Project 3

ISAIAH THOMPSON OCANSEY

PART II: Computer Project

(i) Read the data into R. List the missing rate (in percentage) for each variable.

```
# Importing the data
dat0<-read.csv("C:/Users/thomo/OneDrive/Desktop/Data Mining/HMEQ.csv")
dim(dat0); head(dat0)
```

```
## [1] 5960   13

##   BAD LOAN MORTDUE  VALUE  REASON    JOB  YOJ DEROG DELINQ     CLAGE NINQ CLNO
## 1   1 1100   25860  39025 HomeImp  Other 10.5     0      0  94.36667    1    9
## 2   1 1300   70053  68400 HomeImp  Other  7.0     0      2 121.83333    0   14
## 3   1 1500   13500  16700 HomeImp  Other  4.0     0      0 149.46667    1   10
## 4   1 1500      NA     NA    <NA>   <NA>   NA    NA     NA        NA   NA   NA
## 5   0 1700   97800 112000 HomeImp Office  3.0     0      0  93.33333    0   14
## 6   1 1700   30548  40320 HomeImp  Other  9.0     0      0 101.46600    1    8
##    DEBTINC
## 1       NA
## 2       NA
## 3       NA
## 4       NA
## 5       NA
## 6 37.11361
```

The data (dat0) has 5960 observations with 13 variables

```
## Estimating the missing values rate.
colMeans(is.na(dat0))
```

```
##        BAD       LOAN    MORTDUE      VALUE     REASON        JOB        YOJ
## 0.00000000 0.00000000 0.08691275 0.01879195 0.04228188 0.04681208 0.08640940
##      DEROG     DELINQ      CLAGE       NINQ       CLNO    DEBTINC
## 0.11879195 0.09731544 0.05167785 0.08557047 0.03724832 0.21258389
```

From the above, it can be observed that BAD and LOAN has no missing percentages, MORTDUE has 8.67% of missing values ,VALUE has 1.87% of missing values,REASON has 4.22% of missing values,JoB has 4.68% of missing values,YOJ has 8.64% of missing values,DEROG has 11.88% of missing values,DELINQ has 9.7% of missing values, CLAGE has 5.17% of missing values,NINQ has 8.56% of missing values,CLNO has 3.72% of missing values and DEBTINC has 21.26% of missing values.

(ii) DATA cleaning.

a)

```
#Replacing 'NA' with 'Unknown'
dat0$REASON[which(is.na(dat0$REASON))]<-"Unknown"
dat0$JOB[which(is.na(dat0$JOB))]<-"Unknown"
table(dat0$JOB, useNA = "ifany")
```

```
##
##     Mgr  Office   Other ProfExe   Sales    Self Unknown
##     767     948    2388    1276     109     193     279
```

The 279 missing values for JOB has been replaced by the default contatnt "Unknown".

```
table(dat0$REASON, useNA = "ifany")
```

```
##
## DebtCon HomeImp Unknown
##    3928    1780     252
```

The 252 missing values of REASON are replaced with unknown.

ii)

B) Natural Logarithm transformation on the following variables: LOAN, VALUE, MORTDUE, YOJ, and CLAGE. ===

```
summary(dat0[, c("LOAN", "MORTDUE", "VALUE", "YOJ", "CLAGE")])
```

```
##       LOAN          MORTDUE            VALUE              YOJ
##  Min.   : 1100   Min.   :  2063   Min.   :  8000   Min.   : 0.000
##  1st Qu.:11100   1st Qu.: 46276   1st Qu.: 66076   1st Qu.: 3.000
##  Median :16300   Median : 65019   Median : 89236   Median : 7.000
##  Mean   :18608   Mean   : 73761   Mean   :101776   Mean   : 8.922
##  3rd Qu.:23300   3rd Qu.: 91488   3rd Qu.:119824   3rd Qu.:13.000
##  Max.   :89900   Max.   :399550   Max.   :855909   Max.   :41.000
##                  NA's   :518      NA's   :112      NA's   :515
##      CLAGE
##  Min.   :   0.0
##  1st Qu.: 115.1
##  Median : 173.5
##  Mean   : 179.8
##  3rd Qu.: 231.6
##  Max.   :1168.2
##  NA's   :308
```

We perform summary statistic above to see the minimum values of the above variables so we could add 1 to variables that has zero as minimum values since log 0 is undefined. It can be observed from the summary statistic that the variables YOJ and CLAGE have minimum values as 0 so we add 1 to aid the log transformation.

```
# adding 1 to the variables 'YOJ' and 'CLAGE'
dat0[, c(7, 10)] <- dat0[, c(7, 10)] + 1
```

```
summary(dat0[, c("LOAN", "MORTDUE", "VALUE", "YOJ", "CLAGE")])
```

```
##       LOAN          MORTDUE           VALUE             YOJ
##  Min.   : 1100   Min.   :  2063   Min.   :  8000   Min.   : 1.000
##  1st Qu.:11100   1st Qu.: 46276   1st Qu.: 66076   1st Qu.: 4.000
##  Median :16300   Median : 65019   Median : 89236   Median : 8.000
##  Mean   :18608   Mean   : 73761   Mean   :101776   Mean   : 9.922
##  3rd Qu.:23300   3rd Qu.: 91488   3rd Qu.:119824   3rd Qu.:14.000
##  Max.   :89900   Max.   :399550   Max.   :855909   Max.   :42.000
##                  NA's   :518      NA's   :112      NA's   :515
##      CLAGE
##  Min.   :   1.0
##  1st Qu.: 116.1
##  Median : 174.5
##  Mean   : 180.8
##  3rd Qu.: 232.6
##  Max.   :1169.2
##  NA's   :308
```

Since zero is no longer the minimum value of the variables;YOJ and CLAGE, we will proceed with the log transformation.

```
# taking natural log of the variables LOAN, VALUE, MORTDUE, YOJ, and CLAGE
dat0[, c(2, 3, 4, 7, 10)] <- apply(dat0[, c(2, 3, 4, 7, 10)], 2, FUN = log)
head(dat0[, c(2, 3, 4, 7, 10)])
```

```
##       LOAN   MORTDUE     VALUE      YOJ    CLAGE
## 1 7.003065 10.160453 10.571958 2.442347 4.557729
## 2 7.170120 11.157007 11.133128 2.079442 4.810828
## 3 7.313220  9.510445  9.723164 1.609438 5.013742
## 4 7.313220        NA        NA       NA       NA
## 5 7.438384 11.490680 11.626254 1.386294 4.546835
## 6 7.438384 10.327054 10.604603 2.302585 4.629531
```

ii)

C) Impute all the remaining values with an appropriate imputation procedure of your own choice. In our case, We will be using the mice function for the imputation. Before we perform the imputation, we will exclude the target variable “BAD”

```
library(dplyr)
```

```
##
## 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
```

```r
dat_1<-dat0%>% select(-BAD)
head(dat_1)
```

```
##       LOAN   MORTDUE     VALUE  REASON     JOB      YOJ DEROG DELINQ    CLAGE
## 1 7.003065 10.160453 10.571958 HomeImp   Other 2.442347     0      0 4.557729
## 2 7.170120 11.157007 11.133128 HomeImp   Other 2.079442     0      2 4.810828
## 3 7.313220  9.510445  9.723164 HomeImp   Other 1.609438     0      0 5.013742
## 4 7.313220        NA        NA Unknown Unknown       NA    NA     NA       NA
## 5 7.438384 11.490680 11.626254 HomeImp  Office 1.386294     0      0 4.546835
## 6 7.438384 10.327054 10.604603 HomeImp   Other 2.302585     0      0 4.629531
##   NINQ CLNO  DEBTINC
## 1    1    9       NA
## 2    0   14       NA
## 3    1   10       NA
## 4   NA   NA       NA
## 5    0   14       NA
## 6    1    8 37.11361
```

```r
# impute values for all missing values using the package MICE
library(mice)
```

```
##
## Attaching package: 'mice'

## The following object is masked from 'package:stats':
##
##     filter

## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```r
fit.mice <- mice(dat_1, m=1, maxit=10, method = 'pmm', seed=100,
  diagnostics = FALSE, remove_collinear = FALSE);
```

```
##
##  iter imp variable
##   1   1  MORTDUE  VALUE  YOJ  DEROG  DELINQ  CLAGE  NINQ  CLNO  DEBTINC
##   2   1  MORTDUE  VALUE  YOJ  DEROG  DELINQ  CLAGE  NINQ  CLNO  DEBTINC
##   3   1  MORTDUE  VALUE  YOJ  DEROG  DELINQ  CLAGE  NINQ  CLNO  DEBTINC
##   4   1  MORTDUE  VALUE  YOJ  DEROG  DELINQ  CLAGE  NINQ  CLNO  DEBTINC
##   5   1  MORTDUE  VALUE  YOJ  DEROG  DELINQ  CLAGE  NINQ  CLNO  DEBTINC
##   6   1  MORTDUE  VALUE  YOJ  DEROG  DELINQ  CLAGE  NINQ  CLNO  DEBTINC
##   7   1  MORTDUE  VALUE  YOJ  DEROG  DELINQ  CLAGE  NINQ  CLNO  DEBTINC
##   8   1  MORTDUE  VALUE  YOJ  DEROG  DELINQ  CLAGE  NINQ  CLNO  DEBTINC
##   9   1  MORTDUE  VALUE  YOJ  DEROG  DELINQ  CLAGE  NINQ  CLNO  DEBTINC
##   10   1  MORTDUE  VALUE  YOJ  DEROG  DELINQ  CLAGE  NINQ  CLNO  DEBTINC
```

```
## Warning: Number of logged events: 2
```

```
data_imputed <- mice::complete(fit.mice, 1)
data_imputed<- model.matrix(~.-1, data=data_imputed)
dim(data_imputed)
```

```
## [1] 5960   19
```

```
anyNA(data_imputed)
```

```
## [1] FALSE
```

```
colMeans(is.na(data_imputed))
```

```
##          LOAN       MORTDUE         VALUE REASONDebtCon REASONHomeImp
##             0             0             0             0             0
## REASONUnknown     JOBOffice      JOBOther     JOBProfExe      JOBSales
##             0             0             0             0             0
##       JOBSelf    JOBUnknown           YOJ         DEROG        DELINQ
##             0             0             0             0             0
##         CLAGE          NINQ          CLNO       DEBTINC
##             0             0             0             0
```

Since the output is False, the missing values have been imputed successfully, so we proceed with distance matrix using daisy()

iii) Obtaining the Matrix Distance using daisy()

```
#handling categorical variables.
cols.cat <- c(1,2,3,4,5)
for (j in cols.cat) data_imputed[, j] <- as.factor(data_imputed[, j])
dat<-data.frame(data_imputed)
colMeans(is.na(dat))
```

```
##          LOAN       MORTDUE         VALUE REASONDebtCon REASONHomeImp
##             0             0             0             0             0
## REASONUnknown     JOBOffice      JOBOther     JOBProfExe      JOBSales
##             0             0             0             0             0
##       JOBSelf    JOBUnknown           YOJ         DEROG        DELINQ
##             0             0             0             0             0
##         CLAGE          NINQ          CLNO       DEBTINC
##             0             0             0             0
```

