Q1. Create a file "people.txt" with the following data:

Age	Age Group	Height	Status	Years Married
21	adult	6.0	single	-1
2	child	3	married	0
18	adult	5.7	married	20
221	elderly	5	widowed	2
34	child	-7	married	3

i) Read the data from the file "people.txt".

ii) Create a rule set E that contain rules to check for the following conditions :

- **The age should be in the range 0-150.**
- **The age should be greater than years married.**
- **The status should be married or single or widowed.**
- **If age is less than 18 the age group should be child, if age is between 18 and 65 the age group should be adult, if age is more than 65 the age group should be elderly.**

iii) Check whether rule set E is violated by the data in the file people.txt.

iv) Summarize the results obtained in part (iii).

v) Visualize the results obtained in part (iii).

people.text file:

people - Notepad

File Edit Format View Help

```
age     agegroup    height      status        yearsmarried
21        adult       6.0        single           -1
2         child        3         married           0
18        adult       5.7        married           20
221      elderly      5          widowed           2
34        child       -7         married           3
```

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Q1.r

```
1  install.packages("editrules")
2  library("editrules")
3
4  setwd("C:\\Users\\Lenovo\\Desktop\\DM_files")
5  getwd()
6
7  d <- read.table("people.txt", header = TRUE)
8  attach(d)
9
10 E <- editset(expression(
11     age >= 0,
12     age <= 150,
13     age > yearsmarried,
14     status %in% c('single', 'married', 'widowed'),
15     if (age <= 18) agegroup %in% c('child'),
16     if (age > 18 && age < 65) agegroup %in% c('adult'),
17     if (age >= 65) agegroup %in% c('elderly')
18   ))
19
20 sm <- violatedEdits(E, d)
21 summary(sm)
22 plot(sm)
```

23:1 (Top Level) R Script

Console Terminal Jobs

C:/Users/Lenovo/Desktop/DM_files/

```
>
```

Environment History Connections

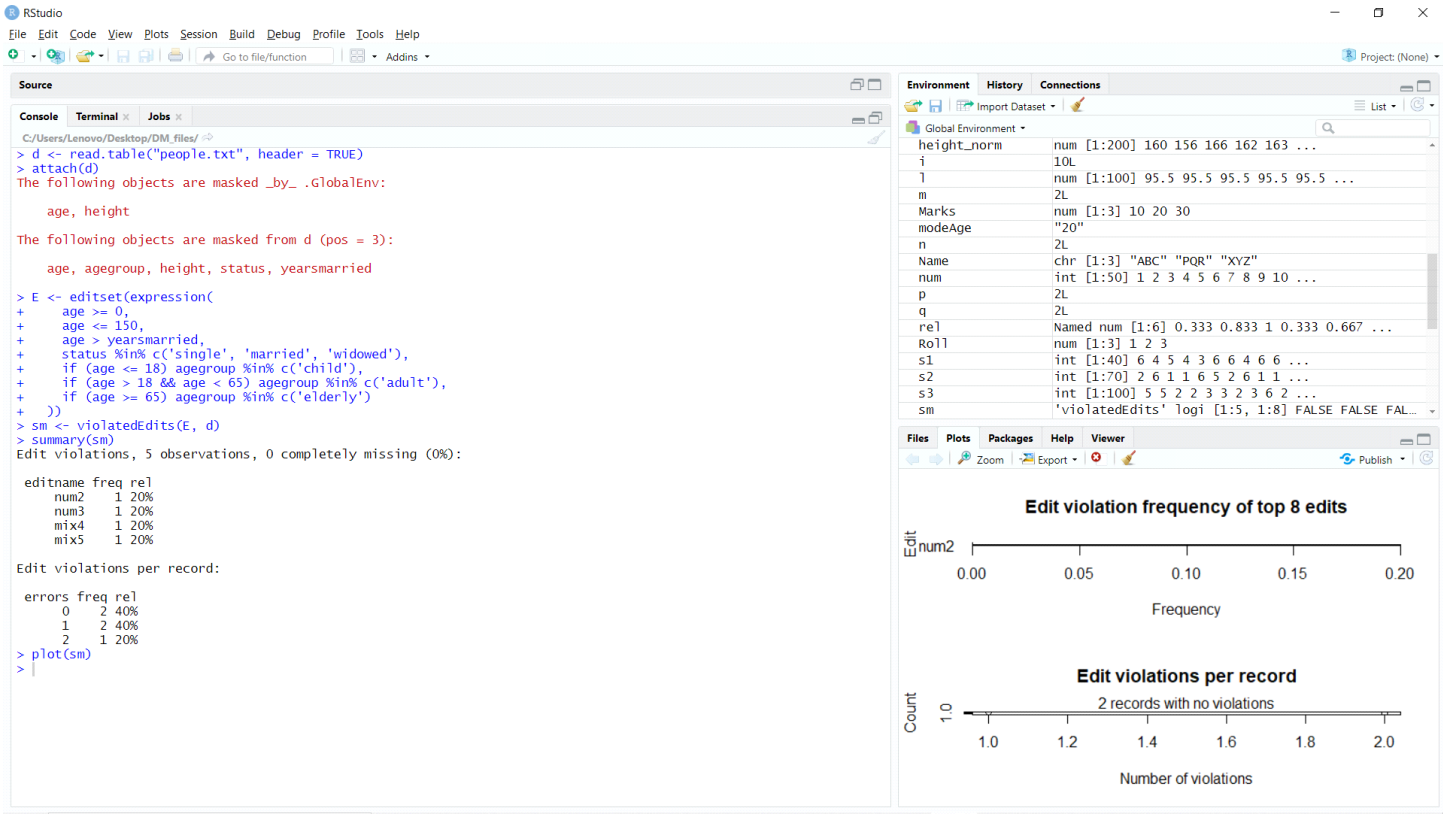
Global Environment

height_norm	num [1:200] 160 156 166 162 163 ...
i	10L
l	num [1:100] 95.5 95.5 95.5 95.5 95.5 ...
m	2L
Marks	num [1:3] 10 20 30
modeAge	"20"
n	2L
Name	chr [1:3] "ABC" "PQR" "XYZ"
num	int [1:50] 1 2 3 4 5 6 7 8 9 10 ...
p	2L
q	2L
rel	Named num [1:6] 0.333 0.833 1 0.333 0.667 ...
Roll	num [1:3] 1 2 3
s1	int [1:40] 6 4 5 4 3 6 6 4 6 6 ...
s2	int [1:70] 2 6 1 1 6 5 2 6 1 1 ...
s3	int [1:100] 5 5 2 2 3 3 2 3 6 2 ...
sm	'violatedEdits' logi [1:5, 1:8] FALSE FALSE FAL...

Files Plots Packages Help Viewer

Zoom Export

Output:



Q2. Perform the following preprocessing tasks on the dirty_iris dataset.

i) Calculate the number and percentage of observations that are complete.

ii) Replace all the special values in data with NA.

iii) Define these rules in a separate text file and read them.

(Use editfile function in R (package editrules). Use similar function in Python).

Print the resulting constraint object.

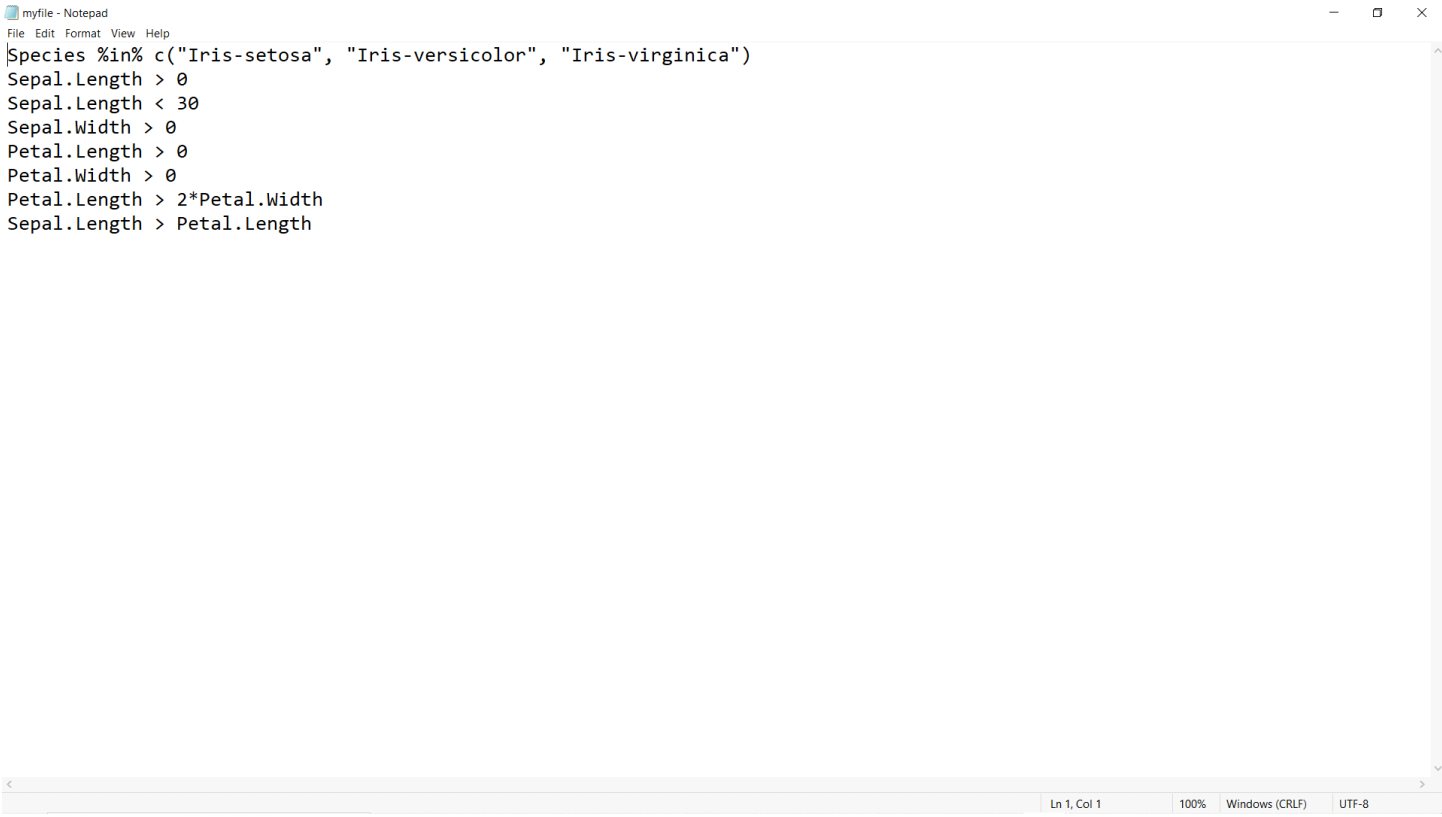
- **– Species should be one of the following values: setosa, versicolor or virginica.**
- **– All measured numerical properties of an iris should be positive.**
- **– The petal length of an iris is atleast 2 times its petal width.**
- **– The sepal length of an iris cannot exceed 30cm.**
- **– The sepals of an iris are longer than its petals.**

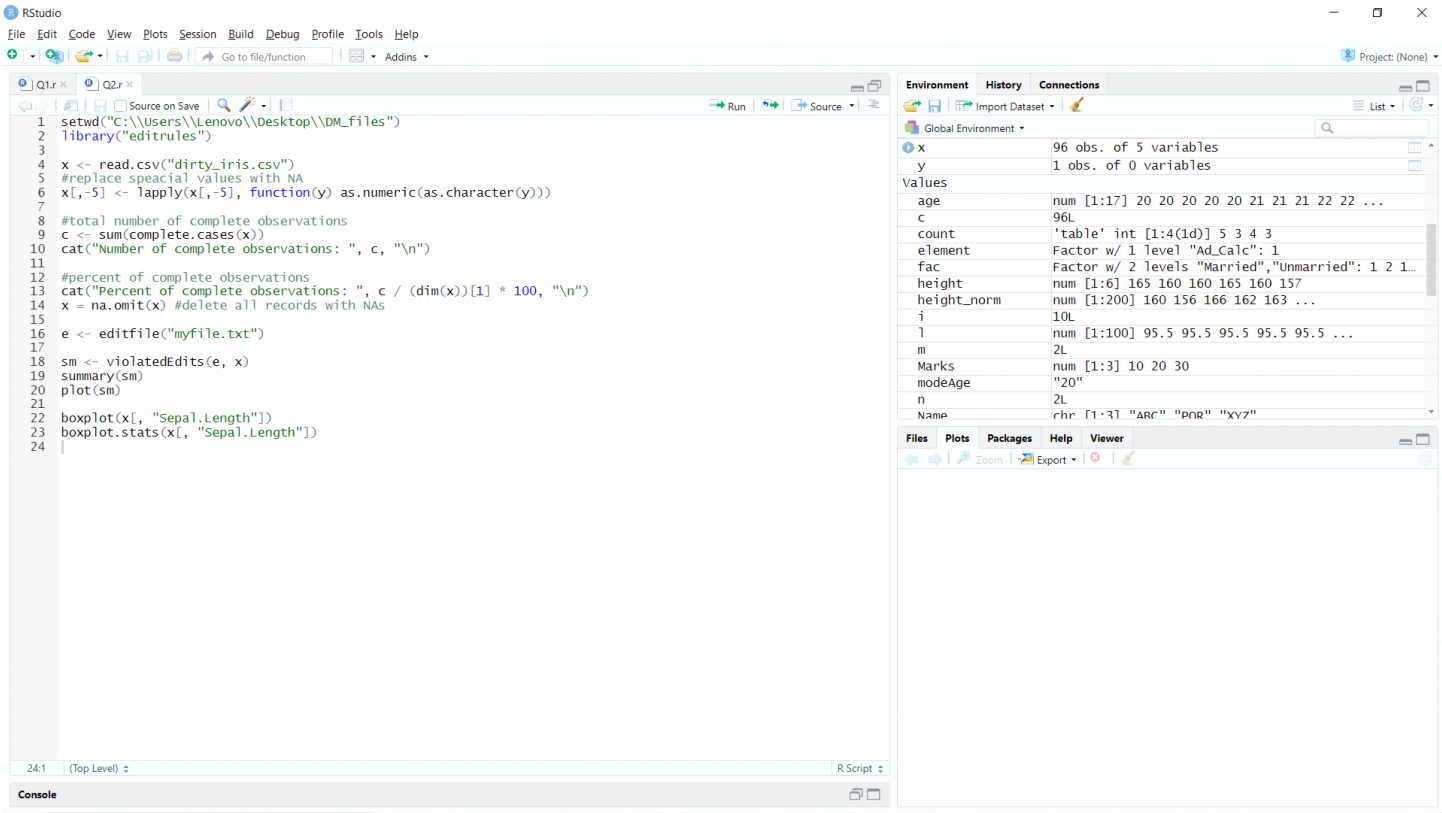
iv) Determine how often each rule is broken (violatedEdits). Also summarize and plot the result.

v) Find outliers in sepal length using boxplot and boxplot.stats

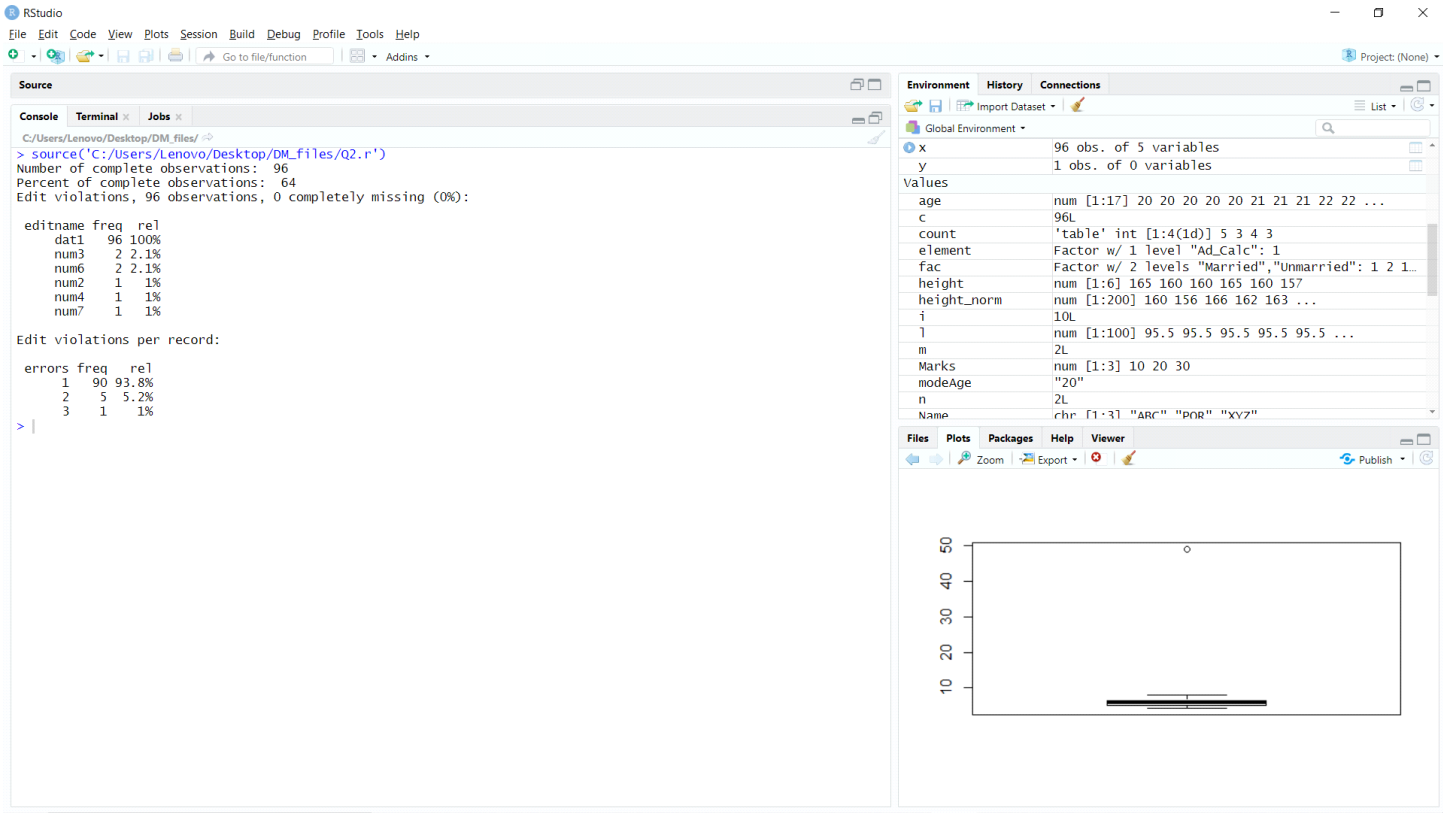
dirty_iris.csv:

myfile.txt:


myfile - Notepad
File Edit Format View Help
Species %in% c("Iris-setosa", "Iris-versicolor", "Iris-virginica")
Sepal.Length > 0
Sepal.Length < 30
Sepal.Width > 0
Petal.Length > 0
Petal.Width > 0
Petal.Length > 2*Petal.Width
Sepal.Length > Petal.Length
Ln 1, Col 1
100%
Windows (CRLF)
UTF-8

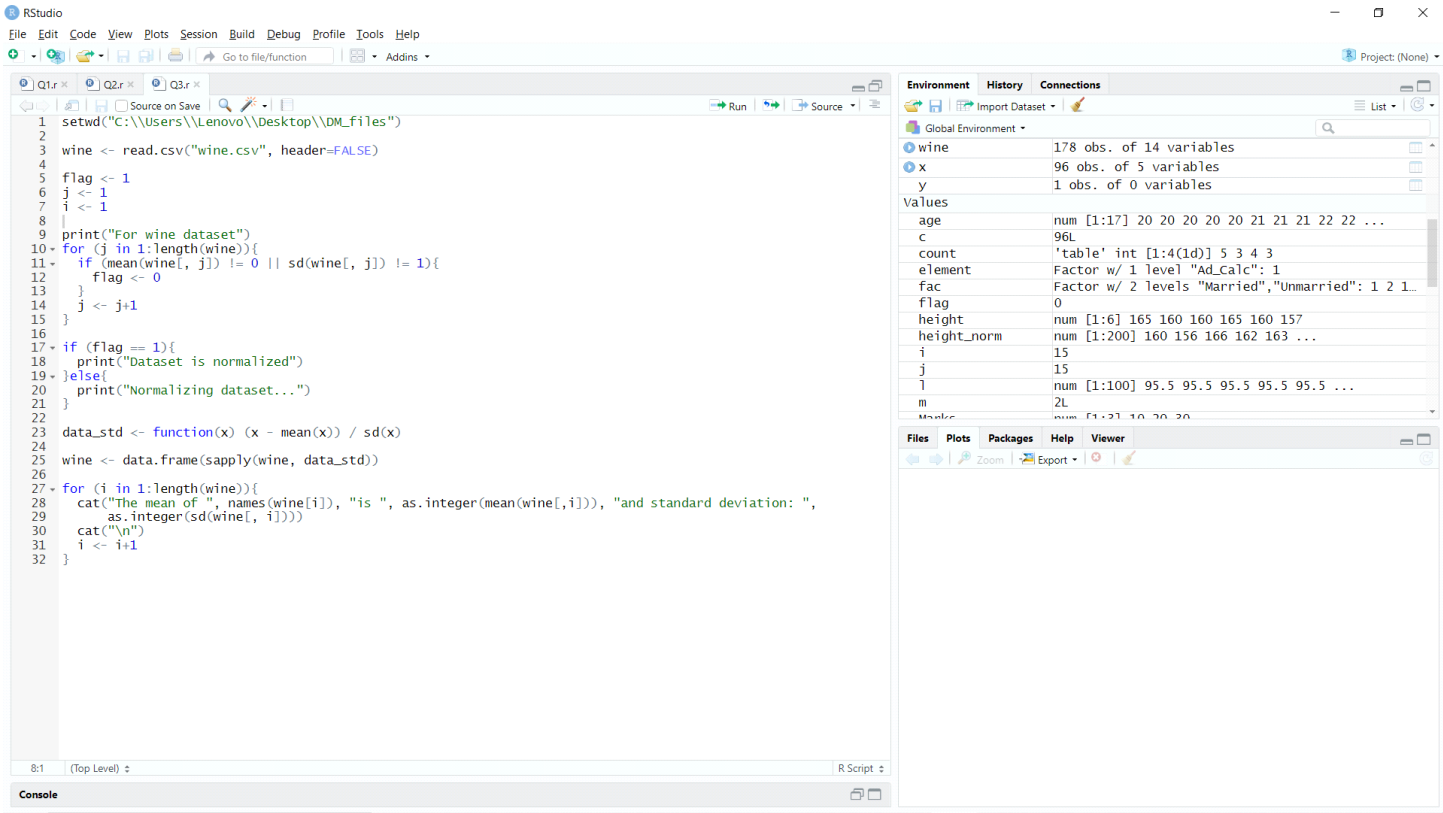

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Q1.r
Q2.r
Source on Save
Run
Source
setwd("C:\\Users\\Lenovo\\Desktop\\DM_files")
library("editrules")
x <- read.csv("dirty_iris.csv")
#replace speacial values with NA
x[,-5] <- lapply(x[,-5], function(y) as.numeric(as.character(y)))
#total number of complete observations
c <- sum(complete.cases(x))
cat("Number of complete observations: ", c, "\n")
#percent of complete observations
cat("Percent of complete observations: ", c / (dim(x))[1] * 100, "\n")
x = na.omit(x) #delete all records with NAs
e <- editfile("myfile.txt")
sm <- violatedEdits(e, x)
summary(sm)
plot(sm)
boxplot(x[, "Sepal.Length"])
boxplot.stats(x[, "Sepal.Length"])
24:1
(Top Level)
R Script
Console
Environment
History
Connections
Import Dataset
List
Global Environment
x 96 obs. of 5 variables
y 1 obs. of 0 variables
Values
age num [1:17] 20 20 20 20 20 21 21 21 22 22 ...
c 96L
count 'table' int [1:4(1d)] 5 3 4 3
element Factor w/ 1 level "Ad_Calc": 1
fac Factor w/ 2 levels "Married","Unmarried": 1 2 1...
height num [1:6] 165 160 160 165 160 157
height_norm num [1:200] 160 156 166 162 163 ...
i 10L
l num [1:100] 95.5 95.5 95.5 95.5 95.5 ...
m 2L
Marks num [1:3] 10 20 30
modeAge "20"
n 2L
Name chr [1:3] "ABC" "POR" "XYZ"
Files
Plots
Packages
Help
Viewer
Zoom
Export

Output:

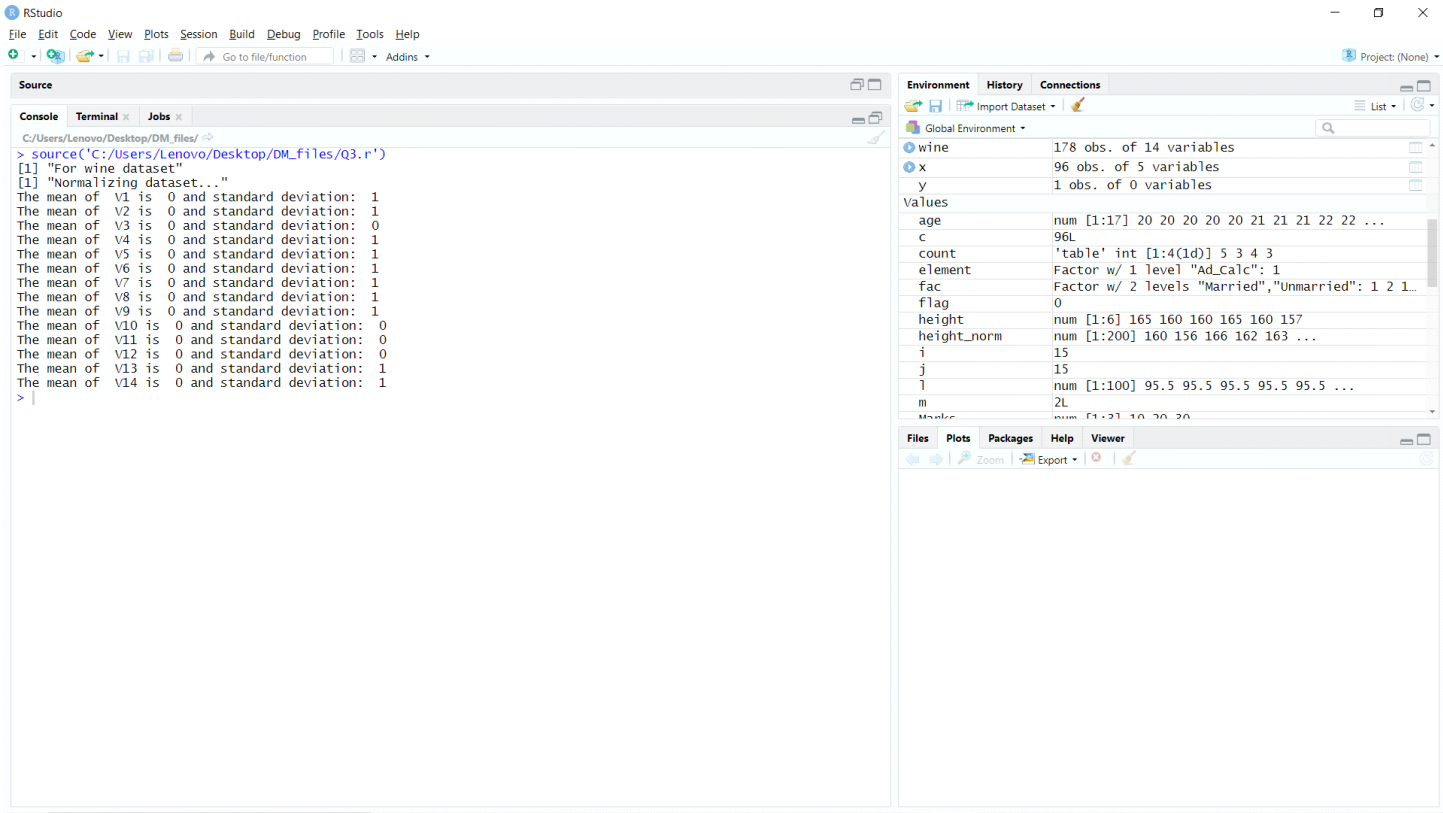


Q3. Load the data from wine dataset. Check whether all attributes are standardized or not (mean is 0 and standard deviation is 1). If not, standardize the attributes. Do the same with Iris dataset.

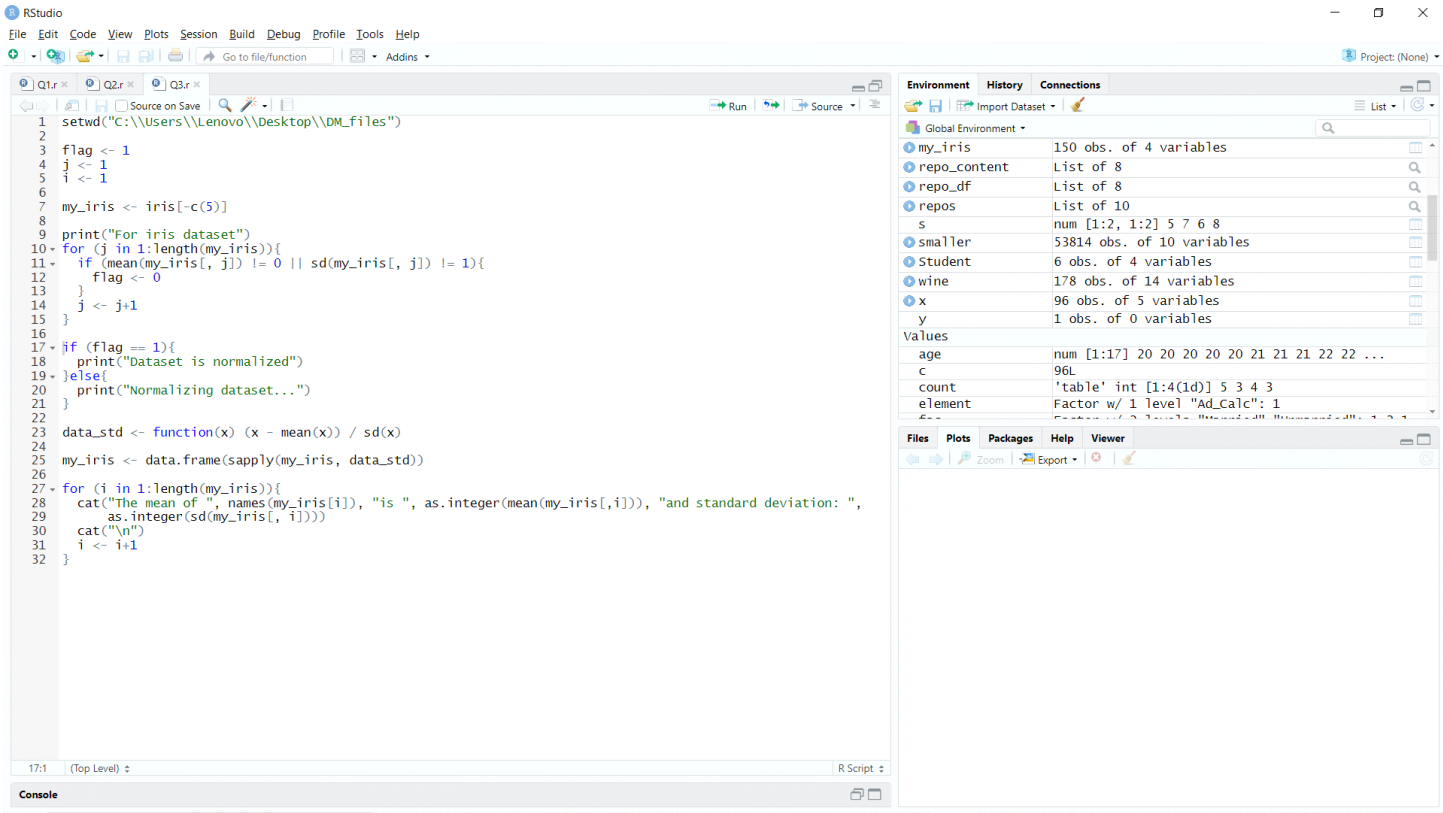
For wine dataset:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r
Source on Save
Run Source
setwd("C:\\Users\\Lenovo\\Desktop\\DM_files")
wine <- read.csv("wine.csv", header=FALSE)
flag <- 1
j <- 1
i <- 1
print("For wine dataset")
for (j in 1:length(wine)){
if (mean(wine[, j]) != 0 || sd(wine[, j]) != 1){
flag <- 0
}
j <- j+1
}
if (flag == 1){
print("Dataset is normalized")
}else{
print("Normalizing dataset...")
}
data_std <- function(x) (x - mean(x)) / sd(x)
wine <- data.frame(sapply(wine, data_std))
for (i in 1:length(wine)){
cat("The mean of ", names(wine[i]), "is ", as.integer(mean(wine[,i])), "and standard deviation: ",
as.integer(sd(wine[, i])))
cat("\n")
i <- i+1
}
8:1 (Top Level) R Script
Console
Environment History Connections
Import Dataset List
Global Environment
wine 178 obs. of 14 variables
x 96 obs. of 5 variables
y 1 obs. of 0 variables
Values
age num [1:17] 20 20 20 20 20 21 21 21 22 22 ...
c 96L
count 'table' int [1:4(1d)] 5 3 4 3
element Factor w/ 1 level "Ad_Calc": 1
fac Factor w/ 2 levels "Married","Unmarried": 1 2 1...
flag 0
height num [1:6] 165 160 160 165 160 157
height_norm num [1:200] 160 156 166 162 163 ...
i 15
j 15
l num [1:100] 95.5 95.5 95.5 95.5 95.5 ...
m 2L
Files Plots Packages Help Viewer
Zoom Export

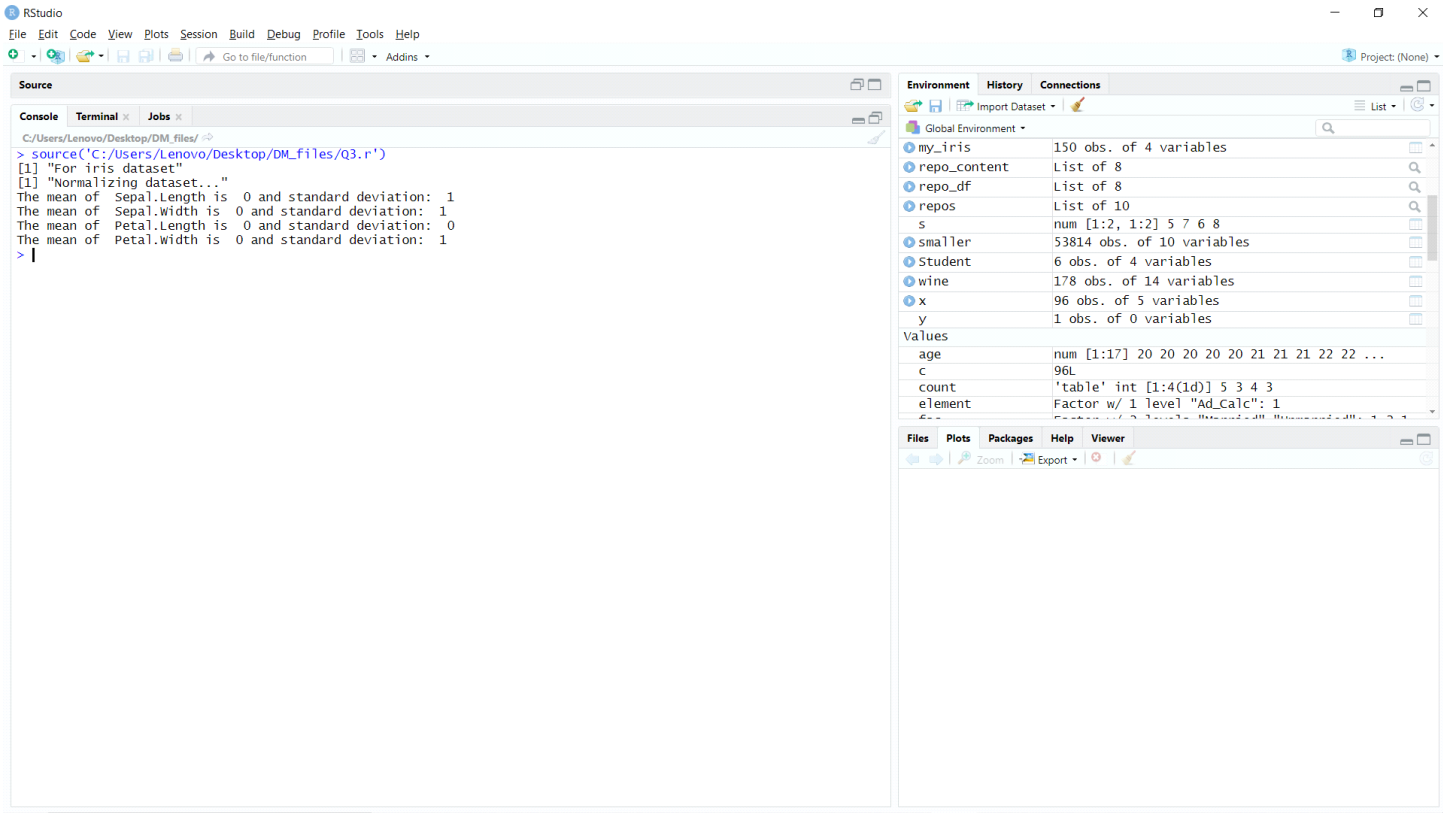
Output:



For iris dataset:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r
Q2.r
Q3.r
Source on Save
Run
Source
setwd("C:\\Users\\Lenovo\\Desktop\\DM_files")
flag <- 1
j <- 1
i <- 1
my_iris <- iris[-c(5)]
print("For iris dataset")
for (j in 1:length(my_iris)){
if (mean(my_iris[, j]) != 0 || sd(my_iris[, j]) != 1){
flag <- 0
}
j <- j+1
}
if (flag == 1){
print("Dataset is normalized")
}else{
print("Normalizing dataset...")
}
data_std <- function(x) (x - mean(x)) / sd(x)
my_iris <- data.frame(sapply(my_iris, data_std))
for (i in 1:length(my_iris)){
cat("The mean of ", names(my_iris[i]), "is ", as.integer(mean(my_iris[,i])), "and standard deviation: ",
as.integer(sd(my_iris[, i])))
cat("\n")
i <- i+1
}
17:1
(Top Level)
R Script
Console
Environment
History
Connections
Import Dataset
List
Global Environment
my_iris 150 obs. of 4 variables
repo_content List of 8
repo_df List of 8
repos List of 10
s num [1:2, 1:2] 5 7 6 8
smaller 53814 obs. of 10 variables
Student 6 obs. of 4 variables
wine 178 obs. of 14 variables
x 96 obs. of 5 variables
y 1 obs. of 0 variables
Values
age num [1:17] 20 20 20 20 20 21 21 21 22 22 ...
c 96L
count 'table' int [1:4(1d)] 5 3 4 3
element Factor w/ 1 level "Ad_Calc": 1
Files
Plots
Packages
Help
Viewer
Zoom
Export

