Module - 5: Machine Learning Techniques using R Part-3

Assignment Solution

Version 2.0

1. Implement the Random-Forest Technique on the Diabetes.csv Dataset.

Package to be used: randomForest

The variable description is:

For Each Attribute: (all numeric-valued)

1. Number of times pregnant
2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test
3. Diastolic blood pressure (mm Hg)
4. Triceps skin fold thickness (mm)
5. 2-Hour serum insulin (mu U/ml)
6. Body mass index (weight in kg/(height in m)^2)
7. Diabetes pedigree function
8. Age (years)
9. Class variable (0 or 1)

For more information please go through the 'Diabetes Description.pdf'.

Solution:

```
setwd("C:\\PERSONAL\\edureka\\Module-4")

DB<-read.csv("Diabetes.csv",head=T)

head(DB)

nrow(DB)

set.seed(2)
DB$ind<-sample(2,nrow(DB),replace=TRUE,prob=c(0.7,0.3))

head(DB)

trainData<-DB[(DB$ind==1),]
testData<-DB[(DB$ind==2),]

nrow(trainData)
nrow(testData)

library(rpart)

head(trainData)
```

```
dt<-rpart(Class.variable~Number.of.times.pregnant
                        +Plasma.glucose.concentration
                        +Diastolic.blood.pressure
                        +Triceps.skin.fold.thickness
                        +X2.Hour.serum.insulin
                        +Body.mass.index
                        +Diabetes.pedigree.function
                        +Age..years.,
          data=trainData,
          control=rpart.control(minsplit=10))

dt

plot(dt)
text(dt)

####Predict####
pred<-predict(dt,testData,type=c("class"))
pred

cbind(as.character(testData$Class.variable),as.character(pred))

#####confusion matrix######
table(as.character(testData$Class.variable),as.character(pred))


####Predict-Prob#####
pred1<-predict(dt,testData,type=c("prob"))
pred1

head(pred)
head(pred1)

attributes(dt)

dt$variable.importance


#####New Prediction######
New<-read.csv("New.csv",head=T)
predict(dt,New,type=c("class"))
```

```
############RANDOM FOREST##############################


setwd("C:\\PERSONAL\\edureka\\Module-4")

DB<-read.csv("Diabetes.csv",head=T)

head(DB)

nrow(DB)

set.seed(2)
DB$ind<-sample(2,nrow(DB),replace=TRUE,prob=c(0.7,0.3))

head(DB)

trainData<-DB[(DB$ind==1),]
testData<-DB[(DB$ind==2),]

nrow(trainData)
nrow(testData)

library(randomForest)

rf<-randomForest(Class.variable~Number.of.times.pregnant
                                +Plasma.glucose.concentration
                                +Diastolic.blood.pressure
                                +Triceps.skin.fold.thickness
                                +X2.Hour.serum.insulin
                                +Body.mass.index
                                +Diabetes.pedigree.function
                                +Age..years.,
                 data=trainData,ntree=600,mtry=4)


rf

attributes(rf)

rf$importance

predRF<-predict(rf,testData,type=c("class"))

#####confusion matrix######
table(as.character(testData$Class.variable),as.character(predRF))
```

2. Implement the Naïve-Bayesian Technique on the Diabetes.csv Dataset.

Package to be used: e1071

Solution:

```
#####################NAIVE BAYES###################

setwd("C:\\PERSONAL\\edureka\\Module-4")

DB<-read.csv("Diabetes.csv",head=T)

head(DB)

nrow(DB)

set.seed(2)
DB$ind<-sample(2,nrow(DB),replace=TRUE,prob=c(0.7,0.3))

head(DB)

trainData<-DB[(DB$ind==1),]
testData<-DB[(DB$ind==2),]

nrow(trainData)
nrow(testData)

library(e1071)

NB<-naiveBayes(Class.variable~Number.of.times.pregnant
                              +Plasma.glucose.concentration
                              +Diastolic.blood.pressure
                              +Triceps.skin.fold.thickness
                              +X2.Hour.serum.insulin
                              +Body.mass.index
                              +Diabetes.pedigree.function
                              +Age..years.,
               data=trainData)

attributes(NB)

NB$apriori
NB$tables


predNB<-predict(NB,testData,type=c("class"))
```

3. Create the Confusion Matrix for the Random Forest and Naïve Bayes Implementation.

Solution:

```
###Confusion Matrix####
table(testData$Class.variable,predNB)



head(trainData)

trainData1<-trainData[,-10]

head(trainData1)
NB1<-naiveBayes(Class.variable~.,
                data=trainData1)


predNB1<-predict(NB1,testData,type=c("class"))


###Confusion Matrix####
table(testData$Class.variable,predNB1)
```