Experiment 5

Student Name:Kanishk Soni **UID: 20BCS9398**
Branch: BE-CSE **Section/Group: 20BCS_DM_708-B**
Semester: 6th
Subject Name: Data Mining Lab **Subject Code: 20CSP-376**

1. Aim/Overview of the practical:

To perform the classification by decision tree induction using WEKA tools.

2. Task to be done:

To perform the classification by decision tree induction using WEKA tools.

3. Apparatus/Simulator used:

- RStudio
- RWeka

4. Decision Tree:

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.

As you can see from the above image the Decision Tree works on the Sum of Product form which is also known as *Disjunctive Normal Form*. In the above image, we are predicting the use of computer in the daily life of people. 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

1. Information Gain When we use a node in a decision tree to partition the training instances into smaller subsets the entropy changes. Information gain is a measure of this change in entropy. ***Definition***: Suppose S is a set of instances, A is an attribute, S_v is the subset of S with A = v, and Values (A) is the set of all possible values of A, then **Entropy** Entropy is the measure of uncertainty of a random variable, it characterizes the impurity of an arbitrary collection of examples. The higher the entropy more the information content. ***Definition***: Suppose S is a set of instances, A is an attribute, S_v is the subset of S with A = v, and Values (A) .

5. Dataset Used :

PlantGrowth

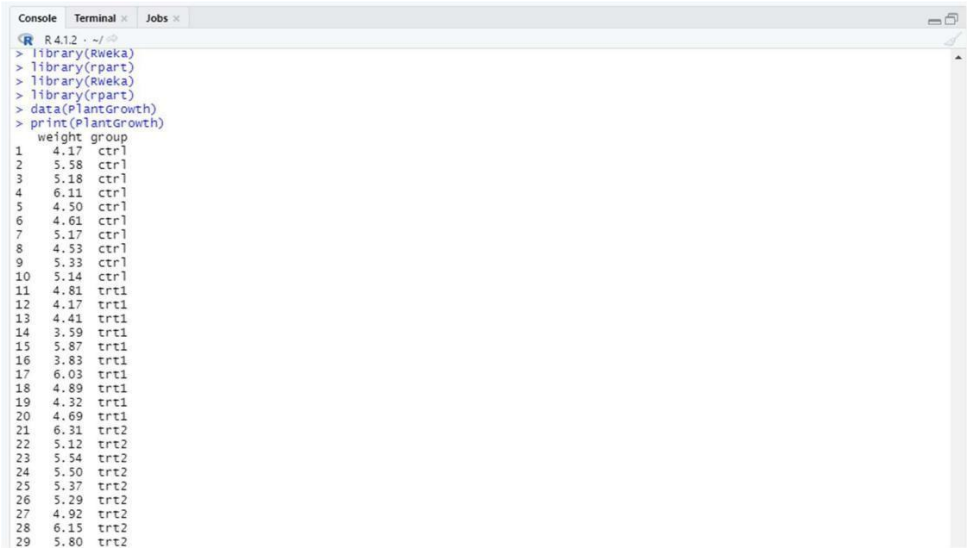
	weight	group
1	4.17	ctrl
2	5.58	ctrl
3	5.18	ctrl
4	6.11	ctrl
5	4.50	ctrl
6	4.61	ctrl
7	5.17	ctrl
8	4.53	ctrl
9	5.33	ctrl
10	5.14	ctrl
11	4.81	trt1
12	4.17	trt1
13	4.41	trt1
14	3.59	trt1
15	5.87	trt1
16	3.83	trt1
17	6.03	trt1
18	4.89	trt1
19	4.32	trt1
20	4.69	trt1
21	6.31	trt2
22	5.12	trt2
23	5.54	trt2
24	5.50	trt2
25	5.37	trt2
26	5.29	trt2
27	4.92	trt2
28	6.15	trt2
29	5.80	trt2

6. Code and Output:

```
install.packages("rpart")

library(RWeka)
library(rpart)

data(PlantGrowth)
print(PlantGrowth)
fit <- rpart(group~., data=PlantGrowth)
```

```
summary(fit)

predictions <- predict(fit, PlantGrowth[,1:2], type="class")
table(predictions, PlantGrowth$group)
fit <- J48(group~., data=PlantGrowth)
summary(fit)

predictions <- predict(fit, PlantGrowth[,1:2])

table(predictions, PlantGrowth$Species)



fit <- PART(group~., data=PlantGrowth)
summary(fit)
predictions <- predict(fit, PlantGrowth[,1:2])
table(predictions, PlantGrowth$group)
```

OUTPUT:



```
Console  Terminal  Jobs
R 4.1.2 · ~/
> library(RWeka)
> library(rpart)
> library(RWeka)
> library(rpart)
> data(PlantGrowth)
> print(PlantGrowth)
   weight group
1    4.17  ctrl
2    5.58  ctrl
3    5.18  ctrl
4    6.11  ctrl
5    4.50  ctrl
6    4.61  ctrl
7    5.17  ctrl
8    4.53  ctrl
9    5.33  ctrl
10   5.14  ctrl
11   4.81  trt1
12   4.17  trt1
13   4.41  trt1
14   3.59  trt1
15   5.87  trt1
16   3.83  trt1
17   6.03  trt1
18   4.89  trt1
19   4.32  trt1
20   4.69  trt1
21   6.31  trt2
22   5.12  trt2
23   5.54  trt2
24   5.50  trt2
25   5.37  trt2
26   5.29  trt2
27   4.92  trt2
28   6.15  trt2
29   5.80  trt2
```

```
Console  Terminal ×  Jobs ×
R 4.1.2 · ~/
  n= 30

    CP nsplit rel error xerror       xstd
1 0.40      0       1.0   1.35 0.08215838
2 0.01      1       0.6   0.85 0.13570802

Variable importance
weight
   100

Node number 1: 30 observations,    complexity param=0.4
  predicted class=ctrl  expected loss=0.6666667  P(node) =1
    class counts:    10    10    10
   probabilities: 0.333 0.333 0.333
  left son=2 (12 obs) right son=3 (18 obs)
  Primary splits:
      weight < 4.905 to the left,  improve=4.444444, (0 missing)

Node number 2: 12 observations
  predicted class=trt1  expected loss=0.3333333  P(node) =0.4
    class counts:     4     8     0
   probabilities: 0.333 0.667 0.000

Node number 3: 18 observations
  predicted class=trt2  expected loss=0.4444444  P(node) =0.6
    class counts:     6     2    10
   probabilities: 0.333 0.111 0.556
```

```
> predictions <- predict(fit, PlantGrowth[,1:2], type="class")
> table(predictions, PlantGrowth$group)

predictions ctrl trt1 trt2
       ctrl    0    0    0
       trt1    4    8    0
       trt2    6    2   10
> fit <- J48(group~., data=PlantGrowth)
> summary(fit)

=== Summary ===

Correctly Classified Instances          20               66.6667 %
Incorrectly Classified Instances        10               33.3333 %
Kappa statistic                          0.5
Mean absolute error                      0.2849
Root mean squared error                  0.3774
Relative absolute error                 64.1026 %
Root relative squared error             80.0641 %
Total Number of Instances               30

=== Confusion Matrix ===

   a  b  c   <-- classified as
   0  4  6 |  a = ctrl
   0 10  0 |  b = trt1
   0  0 10 |  c = trt2
```

Learning outcomes (What I have learnt):

1. Decision tree algorithm falls under the category of supervised learning.
2. 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.
3. A confusion matrix is a table that is used to define the performance of a classification algorithm