Bharathidasan Institute of Technology

Anna University, Tiruchirappalli – 620 024

Hand-out for
Hands-on Training session on

Knowledge Discovery in Databases with WEKA

<u>Prepared and handled by</u>

Resource Person
Dr.D.Asir Antony Gnana Singh
Dept. of CSE
Bharathidasan Institute of Technology
Anna University, Tiruchirappalli – 620 024

Knowledge Discovery in Databases with WEKA

Working with WEKA Explorer

Task 1: Prediction/decision making using classification algorithm

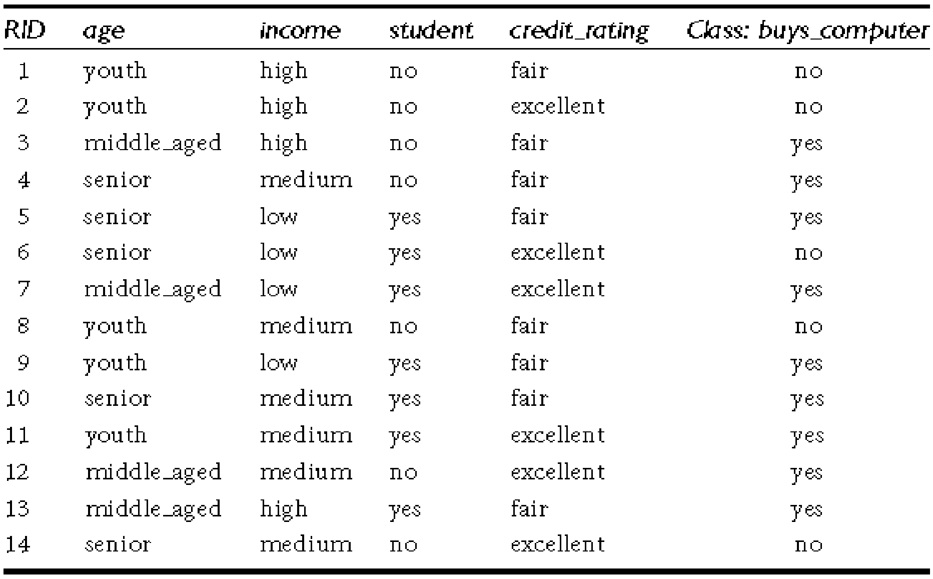
Question 1: Predict the following unknown data/class label from the training data using tree based J48 classifier (Predict the following four customers who will buy the computer and who will not buy the computer from the historical data)

Unknown /unlabeled data:

Age	**Income**	**Student**	**Credit Rating**	**Class (Buys Computer: Yes/No)**
Youth	High	No	Fair	
Youth	High	No	Excellent	
Middle aged	High	No	Fair	
Senior	Medium	No	Fair	

Historical data / Training data:

Class-labeled training tuples from the *AllElectronics* customer database.



RID	*age*	*income*	*student*	*credit_rating*	*Class: buys_computer*
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

Solution:

Step 1: Prepare the training dataset in arff (Attribute-Relation File Format) from the training data.

```
@relation  My_First_Training_Dataset
@attribute age {youth,middle_aged,senior}
@attribute income {high,medium,low}
@attribute student {no,yes}
@attribute credit_rating {fair,excellent}
@attribute Class {no,yes}
@data
youth,high,no,fair,no
youth,high,no,excellent,no
```

```
middle_aged,high,no,fair,yes
senior,medium,no,fair,yes
senior,low,yes,fair,yes
senior,low,yes,excellent,no
middle_aged,low,yes,excellent,yes
youth,medium,no,fair,no
youth,low,yes,fair,yes
senior,medium,yes,fair,yes
youth,medium,yes,excellent,yes
middle_aged,medium,no,excellent,yes
middle_aged,high,yes,fair,yes
senior,medium,no,excellent,no
```

Step 2: Prepare the unknown/unlabeled dataset in arff (Attribute-Relation File Format) from the unknown/unlabeled data.

```
@relation  'My_First_Unknown_Dataset'
@attribute age {youth,middle_aged,senior}
@attribute income {high,medium,low}
@attribute student {no,yes}
@attribute credit_rating {fair,excellent}
@attribute Class {no,yes}
@data
youth,high,no,fair,?
youth,high,no,excellent,?
middle_aged,high,no,fair,?
senior,medium,no,fair,?
```

Step 3: Build the predictive model using tree based j48and save.
Step 4: Supply the unknown dataset as the test dataset.
Step 5: Choose more option and tick the output prediction option.
Step 6: Load the model and reevaluate the model on current test set.

