CS3907/CS6444 Big Data and Analytics

Fall 2018

Class Project #2

Due: October 22, 2018

Group 5

Group Member:

Yunfei Xu G21534523
Xiaohan Li G47313418
Siyuan Huang G43575455
Yanlong Shi G25529891

Exploring Variations in Clustering and Predictive Analysis

1. Data Set: Adult (on Blackboard)
These data sets are described in adult.data, adult.names, adult.test

Objective: There are two classes: >50K and <=50K (as determined by the authors- see adult.names)
1. Your job is to determine which of the adults falls into which category by applying clustering, classification, and prediction techniques discussed in class as well as additional functions from the packages mentioned.
2. Using clustering techniques, determine if there are more than just two classes. How many are there?

……..

Source:

Donor:

Ronny Kohavi and Barry Becker
Data Mining and Visualization
Silicon Graphics.

Data Set Information:

Extraction was done by Barry Becker from the 1994 Census database. A set of reasonably clean records was extracted using the following conditions: ((AAGE>16) && (AGI>100) && (AFNLWGT>1)&& (HRSWK>0))

Prediction task is to determine whether a person makes over 50K a year.

Attribute Information:

Listing of attributes:

>50K, <=50K.

age: continuous.
workclass: Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
fnlwgt: continuous.
education: Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.
education-num: continuous.
marital-status: Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.
occupation: Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces.
relationship: Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
race: White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.
sex: Female, Male.

capital-gain: continuous.
capital-loss: continuous.
hours-per-week: continuous.
native-country: United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands.

```
.............
setwd("D:\\download")
data <- read.table("./adult.data",header=FALSE, sep = ",")
names(data) <- c("age","workclass","fnlwgt","education","education-num",
                 "marital","occupation","relationship","race","sex","capital-gain",
                 "capital-loss","hours-per-week","native-country","salary")
age <- data[[1]]
workclass <- data[[2]]
fnlwgt <- data[[3]]
education <- data[[4]]
education_num <- data[[5]]
marital <- data[[6]]
occupation <- data[[7]]
relationship <- data[[8]]
race <- data[[9]]
sex <- data[[10]]
capital_gain <- data[[11]]
capital_loss <- data[[12]]
hours_per_week <- data[[13]]
native_country <- data[[14]]
salary <- data[[15]]
```

At first, we need to load the adult.data file and name each attribute.

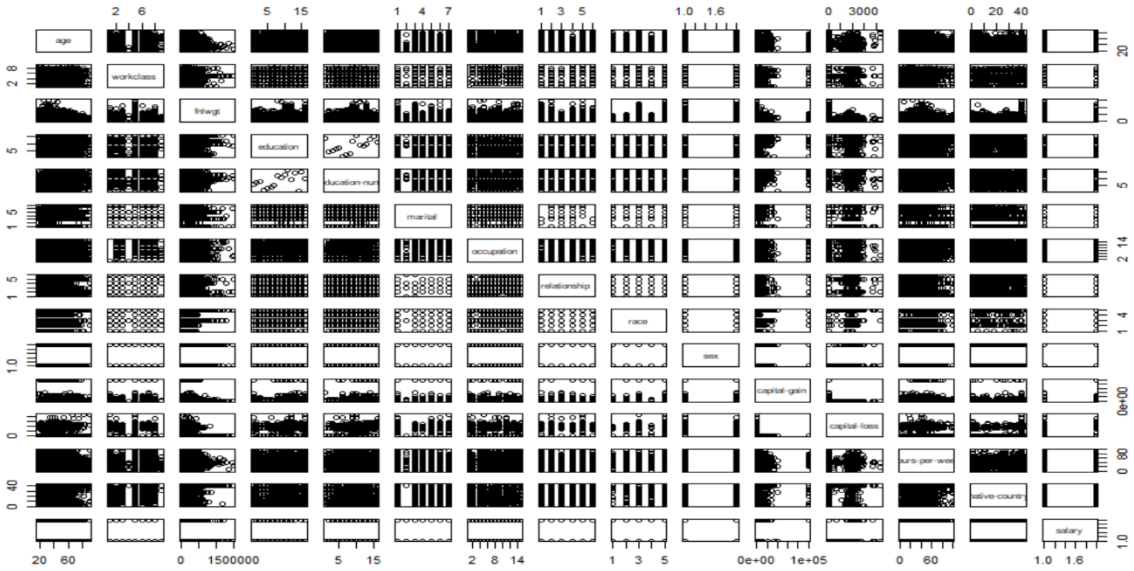
By the way, when you want to execute our source code, please don't forget to change data direction in your own way.

You will need to divide your data set into a training set and a test set. Use samples of 50-50, 60-40, and 70-30 for the training-test ratios. One test set, adult.test, is already provided.

```
tran1=sample(nrow(data),0.7*nrow(data))
tran1.df=data[tran1,]
test1.df=data[-tran1,]
tran2=sample(nrow(data),0.6*nrow(data))
tran2.df=data[tran2,]
test2.df=data[-tran2,]
tran3=sample(nrow(data),0.5*nrow(data))
tran3.df=data[tran3,]
test3.df=data[-tran3,]
```

Try plotting the data using several plotting functions to see what it looks like. Use pairs (e.g., 2D plots) or 3 variables (3D plots) based on the packages.

Using pairs(data) function, we can plot how data in one picture.

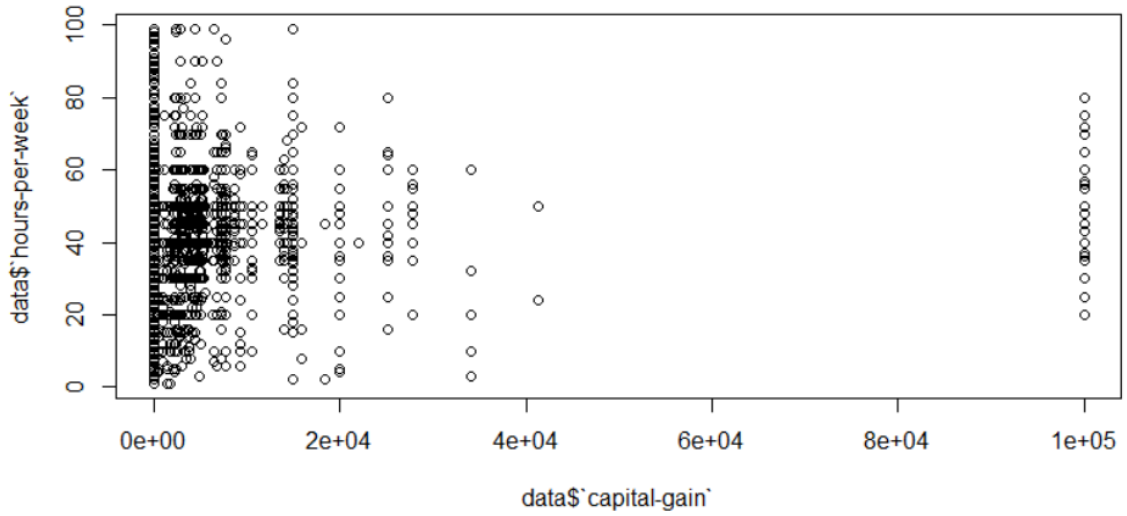


Try to filter the data by selecting samples with only certain attribute values and plotting them.

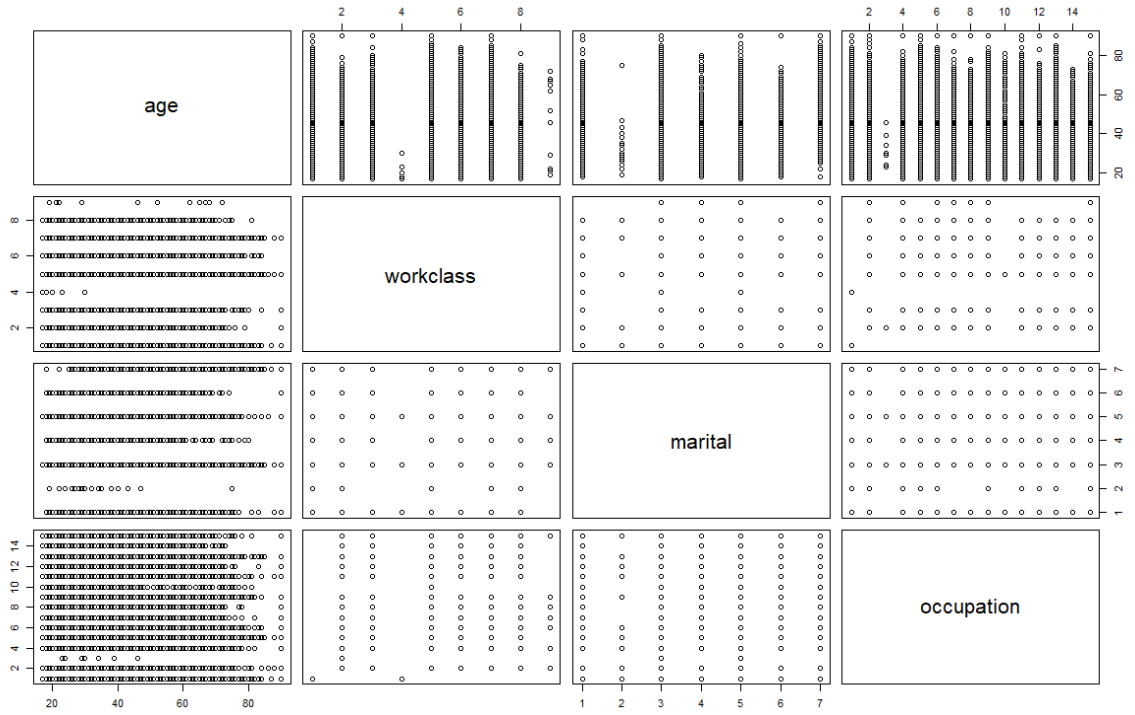
We can use several function to filter the data.

```
plot(data$`capital-gain`, data$`hours-per-week`)
pairs(age ~ workclass+ fnlwgt + occupation, data=data)
ggplot(data, aes(sex)) + geom_bar(aes(fill= salary), alpha=0.8)
plot3d(as.numeric(native_country),as.numeric(race),hours_per_week)
```

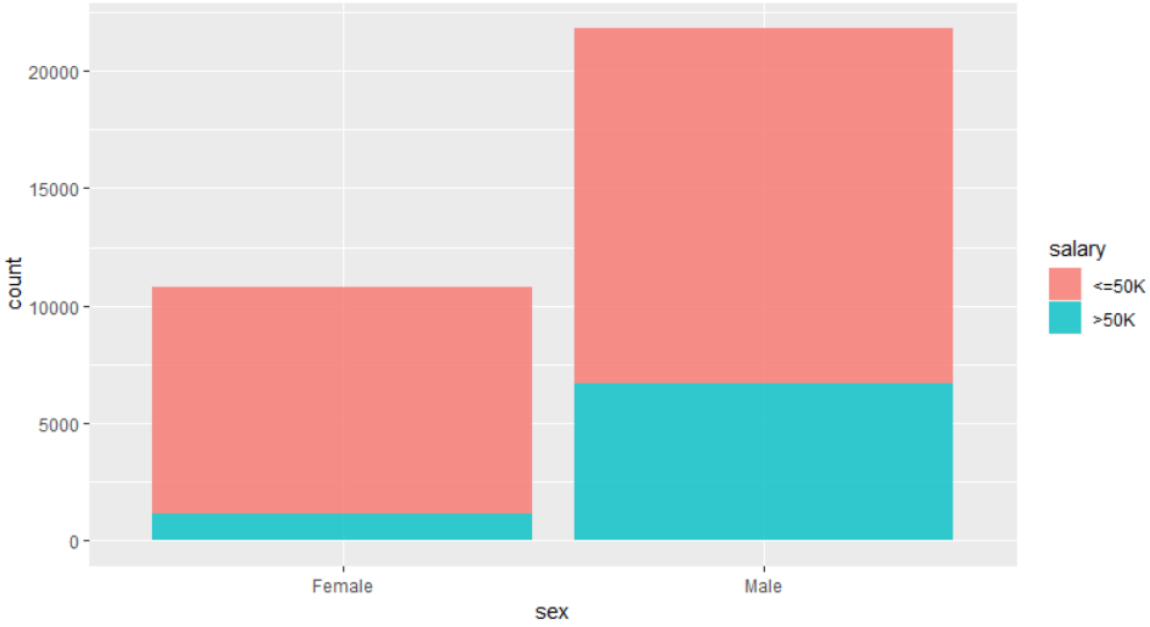
Firstly, we use traditional plots function
plot (data$`capital-gain`, data$`hours-per-week`)



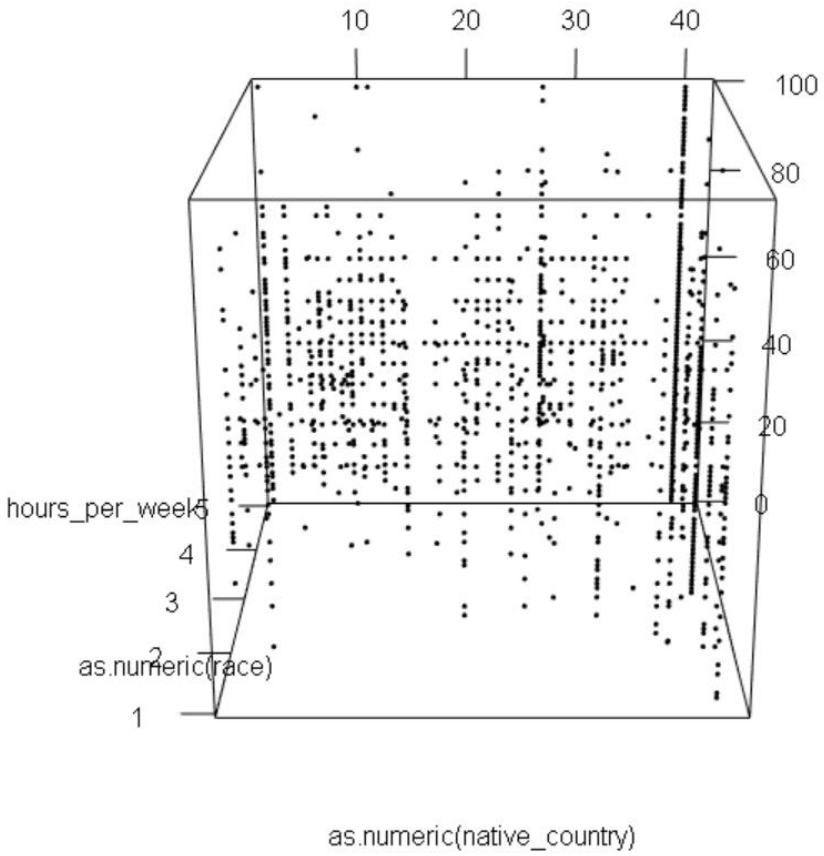
Let us try pairs function.
pairs(age ~ workclass+ fnlwgt + occupation, data=data)
In this function, we can compare 4 specific attributes in data



Or ggplot in package ggplot2
ggplot(data, aes(sex)) + geom_bar(aes(fill= salary), alpha=0.8)
It can compare two attributes.



