STA30005:

MULTIVARIATE ANALYSIS

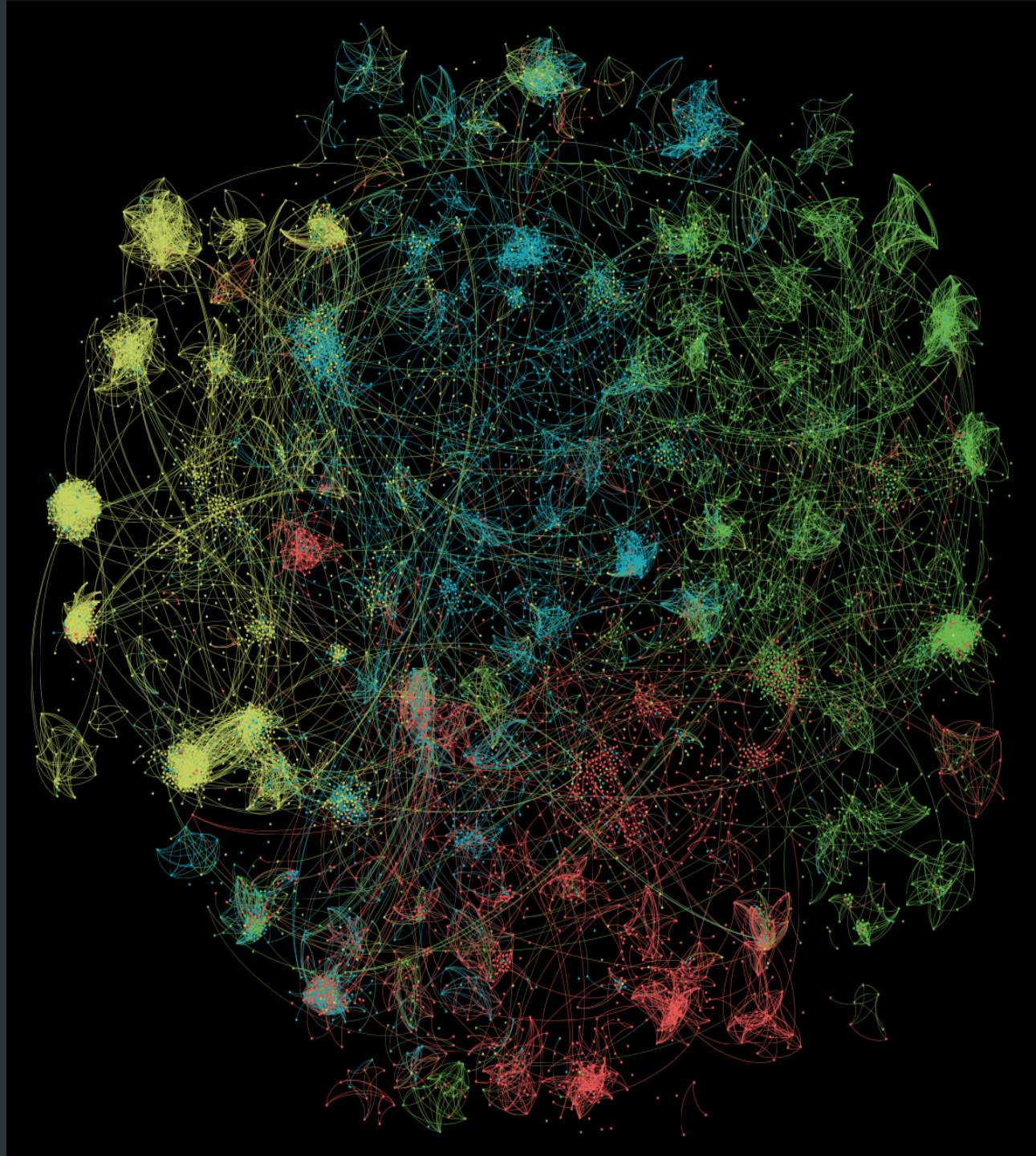
SWINBURNE UNIVERSITY OF TECHNOLOGY

WEEK 8:

CLUSTER ANALYSIS

Tutorial Slides

STA30005: Multivariate Analysis



TASKS

Task A: K-means Clustering

Using the base R data file, USArrests, conduct a K-means cluster analysis and produce an appropriate visualisation to present your findings. Your visualisation should clearly display which US states belong to which cluster.

Task B: Hierarchical Clustering

Using the base R data file, USArrests, conduct a hierarchical (agglomerative) cluster analysis and produce a dendrogram to present your findings. You should compare different distance and linkage methods to find the optimal dendrogram for this scenario.

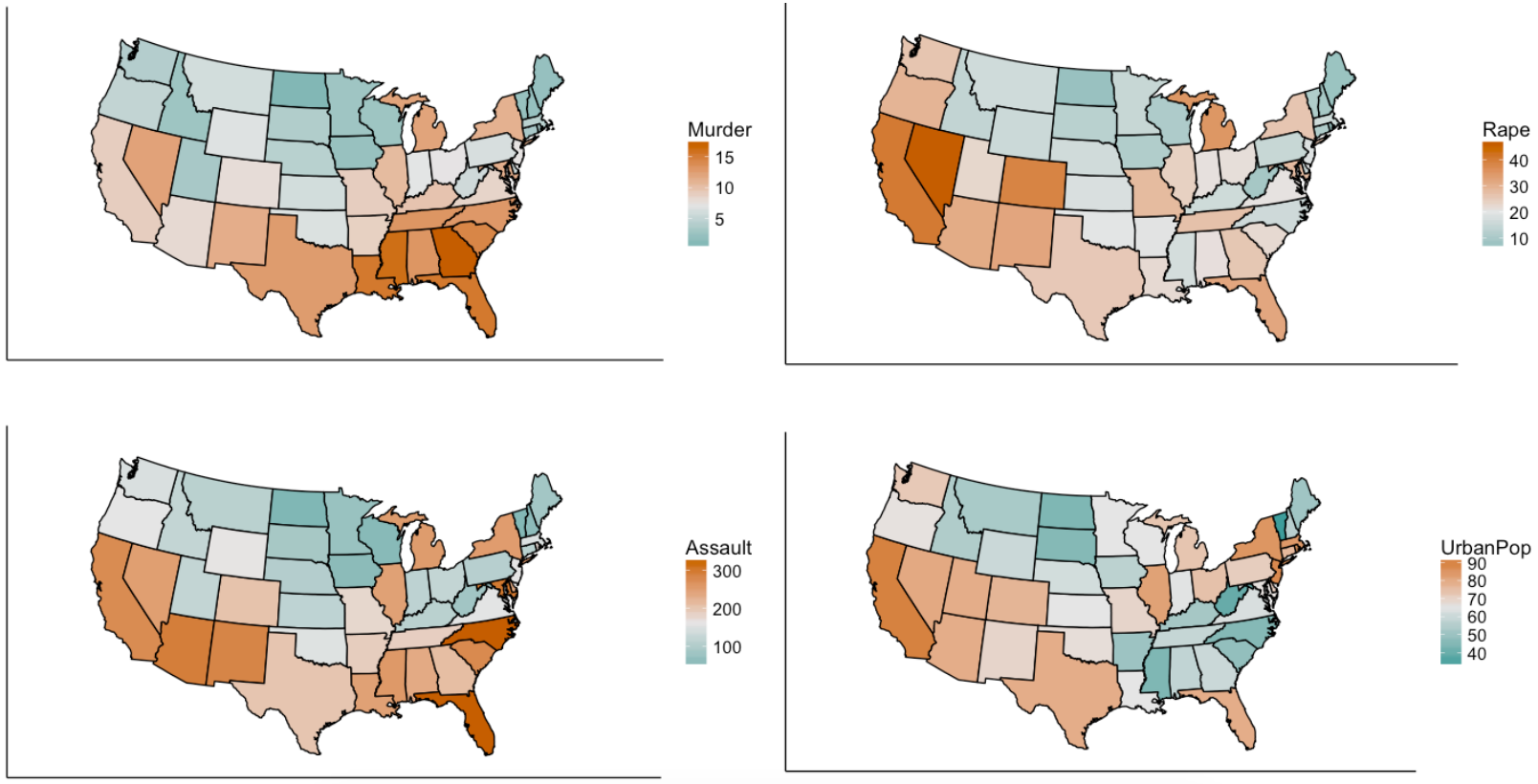
Task C: Enhancing your dendrograms

Recreate your dendrograms (from Task B) as a:

- Circular plot
- Phylogenic tree plot
- Heatmap
- Interactive heatmap

THE DATA

The data file USArrests contains the number of arrests made (per 100,000) in US states for murder, assault and rape as well as the percent of urban population of those states in 1973. We would like to use cluster analysis to determine which states are similar to each other with regards to types of arrests.



TASKS

Task A: K-means Clustering

Using the base R data file, USArrests, conduct a K-means cluster analysis and produce an appropriate visualisation to present your findings. Your visualisation should clearly display which US states belong to which cluster.

Task B: Hierarchical Clustering

Using the base R data file, USArrests, conduct a hierarchical (agglomerative) cluster analysis and produce a dendrogram to present your findings. You should compare different distance and linkage methods to find the optimal dendrogram for this scenario.

Task C: Enhancing your dendrograms

Recreate your dendrograms (from Task B) as a:

- Circular plot
- Phylogenic tree plot
- Heatmap
- Interactive heatmap

K-MEANS CLUSTERING

Data / Package Preparation	Determine K	Compute K-mean clusters	Visualise

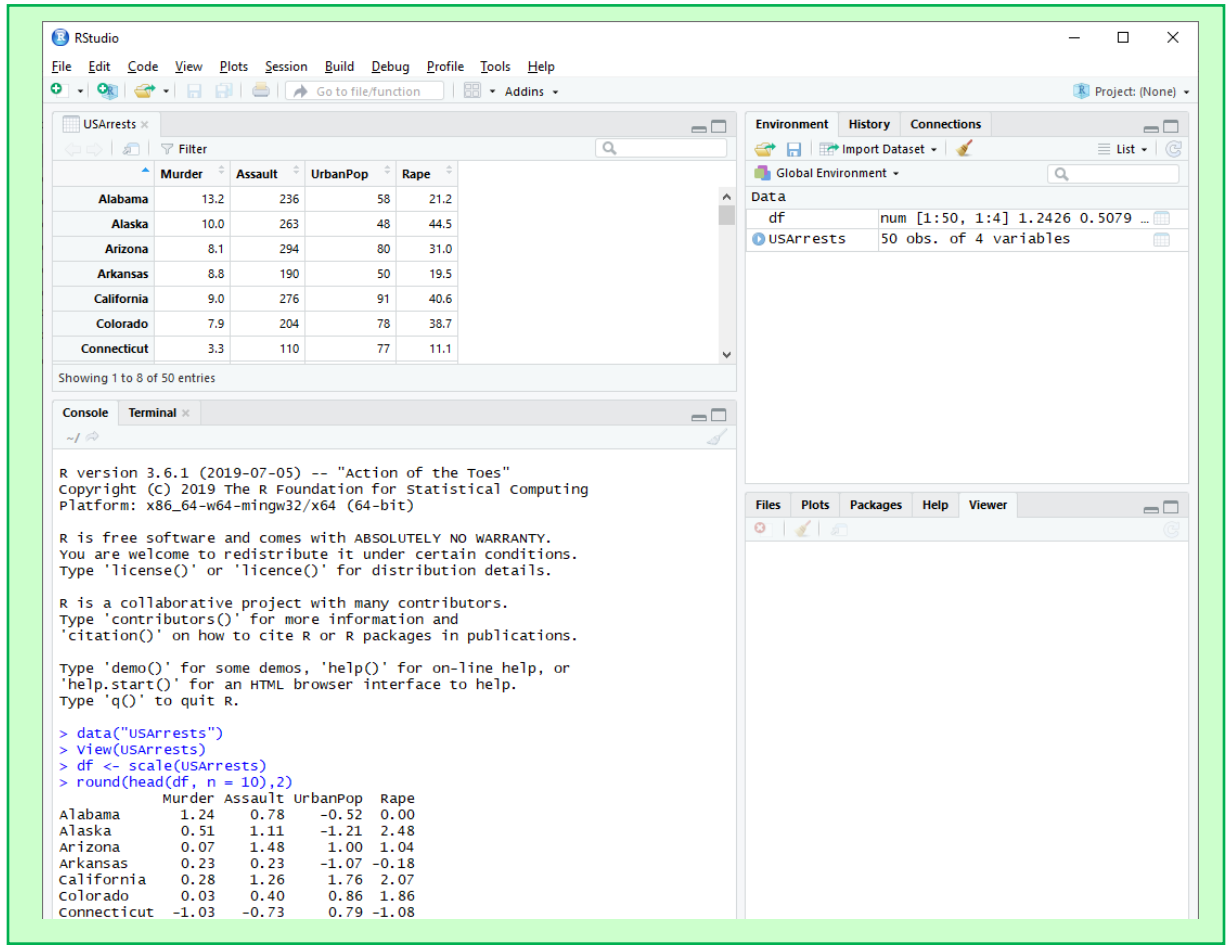
```
# Load the data (this is a base data set for R)
data("USArrests")

# Show the data
View(USArrests)

# Define (I like to use df) and Standardize the data
df <- scale(USArrests)

# Show the first 10 rows (round to 2 decimal places)
round(head(df, n = 10),2)

# Set a seed so that we can reproduce the same results
set.seed(123)
```



