AmanChauhan10oct.R

amssr

2024-10-10

```
library(readr)
```

```
## Warning: package 'readr' was built under R version 4.3.3
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.3.3
```

```
## 
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
## 
##     filter, lag
```

```
## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union
```

```
#Load the Mall_Customer.csv dataset from previous lab.
df <- read.csv("C:/Users/amssr/Desktop/Mall_Customers.csv")
head(df,6)
```

```
##   CustomerID  Genre Age Annual.Income..k.. Spending.Score..1.100.
## 1          1   Male  19                 15                     39
## 2          2   Male  21                 15                     81
## 3          3 Female  20                 16                      6
## 4          4 Female  23                 16                     77
## 5          5 Female  31                 17                     40
## 6          6 Female  22                 17                     76
```

```
tail(df,6)
```

```
##     CustomerID  Genre Age Annual.Income..k.. Spending.Score..1.100.
## 195        195 Female  47                120                     16
## 196        196 Female  35                120                     79
## 197        197 Female  45                126                     28
## 198        198   Male  32                126                     74
## 199        199   Male  32                137                     18
## 200        200   Male  30                137                     83
```

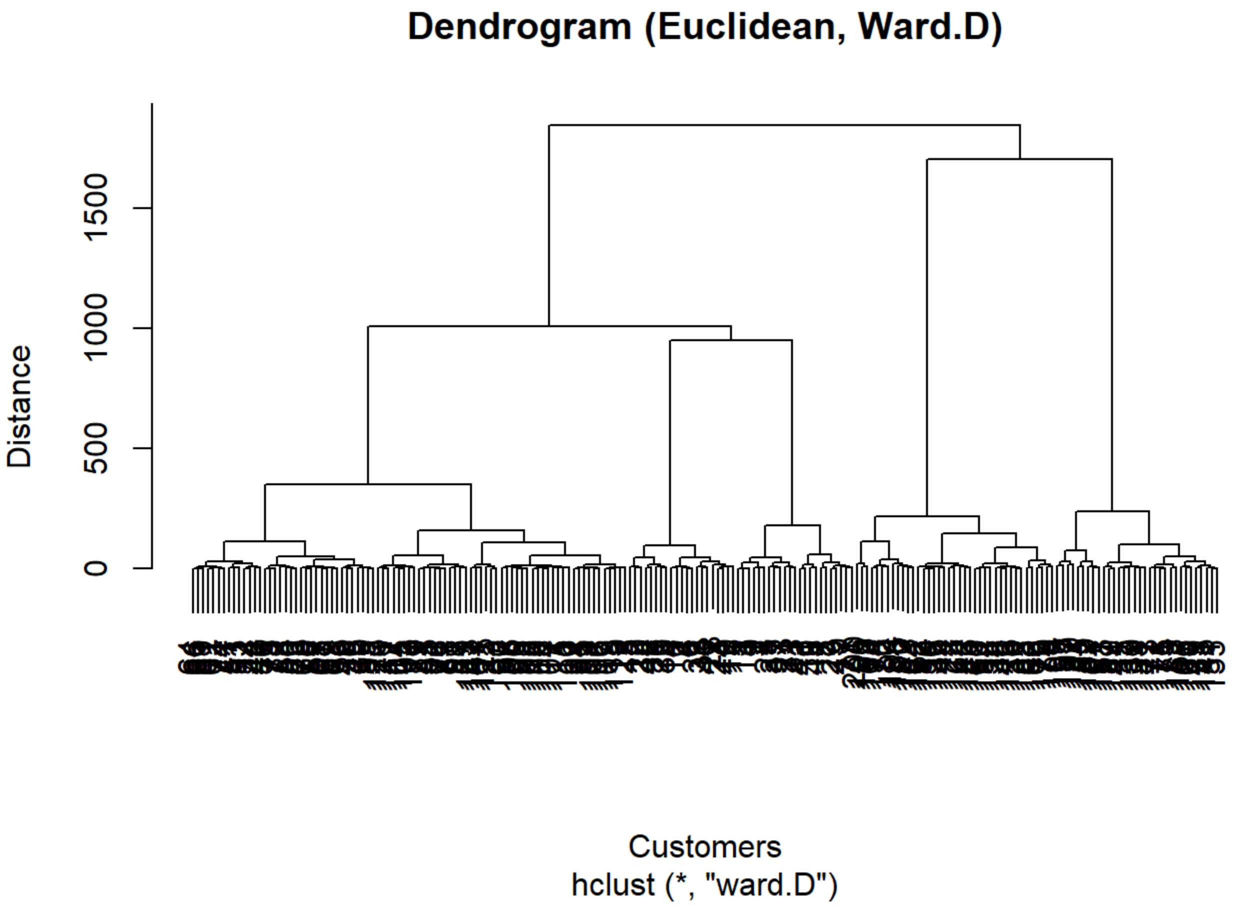
```
#Apply Data Pre-processing and data cleaning
#No need for data cleaning as already there will be no omitted data or NA alue in the data
#Apply Statistical summary
summary(df)
```

```
##    CustomerID        Genre                Age        Annual.Income..k..
##  Min.   :  1.00   Length:200         Min.   :18.00   Min.   : 15.00    
##  1st Qu.: 50.75   Class :character   1st Qu.:28.75   1st Qu.: 41.50    
##  Median :100.50   Mode  :character   Median :36.00   Median : 61.50    
##  Mean   :100.50                      Mean   :38.85   Mean   : 60.56    
##  3rd Qu.:150.25                      3rd Qu.:49.00   3rd Qu.: 78.00    
##  Max.   :200.00                      Max.   :70.00   Max.   :137.00    
##  Spending.Score..1.100.
##  Min.   : 1.00         
##  1st Qu.:34.75         
##  Median :50.00         
##  Mean   :50.20         
##  3rd Qu.:73.00         
##  Max.   :99.00         
```

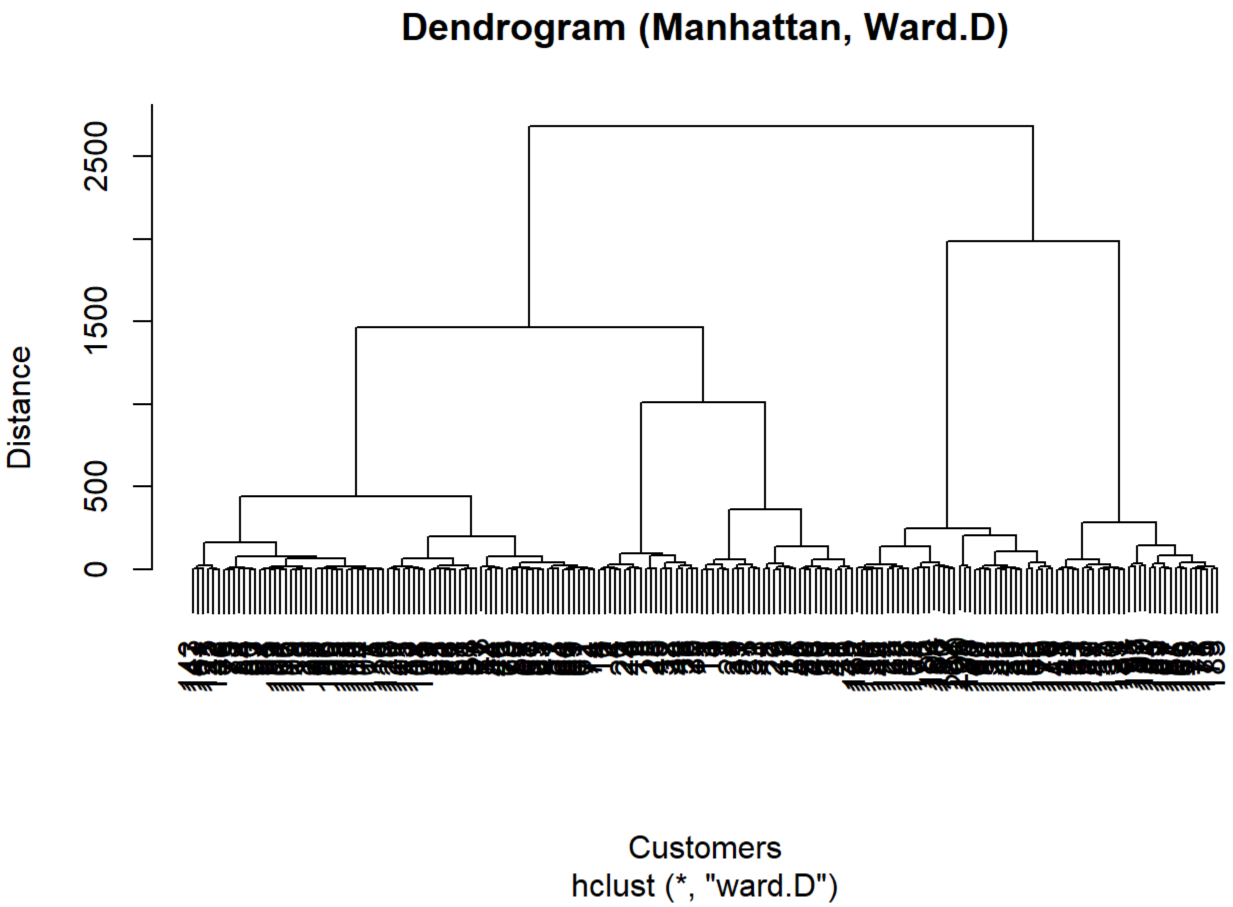
```
#Store the Annual Income and Spending score in an object 'dataset'