Or plot3d(as.numeric(native_country),as.numeric(race),hours_per_week)
Plot 3 attribute using 3d.



You should try data reduction to eliminate some attributes through Principal Components Analysis. The idea is to try and select N attributes that will help you focus on identifying the unsure samples.

We try to eliminate marital attribute in data. You can find that the original marital contains several status: Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse. We find that they are too much to be display, so we just divide them into two attributes, not-married and married.

```
> marital
   [1] Never-married         Married-civ-spouse    Divorced
   [4] Married-civ-spouse    Married-civ-spouse    Married-civ-spouse
   [7] Married-spouse-absent Married-civ-spouse    Never-married
  [10] Married-civ-spouse    Married-civ-spouse    Married-civ-spouse
  [13] Never-married         Never-married         Married-civ-spouse
  [16] Married-civ-spouse    Never-married         Never-married
  [19] Married-civ-spouse    Divorced              Married-civ-spouse
  [22] Separated             Married-civ-spouse    Married-civ-spouse
  [25] Divorced              Married-civ-spouse    Never-married
  [28] Married-civ-spouse    Divorced              Married-civ-spouse
  [31] Never-married         Never-married         Divorced
  [34] Married-civ-spouse    Married-civ-spouse    Never-married
  [37] Never-married         Married-AF-spouse     Married-civ-spouse
  [40] Married-civ-spouse    Married-civ-spouse    Married-civ-spouse
  [43] Married-civ-spouse    Separated             Never-married
  [46] Married-civ-spouse    Married-civ-spouse    Divorced
  [49] Married-civ-spouse    Never-married         Married-civ-spouse
  [52] Never-married         Married-civ-spouse    Divorced
  [55] Divorced              Married-civ-spouse    Married-civ-spouse
```

Use function to set divorced, separated, widowed as not married and married-af-spouse, married-civ-spouse and married-spouse-absent as married.

```
data$marital
marital.s <- function(marriage) {
  marriage <- as.character(marriage)
  if (marriage == "Divorced"|marriage =="Separated" | marriage =="Widowed" ) {
    return("Not-Married")
  } else if (marriage == "Married-AF-spouse" | marriage == "Married-civ-spouse"
             | marriage == "Married-spouse-absent") {
    return("Married")
  } else {
    return(marriage)
  }
}
data$marital <- sapply(data$marital, marital.s)
data$marital
```

The new marital result shows below.

```
> data$marital
   [1] "Never-married" "Married"       "Not-Married"   "Married"       "Married"
   [6] "Married"       "Married"       "Married"       "Never-married" "Married"
  [11] "Married"       "Married"       "Never-married" "Never-married" "Married"
  [16] "Married"       "Never-married" "Never-married" "Married"       "Not-Married"
  [21] "Married"       "Not-Married"   "Married"       "Married"       "Not-Married"
  [26] "Married"       "Never-married" "Married"       "Not-Married"   "Married"
  [31] "Never-married" "Never-married" "Not-Married"   "Married"       "Married"
  [36] "Never-married" "Never-married" "Married"       "Married"       "Married"
  [41] "Married"       "Married"       "Married"       "Not-Married"   "Never-married"
  [46] "Married"       "Married"       "Not-Married"   "Married"       "Never-married"
  [51] "Married"       "Never-married" "Married"       "Not-Married"   "Not-Married"
  [56] "Married"       "Married"       "Married"       "Married"       "Married"
  [61] "Married"       "Married"       "Married"       "Married"       "Not-Married"
  [66] "Married"       "Not-Married"   "Married"       "Married"       "Never-married"
  [71] "Never-married" "Not-Married"   "Married"       "Never-married" "Married"
  [76] "Never-married" "Married"       "Married"       "Never-married" "Married"
  [81] "Never-married" "Married"       "Married"       "Married"       "Not-Married"
  [86] "Not-Married"   "Married"       "Married"       "Never-married" "Never-married"
  [91] "Married"       "Not-Married"   "Not-Married"   "Married"       "Married"
```

3.

At first, we need to translate all factor type in 6 datasets into numeric. These are the example to translate tran1.df and test1.df

```
for(i in 1:ncol(tran1.df)){
  if(class(tran1.df[,i])=="factor")
    tran1.df[,i]=as.numeric(tran1.df[,i])
}
for(i in 1:ncol(test1.df)){
  if(class(test1.df[,i])=="factor")
    test1.df[,i]=as.numeric(test1.df[,i])
}
```

Then we use three different clustering methods, k-means, Cluster Analysis and K nearest neighbor function. For N=3,5,7.

3.a

Now, we will perform k-means:

Kmeans with number of clusters (N)=3 on training data set with 70% of the values:

```
84  a=kmeans(tran1.df,3)
85  a
```

```
> a=kmeans(tran1.df,3)
> a
K-means clustering with 3 clusters of sizes 11101, 3352, 8339

Cluster means:
       age workclass    fnlwgt education education-num  marital occupation relationship     race
1 38.29961  4.857220 205428.84  11.28718     10.075399 3.613098   7.577335     2.461850 4.696784
2 36.64857  4.818616 376454.64  11.16110      9.922434 3.674224   7.647375     2.434069 4.619332
3 39.78930  4.909821  94530.43  11.36947     10.151217 3.566135   7.583763     2.429068 4.645881
       sex capital-gain capital-loss hours-per-week native-country   salary
1 1.663724     1110.930     83.80560       40.35537       37.50059 1.239258
2 1.704057     1033.956     83.17333       40.17512       37.23061 1.236575
3 1.660871     1093.465     91.09330       40.65739       38.20146 1.237558

Clustering vector:
12059 17796  5544 20349 28721  9128 12973 24825 21779  6661 11638 11701 22469 17440 23135 17522 24376
    3     1     2     1     2     3     1     1     3     3     3     1     1     1     3     3     1
13672  5579 25068 28700 17868  9037 15889 30211 11346 31044 22621 28937  5870 20475 32191  4239 10756
    3     1     3     3     3     1     1     2     3     3     1     1     3     1     1     1     3
28140 25292 26909 19623 15977 25379 28756  6755  9987 10749  6461  7664  8938 19227  8239  4015  7475
    1     1     1     1     3     1     3     3     1     3     2     2     3     3     3     1     3
19428  6873 15074 21036 31225 21987 14470 11629 14813 14477  7966 22566 13397 10651 18607 31423 21504
```

Kmeans with number of clusters (N)=3 on test data set with 30% of the values:

```
84  a=kmeans(test1.df,3)
85  a
86
```

```
> a=kmeans(test1.df,3)
> a
K-means clustering with 3 clusters of sizes 4797, 3551, 1421

Cluster means:
       age workclass    fnlwgt education education-num  marital occupation relationship     race
1 38.44069  4.850740 205131.94  11.24745     10.006671 3.644153   7.662497     2.449864 4.688972
2 39.45001  4.886511  91079.73  11.42664     10.245283 3.582090   7.429456     2.440439 4.657280
3 36.56369  4.855735 380929.33  11.14004      9.919775 3.688248   7.351161     2.458832 4.594652
       sex capital-gain capital-loss hours-per-week native-country   salary
1 1.663957     916.7847     89.83427       40.26621       37.40630 1.242652
2 1.667136    1081.9806     93.70910       40.62264       38.26471 1.259082
3 1.701619    1360.1246     77.58902       40.52217       37.43490 1.230120

Clustering vector:
   2    4    9   14   18   29   36   38   43   49   53   55   59   62   64   67   70   73   84   85   90
   2    1    2    1    1    3    1    3    1    2    2    2    2    3    2    1    1    1    1    3    3
  91   93   94  106  108  114  117  125  130  135  136  141  143  145  150  156  162  166  169  173  174
   1    1    2    3    1    3    3    3    2    2    2    3    2    1    1    1    1    2    2    3
 175  178  185  186  187  191  194  195  200  201  203  209  215  218  226  235  240  241  243  246  248
```

Cluster Analysis with number of clusters (N)=3 on training data set with 70% of the values:

```
86  b=iclust(tran1.df, 3)
87  b
```

88:1 (Top Level)

Console D:/download/

```
Item by Cluster Structure matrix:
                  O    P   C12    C2   C10
age             C12  C12  0.34 -0.02  0.15
workclass        C2   C2  0.15  0.41  0.08
fnlwgt          C10  C10 -0.05 -0.02 -0.12
education       C12  C12  0.19  0.00  0.13
education-num   C12  C12  0.41  0.19  0.16
marital         C12  C12 -0.35 -0.08 -0.17
occupation       C2   C2  0.12  0.41  0.00
relationship    C12  C12 -0.59 -0.19 -0.17
race            C10  C10  0.13  0.07  0.29
sex             C12  C12  0.46  0.21  0.07
capital-gain    C12  C12  0.19  0.07  0.03
capital-loss    C12  C12  0.13  0.04  0.03
hours-per-week  C12  C12  0.39  0.25  0.07
native-country  C10  C10  0.04 -0.02  0.32
salary          C12  C12  0.60  0.15  0.13
```

Cluster Analysis with number of clusters (N)=3 on test data set with 30% of the values:

```
87  b=iclust(test1.df, 3)
```

89:1 (Top Level)

Console D:/download/

```
Item by Cluster Structure matrix:
                  O    P   C11   C12    C4
age             C11  C11  0.38  0.09 -0.02
workclass        C4   C4  0.17  0.10  0.42
fnlwgt          C12  C12 -0.04 -0.13 -0.01
education       C12  C12  0.07  0.43  0.01
education-num   C12  C12  0.28  0.51  0.22
marital         C11  C11 -0.38 -0.15 -0.12
occupation       C4   C4  0.11  0.05  0.41
relationship    C11  C11 -0.64 -0.18 -0.22
race            C12  C12  0.13  0.16  0.06
sex             C11  C11  0.53  0.04  0.22
capital-gain    C11  C11  0.18  0.09  0.07
capital-loss    C11  C11  0.13  0.10  0.03
hours-per-week  C11  C11  0.40  0.20  0.29
native-country  C12  C12  0.01  0.22 -0.03
salary          C11  C11  0.56  0.35  0.16
```

K nearest neighbor on training data set with 70% of the values and test data set with 30% of the values, with N=3

```
data.knn <- data
for(i in 1:ncol(data.knn)){
  if(class(data.knn[,i])=="factor")
    data.knn[,i]=as.numeric(data.knn[,i])
}
data.knn[,6] <- as.numeric(as.factor(data.knn[,6]))
data.knn <- as.data.frame(data.knn)
idxs <- sample(nrow(data.knn),0.7*nrow(data.knn))
train <- data.knn[idxs,]
test <- data.knn[-idxs,]
result <- knn(train= train[,-ncol(train)],test=test[,-ncol(test)],cl=train[,ncol(train)],3)
CrossTable(x=test[,ncol(test)],y=result)
```

```
   Cell Contents
|-------------------------|
|                       N |
| Chi-square contribution |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|


Total Observations in Table:  9769


                    | result
test[, ncol(test)] |         1 |         2 | Row Total |
-------------------|-----------|-----------|-----------|
                 1 |      6469 |       886 |      7355 |
                   |    32.660 |   148.004 |           |
                   |     0.880 |     0.120 |     0.753 |
                   |     0.808 |     0.502 |           |
                   |     0.662 |     0.091 |           |
-------------------|-----------|-----------|-----------|
                 2 |      1534 |       880 |      2414 |
                   |    99.508 |   450.940 |           |
                   |     0.635 |     0.365 |     0.247 |
                   |     0.192 |     0.498 |           |
                   |     0.157 |     0.090 |           |
-------------------|-----------|-----------|-----------|
      Column Total |      8003 |      1766 |      9769 |
                   |     0.819 |     0.181 |           |
-------------------|-----------|-----------|-----------|
```

3.b We choose 70-30 for the training -test ratio

K-means:
Kmeans with number of clusters (N)=3 on training data set with 70% of the values:

```
84  a=kmeans(tran1.df,3)
85  a
```

```
> a=kmeans(tran1.df,3)
> a
K-means clustering with 3 clusters of sizes 11101, 3352, 8339

Cluster means:
       age workclass    fnlwgt education education-num  marital occupation relationship     race
1 38.29961  4.857220 205428.84  11.28718     10.075399 3.613098   7.577335     2.461850 4.696784
2 36.64857  4.818616 376454.64  11.16110      9.922434 3.674224   7.647375     2.434069 4.619332
3 39.78930  4.909821  94530.43  11.36947     10.151217 3.566135   7.583763     2.429068 4.645881
       sex capital-gain capital-loss hours-per-week native-country   salary
1 1.663724     1110.930     83.80560       40.35537       37.50059 1.239258
2 1.704057     1033.956     83.17333       40.17512       37.23061 1.236575
3 1.660871     1093.465     91.09330       40.65739       38.20146 1.237558

Clustering vector:
12059 17796  5544 20349 28721  9128 12973 24825 21779  6661 11638 11701 22469 17440 23135 17522 24376
    3     1     2     1     2     3     1     1     3     3     3     1     1     1     3     3     1
13672  5579 25068 28700 17868  9037 15889 30211 11346 31044 22621 28937  5870 20475 32191  4239 10756
    3     1     3     3     3     1     1     2     3     3     1     1     3     1     1     1     3
28140 25292 26909 19623 15977 25379 28756  6755  9987 10749  6461  7664  8938 19227  8239  4015  7475
    1     1     1     1     3     1     3     3     1     3     2     2     3     3     3     1     3
19428  6873 15074 21036 31225 21987 14470 11629 14813 14477  7966 22566 13397 10651 18607 31423 21504
```

