

**CS699 Data Mining
Fall 2020**

Classification of Assessment of Trump Administration's response to the COVID-19 outbreak

Jaehack Jeong

Table of Contents

<u>Statement of Data Mining Goal -----</u>	3
<u>Detailed Description of the Dataset -----</u>	3
<u>Attribute Selection and Algorithms for the Project -----</u>	5
<u>Data Mining Procedure -----</u>	5
<u>Preprocessing -----</u>	5
<u>Splitting Dataset -----</u>	6
<u>Attribute Selection Method -----</u>	8
<u>Data Mining Result and Evaluation -----</u>	12
<u>Performance Comparison -----</u>	63
<u>Discussion and Conclusion -----</u>	70

Statement of Data Mining Goal

The purpose of the project is developing successful classifier models that forecast whether or not the survey participants approve of the Trump administration's response to the outbreak of COVID-19 based on the background information of the participants and their answers to the survey questions.

The survey was conducted by ABC News in May 2020, which is the time that had been only a few months since the outbreak of virus. At that time, most of the states were experiencing the economic shut-down and chaos due to the lack of information of COVID-19.

Detailed Description of the Dataset

Title: ABC News/Ipsos Poll: 2020 Coronavirus Wave 8

<https://ropercenter.cornell.edu/ipoll/study/31117363>

The survey asks 6 questions about the Trump Administration's response to the COVID-19 outbreak along with its impacts, which are presented as below.

- **Q1: Do you approve or disapprove of the way Donald Trump is handling the response to the coronavirus? (Class Attribute)**
 - Approve or Disapprove
- **Q2: How concerned are you that you or someone you know will be infected with the coronavirus?**
 - Very concerned, Somewhat concerned, Not so concerned, Not concerned at all
- **Q3: Which of the following statements comes closest to your point of view?**
 - Opening the country now is worth it because it will keep economic damage to a minimum vs. Opening the country now is not worth it because it will mean more lives being lost
- **Q4: If a safe and effective coronavirus vaccine is developed, how likely would you be to get vaccinated?**
 - Very likely, Somewhat likely, Not so likely, Not likely at all
- **Q5a: Work Situation**
 - I am still working from my regular workplace, outside the home, Because of the coronavirus I am working from home instead of from my regular workplace, I normally work from home, I am not currently employed

- **Q5b: Whether lost job due to COVID-19 outbreak**
 - Yes or No

In addition to the questions, the dataset also contains background information of participants. The attributes are described as below:

1. id = participant's id
2. xspanish = participant's language (either English or Spanish)
3. complete_status = indicating whether participants fully completed the survey
4. ppage = age of participants
5. ppeduc = education of participants (from the 1st grade to doctorate degree)
6. ppeducat = education levels of participants (less than high school, high school, some college, bachelor's degree or higher)
7. ppgender = Gender of the participant (Male/Female)
8. ppethm = Race of the participant
9. pphhsize = household size of participants
10. pphouse = a brief description of house of participants
11. ppincimp = participant's income
12. ppmarit = marital status of participants
13. ppmsacat = indicating whether participants live in metro area or non-metro area
14. ppreg4 = region of participants
15. pprent = participants house rent
16. ppstaten = state where participants live
17. Q1
18. Q2
19. Q3
20. Q4
21. Q5a
22. Q5b
23. QPID = political party that participants support
24. POLIT200 = indicating whether participants are registered to vote at their current address
25. ABCAGE = The participants age range
26. Weights_pid = The study dataset(s) contain weight factors that should be employed in any data analysis. Typically, weights are used in an attempt to assure that the survey sample more accurately represents the population.

Attribute Selection and Algorithms for the Project

Attribute selection methods implemented in Weka:

1. [CfsSubsetEval](#) - Evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them. This evaluator gives us subsets of features that are highly correlated with the class (Q1) while having low intercorrelation.
2. [GainRatioAttributeEval](#) - Evaluates the worth of an attribute by measuring the gain ratio with respect to the class.
3. [InfoGainAttributeEval](#) - Evaluates the worth of an attribute by measuring the information gain with respect to the class.
4. [CorrelationAttributeEval](#) - Evaluates the worth of an attribute by measuring the correlation (Pearson's) between it and the class. Nominal attributes are considered on a value by value basis by treating each value as an indicator. An overall correlation for a nominal attribute is arrived at via a weighted average.

Classification algorithms used in this project:

1. [Naive Bayes](#)
2. [J48](#)
3. [RandomForest](#)
4. [MultilayerPerceptron](#)
5. [IBK](#)

Data Mining Procedure

Preprocessing

The original dataset is a csv file. In order to load the file in Weka, the following changes are needed.

- In the “ppeduc” column of the data set, there were two types of value “Bachelor’s” and “Master’s” that needed to be transformed into “Bachelors”, “Masters” respectively, since the apostrophe bothers it.
- In the “Q1” column, which is the class attribute, there are two ‘Skipped’ answers. We have removed the entire row corresponding to these values.
- We have also applied 3 filters to our data set

- 1) Filter -> NumericToNominal: to convert any categorical attribute that may appear as Numeric back to Nominal.
 - 2) Filter -> StringToNominal with attributeRange 9 (pphhsiz)
 - 3) Filter -> StringToNominal with attributeRange 25 (ABCAGE)
- Finally, we have removed three attributes from our list of attributes ID, weights_pid and complete_status. We did not need the IDs of the participants in our project, and Complete_Status had only one value therefore was non-effective.

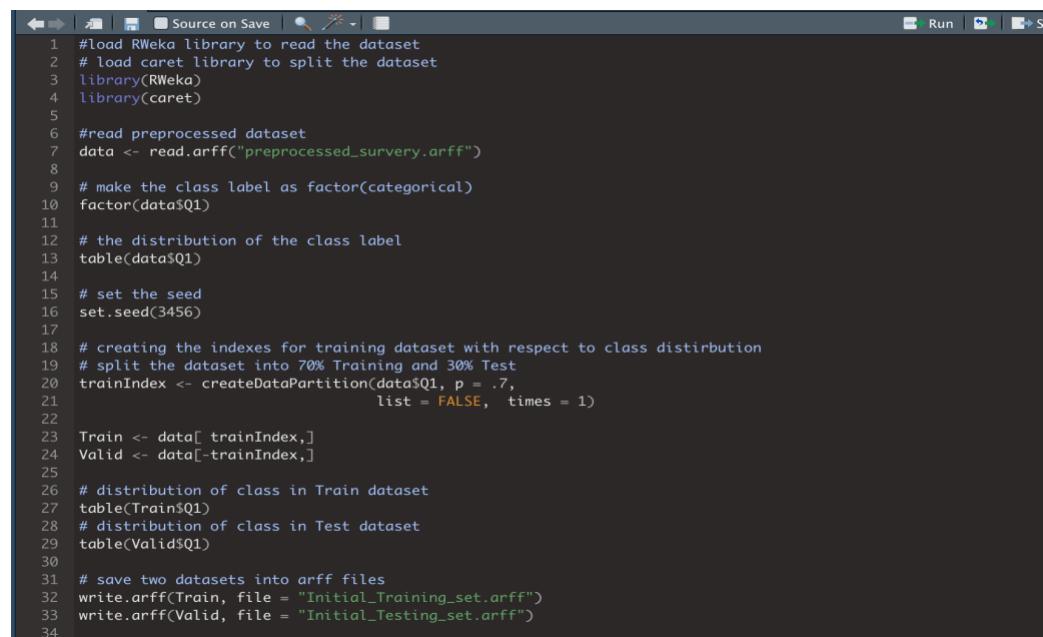
Splitting Dataset

After preprocessing the data set, we used R to split the dataset into the train and test datasets while remaining the distribution of the class attribute in the original dataset in order to prevent any imbalance issue.

The distribution of the class label in the original dataset

Q1	
Approved	Disapproved
237	293

Screenshot of the codes used for this process



```

1 #load RWeka library to read the dataset
2 # load caret library to split the dataset
3 library(RWeka)
4 library(caret)
5
6 #read preprocessed dataset
7 data <- read.arff("preprocessed_survery.arff")
8
9 # make the class label as factor(categorical)
10 factor(data$Q1)
11
12 # the distribution of the class label
13 table(data$Q1)
14
15 # set the seed
16 set.seed(3456)
17
18 # creating the indexes for training dataset with respect to class distribution
19 # split the dataset into 70% Training and 30% Test
20 trainIndex <- createDataPartition(data$Q1, p = .7,
21                                     list = FALSE, times = 1)
22
23 Train <- data[ trainIndex,]
24 Valid <- data[-trainIndex,]
25
26 # distribution of class in Train dataset
27 table(Train$Q1)
28 # distribution of class in Test dataset
29 table(Valid$Q1)
30
31 # save two datasets into arff files
32 write.arff(Train, file = "Initial_Training_set.arff")
33 write.arff(Valid, file = "Initial_Testing_set.arff")
34

```

The distribution of class label in the train dataset

Q1_Train	
Approved	Disapproved
166	206

The distribution of class label in the test dataset

Q1_Test	
Approved	Disapproved
71	87

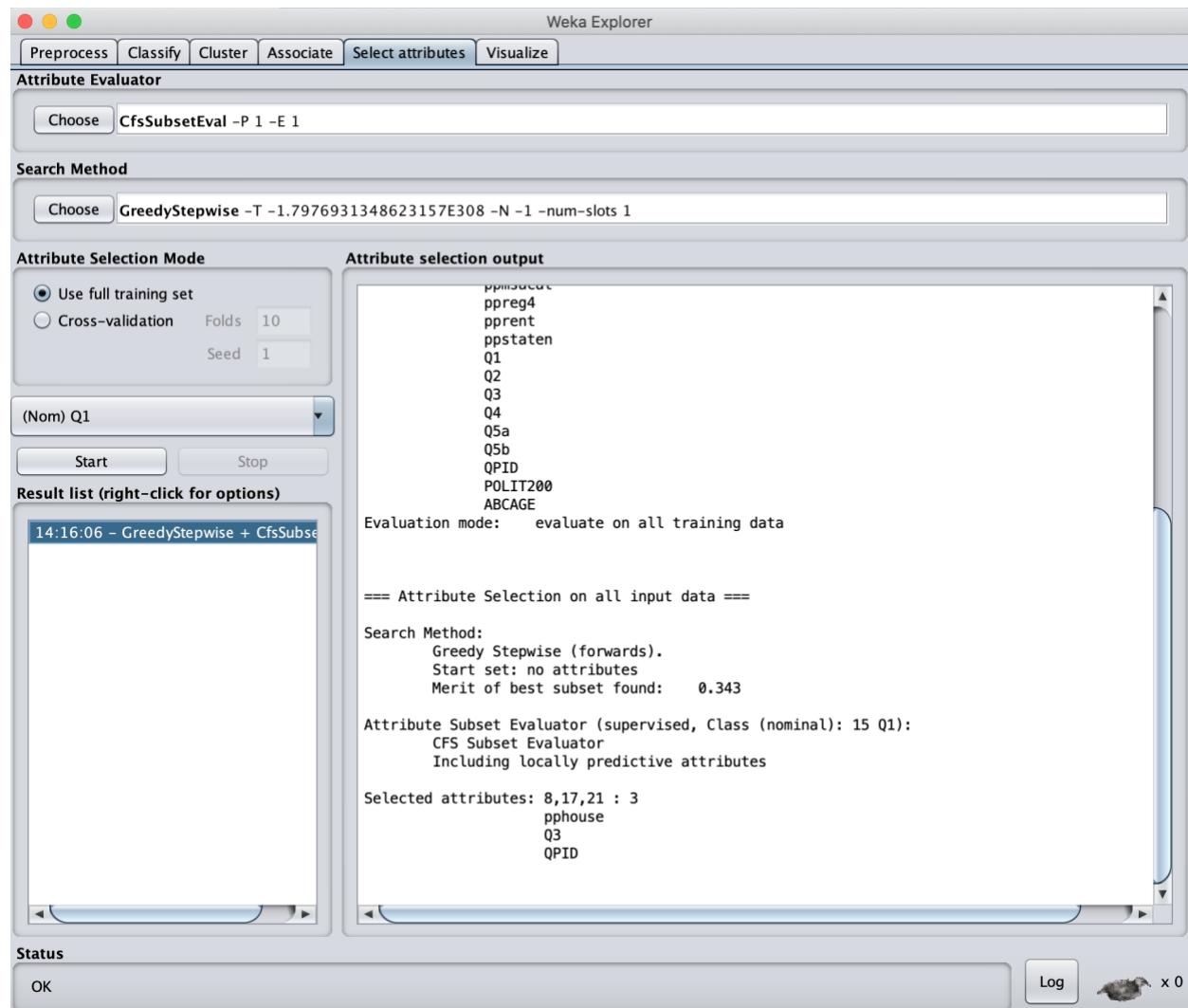
Attribute Selection Methods

With the newly created training and test dataset, we can begin with feature selection. We choose 4 attribute selection methods to select the best attributes from our 22 explanatory attributes.

[Attribute Evaluator #1: CfsSubsetEval](#)

[Search Method: GreedyStepwise](#)

[Attribute selection Mode: Use full training set](#)



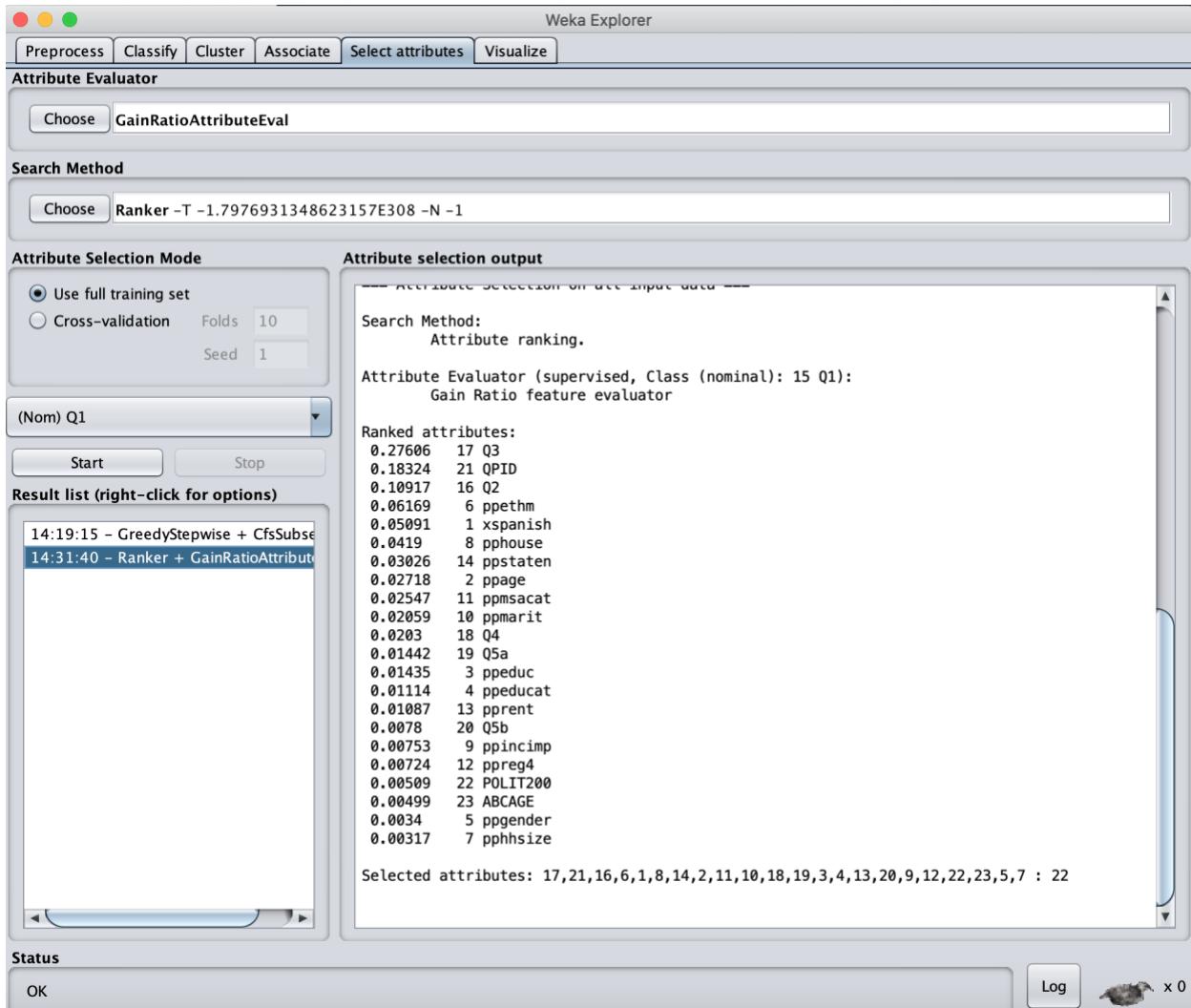
The result of this evaluator is this set of attributes: pphouse, Q3, QPID. For our first attribute sets we are using attributes selected by this method.

Attribute set 1: **[pphouse, Q3, QPID, Q1]**

Attribute Evaluator #2: GainRatioAttributeEval

Search Method: Ranker

Attribute selection Mode: Use full training set



The result returns the ranking of attributes based on their gain ratio (#1 Q3, #2 QPID, #3 Q2, #4 ppethm and so on)

We have chosen the attributes with higher Gain Ratio for our second attribute set. Q3, QPID, and Q2 have the highest gain ratio. After Q2, the gain ratio drops drastically so we exclude the rest of the attributes.