COMPUTING THE DISTANCE MATRIX USING daisy() IN CLUSTER - THE gower METRIC

```r
library(cluster)
dismat <- daisy(dat, metric="gower", stand=TRUE)
```

```
## Warning in daisy(dat, metric = "gower", stand = TRUE): binary variable(s) 4, 5,
## 6, 7, 8, 9, 10, 11, 12 treated as interval scaled
```

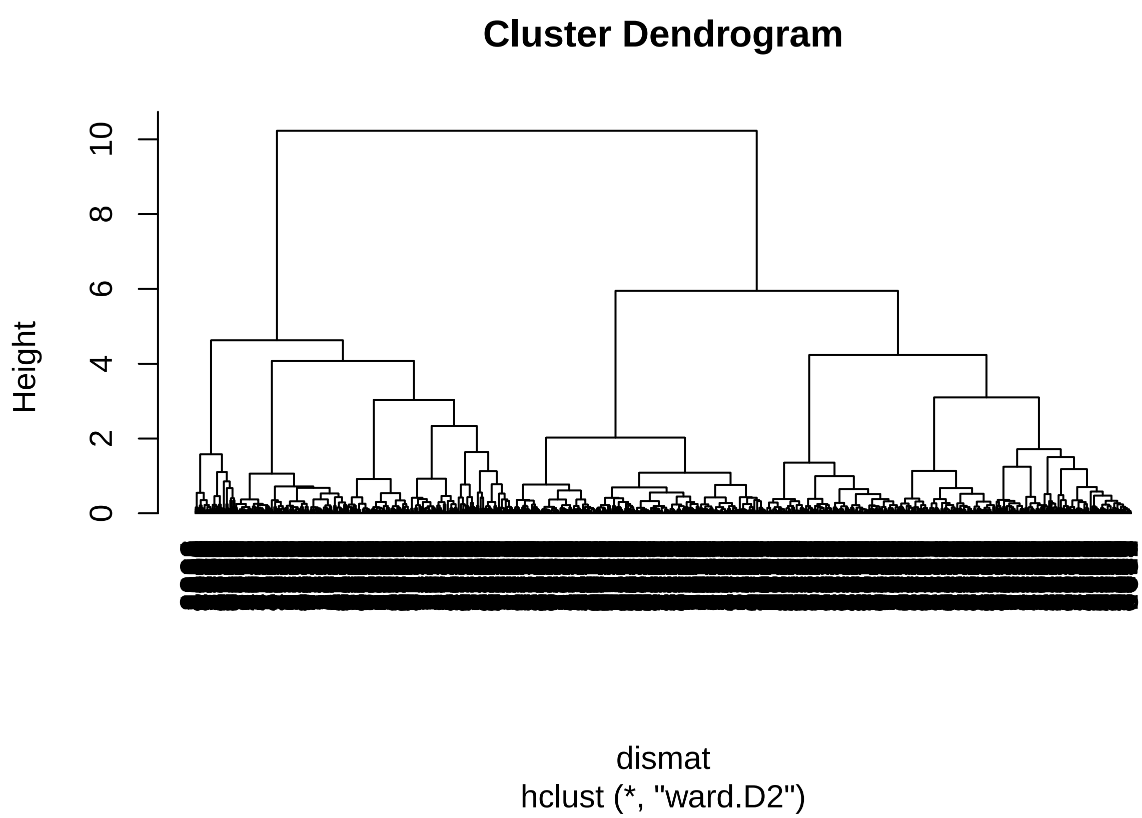
The distance matrix is obtained using the function 'daisy()' with the gower metric.

iv)

Choose two different clustering algorithms (of your choice) to cluster the data by excluding the variable 'BAD'. For each clustering algorithm.

We will perform clustering analysis on the data "dat" (with missing values imputed). In this work, we employ the hierarchical and Kmeans clustering
method 1 (Hierarchical Clustering) . . . .

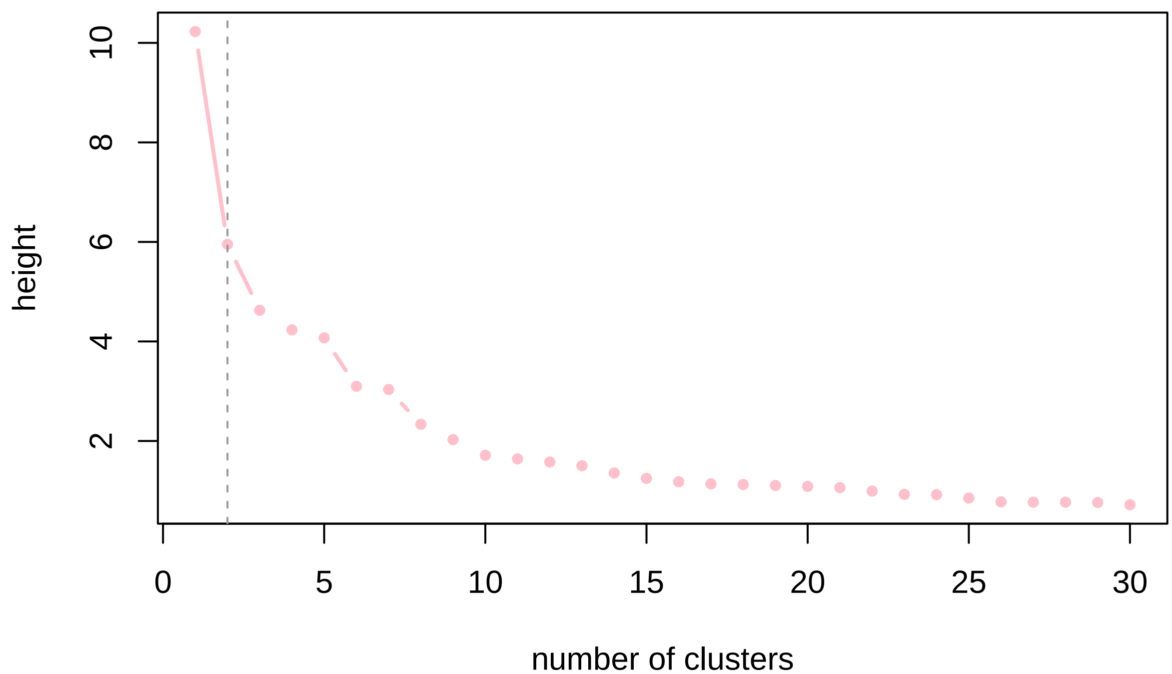
```r
fit.ward<-hclust(dismat,method = "ward.D2")
plot(fit.ward, hang=-0.5)
```



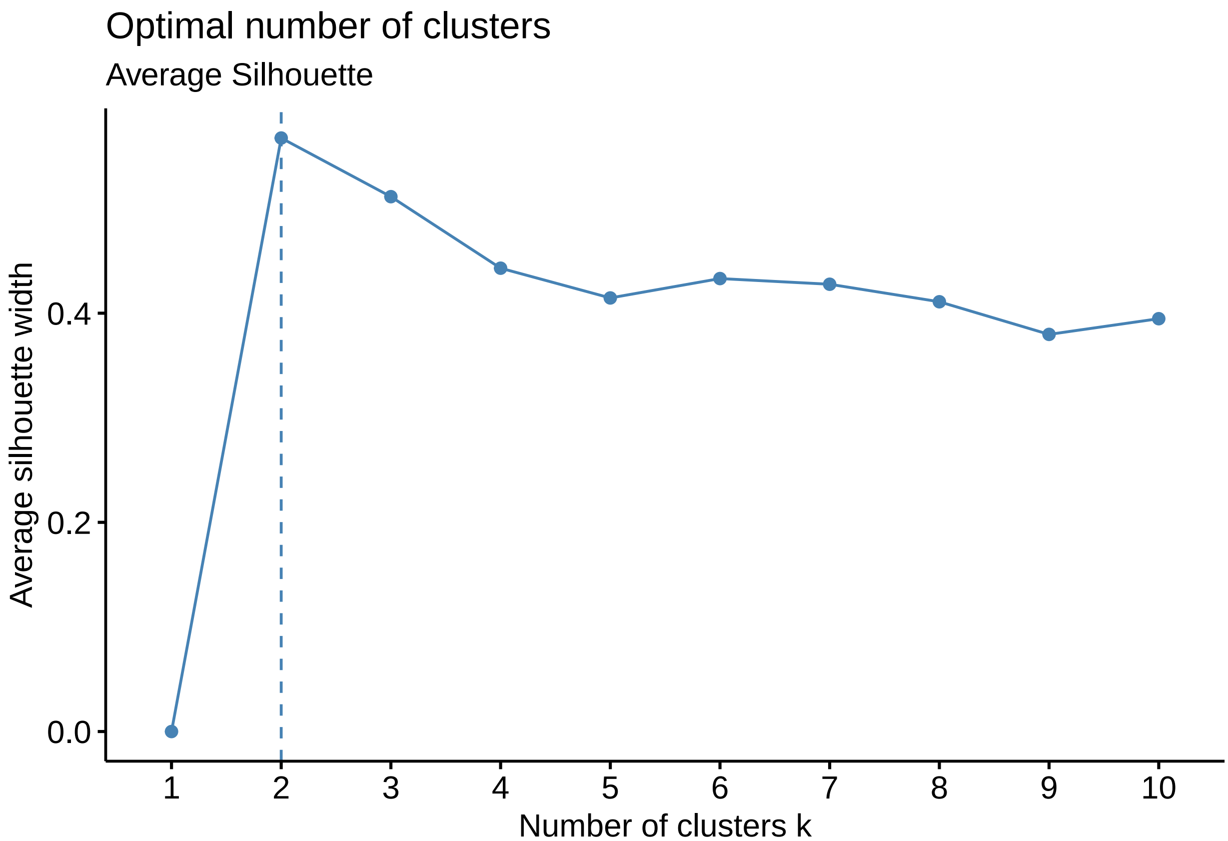
dismat is the distance matrix using the function daisy() . . . .

```r
# SCREE PLOT OF HEIGHT IN HIERARCHICAL CLUSTERING
set.seed(5860)
```

```r
K.max <- 30
height <- tail(fit.ward$height, n=K.max)
n.cluster <- tail((nrow(dat)-1):1, n=K.max)
plot(n.cluster, height,  type="b", pch=19, cex=.5, xlab="number of clusters",
     ylab="height", col="pink", lwd=2)
abline(v=2, col="gray60", lty=2)
```