Kmeans with number of clusters (N)=3 on test data set with 30% of the values:

```
84  a=kmeans(test1.df,3)
85  a
86
```

```
> a=kmeans(test1.df,3)
> a
K-means clustering with 3 clusters of sizes 4797, 3551, 1421

Cluster means:
       age workclass    fnlwgt education education-num  marital occupation relationship     race
1 38.44069  4.850740 205131.94  11.24745     10.006671 3.644153   7.662497     2.449864 4.688972
2 39.45001  4.886511  91079.73  11.42664     10.245283 3.582090   7.429456     2.440439 4.657280
3 36.56369  4.855735 380929.33  11.14004      9.919775 3.688248   7.351161     2.458832 4.594652
       sex capital-gain capital-loss hours-per-week native-country   salary
1 1.663957     916.7847     89.83427       40.26621       37.40630 1.242652
2 1.667136    1081.9806     93.70910       40.62264       38.26471 1.259082
3 1.701619    1360.1246     77.58902       40.52217       37.43490 1.230120

Clustering vector:
   2    4    9   14   18   29   36   38   43   49   53   55   59   62   64   67   70   73   84   85   90
   2    1    2    1    1    3    1    3    1    2    2    2    2    3    2    1    1    1    1    3    3
  91   93   94  106  108  114  117  125  130  135  136  141  143  145  150  156  162  166  169  173  174
   1    1    2    3    1    3    3    3    3    2    2    2    3    2    1    1    1    1    2    2    3
 175  178  185  186  187  191  194  195  200  201  203  209  215  218  226  235  240  241  243  246  248
```

Kmeans with number of clusters (N)=5 on training data set with 70% of the values:

```
84  a=kmeans(tran1.df,5)
85  a
86
```

```
[7] "size"         "iter"         "ifault"
> a=kmeans(tran1.df,5)
> a
K-means clustering with 5 clusters of sizes 3529, 552, 8081, 7014, 3616

Cluster means:
       age workclass   fnlwgt education education-num  marital occupation relationship     race      sex
1 37.20317  4.835647 332095.5  11.18844      9.975630 3.656843   7.670162     2.412865 4.635874 1.706432
2 34.43478  4.827899 534169.2  11.00181      9.695652 3.764493   7.744565     2.489130 4.545290 1.721014
3 37.98354  4.842717 210860.8  11.27348     10.049994 3.621829   7.557604     2.466031 4.704987 1.662047
4 39.93684  4.915883 135616.4  11.32221     10.166952 3.567722   7.550898     2.452096 4.662390 1.655261
5 39.39519  4.886892  57694.8  11.46267     10.142976 3.564159   7.636338     2.413717 4.638551 1.664270
  capital-gain capital-loss hours-per-week native-country   salary
1    1030.8614     90.86427       40.30774       37.43072 1.240578
2    1010.4656     42.01449       39.82428       36.52174 1.202899
3    1028.0468     79.70028       40.30219       37.39228 1.233263
4    1262.4187    102.23852       40.32193       37.95195 1.256487
5     984.1593     72.93667       41.19607       38.45077 1.217091

Clustering vector:
12059 17796  5544 20349 28721  9128 12973 24825 21779  6661 11638 11701 22469 17440 23135 17522 24376
    4     4     1     3     1     4     1     3     5     4     5     1     3     3     5     5     4
13672  5579 25068 28700 17868  9037 15889 30211 11346 31044 22621 28937  5870 20475 32191  4239 10756
    4     1     5     4     5     3     4     1     5     4     4     3     5     3     3     3     5
28140 25292 26909 19623 15977 25379 28756  6755  9987 10749  6461  7664  8938 19227  8239  4015  7475
    4     3     4     4     5     3     5     4     4     5     1     2     4     4     4     3     4
```

Kmeans with number of clusters (N)=5 on test data set with 30% of the values:

```
84  a=kmeans(test1.df,5)
85  a
86
```

```
[7] "size"         "iter"         "ifault"
> a=kmeans(test1.df,5)
> a
K-means clustering with 5 clusters of sizes 1507, 3006, 191, 1601, 3464

Cluster means:
       age workclass    fnlwgt education education-num  marital occupation relationship     race
1 36.59987  4.856005 340231.07  11.27074      9.988719 3.688786   7.481752     2.429993 4.601194
2 39.87658  4.869261 134932.88  11.30273     10.172655 3.587824   7.533932     2.486361 4.667997
3 35.72251  4.858639 562330.98  10.37173      9.345550 3.801047   7.392670     2.617801 4.413613
4 38.86758  4.893192  55153.46  11.59400     10.215490 3.569019   7.378513     2.364772 4.617739
5 38.21276  4.851039 212095.92  11.21709     10.019342 3.654157   7.632217     2.450924 4.722286
       sex capital-gain capital-loss hours-per-week native-country   salary
1 1.701393     1285.153     86.31254       40.79960       37.68082 1.231586
2 1.657352     1277.305     88.78244       40.49634       37.74251 1.272455
3 1.696335     1129.738     68.97382       38.90576       36.52356 1.204188
4 1.667708      661.970     97.03560       40.73891       38.67083 1.234229
5 1.668591      900.914     89.04994       40.16137       37.35104 1.239319

Clustering vector:
   2    4    9   14   18   29   36   38   43   49   53   55   59   62   64   67   70   73   84   85   90
   4    5    4    5    5    1    5    3    2    2    4    2    2    1    2    5    5    2    2    1    1
  91   93   94  106  108  114  117  125  130  135  136  141  143  145  150  156  162  166  169  173  174
   5    5    2    1    5    1    1    1    1    2    2    4    1    2    5    5    5    5    4    2    1
 175  178  185  186  187  191  194  195  200  201  203  209  215  218  226  235  240  241  243  246  248
   2    3    4    1    2    2    2    2    4    1    5    5    2    5    5    4    2    1    5    5    5
```

Kmeans with number of clusters (N)=7 on training data set with 70% of the values:

```
84  a=kmeans(tran1.df,7)
85  a
86
```

```
> a=kmeans(tran1.df,7)
> a
K-means clustering with 7 clusters of sizes 7180, 3708, 3121, 826, 2335, 5514, 108

Cluster means:
       age workclass    fnlwgt education education-num  marital occupation relationship     race
1 38.69791  4.858078 185361.35  11.36100     10.179109 3.590390   7.574791     2.454875 4.730919
2 37.49569  4.846009 250517.95  11.15426      9.866505 3.659115   7.608145     2.471953 4.633225
3 38.97116  4.887536  52236.12  11.49183     10.158923 3.575777   7.618392     2.397949 4.653316
4 35.18039  4.796610 451910.33  10.95884      9.605327 3.734867   7.560533     2.487893 4.601695
5 37.32548  4.833405 336384.99  11.24368     10.029979 3.641542   7.680514     2.408565 4.638972
6 40.14962  4.924012 121539.47  11.28600     10.144541 3.566376   7.546790     2.449946 4.640733
7 34.25926  4.740741 730381.94  10.98148     10.092593 3.731481   7.629630     2.592593 4.370370
       sex capital-gain capital-loss hours-per-week native-country   salary
1 1.654178    1189.2043     87.57507       40.49610       37.75265 1.250418
2 1.683387     932.5750     76.35113       40.06688       37.03533 1.215750
3 1.669657     987.3826     73.51137       41.28709       38.52996 1.214034
4 1.722760    1096.4492     64.19492       39.53511       36.84867 1.217918
5 1.700642    1058.4595     90.23469       40.42355       37.36103 1.245396
6 1.655241    1167.6199    101.19804       40.26587       37.97751 1.251723
7 1.666667     214.2222     52.71296       41.07407       36.36111 1.212963

Clustering vector:
12059 17796  5544 20349 28721  9128 12973 24825 21779  6661 11638 11701 22469 17440 23135 17522 24376
    6     1     5     1     5     6     2     1     3     6     3     2     1     1     6     3     1
13672  5579 25068 28700 17868  9037 15889 30211 11346 31044 22621 28937  5870 20475 32191  4239 10756
```

Kmeans with number of clusters (N)=7 on test data set with 30% of the values:

```
84  a=kmeans(test1.df,7)
85  a
86
```

```
> a=kmeans(test1.df,7)
> a
K-means clustering with 7 clusters of sizes 90, 2252, 1399, 2900, 493, 1033, 1602

Cluster means:
       age workclass    fnlwgt education education-num  marital occupation relationship     race
1 38.11111  4.844444 653238.87  10.48889      9.300000 3.833333   6.988889     2.788889 4.333333
2 39.97158  4.878330 119108.30  11.31217     10.257105 3.594139   7.464032     2.494227 4.682948
3 38.54325  4.884918  50124.53  11.63259     10.205147 3.569693   7.372409     2.354539 4.610436
4 39.03621  4.845862 181726.02  11.27379     10.093448 3.597241   7.644828     2.441724 4.726897
5 35.51116  4.837728 418469.92  11.00406      9.707911 3.636917   7.320487     2.432049 4.636917
6 36.97386  4.859632 321578.91  11.28074     10.060019 3.705712   7.462730     2.432720 4.566312
7 37.55930  4.873283 241006.79  11.17041      9.873283 3.717853   7.705993     2.470037 4.658552
       sex capital-gain capital-loss hours-per-week native-country   salary
1 1.655556     942.9111     36.81111       39.03333       36.53333 1.211111
2 1.664298    1477.3965     92.81483       40.49778       37.89032 1.277975
3 1.669764     590.3152     94.30879       40.89635       38.80629 1.227305
4 1.658276     940.6079     91.78759       40.36793       37.54414 1.254138
5 1.730223    1371.8682     70.34888       40.95538       37.45030 1.231237
6 1.689255    1295.1297     94.74250       40.63601       37.54792 1.224589
7 1.672909     744.6273     81.73845       39.84207       37.12609 1.227840

Clustering vector:
   2    4    9   14   18   29   36   38   43   49   53   55   59   62   64   67   70   73   84   85   90
   3    7    3    4    4    6    7    1    4    2    3    2    2    6    2    4    4    4    4    6    5
  91   93   94  106  108  114  117  125  130  135  136  141  143  145  150  156  162  166  169  173  174
```

Cluster Analysis with number of clusters (N)=3 on training data set with 70% of the values:

```
86  b=iclust(tran1.df, 3)
87  b
```

88:1 (Top Level)

Console D:/download/

```
Item by Cluster Structure matrix:
                  O    P   C12    C2   C10
age             C12  C12  0.34 -0.02  0.15
workclass        C2   C2  0.15  0.41  0.08
fnlwgt          C10  C10 -0.05 -0.02 -0.12
education       C12  C12  0.19  0.00  0.13
education-num   C12  C12  0.41  0.19  0.16
marital         C12  C12 -0.35 -0.08 -0.17
occupation       C2   C2  0.12  0.41  0.00
relationship    C12  C12 -0.59 -0.19 -0.17
race            C10  C10  0.13  0.07  0.29
sex             C12  C12  0.46  0.21  0.07
capital-gain    C12  C12  0.19  0.07  0.03
capital-loss    C12  C12  0.13  0.04  0.03
hours-per-week  C12  C12  0.39  0.25  0.07
native-country  C10  C10  0.04 -0.02  0.32
salary          C12  C12  0.60  0.15  0.13
```

Cluster Analysis with number of clusters (N)=3 on test data set with 30% of the values:

```
87  b=iclust(test1.df, 3)
```

89:1 (Top Level)

Console D:/download/

```
Item by Cluster Structure matrix:
                  O    P   C11   C12    C4
age             C11  C11  0.38  0.09 -0.02
workclass        C4   C4  0.17  0.10  0.42
fnlwgt          C12  C12 -0.04 -0.13 -0.01
education       C12  C12  0.07  0.43  0.01
education-num   C12  C12  0.28  0.51  0.22
marital         C11  C11 -0.38 -0.15 -0.12
occupation       C4   C4  0.11  0.05  0.41
relationship    C11  C11 -0.64 -0.18 -0.22
race            C12  C12  0.13  0.16  0.06
sex             C11  C11  0.53  0.04  0.22
capital-gain    C11  C11  0.18  0.09  0.07
capital-loss    C11  C11  0.13  0.10  0.03
hours-per-week  C11  C11  0.40  0.20  0.29
native-country  C12  C12  0.01  0.22 -0.03
salary          C11  C11  0.56  0.35  0.16
```

Cluster Analysis with number of clusters (N)=5 on training data set with 70% of the values:

```
88  b=iclust(tran1.df, 5)
89  b
```

90:1 (Top Level)

Console D:/download/

```
Item by Cluster Structure matrix:
                  O   P   V12    C9    C4    C2   C10
age              C9 V12  0.37  0.37  0.02 -0.02  0.15
workclass        C2  C2  0.08  0.16  0.06  0.41  0.08
fnlwgt          C10 C10 -0.01 -0.04 -0.06 -0.02 -0.12
education        C4  C4  0.13  0.03  0.48  0.00  0.13
education-num    C4  C4  0.55  0.19  0.58  0.19  0.16
marital          C9  C9 -0.29 -0.37 -0.11 -0.08 -0.17
occupation       C2  C2  0.13  0.10  0.08  0.41  0.00
relationship     C9  C9 -0.42 -0.69 -0.09 -0.19 -0.17
race            C10 C10  0.12  0.14  0.04  0.07  0.29
sex              C9  C9  0.35  0.57 -0.02  0.21  0.07
capital-gain     C9 V12  0.25  0.14  0.15  0.07  0.03
capital-loss    V12 V12  0.23  0.09  0.09  0.04  0.03
hours-per-week   C9  C9  0.37  0.37  0.19  0.25  0.07
native-country  C10 C10  0.02  0.00  0.10 -0.02  0.32
salary           C9 V12  0.54  0.53  0.40  0.15  0.13
```

Cluster Analysis with number of clusters (N)=5 on test data set with 30% of the values:

```
88  b=iclust(test1.df, 5)
89  b
```

90:1 (Top Level)

