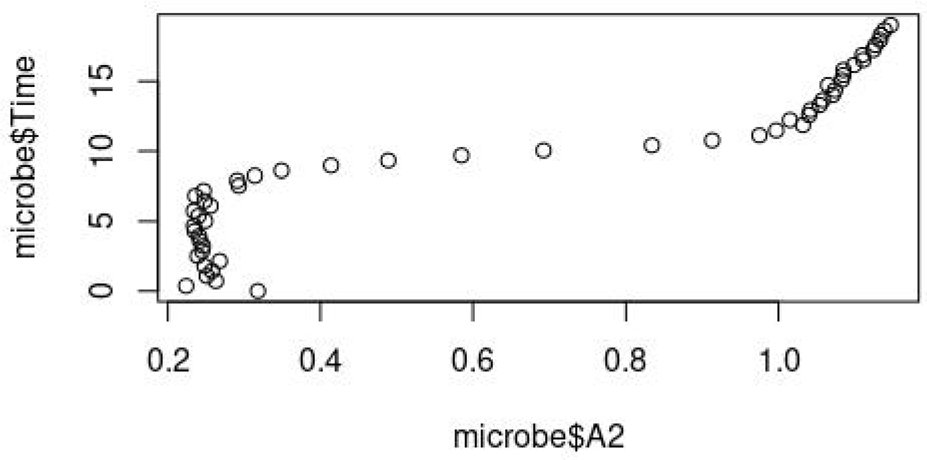
```
microbe <- read.csv(file.choose())
microbe
```

15. For the purpose of visualization, I will like to use any two columns that correlate

```
cor(microbe)
plot (microbe$A1, microbe$Time)
```



```
plot (microbe$A2, microbe$Time)
```



16. So we can have same results. I will set seed which is an arbitrary number. Since I want to get a consistent result from my K-Mean clustering (which is unsupervised). Picking '102' means that I have chosen to index to 102.

```
set.seed(102)
```

17. I will now perform my KMEAN Clustering for now. I want to pick 3 centers (clusters)

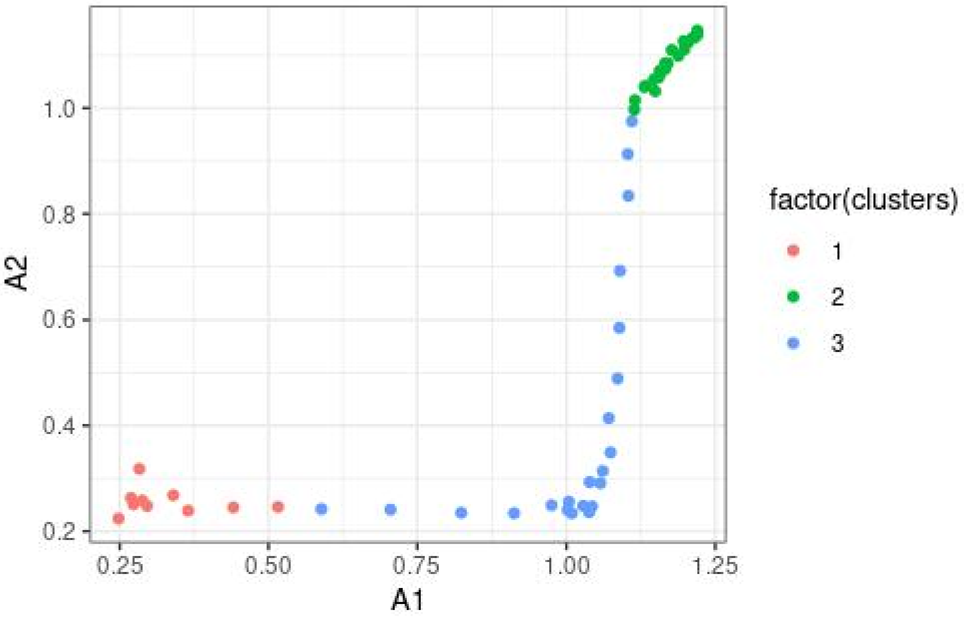
```
microbeK3 <- kmeans (x = microbe, centers = 3)
```

18. Well, I can also add this cluster information to my dataset

```
microbe$clusters <- c(microbeK3$cluster)
```

19. I want to visualize this cluster information for each microbe

```
ggplot(microbe, aes(x = A1, y = A2, color = factor(clusters))) + geom_point() +
theme_bw()
```