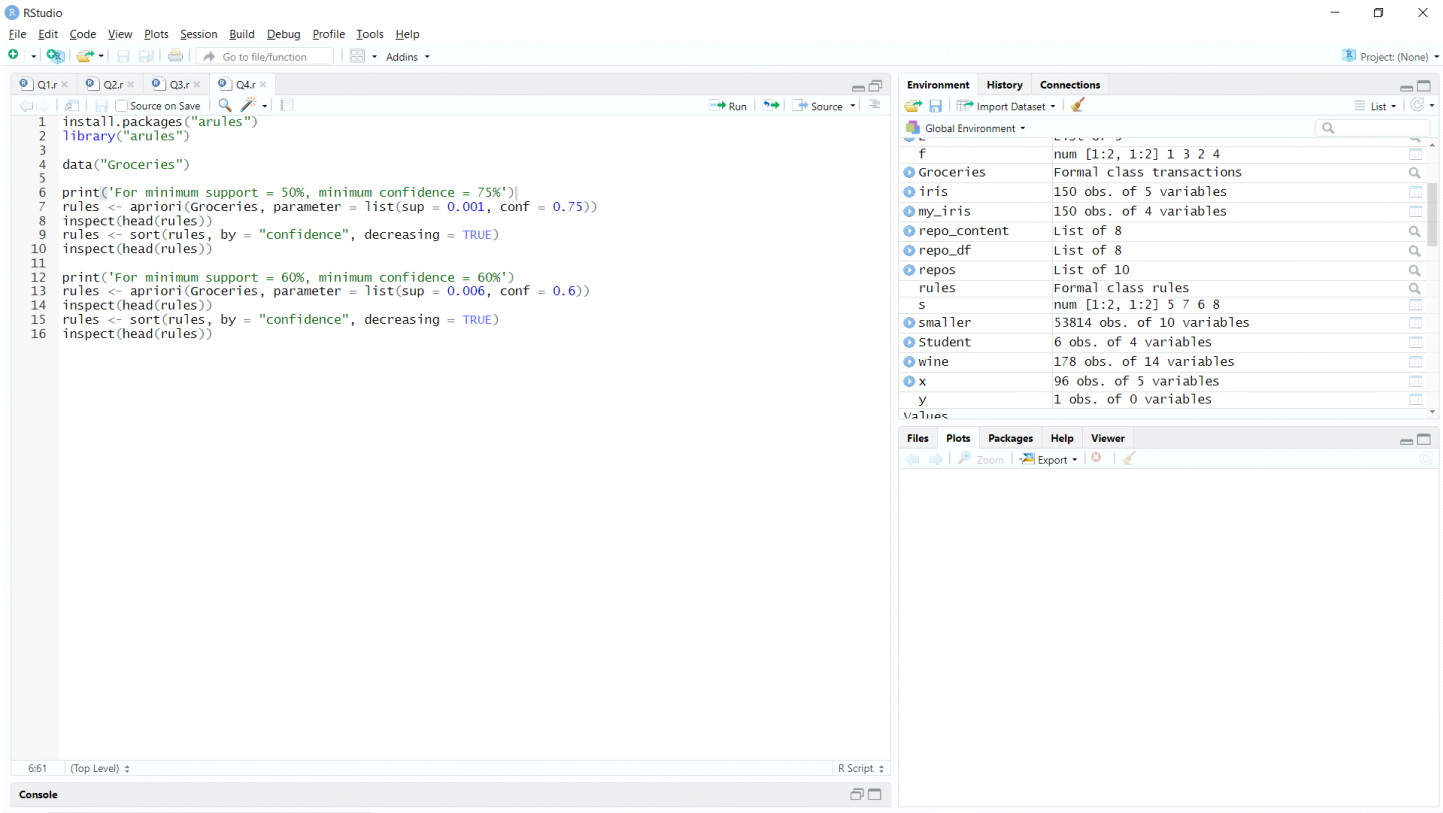
Output:



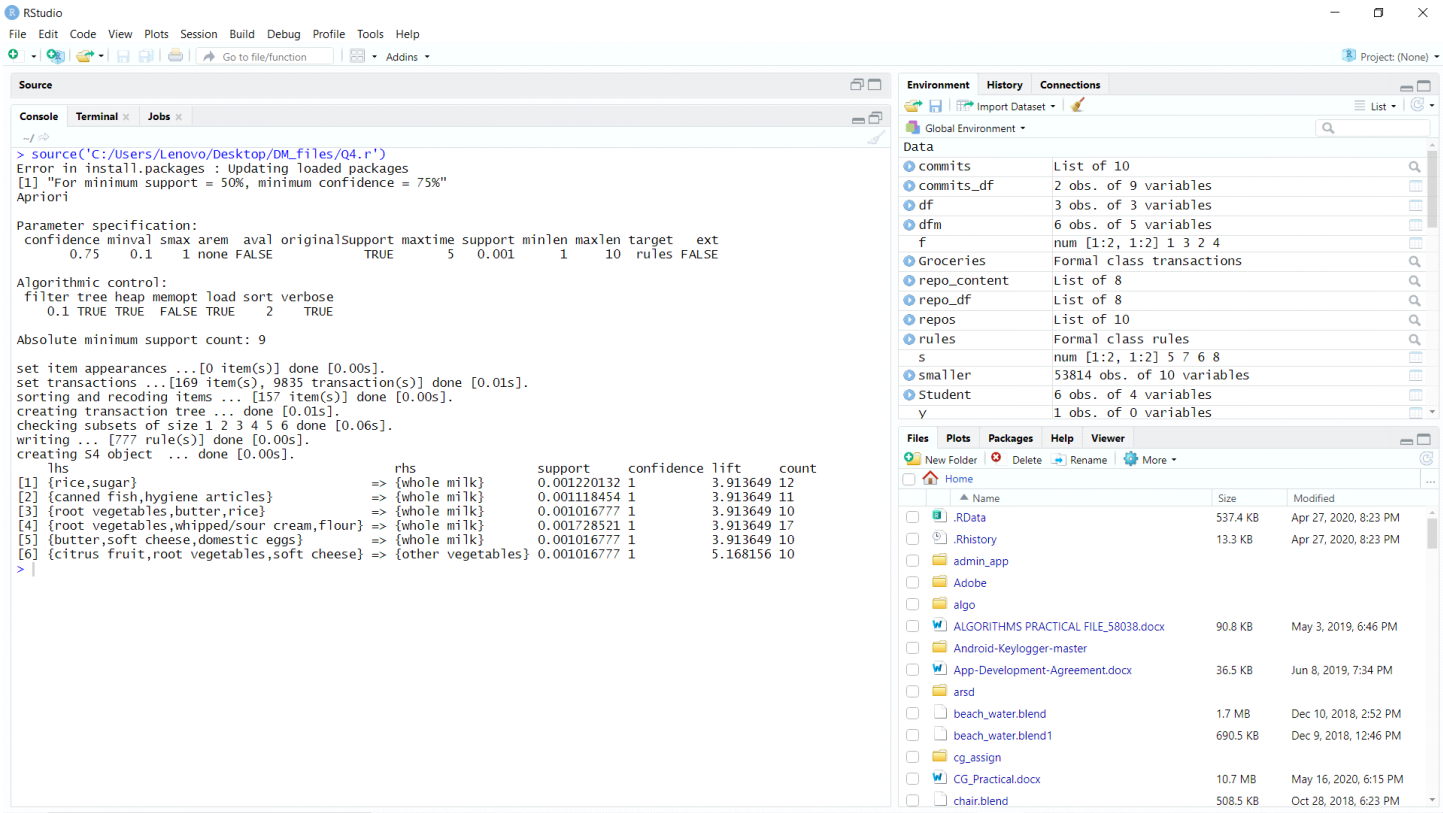
Q4. Run Apriori algorithm to find frequent itemsets and association rules

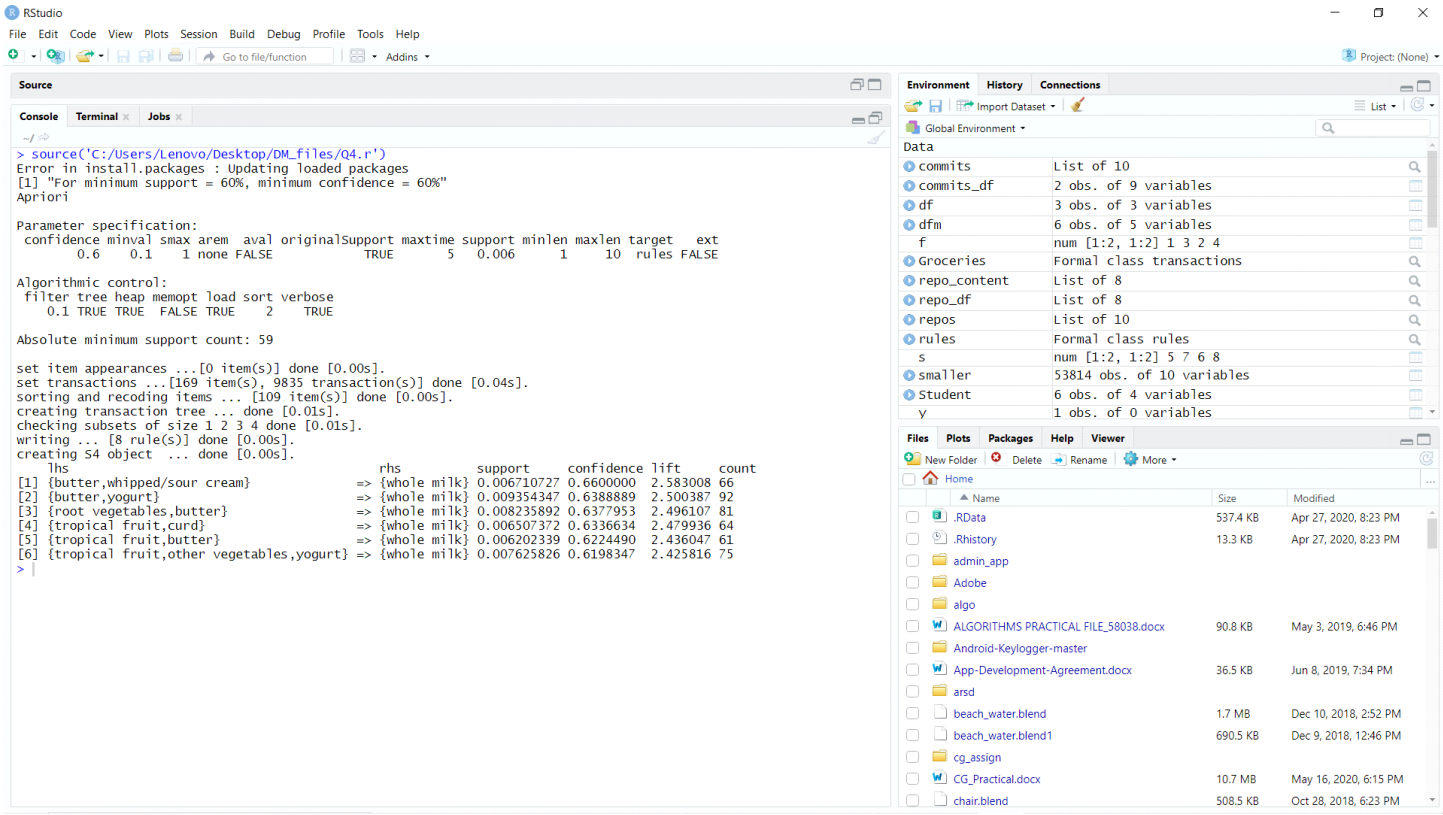
4.1 Use minimum support as 50% and minimum confidence as 75%

4.2 Use minimum support as 60% and minimum confidence as 60%


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r Q4.r
Source on Save
Run Source
1 install.packages("arules")
2 library("arules")
3
4 data("Groceries")
5
6 print('For minimum support = 50%, minimum confidence = 75%')
7 rules <- apriori(Groceries, parameter = list(sup = 0.001, conf = 0.75))
8 inspect(head(rules))
9 rules <- sort(rules, by = "confidence", decreasing = TRUE)
10 inspect(head(rules))
11
12 print('For minimum support = 60%, minimum confidence = 60%')
13 rules <- apriori(Groceries, parameter = list(sup = 0.006, conf = 0.6))
14 inspect(head(rules))
15 rules <- sort(rules, by = "confidence", decreasing = TRUE)
16 inspect(head(rules))
6:61 (Top Level) R Script
Console
Environment History Connections
Import Dataset List
Global Environment
f num [1:2, 1:2] 1 3 2 4
Groceries Formal class transactions
iris 150 obs. of 5 variables
my_iris 150 obs. of 4 variables
repo_content List of 8
repo_df List of 8
repos List of 10
rules Formal class rules
s num [1:2, 1:2] 5 7 6 8
smaller 53814 obs. of 10 variables
Student 6 obs. of 4 variables
wine 178 obs. of 14 variables
x 96 obs. of 5 variables
y 1 obs. of 0 variables
Values
Files Plots Packages Help Viewer
Zoom Export

Output:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source
Console Terminal Jobs
> source('C:/Users/Lenovo/Desktop/DM_files/Q4.r')
Error in install.packages : Updating loaded packages
[1] "For minimum support = 50%, minimum confidence = 75%"
Apriori
Parameter specification:
confidence minval smax arem aval originalSupport maxtime support minlen maxlen target ext
0.75 0.1 1 none FALSE TRUE 5 0.001 1 10 rules FALSE
Algorithmic control:
filter tree heap memopt load sort verbose
0.1 TRUE TRUE FALSE TRUE 2 TRUE
Absolute minimum support count: 9
set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[169 item(s), 9835 transaction(s)] done [0.01s].
sorting and recoding items ... [157 item(s)] done [0.00s].
creating transaction tree ... done [0.01s].
checking subsets of size 1 2 3 4 5 6 done [0.06s].
writing ... [777 rule(s)] done [0.00s].
creating S4 object ... done [0.00s].
lhs rhs support confidence lift count
[1] {rice,sugar} => {whole milk} 0.001220132 1 3.913649 12
[2] {canned fish,hygiene articles} => {whole milk} 0.001118454 1 3.913649 11
[3] {root vegetables,butter,rice} => {whole milk} 0.001016777 1 3.913649 10
[4] {root vegetables,whipped/sour cream,flour} => {whole milk} 0.001728521 1 3.913649 17
[5] {butter,soft cheese,domestic eggs} => {whole milk} 0.001016777 1 3.913649 10
[6] {citrus fruit,root vegetables,soft cheese} => {other vegetables} 0.001016777 1 5.168156 10
Environment History Connections
Import Dataset
Global Environment
Data
commits List of 10
commits_df 2 obs. of 9 variables
df 3 obs. of 3 variables
dfm 6 obs. of 5 variables
f num [1:2, 1:2] 1 3 2 4
Groceries Formal class transactions
repo_content List of 8
repo_df List of 8
repos List of 10
rules Formal class rules
s num [1:2, 1:2] 5 7 6 8
smaller 53814 obs. of 10 variables
Student 6 obs. of 4 variables
y 1 obs. of 0 variables
Files Plots Packages Help Viewer
New Folder Delete Rename More
Home
Name Size Modified
.RData 537.4 KB Apr 27, 2020, 8:23 PM
.Rhistory 13.3 KB Apr 27, 2020, 8:23 PM
admin_app
Adobe
algo
ALGORITHMS PRACTICAL FILE_58038.docx 90.8 KB May 3, 2019, 6:46 PM
Android-Keylogger-master
App-Development-Agreement.docx 36.5 KB Jun 8, 2019, 7:34 PM
arsd
beach_water.blend 1.7 MB Dec 10, 2018, 2:52 PM
beach_water.blend1 690.5 KB Dec 9, 2018, 12:46 PM
cg_assign
CG_Practical.docx 10.7 MB May 16, 2020, 6:15 PM
chair.blend 508.5 KB Oct 28, 2018, 6:23 PM



Q5. Use Naive bayes, K-nearest, and Decision tree classification algorithms and build classifiers. Divide the data set in to training and test set. Compare the accuracy of the different classifiers under the following situations:

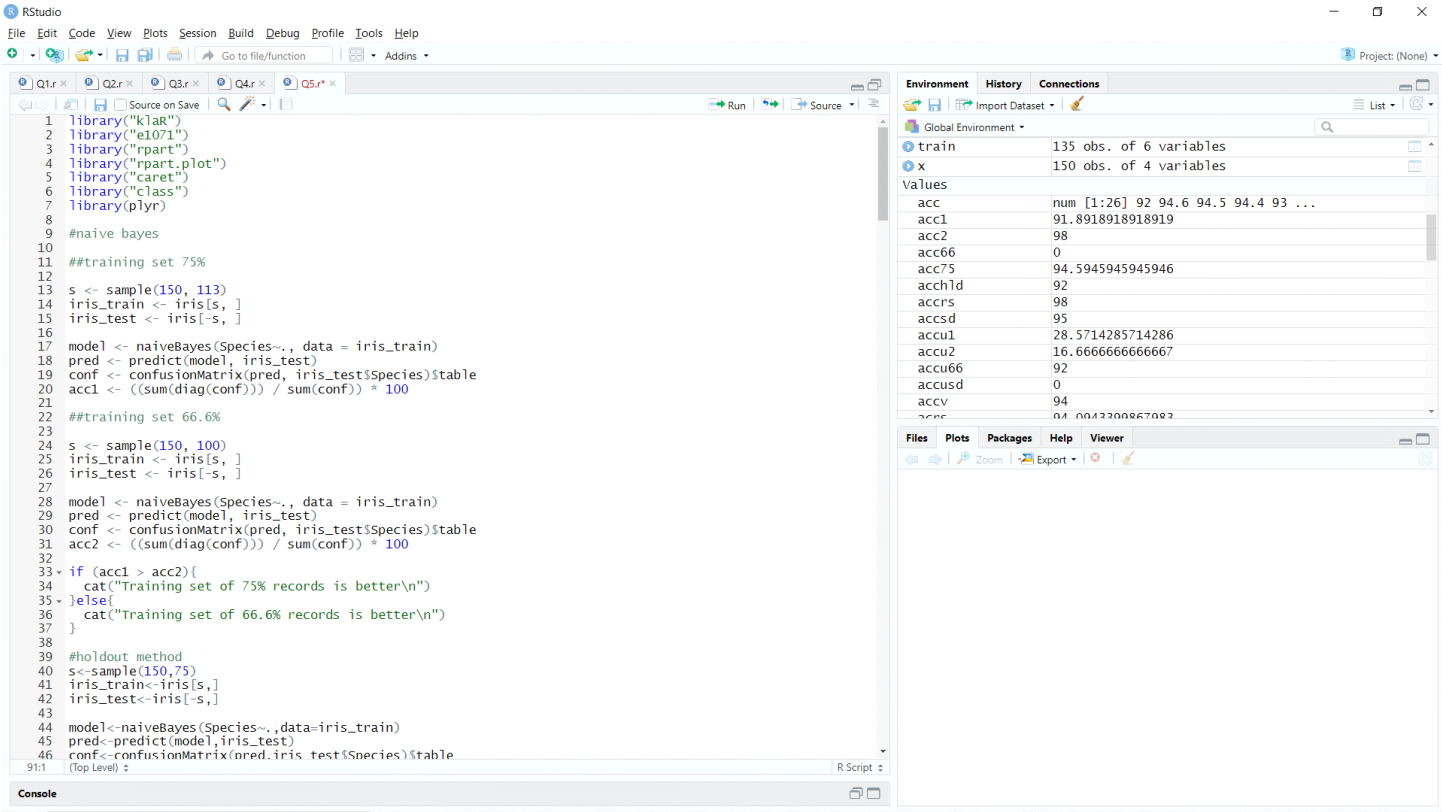
5.1 a) Training set = 75% Test set = 25%