Question 2: Visualize the decision tree of the J48 algorithm and draw the decision tree of My_First_Training_Dataset.

Knowledge Discovery in Databases with WEKA

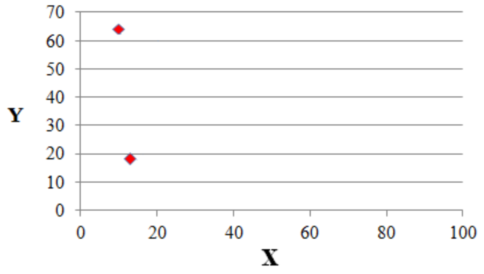
Task 2: Prediction/decision making using clustering algorithm

Question -1: Predict the following unknown data from the training data using simple K-mean clustering algorithm (predict the cluster for the given two objects)

Unknown data:

Obj. ID	X	Y	Cluster
1	13	18	
2	10	64	

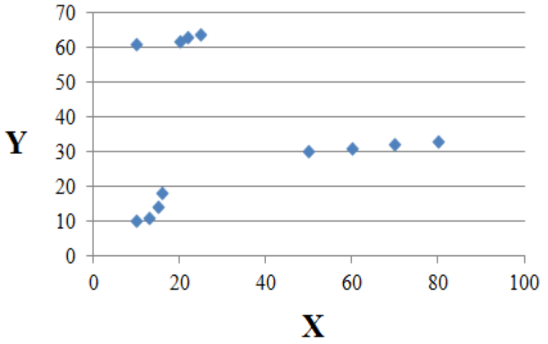
Scatter plot of unknown data



Training data:

Obj. ID	X	Y
1	10	61
2	20	62
3	22	63
4	25	64
5	50	30
6	60	31
7	70	32
8	80	33
9	10	10
10	13	11
11	15	14
12	16	18

Scatter plot of training data



Solution:

Step 1: Prepare the training dataset in arff (Attribute-Relation File Format) from the training data.

```
@relation 'My_Second_Training_Dataset'
@attribute x numeric
@attribute y numeric
@data
10,61
20,62
22,63
25,64
50,30
60,31
70,32
80,33
10,10
13,11
15,14
```

```
16,18
```

Step 2: Prepare the unknown dataset in arff (Attribute-Relation File Format) from the unknown data.

```
@relation 'My_Second_Test_Dataset'
@attribute x numeric
@attribute y numeric
@data
13,18
10,64
```

Step 3: Build the predictive model using simple K-means algorithm.

Step 4: Supply the unknown dataset as the test dataset.

Step 5: Choose more option and tick the output prediction option.

Step 6: Load the model and reevaluate the model on current test set.

Question 2: Visualize the cluster assignment for 'My_Second_Training_Dataset.arff using WEKA and fill-up the cluster number.

Obj. ID	X	Y	Cluster
1	10	61	
2	20	62	
3	22	63	
4	25	64	
5	50	30	
6	60	31	
7	70	32	
8	80	33	
9	10	10	
10	13	11	
11	15	14	
12	16	18	

Task 3: Finding the accuracy of the classification algorithms and sum of the squared errors (SSE) of the clustering algorithms for various dataset

Question 1: Find the **accuracy** and time **taken to build the model** for the dataset **segment challenge** and encircle the **highly accurate** test option mode.