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

K-MEANS CLUSTERING

Data / Package Preparation	Determine K	Compute K-mean clusters	Visualise

```
# Install / Load the factoextra package (this is used to generate cluster plots)
install.packages("factoextra")
library(factoextra)

# Install / Load the cluster package (this is used to determine K)
install.packages(“cluster")
library(cluster)
```

Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

K-MEANS CLUSTERING

Data / Package Preparation	Determine K	Compute K-mean clusters	Visualise

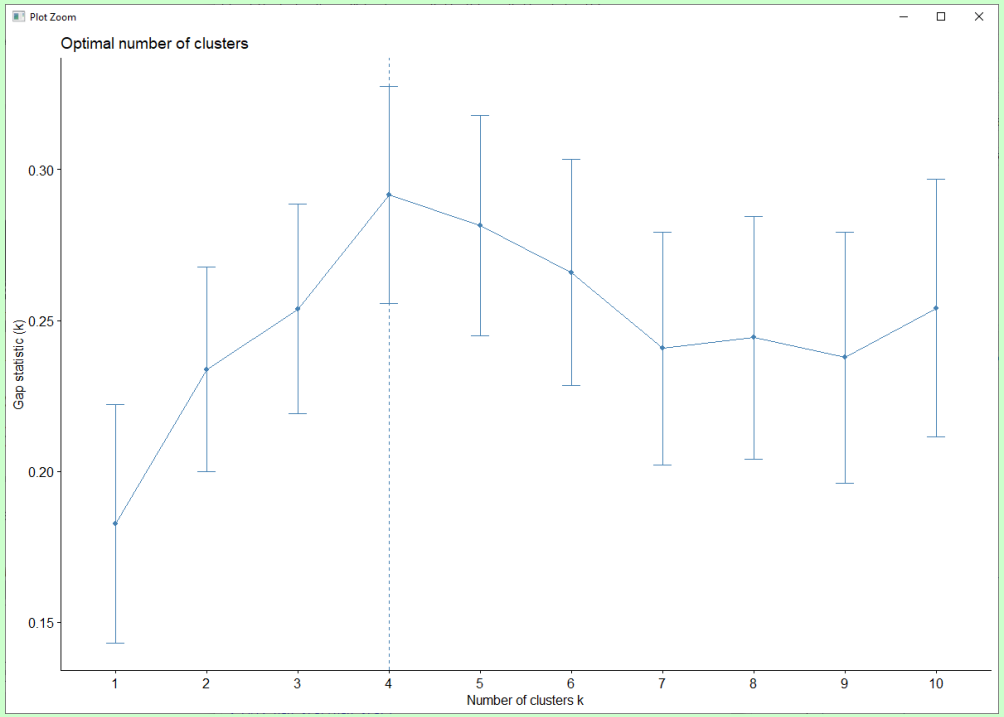
There are several traditional methods for estimating K (such as Elbow and Silhouette plots). Modern methods have become more sophisticated due to advancements in computing. We will use the Gap Statistic method here

```
# Compute the gap statistic
gap_stat <- clusGap(df, FUN = kmeans, nstart = 25, K.max = 10, B = 500)
```

This function will conduct a k-means cluster analysis using the gap statistic method (with 25 random starting points) using 500 bootstrap samples

```
# Visualise the gap statistic plot
fviz_gap_stat(gap_stat)
```

Based upon the gap statistic method, K = 4 is optimal



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

K-MEANS CLUSTERING

Data / Package Preparation	Determine K	Compute K-mean clusters	Visualise

As the final result of k-means clustering result is sensitive to the random starting assignments, we specify nstart = 25. This means that R will try 25 different random starting assignments and then select the best results corresponding to the one with the lowest within cluster variation. The default value of nstart in R is one. But, it's strongly recommended to compute k-means clustering with a large value of nstart such as 25 or 50, in order to have a more stable result.

```
# Compute k-means with K = 4
km.res <- kmeans(df, 4, nstart = 25)

# Print the k-means results
print(km.res)
```

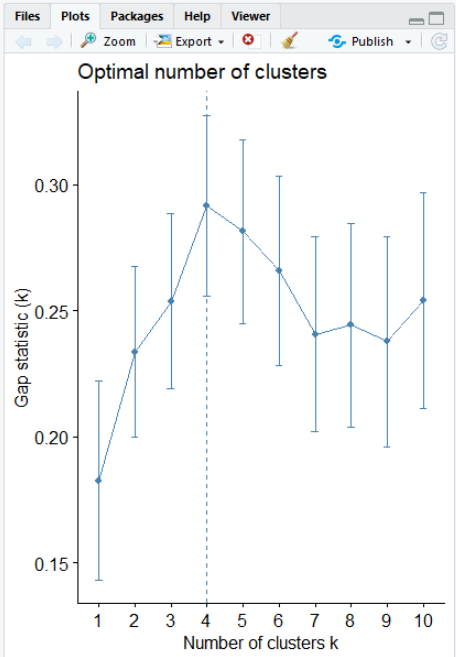
The amount of variation within each of the four clusters is provided with these results

```
> print(km.res)
K-means clustering with 4 clusters of sizes 13, 13, 8, 16

Cluster means:
      Murder    Assault   UrbanPop        Rape
1 -0.9615407 -1.1066010 -0.9301069 -0.96676331
2  0.6950701  1.0394414  0.7226370  1.27693964
3  1.4118898  0.8743346 -0.8145211  0.01927104
4 -0.4894375 -0.3826001  0.5758298 -0.26165379

Clustering vector:
       Alabama         Alaska        Arizona       Arkansas
             3              2              2              3
    California       Colorado    Connecticut       Delaware
             2              2              4              4
       Florida        Georgia         Hawaii          Idaho
             2              3              4              1
      Illinois        Indiana           Iowa         Kansas
             2              4              1              4
      Kentucky      Louisiana          Maine       Maryland
             1              3              1              2
 Massachusetts       Michigan      Minnesota    Mississippi
             4              2              1              3
      Missouri        Montana       Nebraska         Nevada
             2              1              1              2
 New Hampshire     New Jersey     New Mexico       New York
             1              4              2              2
North Carolina   North Dakota           Ohio       Oklahoma
             3              1              4              4
        Oregon   Pennsylvania   Rhode Island South Carolina
             4              4              4              3
  South Dakota      Tennessee          Texas           Utah
             1              3              2              4
       Vermont       Virginia     Washington  West Virginia
             1              4              4              1
     Wisconsin        Wyoming
             1              4

Within cluster sum of squares by cluster:
[1] 11.952463 19.922437  8.316061 16.212213
 (between_SS / total_SS =  71.2 %)
```



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

K-MEANS CLUSTERING

Data / Package Preparation	Determine K	Compute K-mean clusters	Visualise

 # View the cluster means by the original data

round(aggregate(USArrests, by=list(cluster=km.res$cluster), mean),2)

Based upon these results, we can see that:

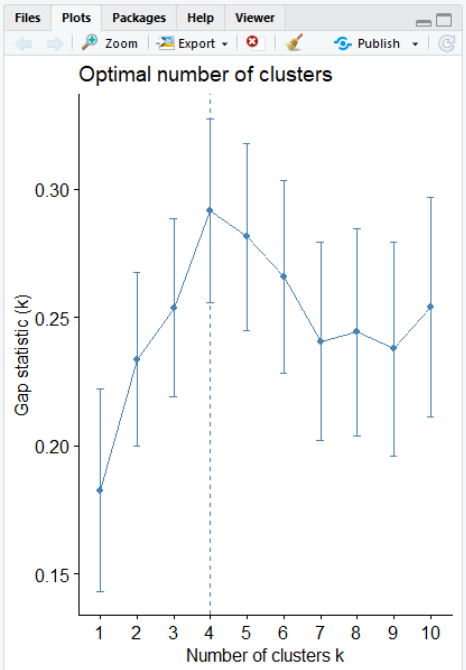
- US states in cluster 1 are low across all variables
- US states in cluster 2 are high across all variables
- US states in cluster 3 are high across all variables
- US states in cluster 4 are low across all variables

```
       Kentucky      Louisiana          Maine       Maryland
              1              3              1              2
  Massachusetts       Michigan      Minnesota    Mississippi
              4              2              1              3
       Missouri        Montana       Nebraska         Nevada
              2              1              1              2
  New Hampshire     New Jersey     New Mexico       New York
              1              4              2              2
 North Carolina   North Dakota           Ohio       Oklahoma
              3              1              4              4
         Oregon   Pennsylvania   Rhode Island South Carolina
              4              4              4              3
   South Dakota      Tennessee          Texas           Utah
              1              3              2              4
        Vermont       Virginia     Washington  West Virginia
              1              4              4              1
      Wisconsin        Wyoming
              1              4

Within cluster sum of squares by cluster:
[1] 11.952463 19.922437  8.316061 16.212213
 (between_SS / total_SS =  71.2 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"
[5] "tot.withinss" "betweenss"    "size"         "iter"
[9] "ifault"
> aggregate(USArrests, by=list(cluster=km.res$cluster), mean)
  cluster   Murder   Assault UrbanPop     Rape
1       1  3.60000  78.53846 52.07692 12.17692
2       2 10.81538 257.38462 76.00000 33.19231
3       3 13.93750 243.62500 53.75000 21.41250
4       4  5.65625 138.87500 73.87500 18.78125
> round(aggregate(USArrests, by=list(cluster=km.res$cluster), mean),2)
  cluster Murder Assault UrbanPop  Rape
1       1   3.60   78.54    52.08 12.18
2       2  10.82  257.38    76.00 33.19
3       3  13.94  243.62    53.75 21.41
4       4   5.66  138.88    73.88 18.78
>
```