dataset <- df[, c(4, 5)]
head(dataset,5)
```

```
##   Annual.Income..k.. Spending.Score..1.100.
## 1                 15                     39
## 2                 15                     81
## 3                 16                      6
## 4                 16                     77
## 5                 17                     40
```

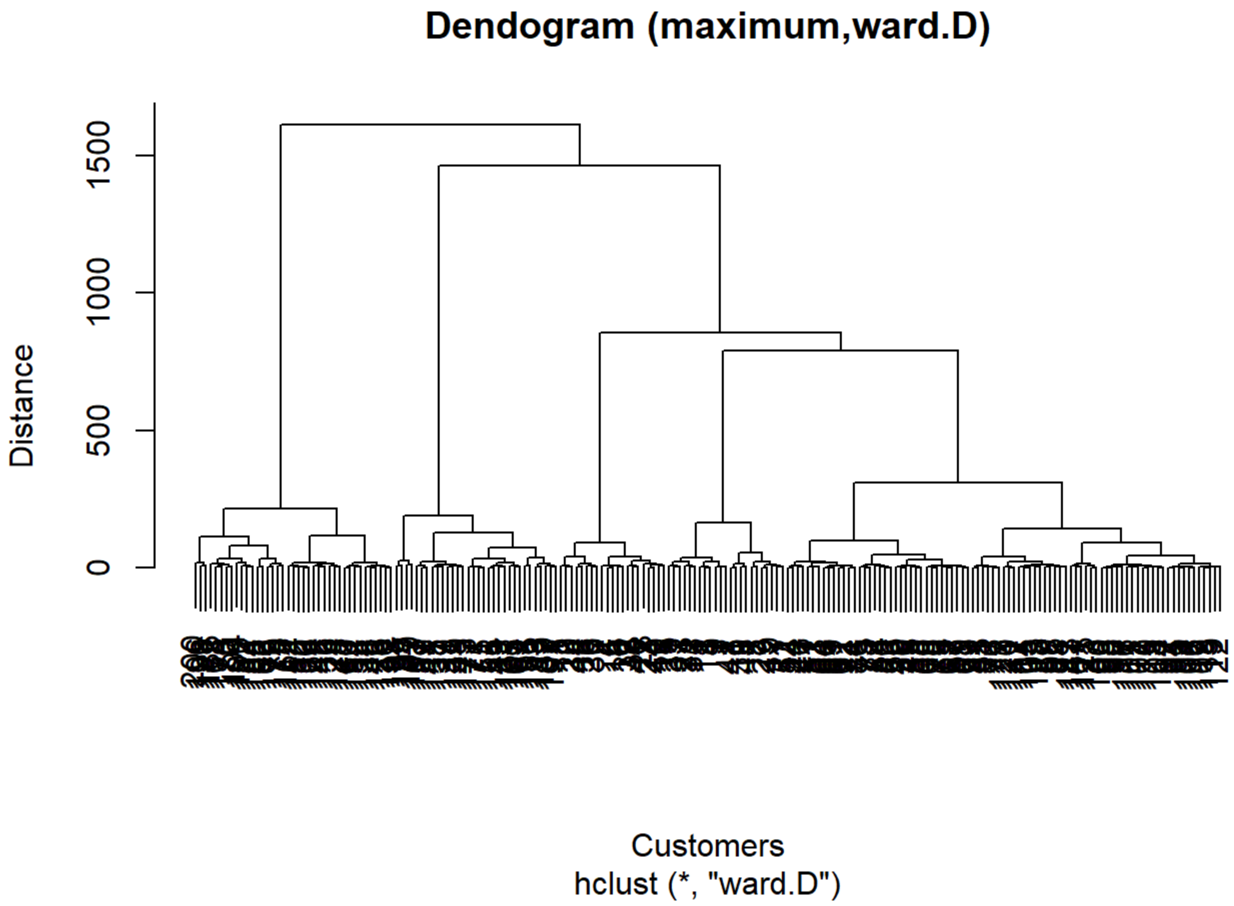
```
#Compute the distance matrix using dist() - apply the euclidean method and display the Matrix ta
ble.
distance_matrix_euclidean <- dist(dataset, method = "euclidean")
# Display the Euclidean distance matrix
#print(as.matrix(distance_matrix_euclidean))
#using the dendrogram to find the optimal cluster - use hclust(). To minimize within the cluster
use method as 'ward.D'
# Perform hierarchical clustering using Ward.D method
hclust_euclidean <- hclust(distance_matrix_euclidean, method = "ward.D")
