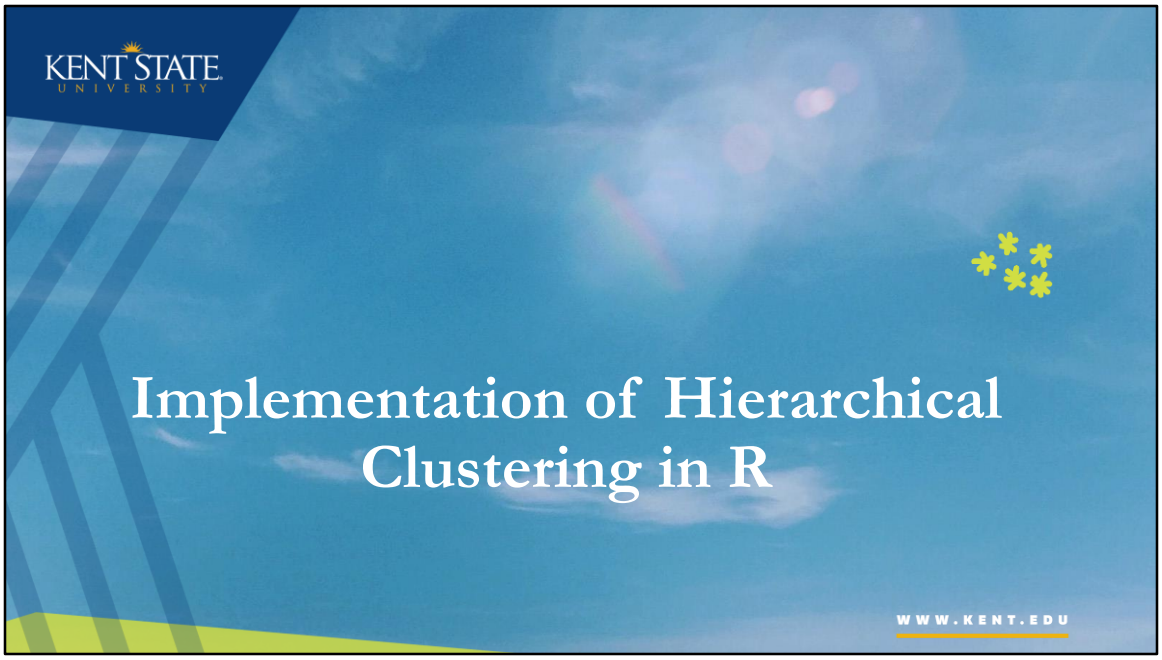


Here, we discuss the implementation of HC in R.

Hierarchical Clustering in R

- There are different functions available in R for computing hierarchical clustering. The commonly used functions are:
- **hclust** [in **stats** package] and **agnes** [in **cluster** package] for agglomerative hierarchical clustering.
- **diana** [in **cluster** package] for divisive hierarchical clustering.

WWW.KENT.EDU

The cluster package provides implementation of Agnes and Diana. You can also find the hclust function in the Stats package.

KENT STATE UNIVERSITY

Dataset

- Here, we'll use the built-in R data set **USArrests**, which contains statistics in arrests per 100,000 residents for **assault**, **murder**, and **rape** in each of the 50 US states in 1973.
- It includes also the percent of the population living in **urban** areas in each state.
- The objective is to cluster different states based on crime attributes and urban population percentages. In other words, we want to form groups of states that are similar with respect to crime attributes and urban population rate.

WWW.KENT.EDU

We will use an in-bullt data set to illustrate HC.

Example: US Arrest Dataset I

```
df <- USArrests
df <- na.omit(df) # Remove NA (missing) values
head(df)          # Examine the dataset
```

```
##            Murder Assault UrbanPop Rape
## Alabama      13.2     236       58 21.2
## Alaska       10.0     263       48 44.5
## Arizona       8.1     294       80 31.0
## Arkansas      8.8     190       50 19.5
## California    9.0     276       91 40.6
## Colorado      7.9     204       78 38.7
```


We have four variables that we take into consideration during clustering. The state will act as the row names.