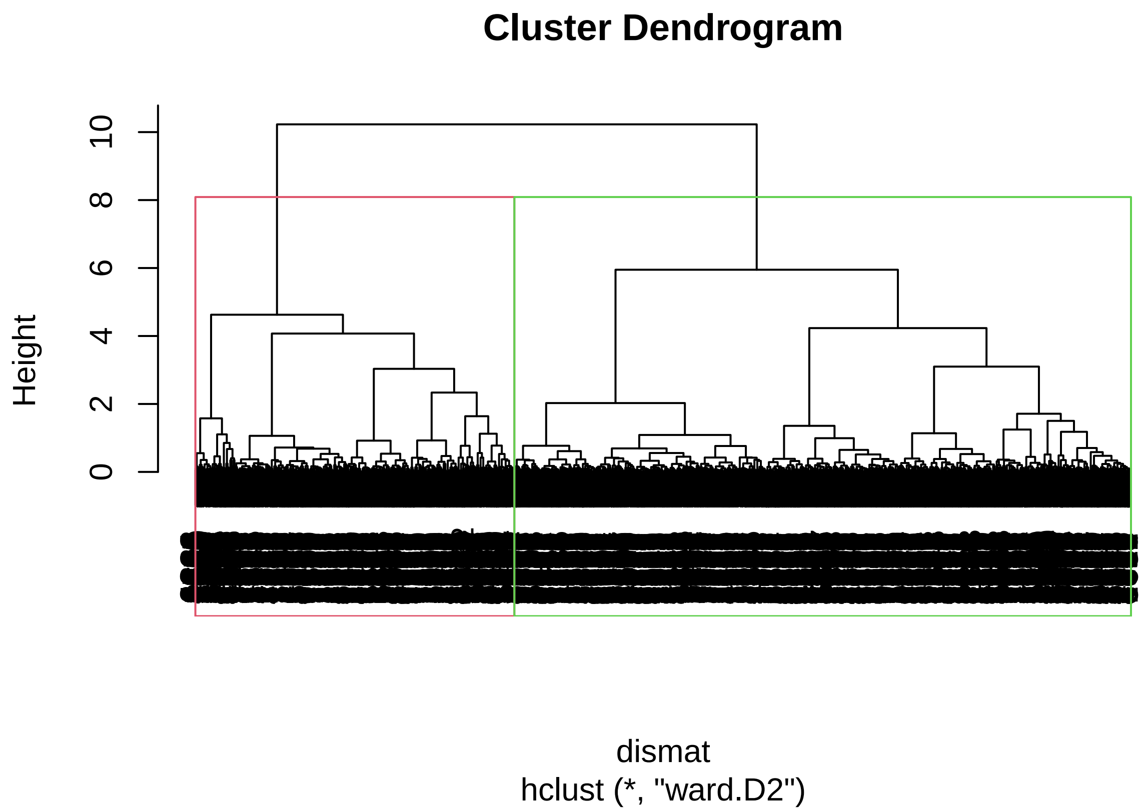

```r
set.seed(5860)
suppressMessages(library(factoextra))
fviz_nbclust(dat, kmeans, method = "silhouette") +
  labs(subtitle = "Average Silhouette")
```



It can be observed from the two plots above that K*=2

DENDROGRAM WITH THE FINAL CLUSTERS

```
k.star<-2
plot(fit.ward)
groups<-cutree(fit.ward, k=k.star)
rect.hclust(fit.ward,k=k.star,border=2:(k.star+1))
```

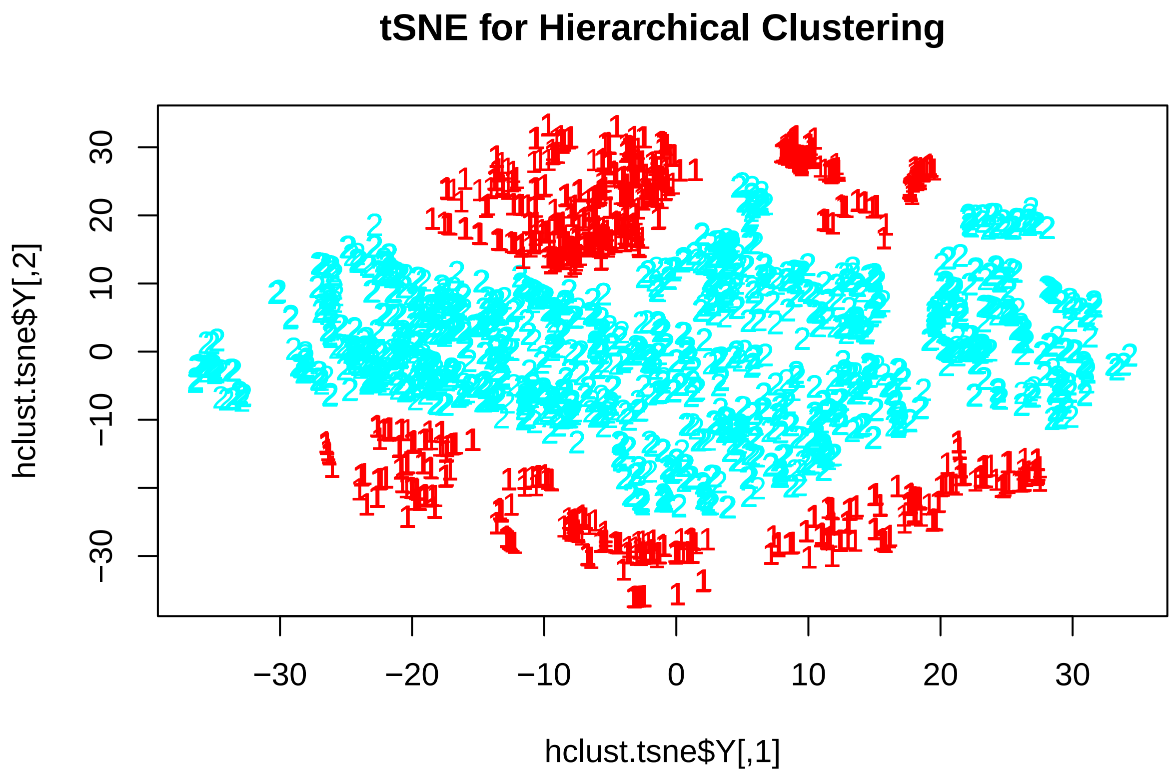



```
hclust.groups <- cutree(fit.ward, k=2)
table(hclust.groups)
```

```
## hclust.groups
##    1    2
## 2032 3928
```