Attribute set 2: [Q3, QPID, Q2, Q1]

Attribute Evaluator #3: InfoGainAttributeEval

Search Method: Ranker

Attribute selection Mode: Use full training set

The screenshot shows the Weka Attribute Evaluator interface. At the top, there are tabs for Preprocess, Classify, Cluster, Associate, Select attributes, and Visualize. Below these are sections for 'Attribute Evaluator' (set to InfoGainAttributeEval), 'Search Method' (set to Ranker), 'Attribute Selection Mode' (set to 'Use full training set'), and a 'Result list' window. The 'Result list' window displays the following information:

14:40:57 – Ranker + InfoGainAttributeEval

Search Method:
Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 15 Q1):
Information Gain Ranking Filter

Ranked attributes:

Rank	Attribute
1	0.33779 21 QPID
2	0.28038 17 Q3
3	0.19235 16 Q2
4	0.16018 2 ppage
5	0.15056 14 ppstaten
6	0.08039 6 ppethm
7	0.05249 8 pphouse
8	0.04082 3 ppeduc
9	0.0355 10 ppmarit
10	0.03253 18 Q4
11	0.03047 9 ppincimp
12	0.02459 19 Q5a
13	0.02035 4 ppeducat
14	0.01504 11 ppmsacat
15	0.01425 12 ppreg4
16	0.0094 23 ABCAGE
17	0.00922 13 pprent
18	0.00763 1 xspanish
19	0.00707 7 pphsize
20	0.00455 20 Q5b
21	0.00338 5 ppgender
22	0.00267 22 POLIT200

Selected attributes: 21,17,16,2,14,6,8,3,10,18,9,19,4,11,12,23,13,1,7,20,5,22 : 22

Status: OK Log x 0

The result returns the ranking of attributes based on the information gain (#1 QPID, #2 Q3, #3 Q2, #4 ppage, #5 ppstaten and so on)

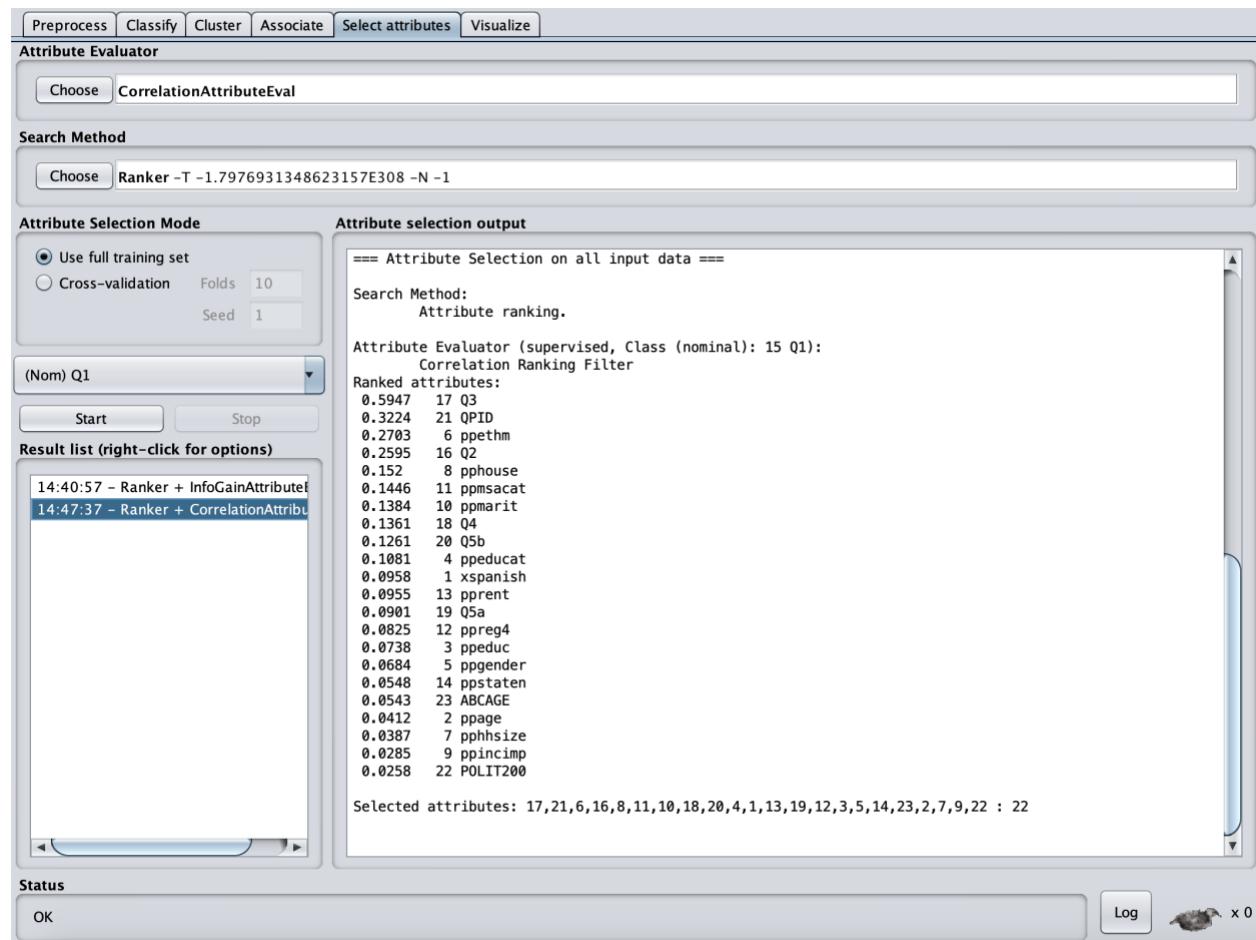
We have chosen the attributes with higher information gain for our third attribute set. QPID, Q3, Q2, ppage, and ppstaten have the highest information gain. After ppstaten, the information gain drops drastically so we exclude the rest of the attributes.

Attribute set 3: **[QPID, Q3, Q2, ppage, ppstaten, Q1]**

Attribute Evaluator #4: CorrelationAttributeEval

Search Method: Ranker

Attribute selection Mode: Use full training set



The result returns the ranking of attributes based on the correlation (#1 Q3, #2 QPID, #3 ppethm, #4 Q2 and so on)

As we can see the attributes with higher correlation to the class label are: Q3, QPID, ppethm, Q2, pphouse, ppmsacat, ppmarit, and Q4

Attribute set 4: [Q3, QPID, ppethm, Q2, pphouse, ppmsacat, ppmarit, Q4, Q1]

The final attribute set was picked by ourselves. We thought these attributes were interesting to perform our classification models on and compare the result with the other attribute sets selected by the methods shown above.

Attribute set 5: **[Q3, QPID, Q2, Q4, Q5a, Q5b, ppethm, ppmarit, ppeducat, ppgender, ABCAGE, Q1]**

Data Mining Result and Evaluation

- a. Attribute Set 1 [pphouse, Q3, QPID, Q1] - by the CfsSubsetEval
 1. Naive Bayes
 - Step 1: Open “Initial_Training_set.arff”, remove all attributes except pphouse, Q3, QPID, Q1 and save it as “set1_training”
 - Step 2: Open “Initial_Testing_set.arff”, remove all attributes except pphouse, Q3, QPID, Q1 and save it as “set1_testing”
 - Step 3: To assess how good Naive Bayse is, we used Cross-validation to build our model on our Train dataset. Open “set1_training”, Classify -> Choose NaiveBayes with Cross-validation (10 Folds), Q1 as class attribute -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose NaiveBayes

Test options

- Use training set
- Supplied test set Set...
- Cross-validation Folds 10
- Percentage split % 66

More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

15:35:41 - bayes.NaiveBayes

Classifier output

```

A Republican      91.0      12.0
Something else    13.0      13.0
A Democrat        9.0       110.0
Skipped           2.0       1.0
[total]            171.0     211.0

```

Time taken to build model: 0 seconds

==== Stratified cross-validation ====
==== Summary ====
Correctly Classified Instances 318 85.4839 %
Incorrectly Classified Instances 54 14.5161 %
Kappa statistic 0.7063
Mean absolute error 0.1432
Root mean squared error 0.2787
Relative absolute error 43.3113 %
Root relative squared error 68.6559 %
Total Number of Instances 372

==== Detailed Accuracy By Class ====

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
Approve	0.837	0.131	0.837	0.837	0.837	0.796	0.902	0.877	Approve
Disapprove	0.869	0.163	0.869	0.869	0.869	0.706	0.904	0.918	Disapprove
Skipped	?	0.000	?	?	?	?	?	?	Skipped
Weighted Avg.	0.855	0.149	0.855	0.855	0.855	0.706	0.903	0.900	

==== Confusion Matrix ====

			<-- classified as
a	b	c	a = Approve
139	27	0	a = Approve
27	179	0	b = Disapprove
0	0	0	c = Skipped

Status

OK Log x 0

- Step 4: In order to test our model, we re-evaluate our model with the test dataset. Select ‘Supplied test set’ and click ‘Set’ -> Open file, open “set1_testing” with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

```

==== Re-evaluation on test set ====
User supplied test set
Relation: R_data_frame-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst-last-weka.filters.unsupervised.
Instances: unknown (yet). Reading incrementally
Attributes: 4

==== Summary ====
Correctly Classified Instances      136          86.0759 %
Incorrectly Classified Instances   22           13.9241 %
Kappa statistic                   0.7194
Mean absolute error               0.1443
Root mean squared error          0.286
Total Number of Instances        158

==== Detailed Accuracy By Class ====

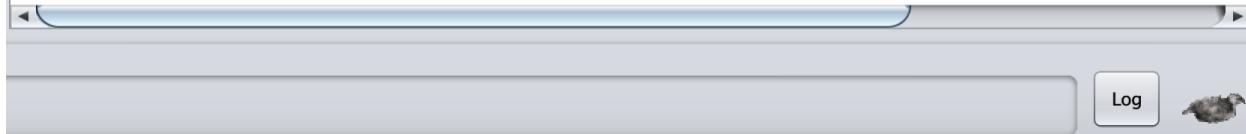
```

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.859	0.138	0.836	0.859	0.847	0.720	0.886	0.849	Approve	
0.862	0.141	0.882	0.862	0.872	0.720	0.887	0.877	Disapprove	
?	0.000	?	?	?	?	?	?	Skipped	
Weighted Avg.	0.861	0.140	0.861	0.861	0.861	0.720	0.886	0.864	

```

==== Confusion Matrix ====
a  b  c  <-- classified as
61 10  0 | a = Approve
12 75  0 | b = Disapprove
 0  0  0 | c = Skipped

```

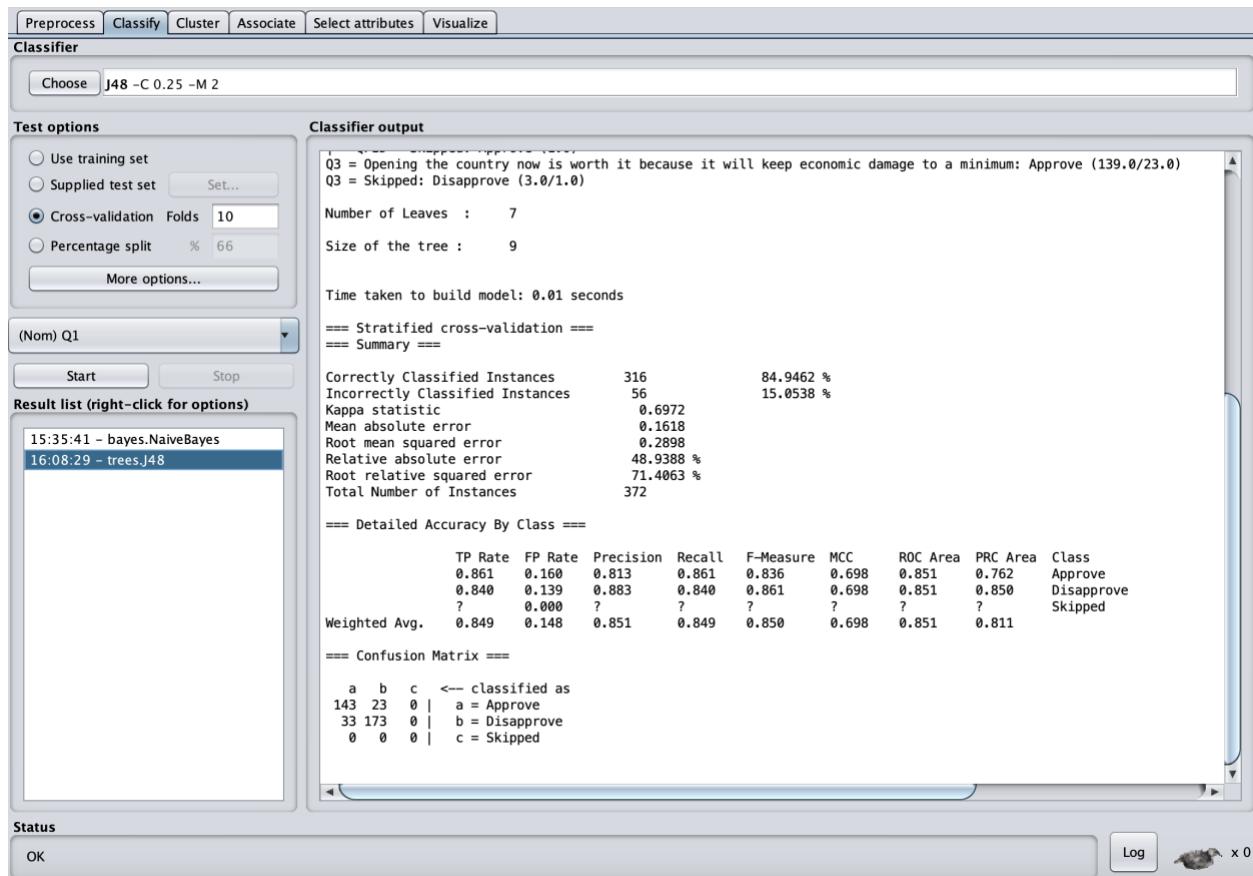


Naive Bayes - Cross Validation (Set 1)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
85.48%	83.7%	13.1%	90.2%	Approve
	86.9%	16.3%	90.4%	Disapprove

Naive Bayes - Test set (Set 1)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
86.08%	0.859	0.138	0.886	Approve
	0.862	0.141	0.887	Disapprove

2. J48

- Step 1: First to assess how good J48 is, use Cross-validation to build our model. Choose J48 and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set1_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

```

==== Re-evaluation on test set ====
User supplied test set
Relation: R_data_frame-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst-last-weka.filters.unsupervised.
Instances: unknown (yet). Reading incrementally
Attributes: 4

==== Summary ====
Correctly Classified Instances      134          84.8101 %
Incorrectly Classified Instances   24           15.1899 %
Kappa statistic                   0.6954
Mean absolute error               0.1636
Root mean squared error          0.2931
Total Number of Instances        158

==== Detailed Accuracy By Class ====

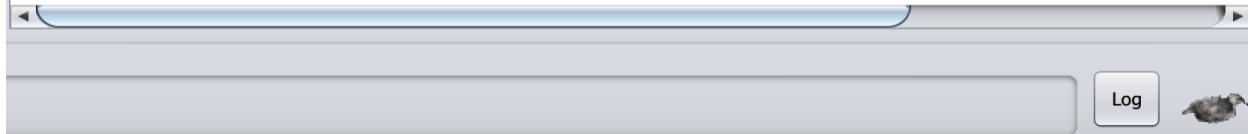
```

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.873	0.172	0.805	0.873	0.838	0.697	0.859	0.776	Approve	
0.828	0.127	0.889	0.828	0.857	0.697	0.859	0.848	Disapprove	
?	0.000	?	?	?	?	?	?	Skipped	
Weighted Avg.	0.848	0.147	0.851	0.848	0.848	0.697	0.859	0.816	

```

==== Confusion Matrix ====
a  b  c  <-- classified as
62 9  0  |  a = Approve
15 72 0  |  b = Disapprove
 0 0 0  |  c = Skipped

```

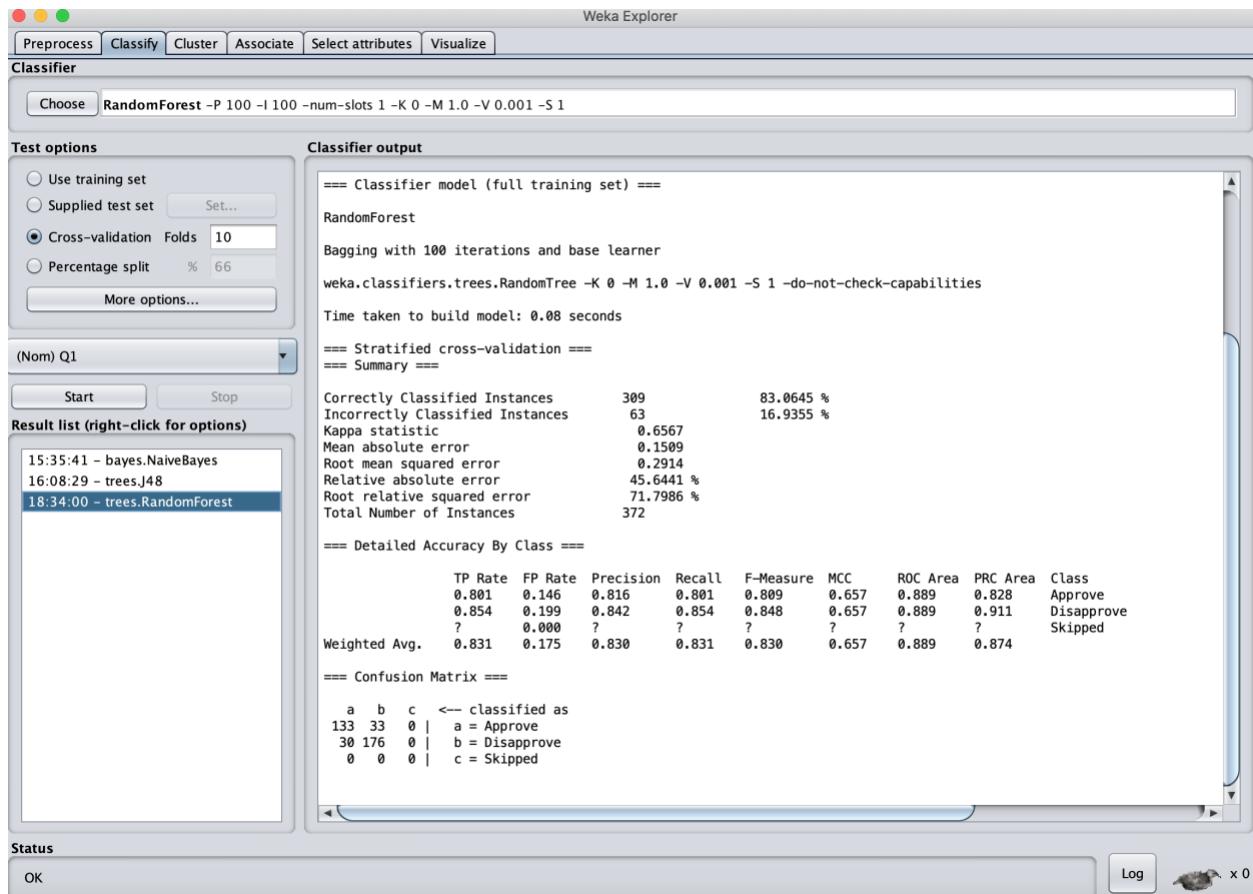


J48 - Cross Validation (Set 1)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
84.95%	86.1%	16.0%	85.1%	Approve
	84.0%	13.9%	85.1%	Disapprove

J48 - Test set (Set 1)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
84.81%	87.3%	17.2%	85.9%	Approve
	82.8%	12.7%	85.9%	Disapprove

3. RandomForest

- Step 1: Choose RandomForest and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set1_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

```

==== Re-evaluation on test set ====

User supplied test set
Relation: R_data_frame-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst-last-weka.filters.unsupervised.
Instances: unknown (yet). Reading incrementally
Attributes: 4

==== Summary ====

Correctly Classified Instances      134          84.8101 %
Incorrectly Classified Instances   24           15.1899 %
Kappa statistic                   0.6954
Mean absolute error               0.1466
Root mean squared error          0.2884
Total Number of Instances        158

==== Detailed Accuracy By Class ====

      TP Rate   FP Rate   Precision   Recall   F-Measure   MCC   ROC Area   PRC Area   Class
      0.873     0.172     0.805      0.873     0.838     0.697   0.883     0.852     Approve
      0.828     0.127     0.889      0.828     0.857     0.697   0.883     0.861     Disapprove
      ?          0.000     ?          ?          ?          ?          ?          ?          Skipped
Weighted Avg.   0.848     0.147     0.851      0.848     0.848     0.697   0.883     0.857

==== Confusion Matrix ====

  a  b  c  <-- classified as
62  9  0 |  a = Approve
15 72  0 |  b = Disapprove
  0  0  0 |  c = Skipped

```

RandomForest - Cross Validation (Set 1)