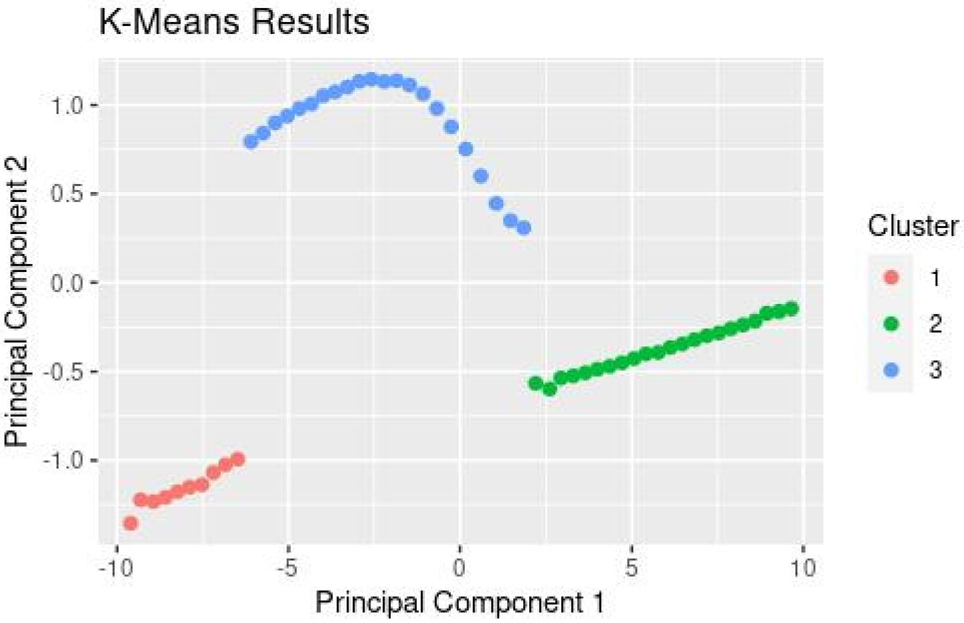


20. Install useful

```
install.packages('useful')
library ('useful')
```

21. Let's plot

```
plot(microbeK3, data = microbe)
```

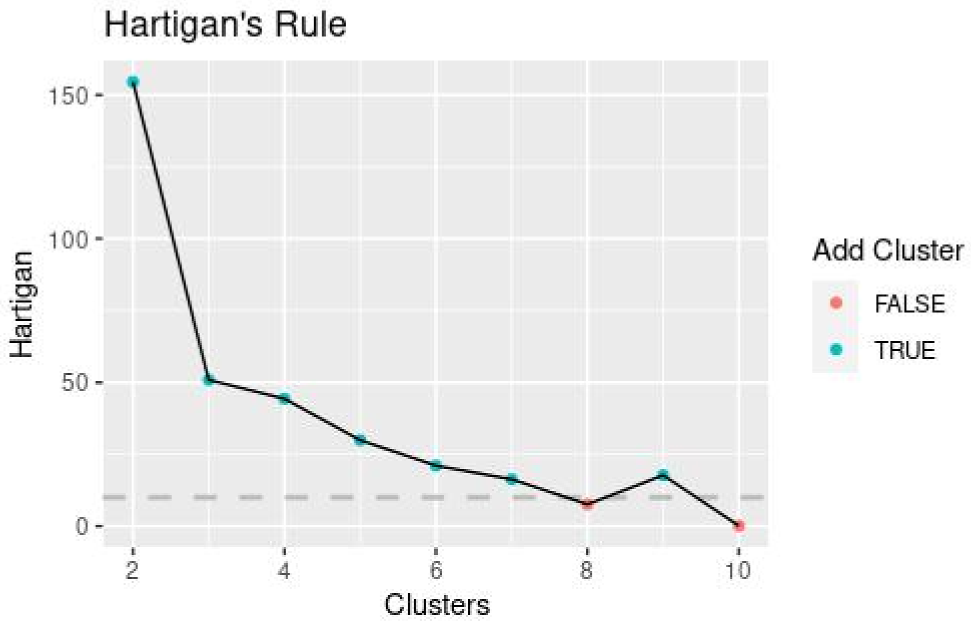


22. Choosing the right number of clusters: I will set my maximum cluster to 10.

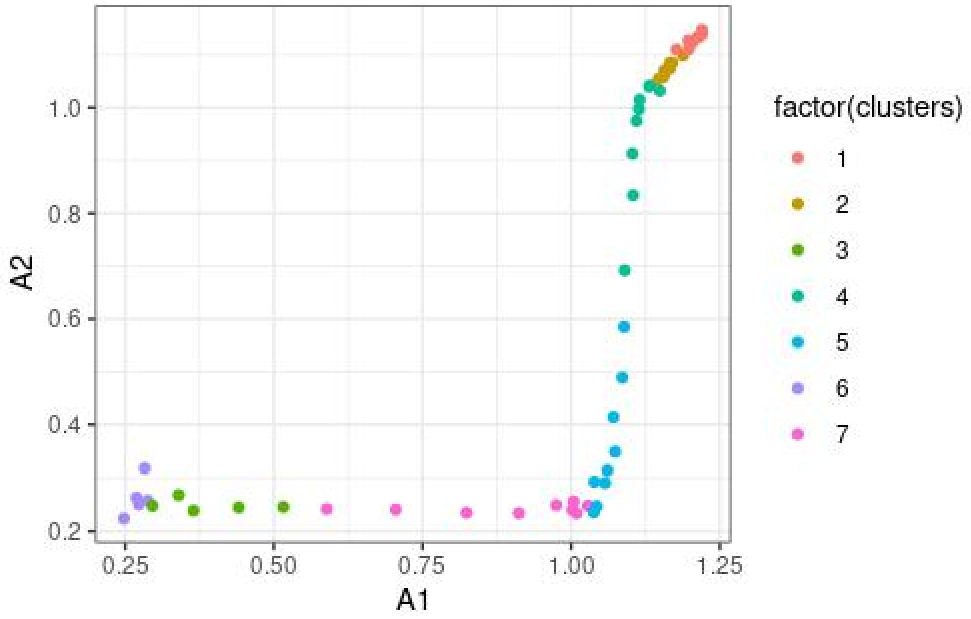
```
microbeBEST <- FitKMeans(microbe, max.clusters=10)
```

23. Once I got a FALSE, I stopped counting for clusters

```
PlotHartigan(microbeBEST)
```

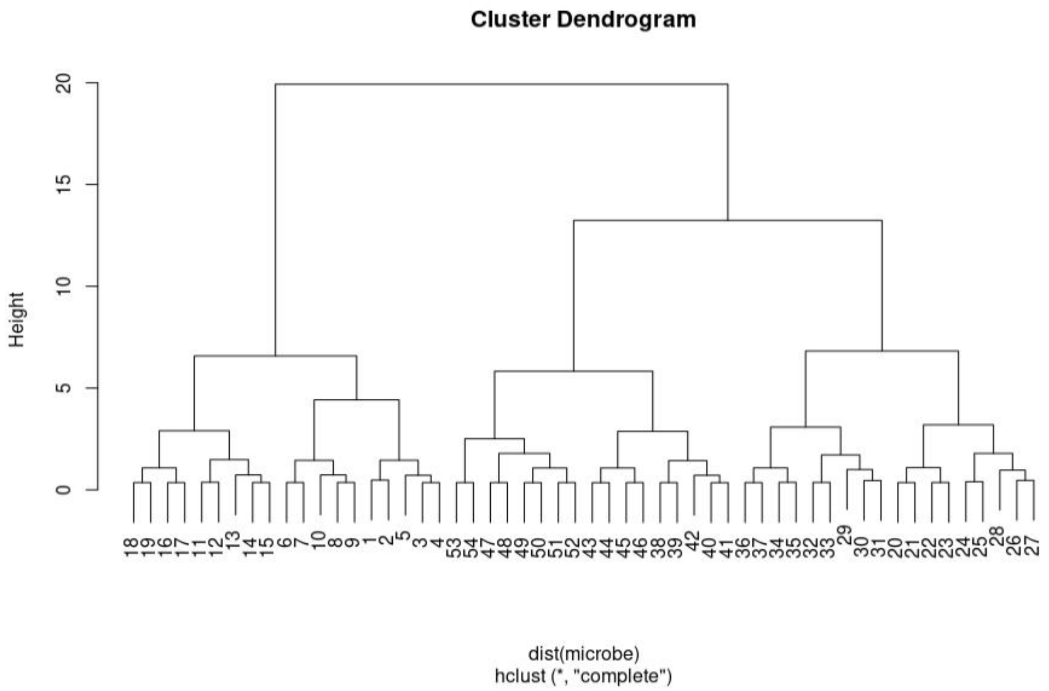


24. I will now plot again, this time using the number of clusters that I determined



25. Hierachical Clustering is used to cluster clusters into clusters. Let's see how to implement it: I will start by calculating the distance between the rows. Next, I will pick a method. In this case I will be using the "complete method". "hclust and dist" are inbuilt functions in R.

```
hcmicrobe <- hclust(d=dist(microbe), method='complete')
plot(hcmicrobe)
```


Cluster Dendrogram
Height
0
5
10
15
20
18 19 16 17 11 12 13 14 15 6 7 10 8 9 1 2 5 3 4 53 54 47 48 49 50 51 52 43 44 45 46 38 39 42 40 41 36 37 34 35 32 33 29 30 31 20 21 22 23 24 25 28 26 27
dist(microbe)
hclust (*, "complete")