Plotting dismat using tsne

```
library(Rtsne)
colors = rainbow(length(unique(hclust.groups)))
names(colors) = unique(hclust.groups)
set.seed(5860)
hclust.tsne <- Rtsne(dismat, dims=2, perplexity=30, max_iter=500)
plot(hclust.tsne$Y, t="n", main = "tSNE for Hierarchical Clustering")
text(hclust.tsne$Y, labels = hclust.groups, col = colors[hclust.groups])
```



It can be observed from the tsne plot that the matrix is put into two distinct subgroups(clusters).

Method 2 (K means Clustering) . . . .

```
# K-Means Cluster Analysis
K <- 2
fit.kmeans <- kmeans(dismat, K) # K cluster solution
```

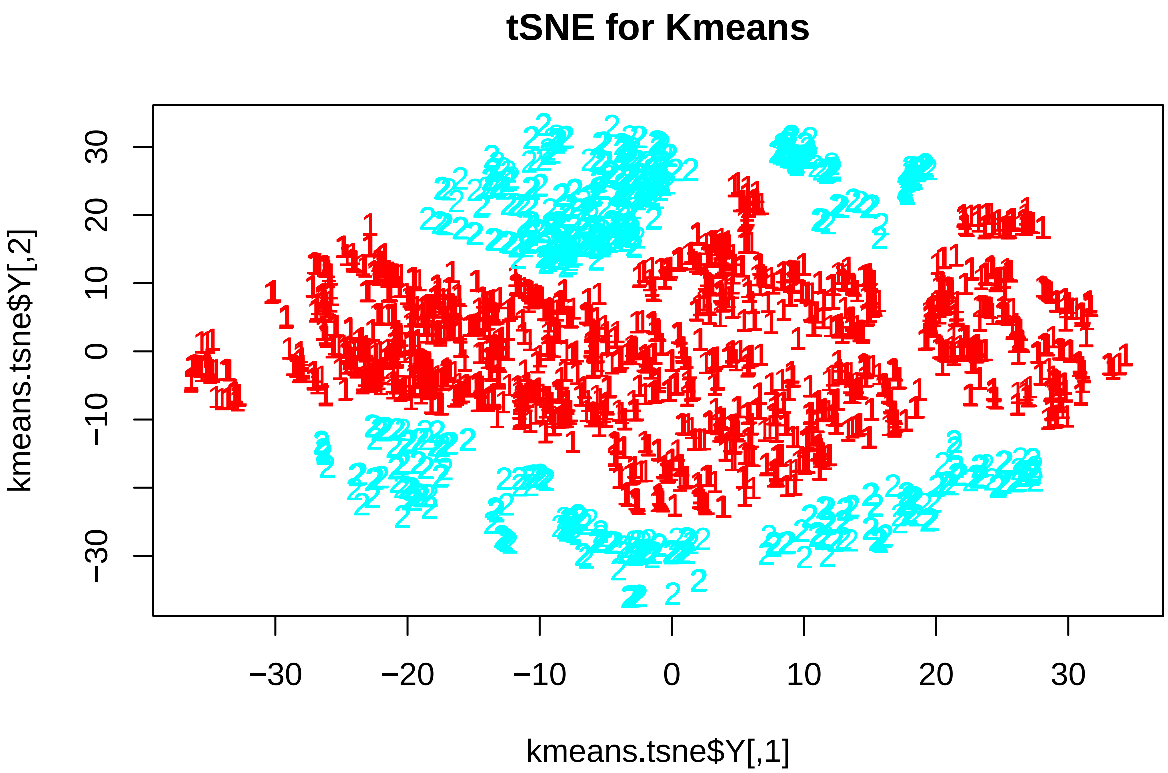
cluster memberships

```
kmeans.groups <- fit.kmeans$cluster
table(kmeans.groups)
```

```
## kmeans.groups
##    1    2
## 3928 2032
```

tSNE for K Means . . . .

