EXPERIMENT NO: 3

Aim: To Demonstrate performing classification on data sets.

Theory:

Classification:

Classification is the process for finding a model that describes the data values and concepts for the purpose of Prediction.

Classification in data mining is a common technique that separates data points into different classes. It allows you to organize data sets of all sorts, including complex and large datasets as well as small and simple ones. It primarily involves using algorithms that you can easily modify to improve the data quality.The algorithm establishes the link between the variables for prediction. The algorithm you use for classification in data mining is called the classifier, and observations you make through the same are called the instances.

There are multiple types of classification algorithms, each with its unique functionality and application. All of those algorithms are used to extract data from a dataset.

Data Mining Algorithms for Classification:

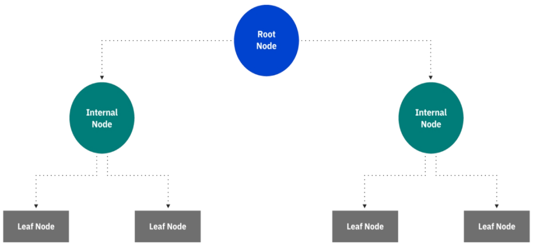
- **Decision Trees**
- Logistic Regression
- Naive Bayes Classification
- k-nearest neighbors
- Support Vector Machine

Decision Tree:

A decision tree is a structure that includes a root node, branches, and leaf nodes. Each internal node denotes a test on an attribute, each branch denotes the outcome of a test, and each leaf node holds a class label. The topmost node in the tree is the root node.

A decision Tree is a classification scheme to generate a tree consisting of root node, internal nodes and external nodes.Root nodes representing the attributes. Internal nodes are also the attributes. External nodes are the classes and each branch represents the values of the attributes.

Decision Tree also contains set of rules for a given data set; there are two subsets in Decision Tree. One is a Training data set and second one is a Testing data set. Training data set is previously classified data. Testing data set is newly generated data.



The benefits of having a decision tree are as follows −

It does not require any domain knowledge.

It is easy to comprehend.

The learning and classification steps of a decision tree are simple and fast.

J48 Classification and its Decision Tree

C4.5 algorithm/J48

The C4.5 algorithm is a classification algorithm which produces decision trees based on information theory. It is an extension of Ross Quinlan’s earlier ID3 algorithm also known in Weka as J48, J standing for Java. The decision trees generated by C4.5 are used for classification, and for this reason, C4.5 is often referred

to as a statistical classifier.
The J48 implementation of the C4.5 algorithm has many additional features including accounting for missing values, decision trees pruning, continuous attribute value ranges, derivation of rules, etc.

In the WEKA data mining tool, J48 is an open-source Java implementation of the C4.5 algorithm. J48 allows classification via either decision trees or rules generated from them.

Generating a decision tree form training tuples of data partition D
Algorithm : Generate_decision_tree
Input:

Data partition, D, which is a set of training tuples
and their associated class labels.
attribute_list, the set of candidate attributes.
Attribute selection method, a procedure to determine the
splitting criterion that best partitions that the data
tuples into individual classes. This criterion includes a
splitting_attribute and either a splitting point or splitting subset.

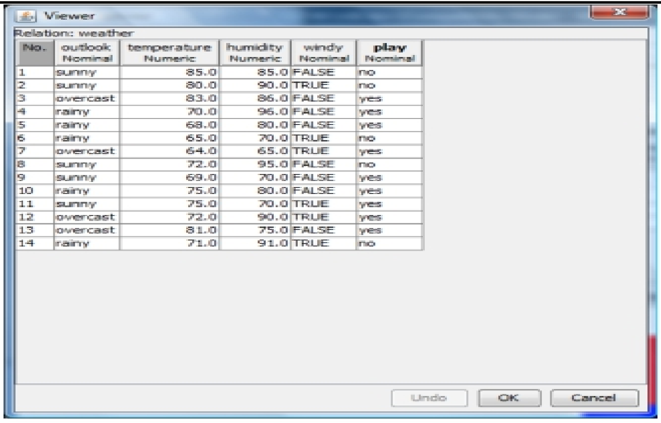
Output:

A Decision Tree

Method:

```
create a node N;
if tuples in D are all of the same class, C then
    return N as leaf node labeled with class C;

  if attribute_list is empty then
    return N as leaf node with labeled
    with majority class in D;|| majority voting

apply attribute_selection_method(D, attribute_list)
to find the best splitting_criterion;
label node N with splitting_criterion;

if splitting_attribute is discrete-valued and
    multiway splits allowed then    // no restricted to binary trees

attribute_list = splitting attribute; // remove splitting attribute
for each outcome j of splitting criterion

    // partition the tuples and grow subtrees for each partition
    let Dj be the set of data tuples in D satisfying outcome j; // a partition

    if Dj is empty then
        attach a leaf labeled with the majority
        class in D to node N;
    else
        attach the node returned by Generate
        decision tree(Dj, attribute list) to node N;
    end for
return N;
```