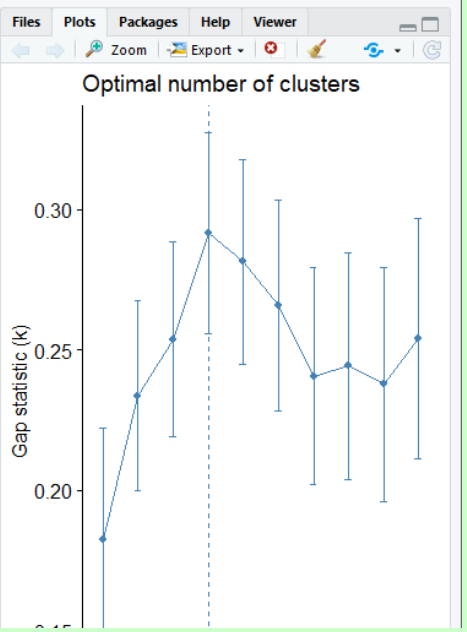
Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

K-MEANS CLUSTERING

Data / Package Preparation	Determine K	Compute K-mean clusters	Visualise

```
# View which US state was assigned to which cluster
dd <- cbind(USArrests, cluster = km.res$cluster)
show(dd)
```

```
> dd <- cbind(USArrests, cluster = km.res$cluster)
> show(dd)
               Murder Assault UrbanPop Rape cluster
Alabama          13.2     236       58 21.2       3
Alaska           10.0     263       48 44.5       2
Arizona           8.1     294       80 31.0       2
Arkansas          8.8     190       50 19.5       3
California        9.0     276       91 40.6       2
Colorado          7.9     204       78 38.7       2
Connecticut       3.3     110       77 11.1       4
Delaware          5.9     238       72 15.8       4
Florida          15.4     335       80 31.9       2
Georgia          17.4     211       60 25.8       3
Hawaii            5.3      46       83 20.2       4
Idaho             2.6     120       54 14.2       1
Illinois         10.4     249       83 24.0       2
Indiana           7.2     113       65 21.0       4
Iowa              2.2      56       57 11.3       1
Kansas            6.0     115       66 18.0       4
Kentucky          9.7     109       52 16.3       1
Louisiana        15.4     249       66 22.2       3
Maine             2.1      83       51  7.8       1
Maryland         11.3     300       67 27.8       2
Massachusetts     4.4     149       85 16.3       4
Michigan         12.1     255       74 35.1       2
Minnesota         2.7      72       66 14.9       1
Mississippi      16.1     259       44 17.1       3
Missouri          9.0     178       70 28.2       2
Montana           6.0     109       53 16.4       1
Nebraska          4.3     102       62 16.5       1
Nevada           12.2     252       81 46.0       2
New Hampshire     2.1      57       56  9.5       1
New Jersey        7.4     159       89 18.8       4
New Mexico       11.4     285       70 32.1       2
New York         11.1     254       86 26.1       2
North Carolina   13.0     337       45 16.1       3
```



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

K-MEANS CLUSTERING

Data / Package Preparation	Determine K	Compute K-mean clusters	Visualise

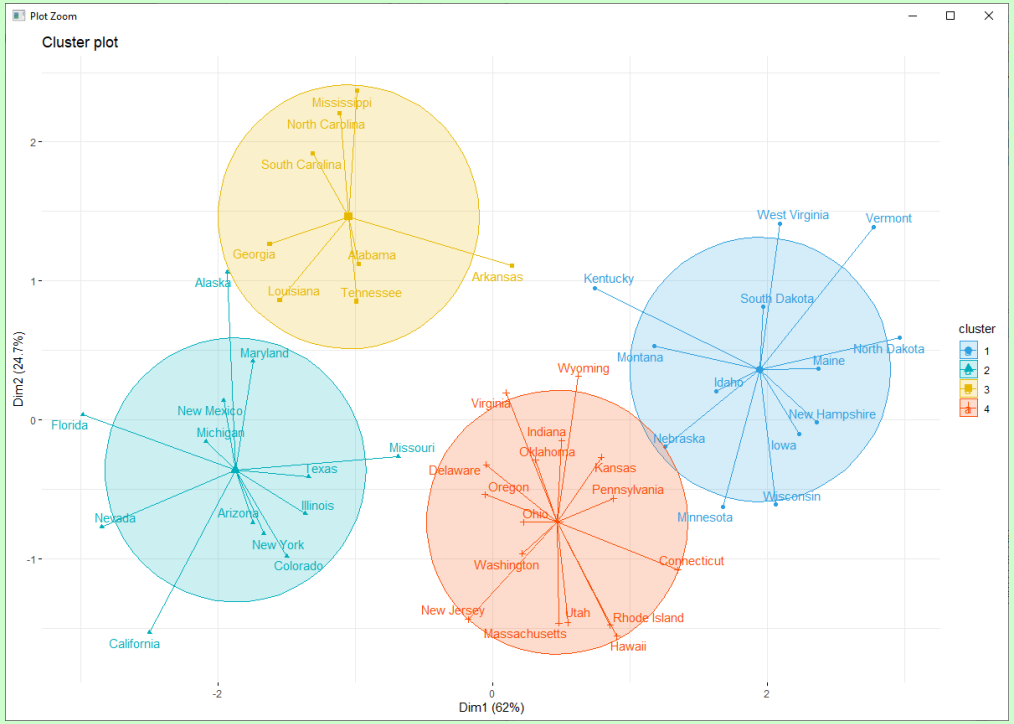
```
# Visualise the clustering
fviz_cluster(km.res, data = df)
```

Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

K-MEANS CLUSTERING

Data / Package Preparation	Determine K	Compute K-mean clusters	Visualise

```
# Visualise the clustering (enhanced)
fviz_cluster(km.res, data = df, palette = c("#2E9FDF",
"#00AFBB", "#E7B800", "#FC4E07"), ellipse.type = "euclid",
star.plot = TRUE, repel = TRUE, ggtheme =
theme_minimal())
```



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

TASKS

Task A: K-means Clustering

Using the base R data file, USArrests, conduct a K-means cluster analysis and produce an appropriate visualisation to present your findings. Your visualisation should clearly display which US states belong to which cluster.

Task B: Hierarchical Clustering

Using the base R data file, USArrests, conduct a hierarchical (agglomerative) cluster analysis and produce a dendrogram to present your findings. You should compare different distance and linkage methods to find the optimal dendrogram for this scenario.

Task C: Enhancing your dendrograms

Recreate your dendrograms (from Task B) as a:

- Circular plot
- Phylogenic tree plot
- Heatmap
- Interactive heatmap

AGGLOMERATIVE CLUSTERING

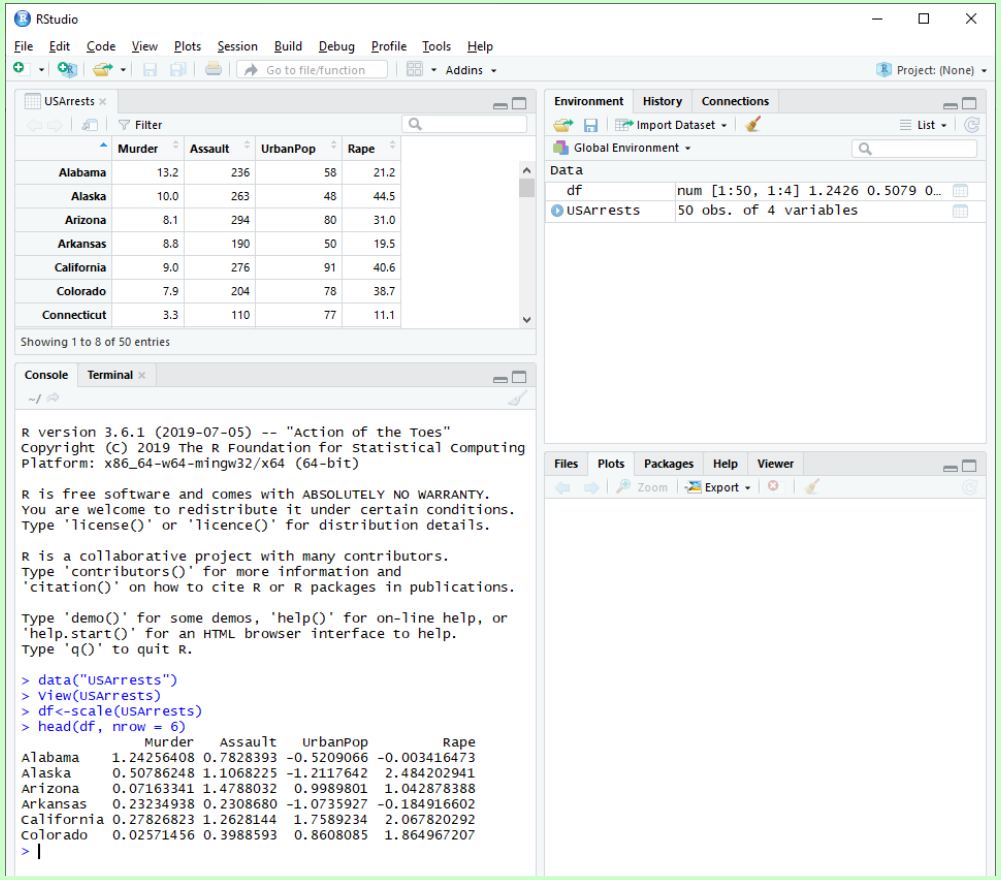
Data Preparation	Similarity Measures	Linkage	Dendrogram	Verification	Cutting

```
# Load the data (this is a base data set for R)
data("USArrests")

# Show the data
View(USArrests)

# Define (I like to use df) and Standardize the data
df <- scale(USArrests)

# Show the first 6 rows (round to 2 decimal places)
round(head(df, nrow = 6),2)
```



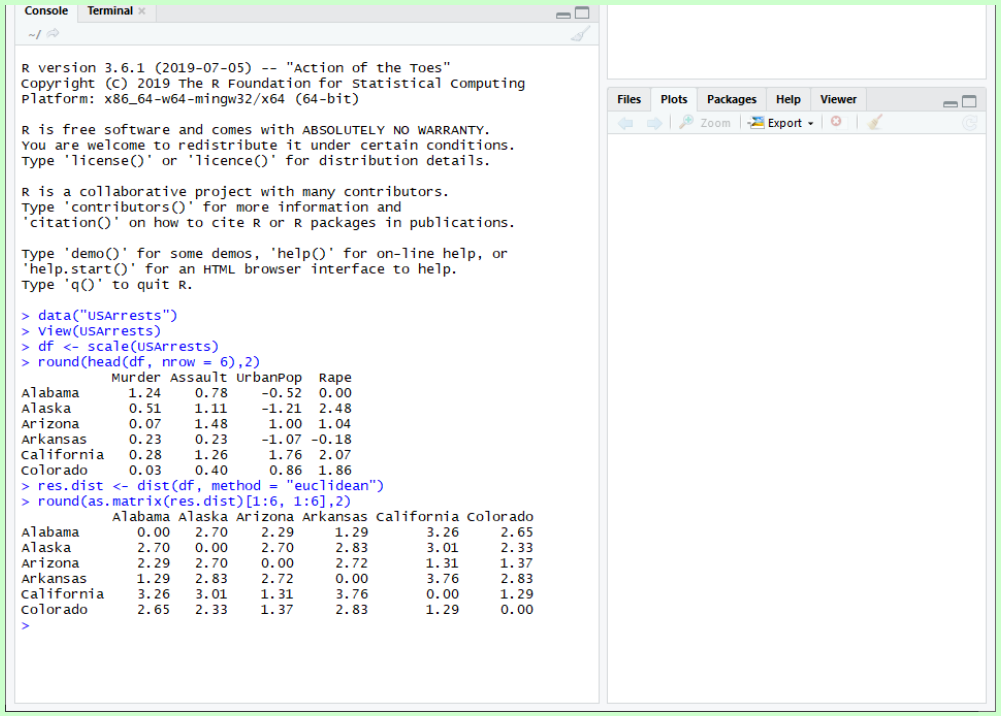
Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