Name of the Dataset	Mode of Test option	Accuracy	Time to build model
Segment-challenge	Use train set as the test dataset		
Segment-challenge	Supply **Segment-test dataset**		
Segment-challenge	Cross Validation Folds =10		
Segment-	Percentage Split =		

challenge	66%		

Question 2: Find the classifier which produces higher **accuracy** for the dataset diabetes with **Naive Bayes NB, tree based J48, rule based IBI** classifiers with the high mode of test option **Cross Validation Folds =10.**

Name of the Dataset	Accuracy NB	Accuracy J48	Accuracy IBI
Diabetes			

Question 3: Find the **sum of squared errors (SSE)** and **time taken to build the model** for the dataset **segment-challenge** with the following specifications and encircle the **lowest SSE** cluster mode

Name of the Dataset	Cluster Mode	Distance Function	Number cluster	SSE	Time to build model
Segment-challenge	Use train set as the test dataset	Euclidian	3		
Segment-challenge	Supply **Segment-test dataset**	Euclidian	3		
Segment-challenge	Percentage Split = 66%	Euclidian	3		

Name of the Dataset	Cluster Mode	Distance Function	Number cluster	SSE	Time to build model
Segment-challenge	Use train set as the test dataset	Manhattan	2		
Segment-challenge	Supply **Segment-test dataset**	Manhattan	2		
Segment-challenge	Percentage Split = 66%	Manhattan	2		

Part-II

Dimensionality Reduction using Attribute Selection Algorithms

Feature/ Variable/ Attribute Selection

1. Attribute Subset Evaluation
2. Attribute Raking (Attribute Evaluation)

Classification algorithms:

- J48
- IB1
- Naive Bayes (NB)

Attribution Selection algorithms:

- CfsSubset Eval (CFS)
- InfoGainAttributeEval(IG)
- ChisquaredAttributeEval (CQ)

Formula for obtaining the Threshold value for feature ranker:

$$Tv = \frac{Max - Min}{2} + Min$$

Where

T_v	–	Threshold value
Min	–	Minimum rank value
Max	–	Maximum rank value

Question 1: Obtain the number of **instances, attributes and classes** for the following datasets using WEKA.

S. No.	Dataset	Instances	Attribute	Classes
1	Diabetes			
2	Vote			
3	Weather.numeric			
4	Car			

Question 2: Obtain the classification accuracy for **NB, J48 and IB1** classifiers for the following datasets using the following Attribute selection algorithms.

S. No.	Dataset	Accuracy of NB			Accuracy of J48			Accuracy of IB1		
		CFS	***IG***	***GR***	***CFS***	***IG***	***GR***	***CFS***	***IG***	***GR***
1	Diabetes									
2	Vote									
3	Weather.Numeric									
4	Car									
Average Accuracy										

Question 3: Plot the average classification accuracy from the below table with respect to classifiers.

	Accuracy of NB			Accuracy of J48			Accuracy of IB1		
	CFS	***IG***	***GR***	***CFS***	***IG***	***GR***	***CFS***	***IG***	***GR***
Average Accuracy									

Question 4: Plot the classification accuracy for dataset **Diabetes** with respect the various feature selection algorithms and classifier.

	Accuracy of NB			**Accuracy of J48**			**Accuracy of IB1**		
	CFS	***IG***	***GR***	***CFS***	***IG***	***GR***	***CFS***	***IG***	***GR***
Diabetes									

Question 5: Which feature/attribute selection method is producing better classification accuracy for the given dataset?

Question 6: Write the conclusion from the above two plots.

Part-III

Outliers and Extreme Values detection using WEKA

Procedure:

1. Obtain the dataset **spambase**.
2. WEKA explorer – Preprocess-Filter – Choose- Unsupervised – Attribute – **InterquartileRange** – Apply – Save with the name 'My_First_Outlier_Detection'
3. View the outlier attribute and Extreme value attribute.
4. Remove the outlier objects/records/instances/tuples.
    a. WEKA explorer – Filter – Choose- Unsupervised –Instance–Remove With Values - Click on '**RemoveWithValues**'- set the attribute index of the outlier attribute – set the nominal indices as "last" – ok- click on apply button
    b. WEKA explorer – Filter – Choose- Unsupervised –Instance–Remove With Values - Click on '**RemoveWithValues**'- set the attribute index of the

Extreme Value attribute – set the nominal indices as “last” – ok- click on apply button

Question 1: Find the number of outliers and extreme value instances and plot the same.

Dataset	Number of outlier instances	Number of extreme value instances
Spambase		
Diabetes		

Question 2: Find the **sum of squared errors** (SSE) and **time to build model** (TBM) for the following dataset using simple K-mean clustering algorithm for various number of clusters.