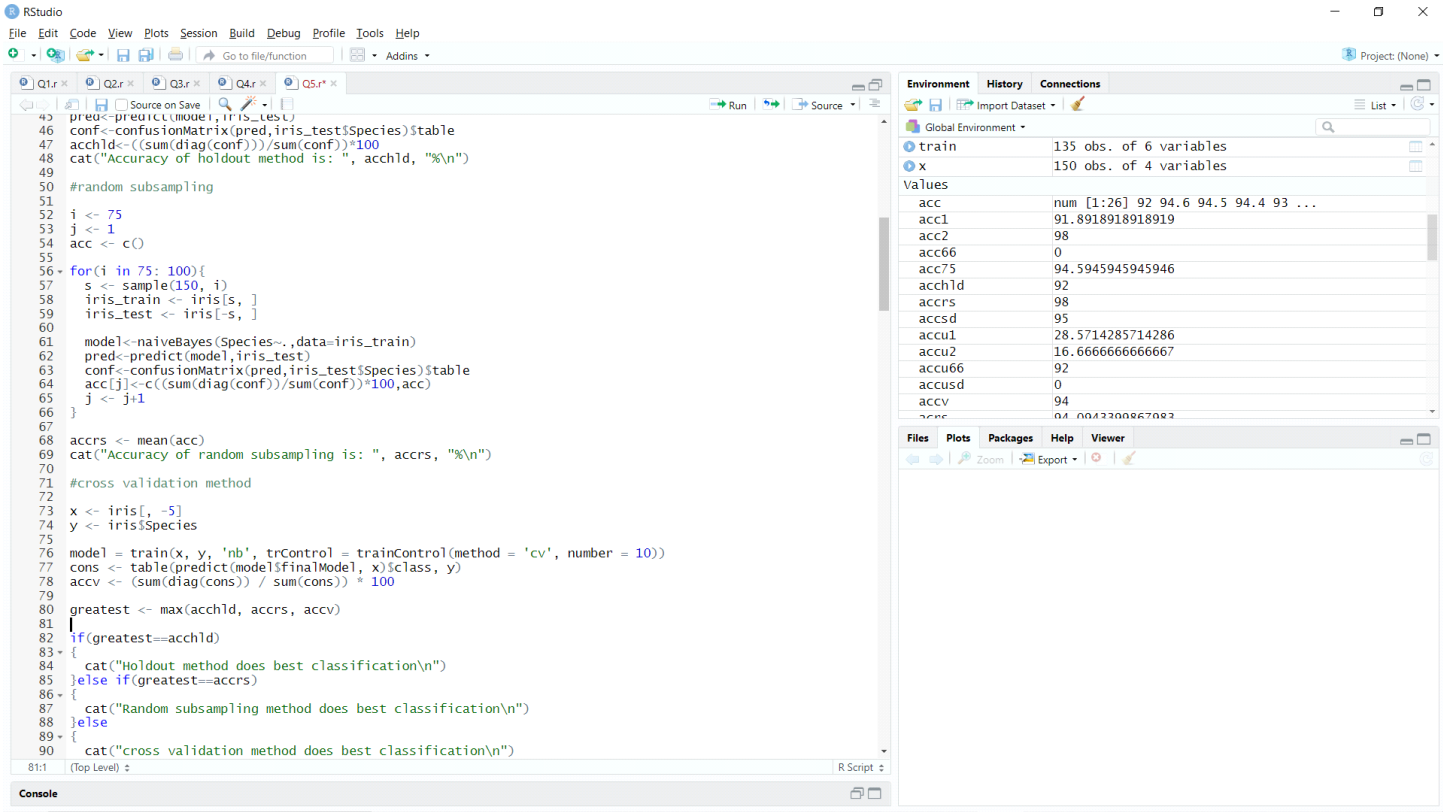
5.1 b) Training set = 66.6% (2/3rd of total), Test set = 33.3%

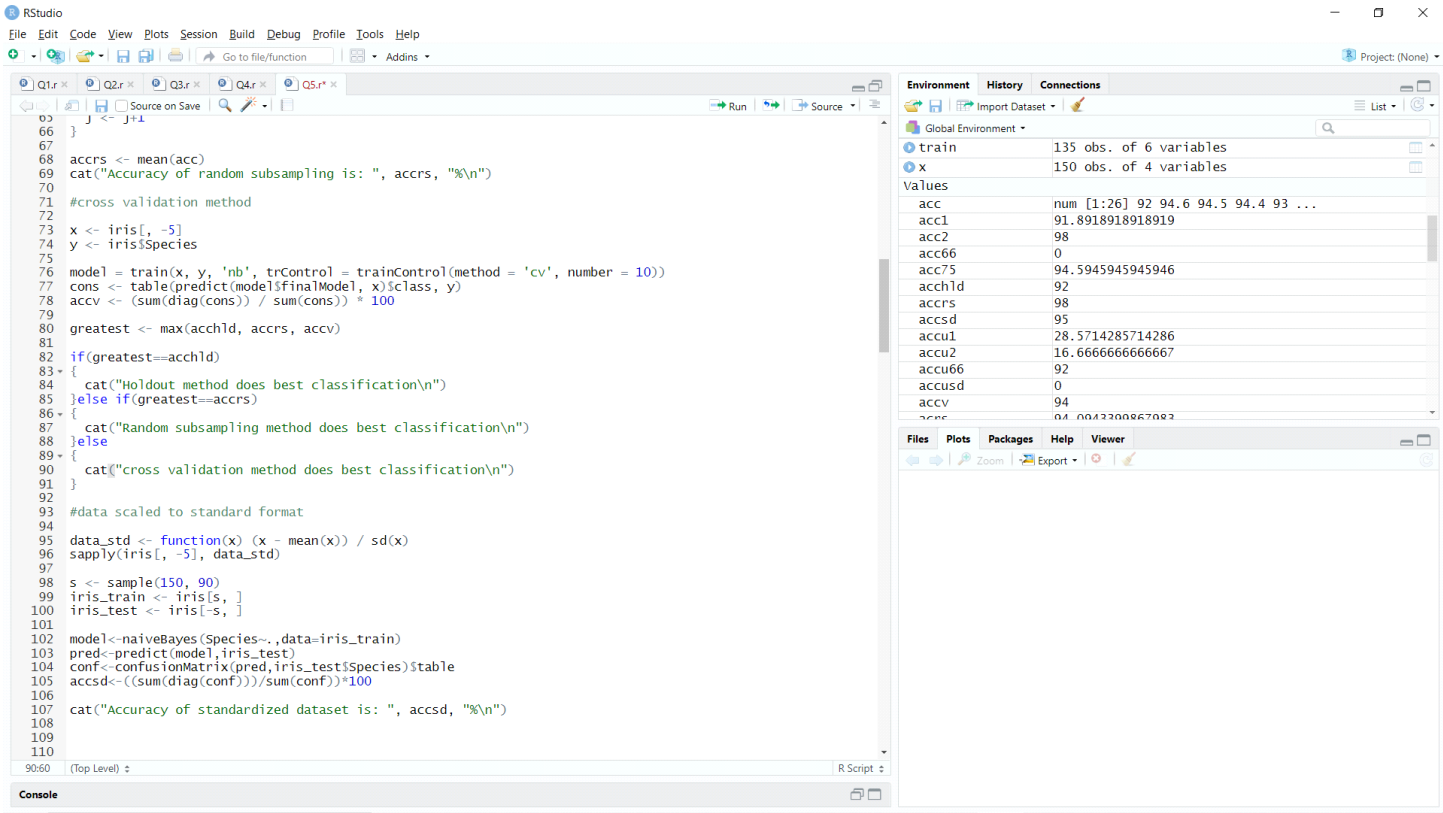
5.2 Training set is chosen by i)hold out method ii)Random subsampling iii)Cross-Validation. Compare the accuracy of the classifiers obtained.

5.3 Data is scaled to standard format.

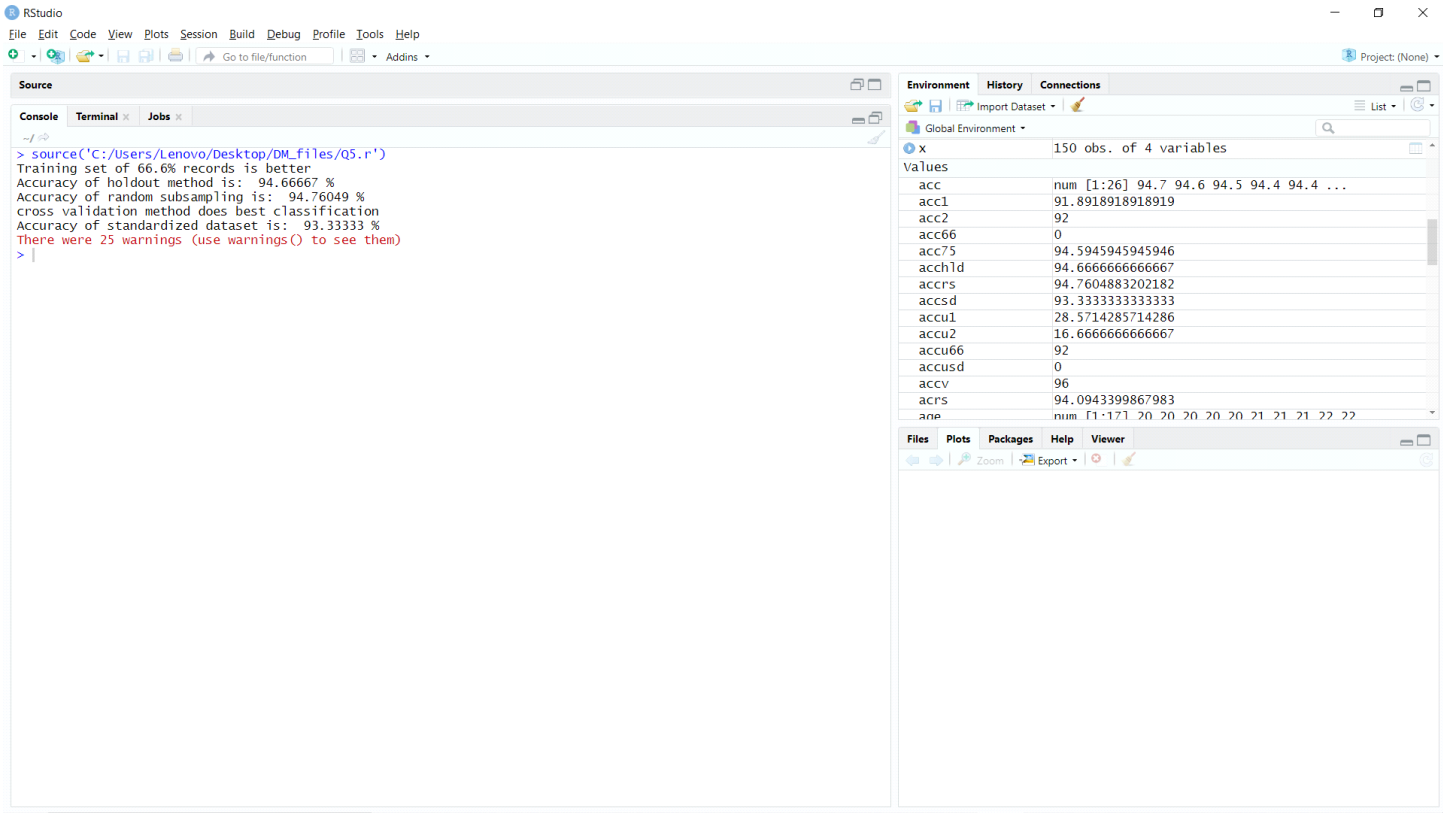
Naive Bayes algorithm:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r Q4.r Q5.r*
Source on Save
Run
Source
library("klaR")
library("e1071")
library("rpart")
library("rpart.plot")
library("caret")
library("class")
library(plyr)
#naive bayes
##training set 75%
s <- sample(150, 113)
iris_train <- iris[s,]
iris_test <- iris[-s,]
model <- naiveBayes(Species~., data = iris_train)
pred <- predict(model, iris_test)
conf <- confusionMatrix(pred, iris_test$Species)$table
acc1 <- ((sum(diag(conf))) / sum(conf)) * 100
##training set 66.6%
s <- sample(150, 100)
iris_train <- iris[s,]
iris_test <- iris[-s,]
model <- naiveBayes(Species~., data = iris_train)
pred <- predict(model, iris_test)
conf <- confusionMatrix(pred, iris_test$Species)$table
acc2 <- ((sum(diag(conf))) / sum(conf)) * 100
if (acc1 > acc2){
cat("Training set of 75% records is better\n")
}else{
cat("Training set of 66.6% records is better\n")
}
#holdout method
s<-sample(150,75)
iris_train<-iris[s,]
iris_test<-iris[-s,]
model<-naiveBayes(Species~.,data=iris_train)
pred<-predict(model,iris_test)
conf<-confusionMatrix(pred,iris_test$Species)$table
91:1
(Top Level)
R Script
Console
Environment History Connections
Import Dataset
List
Global Environment
train 135 obs. of 6 variables
x 150 obs. of 4 variables
Values
acc num [1:26] 92 94.6 94.5 94.4 93 ...
acc1 91.8918918918919
acc2 98
acc66 0
acc75 94.5945945945946
acchld 92
accrs 98
accsd 95
accu1 28.5714285714286
accu2 16.6666666666667
accu66 92
accusd 0
accv 94
Files Plots Packages Help Viewer
Zoom
Export

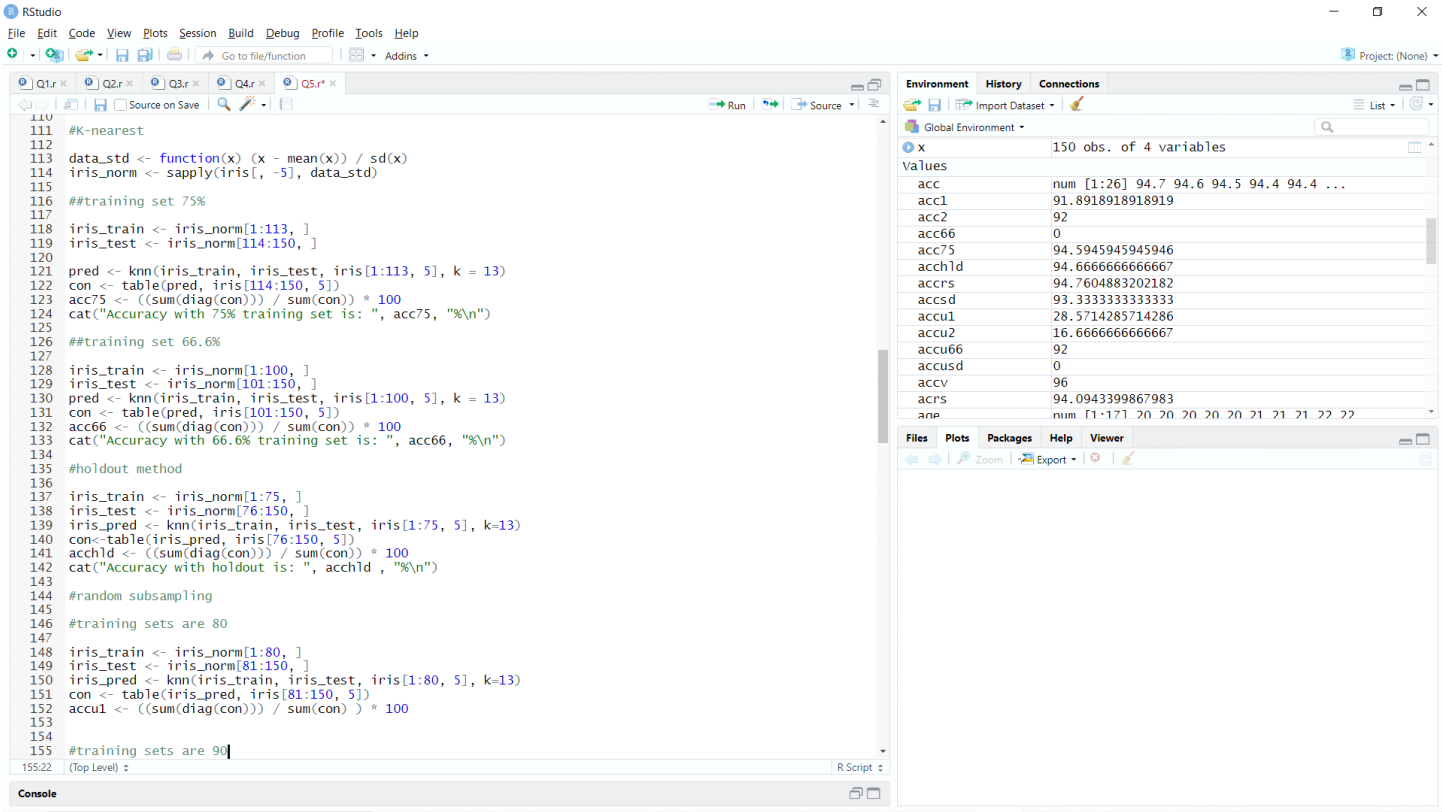

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r Q4.r Q5.r*
Source on Save
Run
Source
pred<-predict(model,iris_test)
conf<-confusionMatrix(pred,iris_test$Species)$table
acchld<-((sum(diag(conf)))/sum(conf))*100
cat("Accuracy of holdout method is: ", acchld, "%\n")
#random subsampling
i <- 75
j <- 1
acc <- c()
for(i in 75: 100){
s <- sample(150, i)
iris_train <- iris[s,]
iris_test <- iris[-s,]
model<-naiveBayes(Species~.,data=iris_train)
pred<-predict(model,iris_test)
conf<-confusionMatrix(pred,iris_test$Species)$table
acc[j]<-c((sum(diag(conf))/sum(conf))*100,acc)
j <- j+1
}
accrs <- mean(acc)
cat("Accuracy of random subsampling is: ", accrs, "%\n")
#cross validation method
x <- iris[, -5]
y <- iris$Species
model = train(x, y, 'nb', trControl = trainControl(method = 'cv', number = 10))
cons <- table(predict(model$finalModel, x)$class, y)
accv <- (sum(diag(cons)) / sum(cons)) * 100
greatest <- max(acchld, accrs, accv)
if(greatest==acchld)
{
cat("Holdout method does best classification\n")
}else if(greatest==accrs)
{
cat("Random subsampling method does best classification\n")
}else
{
cat("cross validation method does best classification\n")
81:1 (Top Level) R Script
Console
Environment History Connections
Import Dataset List
Global Environment
train 135 obs. of 6 variables
x 150 obs. of 4 variables
Values
acc num [1:26] 92 94.6 94.5 94.4 93 ...
acc1 91.8918918918919
acc2 98
acc66 0
acc75 94.5945945945946
acchld 92
accrs 98
accsd 95
accu1 28.5714285714286
accu2 16.6666666666667
accu66 92
accusd 0
accv 94
Files Plots Packages Help Viewer
Zoom Export

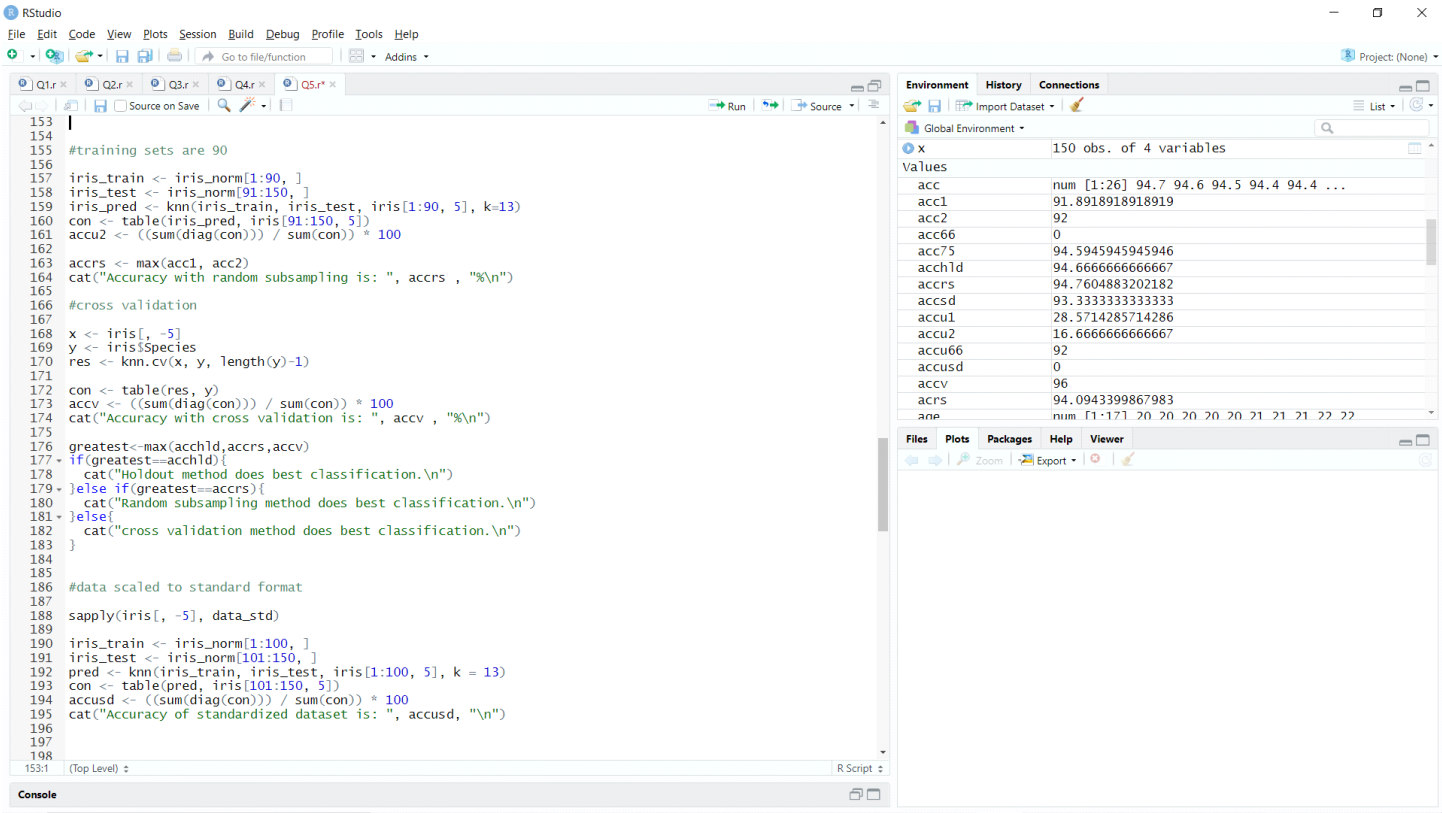

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r Q4.r Q5.r*
Source on Save
Run Source
j <- j+1
}
accrs <- mean(acc)
cat("Accuracy of random subsampling is: ", accrs, "%\n")
#cross validation method
x <- iris[, -5]
y <- iris$Species
model = train(x, y, 'nb', trControl = trainControl(method = 'cv', number = 10))
cons <- table(predict(model$finalModel, x)$class, y)
accv <- (sum(diag(cons)) / sum(cons)) * 100
greatest <- max(acchld, accrs, accv)
if(greatest==acchld)
{
cat("Holdout method does best classification\n")
}else if(greatest==accrs)
{
cat("Random subsampling method does best classification\n")
}else
{
cat("cross validation method does best classification\n")
}
#data scaled to standard format
data_std <- function(x) (x - mean(x)) / sd(x)
sapply(iris[, -5], data_std)
s <- sample(150, 90)
iris_train <- iris[s,]
iris_test <- iris[-s,]
model<-naiveBayes(Species~.,data=iris_train)
pred<-predict(model,iris_test)
conf<-confusionMatrix(pred,iris_test$Species)$table
accsd<-((sum(diag(conf)))/sum(conf))*100
cat("Accuracy of standardized dataset is: ", accsd, "%\n")
90:60 (Top Level)
R Script
Console
Environment History Connections
Import Dataset
List
Global Environment
train 135 obs. of 6 variables
x 150 obs. of 4 variables
Values
acc num [1:26] 92 94.6 94.5 94.4 93 ...
acc1 91.8918918918919
acc2 98
acc66 0
acc75 94.5945945945946
acchld 92
accrs 98
accsd 95
accu1 28.5714285714286
accu2 16.6666666666667
accu66 92
accusd 0
accv 94
Files Plots Packages Help Viewer
Zoom Export