AGGLOMERATIVE CLUSTERING

There are many methods to calculate the (dis)similarity information (e.g. Euclidean, Manhattan, etc.). In R, we can use the function *dist*() to compute distances between every pair of objects in a data set

```
# Define and compute the dissimilarity matrix
res.dist <- dist(df, method = "euclidean")

# View distance information in matrix form (first 6)
round(as.matrix(res.dist)[1:6, 1:6],2)
```



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

AGGLOMERATIVE CLUSTERING

The linkage function takes the distance information, returned by the function *dist*(), and groups pairs of objects into clusters based on their similarity. Next, these newly formed clusters are linked to each other to create bigger clusters. We can use *hclust*() to create the hierarchical tree

```
# Define and apply a hierarchical clustering linkage (in this example we will use Ward's method)
res.hc <- hclust(d = res.dist, method = "ward.D2")
```

Common linkage methods are described below:

- **Maximum or complete linkage**: The distance between two clusters is defined as the maximum value of all pairwise distances between the elements in cluster 1 and the elements in cluster 2. It tends to produce more compact clusters.
- **Minimum or single linkage**: The distance between two clusters is defined as the minimum value of all pairwise distances between the elements in cluster 1 and the elements in cluster 2. It tends to produce long, "loose" clusters.
- **Mean or average linkage**: The distance between two clusters is defined as the average distance between the elements in cluster 1 and the elements in cluster 2.
- **Centroid linkage**: The distance between two clusters is defined as the distance between the centroid for cluster 1 (a mean vector of length p variables) and the centroid for cluster 2.
- **Ward's minimum variance method**: It minimizes the total within-cluster variance. At each step the pair of clusters with minimum between-cluster distance are merged.

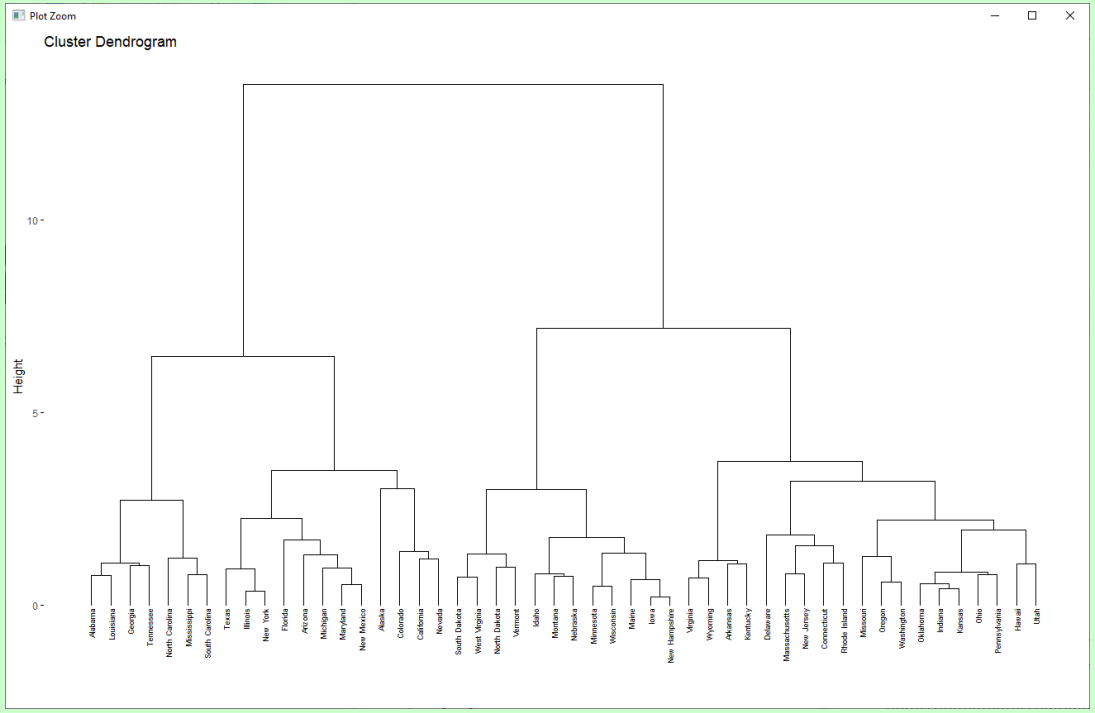
AGGLOMERATIVE CLUSTERING

Data Preparation	Similarity Measures	Linkage	Dendrogram	Verification	Cutting

Dendrograms correspond to the graphical representation of the hierarchical tree generated by the function *hclust*(). Here, we'll use the function *fviz_dend*() [in factoextra R package] to produce our dendrograms.

```
# Install and Load the factoextra package
install.packages(factoextra)
library(factoextra)

# Plot the dendrogram
fviz_dend(res.hc, cex = 0.5)
```

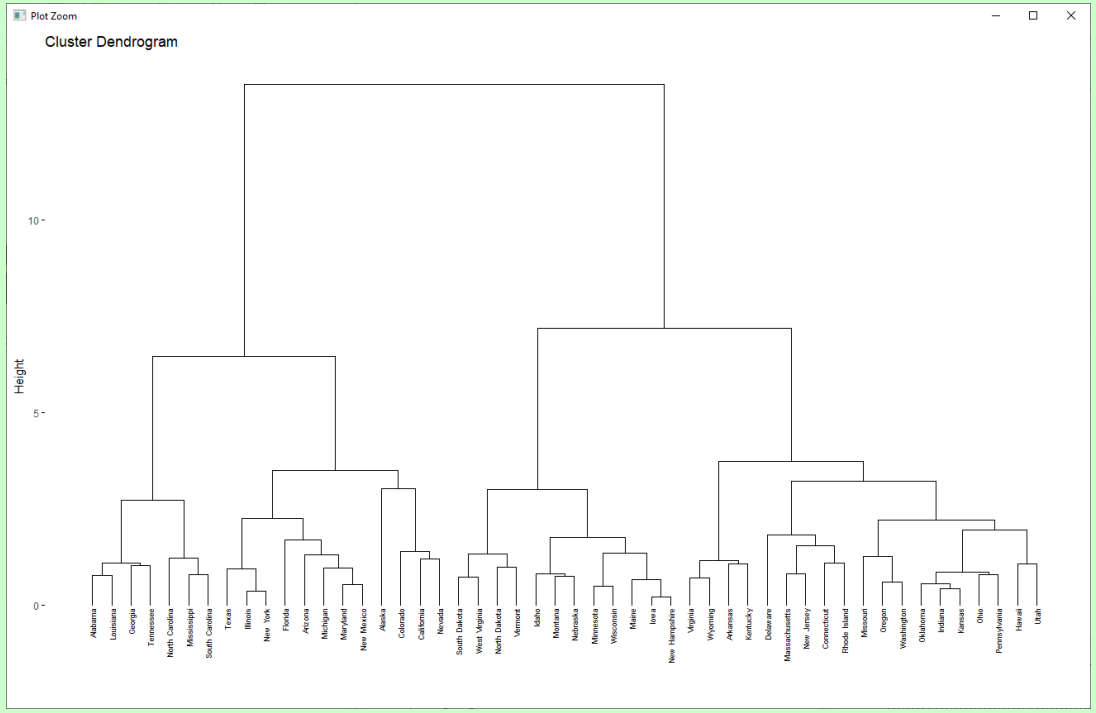


Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

AGGLOMERATIVE CLUSTERING

Notes about this dendrogram:

- In this dendrogram, each leaf corresponds to one object.
- As we move up the tree, objects that are similar to each other are combined into branches, which are themselves fused at a higher height.
- The height of the fusion, provided on the vertical axis, indicates the (dis)similarity/distance between two objects/clusters.
- The higher the height of the fusion, the less similar the objects are. This height is known as the *cophenetic distance* between the two objects.



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

AGGLOMERATIVE CLUSTERING

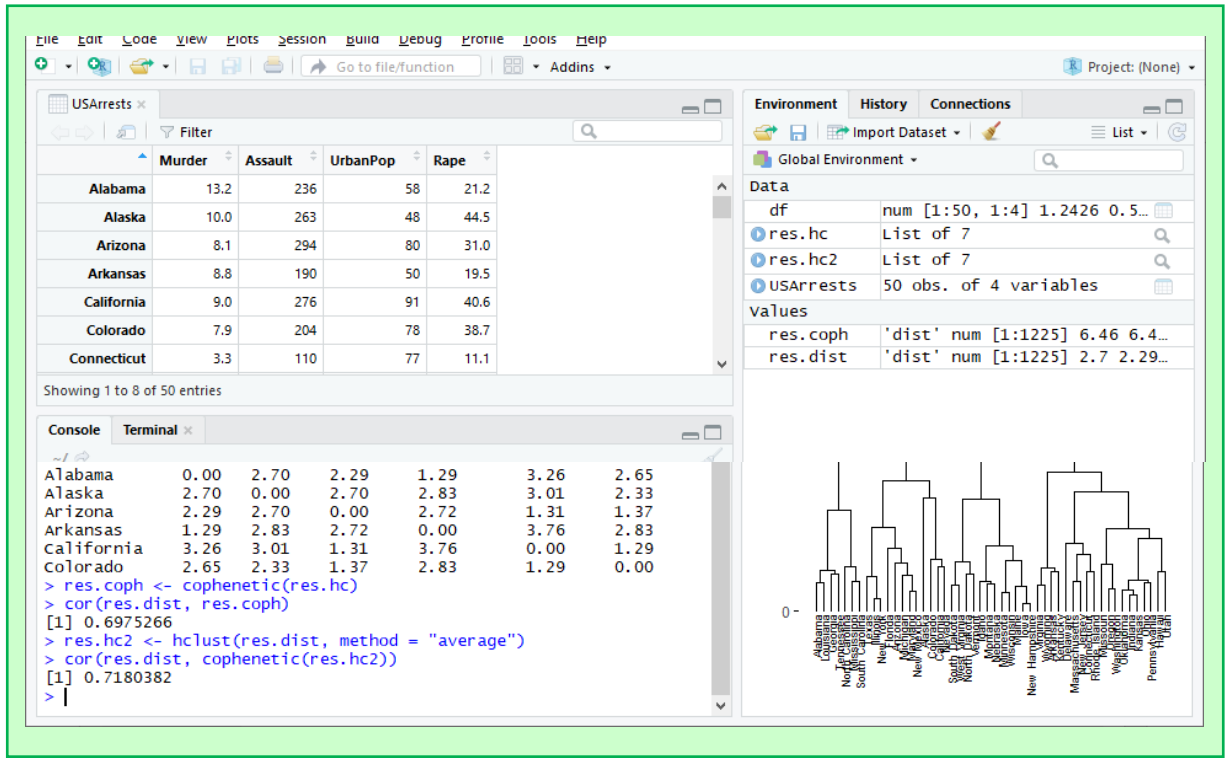
We should now assess the dendrogram heights against the original distances.

```
# Compute the cophentic distance
res.coph <- cophenetic(res.hc)

# Correlate cophenetic distance and original distance
cor(res.dist, res.coph)
```

Note: the closer the value of the correlation coefficient is to 1, the more accurate the clustering solution reflects your data (with values above 0.75 being very good). You may want to try different linkage methods and evaluate the correlation again:

```
res.hc2 <- hclust(res.dist, method = "average")
cor(res.dist, cophenetic(res.hc2))
```