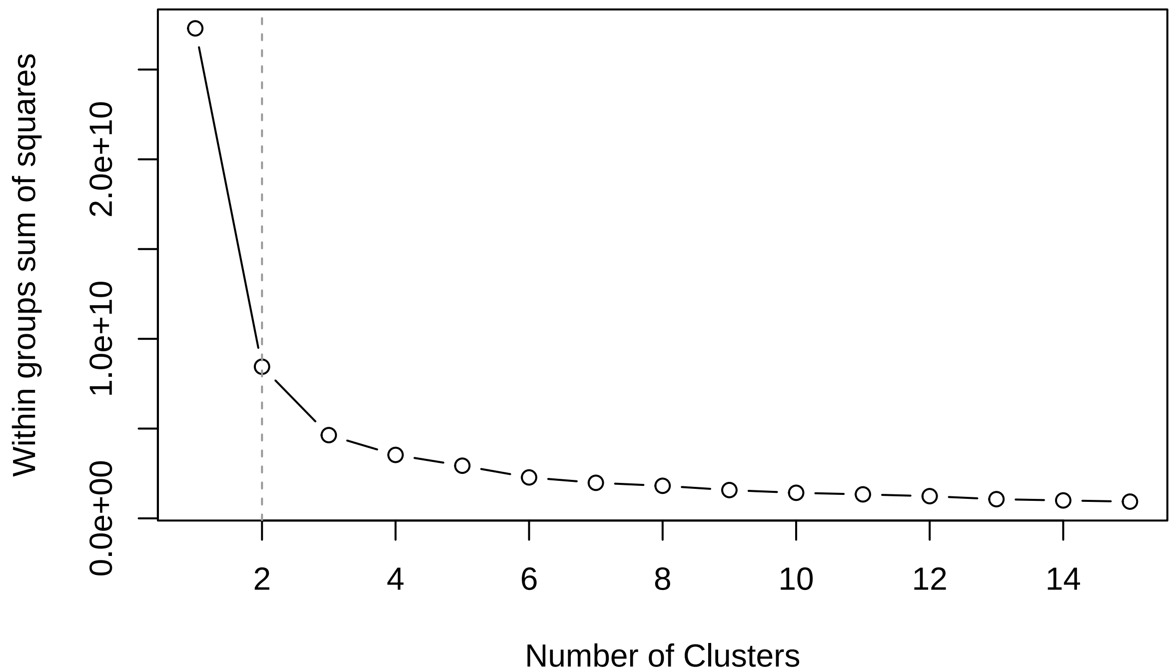
```
# plotting data
colors = rainbow(length(unique(kmeans.groups)))
names(colors) = unique(kmeans.groups)
set.seed(5860)
kmeans.tsne <- Rtsne(dismat, dims=2, perplexity=30, max_iter=500)
plot(kmeans.tsne$Y, t="n", main = "tSNE for Kmeans")
text(kmeans.tsne$Y, labels = kmeans.groups, col = colors[kmeans.groups])
```



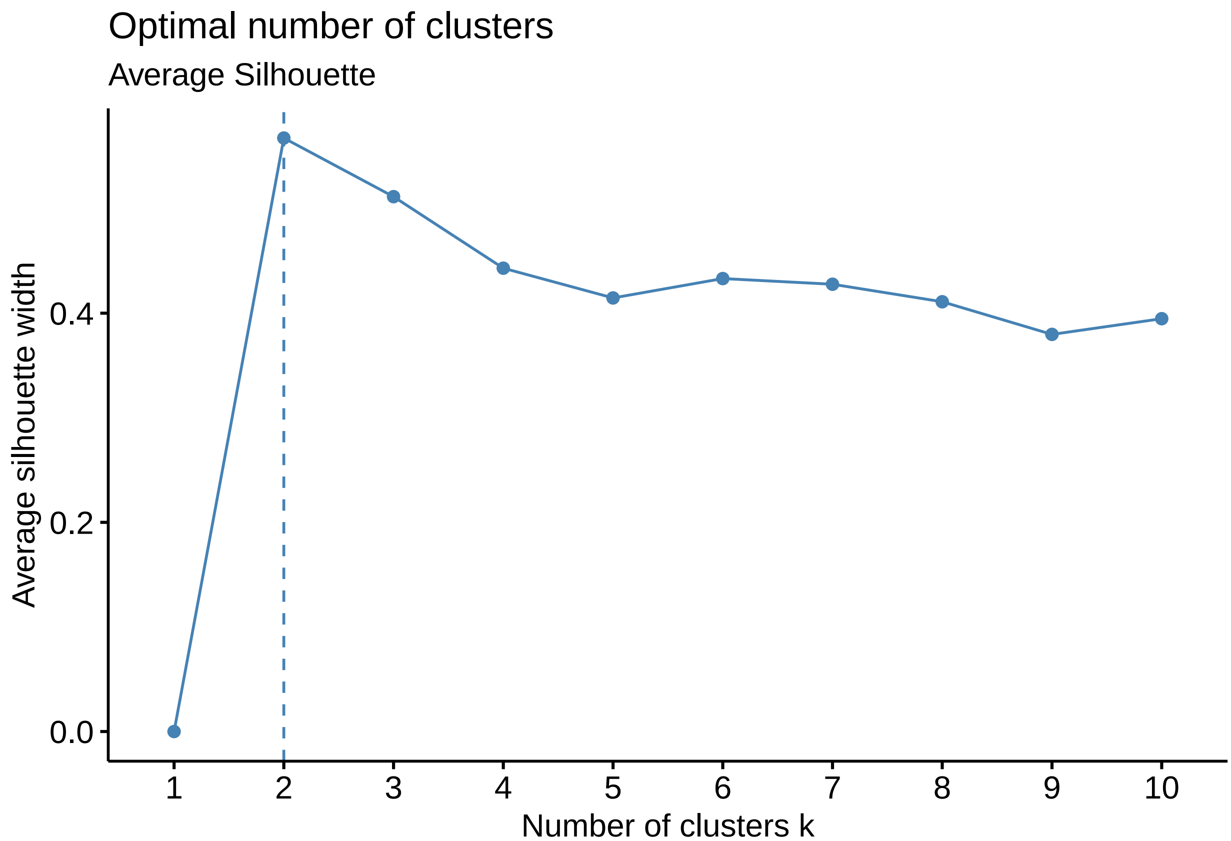
The hierarchical clustering appropriately clusters the data into the two clusters just as the K means.