hclust_euclidean<- hclust(distance_matrix_euclidean,method = "ward.D")
plot(hclust_euclidean, main = "Dendrogram (Euclidean, Ward.D)", xlab = "Customers", ylab = "Dist
ance")
```



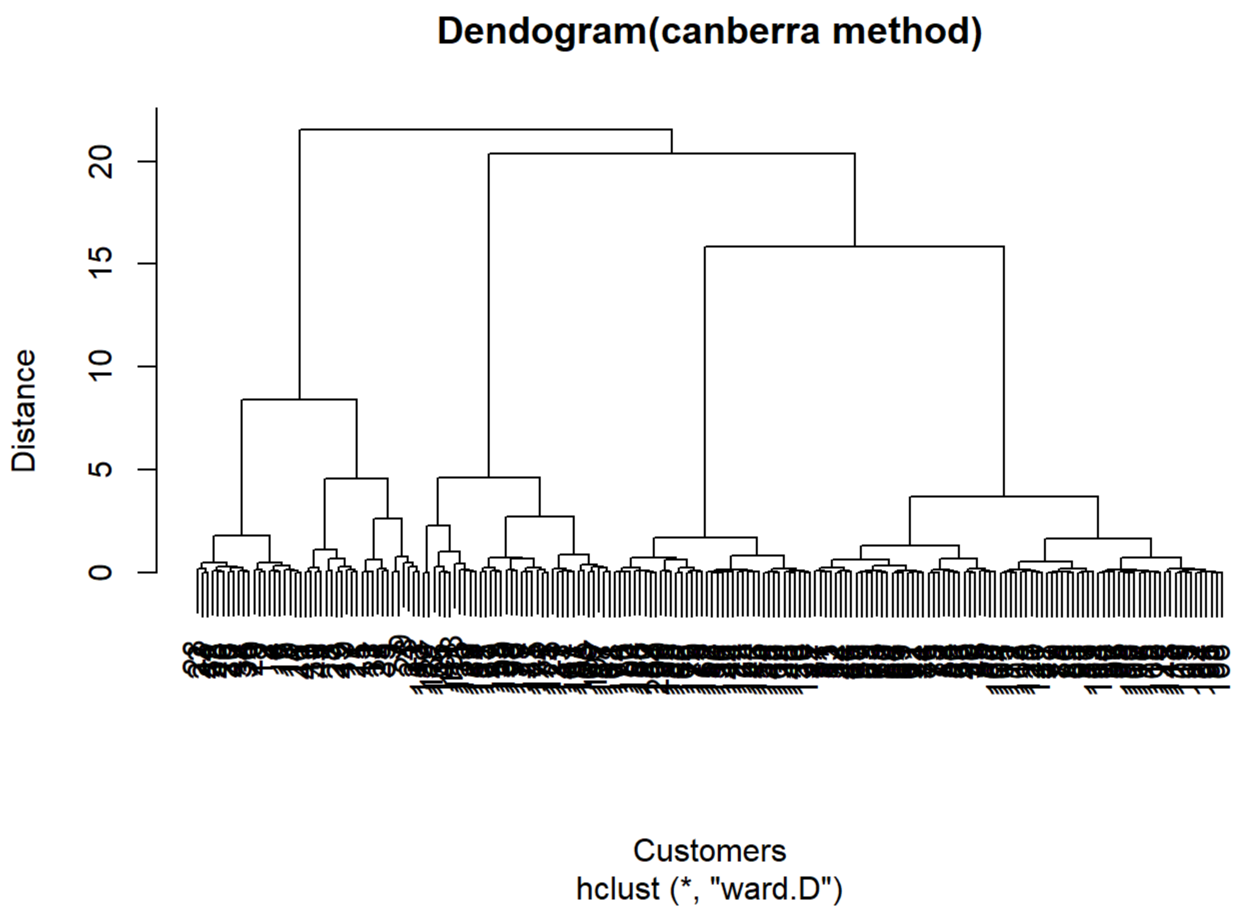
```
#Using basic plot function visualize dendrogram
#Compute the distance matrix using dist() - apply the manhattan method and display the Matrix ta
ble.
distance_matrix_manhattan <- dist(dataset, method = "manhattan")

# Display the Manhattan distance matrix
#print(as.matrix(distance_matrix_manhattan))
# Perform hierarchical clustering using Ward.D method
hclust_manhattan <- hclust(distance_matrix_manhattan, method = "ward.D")
#using the dendrogram to find the optimal cluster - use hclust(). To minimize within the cluster
use method as 'ward.D'