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

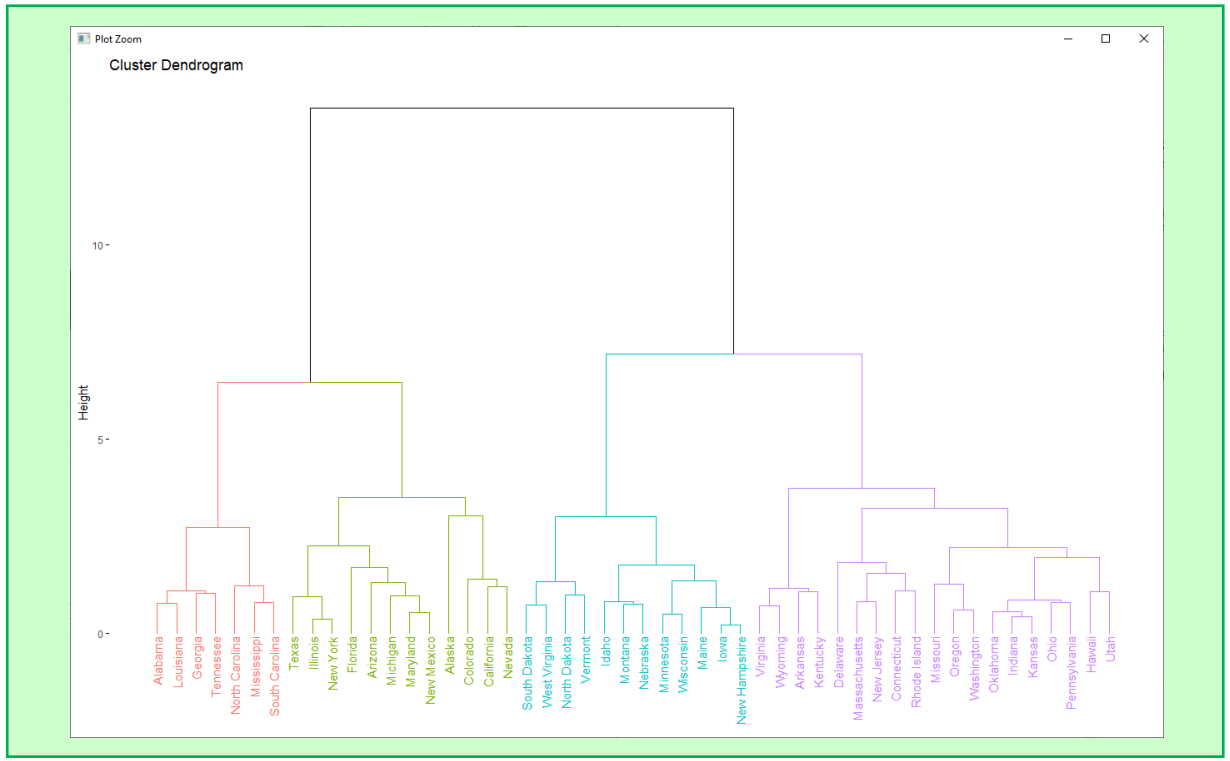
AGGLOMERATIVE CLUSTERING

The R base function *cutree*() can be used to cut a tree, generated by the *hclust*() function, into several groups either by specifying the desired number of groups or the cut height.

```
# Cut the dendrogram into 4 groups
grp <- cutree(res.hc, k = 4)

# Plot this new dendrogram
fviz_dend(res.hc, k = 4)
```

Try recreating the dendrogram with different values of k or cut points



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

TASKS

Task A: K-means Clustering

Using the base R data file, USArrests, conduct a K-means cluster analysis and produce an appropriate visualisation to present your findings. Your visualisation should clearly display which US states belong to which cluster.

Task B: Hierarchical Clustering

Using the base R data file, USArrests, conduct a hierarchical (agglomerative) cluster analysis and produce a dendrogram to present your findings. You should compare different distance and linkage methods to find the optimal dendrogram for this scenario.

Task C: Enhancing your dendrograms

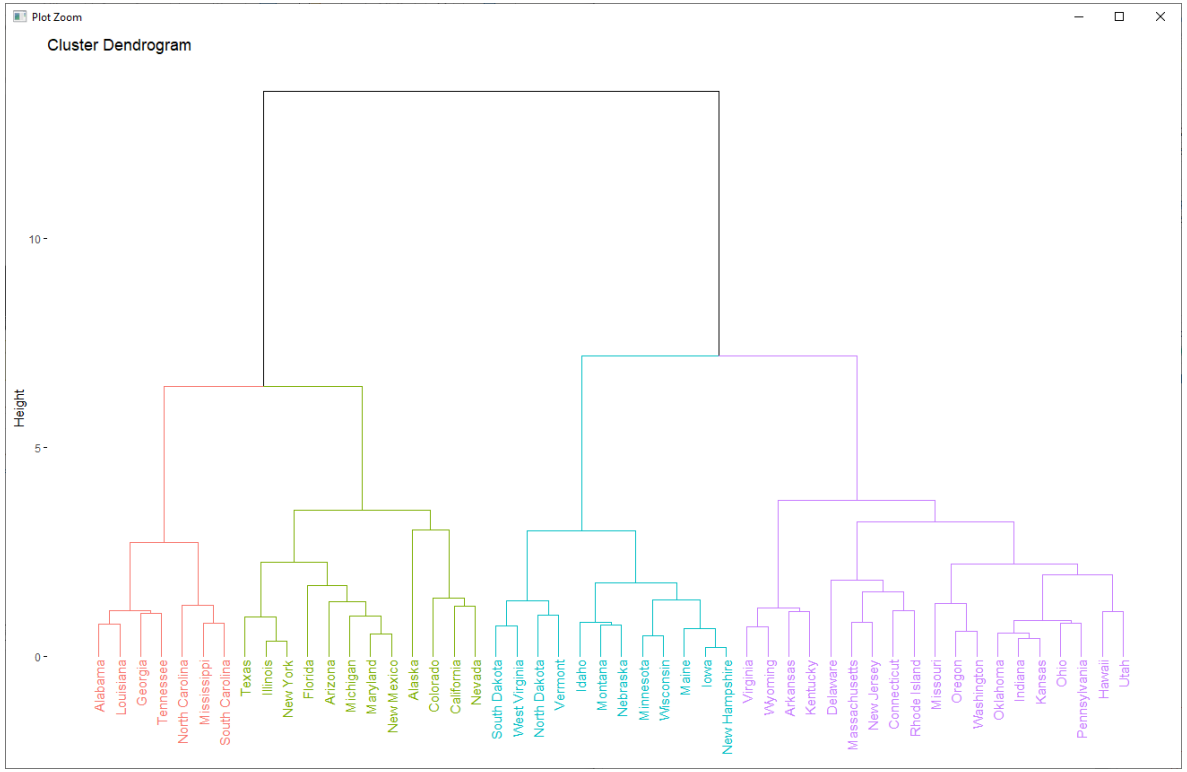
Recreate your dendrograms (from Task B) as a:

- Circular plot
- Phylogenic tree plot
- Heatmap
- Interactive heatmap

ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

In this task we will enhance the dendrogram we produced earlier with 4 new plots:



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

ENHANCING YOUR DENDROGRAMS

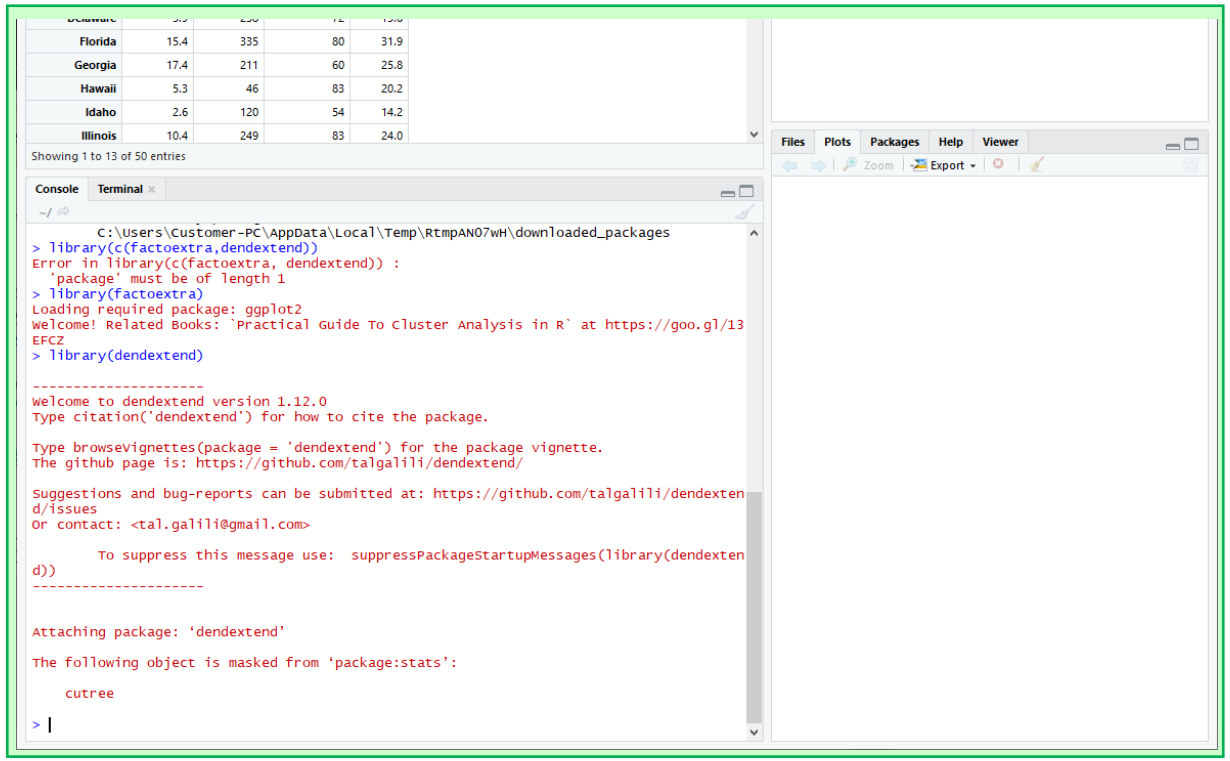
Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

Suppose we are starting from scratch and have not produced the outputs from the previous task

```
# Load the data (this is a base data set for R)
data("USArrests")

# Install / Load the required packages
install.packages(c("factoextra", "dendextend"))
library(factoextra)
library(dendextend)

# Compute distances and hierarchical clustering
dd <- dist(scale(USArrests), method = "euclidean")
hc <- hclust(dd, method = "ward.D2")
```