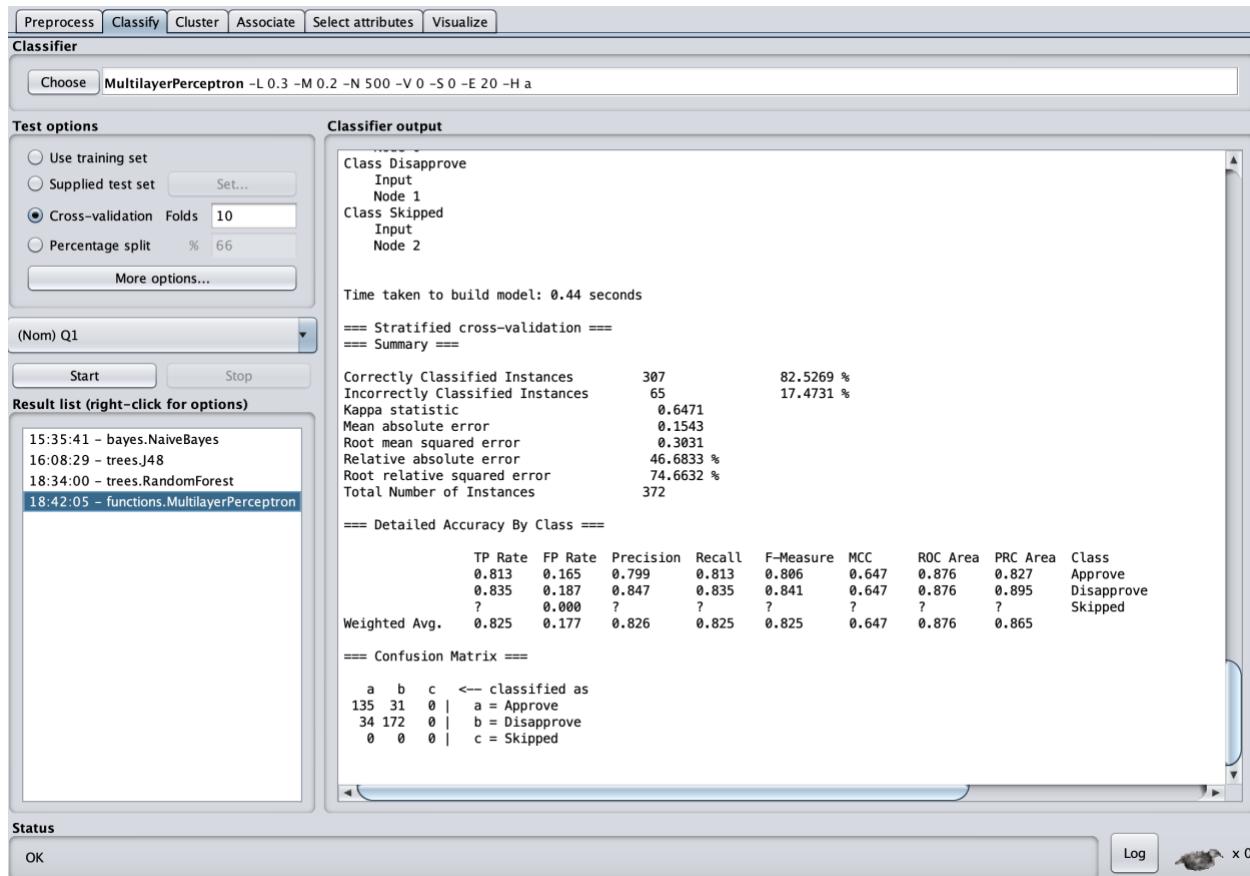
Accuracy	TP Rate	FP Rate	ROC Area	Class
83.06%	80.1%	14.6%	88.9%	Approve
	85.4%	19.9%	88.9%	Disapprove

RandomForest - Test set (Set 1)

Accuracy	TP Rate	FP Rate	ROC Area	Class
84.81%	87.3%	17.2%	88.3%	Approve
	82.8%	12.7%	88.3%	Disapprove

4. MultilayerPerceptron

- Step 1: Choose MultilayerPerceptron and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set1_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

```

==== Re-evaluation on test set ====
User supplied test set
Relation: R_data_frame-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst-last-weka.filters.unsupervis
Instances: unknown (yet). Reading incrementally
Attributes: 4

==== Summary ====
Correctly Classified Instances      134          84.8101 %
Incorrectly Classified Instances   24           15.1899 %
Kappa statistic                   0.6938
Mean absolute error               0.1474
Root mean squared error          0.2893
Total Number of Instances        158

==== Detailed Accuracy By Class ====
      TP Rate  FP Rate  Precision  Recall   F-Measure  MCC    ROC Area  PRC Area  Class
      0.845    0.149    0.822     0.845    0.833     0.694   0.877    0.844    Approve
      0.851    0.155    0.871     0.851    0.860     0.694   0.877    0.849    Disapprove
      ?         0.000    ?         ?         ?         ?         ?         ?         Skipped
Weighted Avg.      0.848    0.152    0.849     0.848    0.848     0.694   0.877    0.847

==== Confusion Matrix ====
a  b  c  <-- classified as
60 11  0 |  a = Approve
13 74  0 |  b = Disapprove
 0  0  0 |  c = Skipped

```

MultiLayerPerceptron - Cross Validation (Set 1)

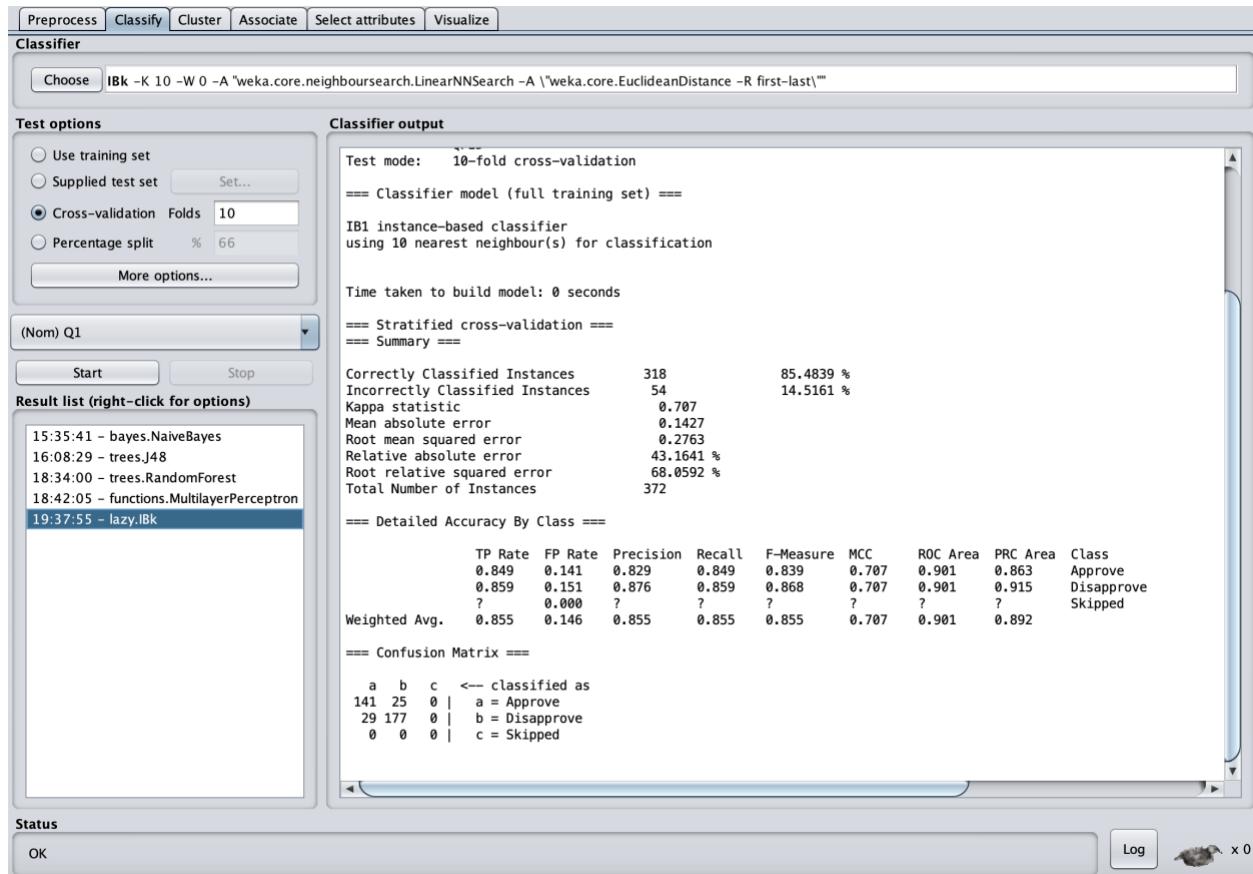
Accuracy	TP Rate	FP Rate	ROC Area	Class
82.53%	81.3%	16.5%	87.6%	Approve
	83.5%	18.7%	87.6%	Disapprove

MultiLayerPerceptron - Test set (Set 1)

Accuracy	TP Rate	FP Rate	ROC Area	Class
84.81%	84.5%	14.9%	87.7%	Approve
	85.1%	15.5%	87.7%	Disapprove

5. K-nearest Neighbor (IBK) with K = 10

- Step 1: Choose IBK and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: Select ‘Supplied test set’ and click ‘Set’ -> Open file, open “set1_testing” with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

- 15:35:41 - bayes.NaiveBayes
- 16:08:29 - trees.J48
- 18:34:00 - trees.RandomForest
- 18:42:05 - functions.MultilayerPerceptron
- 19:37:55 - lazy.IBk**

```
== Re-evaluation on test set ==
User supplied test set
Relation: R_data_frame-weka.filters.unsupervised.attribute.NumericToNominal-Rfirst-last-weka.filters.unsupervised
Instances: unknown (yet). Reading incrementally
Attributes: 4

== Summary ==
Correctly Classified Instances      133          84.1772 %
Incorrectly Classified Instances    25           15.8228 %
Kappa statistic                   0.6823
Mean absolute error               0.1469
Root mean squared error           0.2839
Total Number of Instances         158

== Detailed Accuracy By Class ==
      TP Rate   FP Rate   Precision   Recall   F-Measure   MCC   ROC Area   PRC Area   Class
      0.859     0.172     0.803     0.859     0.830     0.684     0.891     0.853     Approve
      0.828     0.141     0.878     0.828     0.852     0.684     0.891     0.880     Disapprove
      ?          0.000     ?          ?          ?          ?          ?          ?          Skipped
Weighted Avg.      0.842     0.155     0.844     0.842     0.842     0.684     0.891     0.868

== Confusion Matrix ==
      a   b   c   <-- classified as
  61 10  0 |  a = Approve
  15 72  0 |  b = Disapprove
    0  0  0 |  c = Skipped
```

Status OK Log x 0

IBK - Cross Validation (Set1)

Accuracy	TP Rate	FP Rate	ROC Area	Class
85.48%	84.9%	14.1%	90.1%	Approve
	85.9%	15.1%	90.1%	Disapprove

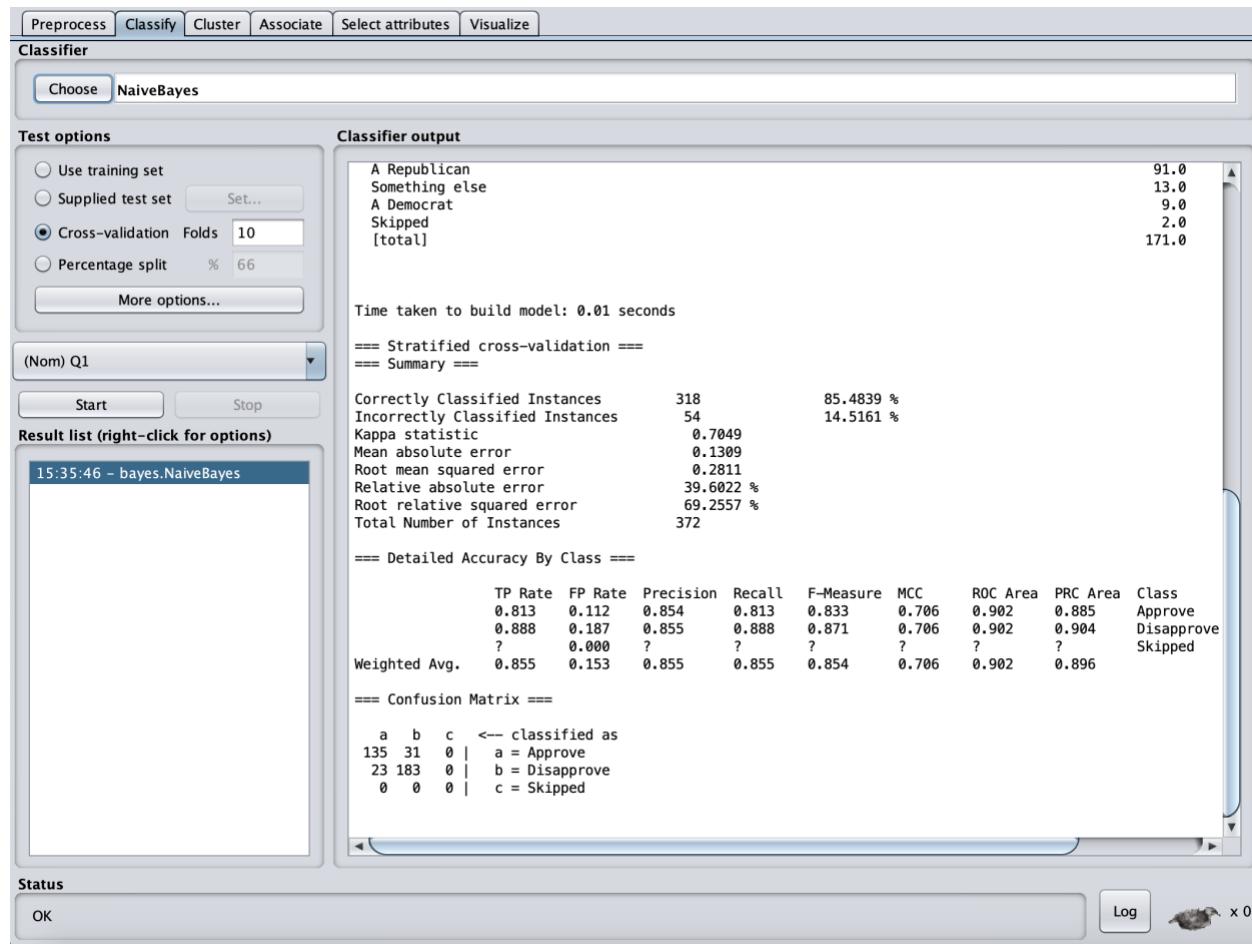
IBK - Test set (Set1)

Accuracy	TP Rate	FP Rate	ROC Area	Class
84.18%	85.9%	17.2%	89.1%	Approve
	82.8%	14.1%	89.1%	Disapprove

b. *Attribute Set 2: [Q3, QPID, Q2, Q1] - by the GainRatioAttributeEval*

1. Naive Bayes

- Step 1: Open “Initial_Training_set.arff”, remove all attributes except Q3, QPID, Q2, Q1 and save it as “set2_training”
- Step 2: Open “Initial_Testing_set.arff”, remove all attributes except Q3, QPID, Q2, Q1 and save it as “set2_testing”
- Step 3: Open “set2_training”, Classify -> Choose NaiveBayes with Cross-validation (10 Folds), Q1 as class attribute -> Start



- Step 4: In order to test our model, we re-evaluate our model with the test dataset. Select ‘Supplied test set’ and click ‘Set’ -> Open file, open “set2_testing” with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose NaiveBayes

Test options

- Use training set
- Supplied test set Set...
- Cross-validation Folds 10
- Percentage split % 66
- [More options...](#)

(Nom) Q1

Start Stop

Result list (right-click for options)

```

20:46:57 - lazy.IBk
20:48:50 - lazy.IBk
21:05:49 - bayes.NaiveBayes
21:09:34 - bayes.NaiveBayes
21:20:32 - trees.J48
21:21:57 - trees.J48
21:26:12 - trees.RandomForest
21:26:54 - trees.RandomForest
21:30:32 - functions.MultilayerPerceptron
21:31:46 - functions.MultilayerPerceptron
21:36:13 - lazy.IBk
21:37:14 - lazy.IBk
22:06:50 - bayes.NaiveBayes
22:07:46 - bayes.NaiveBayes
22:08:57 - trees.J48
22:09:17 - trees.J48
22:10:19 - trees.RandomForest
22:10:45 - trees.RandomForest
22:12:11 - functions.MultilayerPerceptron
22:12:38 - functions.MultilayerPerceptron
22:13:56 - lazy.IBk
22:14:18 - lazy.IBk
22:15:39 - bayes.NaiveBayes
22:17:00 - bayes.NaiveBayes
22:19:36 - bayes.NaiveBayes
22:20:10 - bayes.NaiveBayes

```

Classifier output

14Z	2:Disapprove	2:Disapprove	0.951
143	2:Disapprove	2:Disapprove	0.994
144	2:Disapprove	2:Disapprove	0.951
145	2:Disapprove	2:Disapprove	0.994
146	2:Disapprove	2:Disapprove	0.994
147	2:Disapprove	2:Disapprove	0.965
148	2:Disapprove	2:Disapprove	0.965
149	2:Disapprove	2:Disapprove	0.532
150	2:Disapprove	2:Disapprove	0.753
151	2:Disapprove	2:Disapprove	0.934
152	2:Disapprove	2:Disapprove	0.951
153	2:Disapprove	2:Disapprove	0.657
154	2:Disapprove	2:Disapprove	0.994
155	2:Disapprove	2:Disapprove	0.965
156	2:Disapprove	1:Approve	+ 0.993
157	2:Disapprove	2:Disapprove	0.91
158	2:Disapprove	2:Disapprove	0.951

== Evaluation on test set ==

Time taken to test model on supplied test set: 0.01 seconds

== Summary ==

Correctly Classified Instances	133	84.1772 %
Incorrectly Classified Instances	25	15.8228 %
Kappa statistic	0.6773	
Mean absolute error	0.1397	
Root mean squared error	0.293	
Relative absolute error	42.2544 %	
Root relative squared error	72.1456 %	
Total Number of Instances	158	

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.775	0.103	0.859	0.775	0.815	0.680	0.891	0.841	0.841	Approve
0.897	0.225	0.830	0.897	0.862	0.680	0.891	0.896	0.896	Disapprove
?	0.000	?	?	?	?	?	?	?	Skipped
Weighted Avg.	0.842	0.171	0.843	0.842	0.841	0.680	0.891	0.871	

== Confusion Matrix ==

a	b	c	<-- classified as
55	16	0	a = Approve
9	78	0	b = Disapprove
0	0	0	c = Skipped

Status

OK Log x 0

Naive Bayes - Cross Validation (Set 2)