Console D:/download/

```
Item by Cluster Structure matrix:
                  O   P    C9   V12    C1    C4   C10
age              C9 V12  0.36  0.39  0.04 -0.02 -0.13
workclass        C4  C4  0.19  0.09  0.09  0.42 -0.08
fnlwgt          C10 C10 -0.04 -0.05 -0.09 -0.01  0.13
education        C1  C1  0.05  0.11  0.47  0.01 -0.19
education-num    C1  C1  0.20  0.52  0.57  0.22 -0.22
marital          C9  C9 -0.38 -0.33 -0.09 -0.12  0.18
occupation       C4  C4  0.11  0.11  0.10  0.41  0.03
relationship     C9  C9 -0.69 -0.38 -0.12 -0.22  0.21
race            C10 C10  0.14  0.11  0.06  0.06 -0.26
sex              C9  C9  0.57  0.33  0.00  0.22 -0.09
capital-gain     C9 V12  0.15  0.24  0.14  0.07  0.03
capital-loss    V12 V12  0.09  0.25  0.09  0.03 -0.07
hours-per-week   C9  C9  0.39  0.37  0.21  0.29 -0.11
native-country  C10 C10  0.01  0.02  0.12 -0.03 -0.28
salary           C9  C9  0.54  0.54  0.39  0.16 -0.15
```

Cluster Analysis with number of clusters (N)=7 on training data set with 70% of the values:

```
88  b=iclust(tran1.df, 7)
89  b
```

90:1 (Top Level)

Console D:/download/

```
Item by Cluster Structure matrix:
                 O    P   V12    C1    V3    C8    C4    C2    C3
age             C8   C8  0.37  0.26 -0.37  0.44  0.02 -0.02  0.36
workclass       C2   C2  0.08  0.13  0.02  0.23  0.06  0.41  0.23
fnlwgt          V3   V3 -0.01  0.02  0.30 -0.05 -0.06 -0.02 -0.11
education       C4   C4  0.13 -0.02 -0.11  0.13  0.48  0.00  0.11
education-num   C4   C4  0.55  0.07 -0.19  0.36  0.58  0.19  0.22
marital         C8   V3 -0.29 -0.22  0.67 -0.36 -0.11 -0.08 -0.31
occupation      C2   C2  0.13  0.11  0.06  0.11  0.08  0.41  0.02
relationship    C1   C1 -0.42 -0.71  0.60 -0.51 -0.09 -0.19 -0.40
race            C3   V3  0.12  0.15 -0.40  0.13  0.04  0.07  0.20
sex             C1   C1  0.35  0.68 -0.19  0.42 -0.02  0.21  0.19
capital-gain    C8  V12  0.25  0.07 -0.18  0.19  0.15  0.07  0.07
capital-loss   V12  V12  0.23  0.08 -0.12  0.05  0.09  0.04  0.08
hours-per-week  C8   C8  0.37  0.34 -0.13  0.41  0.19  0.25  0.21
native-country  C3   C3  0.02 -0.01 -0.07  0.01  0.10 -0.02  0.66
salary          C8   C8  0.54  0.34 -0.56  0.67  0.40  0.15  0.26
```

Cluster Analysis with number of clusters (N)=7 on test data set with 30% of the values:

```
88  b=iclust(test1.df, 7)
89  b
```

90:1 (Top Level)

Console D:/download/

```
Item by Cluster Structure matrix:
                 O    P   V12    V3    C2    C8    C1    C4    C5
age             C8   C8  0.39 -0.38  0.25  0.40  0.04 -0.02  0.32
workclass       C4   C4  0.09 -0.01  0.15  0.27  0.09  0.42  0.19
fnlwgt          V3   V3 -0.05  0.28  0.01 -0.03 -0.09 -0.01 -0.12
education       C1   C1  0.11 -0.17  0.00  0.12  0.47  0.01  0.19
education-num   C1   C1  0.52 -0.24  0.08  0.34  0.57  0.22  0.31
marital         C8   V3 -0.33  0.68 -0.24 -0.38 -0.09 -0.12 -0.32
occupation      C4   C4  0.11  0.12  0.12  0.14  0.10  0.41  0.00
relationship    C2   C2 -0.38  0.67 -0.71 -0.51 -0.12 -0.22 -0.42
race            C5   V3  0.11 -0.36  0.15  0.11  0.06  0.06  0.19
sex             C2   C2  0.33 -0.19  0.68  0.42  0.00  0.22  0.21
capital-gain    C8  V12  0.24 -0.17  0.08  0.19  0.14  0.07 -0.04
capital-loss   V12  V12  0.25 -0.16  0.07  0.07  0.09  0.03  0.13
hours-per-week  C8   C8  0.37 -0.20  0.35  0.43  0.21  0.29  0.17
native-country  C5   C5  0.02 -0.03  0.01  0.03  0.12 -0.03  0.53
salary          C8   C8  0.54 -0.64  0.33  0.67  0.39  0.16  0.28
```

K nearest neighbor on training data set with 70% of the values and test data set with 30% of the values, with N=3

```
Total Observations in Table:  9769
```

test[, ncol(test)]	result 1	2	Row Total
1	6508	873	7381
	34.243	156.041	
	0.882	0.118	0.756
	0.812	0.497	
	0.666	0.089	
2	1503	885	2388
	105.841	482.304	
	0.629	0.371	0.244
	0.188	0.503	
	0.154	0.091	
Column Total	8011	1758	9769
	0.820	0.180	

K nearest neighbor on training data set with 70% of the values and test data set with 30% of the values, with N=5

test[, ncol(test)]	result 1	2	Row Total
1	6783	598	7381
	27.378	171.622	
	0.919	0.081	0.756
	0.805	0.445	
	0.694	0.061	
2	1642	746	2388
	84.622	530.461	
	0.688	0.312	0.244
	0.195	0.555	
	0.168	0.076	
Column Total	8425	1344	9769
	0.862	0.138	

K nearest neighbor on training data set with 70% of the values and test data set with 30% of the values, with N=7

	result		
test[, ncol(test)]	1	2	Row Total
1	6961	420	7381
	23.512	189.956	
	0.943	0.057	0.756
	0.801	0.390	
	0.713	0.043	
2	1732	656	2388
	72.674	587.130	
	0.725	0.275	0.244
	0.199	0.610	
	0.177	0.067	
Column Total	8693	1076	9769
	0.890	0.110	

K nearest neighbor on training data set with 60% of the values and test data set with 40% of the values, with N=3

Total Observations in Table: 13025

	result		
test[, ncol(test)]	1	2	Row Total
1	8701	1130	9831
	39.487	189.303	
	0.885	0.115	0.755
	0.807	0.503	
	0.668	0.087	
2	2076	1118	3194
	121.540	582.667	
	0.650	0.350	0.245
	0.193	0.497	
	0.159	0.086	
Column Total	10777	2248	13025
	0.827	0.173	

K nearest neighbor on training data set with 60% of the values and test data set with 40% of the values, with N=5

	result		
test[, ncol(test)]	1	2	Row Total
1	9096	768	9864
	34.859	226.836	
	0.922	0.078	0.757
	0.806	0.443	
	0.698	0.059	
2	2194	967	3161
	108.779	707.848	
	0.694	0.306	0.243
	0.194	0.557	
	0.168	0.074	
Column Total	11290	1735	13025
	0.867	0.133	

K nearest neighbor on training data set with 60% of the values and test data set with 40% of the values, with N=7

	result		
test[, ncol(test)]	1	2	Row Total
1	9320	544	9864
	28.472	240.259	
	0.945	0.055	0.757
	0.800	0.394	
	0.716	0.042	
2	2325	836	3161
	88.848	749.736	
	0.736	0.264	0.243
	0.200	0.606	
	0.179	0.064	
Column Total	11645	1380	13025
	0.894	0.106	

K nearest neighbor on training data set with 50% of the values and test data set with 50% of the values, with N=3

	result		
test[, ncol(test)]	1	2	Row Total
1	10848	1463	12311
	46.945	221.422	
	0.881	0.119	0.756
	0.808	0.514	
	0.666	0.090	
2	2585	1385	3970
	145.576	686.632	
	0.651	0.349	0.244
	0.192	0.486	
	0.159	0.085	
Column Total	13433	2848	16281
	0.825	0.175	

K nearest neighbor on training data set with 50% of the values and test data set with 50% of the values, with N=5

	result		
test[, ncol(test)]	1	2	Row Total
1	11453	858	12311
	38.801	278.490	
	0.930	0.070	0.756
	0.801	0.431	
	0.703	0.053	
2	2837	1133	3970
	120.324	863.598	
	0.715	0.285	0.244
	0.199	0.569	
	0.174	0.070	
Column Total	14290	1991	16281
	0.878	0.122	

K nearest neighbor on training data set with 50% of the values and test data set with 50% of the values, with N=7

	result		
test[, ncol(test)]	1	2	Row Total
1	11703	608	12311
	36.446	324.490	
	0.951	0.049	0.756
	0.800	0.370	
	0.719	0.037	
2	2934	1036	3970
	113.020	1006.246	
	0.739	0.261	0.244
	0.200	0.630	
	0.180	0.064	
Column Total	14637	1644	16281
	0.899	0.101	

3.c Table (answer of 6.b)

Table matrix is as follows:

Kmeans accuracy

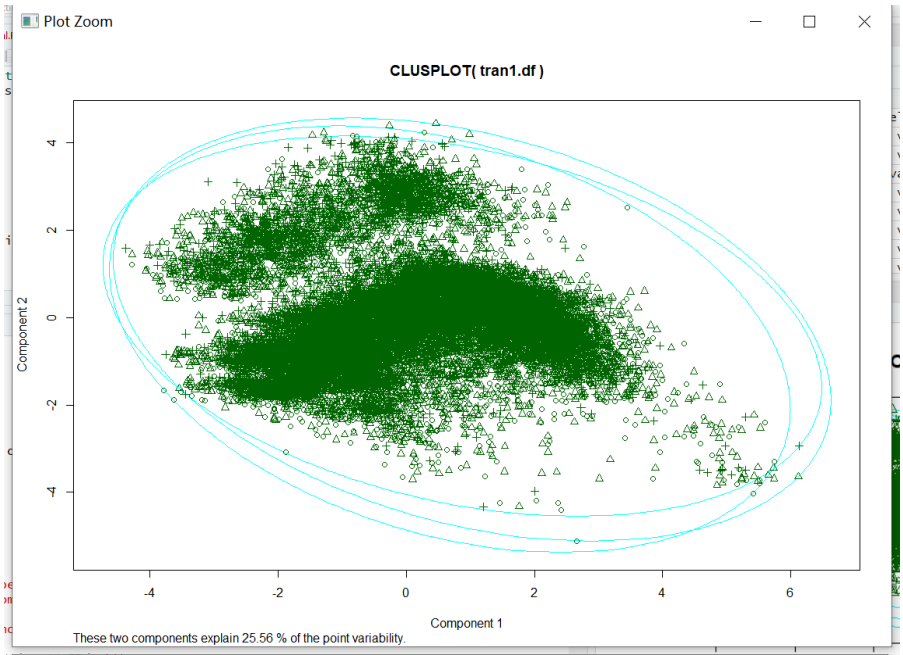
	3	5	7
70-30	77.5%	89.0%	92.4%
60-40	77.2%	89.0%	93.9%
50-50	78.0%	89.3%	94.1%

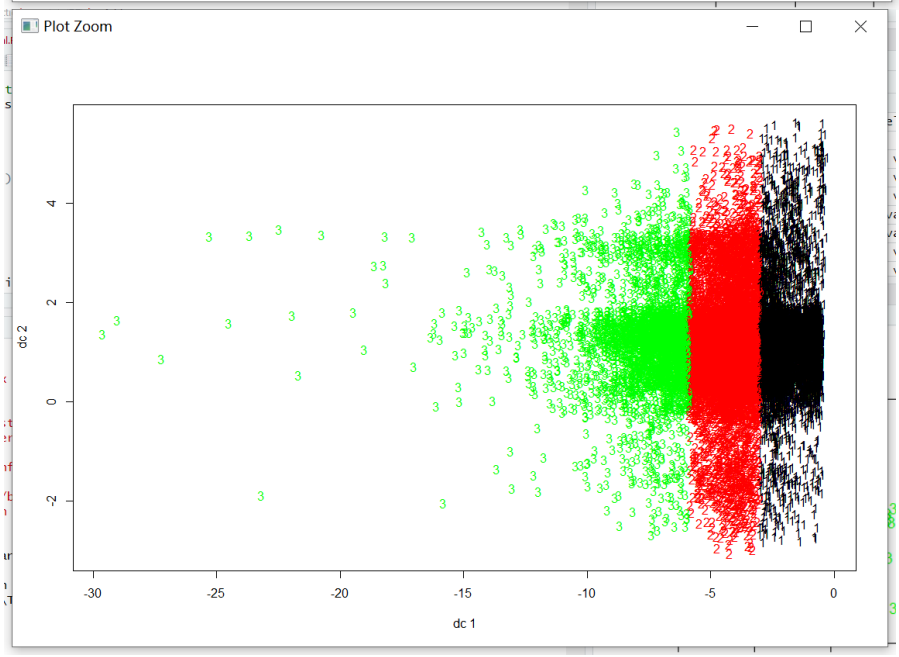
Knn miss rate

	3	5	7
70-30	0.2503839	0.2297062	0.2204934
60-40	0.2459885	0.2284837	0.2148944
50-50	0.243474	0.2259689	0.2174314