Dataset	Spambase			
No. of clusters	TBM Without removal of outlier and extreme value instances	TBM With removal of outlier and extreme value instances	TBM Without removal of outlier and extreme value instances	TBM With removal of outlier and extreme value instances
2				
3				
4				
5				

Dataset	Diabetes			
No. of clusters	TBM Without removal of outlier and extreme value instances	TBM With removal of outlier and extreme value instances	TBM Without removal of outlier and extreme value instances	TBM With removal of outlier and extreme value instances
2				
3				
4				
5				

Question 3: Plot the comparison of **SSE** with and without removal of outlier and extreme value instances for the dataset **spambase** for various clusters using simple K-mean clustering algorithm.

	2	3	4	5
SSE With removal of outlier and extreme value instances				
SSE Without removal of outlier and extreme value instances				

Part-IV

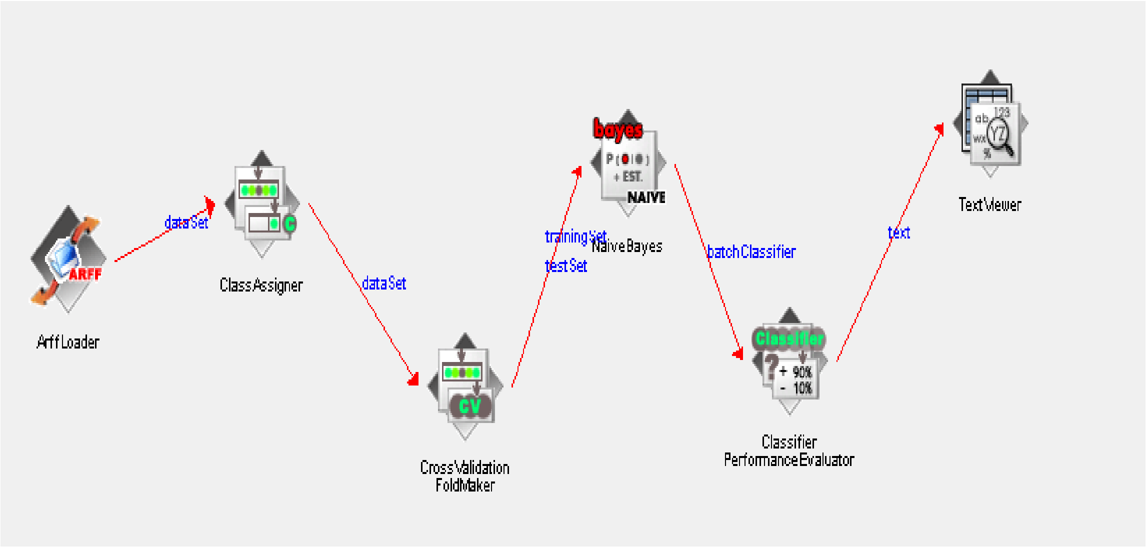
Working with WEKA Knowledge Flow Environment

Question 1: Develop a knowledge flow to find the classification accuracy for the dataset **My_First_Dataset.arff** using tree based **J48** classifier.

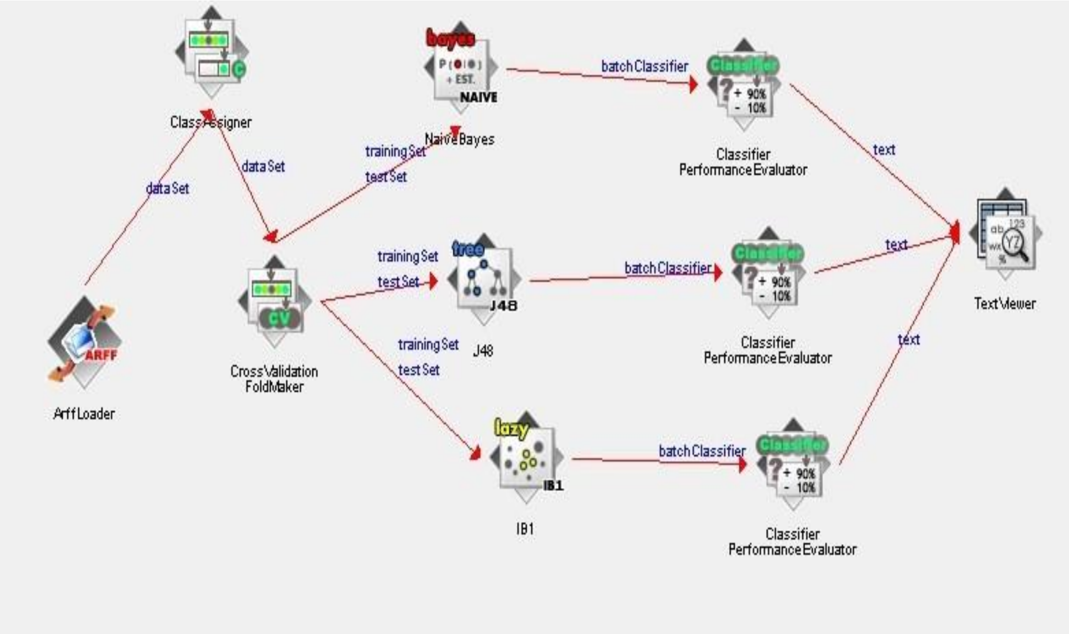
Steps:

- Go to Weka GUI chooser and select **Knowledge Flow**
- Go to Data Sources tab
  - Drag and drop the **Arff Loader** to Knowledge flow layout
  - Right click on the **Arff Loader** and choose the **configuration** and load the dataset **My_First_Dataset.arff**
- Go to Evaluation tab
  - Drag and drop the **Class Assigner** to Knowledge flow layout
  - Right click on the **Arff Loader** and choose the **dataset** and connect the **Arff Loader** and **Class Assigner** using rubber band connector
  - Right click on the **Class Assigner** and choose the **configuration** and choose the **class attribute**
- Go to Evaluation tab
  - Drag and drop the **Cross Validation Fold Maker** to Knowledge flow layout
  - Right click on the **Cross Validation Fold Maker** and choose the **configuration** to set the number of fold (default value is 10)
- Go to Classifier tab
  - Drag and drop **J48 classifier** to Knowledge flow layout
  - Right click on the **Class Assigner** and choose the **dataset** and connect with the **J48 classifier** using **rubber band connector**
  - Right click on the **Cross Validation Fold Maker** and choose the **training set** and connect with the **J48 classifier** using **rubber band connector**
  - Right click on the **Cross Validation Fold Maker** and choose the **test set** and connect with the **J48 classifier** using **rubber band connector**
- Go to the Evaluation tab
  - Drag and drop the **classifier performance evaluator** to Knowledge flow layout
  - Right click on the **J48 classifier** and choose the **batch classifier** and connect the **Classifier Performance Evaluator** using **rubber band connector**
- Go to the Visualization tab

    - Drag and drop the **Text viewer** to Knowledge flow layout
    - Right click on the **Classifier Performance Evaluator** and choose the **text** to connect the classifier performance evaluator using rubber band connector
- To start/run the work flow
    - Right click on the **arff loader** and choose the **start loading**
- To view the result
    - Do right click on the **Text viewer** and choose the **show results**
- To save the knowledge flow layout in the image format for publishing article or books
    - Press Control +Alt + shift + left click
    - Save in your decried image format



Question 2 : Set a knowledge flow environment for compute the accuracy of the Naïve Bayes , tree based J48, IB1 classifiers for the dataset diabetes and identify which classifier gives better accuracy