Output:

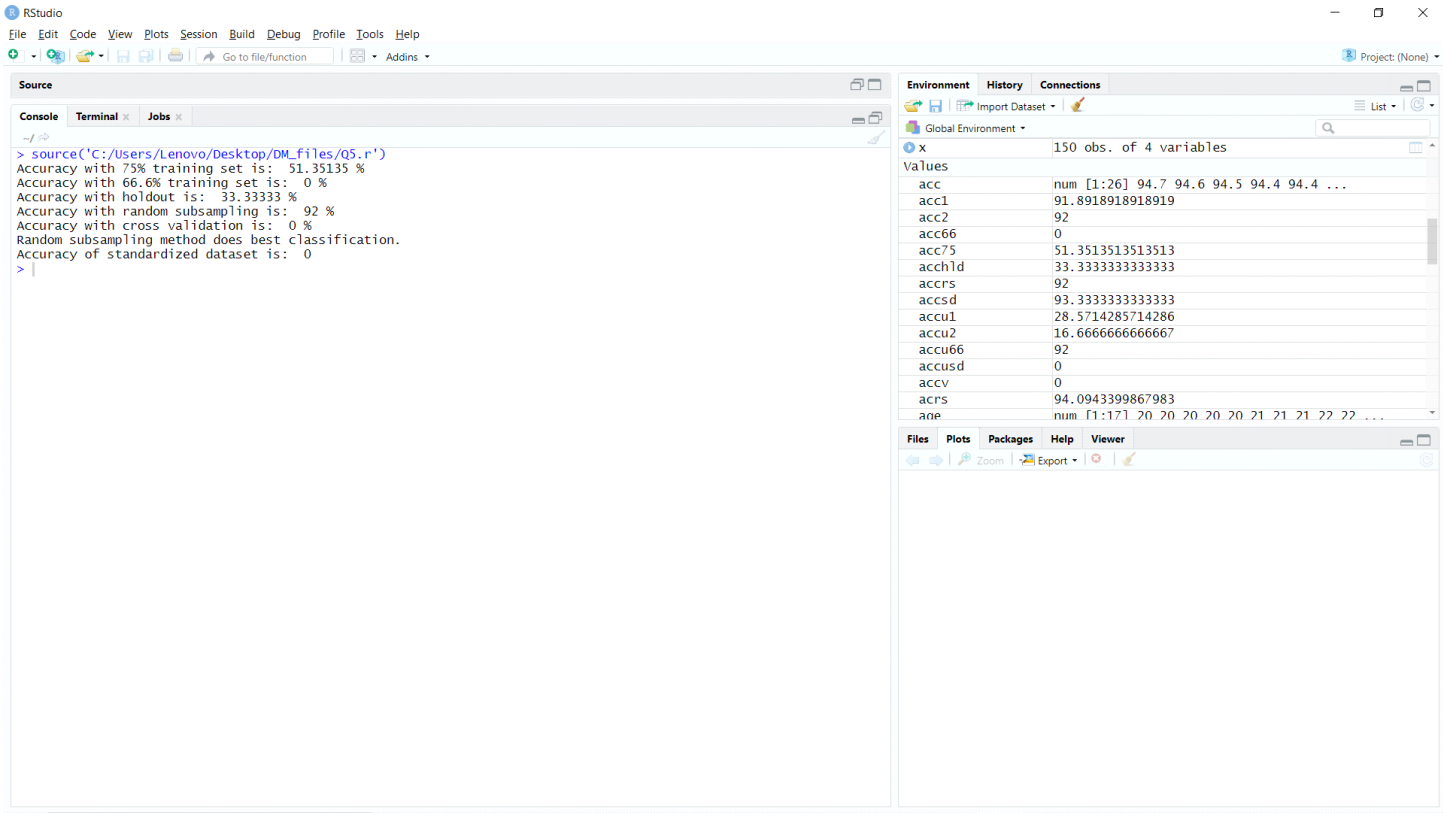


K-nearest algorithm:

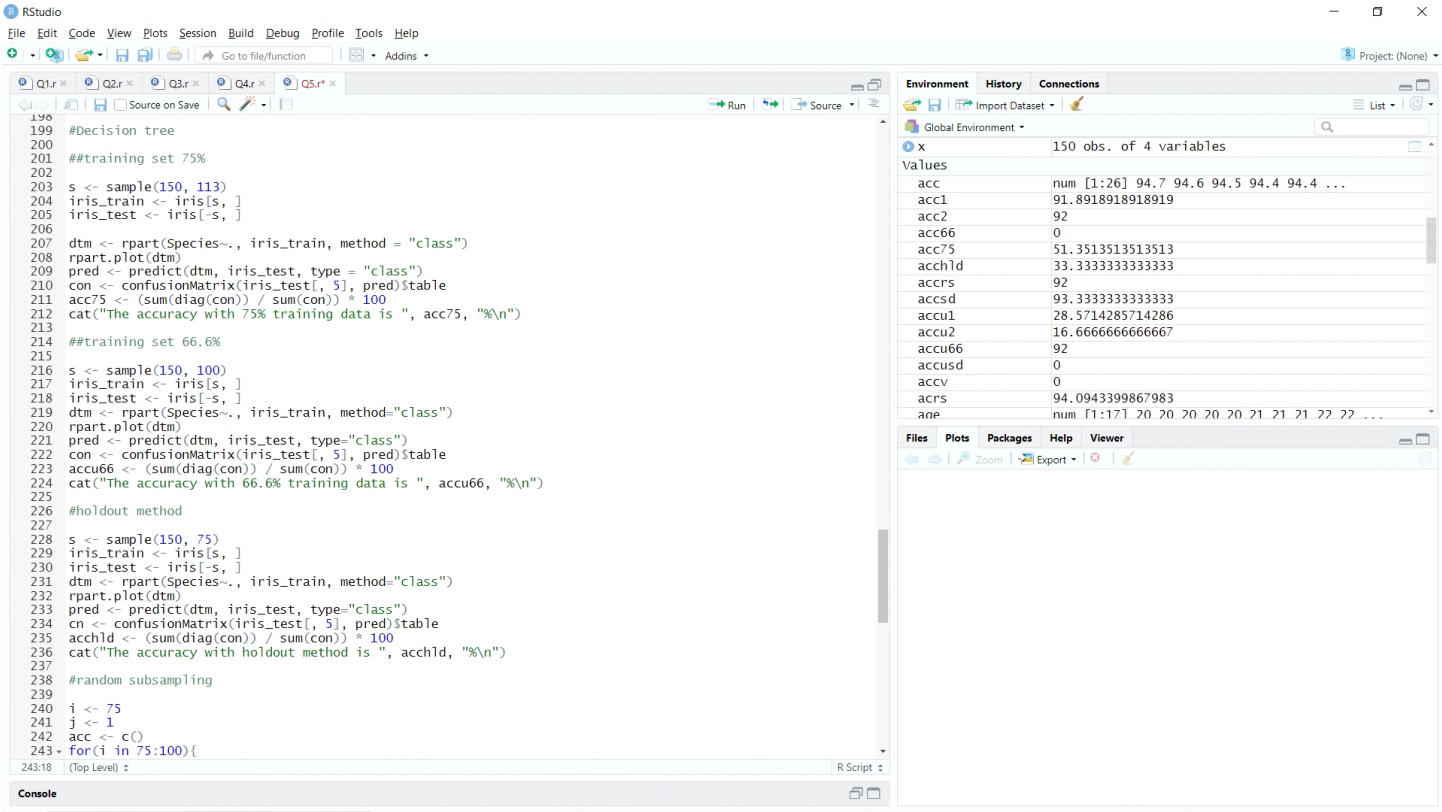

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r Q4.r Q5.r*
Source on Save
Run
Source
#K-nearest
data_std <- function(x) (x - mean(x)) / sd(x)
iris_norm <- sapply(iris[, -5], data_std)
##training set 75%
iris_train <- iris_norm[1:113,]
iris_test <- iris_norm[114:150,]
pred <- knn(iris_train, iris_test, iris[1:113, 5], k = 13)
con <- table(pred, iris[114:150, 5])
acc75 <- ((sum(diag(con))) / sum(con)) * 100
cat("Accuracy with 75% training set is: ", acc75, "%\n")
##training set 66.6%
iris_train <- iris_norm[1:100,]
iris_test <- iris_norm[101:150,]
pred <- knn(iris_train, iris_test, iris[1:100, 5], k = 13)
con <- table(pred, iris[101:150, 5])
acc66 <- ((sum(diag(con))) / sum(con)) * 100
cat("Accuracy with 66.6% training set is: ", acc66, "%\n")
#holdout method
iris_train <- iris_norm[1:75,]
iris_test <- iris_norm[76:150,]
iris_pred <- knn(iris_train, iris_test, iris[1:75, 5], k=13)
con<-table(iris_pred, iris[76:150, 5])
acchld <- ((sum(diag(con))) / sum(con)) * 100
cat("Accuracy with holdout is: ", acchld , "%\n")
#random subsampling
#training sets are 80
iris_train <- iris_norm[1:80,]
iris_test <- iris_norm[81:150,]
iris_pred <- knn(iris_train, iris_test, iris[1:80, 5], k=13)
con <- table(iris_pred, iris[81:150, 5])
accu1 <- ((sum(diag(con))) / sum(con)) * 100
#training sets are 90
155:22 (Top Level) R Script
Console
Environment History Connections
Import Dataset List
Global Environment
x 150 obs. of 4 variables
Values
acc num [1:26] 94.7 94.6 94.5 94.4 94.4 ...
acc1 91.8918918918919
acc2 92
acc66 0
acc75 94.5945945945946
acchld 94.6666666666667
accrs 94.7604883202182
accsd 93.3333333333333
accu1 28.5714285714286
accu2 16.6666666666667
accu66 92
accusd 0
accv 96
acrs 94.0943399867983
Files Plots Packages Help Viewer
Zoom Export

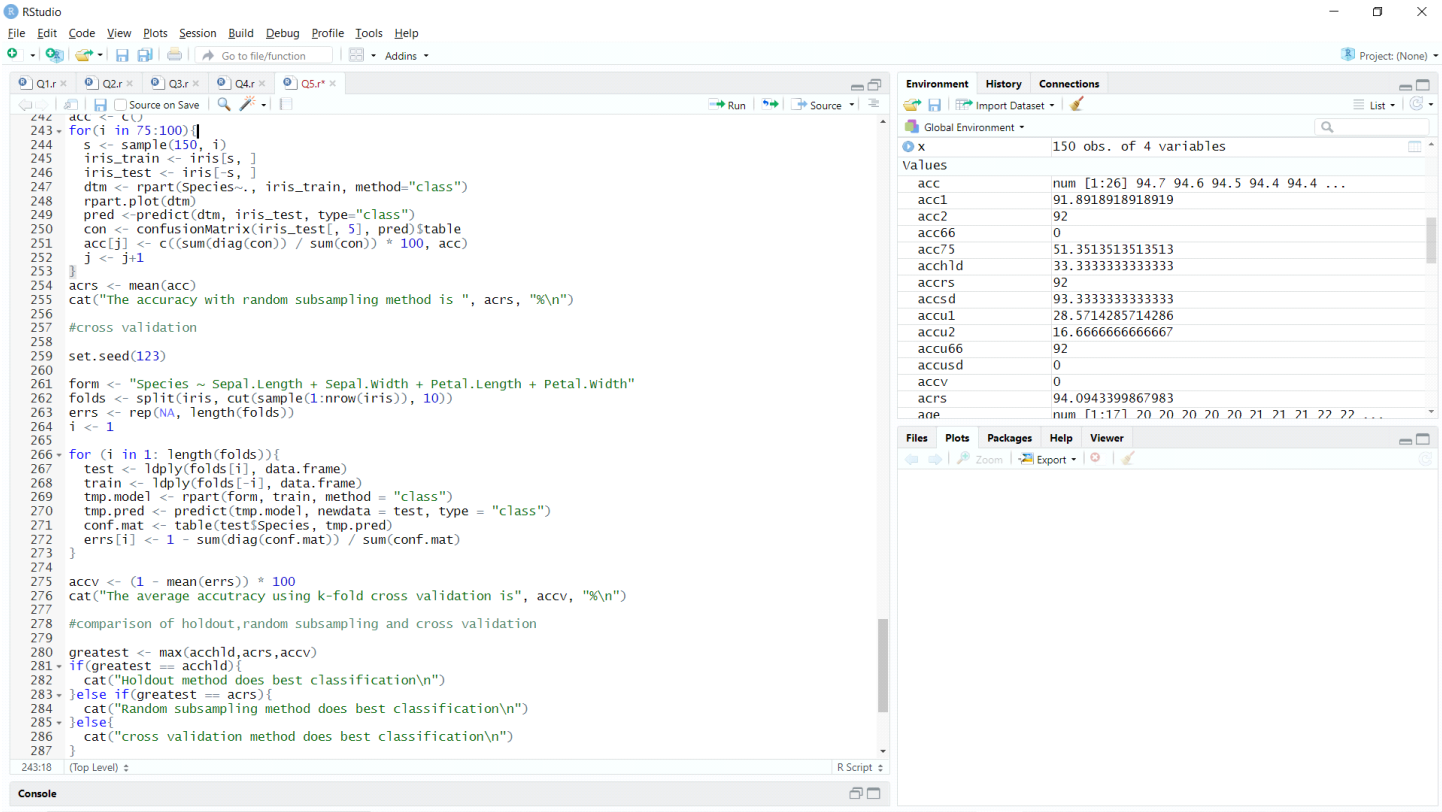

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r Q4.r Q5.r*
Source on Save
Run
Source
#training sets are 90
iris_train <- iris_norm[1:90,]
iris_test <- iris_norm[91:150,]
iris_pred <- knn(iris_train, iris_test, iris[1:90, 5], k=13)
con <- table(iris_pred, iris[91:150, 5])
accu2 <- ((sum(diag(con))) / sum(con)) * 100
accrs <- max(acc1, acc2)
cat("Accuracy with random subsampling is: ", accrs , "%\n")
#cross validation
x <- iris[, -5]
y <- iris$Species
res <- knn.cv(x, y, length(y)-1)
con <- table(res, y)
accv <- ((sum(diag(con))) / sum(con)) * 100
cat("Accuracy with cross validation is: ", accv , "%\n")
greatest<-max(acchld,accrs,accv)
if(greatest==acchld){
cat("Holdout method does best classification.\n")
}else if(greatest==accrs){
cat("Random subsampling method does best classification.\n")
}else{
cat("cross validation method does best classification.\n")
}
#data scaled to standard format
sapply(iris[, -5], data_std)
iris_train <- iris_norm[1:100,]
iris_test <- iris_norm[101:150,]
pred <- knn(iris_train, iris_test, iris[1:100, 5], k = 13)
con <- table(pred, iris[101:150, 5])
accusd <- ((sum(diag(con))) / sum(con)) * 100
cat("Accuracy of standardized dataset is: ", accusd, "\n")
153:1 (Top Level)
R Script
Console
Environment History Connections
Import Dataset
List
Global Environment
x 150 obs. of 4 variables
Values
acc num [1:26] 94.7 94.6 94.5 94.4 94.4 ...
acc1 91.8918918918919
acc2 92
acc66 0
acc75 94.5945945945946
acchld 94.6666666666667
accrs 94.7604883202182
accsd 93.3333333333333
accu1 28.5714285714286
accu2 16.6666666666667
accu66 92
accusd 0
accv 96
acrs 94.0943399867983
Files Plots Packages Help Viewer
Zoom
Export

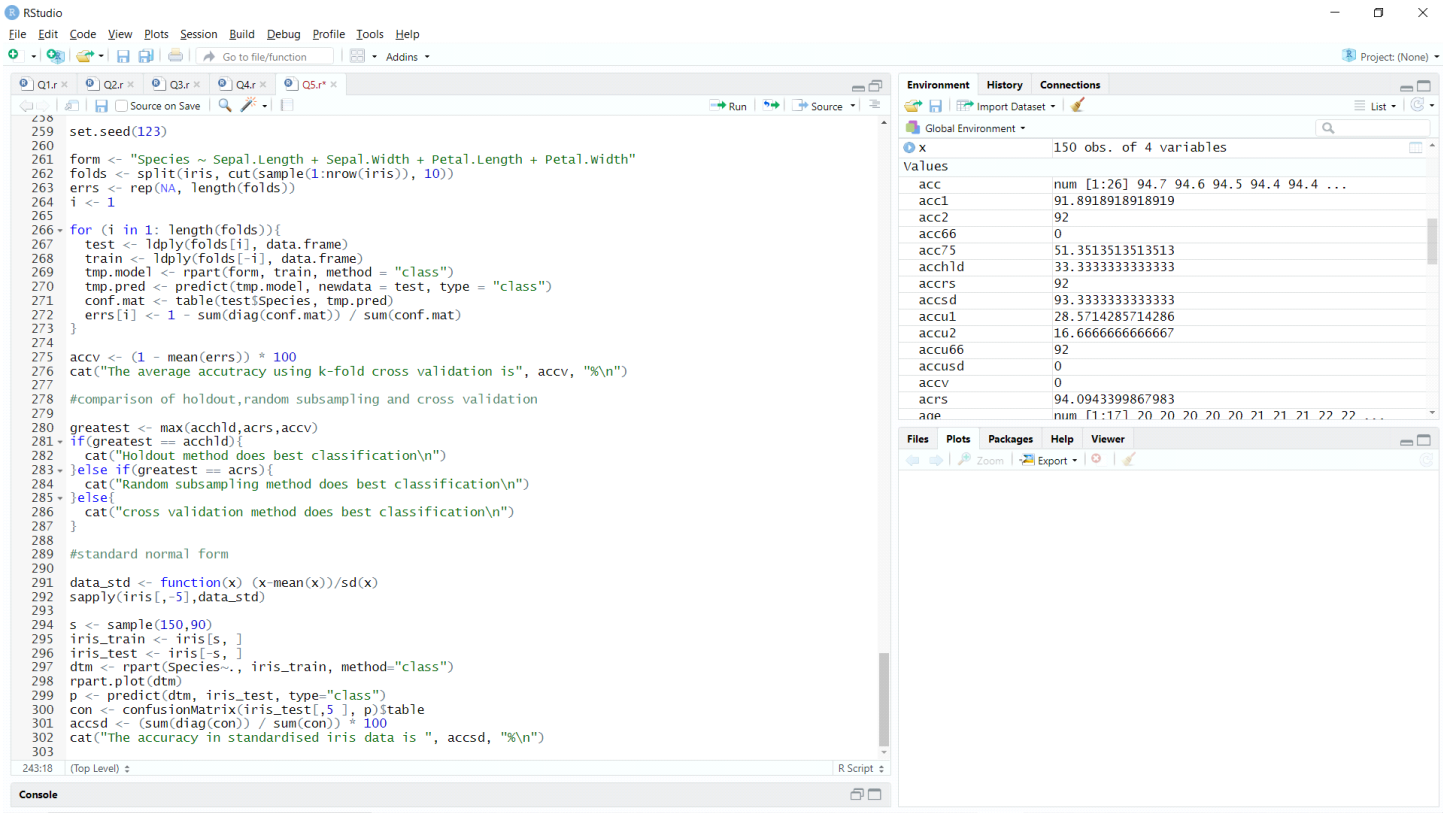
Output:



Decision tree algorithm:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Q1.r
Q2.r
Q3.r
Q4.r
Q5.r*
Source on Save
Run
Source
#Decision tree
##training set 75%
s <- sample(150, 113)
iris_train <- iris[s,]
iris_test <- iris[-s,]
dtm <- rpart(Species~., iris_train, method = "class")
rpart.plot(dtm)
pred <- predict(dtm, iris_test, type = "class")
con <- confusionMatrix(iris_test[, 5], pred)$table
acc75 <- (sum(diag(con)) / sum(con)) * 100
cat("The accuracy with 75% training data is ", acc75, "%\n")
##training set 66.6%
s <- sample(150, 100)
iris_train <- iris[s,]
iris_test <- iris[-s,]
dtm <- rpart(Species~., iris_train, method="class")
rpart.plot(dtm)
pred <- predict(dtm, iris_test, type="class")
con <- confusionMatrix(iris_test[, 5], pred)$table
accu66 <- (sum(diag(con)) / sum(con)) * 100
cat("The accuracy with 66.6% training data is ", accu66, "%\n")
#holdout method
s <- sample(150, 75)
iris_train <- iris[s,]
iris_test <- iris[-s,]
dtm <- rpart(Species~., iris_train, method="class")
rpart.plot(dtm)
pred <- predict(dtm, iris_test, type="class")
cn <- confusionMatrix(iris_test[, 5], pred)$table
acchld <- (sum(diag(con)) / sum(con)) * 100
cat("The accuracy with holdout method is ", acchld, "%\n")
#random subsampling
i <- 75
j <- 1
acc <- c()
for(i in 75:100){
243:18 (Top Level)
R Script
Console
Environment History Connections
Import Dataset
List
Global Environment
x 150 obs. of 4 variables
Values
acc num [1:26] 94.7 94.6 94.5 94.4 94.4 ...
acc1 91.8918918918919
acc2 92
acc66 0
acc75 51.3513513513513
acchld 33.3333333333333
accrs 92
accsd 93.3333333333333
accu1 28.5714285714286
accu2 16.6666666666667
accu66 92
accusd 0
accv 0
acrs 94.0943399867983
Files Plots Packages Help Viewer
Zoom
Export


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r Q4.r Q5.r*
Source on Save
Run
Source
acc <- c()
for(i in 75:100){
s <- sample(150, i)
iris_train <- iris[s,]
iris_test <- iris[-s,]
dtm <- rpart(Species~., iris_train, method="class")
rpart.plot(dtm)
pred <-predict(dtm, iris_test, type="class")
con <- confusionMatrix(iris_test[, 5], pred)$table
acc[j] <- c((sum(diag(con)) / sum(con)) * 100, acc)
j <- j+1
}
acrs <- mean(acc)
cat("The accuracy with random subsampling method is ", acrs, "%\n")
#cross validation
set.seed(123)
form <- "Species ~ Sepal.Length + Sepal.Width + Petal.Length + Petal.Width"
folds <- split(iris, cut(sample(1:nrow(iris)), 10))
errs <- rep(NA, length(folds))
i <- 1
for (i in 1: length(folds)){
test <- ldply(folds[i], data.frame)
train <- ldply(folds[-i], data.frame)
tmp.model <- rpart(form, train, method = "class")
tmp.pred <- predict(tmp.model, newdata = test, type = "class")
conf.mat <- table(test$Species, tmp.pred)
errs[i] <- 1 - sum(diag(conf.mat)) / sum(conf.mat)
}
accv <- (1 - mean(errs)) * 100
cat("The average accutracy using k-fold cross validation is", accv, "%\n")
#comparison of holdout,random subsampling and cross validation
greatest <- max(acchld,acrs,accv)
if(greatest == acchld){
cat("Holdout method does best classification\n")
}else if(greatest == acrs){
cat("Random subsampling method does best classification\n")
}else{
cat("cross validation method does best classification\n")
}
243:18 (Top Level)
R Script
Console
Environment History Connections
Import Dataset
List
Global Environment
x 150 obs. of 4 variables
Values
acc num [1:26] 94.7 94.6 94.5 94.4 94.4 ...
acc1 91.8918918918919
acc2 92
acc66 0
acc75 51.3513513513513
acchld 33.3333333333333
accrs 92
accsd 93.3333333333333
accu1 28.5714285714286
accu2 16.6666666666667
accu66 92
accusd 0
accv 0
acrs 94.0943399867983
Files Plots Packages Help Viewer
Zoom
Export


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r Q4.r Q5.r*
Source on Save
Run
Source
set.seed(123)
form <- "Species ~ Sepal.Length + Sepal.Width + Petal.Length + Petal.Width"
folds <- split(iris, cut(sample(1:nrow(iris)), 10))
errs <- rep(NA, length(folds))
i <- 1
for (i in 1: length(folds)){
test <- ldply(folds[i], data.frame)
train <- ldply(folds[-i], data.frame)
tmp.model <- rpart(form, train, method = "class")
tmp.pred <- predict(tmp.model, newdata = test, type = "class")
conf.mat <- table(test$Species, tmp.pred)
errs[i] <- 1 - sum(diag(conf.mat)) / sum(conf.mat)
}
accv <- (1 - mean(errs)) * 100
cat("The average accutracy using k-fold cross validation is", accv, "%\n")
#comparison of holdout,random subsampling and cross validation
greatest <- max(acchld,acrs,accv)
if(greatest == acchld){
cat("Holdout method does best classification\n")
}else if(greatest == acrs){
cat("Random subsampling method does best classification\n")
}else{
cat("cross validation method does best classification\n")
}
#standard normal form
data_std <- function(x) (x-mean(x))/sd(x)
sapply(iris[,-5],data_std)
s <- sample(150,90)
iris_train <- iris[s,]
iris_test <- iris[-s,]
dtm <- rpart(Species~., iris_train, method="class")
rpart.plot(dtm)
p <- predict(dtm, iris_test, type="class")
con <- confusionMatrix(iris_test[,5], p)$table
accsd <- (sum(diag(con)) / sum(con)) * 100
cat("The accuracy in standardised iris data is ", accsd, "%\n")
243:18 (Top Level)
R Script
Console
Environment History Connections
Import Dataset
List
Global Environment
x 150 obs. of 4 variables
Values
acc num [1:26] 94.7 94.6 94.5 94.4 94.4 ...
acc1 91.8918918918919
acc2 92
acc66 0
acc75 51.3513513513513
acchld 33.3333333333333
accrs 92
accsd 93.3333333333333
accu1 28.5714285714286
accu2 16.6666666666667
accu66 92
accusd 0
accv 0
acrs 94.0943399867983
Files Plots Packages Help Viewer
Zoom
Export

Output:

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function | Addins | Project: (None)

Source

Console | **Terminal** | **Jobs**

```
> source('C:/Users/Lenovo/Desktop/DM_files/Q5.r')
The accuracy with 75% training data is  91.89189 %
>
```

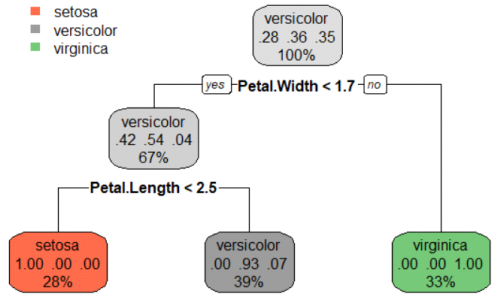
Environment | **History** | **Connections**

Import Dataset | List | Global Environment

i	100L
iris_pred	Factor w/ 3 levels "setosa","versicolor",..: 2 …
j	27
l	num [1:100] 95.5 95.5 95.5 95.5 95.5 ...
m	2L
Marks	num [1:3] 10 20 30
modeAge	"20"
n	2L
Name	chr [1:3] "ABC" "PQR" "XYZ"
num	int [1:50] 1 2 3 4 5 6 7 8 9 10 ...
p	Factor w/ 3 levels "setosa","versicolor",..: 1 …
pred	Factor w/ 3 levels "setosa","versicolor",..: 1 …
q	2L
rel	Named num [1:6] 0.333 0.833 1 0.333 0.667 ...
res	Factor w/ 3 levels "setosa","versicolor",..: 3 …
Roll	num [1:3] 1 2 3
s	int [1:113] 42 121 109 111 36 50 89 137 26 19 …

Files | **Plots** | **Packages** | **Help** | **Viewer**

Zoom | Export | Publish



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function | Addins | Project: (None)

Source

Console | **Terminal** | **Jobs**

```
> source('C:/Users/Lenovo/Desktop/DM_files/Q5.r')
The accuracy with 75% training data is  94.59459 %
The accuracy with 66.6% training data is  96 %
>
```

Environment | **History** | **Connections**

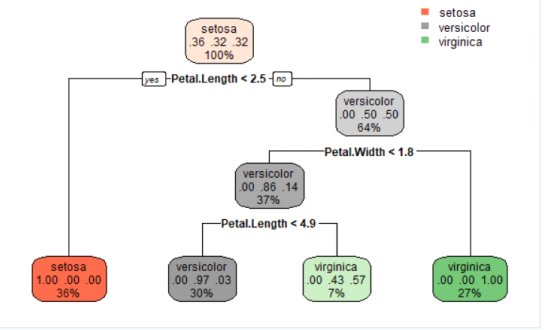
Import Dataset | List

Global Environment

i	100L
iris_pred	Factor w/ 3 levels "setosa","versicolor",..: 2 …
j	27
l	num [1:100] 95.5 95.5 95.5 95.5 95.5 ...
m	2L
Marks	num [1:3] 10 20 30
modeAge	"20"
n	2L
Name	chr [1:3] "ABC" "PQR" "XYZ"
num	int [1:50] 1 2 3 4 5 6 7 8 9 10 ...
p	Factor w/ 3 levels "setosa","versicolor",..: 1 …
pred	Factor w/ 3 levels "setosa","versicolor",..: 1 …
q	2L
rel	Named num [1:6] 0.333 0.833 1 0.333 0.667 ...
res	Factor w/ 3 levels "setosa","versicolor",..: 3 …
Roll	num [1:3] 1 2 3
s	int [1:100] 119 107 149 138 58 2 78 118 125 44 …

Files | **Plots** | **Packages** | **Help** | **Viewer**

Zoom | Export | Publish



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function | Addins | Project: (None)

Source

Console | **Terminal** | **Jobs**

```
> source('C:/Users/Lenovo/Desktop/DM_files/Q5.r')
The accuracy with 75% training data is  97.2973 %
The accuracy with 66.6% training data is  92 %
The accuracy with holdout method is  92 %
>
```

Environment | **History** | **Connections**

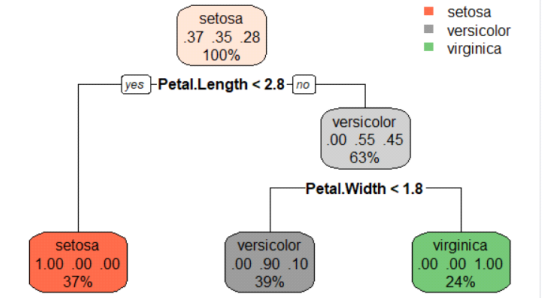
Import Dataset | List

Global Environment

i	100L
iris_pred	Factor w/ 3 levels "setosa","versicolor",..: 2 …
j	27
l	num [1:100] 95.5 95.5 95.5 95.5 95.5 ...
m	2L
Marks	num [1:3] 10 20 30
modeAge	"20"
n	2L
Name	chr [1:3] "ABC" "PQR" "XYZ"
num	int [1:50] 1 2 3 4 5 6 7 8 9 10 ...
p	Factor w/ 3 levels "setosa","versicolor",..: 1 …
pred	Factor w/ 3 levels "setosa","versicolor",..: 1 …
q	2L
rel	Named num [1:6] 0.333 0.833 1 0.333 0.667 ...
res	Factor w/ 3 levels "setosa","versicolor",..: 3 …
Roll	num [1:3] 1 2 3
s	int [1:75] 113 77 131 46 43 116 64 132 136 26 …

Files | **Plots** | **Packages** | **Help** | **Viewer**

Zoom | Export | Publish



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function | Addins | Project: (None)

Source

Console | **Terminal** | **Jobs**

```
> source('C:/Users/Lenovo/Desktop/DM_files/Q5.r')
The accuracy with 75% training data is  94.59459 %
The accuracy with 66.6% training data is  98 %
The accuracy with holdout method is  98 %
The accuracy with random subsampling method is  93.47799 %
There were 25 warnings (use warnings() to see them)
>
```

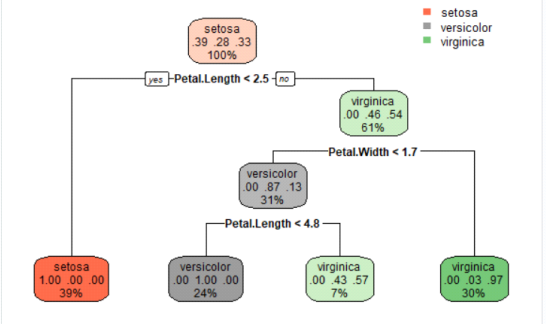
Environment | **History** | **Connections**

Import Dataset | List | Global Environment

i	100L
iris_pred	Factor w/ 3 levels "setosa","versicolor",..: 2 …
j	27
l	num [1:100] 95.5 95.5 95.5 95.5 95.5 ...
m	2L
Marks	num [1:3] 10 20 30
modeAge	"20"
n	2L
Name	chr [1:3] "ABC" "PQR" "XYZ"
num	int [1:50] 1 2 3 4 5 6 7 8 9 10 ...
p	Factor w/ 3 levels "setosa","versicolor",..: 1 …
pred	Factor w/ 3 levels "setosa","versicolor",..: 1 …
q	2L
rel	Named num [1:6] 0.333 0.833 1 0.333 0.667 ...
res	Factor w/ 3 levels "setosa","versicolor",..: 3 …
Roll	num [1:3] 1 2 3
s	int [1:100] 3 93 24 71 51 6 84 133 61 103 ...

Files | **Plots** | **Packages** | **Help** | **Viewer**

Zoom | Export | Publish



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function | Addins | Project: (None)

Source

Console | **Terminal** | **Jobs**

```
> source('C:/Users/Lenovo/Desktop/DM_files/Q5.r')
The accuracy with 75% training data is  97.2973 %
The accuracy with 66.6% training data is  94 %
The accuracy with holdout method is  94 %
The accuracy with random subsampling method is  93.80274 %
The average accutracy using k-fold cross validation is 94 %
Holdout method does best classification
There were 25 warnings (use warnings() to see them)
>
```

Environment | **History** | **Connections**

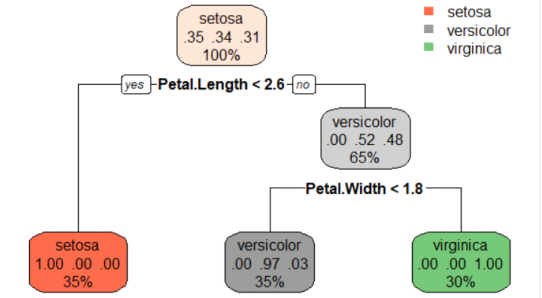
Import Dataset | List

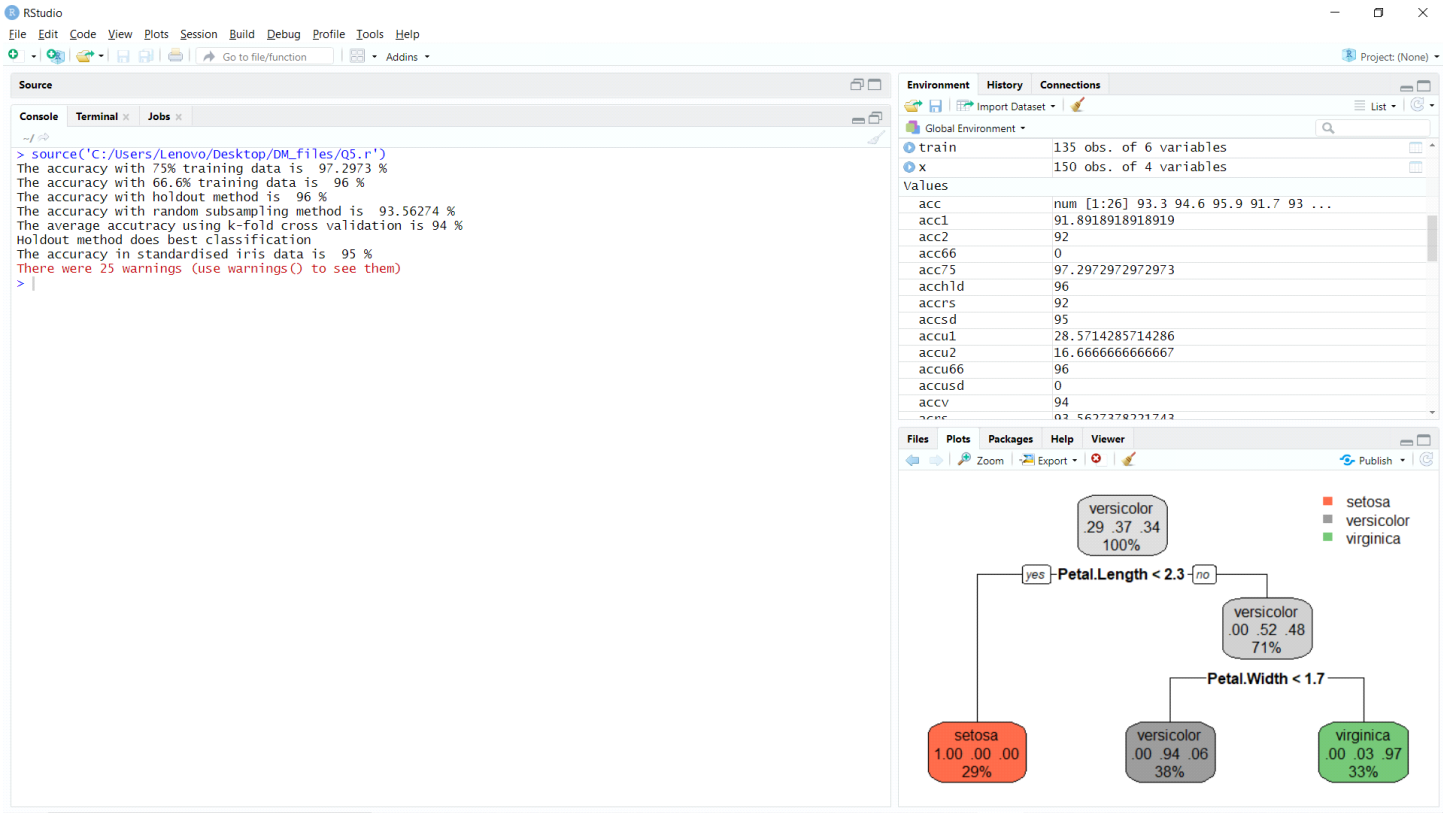
Global Environment

train	135 obs. of 6 variables
x	150 obs. of 4 variables
Values	
acc	num [1:26] 96 93.2 93.2 95.8 93 ...
acc1	91.8918918918919
acc2	92
acc66	0
acc75	97.2972972972973
acchld	94
accrs	92
accsd	93.3333333333333
accu1	28.5714285714286
accu2	16.6666666666667
accu66	94
accusd	0
accv	94
acrs	93.8027365364394