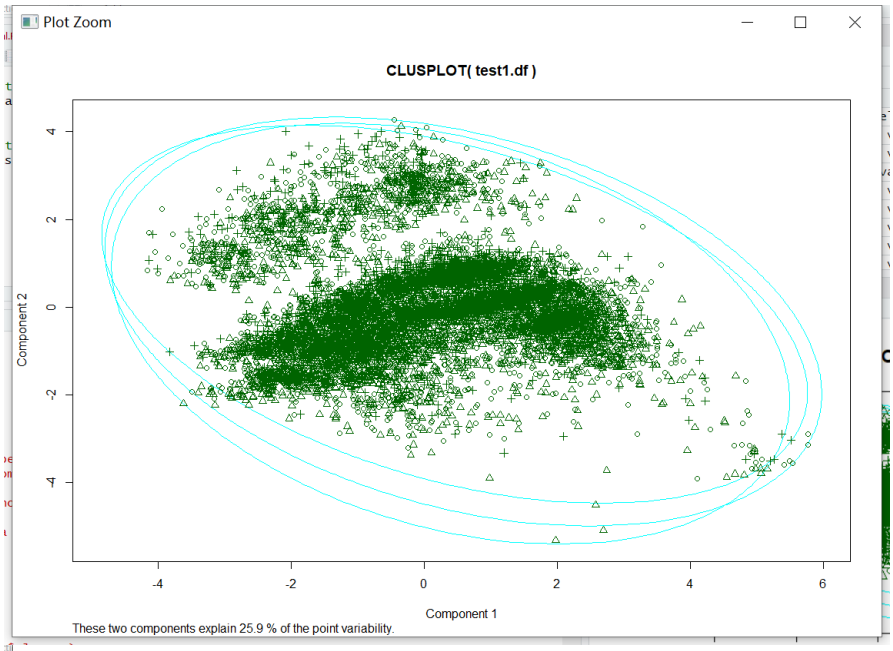
3.d Plot the result

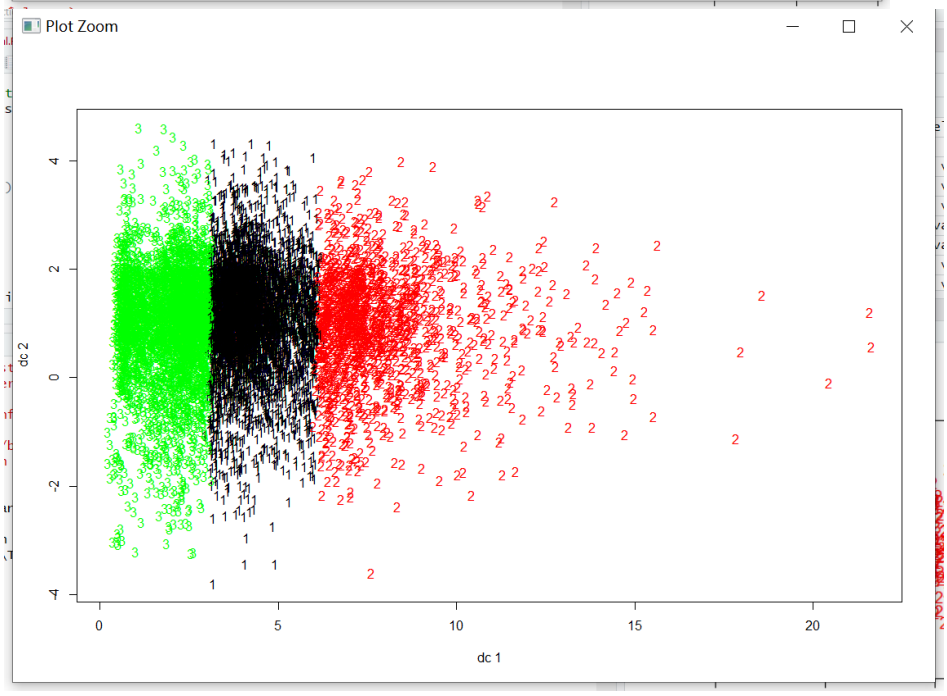
Kmeans with number of clusters (N)=3 on training data set with 70% of the values:



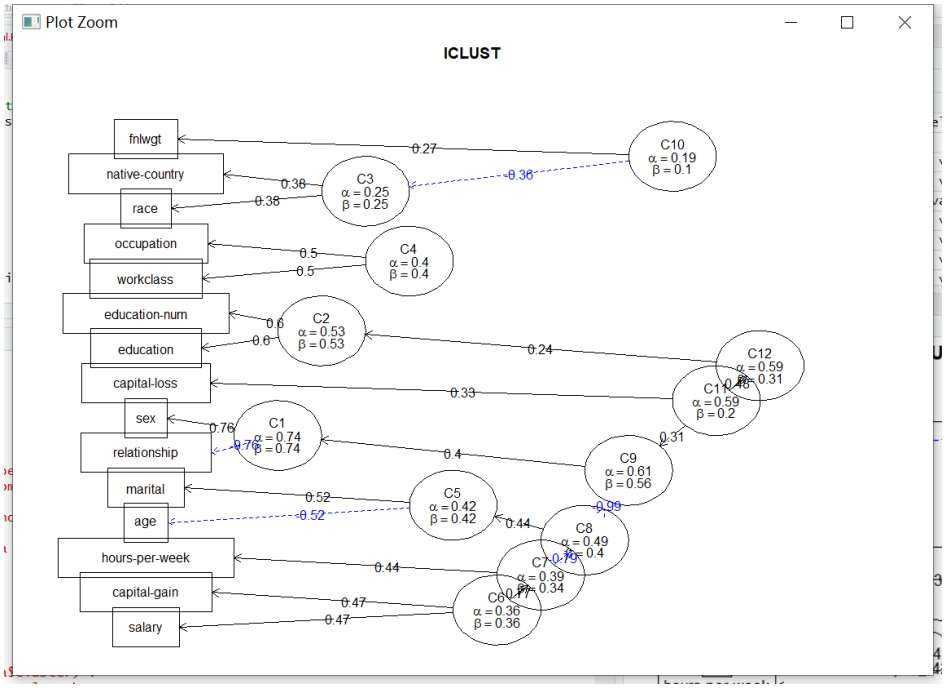


Kmeans with number of clusters (N)=3 on test data set with 30% of the values:

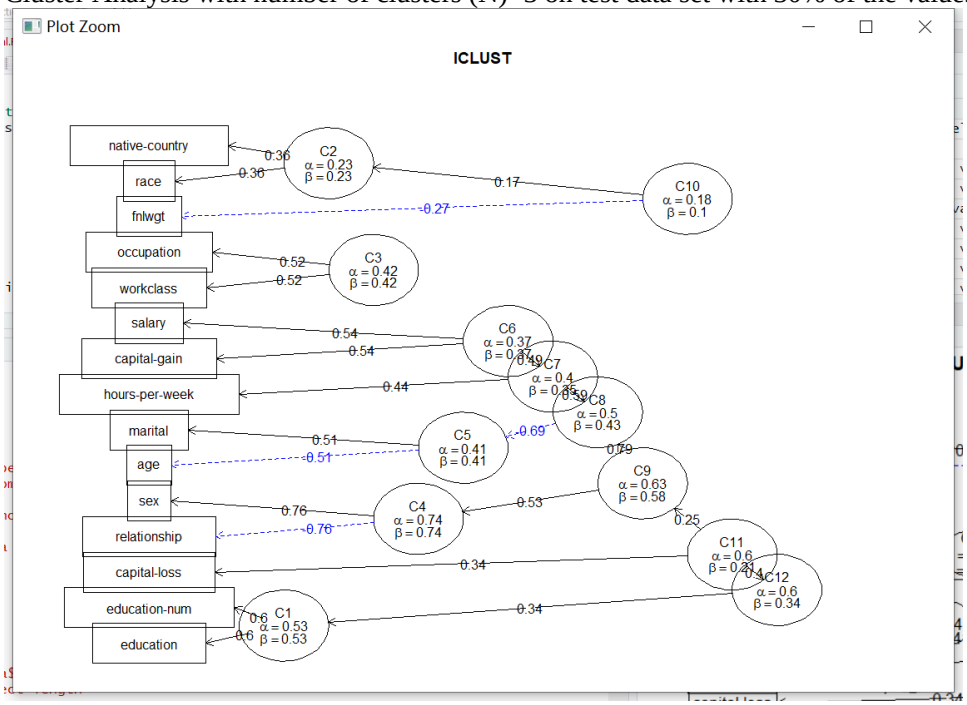




Cluster Analysis with number of clusters (N)=3 on training data set with 70% of the values:



Cluster Analysis with number of clusters (N)=3 on test data set with 30% of the values:



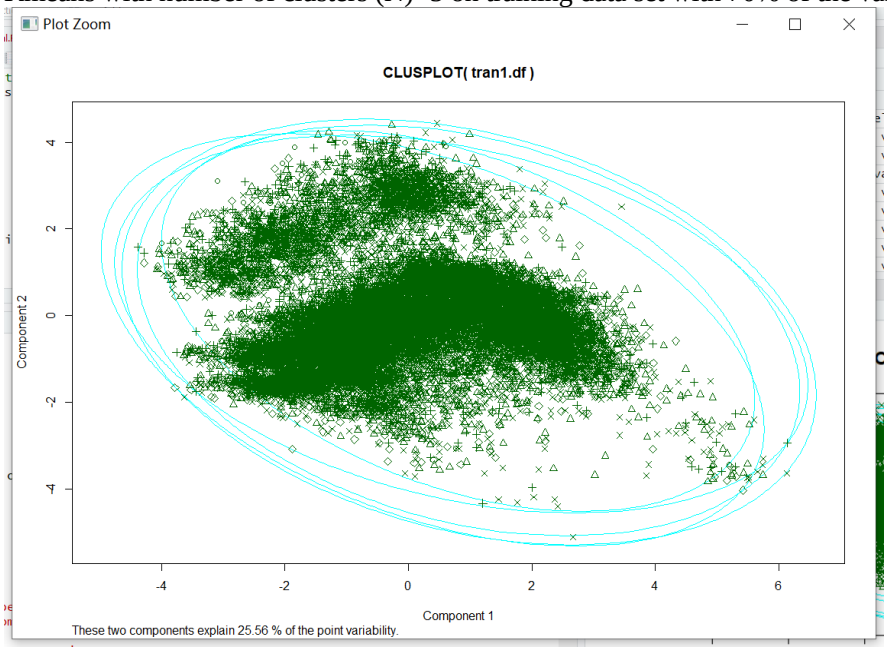
K nearest neighbor on training data set with 70% of the values and test data set with 30% of the values, with N=3

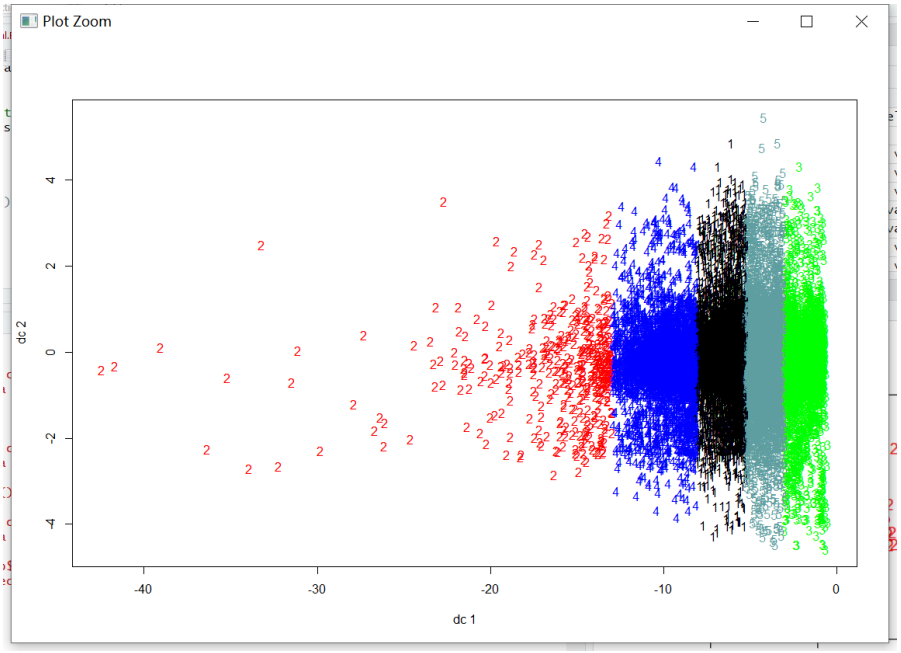
```
   Cell Contents
|-------------------------|
|                       N |
| Chi-square contribution |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|

Total Observations in Table:  9769
```

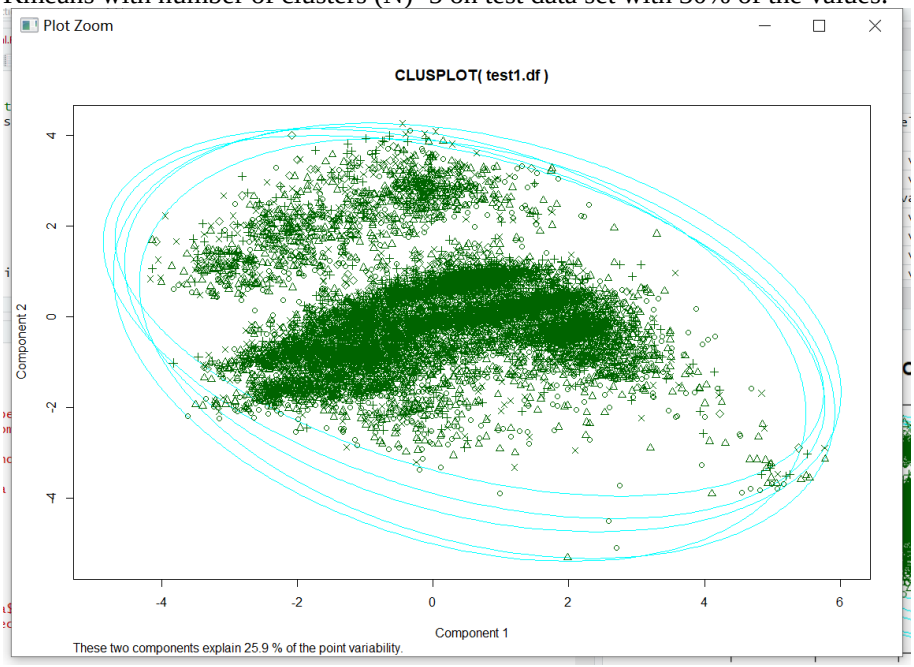
test[, ncol(test)]	result 1	2	Row Total
1	6469	886	7355
	32.660	148.004	
	0.880	0.120	0.753
	0.808	0.502	
	0.662	0.091	
2	1534	880	2414
	99.508	450.940	
	0.635	0.365	0.247
	0.192	0.498	
	0.157	0.090	
Column Total	8003	1766	9769
	0.819	0.181	