Part-V

Working with WEKA Experimenter

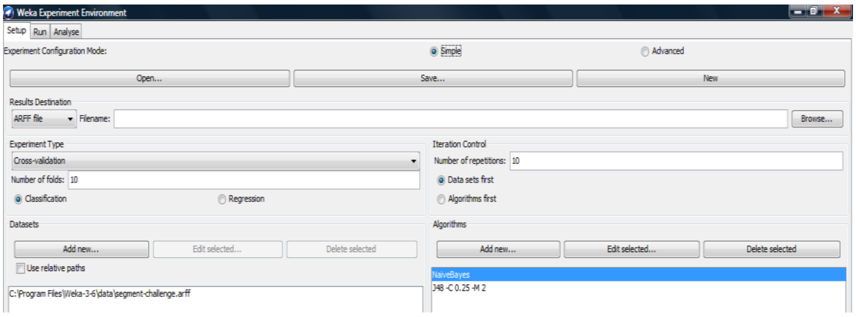
- Go to setup tab
    - Go to **dataset** section area and Click on the **Add new** button and open the secment challenge.arff
    - Go to **Algorithms** selection area and Click on the **Add New** button and click on **choose** button and expand the **tree folder** and choose **J48** classifier and click on **ok** button
    - Go to **Algorithms** selection area and Click on the **Add New** button and click on **choose** button and expand the **Bayes folder** and choose **Naïve Bayes** classifier and click on **ok** button
    - Go to **Experimenter type** tab and choose **cross validation** and set **number of folds** (Default value is 10).
    - Go to **Iteration control** tab and set **number of repetitions** (Default value is 10)
    - Go to **Run** tab and click on the **start** button
    - Go to **Analyses** tab and click on the **experiment** button
    - Go to configure **test option** set the Paired T-test correct-row (dataset)-column (schema algorithm)-comparison measure (F_measure)-significance (0.05)-sorting (default)-rest of them (default)
    - Click on **Perform test** button

Symbols interpretation: (*/ /V)

* - denotes loss, i.e. statistically significantly poor performance

V - denotes victory, i.e. statistically significantly better /good performance

Blank space - denotes significantly not better and not poor performance



Experiment Configuration Mode: ● Simple ○ Advanced

Results Destination: ARFF file Filename:

Experiment Type: Cross-validation

Number of folds: 10

● Classification ○ Regression

Iteration Control: Number of repetitions: 10

● Data sets first

○ Algorithms first

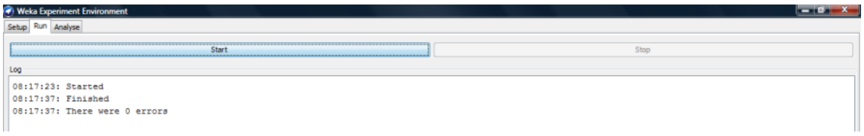
Datasets: □ Use relative paths

C:\Program Files\Weka-3-6\data\segment-challenge.arff

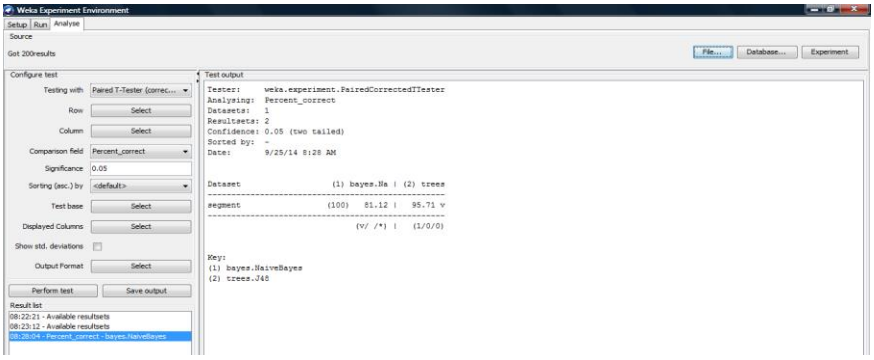
Algorithms:

NaiveBayes

J48 -C 0.25 -M 2



```
08:17:23: Started
08:17:37: Finished
08:17:37: There were 0 errors
```



Got 200results

Testing with: Paired T-Tester (correc...

Comparison field: Percent_correct

Significance: 0.05

Sorting (asc.) by: <default>

Show std. deviations: □

Result list:

08:22:21 - Available resultsets

08:23:12 - Available resultsets

08:28:04 - Percent_correct - bayes.NaiveBayes

```
Tester:     weka.experiment.PairedCorrectedTTester
Analysing:  Percent_correct
Datasets:   1
Resultsets: 2
Confidence: 0.05 (two tailed)
Sorted by:  -
Date:       9/25/14 8:28 AM


Dataset                   (1) bayes.Na | (2) trees
---------------------------------------------------
segment                  (100)   81.12 |   95.71 v
---------------------------------------------------
                               (v/ /*) |   (1/0/0)


Key:
(1) bayes.NaiveBayes
(2) trees.J48
```

Question 1: Which classifier is yielding significantly better result on the Paired T-test statistical analysis for the given dataset?

Gaining knowledge,
is the first step to wisdom.
Sharing it,
is the first step to humanity.