Files | **Plots** | **Packages** | **Help** | **Viewer**

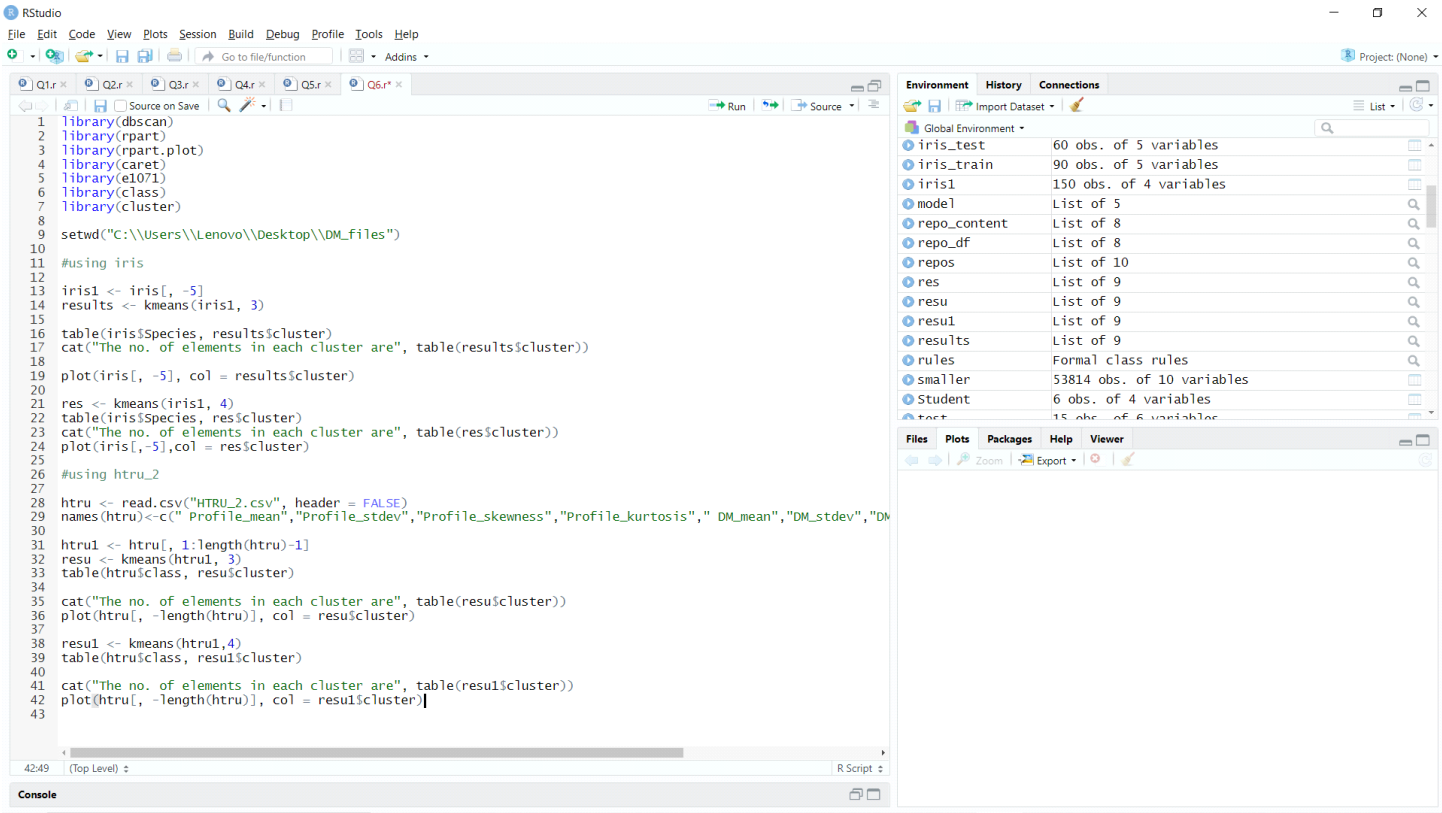
Zoom | Export | Publish





Q6. Use Simple K-means, DBScan, and Hierarchical clustering algorithms for clustering. Compare the performance of clusters by changing the parameters involved in the algorithms.

K-means algorithm:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r Q4.r Q5.r Q6.r*
Source on Save
Run
Source
1 library(dbscan)
2 library(rpart)
3 library(rpart.plot)
4 library(caret)
5 library(e1071)
6 library(class)
7 library(cluster)
8
9 setwd("C:\\Users\\Lenovo\\Desktop\\DM_files")
10
11 #using iris
12
13 iris1 <- iris[, -5]
14 results <- kmeans(iris1, 3)
15
16 table(iris$Species, results$cluster)
17 cat("The no. of elements in each cluster are", table(results$cluster))
18
19 plot(iris[, -5], col = results$cluster)
20
21 res <- kmeans(iris1, 4)
22 table(iris$Species, res$cluster)
23 cat("The no. of elements in each cluster are", table(res$cluster))
24 plot(iris[,-5],col = res$cluster)
25
26 #using htru_2
27
28 htru <- read.csv("HTRU_2.csv", header = FALSE)
29 names(htru)<-c(" Profile_mean","Profile_stdev","Profile_skewness","Profile_kurtosis"," DM_mean","DM_stdev","DM
30
31 htru1 <- htru[, 1:length(htru)-1]
32 resu <- kmeans(htru1, 3)
33 table(htru$class, resu$cluster)
34
35 cat("The no. of elements in each cluster are", table(resu$cluster))
36 plot(htru[, -length(htru)], col = resu$cluster)
37
38 resu1 <- kmeans(htru1,4)
39 table(htru$class, resu1$cluster)
40
41 cat("The no. of elements in each cluster are", table(resu1$cluster))
42 plot(htru[, -length(htru)], col = resu1$cluster)
43
42:49 (Top Level) R Script
Console
Environment History Connections
Import Dataset List
Global Environment
iris_test 60 obs. of 5 variables
iris_train 90 obs. of 5 variables
iris1 150 obs. of 4 variables
model List of 5
repo_content List of 8
repo_df List of 8
repos List of 10
res List of 9
resu List of 9
resu1 List of 9
results List of 9
rules Formal class rules
smaller 53814 obs. of 10 variables
Student 6 obs. of 4 variables
Files Plots Packages Help Viewer
Zoom Export

Output:

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function | Addins | Project: (None)

Source

Console | **Terminal** | **Jobs**

C:/Users/Lenovo/Desktop/DM_files/

```
> source('C:/Users/Lenovo/Desktop/DM_files/Q6.r')
The no. of elements in each cluster are 50 62 38
>
```

Environment | **History** | **Connections**

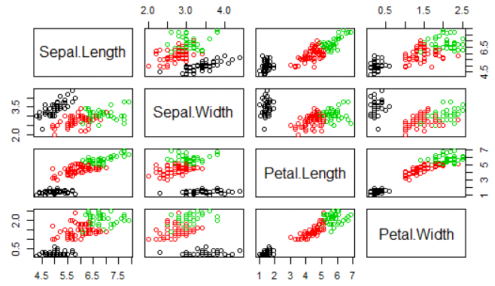
Import Dataset | List

Global Environment

iris_test	60 obs. of 5 variables
iris_train	90 obs. of 5 variables
iris1	150 obs. of 4 variables
model	List of 5
repo_content	List of 8
repo_df	List of 8
repos	List of 10
res	List of 9
resu	List of 9
resu1	List of 9
results	List of 9
rules	Formal class rules
smaller	53814 obs. of 10 variables
Student	6 obs. of 4 variables
test	15 obs. of 6 variables

Files | **Plots** | **Packages** | **Help** | **Viewer**

Zoom | Export | Publish



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function | Addins | Project: (None)

Source

Console **Terminal** **Jobs**

C:/Users/Lenovo/Desktop/DM_files/

```
> source('C:/Users/Lenovo/Desktop/DM_files/Q6.r')
The no. of elements in each cluster are 50 28 40 32
>
```

Environment **History** **Connections**

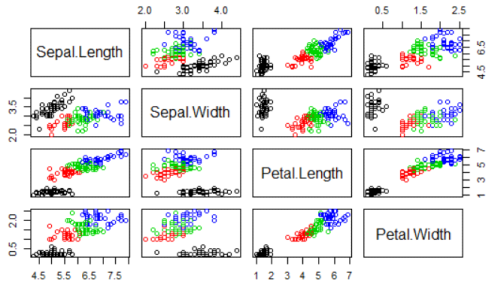
Import Dataset | List

Global Environment

iris_test	60 obs. of 5 variables
iris_train	90 obs. of 5 variables
iris1	150 obs. of 4 variables
model	List of 5
repo_content	List of 8
repo_df	List of 8
repos	List of 10
res	List of 9
resu	List of 9
resu1	List of 9
results	List of 9
rules	Formal class rules
smaller	53814 obs. of 10 variables
Student	6 obs. of 4 variables
test	15 obs. of 6 variables

Files **Plots** **Packages** **Help** **Viewer**

Zoom | Export | Publish



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function | Addins | Project: (None)

Source

Console **Terminal** **Jobs**

C:/Users/Lenovo/Desktop/DM_files/

```
> source('C:/Users/Lenovo/Desktop/DM_files/Q6.r')
The no. of elements in each cluster are 7023 9947 928
>
```

Environment **History** **Connections**

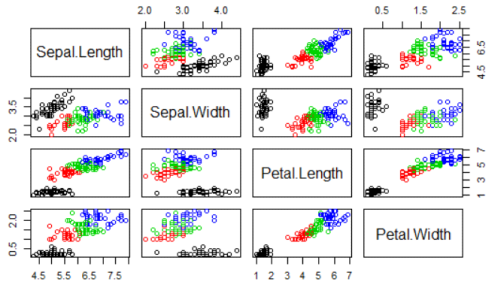
Import Dataset | List

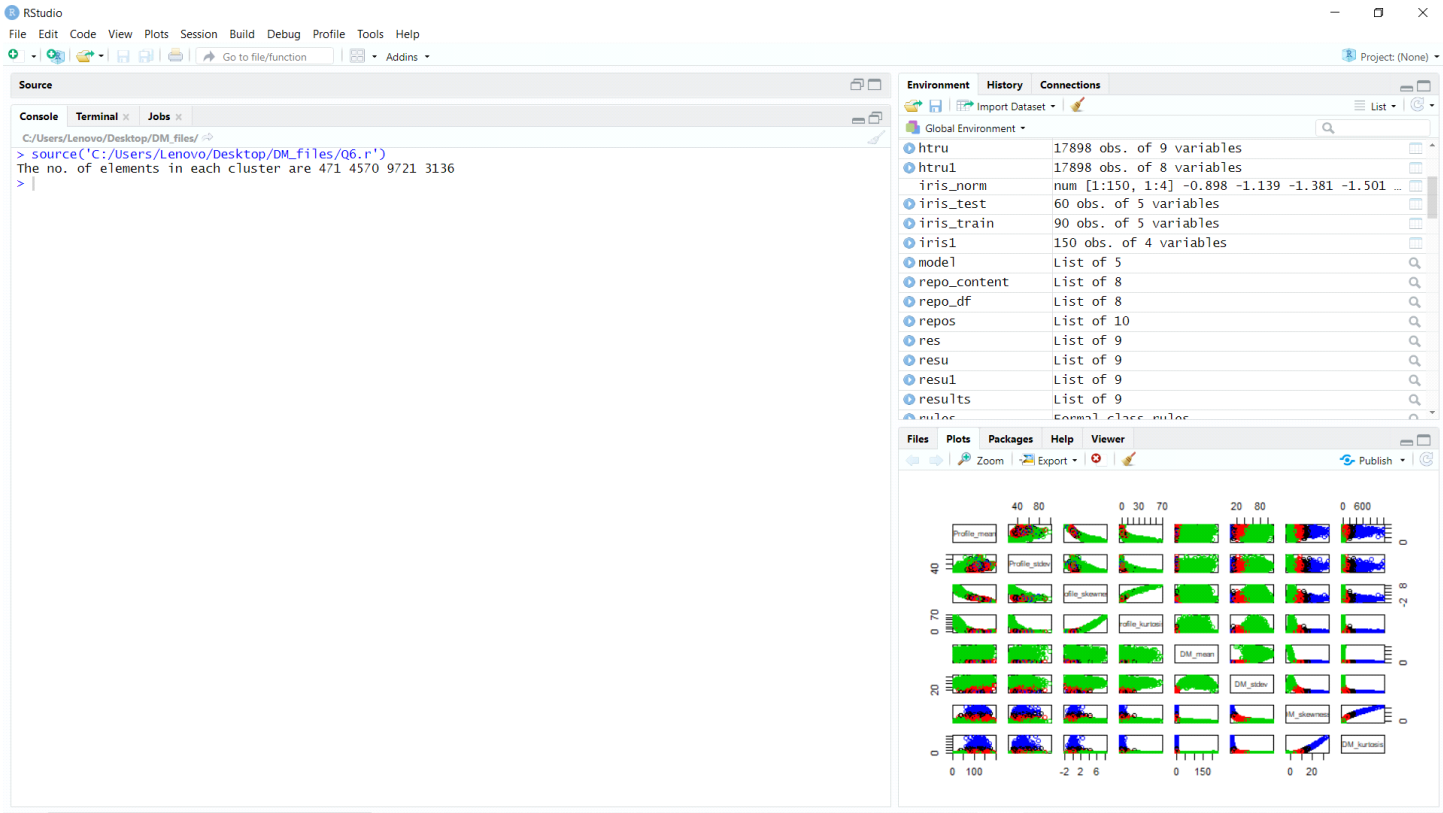
Global Environment

htru	17898 obs. of 9 variables
htru1	17898 obs. of 8 variables
iris_norm	num [1:150, 1:4] -0.898 -1.139 -1.381 -1.501 …
iris_test	60 obs. of 5 variables
iris_train	90 obs. of 5 variables
iris1	150 obs. of 4 variables
model	List of 5
repo_content	List of 8
repo_df	List of 8
repos	List of 10
res	List of 9
resu	List of 9
resu1	List of 9
results	List of 9
rules	Formal class rules

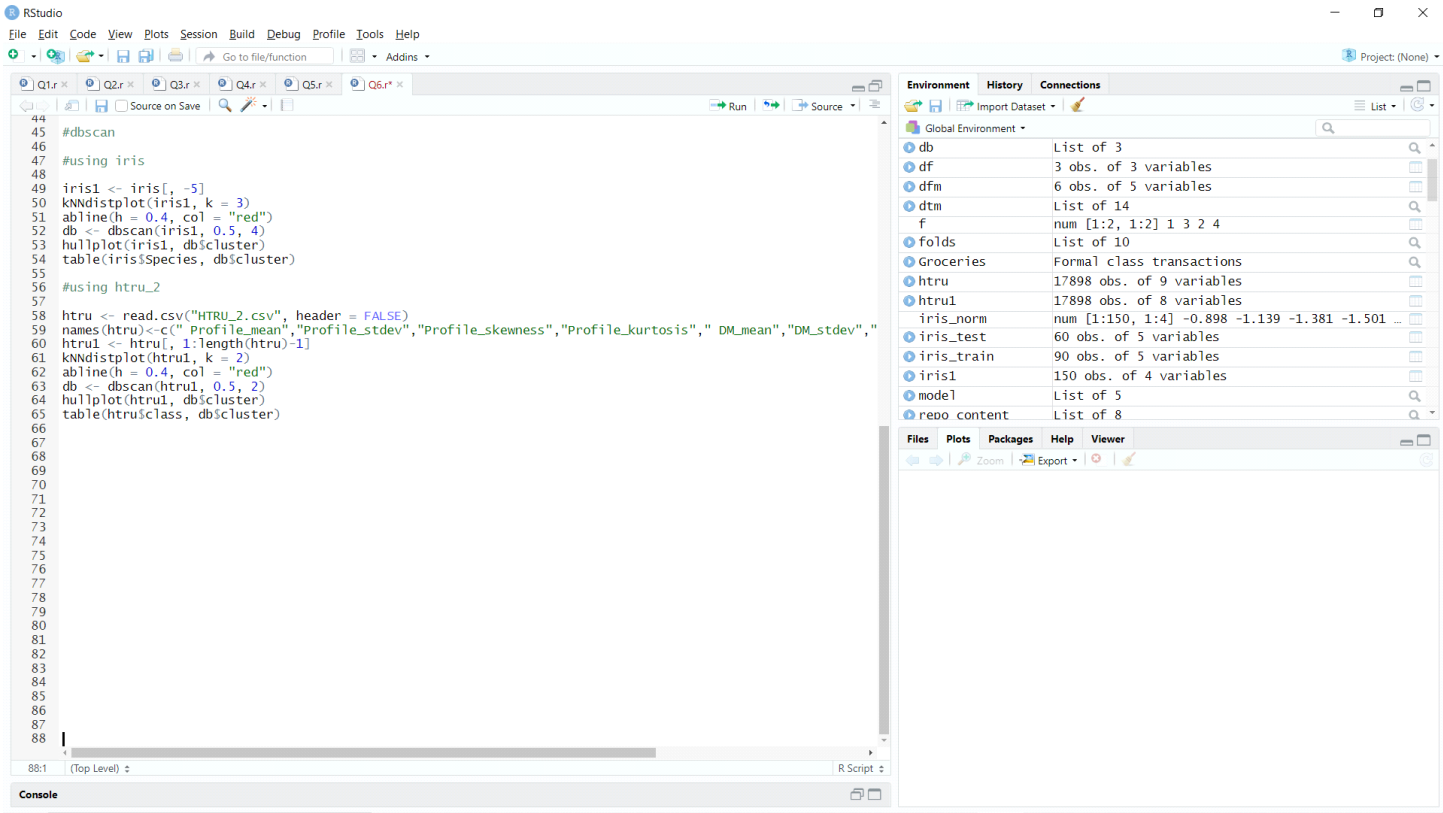
Files **Plots** **Packages** **Help** **Viewer**