Training Data Set Weather Table



Viewer

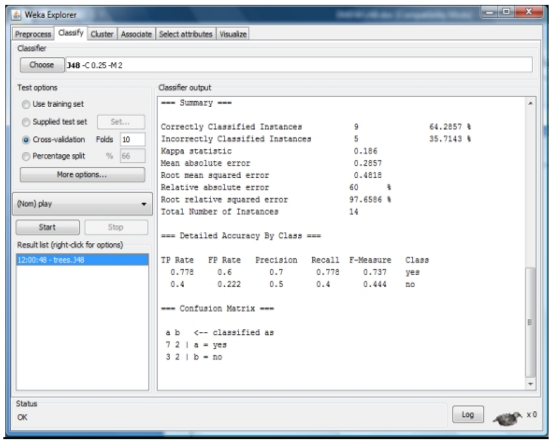
Relation: weather

No.	outlook Nominal	temperature Numeric	humidity Numeric	windy Nominal	play Nominal
1	sunny	85.0	85.0	FALSE	no
2	sunny	80.0	90.0	TRUE	no
3	overcast	83.0	86.0	FALSE	yes
4	rainy	70.0	96.0	FALSE	yes
5	rainy	68.0	80.0	FALSE	yes
6	rainy	65.0	70.0	TRUE	no
7	overcast	64.0	65.0	TRUE	yes
8	sunny	72.0	95.0	FALSE	no
9	sunny	69.0	70.0	FALSE	yes
10	rainy	75.0	80.0	FALSE	yes
11	sunny	75.0	70.0	TRUE	yes
12	overcast	72.0	90.0	TRUE	yes
13	overcast	81.0	75.0	FALSE	yes
14	rainy	71.0	91.0	TRUE	no

Procedure for Decision Trees:

1) Open Start Programs Weka-3-4 Weka-3-4
2) Open explorer.
3) Click on open file and select weather.arff
4) Select Classifier option on the top of the Menu bar.
5) Select Choose button and click on Tree option.
6) Click on J48.
7) Click on Start button and output will be displayed on the right side of the window.
8) Select the result list and right click on result list and select Visualize Tree option.
9) Then Decision Tree will be displayed on new window

Output:



Weka Explorer

Classifier: J48 -C 0.25 -M 2

Test options: ● Cross-validation Folds 10

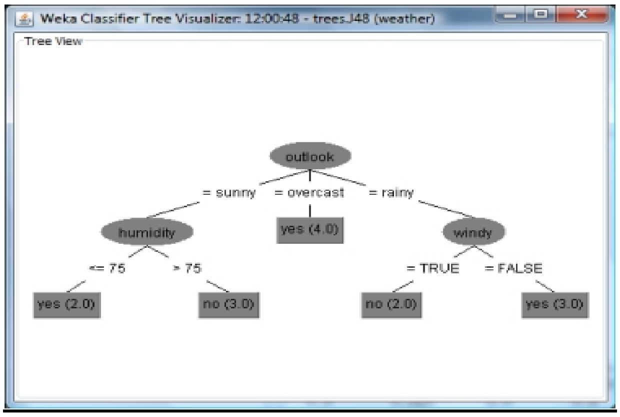
(Nom) play

Result list (right-click for options): 12:00:48 - trees.J48

Classifier output

```
=== Summary ===

Correctly Classified Instances           9               64.2857 %
Incorrectly Classified Instances         5               35.7143 %
Kappa statistic                          0.186
Mean absolute error                      0.2857
Root mean squared error                  0.4818
Relative absolute error                 60      %
Root relative squared error             97.6586 %
Total Number of Instances               14

=== Detailed Accuracy By Class ===

TP Rate   FP Rate   Precision   Recall  F-Measure   Class
  0.778     0.6        0.7       0.778    0.737    yes
  0.4       0.222      0.5       0.4      0.444    no

=== Confusion Matrix ===

 a b   <-- classified as
 7 2 | a = yes
 3 2 | b = no
```

Status: OK



Result: Thus the classification on Data set is performed by decision tree (J48) Method.

Viva Questions.

Q.1) What is Classification?

Q.2)What is the need of classification?

Q.3) What are the different methods of classification?

Q.4)What are the advantages of a decision tree classifier?

Decision Tree:

Decision Tree:

Result: This program has been successfully executed.

Viva Questions.

Q.1) What are the advantages of a decision tree classifier?
Q.2) What is the need of classification?