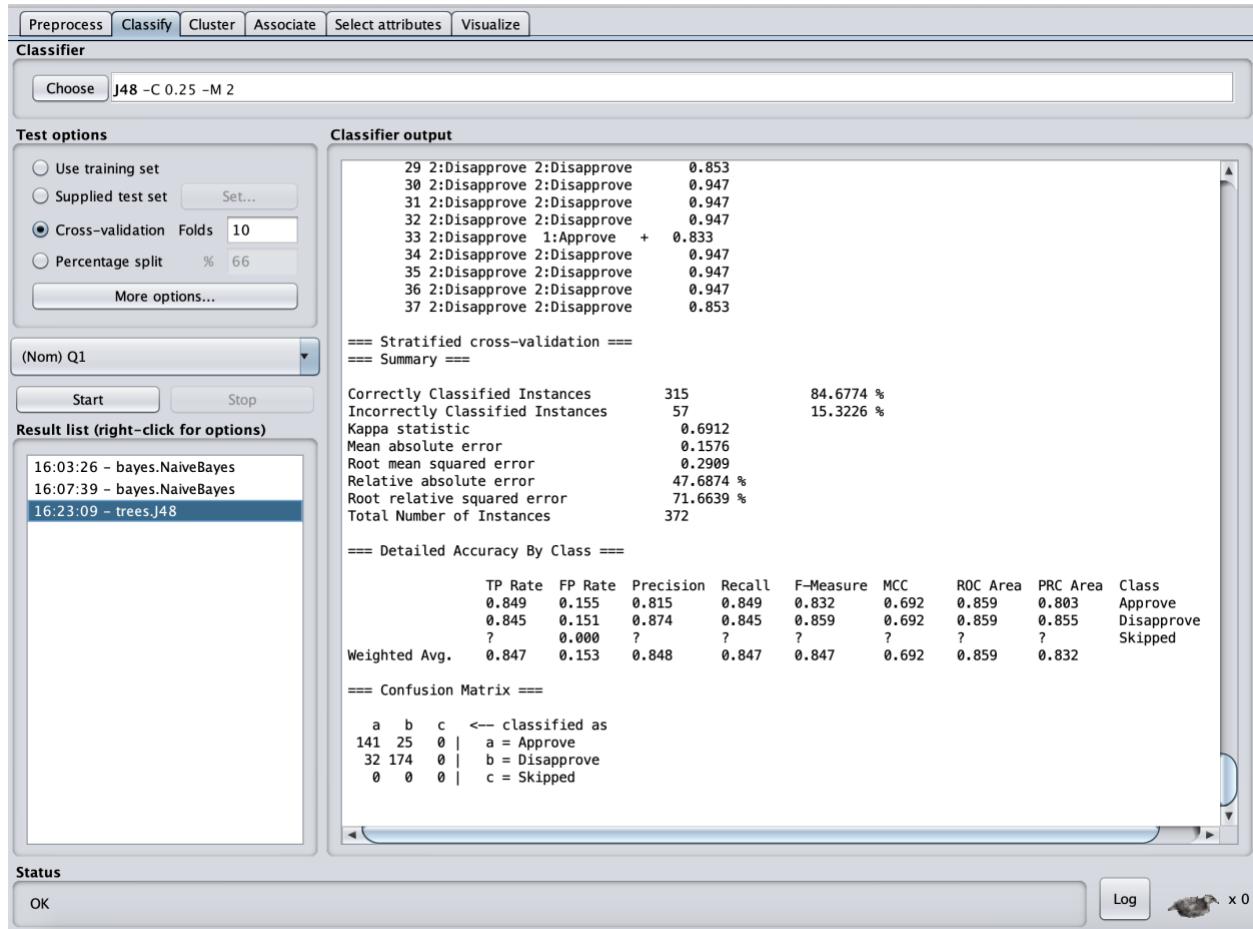
Accuracy	TP Rate	FP Rate	ROC Area	Class
85.48%	81.3%	11.2%	90.2%	Approve
	88.8%	18.7%	90.2%	Disapprove

Naive Bayes - Test set (Set 2)

Accuracy	TP Rate	FP Rate	ROC Area	Class
84.18%	77.5%	10.3%	89.1%	Approve
	89.7%	22.5%	89.1%	Disapprove

2. J48

- Step 1: Choose J48 and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: Select 'Supplied test set' and click 'Set' -> Open file, open set2_testing with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

- Use training set
- Supplied test set Set...
- Cross-validation Folds 10
- Percentage split % 66

More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

- 16:03:26 - bayes.NaiveBayes
- 16:07:39 - bayes.NaiveBayes
- 16:23:09 - trees.J48
- 16:25:13 - trees.J48

Classifier output

```

153 2:Disapprove 2:Disapprove 0.667
154 2:Disapprove 2:Disapprove 0.953
155 2:Disapprove 2:Disapprove 0.953
156 2:Disapprove 1:Approve + 0.835
157 2:Disapprove 2:Disapprove 0.953
158 2:Disapprove 2:Disapprove 0.831

```

== Evaluation on test set ==

Time taken to test model on supplied test set: 0.03 seconds

== Summary ==

	Correctly Classified Instances	131	82.9114 %
Incorrectly Classified Instances	27	17.0886 %	
Kappa statistic	0.6542		
Mean absolute error	0.1668		
Root mean squared error	0.3002		
Relative absolute error	50.4455 %		
Root relative squared error	73.9164 %		
Total Number of Instances	158		

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.803	0.149	0.814	0.803	0.809	0.654	0.860	0.778	0.848	Approve
0.851	0.197	0.841	0.851	0.846	0.654	0.860	0.848	?	Disapprove
?	0.000	?	?	?	?	?	?	?	Skipped
Weighted Avg.	0.829	0.176	0.829	0.829	0.654	0.860	0.817		

== Confusion Matrix ==

a	b	c	<-- classified as
57	14	0	a = Approve
13	74	0	b = Disapprove
0	0	0	c = Skipped

Status OK Log x 0

J48 - Cross Validation (Set 2)

Accuracy	TP Rate	FP Rate	ROC Area	Class
84.68%	84.9%	15.5%	85.9%	Approve
	84.5%	15.1%	85.9%	Disapprove

J48 - Test set (Set 2)

Accuracy	TP Rate	FP Rate	ROC Area	Class
82.91%	80.3%	14.9%	86.0%	Approve
	85.1%	19.7%	86.0%	Disapprove

3. Random Forest

- Step 1: Choose RandomForest and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start

The screenshot shows the Weka interface with the 'Classifier' tab selected. The 'Choose' button is set to 'RandomForest'. In the 'Test options' section, 'Cross-validation' is selected with 'Folds' set to 10. The 'Classifier output' pane displays the following results:

```

28 2:Disapprove 2:Disapprove 0.799
29 2:Disapprove 2:Disapprove 0.799
30 2:Disapprove 2:Disapprove 0.982
31 2:Disapprove 2:Disapprove 0.883
32 2:Disapprove 2:Disapprove 0.982
33 2:Disapprove 1:Approve + 0.921
34 2:Disapprove 2:Disapprove 0.982
35 2:Disapprove 2:Disapprove 0.982
36 2:Disapprove 2:Disapprove 0.982
37 2:Disapprove 2:Disapprove 0.893

== Stratified cross-validation ==
== Summary ==

Correctly Classified Instances      316          84.9462 %
Incorrectly Classified Instances   56           15.0538 %
Kappa statistic                   0.6943
Mean absolute error               0.1455
Root mean squared error          0.2838
Relative absolute error          44.01 %
Root relative squared error     69.9051 %
Total Number of Instances        372

== Detailed Accuracy By Class ==

      TP Rate  FP Rate  Precision  Recall   F-Measure  MCC    ROC Area  PRC Area  Class
          0.813   0.121    0.844    0.813    0.828    0.695   0.890   0.870   Approve
          0.879   0.187    0.854    0.879    0.866    0.695   0.890   0.877   Disapprove
            ?      0.000    ?         ?       ?        ?       ?        ?       Skipped
Weighted Avg.    0.849   0.158    0.849    0.849    0.849    0.695   0.890   0.874

== Confusion Matrix ==

  a   b   c   <-- classified as
135  31  0 |  a = Approve
 25 181  0 |  b = Disapprove
  0   0  0 |  c = Skipped

```

The 'Status' pane shows 'OK'.

- Step 2: Select 'Supplied test set' and click 'Set' -> Open file, open set2_testing with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1

Test options

- Use training set
- Supplied test set Set...
- Cross-validation Folds 10
- Percentage split % 66

More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

- 16:03:26 - bayes.NaiveBayes
- 16:07:39 - bayes.NaiveBayes
- 16:23:09 - trees.J48
- 16:25:13 - trees.J48
- 16:33:19 - trees.RandomForest
- 16:35:24 - trees.RandomForest

Classifier output

```

152 2:Disapprove 2:Disapprove 0.847
153 2:Disapprove 2:Disapprove 0.67
154 2:Disapprove 2:Disapprove 0.988
155 2:Disapprove 2:Disapprove 0.896
156 2:Disapprove 1:Approve + 0.962
157 2:Disapprove 2:Disapprove 1
158 2:Disapprove 2:Disapprove 0.847

== Evaluation on test set ==

Time taken to test model on supplied test set: 0.03 seconds

== Summary ==

Correctly Classified Instances 133 84.1772 %
Incorrectly Classified Instances 25 15.8228 %
Kappa statistic 0.6773
Mean absolute error 0.1552
Root mean squared error 0.2972
Relative absolute error 46.9484 %
Root relative squared error 73.1764 %
Total Number of Instances 158

== Detailed Accuracy By Class ==

      TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
0.775 0.183 0.859 0.775 0.815 0.680 0.874 0.817 Approve
0.897 0.225 0.830 0.897 0.862 0.680 0.874 0.870 Disapprove
? 0.000 ? ? ? ? ? ? Skipped
Weighted Avg. 0.842 0.171 0.843 0.842 0.841 0.680 0.874 0.846

== Confusion Matrix ==

a b c <-- classified as
55 16 0 | a = Approve
9 78 0 | b = Disapprove
0 0 0 | c = Skipped

```

Status

OK

Log

Random Forest - Cross Validation (Set 2)

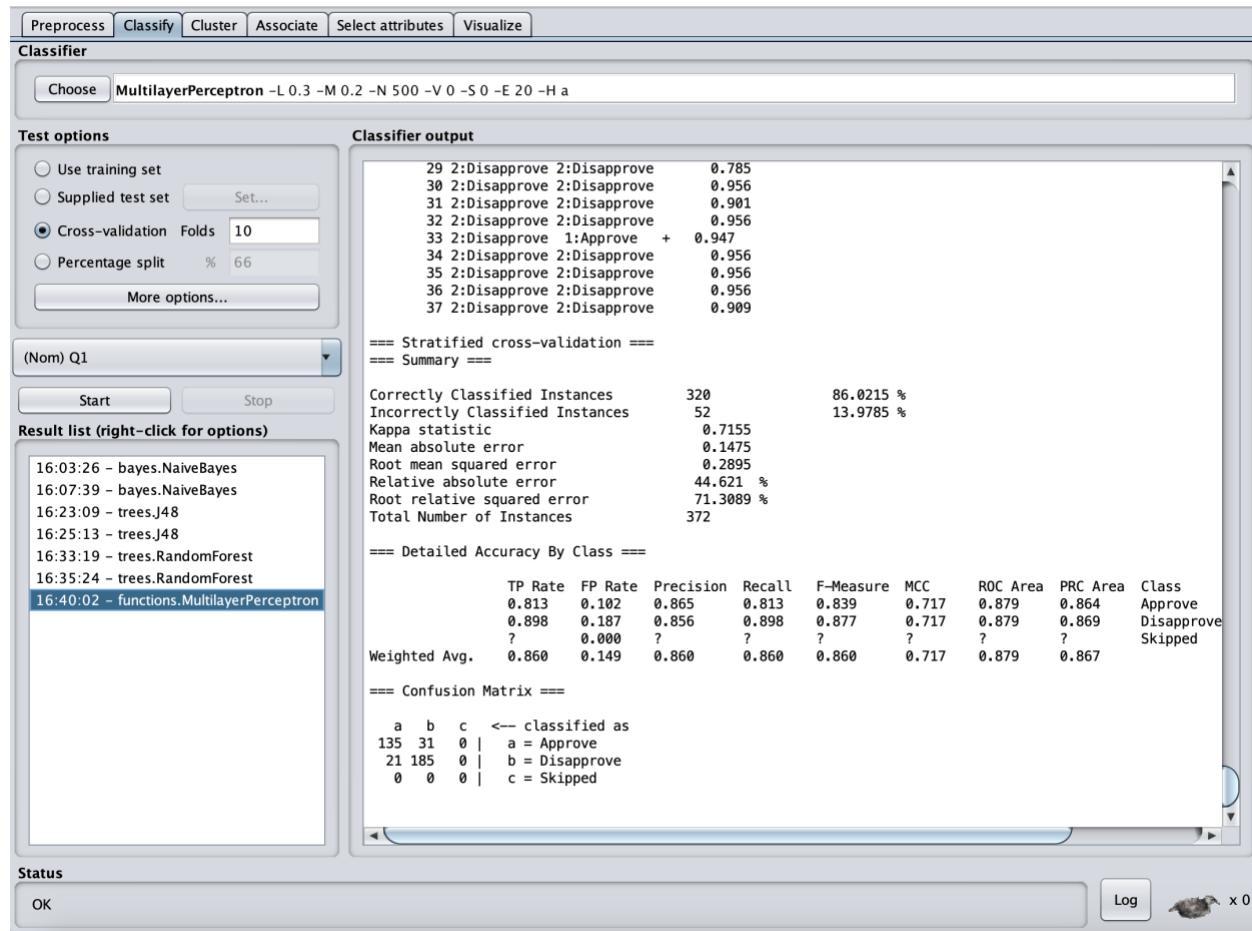
Accuracy	TP Rate	FP Rate	ROC Area	Class
84.95%	81.3%	12.1%	89.0%	Approve
	87.9%	18.7%	89.0%	Disapprove

Random Forest - Test set (Set 2)

Accuracy	TP Rate	FP Rate	ROC Area	Class
84.18%	77.5%	10.3%	87.4%	Approve
	89.7%	22.5%	87.4%	Disapprove

4. MultilayerPerceptron

- Step 1: Choose MultilayerPerceptron and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: Select 'Supplied test set' and click 'Set' -> Open file, open set2_testing with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose MultilayerPerceptron -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a

Test options

- Use training set
- Supplied test set
- Cross-validation Folds 10
- Percentage split % 66

(Nom) Q1

Result list (right-click for options)

```
16:03:26 - bayes.NaiveBayes
16:07:39 - bayes.NaiveBayes
16:23:09 - trees.J48
16:25:13 - trees.J48
16:33:19 - trees.RandomForest
16:35:24 - trees.RandomForest
16:40:02 - functions.MultilayerPerceptron
16:41:17 - functions.MultilayerPerceptron
```

Classifier output

```
153 2:Disapprove 2:Disapprove 0.682
154 2:Disapprove 2:Disapprove 0.982
155 2:Disapprove 2:Disapprove 0.907
156 2:Disapprove 1:Approve + 0.966
157 2:Disapprove 2:Disapprove 0.983
158 2:Disapprove 2:Disapprove 0.892

==== Evaluation on test set ====
Time taken to test model on supplied test set: 0.02 seconds

==== Summary ====
Correctly Classified Instances 133 84.1772 %
Incorrectly Classified Instances 25 15.8228 %
Kappa statistic 0.6773
Mean absolute error 0.1519
Root mean squared error 0.2943
Relative absolute error 45.9494 %
Root relative squared error 72.4522 %
Total Number of Instances 158

==== Detailed Accuracy By Class ====


|               | TP Rate | FP Rate | Precision | Recall | F-Measure | MCC   | ROC Area | PRC Area | Class      |
|---------------|---------|---------|-----------|--------|-----------|-------|----------|----------|------------|
| 0.775         | 0.103   | 0.859   | 0.775     | 0.815  | 0.680     | 0.881 | 0.838    | 0.881    | Approve    |
| 0.897         | 0.225   | 0.830   | 0.897     | 0.862  | 0.680     | 0.881 | 0.873    | 0.881    | Disapprove |
| ?             | 0.000   | ?       | ?         | ?      | ?         | ?     | ?        | ?        | Skipped    |
| Weighted Avg. | 0.842   | 0.171   | 0.843     | 0.842  | 0.841     | 0.680 | 0.881    | 0.857    |            |


==== Confusion Matrix ====


|    |    |   |                   |
|----|----|---|-------------------|
| a  | b  | c | <-- classified as |
| 55 | 16 | 0 | a = Approve       |
| 9  | 78 | 0 | b = Disapprove    |
| 0  | 0  | 0 | c = Skipped       |


```

Status

OK  x 0

MultilayerPerceptron - Cross Validation (Set 2)

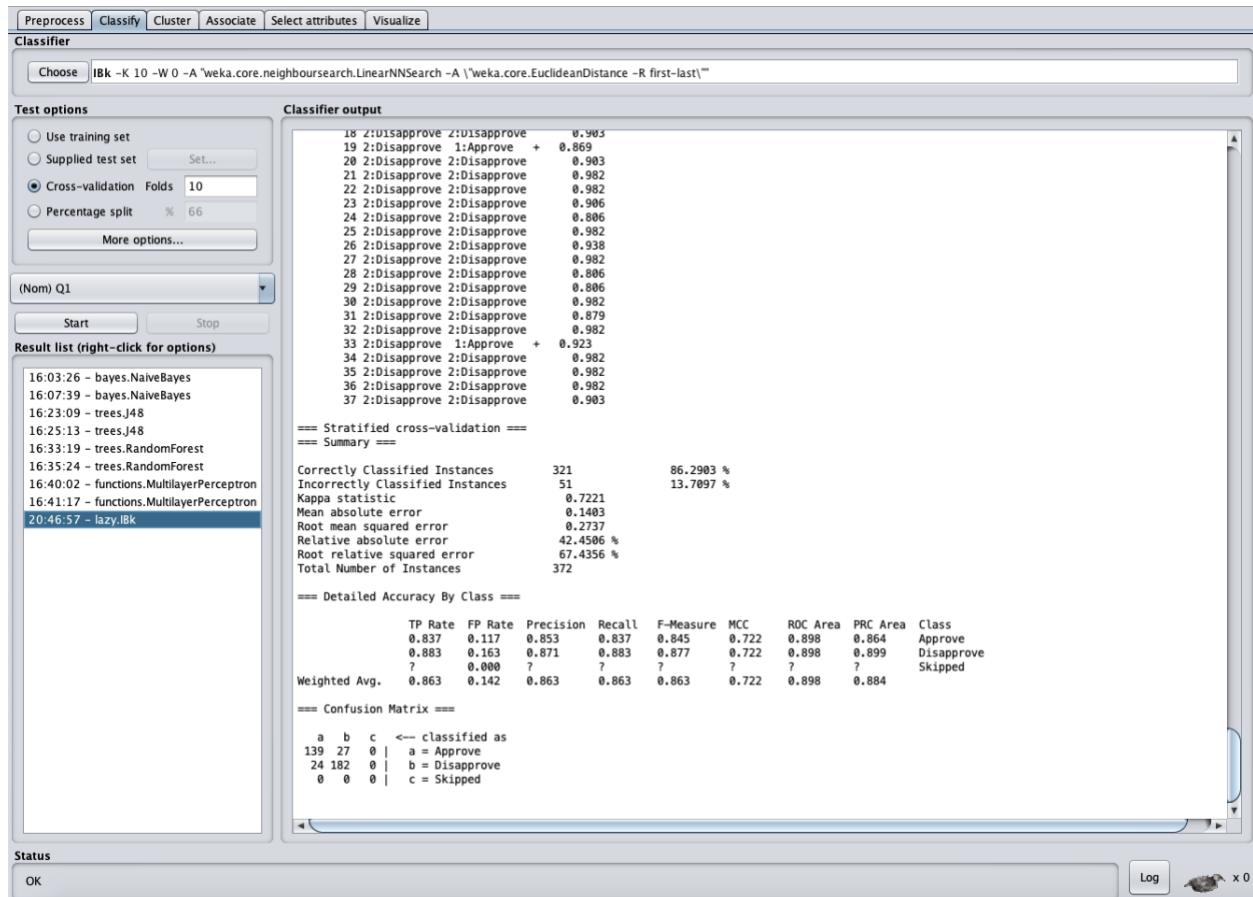
Accuracy	TP Rate	FP Rate	ROC Area	Class
86.02%	81.3%	10.2%	87.9%	Approve
	89.8%	18.7%	87.9%	Disapprove