Example: US Arrest Dataset II

```
#As we don't want the clustering algorithm to depend to an
# arbitrary variable unit, we start by
#scaling/standardizing the data

df <- scale(df)
head(df) #re-examine the scaled data
##               Murder   Assault   UrbanPop         Rape
## Alabama    1.24256408 0.7828393 -0.5209066 -0.003416473
## Alaska     0.50786248 1.1068225 -1.2117642  2.484202941
## Arizona    0.07163341 1.4788032  0.9989801  1.042878388
## Arkansas   0.23234938 0.2308680 -1.0735927 -0.184916602
## California 0.27826823 1.2628144  1.7589234  2.067820292
## Colorado   0.02571456 0.3988593  0.8608085  1.864967207
```


As all distance measures are affected by scale, we first standardize the data.

KENT STATE UNIVERSITY

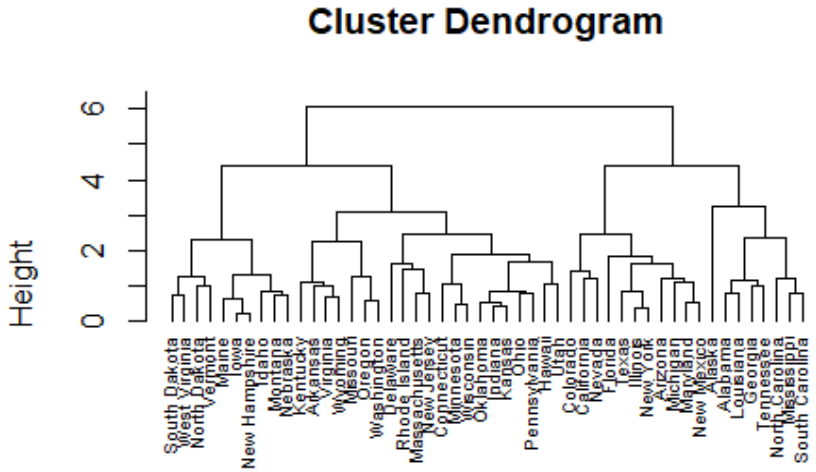
Example: US Arrest Dataset III

```
# Dissimilarity matrix
d <- dist(df, method = "euclidean")

# Hierarchical clustering using Complete Linkage
hc1 <- hclust(d, method = "complete" )

# Plot the obtained dendrogram
plot(hc1, cex = 0.6, hang = -1)
```

WWW.KENT.EDU

We now apply the hclust function from the Stats package. Remember to install the package first if necessary. The plot function provides the dendrogram.

Example: US Arrest Dataset IV



The height (Y axis) provides the distance between clusters. If we use a height of 3, how many clusters will you have?

KENT STATE UNIVERSITY

Using agnes() function I

- Alternatively, we can use the **agnes( )** function. The function is part of **cluster** package.
- **agnes ( )** and **hclust ( )** functions behave very similarly; however, with the **agnes ( )** function you can also get the **agglomerative coefficient.**
- Agglomerative coefficient measures the amount of clustering structure found (values closer to 1 suggest strong clustering structure).

WWW.KENT.EDU

Let us now implement HC using the Agnes function. We can also get a numerical measure of the strength of the clustering structure by calculating the agglomerative coefficient.

Using agnes() function II

```
library(cluster )

df <- USArrests

# Compute with agnes and with different linkage methods
hc_single <- agnes(df, method = "single")
hc_complete <- agnes(df, method = "complete")
hc_average <- agnes(df, method = "average")

# Compare Agglomerative coefficients
print(hc_single$ac)

## [1] 0.6625233

print(hc_complete$ac)

## [1] 0.9498031

print(hc_average$ac)

## [1] 0.9073773

pltree(hc_complete, cex = 0.6, hang = -1, main = "Dendrogram of agnes")
```

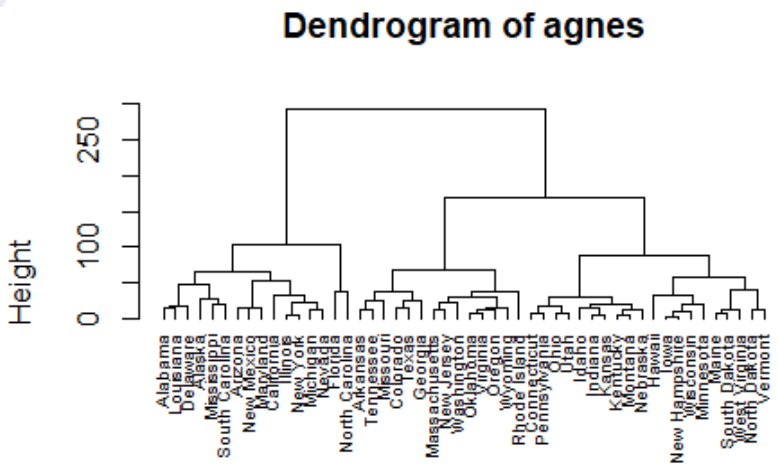
Best linkage method

WWW.KENT.EDU

Notice here that using the ”Complete” linkage provides the strongest clustering structure. Can you re-run this using the “Wards” approach?