#Using basic plot function visualize dendrogram
plot(hclust_manhattan, main = "Dendrogram (Manhattan, Ward.D)", xlab = "Customers", ylab = "Dist
ance")
```



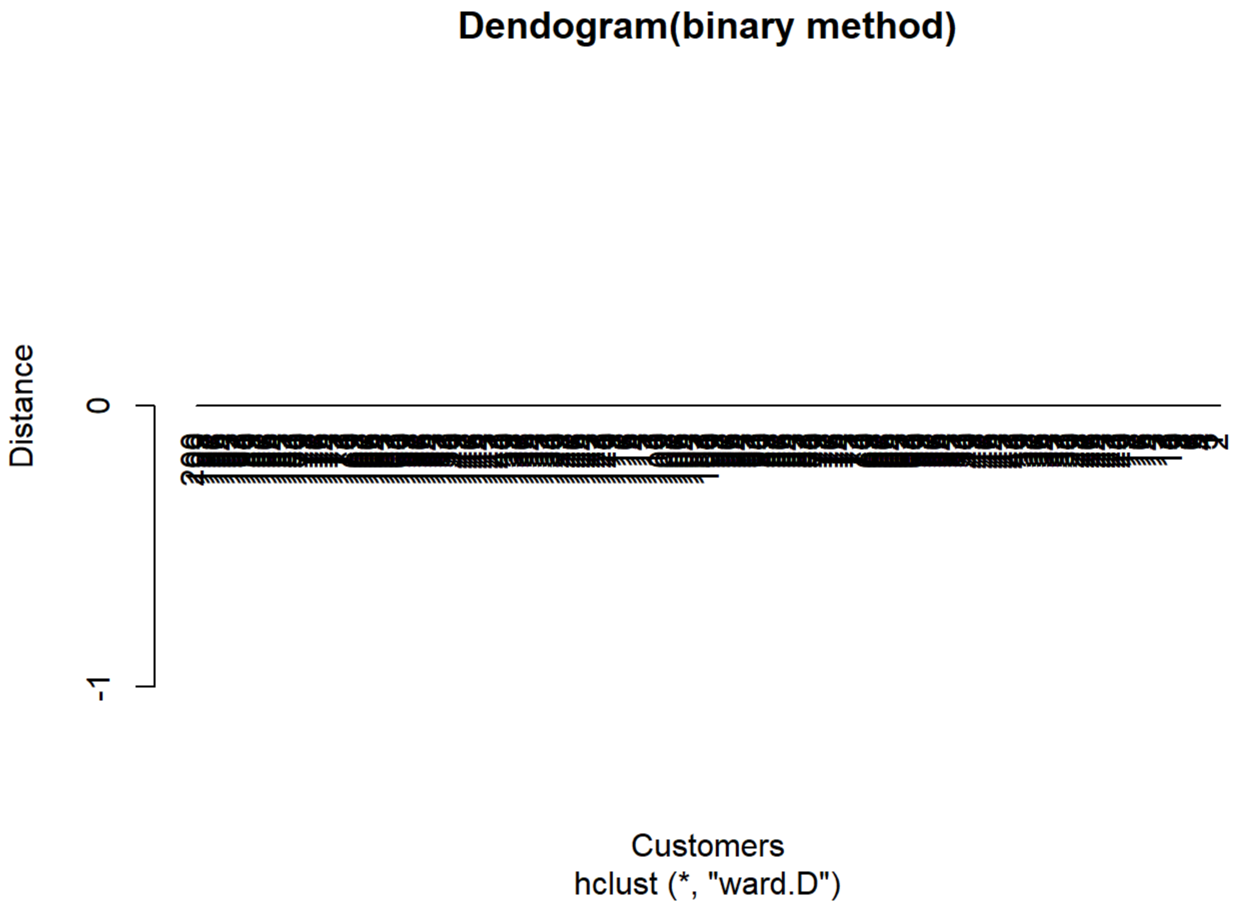
```
#Compute the distance matrix using dist() - apply the maximum method and display the Matrix tabl
e.
distance_matrix_maximum <- dist(dataset, method="maximum")
#print(as.matrix(distance_matrix_maximum))
#using the dendrogram to find the optimal cluster - use hclust(). To minimize within the cluster
use method as 'ward.D'
hclust_maximum<-hclust(distance_matrix_maximum,method="ward.D")
#Using basic plot function visualize dendrogram
plot(hclust_maximum, main = "Dendogram (maximum,ward.D)",xlab="Customers",ylab="Distance")
```



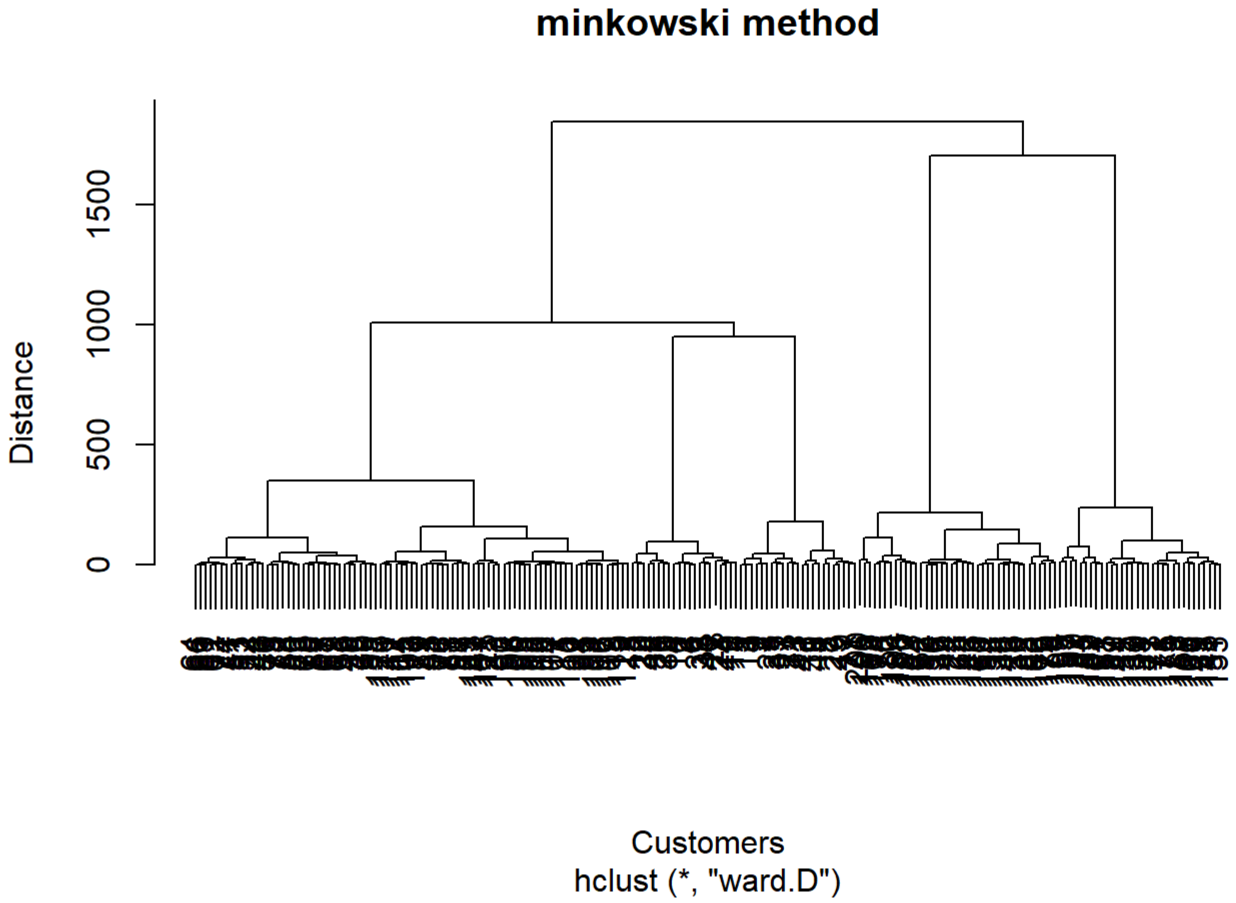
```