MultilayerPerceptron - Test set (Set 2)

Accuracy	TP Rate	FP Rate	ROC Area	Class
84.18%	77.5%	10.3%	88.1%	Approve
	89.7%	22.5%	88.1%	Disapprove

5. K-nearest Neighbor (IBK) with K = 10

- Step 1: Choose IBK and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: Select 'Supplied test set' and click 'Set' -> Open file, open set2_testing with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose: IBk -K 10 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

Test options

- Use training set
- Supplied test set Set...
- Cross-validation Folds 10
- Percentage split % 66
- [More options...](#)

(Nom) Q1

Start Stop

Result list (right-click for options)

```

16:03:26 - bayes.NaiveBayes
16:07:39 - bayes.NaiveBayes
16:23:09 - trees.J48
16:25:13 - trees.J48
16:33:19 - trees.RandomForest
16:35:24 - trees.RandomForest
16:40:02 - functions.MultilayerPerceptron
16:41:17 - functions.MultilayerPerceptron
20:46:57 - lazy.IBk
20:48:50 - lazy.IBk
  
```

16:48:50 - lazy.IBk

Classifier output

```

142 2:Disapprove 2:Disapprove 0.838
143 2:Disapprove 2:Disapprove 0.985
144 2:Disapprove 2:Disapprove 0.838
145 2:Disapprove 2:Disapprove 0.985
146 2:Disapprove 2:Disapprove 0.985
147 2:Disapprove 2:Disapprove 0.882
148 2:Disapprove 2:Disapprove 0.882
149 2:Disapprove 2:Disapprove 0.696
150 2:Disapprove 2:Disapprove 0.823
151 2:Disapprove 2:Disapprove 0.894
152 2:Disapprove 2:Disapprove 0.838
153 2:Disapprove 2:Disapprove 0.747
154 2:Disapprove 2:Disapprove 0.985
155 2:Disapprove 2:Disapprove 0.882
156 2:Disapprove 1:Approve + 0.956
157 2:Disapprove 2:Disapprove 0.922
158 2:Disapprove 2:Disapprove 0.838
  
```

== Evaluation on test set ==

Time taken to test model on supplied test set: 0.03 seconds

== Summary ==

	Correctly Classified Instances	132	83.5443 %
Incorrectly Classified Instances	26	16.4557 %	
Kappa statistic	0.6666		
Mean absolute error	0.1558		
Root mean squared error	0.2938		
Relative absolute error	47.1295 %		
Root relative squared error	72.3466 %		
Total Number of Instances	158		

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.803	0.138	0.826	0.803	0.814	0.667	0.885	0.852	0.852	Approve
0.862	0.197	0.843	0.862	0.852	0.667	0.885	0.885	0.885	Disapprove
?	0.000	?	?	?	?	?	?	?	Skipped
Weighted Avg.	0.835	0.171	0.835	0.835	0.835	0.667	0.885	0.870	

== Confusion Matrix ==

a	b	c	<-- classified as
57	14	0	a = Approve
12	75	0	b = Disapprove
0	0	0	c = Skipped

Status OK Log x 0

IBK - Cross Validation (Set 2)

Accuracy	TP Rate	FP Rate	ROC Area	Class
86.29%	83.7%	11.7%	89.8%	Approve
	88.3%	16.3%	89.8%	Disapprove

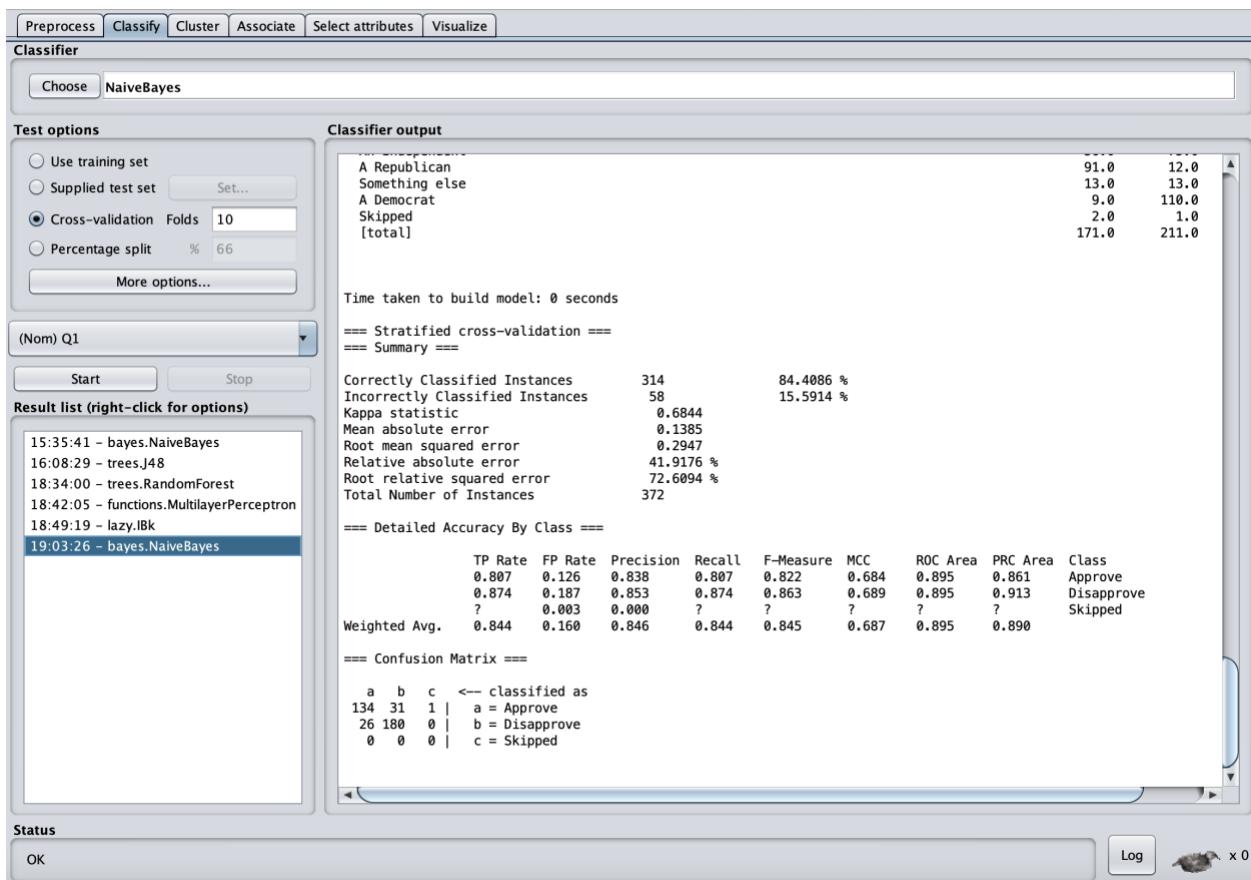
IBK - Test set (Set 2)

Accuracy	TP Rate	FP Rate	ROC Area	Class
83.54%	80.3%	13.8%	88.5%	Approve
	86.2%	19.7%	88.5%	Disapprove

c. Attribute Set 3: [QPID, Q3, Q2, ppage, ppstaten, Q1] - by the InfoGainAttributeEval

1. Naive Bayse:

- Step 1: Open “Initial_Training_set.arff”, remove all attributes except Q3, QPID, Q2, ppage, ppstaten, and Q1. Save it as “set3_training”
- Step 2: Open “Initial_Testing_set.arff”, remove all attributes except Q3, QPID, Q2, ppage, ppstaten, and Q1. Save it as “set3_testing”
- Step 3: Open “set3_training”, Classify -> Choose NaiveBayes with Cross-validation (10 Folds), Q1 as class attribute -> Start



- Step 4: In order to test our model, we re-evaluate our model with the test dataset. Select ‘Supplied test set’ and click ‘Set’ -> Open file, open “set3_testing” with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

```

More options...
(Nom) Q1
Start Stop
Result list (right-click for options)
15:35:41 - bayes.NaiveBayes
16:08:29 - trees.J48
18:34:00 - trees.RandomForest
18:42:05 - functions.MultilayerPerceptron
18:49:19 - lazy.IBK
19:03:26 - bayes.NaiveBayes

== Re-evaluation on test set ==
User supplied test set
Relation: R_data_frame-weka.filters.unsupervised.attribute.Remove-R1,3-13,18-20,22-23
Instances: unknown (yet). Reading incrementally
Attributes: 6

== Summary ==
Correctly Classified Instances      129          81.6456 %
Incorrectly Classified Instances   29           18.3544 %
Kappa statistic                   0.6286
Mean absolute error               0.1463
Root mean squared error          0.3015
Total Number of Instances        158

== Detailed Accuracy By Class ==
      TP Rate  FP Rate  Precision  Recall   F-Measure  MCC    ROC Area  PRC Area  Class
0.789     0.161    0.800      0.789    0.794     0.629    0.890    0.868    Approve
0.839     0.211    0.830      0.839    0.834     0.629    0.887    0.879    Disapprove
?         0.000    ?          ?        ?        ?        ?        ?        ?
Weighted Avg.  0.816    0.189    0.816      0.816    0.816     0.629    0.888    0.874    Skipped

== Confusion Matrix ==
      a   b   c  <-- classified as
56  15  0 |  a = Approve
14  73  0 |  b = Disapprove
 0   0  1 |  c = Skipped

```

Status: OK

Naive Bayes - Cross Validation (Set 3)

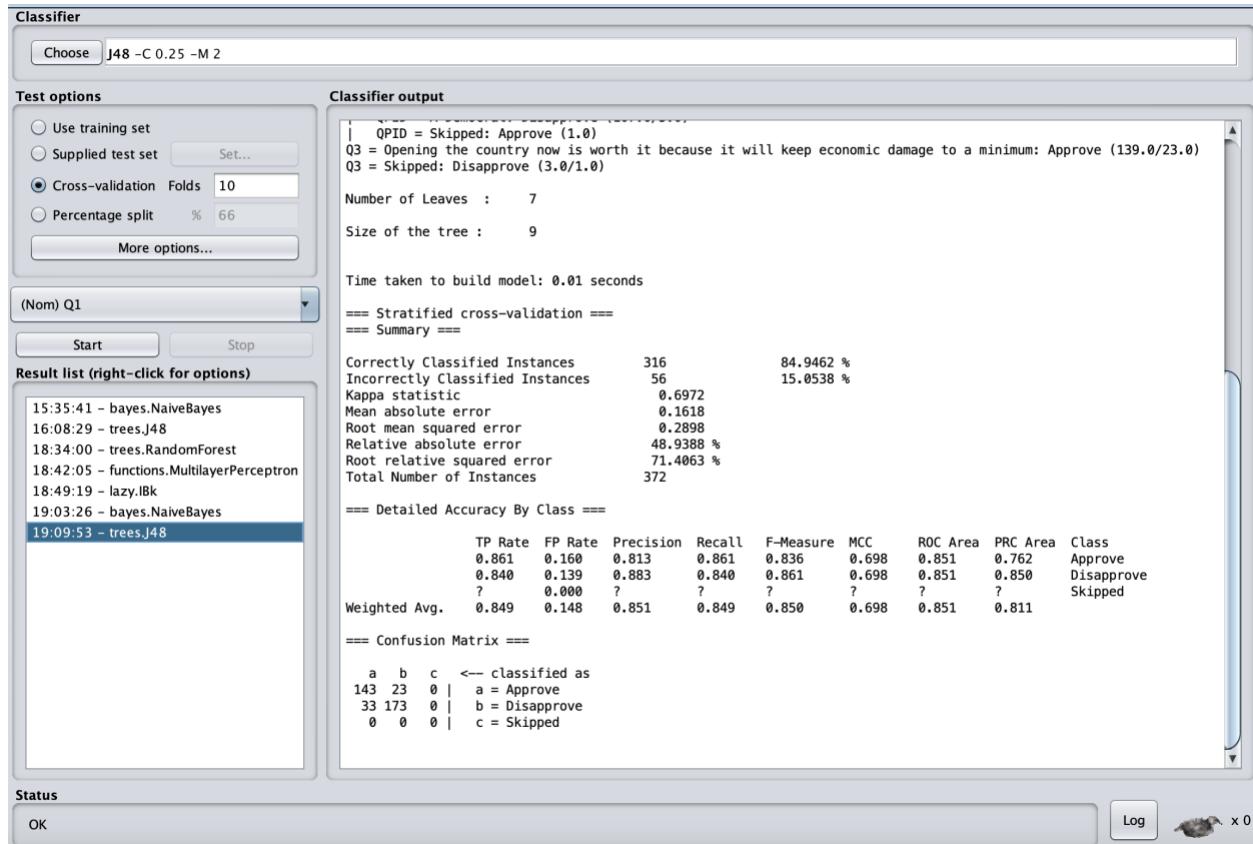
Accuracy	TP Rate	FP Rate	ROC Area	Class
84.41%	80.7%	12.6%	89.5%	Approve
	87.4%	18.7%	89.5%	Disapprove

Naive Bayes - Test set (Set 3)

Accuracy	TP Rate	FP Rate	ROC Area	Class
81.65%	78.9%	16.1%	89.0%	Approve
	83.9%	21.1%	88.7%	Disapprove

2. J48

- Step 1: First to assess how good J48 is, use Cross-validation to build our model. Choose J48 and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set3_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

```

15:35:41 - bayes.NaiveBayes
16:08:29 - trees.J48
18:34:00 - trees.RandomForest
18:42:05 - functions.MultilayerPerceptron
18:49:19 - lazy.IBK
19:03:26 - bayes.NaiveBayes
19:09:53 - trees.J48

```

== Re-evaluation on test set ==

User supplied test set
Relation: R_data_frame-weka.filters.unsupervised.attribute.Remove-R1,3-13,18-20,22-23
Instances: unknown (yet). Reading incrementally
Attributes: 6

== Summary ==

	Correctly Classified Instances	134	84.8101 %
Incorrectly Classified Instances	24	15.1899 %	
Kappa statistic	0.6954		
Mean absolute error	0.1636		
Root mean squared error	0.2931		
Total Number of Instances	158		

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.873	0.172	0.805	0.873	0.838	0.697	0.859	0.776	Approve	
0.828	0.127	0.889	0.828	0.857	0.697	0.859	0.848	Disapprove	
?	0.000	?	?	?	?	?	?	Skipped	
Weighted Avg.	0.848	0.147	0.851	0.848	0.848	0.697	0.859	0.816	

== Confusion Matrix ==

a	b	c	<-- classified as
62	9	0	a = Approve
15	72	0	b = Disapprove
0	0	0	c = Skipped

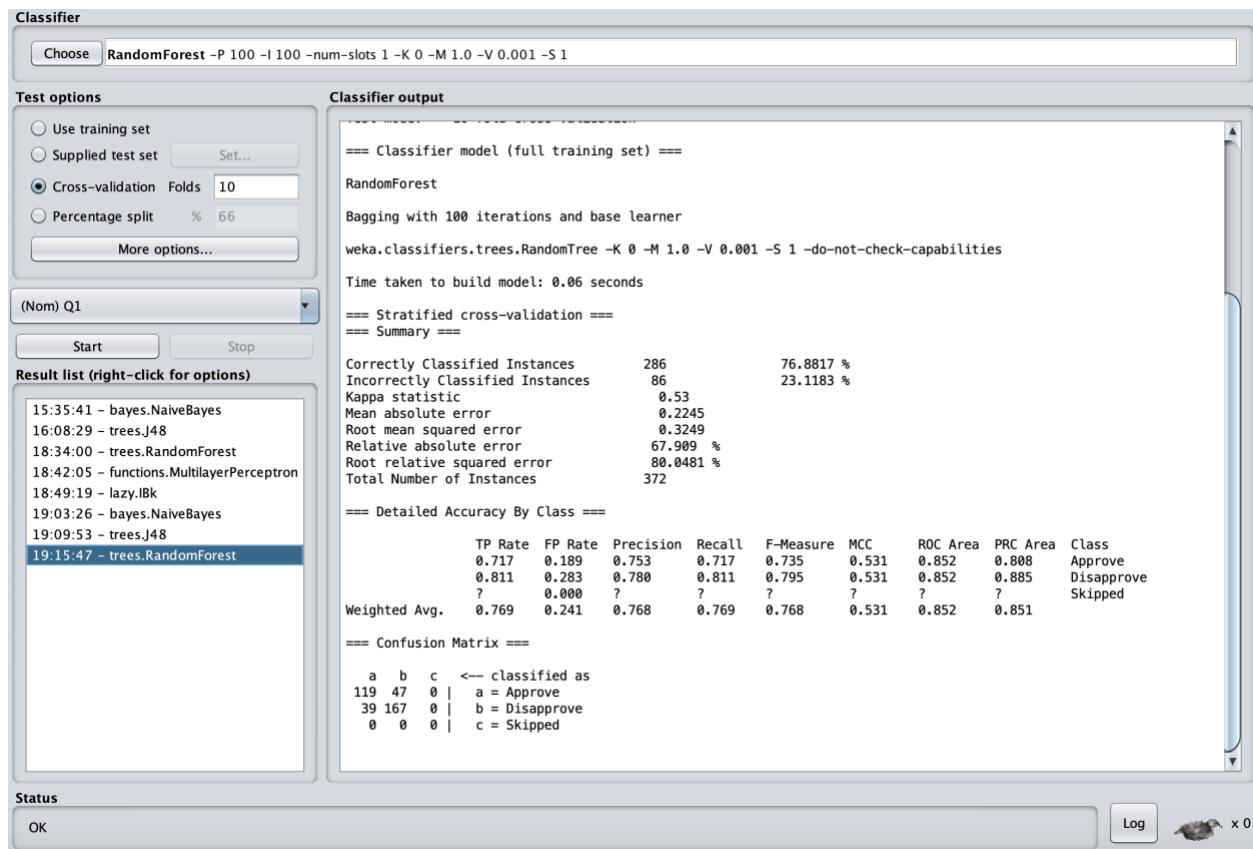
Status OK Log x 0

J48 - Cross Validation (Set 3)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
84.95%	86.1%	16.0%	85.1%	Approve
	84.0%	13.9%	85.1%	Disapprove