Determining the number of clusters for K means

```
library(cluster)
set.seed(5600)
# SCREE PLOT OF HEIGHT IN Kmeans clustering
wss <- (nrow(dat)-1)*sum(apply(dat,2,var))
K.max <- 15
for (K in 2:K.max) wss[K] <- sum(kmeans(dat, centers=K)$withinss)
plot(1:K.max, wss, type="b", xlab="Number of Clusters",
  ylab="Within groups sum of squares")
abline(v=2, col="gray60", lty=2)
```




```
set.seed(5600)
suppressMessages(library(factoextra))
fviz_nbclust(dat, kmeans, method = "silhouette") +
  labs(subtitle = "Average Silhouette")
```



It can be observed from both plots above for the K Means that K*=2.

Comparing Hierarchical Clustering and K means Clustering ....

```
library(clusteval)
jaccard <- cluster_similarity(hclust.groups, kmeans.groups, similarity="jaccard", method="independence")
rand <- cluster_similarity(hclust.groups, kmeans.groups, similarity = "rand")
matrix(c("Jaccard", jaccard, "Rand", rand), byrow = T, ncol = 2)
```

```
##      [,1]      [,2]
## [1,] "Jaccard" "1"
## [2,] "Rand"    "1"
```

In comparing the two clustering methods using two-way contingency table,It is observed that the two methods are fairly similar, as their indices are fairly high.

v) The Post Hoc Analysis ....

We will use the result from the hierarchical clustering method to perform post hoc analysis.

```
dat<-data.frame(dat,dat0$BAD)
```

```
aggregate(dat[, c(1,2,3,6,9,12)], list(hclust.groups), mean, na.rm =T)
```

```
##   Group.1     LOAN  MORTDUE    VALUE REASONUnknown JOBProfExe JOBUnknown
## 1       1 138.9488 2207.310 2555.198     0.1240157  0.2111220 0.08070866
## 2       2 179.5601 2479.984 2723.914     0.0000000  0.2156314 0.02927699
```

```
cond1 <- hclust.groups == 1
cond2 <- hclust.groups == 2
var.test(dat$DEBTINC[cond1], dat$DEBTINC[cond2], alternative = c("two.sided"))
```

```
##
##  F test to compare two variances
##
## data:  dat$DEBTINC[cond1] and dat$DEBTINC[cond2]
## F = 1.3804, num df = 2031, denom df = 3927, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  1.280299 1.489803
## sample estimates:
## ratio of variances
##           1.380444
```

```
t.test(dat$DEBTINC[cond1], dat$DEBTINC[cond2], alternative = c("two.sided"), var.equal = T)
```

```
##
##  Two Sample t-test
##
## data:  dat$DEBTINC[cond1] and dat$DEBTINC[cond2]
## t = -6.8938, df = 5958, p-value = 5.992e-12
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.068237 -1.152399
## sample estimates:
## mean of x mean of y
##  32.64717  34.25749
```

It is observed from the output above that there is a statistically significant difference in the mean value of DEBTINC: Debt-to-income ratio for the two groups indicated by the smaller p-value. Cluster 2 turns to have high Debt-to-income ratio than Cluster 1.

Difference in YOJ

```
dat <- table(dat$YOJ , hclust.groups)
dat <- as.data.frame(dat)
dat$Var1 <- as.numeric(dat$Var1)
dat$hclust.groups <- as.numeric(dat$hclust.groups)
cond1 <- (dat$Var1 <= 10) & (dat$hclust.groups == 1)
cond2 <- (dat$Var1 <= 10) & (dat$hclust.groups == 2)
lessq1 <- sum(dat$Freq[cond1])
lessq2 <- sum(dat$Freq[cond2])
cond11 <- (dat$Var1 > 10) & (dat$hclust.groups == 1)
cond22 <- (dat$Var1 > 10) & (dat$hclust.groups == 2)
grt1 <- sum(dat$Freq[cond11])
grt2 <- sum(dat$Freq[cond22])
```

```r
matrix(c("YOJ", "Cluster 1", "Cluster 2", "<= 10", lessq1, lessq2, "> 10", grt1, grt2),
       byrow = T, ncol = 3)
```

```
##      [,1]    [,2]        [,3]
## [1,] "YOJ"   "Cluster 1" "Cluster 2"
## [2,] "<= 10" "137"       "379"
## [3,] "> 10"  "1895"      "3549"
```

It is observed from the table above that most of the individuals have the number of 'Year at present job' greater than 10 years in both clusters but the number of people with less than or equal to 10 years at their present job is much less in cluster 1 than cluster 2.

Determining the relationship between Predictors(JOB, REASON) and clusters.

```r
table(dat0$JOB,hclust.groups)
```

```
##          hclust.groups
##              1    2
##   Mgr      195  572
##   Office   328  620
##   Other    784 1604
##   ProfExe  429  847
##   Sales     12   97
##   Self     120   73
##   Unknown  164  115
```

```r
table(dat0$REASON, hclust.groups)
```

```
##          hclust.groups
##              1    2
##   DebtCon    0 3928
##   HomeImp 1780    0
##   Unknown  252    0
```

A relationship of the clusters and the applicant's job, and reason for the loan since that was the prudent and most effective relationship to draw.It was observed from the first table that majority of Managers and Sales personnel dominate in Cluster 2,whereas in other occupation the situation wasn't so.

From the second table, It is observed that debt consolidation was the major reason given by the applicants in the two clusters.