Wyniki

Started on	czwartek, 11 stycznia 2024, 10:22
State	Finished
Completed on	czwartek, 11 stycznia 2024, 11:21
Time taken	58 mins 52 secs

Grade Not yet graded

Question 1

Complete Marked out of 25.00

For the dataset water.csv, to be downloaded from the data directory (OneDrive)

- transform the variable Potability into a factor;
- build a decision tree;
- evaluate the classifier, provide (copy to answer) the confusion matrix and AUC value;
- give the numbers of the leaves where the observations will go.

ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity
6.425874	188.9803	11965.40	7.571123	365.4537	502.9911	13.86649	96.81811	3.731937
7.748655	239.7883	29331.24	10.713097	217.0006	441.5295	16.38938	63.19615	2.511810
5.374223	201.3314	19410.23	3.239580	384.5628	350.1780	13.45109	66.18096	3.642778

In the task, set seed=123.

```
# EXERCISE 1

# a) transform the variable Potability into a factor

dataset = read.csv("water.csv")
dataset$Potability=as.factor(dataset$Potability)

# b) build a decision tree

# We divide the full set into a training set and a test set.
# We place 70% of the observations in the training set and 30%
# of all observations in the test set.

library(caTools)
set.seed(123)
split = sample.split(dataset$Potability, SplitRatio = 0.7)
```

```
training_set = subset(dataset, split == TRUE)
test_set = subset(dataset, split == FALSE)

# Now, we use the rpart algorithm to build our decision tree model.

library(rpart)
model= rpart(formula = Potability ~ ., data = training_set)

# c) evaluate the classifier, provide (copy to answer) the confusion
# matrix and AUC value

# Confusion matrix

y_pred = predict(model, newdata = test_set, type = 'class')


y_pred
0 1
0 579 20
1 340 43


table(test_set$Potability, y_pred)

# AUC value

library(ROCR)

roc.function<-function(y_pred,testY){
pred <- prediction(as.numeric(y_pred), as.numeric(testY))
perf.auc <- performance(pred, measure = "auc")
auc<-round(unlist(perf.auc@y.values),2)
perf <- performance(pred,"tpr","fpr")
plot(perf,main=paste("ROC curve and AUC=",auc),colorize=TRUE, lwd = 3)
abline(a = 0, b = 1, lwd = 2, lty = 2)
}

roc.function(y_pred,test_set$Potability) # The value of AUC is 0.54

# The accuracy is also a good way to evaluate the classifier
```

```
acc<-function(y1,y2){
sum(y1==y2)/length(y1)
}

acc(y_pred,test_set$Potability)

# The value of AUC is 0.54 and acc = 0.6334012. We can say
# that our model is not very good because it predicts
# correctly a bit more than a half of the cases. It also
# has lots of FALSE NEGATIVES.

# d) give the numbers of the leaves where the observations will go

library(rattle)
asRules(model)

# The numbers of the leaves are 4, 5, 12, 13, 14 and 15
```

Question 2

Complete Marked out of 15.00

For the dataset water.csv, to be downloaded from the data directory (OneDrive)

- apply the knn algorithm;
- determine the best value of k; build a knn model for the optimal k;
- provide (copy to answer) the confusion matrix for the given classifier.

```
# EXERCISE 2

# a) apply the knn algorithm

# First we have normalize all the values in our dataset

normalize <- function(x) {
return ((x - min(x)) / (max(x) - min(x)))
}

# The 10th column is what we want to predict and it is a
# factor so we need to delete it to be able to use
# normalization. Later, we add it again

water_n <- as.data.frame(lapply(dataset[-10], normalize))
water_n=cbind(dataset[10],water_n)


# Now, let's be careful because potability is in the first column

# after using cbind

# We divide the full set into a training set and a test set.
# We place 70% of the observations in the training set and 30%
# of all observations in the test set.
```

```
set.seed(123)
Train_Test <- sample(c("train","test"),nrow(dataset),replace =TRUE, prob = c(0.7,0.3))

water_train=water_n[Train_Test=="train",]
water_test=water_n[Train_Test=="test",]

# Now, we will apply the knn algorithm using k = 20 neighbour

library(class)
water_test_pred <- knn(train = water_train[-1], test = water_test[-1],
cl = water_train$Potability, k = 20)

table(water_test$Potability, water_test_pred)

# b) determine the best value of k; build a knn model for the optimal k

k <- c(2:15,seq(21,50,4))
n<-nrow(water_test)
knn_acc <- NULL
for(i in 1:length(k)){
knn_test <- knn(train = water_train[-1], test = water_test[-1],
cl = water_train$Potability, k=k[i])
knn_acc <- c(knn_acc,sum(water_test$Potability==knn_test)/n)
}

knn_acc
df=data.frame(k,knn_acc)
m=max(knn_acc)

library(dplyr)
filter(df,df$knn_acc==m)
plot(k,df$knn_acc, type = "b")

# The best value of k is 12

# c) provide (copy to answer) the confusion matrix for the given classifier

water_test_pred_opty <- knn(train = water_train[-1], test = water_test[-1],
cl = water_train$Potability, k = 12)
```

```
# Confusion matrix

table(water_test$Potability,water_test_pred_opty)

water_test_pred_opty
0 1
0 520 92
1 248 127
```