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

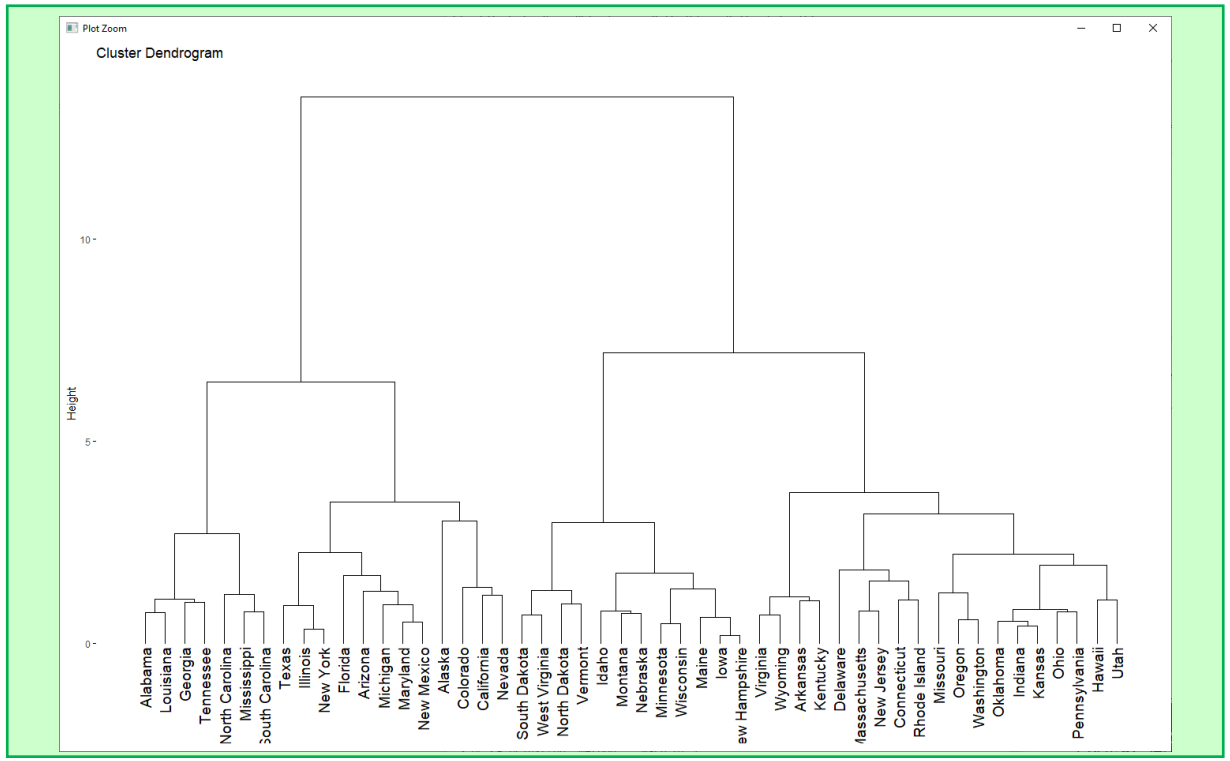
ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

We will begin by recreating the dendrogram with the *fviz_dend* function

```
# Create the default dendrogram
fviz_dend(hc, cex = 1)
```

This changes the size of the labels



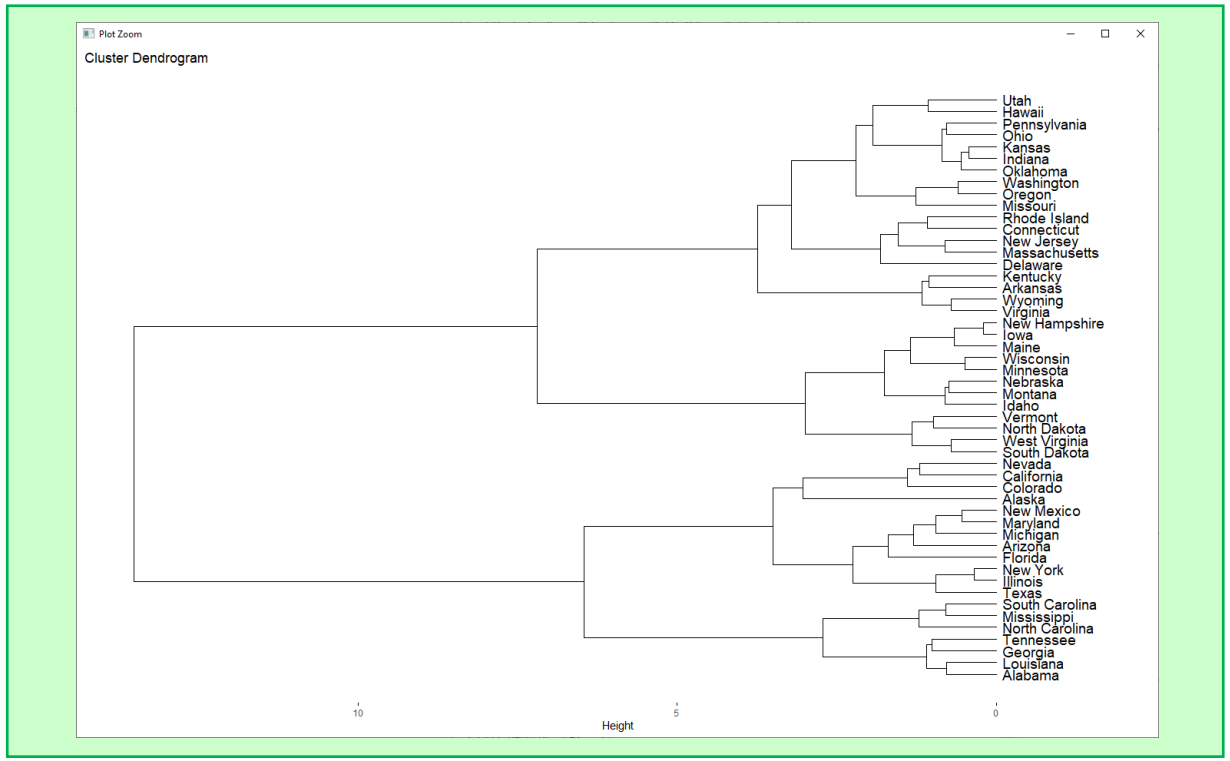
Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

We can rotate the orientation of the dendrogram by adding the *horiz* command

```
# Create a rotated dendrogram
fviz_dend(hc, cex = 1, horiz = TRUE)
```



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

```
# Adding colour dendrogram
fviz_dend(hc, k = 4, cex = 0.8, horiz = TRUE, k_colors = "jco", rect = TRUE, rect_border = "jco", rect_fill = TRUE)
```

Cut the dendrogram with 4 clusters

Label size

Rotate dendrogram horizontally

Use the jco colours

Add a rectangle around the clusters

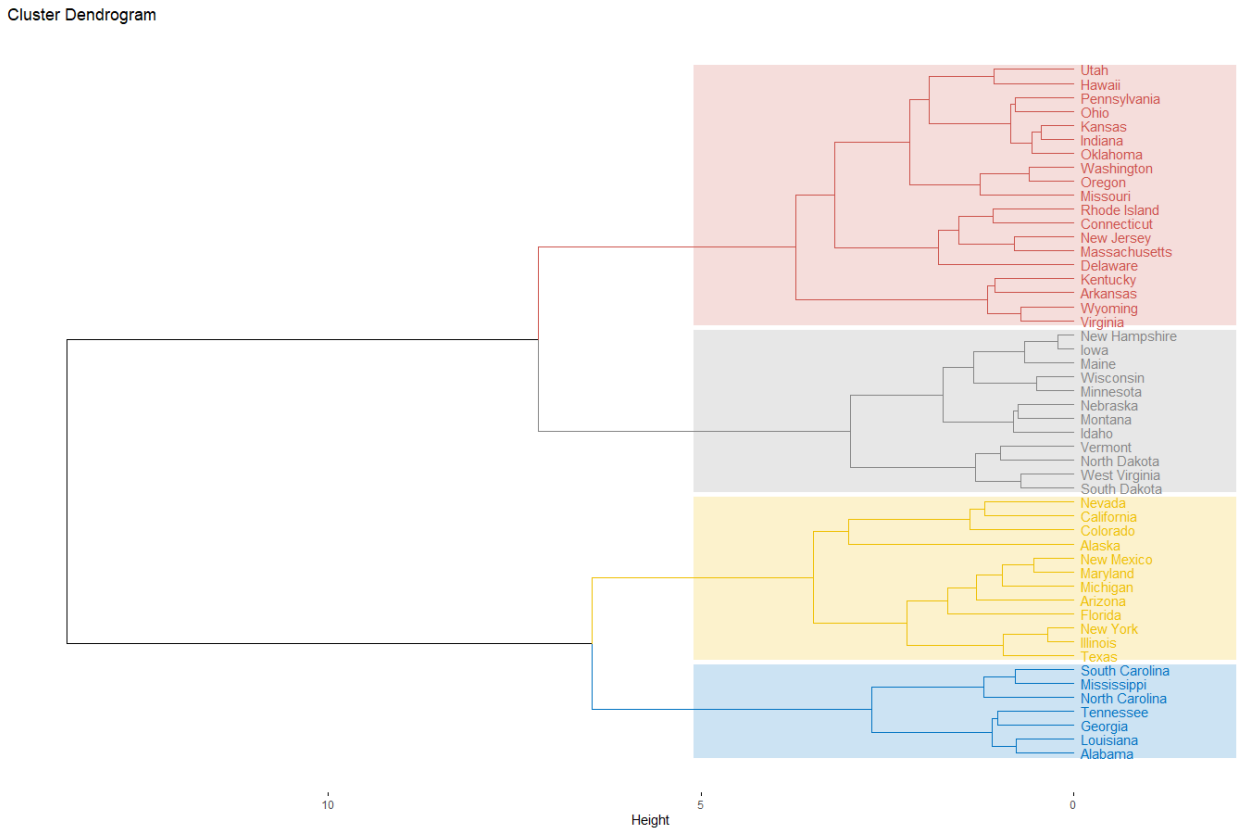
Fill the rectangle with jco colours

Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

```
# Adding colour dendrogram
fviz_dend(hc, k = 4, cex = 0.8, horiz = TRUE, k_colors = "jco", rect = TRUE, rect_border = "jco", rect_fill = TRUE)
```

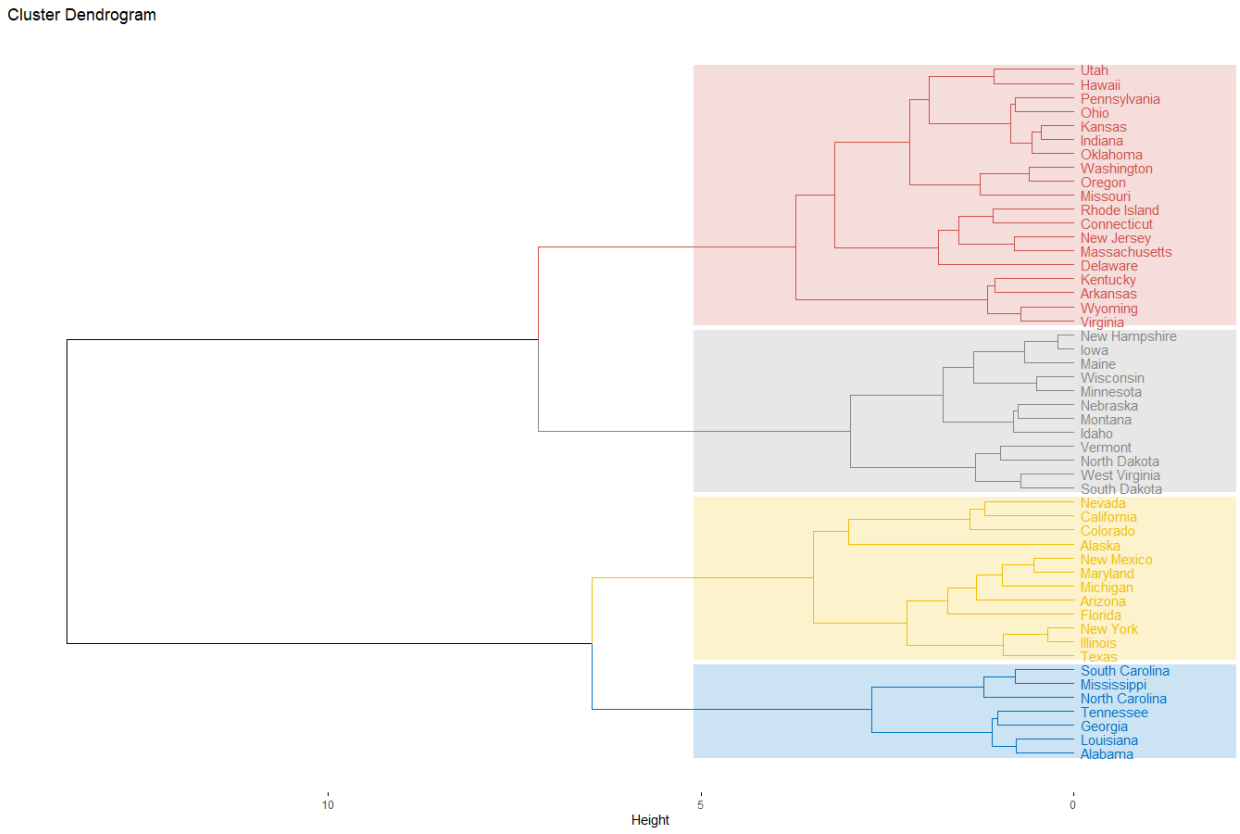


Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

```
# Adding colour dendrogram
fviz_dend(hc, k = 4, cex = 0.8, horiz = TRUE, k_colors = "jco", rect = TRUE, rect_border = "jco", rect_fill = TRUE)
```