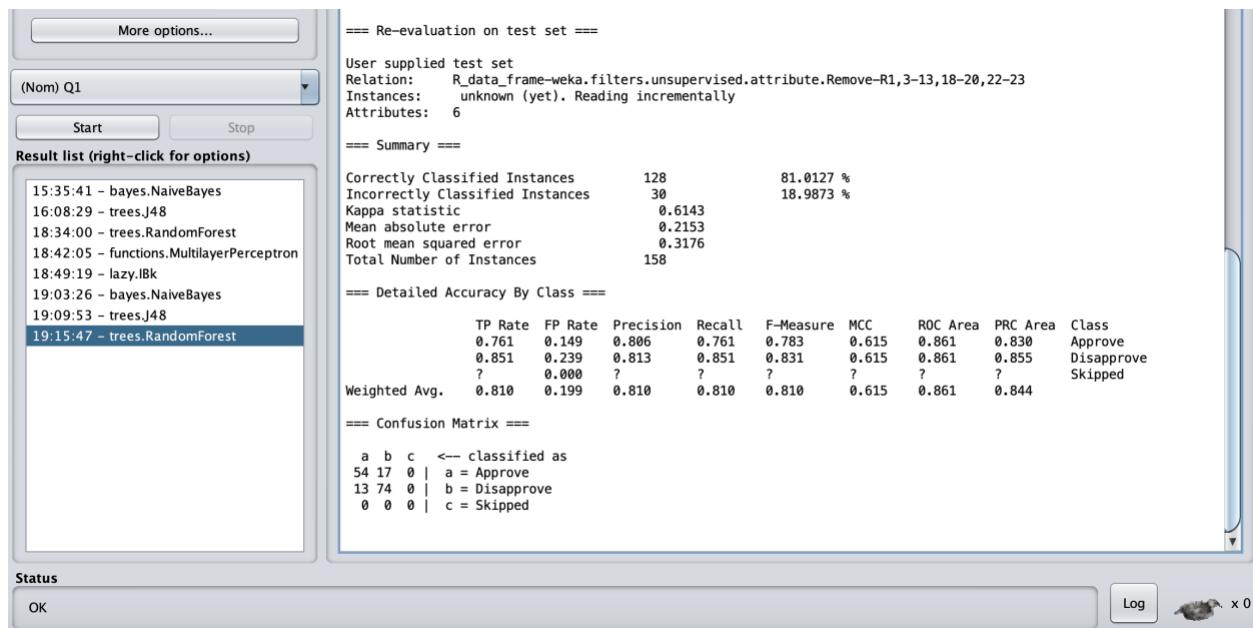
J48 - Test set (Set 3)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
84.81%	87.3%	17.2%	85.9%	Approve
	82.8%	12.7%	85.9%	Disapprove

3. Random Forest

- Step 1: Choose RandomForest and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: Select 'Supplied test set' and click 'Set' -> Open file, open "set3_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

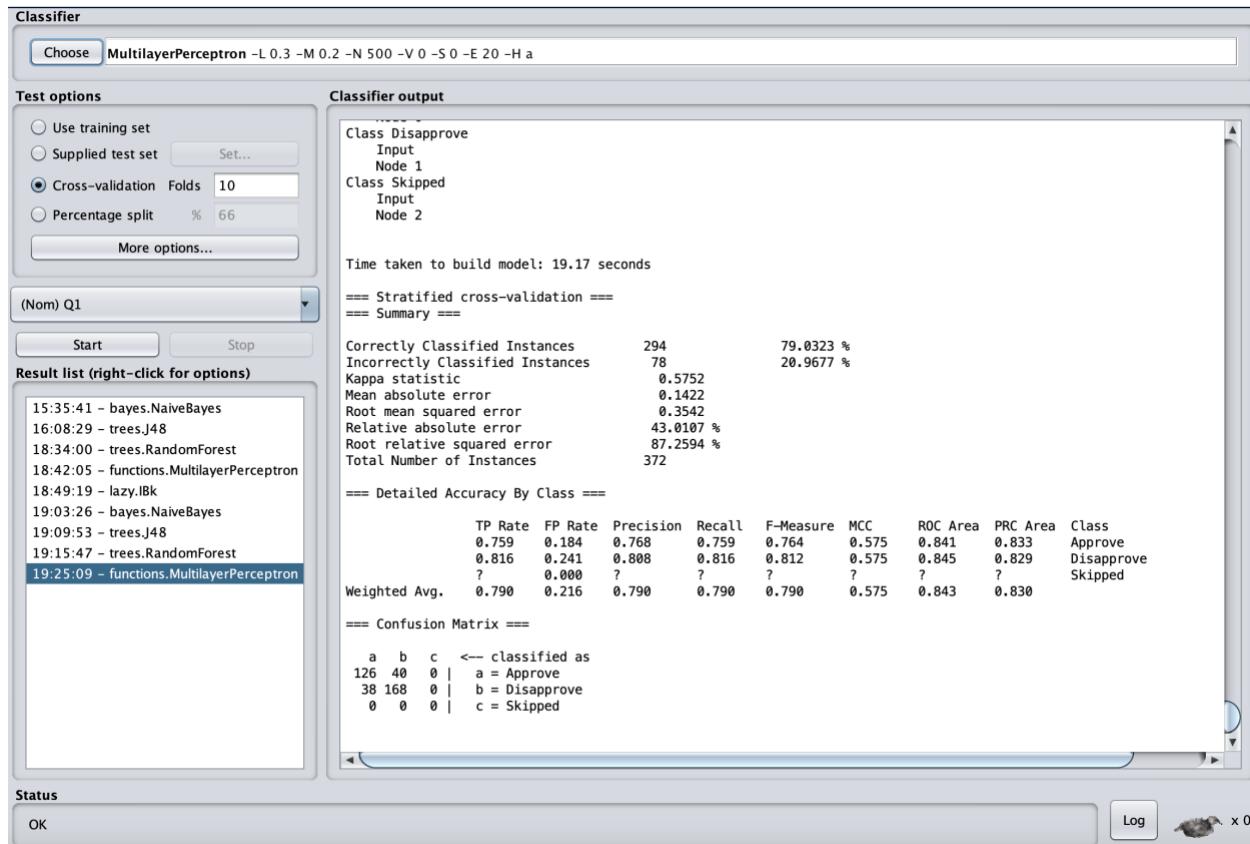


RandomForest - Cross Validation (Set 3)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
76.88%	71.7%	18.9%	85.2%	Approve
	81.1%	28.3%	85.2%	Disapprove

RandomForest - Test set (Set 3)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
81.01%	76.1%	14.9%	86.1%	Approve
	85.1%	23.9%	86.1%	Disapprove

4. MultiLayerPerceptron

- Step 1: Choose MultilayerPerceptron and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: Select 'Supplied test set' and click 'Set' -> Open file, open "set3_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Percentage split % 66
 More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

```

15:35:41 - bayes.NaiveBayes
16:08:29 - trees.J48
18:34:00 - trees.RandomForest
18:42:05 - functions.MultilayerPerceptron
18:49:19 - lazy.IBk
19:03:26 - bayes.NaiveBayes
19:09:53 - trees.J48
19:15:47 - trees.RandomForest
19:25:09 - functions.MultilayerPerceptron

```

== Re-evaluation on test set ==
User supplied test set
Relation: R_data_frame-weka.filters.unsupervised.attribute.Remove-R1,3-13,18-20,22-23
Instances: unknown (yet). Reading incrementally
Attributes: 6

== Summary ==

Correctly Classified Instances	126	79.7468 %
Incorrectly Classified Instances	32	20.2532 %
Kappa statistic	0.5897	
Mean absolute error	0.1459	
Root mean squared error	0.3412	
Total Number of Instances	158	

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.761	0.172	0.783	0.761	0.771	0.590	0.859	0.846	Approve	
0.828	0.239	0.809	0.828	0.818	0.590	0.862	0.872	Disapprove	
?	0.000	?	?	?	?	?	?	Skipped	
Weighted Avg.	0.797	0.209	0.797	0.797	0.797	0.590	0.861	0.860	

== Confusion Matrix ==

a	b	c	<-- classified as
54	17	0	a = Approve
15	72	0	b = Disapprove
0	0	0	c = Skipped

Status OK Log x 0

MultiLayerPerceptron - Cross Validation (Set 3)

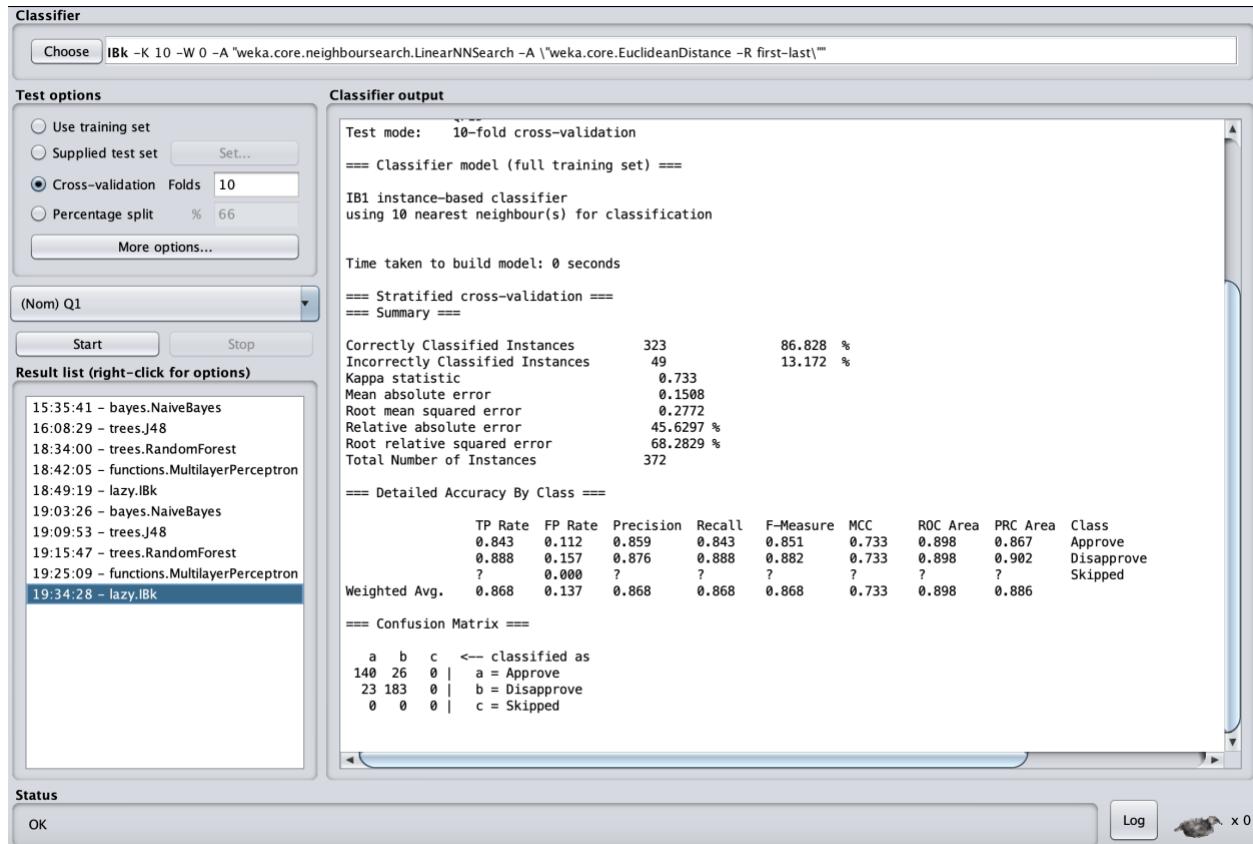
Accuracy	TP Rate	FP Rate	ROC Area	Class
79.03%	75.9%	18.4%	84.1%	Approve
	81.6%	24.1%	84.5%	Disapprove

MultiLayerPerceptron - Test set (Set 3)

Accuracy	TP Rate	FP Rate	ROC Area	Class
79.75%	76.1%	17.2%	85.9%	Approve
	82.8%	23.9%	86.2%	Disapprove

5. K-nearest Neighbor (IBK) with K = 10

- Step 1: Choose IBK and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: Select 'Supplied test set' and click 'Set' -> Open file, open "set3_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

```

15:35:41 - bayes.NaiveBayes
16:08:29 - trees.J48
18:34:00 - trees.RandomForest
18:42:05 - functions.MultilayerPerceptron
18:49:19 - lazy.IBK
19:03:26 - bayes.NaiveBayes
19:09:53 - trees.J48
19:15:47 - trees.RandomForest
19:25:09 - functions.MultilayerPerceptron
19:34:28 - lazy.IBK

```

== Re-evaluation on test set ==

User supplied test set

Relation: R_data_frame-weka.filters.unsupervised.attribute.Remove-R1,3-13,18-20,22-23

Instances: unknown (yet). Reading incrementally

Attributes: 6

== Summary ==

	Correctly Classified Instances	133	84.1772 %
Incorrectly Classified Instances	25	15.8228 %	
Kappa statistic	0.679		
Mean absolute error	0.1631		
Root mean squared error	0.2939		
Total Number of Instances	158		

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.803	0.126	0.838	0.803	0.820	0.680	0.884	0.838	0.884	Approve
0.874	0.197	0.844	0.874	0.859	0.680	0.884	0.872	0.872	Disapprove
?	0.000	?	?	?	?	?	?	?	Skipped
Weighted Avg.	0.842	0.165	0.842	0.842	0.841	0.680	0.884	0.857	

== Confusion Matrix ==

a	b	c	<-- classified as
57	14	0	a = Approve
11	76	0	b = Disapprove
0	0	0	c = Skipped

Status OK Log x 0

IBK - Cross Validation (Set 3)

Accuracy	TP Rate	FP Rate	ROC Area	Class
86.83%	84.3%	11.2%	89.8%	Approve
	88.8%	15.7%	89.8%	Disapprove

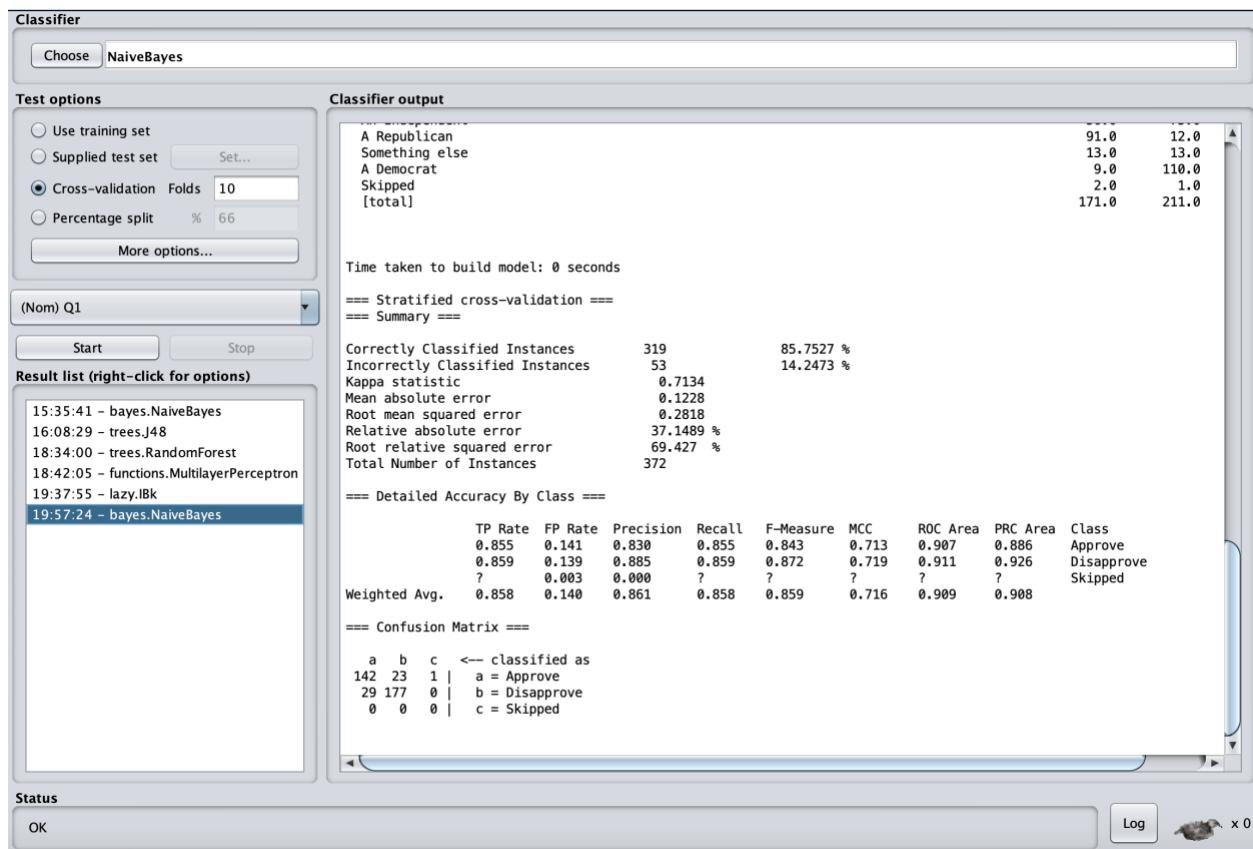
IBK - Test set (Set 3)

Accuracy	TP Rate	FP Rate	ROC Area	Class
84.18%	80.3%	12.6%	88.4%	Approve
	87.4%	19.7%	88.4%	Disapprove

d. Attribute set 4: [Q3, QPID, ppethm, Q2, pphouse, ppmsacat, ppmarit, Q4, Q1] - by CorrelationAttributeEval

1. Naive Bayes

- Step 1: Open “Initial_Training_set.arff”, remove all attributes except Q3, QPID, ppethm, Q2, pphouse, ppmsacat, ppmarit, Q4, and Q1. Save it as “set4_training”
- Step 2: Open “Initial_Testing_set.arff”, remove all attributes except Q3, QPID, ppethm, Q2, pphouse, ppmsacat, ppmarit, Q4, and Q1. Save it as “set4_testing”
- Step 3: Open “set4_training”, Classify -> Choose NaiveBayes with Cross-validation (10 Folds), Q1 as class attribute -> Start



- Step 4: In order to test our model, we re-evaluate our model with the test dataset. Select ‘Supplied test set’ and click ‘Set’ -> Open file, open “set4_testing” with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Percentage split

More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

- 15:35:41 - bayes.NaiveBayes
- 16:08:29 - trees.J48
- 18:34:00 - trees.RandomForest
- 18:42:05 - functions.MultilayerPerceptron
- 19:37:55 - lazy.IBK
- 19:57:24 - bayes.NaiveBayes**

```

==== Re-evaluation on test set ====
User supplied test set
Relation: R_data_frame-weka.filters.unsupervised.attribute.Remove-R1-5,7,9,12-14,19-20,22-23
Instances: unknown (yet). Reading incrementally
Attributes: 9

==== Summary ====
Correctly Classified Instances      133          84.1772 %
Incorrectly Classified Instances   25           15.8228 %
Kappa statistic                   0.6815
Mean absolute error               0.1309
Root mean squared error          0.2918
Total Number of Instances        158

==== Detailed Accuracy By Class ====
      TP Rate   FP Rate   Precision   Recall   F-Measure   MCC   ROC Area   PRC Area   Class
      0.845     0.161     0.811     0.845     0.828     0.682     0.900     0.847   Approve
      0.839     0.155     0.869     0.839     0.854     0.682     0.900     0.926   Disapprove
      ?         0.000     ?         ?         ?         ?         ?         ?       Skipped
Weighted Avg.      0.842     0.158     0.843     0.842     0.842     0.682     0.900     0.890

==== Confusion Matrix ====
a  b  c  <-- classified as
60 11  0 | a = Approve
14 73  0 | b = Disapprove
 0  0  0 | c = Skipped

```

Status OK Log x 0

Naive Bayes - Cross Validation (Set 4)