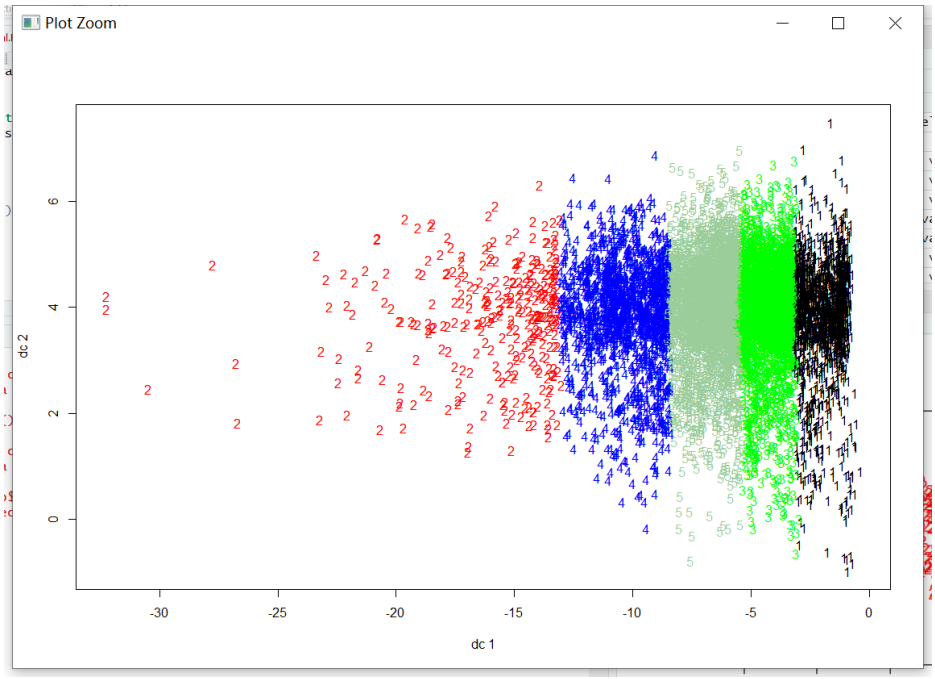
Kmeans with number of clusters (N)=5 on training data set with 70% of the values:



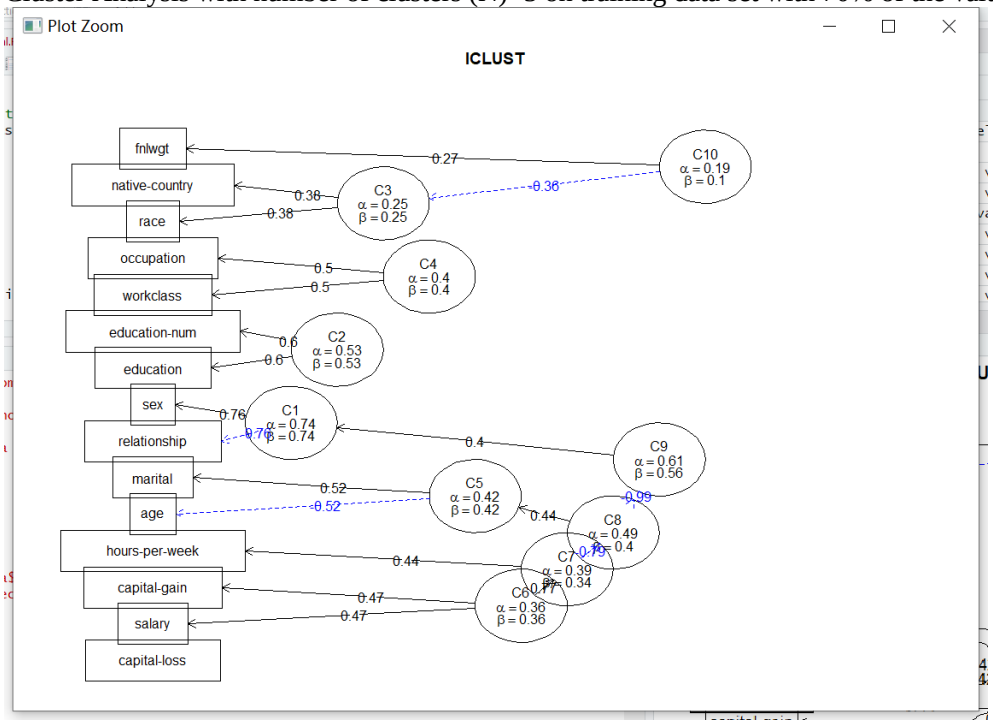


Kmeans with number of clusters (N)=5 on test data set with 30% of the values:

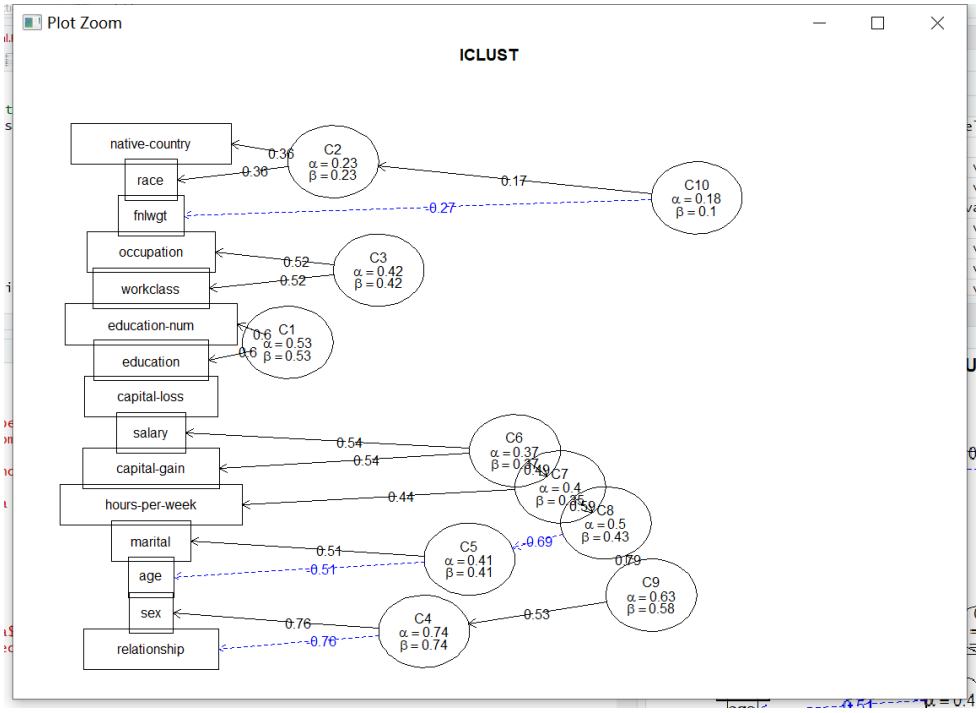




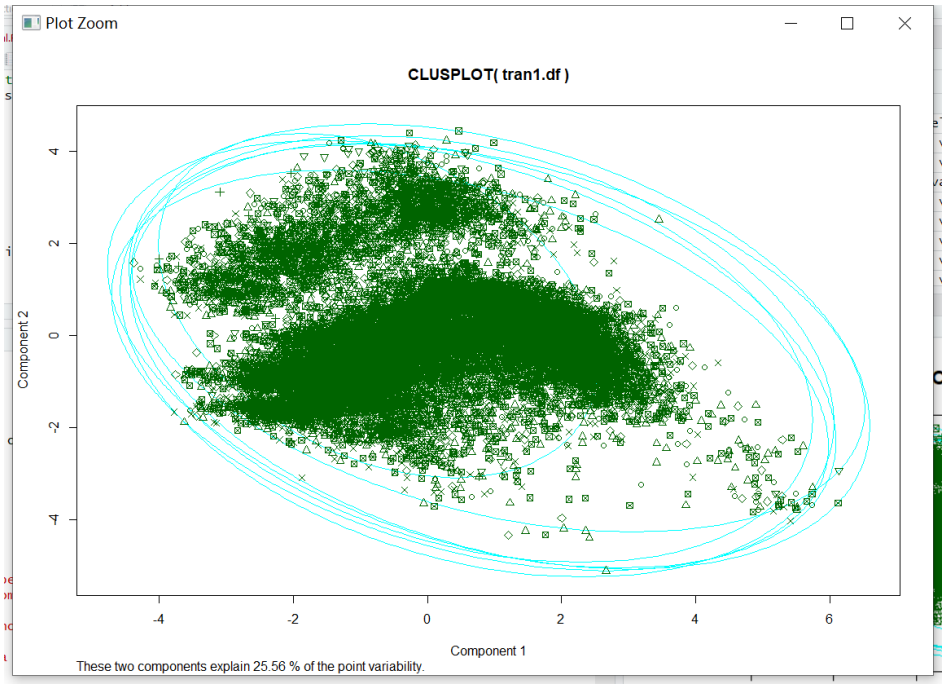
Cluster Analysis with number of clusters (N)=5 on training data set with 70% of the values:

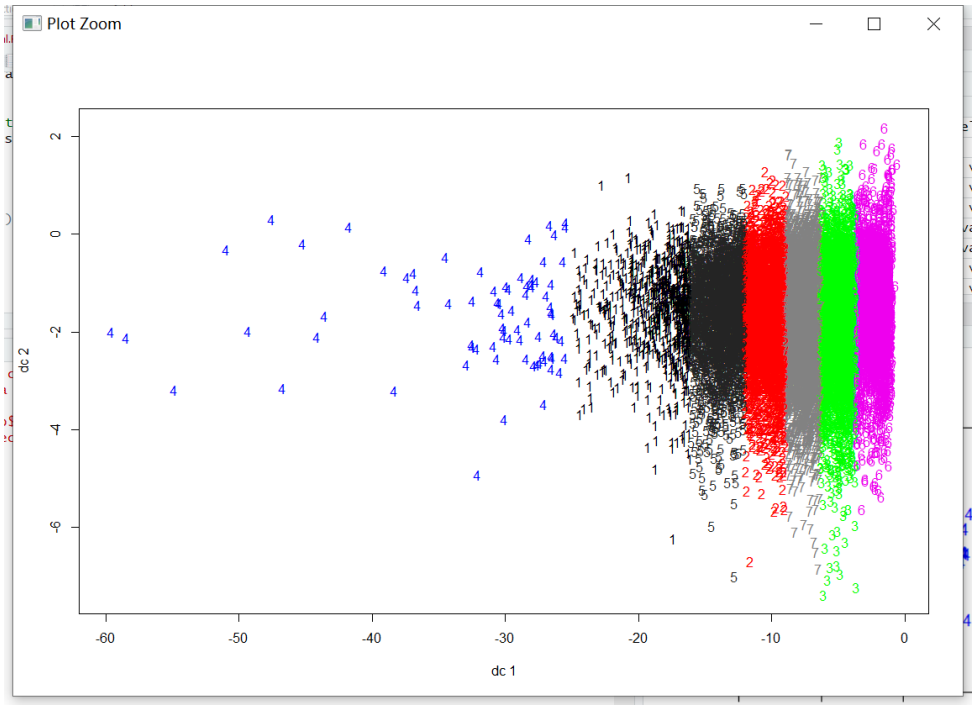


Cluster Analysis with number of clusters (N)=5 on test data set with 30% of the values:

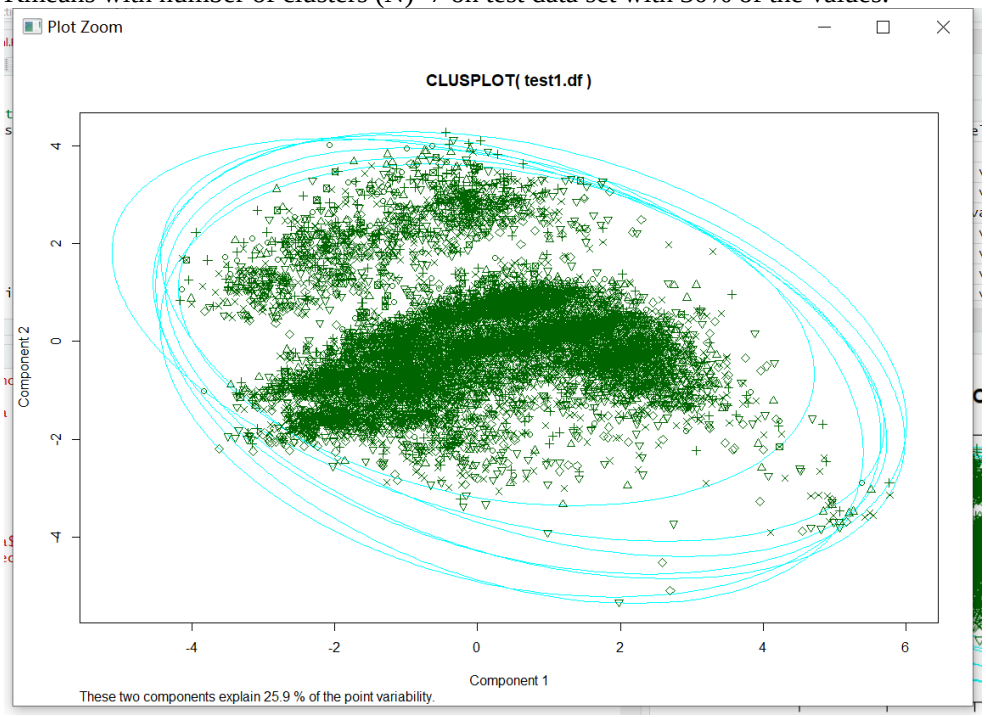


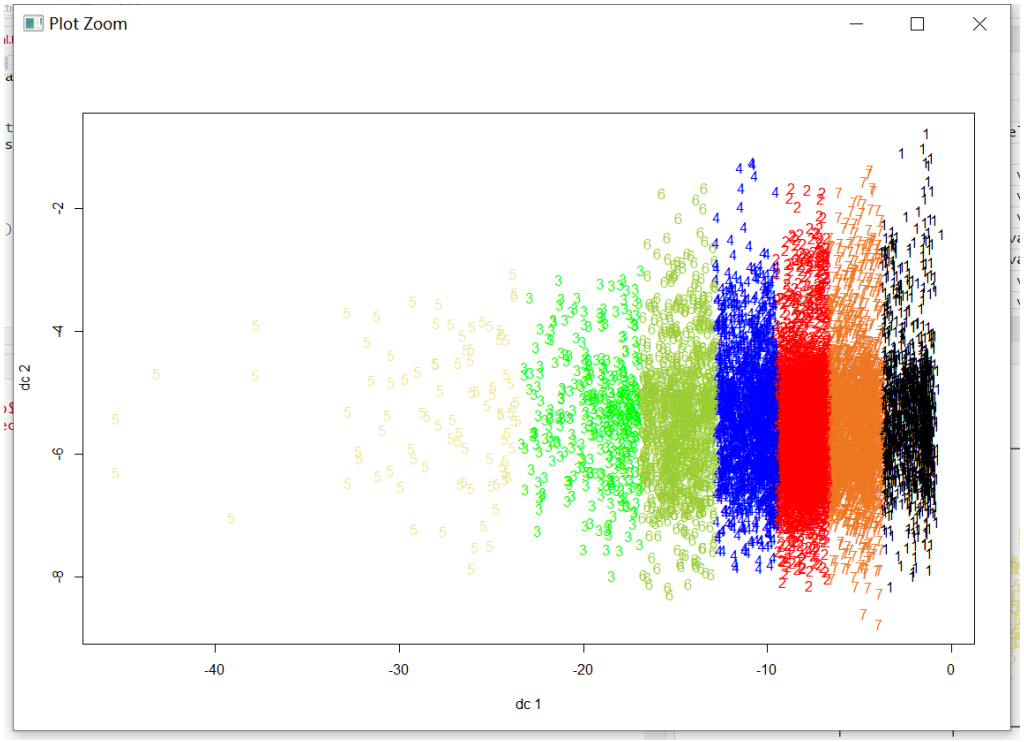
Kmeans with number of clusters (N)=7 on training data set with 70% of the values:



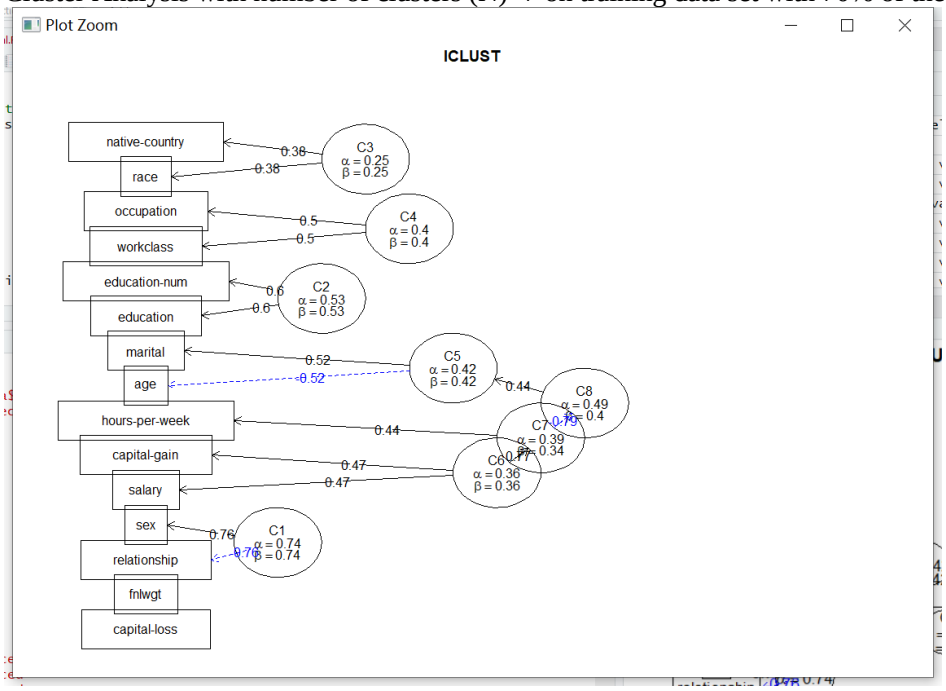


Kmeans with number of clusters (N)=7 on test data set with 30% of the values:

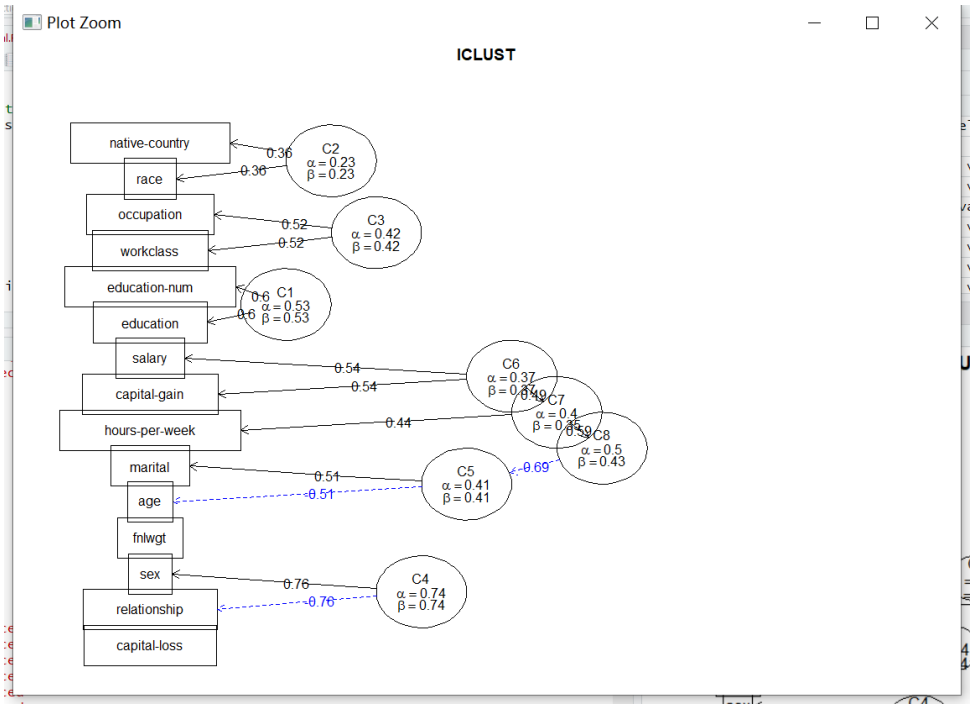




Cluster Analysis with number of clusters (N)=7 on training data set with 70% of the values:



Cluster Analysis with number of clusters (N)=7 on test data set with 30% of the values:



Answer of question 6.c:

Which method is better?

From the plots we get, we find that K-means is a better function, the character of different clustering shown in different style of plot function result, which let us understand the data in a great degree.

What seems to be the best number of clusters for each method?

For all methods, it seems that as the number of clusters increase, the accuracy of knn and k-means increase at the same time. And the result of icluster more clearly to shows the feature of the data. So the best number of cluster in this assignment is N=7

4.

Before fit the data into lm or glm, we made some preprocessing. All features not numeric are turned into numeric. Firstly, we need to know what features are related, what are not. So, we did PCA analysis of all features:

```
Importance of components:
                          Comp.1    Comp.2     Comp.3     Comp.4     Comp.5     Comp.6     Comp.7    Comp.8     Comp.9
Standard deviation     1.4558615 1.1882003 1.12203820 1.06004705 1.03591158 1.01489305 0.97515184 0.9635222 0.92564900
Proportion of Variance 0.1513952 0.1008443 0.08992641 0.08026427 0.07665091 0.07357199 0.06792294 0.0663125 0.06120186
Cumulative Proportion  0.1513952 0.2522395 0.34216588 0.42243015 0.49908107 0.57265306 0.64057599 0.7068885 0.76809035
                          Comp.10    Comp.11    Comp.12    Comp.13    Comp.14
Standard deviation     0.91909555 0.86506392 0.82528819 0.76716282 0.61969626
Proportion of Variance 0.06033833 0.05345254 0.04865004 0.04203849 0.02743025
Cumulative Proportion  0.82842868 0.88188123 0.93053127 0.97256975 1.00000000
```

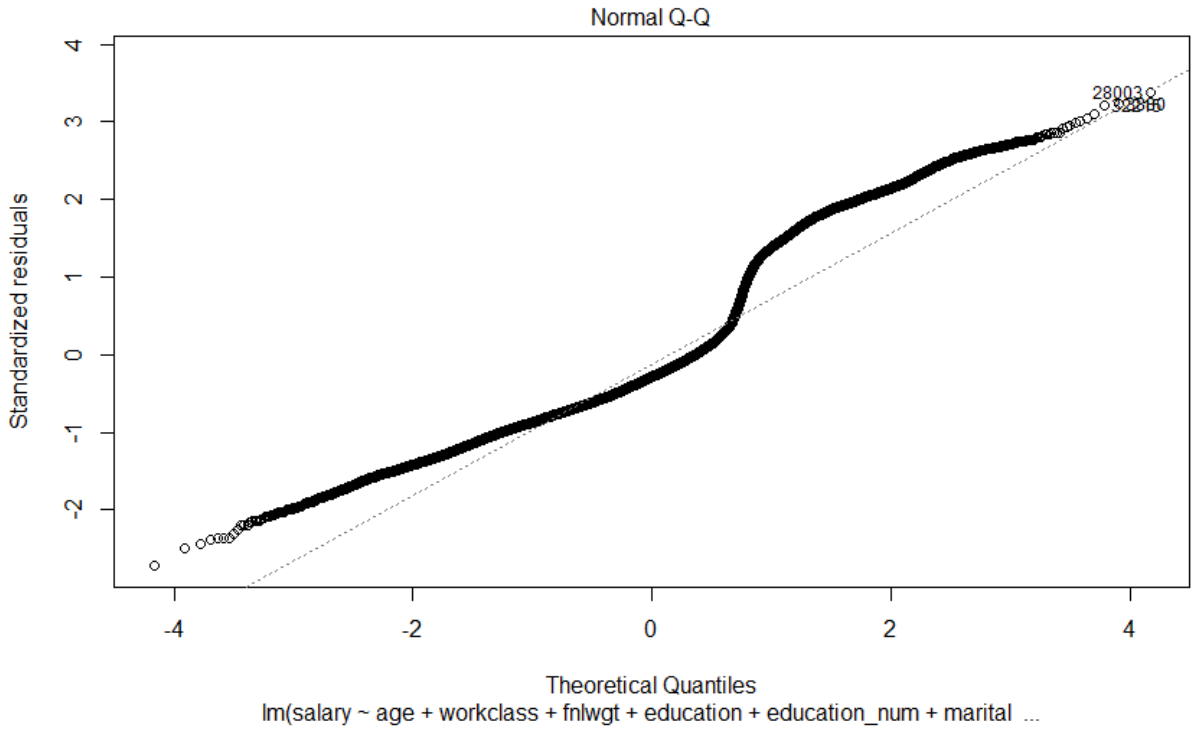
From this figure we can know that 13 features have got 97% cummulative proportion. Therefore, we can omit the 14th feature.

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8	Comp.9	Comp.10	Comp.11	Comp.12	Comp.13	Comp.14
age	0.285	0.126	0.396	0.193	0.398		0.157		0.410			0.539	0.108	0.224
workclass	0.211		-0.530	-0.129	0.332			0.123			0.684		-0.190	
fnlwgt		0.126	-0.192	0.152	-0.516	-0.124	0.729	0.236	0.136	0.122		0.137		
education	0.103	-0.605	0.132		-0.245		-0.114	0.251	0.166	-0.106	0.260	-0.100	0.587	
education_num	0.218	-0.609		0.137	-0.137				0.127		-0.228		-0.672	
marital	-0.322		-0.273	-0.108	-0.344		-0.320	-0.409	0.168		0.117	0.610		
occupation	0.165	-0.114	-0.592		0.266				0.336	0.132	-0.548		0.298	
relationship	-0.521	-0.234			0.235		0.199		-0.222					0.720
race	0.158		0.116	-0.633		-0.105	0.255			-0.666	-0.145			
sex	0.459	0.293			-0.360		-0.249					-0.242		0.647
capital_gain	0.141	-0.152		0.243		-0.613	0.191	-0.645	-0.108		0.101	-0.162	0.121	
capital_loss	0.111					0.762	0.322	-0.499	-0.111		0.105	-0.107		
hours_per_week	0.376		-0.104					0.134	-0.738		-0.197	0.432	0.184	
native_country		-0.188	0.210	-0.636		-0.105				0.699				

From this figure we can know that the most related to Comp.14 is feature “race”. Therefore, we don’t have to consider “race” feature in this question. Then, we can do lm for 13 features.

```
Coefficients:
   (Intercept)             age       workclass          fnlwgt       education    education_num          marital
    -6.078e-01       4.717e-03      -3.238e-03       6.567e-08      -3.679e-03       4.718e-02      -2.424e-02
    occupation    relationship             sex    capital_gain    capital_loss  hours_per_week  native_country
     2.092e-03      -1.602e-02       1.043e-01       9.275e-06       1.138e-04       3.580e-03       2.106e-04
```

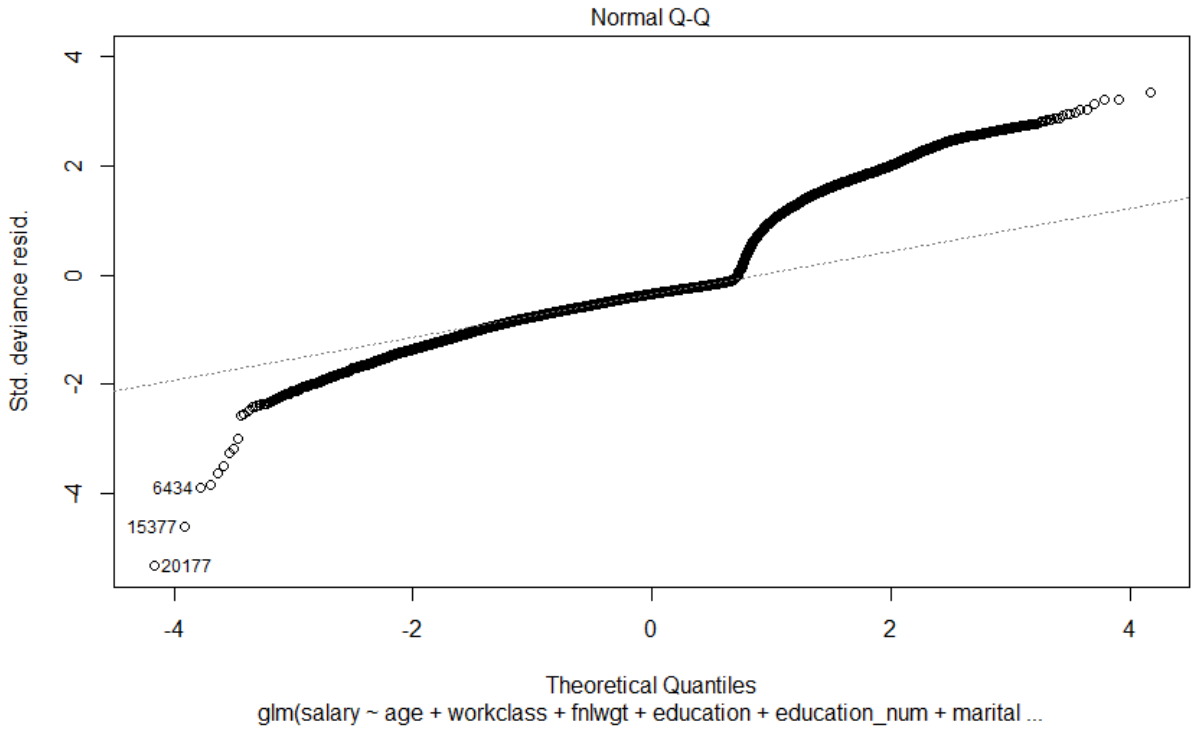
By this regression equation, the output should be around [1, 2]. 1 means “<=50K” and 2 means “>50K”. The regression visualization result Q-Q plot is shown as follows:



For glm, binomimal prior probability assumption:

```
Coefficients:
   (Intercept)             age       workclass          fnlwgt       education    education_num          marital
     -8.585e+00       3.418e-02      -1.962e-02       4.893e-07       1.579e-02       3.319e-01      -2.365e-01
    occupation    relationship             sex    capital_gain    capital_loss  hours_per_week  native_country
     1.048e-02      -1.235e-01       8.983e-01       3.160e-04       6.805e-04       3.003e-02       5.441e-03
```

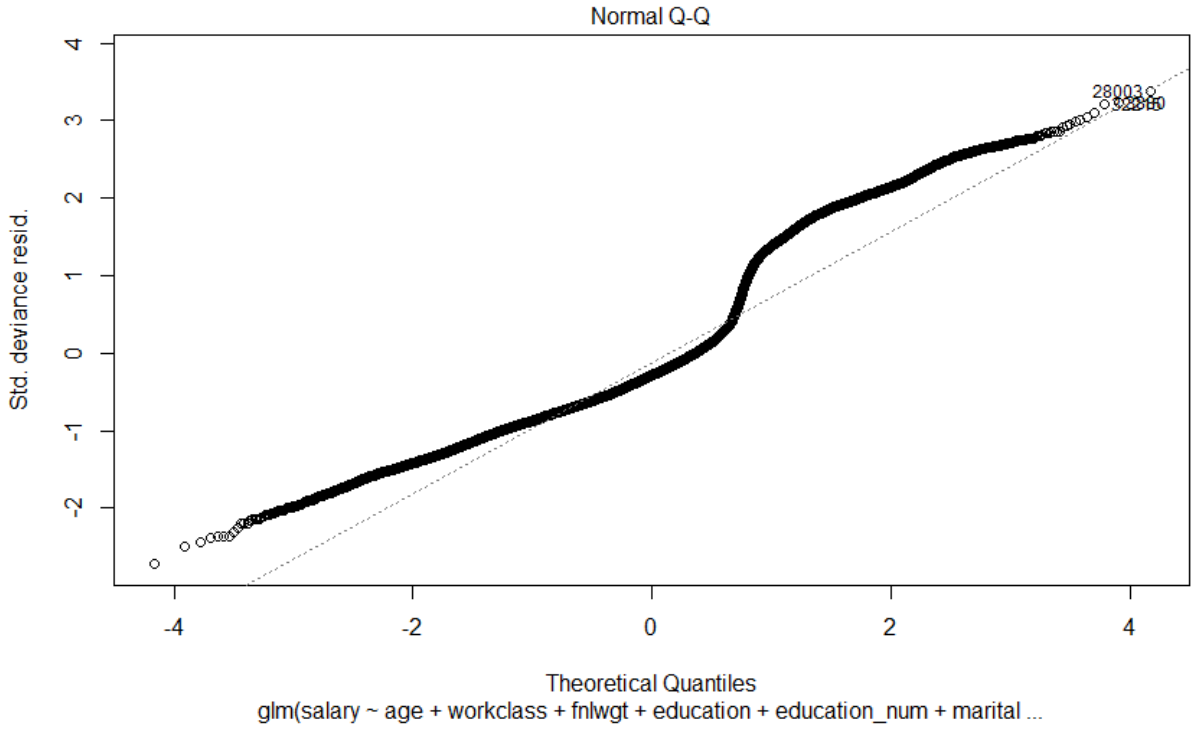
Q-Q plot:



glm, gaussian prior probability assumption:

```
Coefficients:
   (Intercept)             age       workclass          fnlwgt       education    education_num         marital
    -6.078e-01       4.717e-03      -3.238e-03       6.567e-08      -3.679e-03       4.718e-02      -2.424e-02
    occupation    relationship             sex    capital_gain    capital_loss  hours_per_week  native_country
     2.092e-03      -1.602e-02       1.043e-01       9.275e-06       1.138e-04       3.580e-03       2.106e-04
```

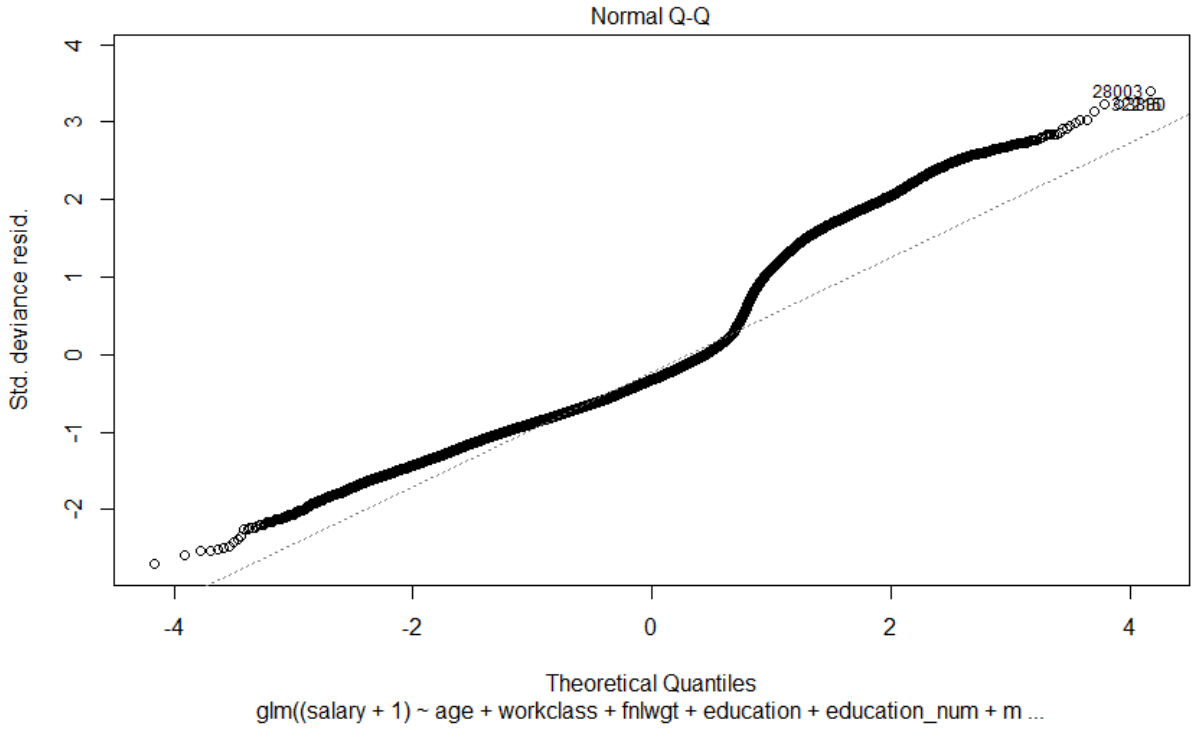
Q-Q plot:



glm, gamma prior probability assumption:

```
Coefficients:
   (Intercept)             age       workclass          fnlwgt       education   education_num         marital
      1.372e+00      -3.036e-03       2.131e-03      -3.733e-08       9.116e-04      -2.895e-02       1.802e-02
    occupation    relationship             sex    capital_gain    capital_loss  hours_per_week  native_country
     -1.137e-03       1.100e-02      -7.123e-02      -2.398e-06      -5.100e-05      -2.323e-03      -3.152e-04
```

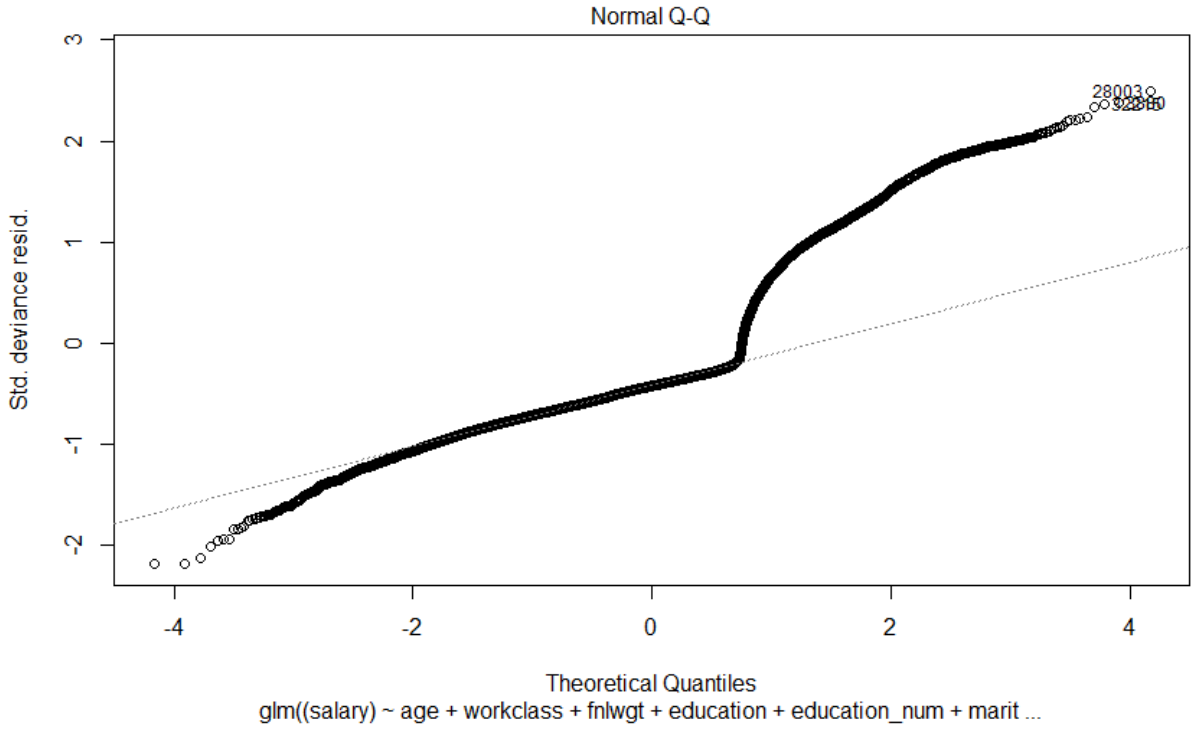
Q-Q plot:



glm, poisson prior probability assumption:

```
Coefficients:
   (Intercept)             age       workclass          fnlwgt       education   education_num          marital
     -5.565e+00       2.029e-02      -1.649e-02       2.131e-07       1.111e-02       1.883e-01      -1.647e-01
    occupation    relationship             sex    capital_gain    capital_loss  hours_per_week  native_country
      6.185e-03      -9.642e-02       5.655e-01       8.317e-06       2.280e-04       1.580e-02       3.634e-03
```

Q-Q plot:



Final summary table (anwser of 6.d):

	age	workclass	fnlwgt	education	education-num	marital	occupation	relationship	sex	capital-gain	capital-loss	hours-per-week	native-country	bias
lm	0.004717	-0.003238	0.000000	-0.003679	0.047180	-0.024240	0.002092	-0.016020	0.104300	0.000009	0.000114	0.003580	0.000211	-0.607800
glm-binominal	0.034180	-0.019620	0.000000	0.015790	0.331900	-0.236500	0.010480	-0.123500	0.898300	0.000316	0.000681	0.030030	0.005441	-8.585000
glm-gaussian	0.004717	-0.003238	0.000000	-0.003679	0.047180	-0.024240	0.002092	-0.016020	0.104300	0.000009	0.000114	0.003580	0.000211	-0.607800
glm-gamma	-0.003036	0.002131	0.000000	0.000912	-0.028950	0.018020	-0.001137	0.011000	-0.071230	-0.000002	-0.000051	-0.002323	-0.000315	1.372000
glm-poisson	0.020290	-0.016490	0.000000	0.011110	0.188300	-0.164700	0.006185	-0.096420	0.565500	0.000008	0.000228	0.015800	0.003634	-5.565000

From the above table and plot we can conclude the following patterns: (answer of 6.e)

1. glm with Gaussian prior probability is equal to lm, or we can say that normally linear regression is assuming that data distribution is Gaussian distribution.
2. fnlwgt, capital-gain, capital-loss and native-country are basically have no direct relation to a man's salary. With the former race feature, we can say that these five features are unrelated features in this researching question.
3. glm with Binominal and Poisson prior probability have very high bias, which means men's salary basically do not allow these 2 features. Since Gaussian distribution's bias is lowest, we can say that men's salary indeed allow Gaussian distribution, which fits our general intuition.
4. Different prior distribution can cause different coefficients for same feature, some even differs largely.

Answer of 6.f:

In all, from this data set analysis work, my teammate and I know that big data analysis can explore valuable information that cannot be reached by see the data itself. Using the clustering technologies such as k-means and knn and principal component analysis, we can filter some useless part of the data and analyze the core data into a new degree and know the connection of each value, the macro view of them. Furthermore, we even can make predictive analyze using such as glm function. We ensure that learning these can help us understand the truly meaning of the data we use.