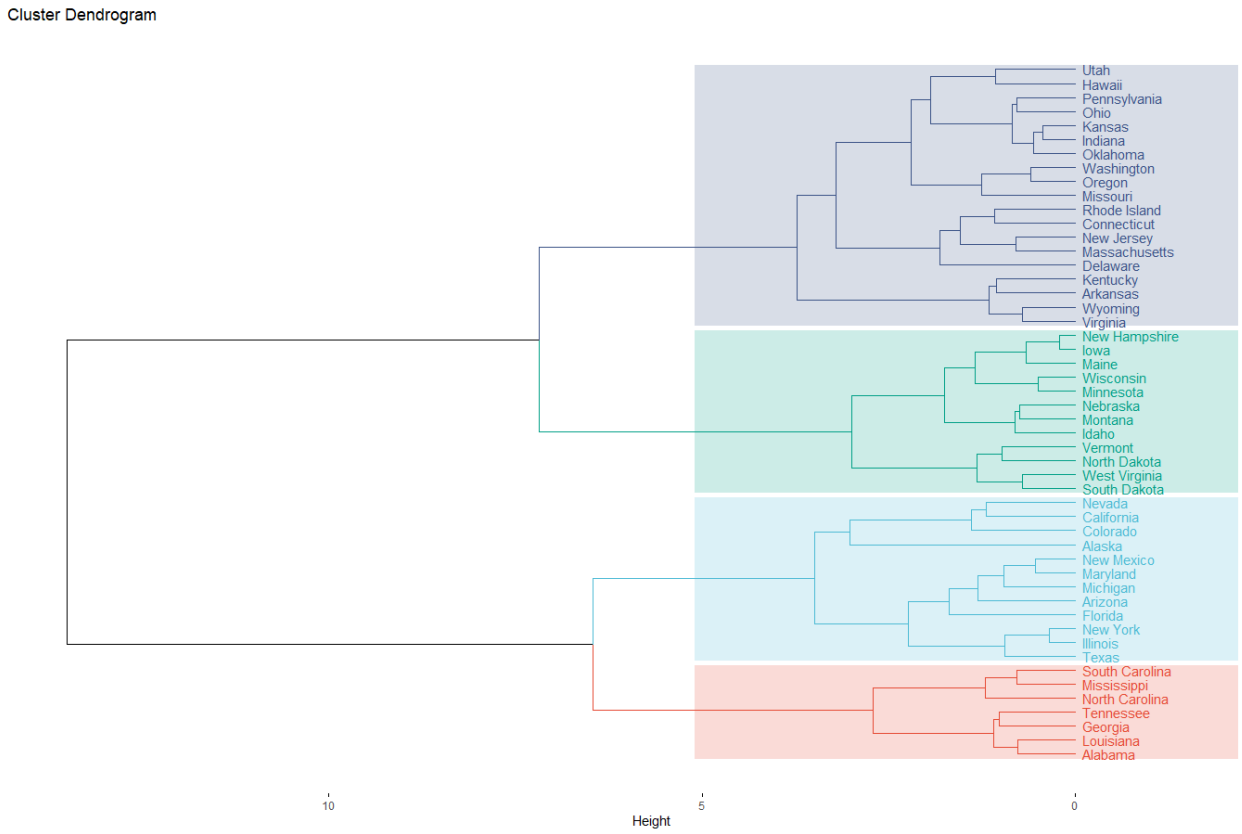
Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

 # Adding colour dendrogram

fviz_dend(hc, k = 4, cex = 0.8, horiz = TRUE, k_colors = "npg", rect = TRUE, rect_border = "npg", rect_fill = TRUE)

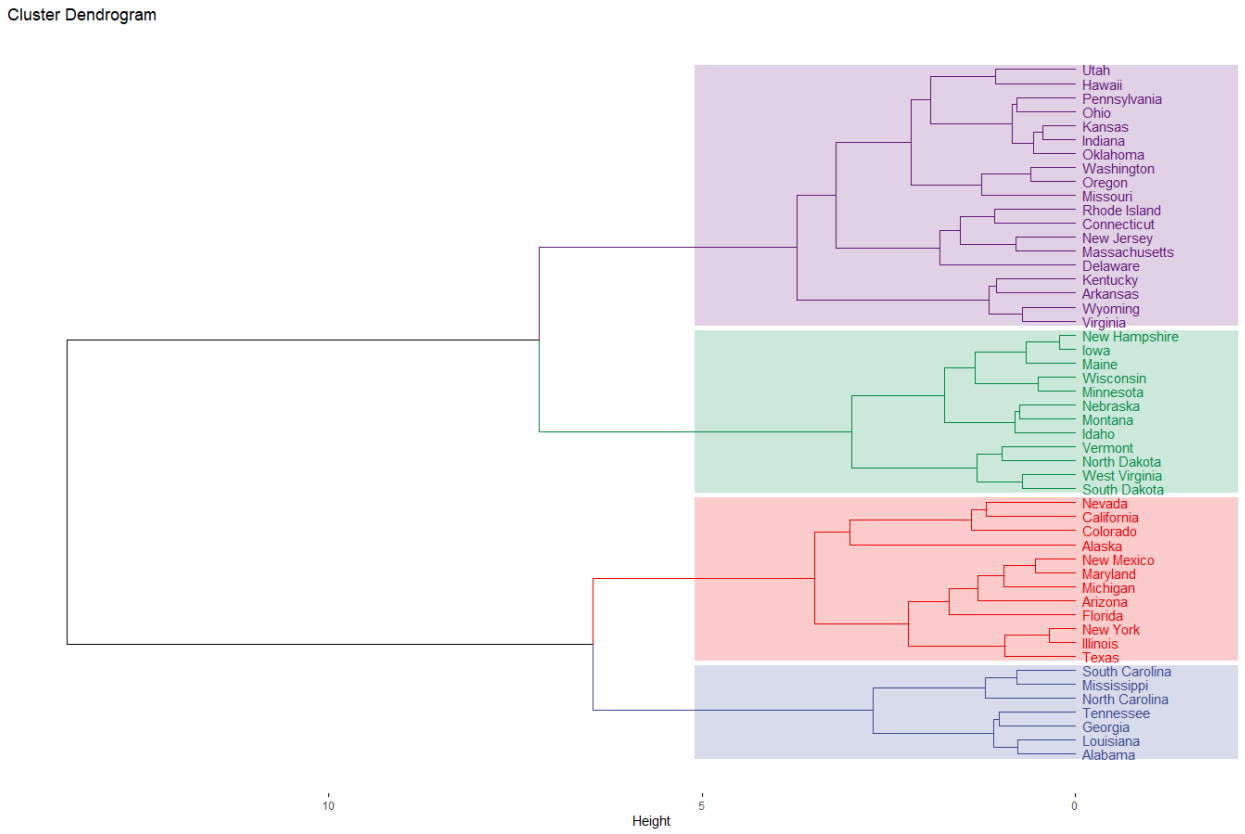


Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

```
# Adding colour dendrogram
fviz_dend(hc, k = 4, cex = 0.8, horiz = TRUE, k_colors = "aaas", rect = TRUE, rect_border = "aaas", rect_fill = TRUE)
```

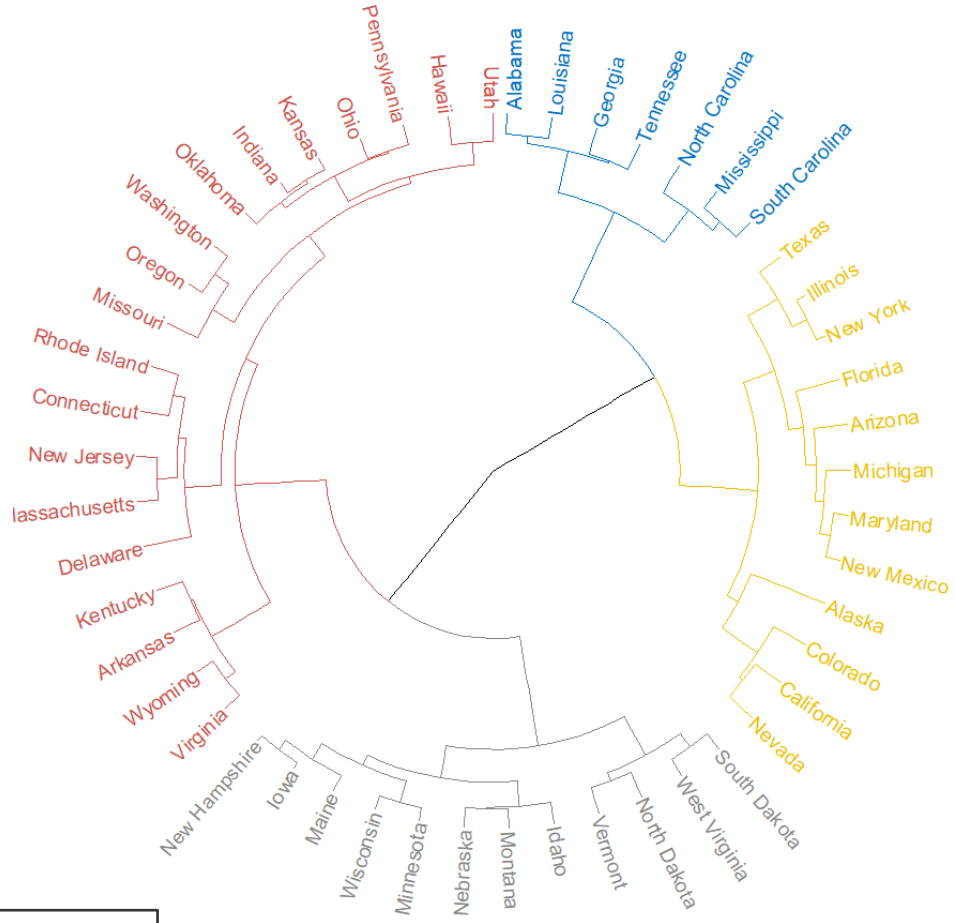


Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

```
# Creating a circular dendrogram chart
fviz_dend(hc, cex = 1, k = 4, k_colors = "jco", type = "circular")
```