#Compute the distance matrix using dist() - apply the canberra distance method and display the Matrix table.
distance_matrix_canberra <- dist(dataset,method="canberra")
#print(as.matrix(distance_matrix_canberra))
#using the dendrogram to find the optimal cluster - use hclust(). To minimize within the cluster use method as 'ward.D'
hclust_canberra<-hclust(distance_matrix_canberra,method = "ward.D")
#Using basic plot function visualize dendrogram
plot(hclust_canberra,main="Dendogram(canberra method)",xlab="Customers",ylab="Distance")
```



```
#Compute the distance matrix using dist() - apply the binary distance method and display the Matrix table.
distance_matrix_binary <- dist(dataset,method="binary")
#print(as.matrix(distance_matrix_binary))
#using the dendrogram to find the optimal cluster - use hclust(). To minimize within the cluster use method as 'ward.D'
hclust_binary<-hclust(distance_matrix_binary,method = "ward.D")
#Using basic plot function visualize dendrogram
plot(hclust_binary,main="Dendogram(binary method)",xlab="Customers",ylab="Distance")
```



```
#Compute the distance matrix using dist() - apply the Minkowski distance method and display the
Matrix table.
distance_matrix_minkowski <- dist(dataset, method="minkowski")
#print(as.matrix(distance_matrix_minkowski))
#using the dendrogram to find the optimal cluster - use hclust(). To minimize within the cluster
use method as 'ward.D'
hclust_minkowski<-hclust(distance_matrix_minkowski,method="ward.D")
#Using basic plot function visualize dendrogram
plot(hclust_minkowski,main="minkowski method",xlab="Customers",ylab="Distance")
```


minkowski method
Distance
0
500
1000
1500
Customers
hclust (*, "ward.D")