KENT STATE UNIVERSITY

Using agnes() function III



WWW.KENT.EDU

Why are the Height values here so different from when we first ran HC using the hclust function?

What did we forget to do?

If we use a height of 100, how many clusters will we have?

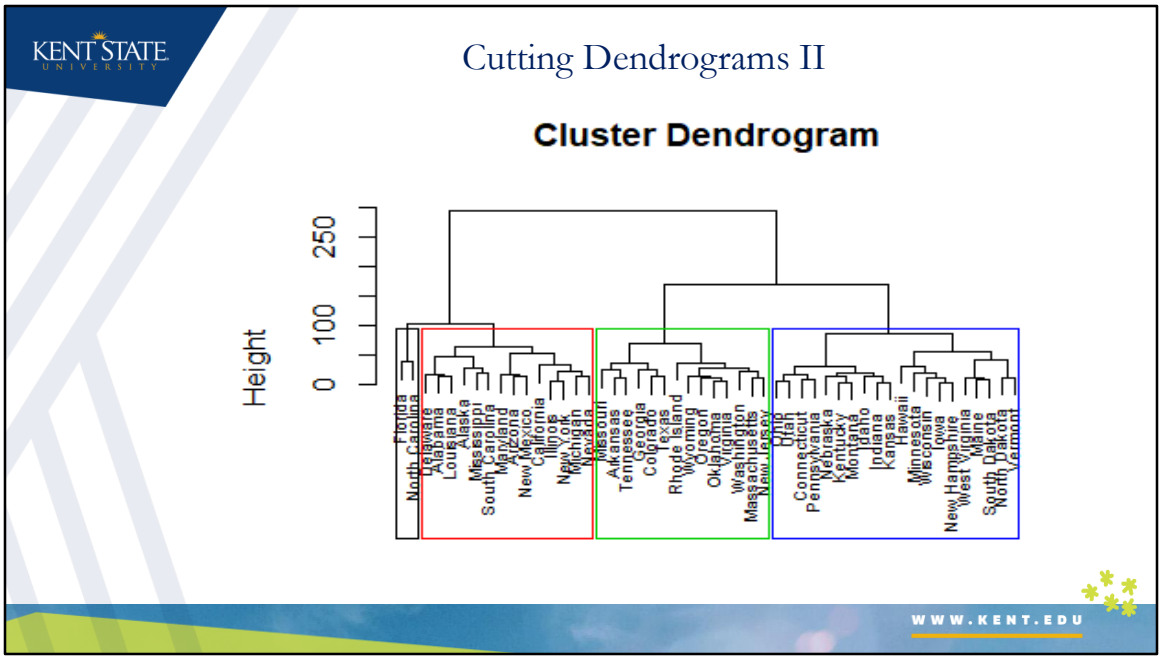
Cutting Dendrograms I

- It's also possible to draw the dendrogram with a border around the 4 clusters.

- The argument border is used to specify the border colors for the rectangles:

```
df <- USArrests
d <- dist(df, method = "euclidean")

# compute divisive hierarchical clustering
hc_complete <- hclust(d,method = "complete")

# plot dendrogram
plot(hc_complete, cex = 0.6)
rect.hclust(hc_complete, k = 4, border = 1:4)
```


Let us now output a dendrogram and draw a box around the clusters. We will assume that we have four clusters based on our selection of height cutoff. Remember the height indicates the distance between clusters.



This provides a clearer depiction of the 4 clusters.

KENT STATE UNIVERSITY

Using diana() function

- The R function diana provided by the cluster package allows us to perform divisive hierarchical clustering.
- diana works similar to agnes; however, there is no method to provide.

```
# compute divisive hierarchical clustering
hc_diana <- diana(df)

# Divise coefficient; amount of clustering structure found
hc_diana$dc

## [1] 0.9464692

# plot dendrogram
pltree(hc_diana, cex = 0.6, hang = -1, main = "Dendrogram of diana")
```

WWW.KENT.EDU

Diana works similarly to Agnes. The output is similar.

This concludes our module on HC and on Agnes and Diana.