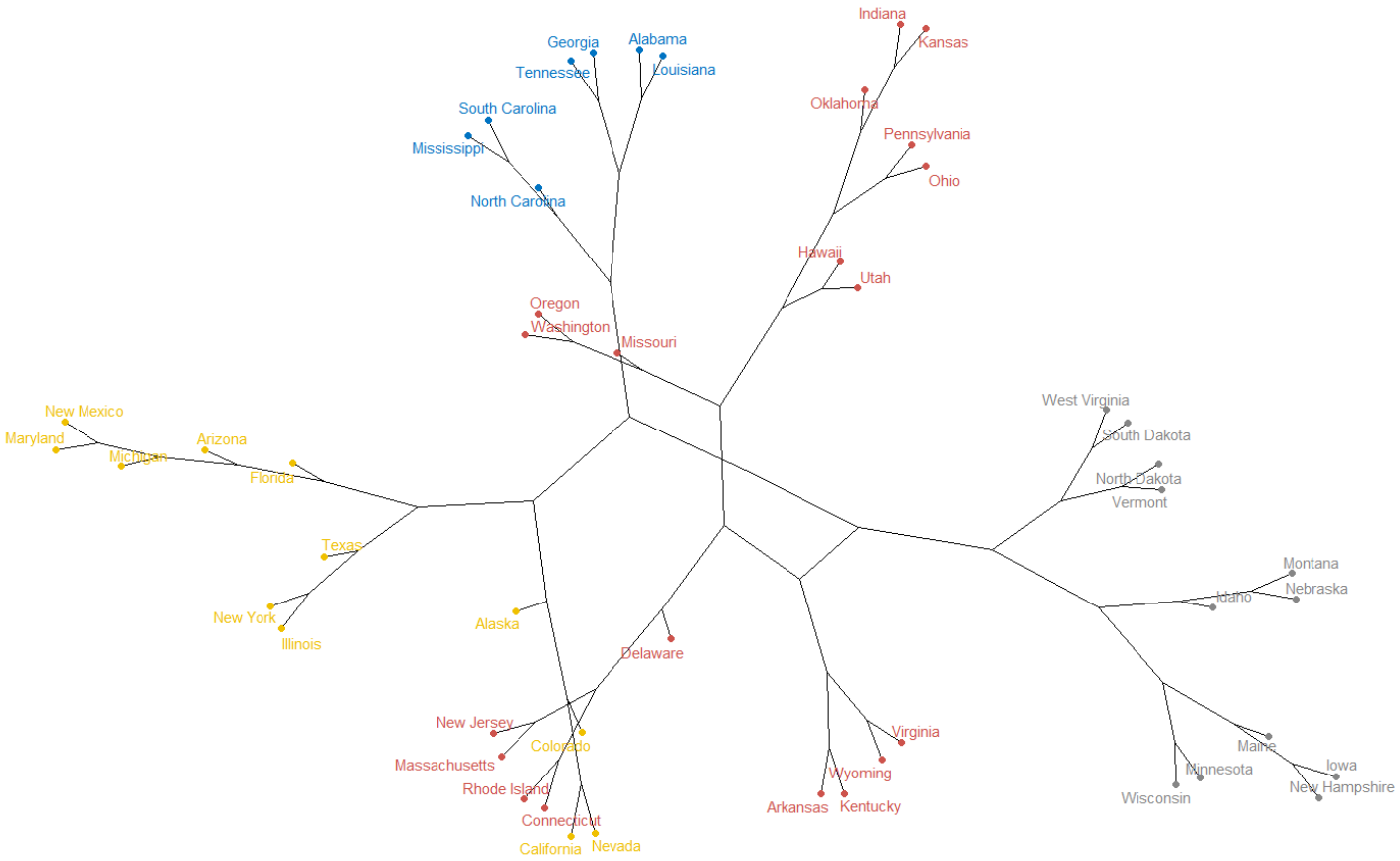


Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

```
# Creating a circular dendrogram chart
fviz_dend(hc, cex = 1, k = 4, k_colors = "jco", type = "phylogenic", repel = TRUE)
```

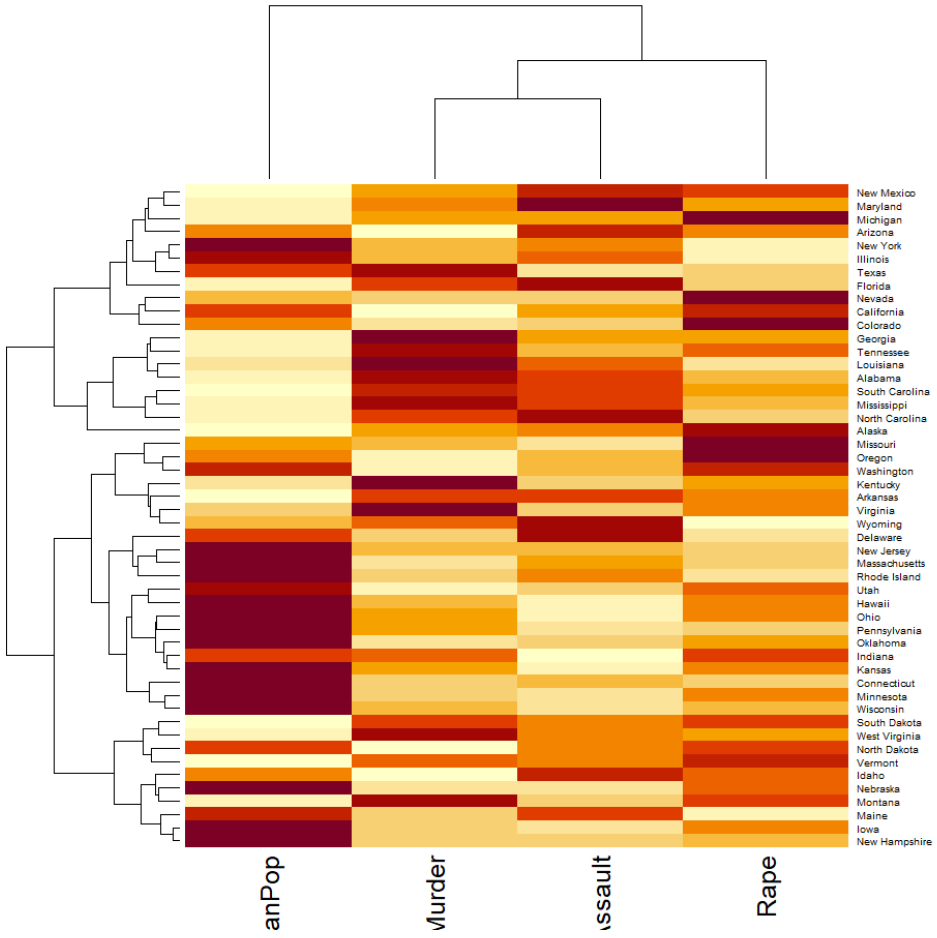


Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

```
# Create a heat map
df<-scale(USArrests)
heatmap(df)
```



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

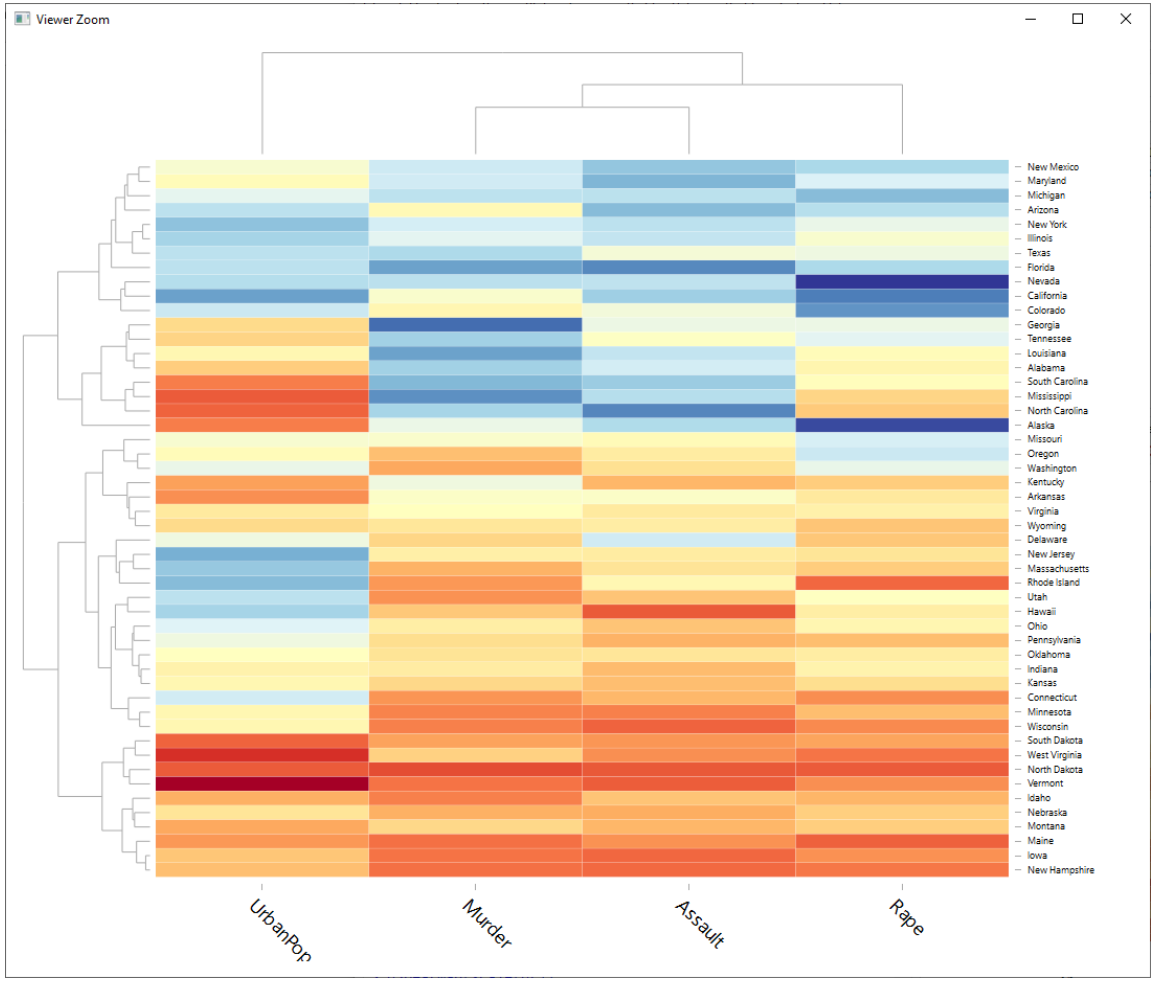
ENHANCING YOUR DENDROGRAMS

Circular plot	Phylogenic tree plot	Heatmap	Interactive heatmap

```
# Install / Load the required packages
install.packages("d3heatmap")
library(d3heatmap)

# Create the interactive heatmap
d3heatmap(df)
```

You can click on the states and / or variables which will highlight parts of the heat map and provide you with cluster values. You can also hover your mouse over parts of the heat map as well for more information. This is what makes this an interactive chart



Source: Kassambara (2017). Practical Guide To Cluster Analysis in R

TASKS

Now repeat these three tasks using the mtcars data file (this is also a base R data set)

Task A: K-means Clustering

Using the base R data file, USArrests, conduct a K-means cluster analysis and produce an appropriate visualisation to present your findings. Your visualisation should clearly display which US states belong to which cluster.

Task B: Hierarchical Clustering

Using the base R data file, USArrests, conduct a hierarchical (agglomerative) cluster analysis and produce a dendrogram to present your findings. You should compare different distance and linkage methods to find the optimal dendrogram for this scenario.

Task C: Enhancing your dendrograms

Recreate your dendrograms (from Task B) as a:

- Circular plot
- Phylogenic tree plot
- Heatmap
- Interactive heatmap