Zoom | Export | Publish

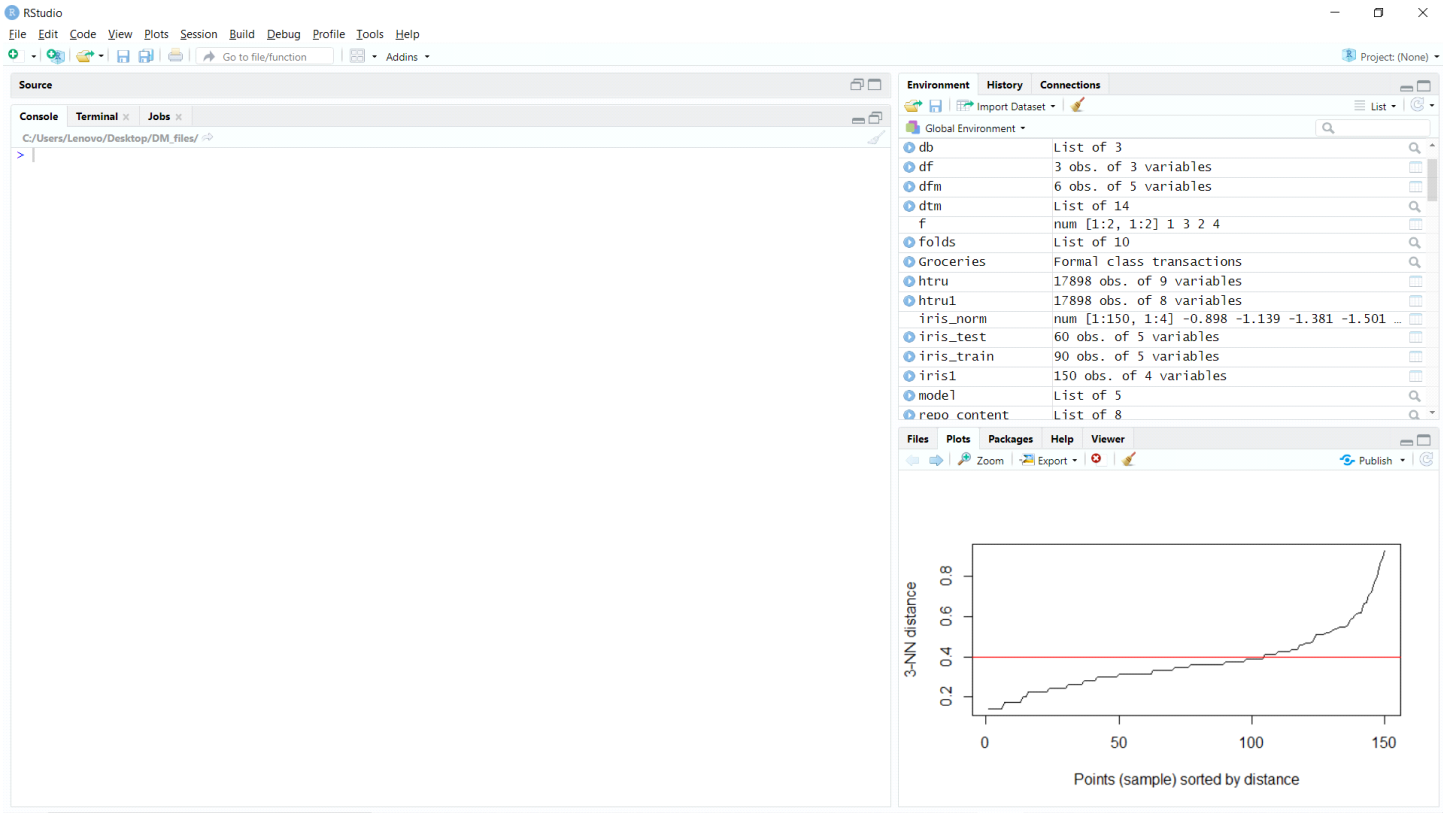




DBScan algorithm:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r Q4.r Q5.r Q6.r*
Source on Save
Run
Source
#dbscan
#using iris
iris1 <- iris[, -5]
kNNdistplot(iris1, k = 3)
abline(h = 0.4, col = "red")
db <- dbscan(iris1, 0.5, 4)
hullplot(iris1, db$cluster)
table(iris$Species, db$cluster)
#using htru_2
htru <- read.csv("HTRU_2.csv", header = FALSE)
names(htru)<-c(" Profile_mean","Profile_stdev","Profile_skewness","Profile_kurtosis"," DM_mean","DM_stdev","
htru1 <- htru[, 1:length(htru)-1]
kNNdistplot(htru1, k = 2)
abline(h = 0.4, col = "red")
db <- dbscan(htru1, 0.5, 2)
hullplot(htru1, db$cluster)
table(htru$class, db$cluster)
88:1 (Top Level)
R Script
Console
Environment History Connections
Import Dataset
Global Environment
db List of 3
df 3 obs. of 3 variables
dfm 6 obs. of 5 variables
dtm List of 14
f num [1:2, 1:2] 1 3 2 4
folds List of 10
Groceries Formal class transactions
htru 17898 obs. of 9 variables
htru1 17898 obs. of 8 variables
iris_norm num [1:150, 1:4] -0.898 -1.139 -1.381 -1.501 ...
iris_test 60 obs. of 5 variables
iris_train 90 obs. of 5 variables
iris1 150 obs. of 4 variables
model List of 5
repo content List of 8
Files Plots Packages Help Viewer
Zoom
Export

Output:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Source
Console
Terminal
Jobs
C:/Users/Lenovo/Desktop/DM_files/
>
Environment
History
Connections
Import Dataset
List
Global Environment
db List of 3
df 3 obs. of 3 variables
dfm 6 obs. of 5 variables
dtm List of 14
f num [1:2, 1:2] 1 3 2 4
folds List of 10
Groceries Formal class transactions
htru 17898 obs. of 9 variables
htru1 17898 obs. of 8 variables
iris_norm num [1:150, 1:4] -0.898 -1.139 -1.381 -1.501 ...
iris_test 60 obs. of 5 variables
iris_train 90 obs. of 5 variables
iris1 150 obs. of 4 variables
model List of 5
repo_content List of 8
Files
Plots
Packages
Help
Viewer
Zoom
Export
Publish
3-NN distance
0.2
0.4
0.6
0.8
0
50
100
150
Points (sample) sorted by distance

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Q1.r | Q2.r | Q3.r | Q4.r | Q5.r | Q6.r*

Source on Save | Run | Source

```
#
# resu1 <- kmeans(htru1,4)
# table(htru$class, resu1$cluster)
#
# cat("The no. of elements in each cluster are", table(resu1$cluster))
# plot(htru[, -length(htru)], col = resu1$cluster)

#dbscan

#using iris

iris1 <- iris[, -5]
kNNdistplot(iris1, k = 3)
abline(h = 0.4, col = "red")
db <- dbscan(iris1, 0.5, 4)
hullplot(iris1, db$cluster)
table(iris$Species, db$cluster)

#using htru_2

# htru <- read.csv("HTRU_2.csv", header = FALSE)
```

50:25 (Top Level) R Script

Console | Terminal | Jobs

C:/Users/Lenovo/Desktop/DM_files/

```
>
```

Environment | History | Connections

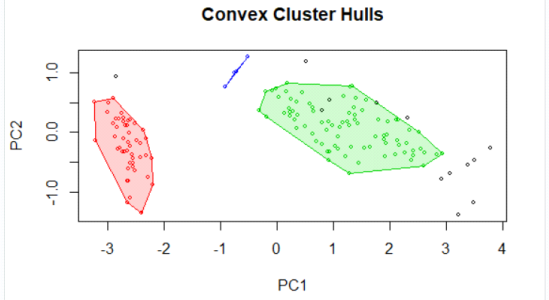
Import Dataset | List

Global Environment

db	List of 3
df	3 obs. of 3 variables
dfm	6 obs. of 5 variables
dtm	List of 14
f	num [1:2, 1:2] 1 3 2 4
folds	List of 10
Groceries	Formal class transactions
htru	17898 obs. of 9 variables
htru1	17898 obs. of 8 variables
iris_norm	num [1:150, 1:4] -0.898 -1.139 -1.381 -1.501 …
iris_test	60 obs. of 5 variables
iris_train	90 obs. of 5 variables
iris1	150 obs. of 4 variables
model	List of 5
repo_content	List of 8

Files | Plots | Packages | Help | Viewer

Zoom | Export | Publish



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function | Addins | Project: (None)

Source

Console | **Terminal** | **Jobs**

C:/Users/Lenovo/Desktop/DM_files/

```
>
```

Environment | **History** | **Connections**

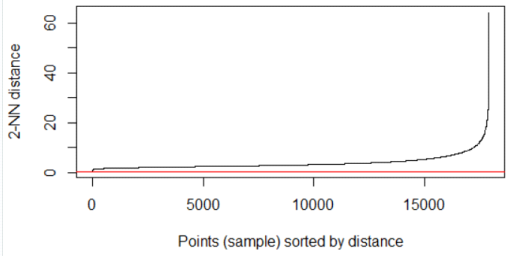
Import Dataset | List

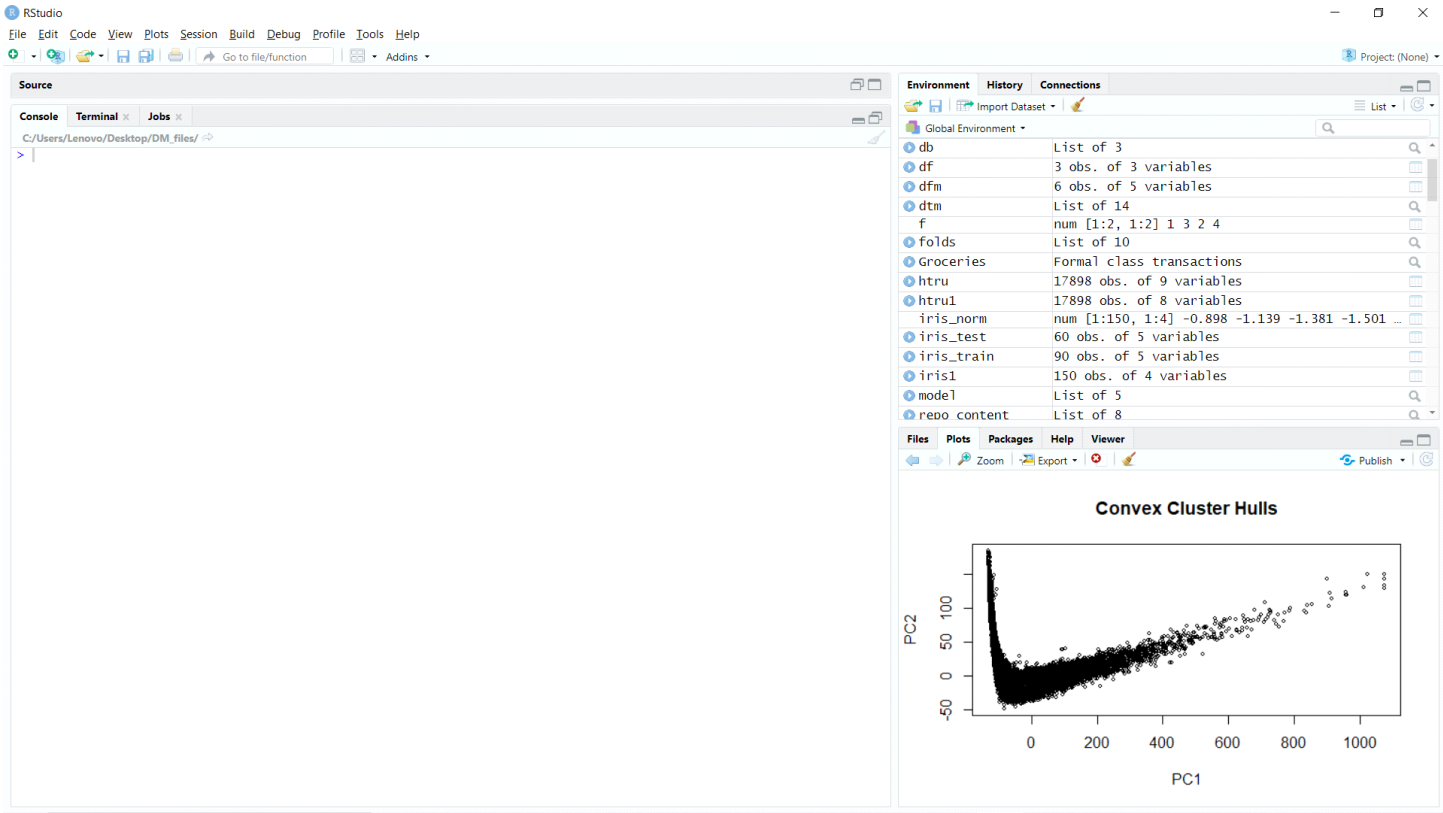
Global Environment

db	List of 3
df	3 obs. of 3 variables
dfm	6 obs. of 5 variables
dtm	List of 14
f	num [1:2, 1:2] 1 3 2 4
folds	List of 10
Groceries	Formal class transactions
htru	17898 obs. of 9 variables
htru1	17898 obs. of 8 variables
iris_norm	num [1:150, 1:4] -0.898 -1.139 -1.381 -1.501 …
iris_test	60 obs. of 5 variables
iris_train	90 obs. of 5 variables
iris1	150 obs. of 4 variables
model	List of 5
repo_content	List of 8

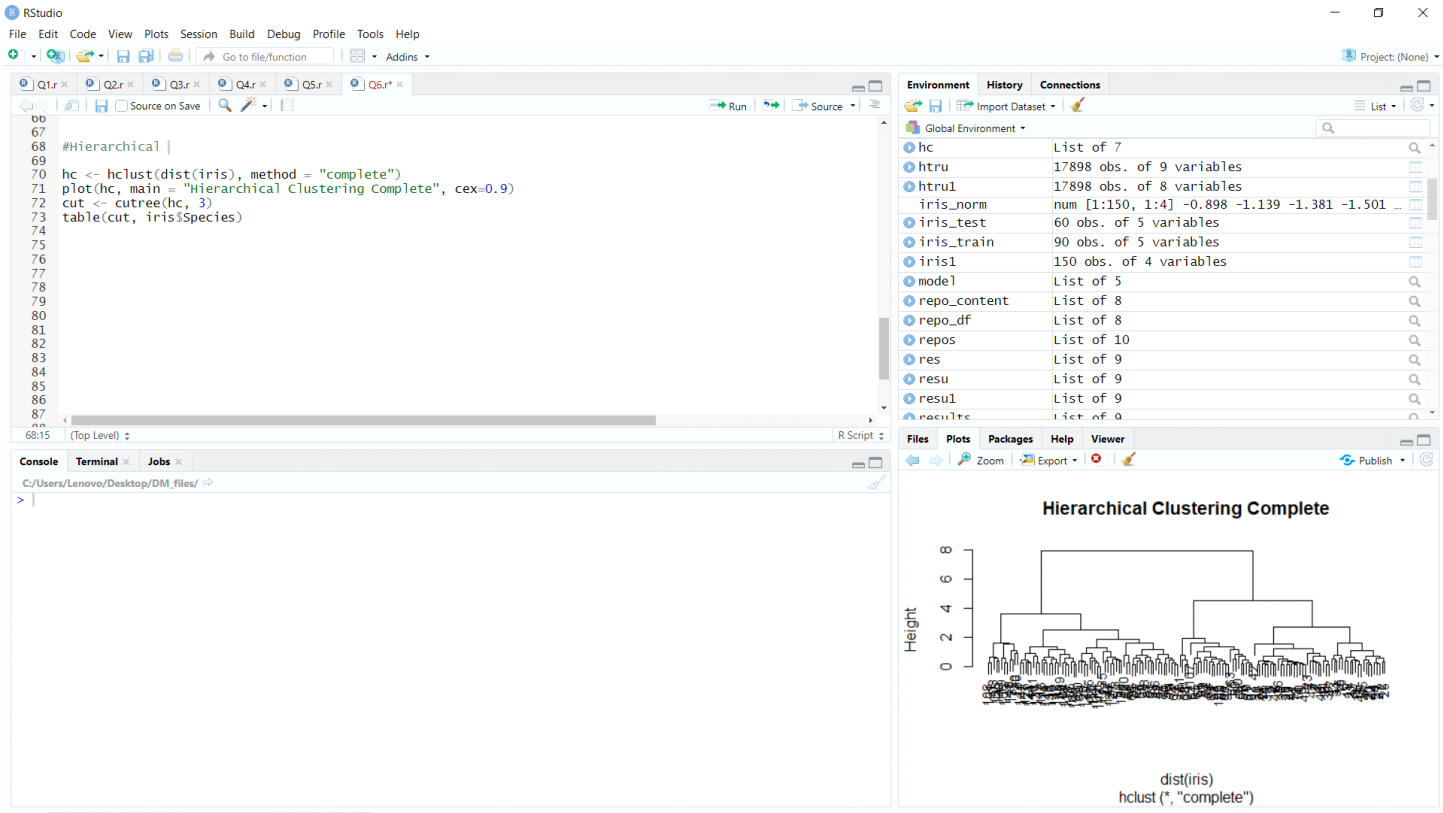
Files | **Plots** | **Packages** | **Help** | **Viewer**

Zoom | Export | Publish

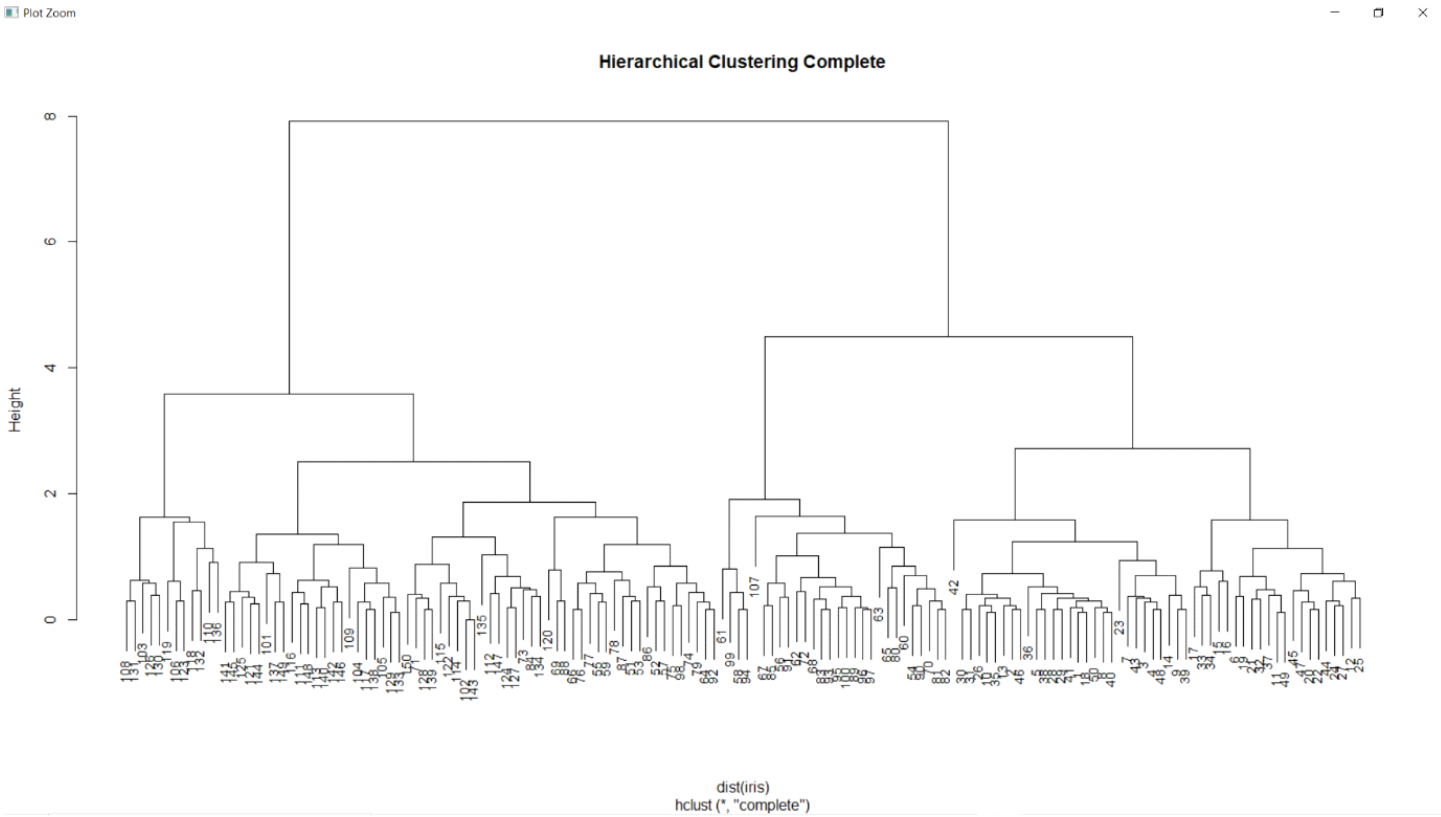




Hierarchical algorithm:


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Q1.r Q2.r Q3.r Q4.r Q5.r Q6.r*
Source on Save
Run
Source
#Hierarchical
hc <- hclust(dist(iris), method = "complete")
plot(hc, main = "Hierarchical Clustering Complete", cex=0.9)
cut <- cutree(hc, 3)
table(cut, iris$Species)
68:15 (Top Level)
R Script
Console Terminal Jobs
C:/Users/Lenovo/Desktop/DM_files/
Environment History Connections
Import Dataset
Global Environment
hc List of 7
htru 17898 obs. of 9 variables
htru1 17898 obs. of 8 variables
iris_norm num [1:150, 1:4] -0.898 -1.139 -1.381 -1.501
iris_test 60 obs. of 5 variables
iris_train 90 obs. of 5 variables
iris1 150 obs. of 4 variables
model List of 5
repo_content List of 8
repo_df List of 8
repos List of 10
res List of 9
resu List of 9
resu1 List of 9
Files Plots Packages Help Viewer
Zoom
Export
Publish
Hierarchical Clustering Complete
Height
0
2
4
6
8
dist(iris)
hclust (*, "complete")

Output:


Plot Zoom
Hierarchical Clustering Complete
Height
0
2
4
6
8
dist(iris)
hclust (*, "complete")