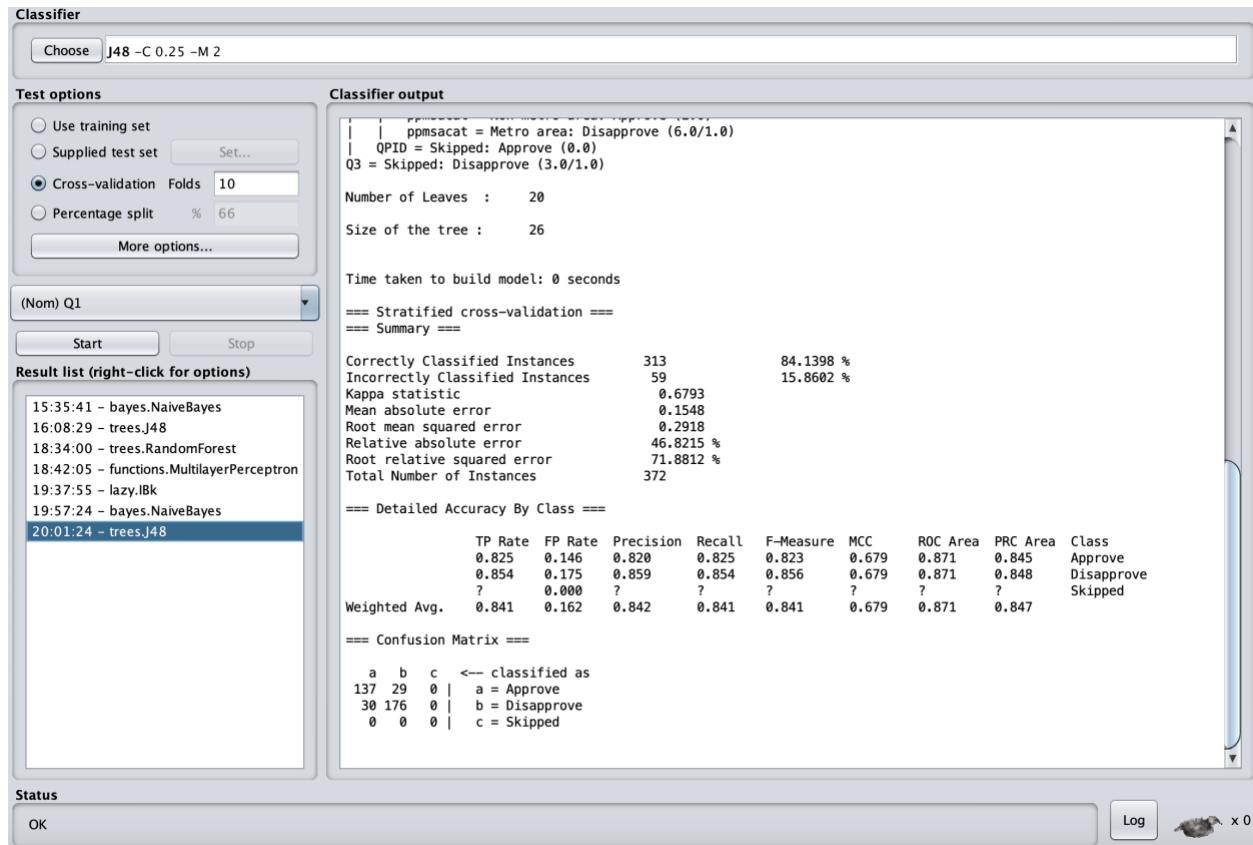
Accuracy	TP Rate	FP Rate	ROC Area	Class
85.75%	85.5%	14.1%	90.7%	Approve
	85.9%	13.9%	91.1%	Disapprove

Naive Bayes - Test set (Set 4)

Accuracy	TP Rate	FP Rate	ROC Area	Class
84.18%	84.5%	16.1%	90.0%	Approve
	83.9%	15.5%	90.0%	Disapprove

2. J48

- Step 1: First to assess how good J48 is, use Cross-validation to build our model. Choose J48 and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set4_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

```

15:35:41 - bayes.NaiveBayes
16:08:29 - trees.J48
18:34:00 - trees.RandomForest
18:42:05 - functions.MultilayerPerceptron
19:37:55 - lazy.IBk
19:57:24 - bayes.NaiveBayes
20:01:24 - trees.J48
20:01:55 - trees.J48

```

== Re-evaluation on test set ==

User supplied test set
Relation: R_data_frame-weka.filters.unsupervised.attribute.Remove-R1-5,7,9,12-14,19-20,22-23
Instances: unknown (yet). Reading incrementally
Attributes: 9

== Summary ==

	Correctly Classified Instances	132	83.5443 %
Incorrectly Classified Instances	26	16.4557 %	
Kappa statistic	0.6649		
Mean absolute error	0.1509		
Root mean squared error	0.2981		
Total Number of Instances	158		

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.775	0.115	0.846	0.775	0.809	0.667	0.865	0.810	Approve	
0.885	0.225	0.828	0.885	0.856	0.667	0.865	0.862	Disapprove	
?	0.000	?	?	?	?	?	?	Skipped	
Weighted Avg.	0.835	0.176	0.836	0.835	0.835	0.667	0.865	0.838	

== Confusion Matrix ==

a	b	c	<-- classified as
55	16	0	a = Approve
10	77	0	b = Disapprove
0	0	0	c = Skipped

Status OK Log x 0

J48 - Cross Validation (Set 4)

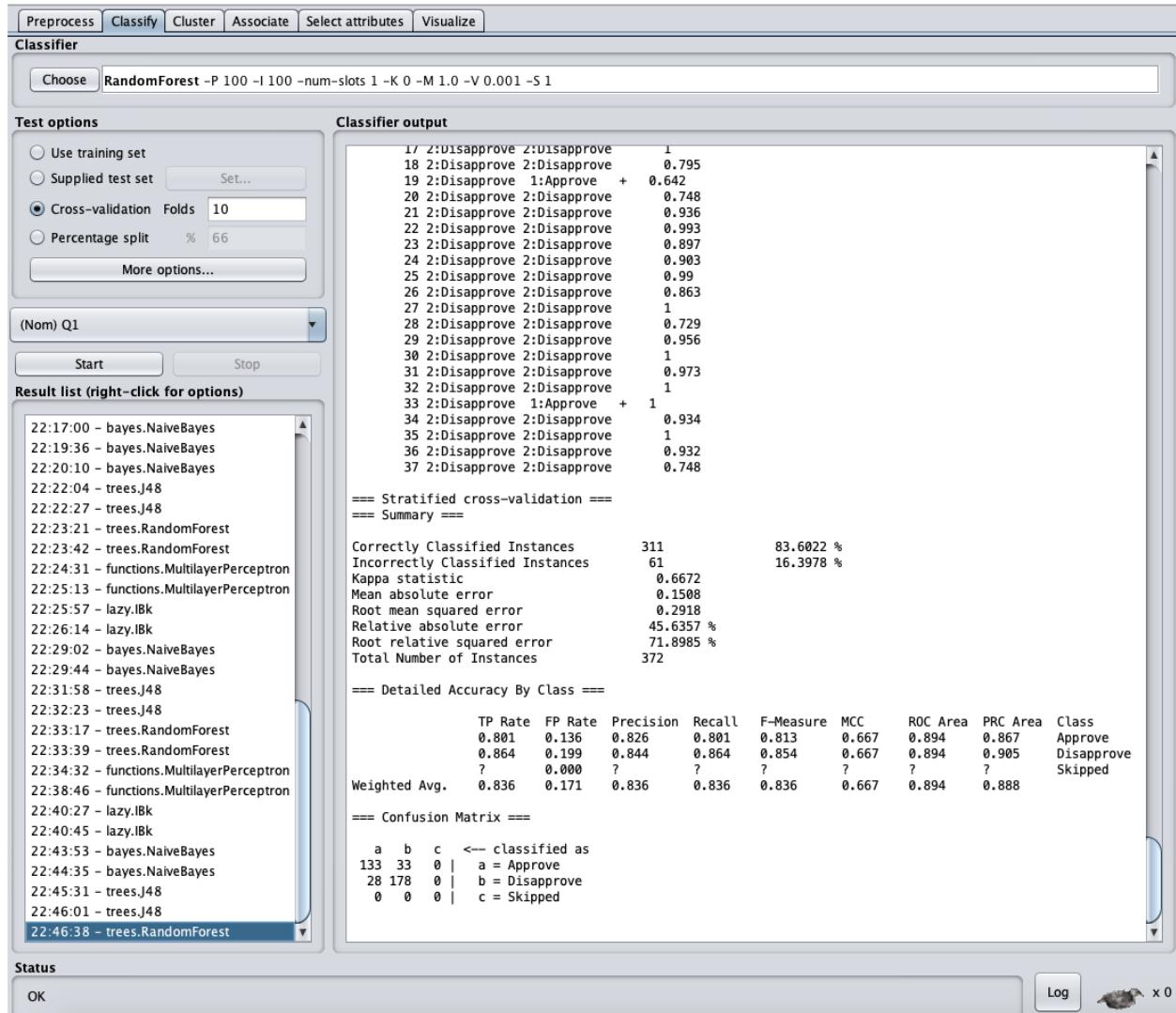
Accuracy	TP Rate	FP Rate	ROC Area	Class
84.14%	82.5%	14.6%	87.1%	Approve
	85.4%	17.5%	87.1%	Disapprove

J48 - Test set (Set 4)

Accuracy	TP Rate	FP Rate	ROC Area	Class
83.54%	77.5%	11.5%	86.5%	Approve
	88.5%	22.5%	86.5%	Disapprove

3. Random Forest

- Step 1: Choose RandomForest and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set4_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose RandomForest -P 100 -l 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1

Test options

- Use training set
- Supplied test set Set...
- Cross-validation Folds 10
- Percentage split % 66
- [More options...](#)

(Nom) Q1

Start Stop

Result list (right-click for options)

```

22:19:36 - bayes.NaiveBayes
22:20:10 - bayes.NaiveBayes
22:22:04 - trees.J48
22:22:27 - trees.J48
22:23:21 - trees.RandomForest
22:23:42 - trees.RandomForest
22:24:31 - functions.MultilayerPerceptron
22:25:13 - functions.MultilayerPerceptron
22:25:57 - lazy.IBk
22:26:14 - lazy.IBk
22:29:02 - bayes.NaiveBayes
22:29:44 - bayes.NaiveBayes
22:31:58 - trees.J48
22:32:23 - trees.J48
22:33:17 - trees.RandomForest
22:33:39 - trees.RandomForest
22:34:32 - functions.MultilayerPerceptron
22:38:46 - functions.MultilayerPerceptron
22:40:27 - lazy.IBk
22:40:45 - lazy.IBk
22:43:53 - bayes.NaiveBayes
22:44:35 - bayes.NaiveBayes
22:45:31 - trees.J48
22:46:01 - trees.J48
22:46:38 - trees.RandomForest
22:48:00 - trees.RandomForest

```

Classifier output

```

141 2:Disapprove 2:Disapprove 0.841
142 2:Disapprove 2:Disapprove 0.898
143 2:Disapprove 2:Disapprove 1
144 2:Disapprove 2:Disapprove 0.924
145 2:Disapprove 2:Disapprove 1
146 2:Disapprove 2:Disapprove 0.972
147 2:Disapprove 2:Disapprove 0.975
148 2:Disapprove 2:Disapprove 0.975
149 2:Disapprove 2:Disapprove 0.776
150 2:Disapprove 2:Disapprove 0.75
151 2:Disapprove 2:Disapprove 0.923
152 2:Disapprove 2:Disapprove 0.658
153 2:Disapprove 1:Approve + 0.587
154 2:Disapprove 2:Disapprove 0.998
155 2:Disapprove 2:Disapprove 0.808
156 2:Disapprove 1:Approve + 0.968
157 2:Disapprove 2:Disapprove 0.982
158 2:Disapprove 2:Disapprove 0.946

```

== Evaluation on test set ==

Time taken to test model on supplied test set: 0.01 seconds

== Summary ==

Correctly Classified Instances	131	82.9114 %
Incorrectly Classified Instances	27	17.0886 %
Kappa statistic	0.6524	
Mean absolute error	0.1625	
Root mean squared error	0.3001	
Relative absolute error	49.1563 %	
Root relative squared error	73.8941 %	
Total Number of Instances	158	

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.775	0.126	0.833	0.775	0.803	0.654	0.884	0.849	0.884	Approve
0.874	0.225	0.826	0.874	0.849	0.654	0.884	0.901	0.884	Disapprove
?	0.000	?	?	?	?	?	?	?	Skipped
Weighted Avg.	0.829	0.181	0.829	0.829	0.654	0.884	0.878	0.884	

== Confusion Matrix ==

a	b	c	<-- classified as
55	16	0	a = Approve
11	76	0	b = Disapprove
0	0	0	c = Skipped

Status OK Log x 0

Random Forest - Cross Validation (Set 4)

Accuracy	TP Rate	FP Rate	ROC Area	Class
83.60%	80.1%	13.6%	89.4%	Approve
	86.4%	19.9%	89.4%	Disapprove

Random Forest - Test set (Set 4)

Accuracy	TP Rate	FP Rate	ROC Area	Class
82.91%	77.5%	12.6%	88.4%	Approve
	87.4%	22.5%	88.4%	Disapprove

4. MultiLayerPerceptron

- Step 1: Choose MultilayerPerceptron and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start

Classifier output

```

18 2:Disapprove 1:Approve + 0.89/
19 2:Disapprove 1:Approve + 0.994
20 2:Disapprove 2:Disapprove 0.791
21 2:Disapprove 2:Disapprove 1
22 2:Disapprove 2:Disapprove 1
23 2:Disapprove 2:Disapprove 0.971
24 2:Disapprove 2:Disapprove 0.913
25 2:Disapprove 2:Disapprove 0.983
26 2:Disapprove 2:Disapprove 1
27 2:Disapprove 2:Disapprove 1
28 2:Disapprove 2:Disapprove 0.765
29 2:Disapprove 2:Disapprove 0.979
30 2:Disapprove 2:Disapprove 1
31 2:Disapprove 2:Disapprove 1
32 2:Disapprove 2:Disapprove 0.98
33 2:Disapprove 1:Approve + 0.994
34 2:Disapprove 2:Disapprove 1
35 2:Disapprove 2:Disapprove 1
36 2:Disapprove 2:Disapprove 0.997
37 2:Disapprove 2:Disapprove 0.791

== Stratified cross-validation ===
== Summary ===

Correctly Classified Instances      295           79.3011 %
Incorrectly Classified Instances    77            20.6989 %
Kappa statistic                   0.5809
Mean absolute error               0.15
Root mean squared error          0.3399
Relative absolute error          45.3697 %
Root relative squared error     83.7297 %
Total Number of Instances        372

== Detailed Accuracy By Class ==

      TP Rate   FP Rate   Precision   Recall   F-Measure   MCC   ROC Area   PRC Area   Class
0.765     0.184     0.770     0.765     0.767     0.581   0.850     0.812     Approve
0.816     0.235     0.812     0.816     0.814     0.581   0.850     0.857     Disapprove
?         0.000     ?         ?         ?         ?         ?         ?         Skipped
Weighted Avg. 0.793     0.212     0.793     0.793     0.793     0.581   0.850     0.837

== Confusion Matrix ==

a   b   c   <-- classified as
127 39  0   |   a = Approve
38 168 0   |   b = Disapprove
0   0   0   |   c = Skipped

```

- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set4_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose MultilayerPerceptron -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a

Test options

- Use training set
- Supplied test set Set...
- Cross-validation Folds 10
- Percentage split % 66
- More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

```

22:22:04 - trees.J48
22:22:27 - trees.J48
22:23:21 - trees.RandomForest
22:23:42 - trees.RandomForest
22:24:31 - functions.MultilayerPerceptron
22:25:13 - functions.MultilayerPerceptron
22:25:57 - lazy.IBk
22:26:14 - lazy.IBk
22:29:02 - bayes.NaiveBayes
22:29:44 - bayes.NaiveBayes
22:31:58 - trees.J48
22:32:23 - trees.J48
22:33:17 - trees.RandomForest
22:33:39 - trees.RandomForest
22:34:32 - functions.MultilayerPerceptron
22:38:46 - functions.MultilayerPerceptron
22:40:27 - lazy.IBk
22:40:45 - lazy.IBk
22:43:53 - bayes.NaiveBayes
22:44:35 - bayes.NaiveBayes
22:45:31 - trees.J48
22:46:01 - trees.J48
22:46:38 - trees.RandomForest
22:48:00 - trees.RandomForest
22:49:52 - functions.MultilayerPerceptron
22:50:50 - functions.MultilayerPerceptron
  
```

Classifier output

```

142 2:Disapprove 2:Disapprove 0.996
143 2:Disapprove 2:Disapprove 1
144 2:Disapprove 2:Disapprove 1
145 2:Disapprove 2:Disapprove 1
146 2:Disapprove 2:Disapprove 1
147 2:Disapprove 2:Disapprove 0.999
148 2:Disapprove 2:Disapprove 0.999
149 2:Disapprove 2:Disapprove 0.999
150 2:Disapprove 2:Disapprove 0.963
151 2:Disapprove 1:Approve + 0.977
152 2:Disapprove 2:Disapprove 0.901
153 2:Disapprove 2:Disapprove 0.96
154 2:Disapprove 2:Disapprove 1
155 2:Disapprove 2:Disapprove 0.882
156 2:Disapprove 1:Approve + 1
157 2:Disapprove 2:Disapprove 0.999
158 2:Disapprove 2:Disapprove 0.998
  
```

== Evaluation on test set ==

Time taken to test model on supplied test set: 0.01 seconds

== Summary ==

Correctly Classified Instances	130	82.2785 %
Incorrectly Classified Instances	28	17.7215 %
Kappa statistic	0.641	
Mean absolute error	0.1351	
Root mean squared error	0.3276	
Relative absolute error	40.8479 %	
Root relative squared error	80.668 %	
Total Number of Instances	158	

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.789	0.149	0.812	0.789	0.800	0.641	0.865	0.806	0.806	Approve
0.851	0.211	0.831	0.851	0.841	0.641	0.867	0.859	0.859	Disapprove
?	0.000	?	?	?	?	?	?	?	Skipped
Weighted Avg.	0.823	0.183	0.823	0.823	0.641	0.866	0.835	0.835	

== Confusion Matrix ==

a	b	c	<-- classified as
56	15	0	a = Approve
13	74	0	b = Disapprove
0	0	0	c = Skipped

Status

OK Log x 0

MultiLayerPerceptron - Cross Validation (Set 4)

