Experiment 2.1

Student Name: Mayank Kumar **UID: 20BCS1353**
Branch: CSE **Section/Group: DM_705_A**
Semester: 6TH **Date_of_Performance:03/04/2023**
Subject Name: Data Mining

1. **Aim/Overview of the practical:** Performing classification by decision tree induction using WEKA tools.

2. **Theory:**

Decision tree algorithm falls under the category of supervised learning. They can be used to solve both regression and classification problems. Decision tree uses the tree representation to solve the problem in which each leaf node corresponds to a class label and attributes are represented on the internal node of the tree. We can represent any boolean function on discrete attributes using the decision tree. In the Decision Tree, the major challenge is the identification of the attribute for the root node at each level. This process is known as attribute selection.

We have two popular attribute selection measures:

1. Information Gain
2. Gini Index

3. **Steps for experiment/practical/Code:**

```
library(RWeka)

library(partykit)

library(caTools)

setwd("C://Users//Ncs//Desktop//r work")
```

```
iris_data = iris

str(iris_data)

summary(iris_data)

spl = sample.split(iris_data, SplitRatio = 0.7)

dataTrain = subset(iris_data, spl==TRUE)
dataTest = subset(iris_data, spl==FALSE)

m1 <- J48(Species~., dataTrain)
summary(m1)

dataTestPred <- predict(m1, newdata = dataTest)
table_matrix <- table(dataTest$Species, dataTestPred)

print(table_matrix)

accuracy_Test <- sum(diag(table_matrix)) / sum(table_matrix)

cat("Test Accuracy is: ", accuracy_Test)
```

```
# Initate PDF File

pdf("Iris_decision_plot.pdf", paper="a4")


plot(m1, type="simple")


#Close PDF file

dev.off()
```

Output-

Console | Terminal | Background Jobs

R 4.2.2 · C:/Users/Ncs/Desktop/r work/

```
> library(RWeka)
> library(partykit)
> library(caTools)
> setwd("C://Users//Ncs//Desktop//r work")
> iris_data = iris
>
> str(iris_data)
'data.frame':   150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
>
> summary(iris_data)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
>
> spl = sample.split(iris_data, SplitRatio = 0.7)
>
> dataTrain = subset(iris_data, spl==TRUE)
> dataTest = subset(iris_data, spl==FALSE)
>
> m1 <- J48(Species~., dataTrain)
> summary(m1)

=== Summary ===

Correctly Classified Instances          89               98.8889 %
Incorrectly Classified Instances         1                1.1111 %
```

```
Kappa statistic                          0.9833
Mean absolute error                      0.0119
Root mean squared error                  0.077
Relative absolute error                  2.6667 %
Root relative squared error             16.3299 %
Total Number of Instances               90

=== Confusion Matrix ===

  a  b  c   <-- classified as
 30  0  0 |  a = setosa
  0 29  1 |  b = versicolor
  0  0 30 |  c = virginica
>
> dataTestPred <- predict(m1, newdata = dataTest)
> table_matrix <- table(dataTest$Species, dataTestPred)
>
> print(table_matrix)
            dataTestPred
             setosa versicolor virginica
  setosa         20          0         0
  versicolor      0         18         2
  virginica       0          0        20
>
> accuracy_Test <- sum(diag(table_matrix)) / sum(table_matrix)
>
> cat("Test Accuracy is: ", accuracy_Test)
Test Accuracy is:  0.9666667>
> # Initate PDF File
> pdf("Iris_decision_plot.pdf", paper="a4")
>
> plot(m1, type="simple")
>
> #Close PDF file
> dev.off()
```

Decision Tree: