Programming Assignment Unit 5_R

October 5, 2025

1 ==========

2 Ionosphere Radar Data - Decision Tree Classification (R)

3 Student: Sana Ur Rehman Arain

4 Course: Data Mining & Mahcine Learning

5 Assignment Activity Unit 5

6 ==========

6.1 Part 1: Print decision tree

6.1.1 a. We begin by setting the working directory, loading the required packages (rpart and mlbench) and then loading the Ionosphere dataset.

[24]:
```
# Set working directory (modify as needed)
setwd("C:/Users/Yahya/Desktop/My folder/UOPEOPLE/Computer Science/CS 4407 Data␣
↪Mining & Machine Learning")
```

[25]:
```
# Load required libraries
library(rpart)
library(mlbench)
data(Ionosphere)
```

6.1.2 b. Use the rpart() method to create a regression tree for the data.

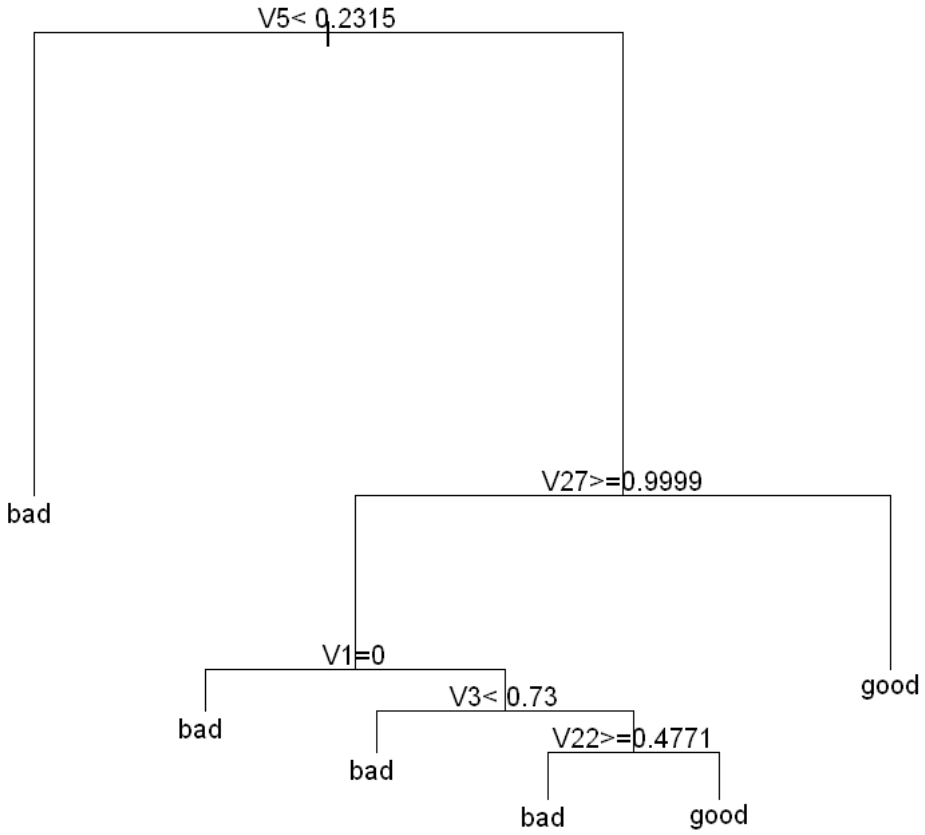
[26]:
```
rpart.ionosphere=rpart(Class~.,Ionosphere)
rpart.ionosphere
```

```
n= 351

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 351 126 good (0.35897436 0.64102564)
   2) V5< 0.23154 77   4 bad (0.94805195 0.05194805) *
```

```
3) V5>=0.23154 274  53 good (0.19343066 0.80656934)
  6) V27>=0.999945 52  13 bad (0.75000000 0.25000000)
   12) V1=0 19   0 bad (1.00000000 0.00000000) *
   13) V1=1 33  13 bad (0.60606061 0.39393939)
     26) V3< 0.73004 8   0 bad (1.00000000 0.00000000) *
     27) V3>=0.73004 25  12 good (0.48000000 0.52000000)
       54) V22>=0.47714 9   1 bad (0.88888889 0.11111111) *
       55) V22< 0.47714 16   4 good (0.25000000 0.75000000) *
  7) V27< 0.999945 222  14 good (0.06306306 0.93693694) *
```

6.1.3 c. Use the plot() and text() methods to plot the decision tree.

[27]:
```
plot(rpart.ionosphere)
text(rpart.ionosphere,pretty=0)
```



6.2 Part 2: Estimate accuracy

6.2.1 a. Split the data a test and train subsets using the sample() method.

[28]:
```
set.seed = (42)
train=sample(1:nrow(Ionosphere),200)
```

6.2.2 b. Use the rpart method to create a decision tree using the training data.

[29]:
```
rpart.ionosphere=rpart(Class~.,Ionosphere,subset=train)
rpart.ionosphere
```

```
n= 200

node), split, n, loss, yval, (yprob)
      * denotes terminal node

1) root 200 74 good (0.37000000 0.63000000)
  2) V5< 0.2318 44  2 bad (0.95454545 0.04545455) *
  3) V5>=0.2318 156 32 good (0.20512821 0.79487179)
    6) V27>=0.99921 28  7 bad (0.75000000 0.25000000) *
    7) V27< 0.99921 128 11 good (0.08593750 0.91406250) *
```

6.2.3 c. Use the predict method to find the predicted class labels for the testing data.

[31]:
```
# Create training and test sets
Ionosphere.train <- Ionosphere[train, ]
Ionosphere.test  <- Ionosphere[-train, ]
```

[32]:
```
# Train model on training data
library(rpart)
rpart.ionosphere <- rpart(Class ~ ., data = Ionosphere.train)
```

[33]:
```
rpart.pred=predict(rpart.ionosphere,Ionosphere.test,type="class")
```

[34]:
```
# View predictions
head(rpart.pred)
```

3 good **7** good **8** bad **14** bad **15** bad **16** good

Levels: 1. 'bad' 2. 'good'

6.2.4 d. Use the table method to create a table of the predictions versus true labels and then compute the accuracy. The accuracy is the number of correctly assigned good cases (true positives) plus the number of correctly assigned bad cases (true negatives) divided by the total number of testing cases.

[35]:
```
table(rpart.pred,Ionosphere$Class[-train])
```

```
rpart.pred bad good
      bad   49    9
      good   3   90
```

[36]:
```
(37+95)/(37+3+16+95)
```

0.874172185430464

[ ]:
```
cat("\n")
cat(rep("=", 70), sep="", "\n")
cat("ANALYSIS COMPLETE\n")
cat(rep("=", 70), sep="", "\n")
```

```

======================================================================
ANALYSIS COMPLETE
======================================================================
```