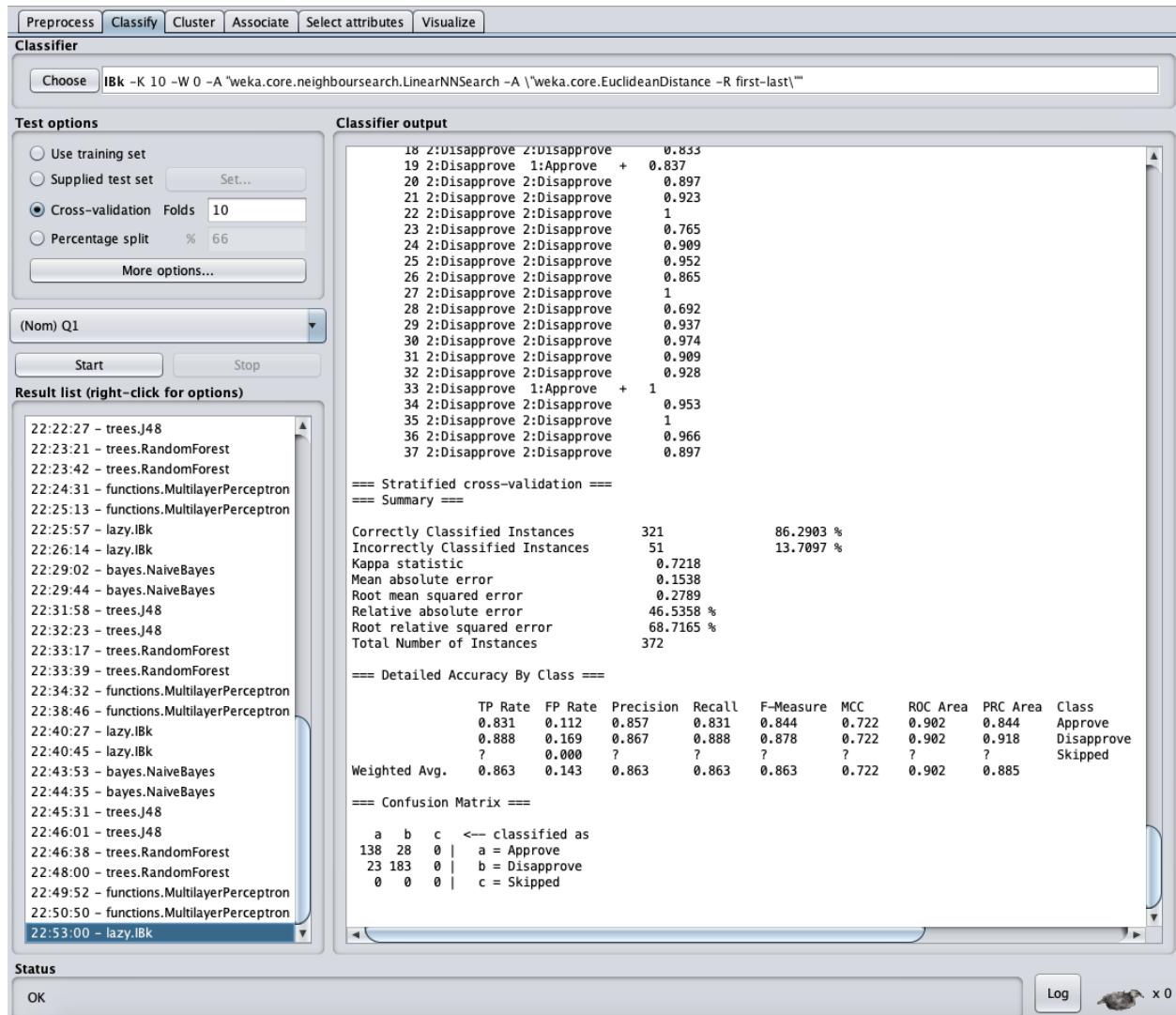
Accuracy	TP Rate	FP Rate	ROC Area	Class
79.30%	76.5%	18.4%	85.0%	Approve
	81.6%	23.5%	85.0%	Disapprove

MultiLayerPerceptron - Test set (Set 4)

Accuracy	TP Rate	FP Rate	ROC Area	Class
82.28%	78.9%	14.9%	86.5%	Approve
	85.1%	21.1%	86.7%	Disapprove

5. K-nearest Neighbor (IBK) with K = 10

- Step 1: Choose IBK and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set4_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose IBk -K 10 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A "weka.core.EuclideanDistance -R first-last"

Test options

- Use training set
- Supplied test set Set...
- Cross-validation Folds 10
- Percentage split % 66
- More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

```

22:23:21 - trees.RandomForest
22:23:42 - trees.RandomForest
22:24:31 - functions.MultilayerPerceptron
22:25:13 - functions.MultilayerPerceptron
22:25:57 - lazy.IBk
22:26:14 - lazy.IBk
22:29:02 - bayes.NaiveBayes
22:29:44 - bayes.NaiveBayes
22:31:58 - trees.J48
22:32:23 - trees.J48
22:33:17 - trees.RandomForest
22:33:39 - trees.RandomForest
22:34:32 - functions.MultilayerPerceptron
22:38:46 - functions.MultilayerPerceptron
22:40:27 - lazy.IBk
22:40:45 - lazy.IBk
22:43:53 - bayes.NaiveBayes
22:44:35 - bayes.NaiveBayes
22:45:31 - trees.J48
22:46:01 - trees.J48
22:46:38 - trees.RandomForest
22:48:00 - trees.RandomForest
22:49:52 - functions.MultilayerPerceptron
22:50:50 - functions.MultilayerPerceptron
22:53:00 - lazy.IBk
22:53:52 - lazy.IBk

```

Classifier output

```

142 2:Disapprove 2:Disapprove 0.875
143 2:Disapprove 2:Disapprove 1
144 2:Disapprove 2:Disapprove 1
145 2:Disapprove 2:Disapprove 1
146 2:Disapprove 2:Disapprove 1
147 2:Disapprove 2:Disapprove 0.905
148 2:Disapprove 2:Disapprove 0.947
149 2:Disapprove 2:Disapprove 0.666
150 2:Disapprove 2:Disapprove 0.677
151 2:Disapprove 2:Disapprove 0.811
152 2:Disapprove 2:Disapprove 0.769
153 2:Disapprove 2:Disapprove 0.682
154 2:Disapprove 2:Disapprove 1
155 2:Disapprove 2:Disapprove 0.791
156 2:Disapprove 1:Approve + 0.937
157 2:Disapprove 2:Disapprove 0.894
158 2:Disapprove 2:Disapprove 0.889

```

== Evaluation on test set ==

Time taken to test model on supplied test set: 0.02 seconds

== Summary ==

Correctly Classified Instances	134	84.8101 %
Incorrectly Classified Instances	24	15.1899 %
Kappa statistic	0.6923	
Mean absolute error	0.1611	
Root mean squared error	0.2945	
Relative absolute error	48.7257 %	
Root relative squared error	72.5014 %	
Total Number of Instances	158	

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.817	0.126	0.841	0.817	0.829	0.692	0.886	0.863	0.891	Approve
0.874	0.183	0.854	0.874	0.864	0.692	0.886	0.891	0.886	Disapprove
?	0.000	?	?	?	?	?	?	?	Skipped
Weighted Avg.	0.848	0.158	0.848	0.848	0.692	0.886	0.878	0.886	

== Confusion Matrix ==

a	b	c	<-- classified as
58	13	0	a = Approve
11	76	0	b = Disapprove
0	0	0	c = Skipped

Status OK Log x 0

IBK - Cross Validation (Set 4)

Accuracy	TP Rate	FP Rate	ROC Area	Class
86.29%	83.1%	11.2%	90.2%	Approve
	88.8%	16.9%	90.2%	Disapprove

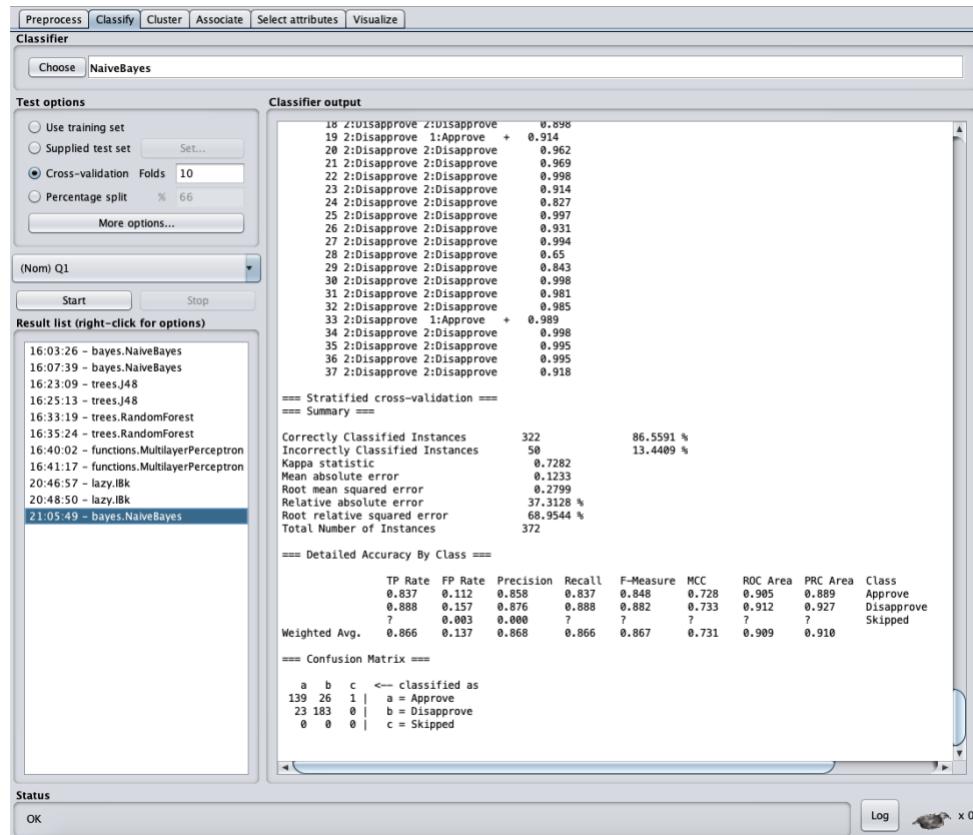
IBK - Test set (Set 4)

Accuracy	TP Rate	FP Rate	ROC Area	Class
84.81%	81.7%	12.6%	88.6%	Approve
	87.4%	18.3%	88.6%	Disapprove

e. Attribute set 5: [Q3, QPID, Q2, Q4, Q5a, Q5b, ppethm, ppmarit, ppeducat, ppgender, ABCAGE, Q1] - by us

1. Naive Bayes:

- Step 1: Open “Initial_Training_set.arff”, remove all attributes except Q3, QPID, Q2, Q4, Q5a, Q5b, ppethm, ppmarit, ppeducat, ppgender, ABCAGE and Q1. Save it as “set5_training”
- Step 2: Open “Initial_Testing_set.arff”, remove all attributes except Q3, QPID, Q2, Q4, Q5a, Q5b, ppethm, ppmarit, ppeducat, ppgender, ABCAGE and Q1. Save it as “set4_testing”
- Step 3: Open “set5_training”, Classify -> Choose NaiveBayes with Cross-validation (10 Folds), Q1 as class attribute -> Start



- Step 4: In order to test our model, we re-evaluate our model with the test dataset. Select ‘Supplied test set’ and click ‘Set’ -> Open file, open “set5_testing” with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose NaiveBayes

Test options

- Use training set
- Supplied test set
- Cross-validation Folds 10
- Percentage split % 66
-

(Nom) Q1

Result list (right-click for options)

```

16:03:26 - bayes.NaiveBayes
16:07:39 - bayes.NaiveBayes
16:23:09 - trees.J48
16:25:13 - trees.J48
16:33:19 - trees.RandomForest
16:35:24 - trees.RandomForest
16:40:02 - functions.MultilayerPerceptron
16:41:17 - functions.MultilayerPerceptron
20:46:57 - lazy.IBk
20:48:50 - lazy.IBk
21:05:49 - bayes.NaiveBayes
21:09:34 - bayes.NaiveBayes
  
```

Classifier output

```

142 2:Disapprove 2:Disapprove 0.994
143 2:Disapprove 2:Disapprove 0.999
144 2:Disapprove 2:Disapprove 0.981
145 2:Disapprove 2:Disapprove 1
146 2:Disapprove 2:Disapprove 0.999
147 2:Disapprove 2:Disapprove 0.992
148 2:Disapprove 2:Disapprove 0.995
149 2:Disapprove 1:Approve + 0.722
150 2:Disapprove 2:Disapprove 0.797
151 2:Disapprove 2:Disapprove 0.947
152 2:Disapprove 2:Disapprove 0.994
153 2:Disapprove 2:Disapprove 0.915
154 2:Disapprove 2:Disapprove 0.995
155 2:Disapprove 2:Disapprove 0.969
156 2:Disapprove 1:Approve + 0.997
157 2:Disapprove 2:Disapprove 0.927
158 2:Disapprove 2:Disapprove 0.989
  
```

== Evaluation on test set ==

Time taken to test model on supplied test set: 0.01 seconds

== Summary ==

Correctly Classified Instances	134	84.8101 %
Incorrectly Classified Instances	24	15.1899 %
Kappa statistic	0.6938	
Mean absolute error	0.124	
Root mean squared error	0.2795	
Relative absolute error	37.5107 %	
Root relative squared error	68.8257 %	
Total Number of Instances	158	

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.845	0.149	0.822	0.845	0.833	0.694	0.909	0.857	0.857	Approve
0.851	0.155	0.871	0.851	0.860	0.694	0.907	0.928	0.928	Disapprove
?	0.000	?	?	?	?	?	?	?	Skipped
Weighted Avg.	0.848	0.152	0.849	0.848	0.848	0.694	0.908	0.896	

== Confusion Matrix ==

a b c	<-- classified as
60 11 0	a = Approve
13 74 0	b = Disapprove
0 0 0	c = Skipped

Status

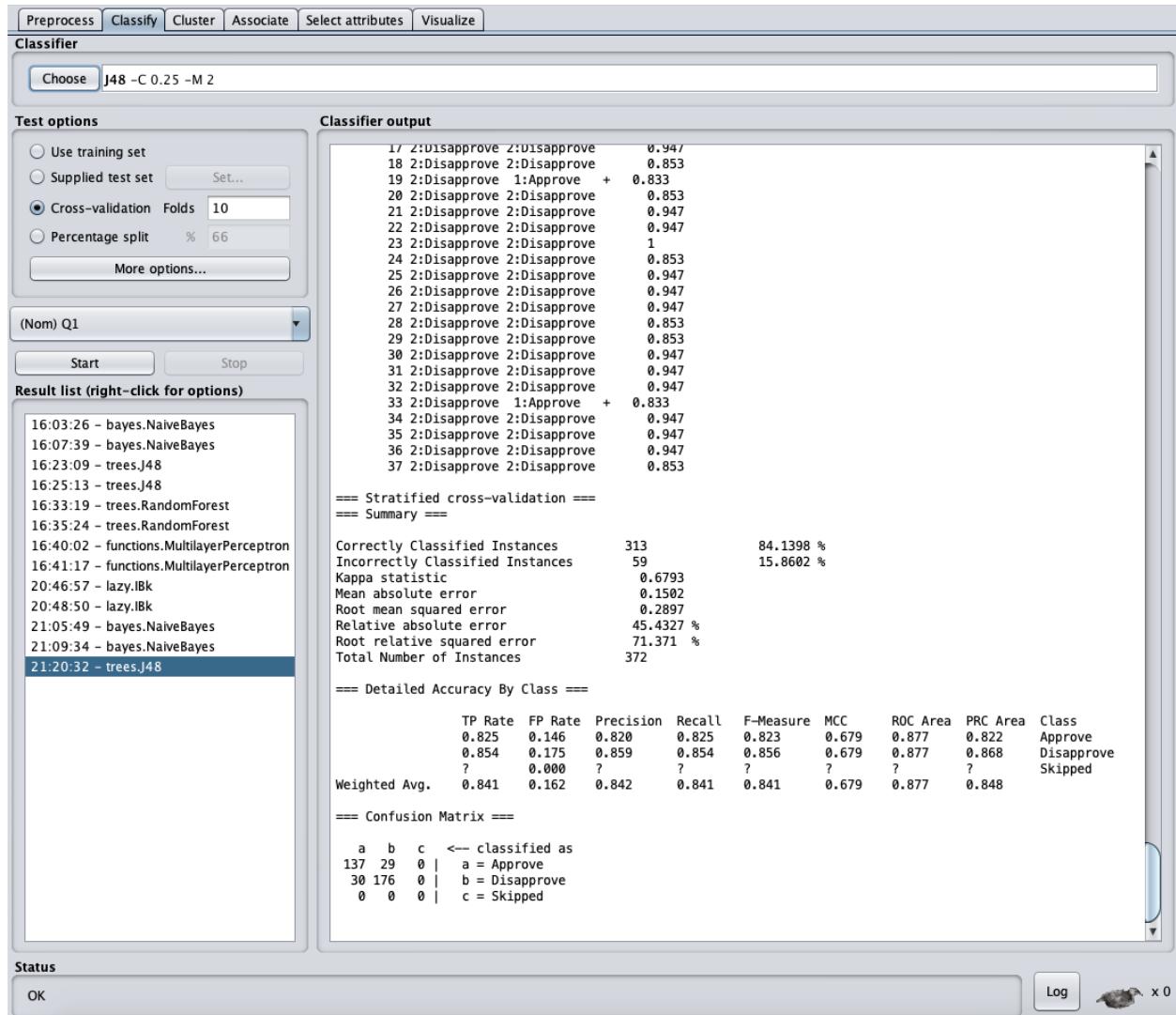
OK x 0

Naive Bayes - Cross Validation (Set 5)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
86.56%	83.7%	11.2%	90.5%	Approve
	88.8%	15.7%	91.2%	Disapprove

Naive Bayes - Test set (Set 5)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
84.81%	84.5%	14.9%	90.9%	Approve
	85.1%	15.5%	90.7%	Disapprove

2. J48

- Step 1: First to assess how good J48 is, use Cross-validation to build our model. Choose J48 and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set5_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose J48 - C 0.25 - M 2

Test options

- Use training set
- Supplied test set
- Cross-validation Folds 10
- Percentage split % 66
-

(Nom) Q1

Result list (right-click for options)

```

16:03:26 - bayes.NaiveBayes
16:07:39 - bayes.NaiveBayes
16:23:09 - trees.J48
16:25:13 - trees.J48
16:33:19 - trees.RandomForest
16:35:24 - trees.RandomForest
16:40:02 - functions.MultilayerPerceptron
16:41:17 - functions.MultilayerPerceptron
20:46:57 - lazy.IBk
20:48:50 - lazy.IBk
21:05:49 - bayes.NaiveBayes
21:09:34 - bayes.NaiveBayes
21:20:32 - trees.J48
21:21:57 - trees.J48

```

Classifier output

```

141 2:Disapprove 1:Approve + 0.75
142 2:Disapprove 2:Disapprove 0.831
143 2:Disapprove 2:Disapprove 0.953
144 2:Disapprove 2:Disapprove 0.831
145 2:Disapprove 2:Disapprove 0.953
146 2:Disapprove 2:Disapprove 0.953
147 2:Disapprove 2:Disapprove 0.953
148 2:Disapprove 2:Disapprove 0.953
149 2:Disapprove 2:Disapprove 0.831
150 2:Disapprove 2:Disapprove 0.831
151 2:Disapprove 2:Disapprove 1
152 2:Disapprove 2:Disapprove 0.831
153 2:Disapprove 2:Disapprove 0.667
154 2:Disapprove 2:Disapprove 0.953
155 2:Disapprove 2:Disapprove 0.953
156 2:Disapprove 1:Approve + 0.955
157 2:Disapprove 2:Disapprove 0.953
158 2:Disapprove 2:Disapprove 0.831

```

== Evaluation on test set ==

Time taken to test model on supplied test set: 0.01 seconds

== Summary ==

Correctly Classified Instances	131	82.9114 %
Incorrectly Classified Instances	27	17.0886 %
Kappa statistic	0.6542	
Mean absolute error	0.1505	
Root mean squared error	0.2964	
Relative absolute error	45.5257 %	
Root relative squared error	72.9652 %	
Total Number of Instances	158	

== Detailed Accuracy By Class ==

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.803	0.149	0.814	0.803	0.809	0.654	0.882	0.830	Approve
0.851	0.197	0.841	0.851	0.846	0.654	0.882	0.873	Disapprove
?	0.000	?	?	?	?	?	?	Skipped
Weighted Avg.	0.829	0.176	0.829	0.829	0.654	0.882	0.853	

== Confusion Matrix ==

```

a b c <-- classified as
57 14 0 | a = Approve
13 74 0 | b = Disapprove
0 0 0 | c = Skipped

```

Status

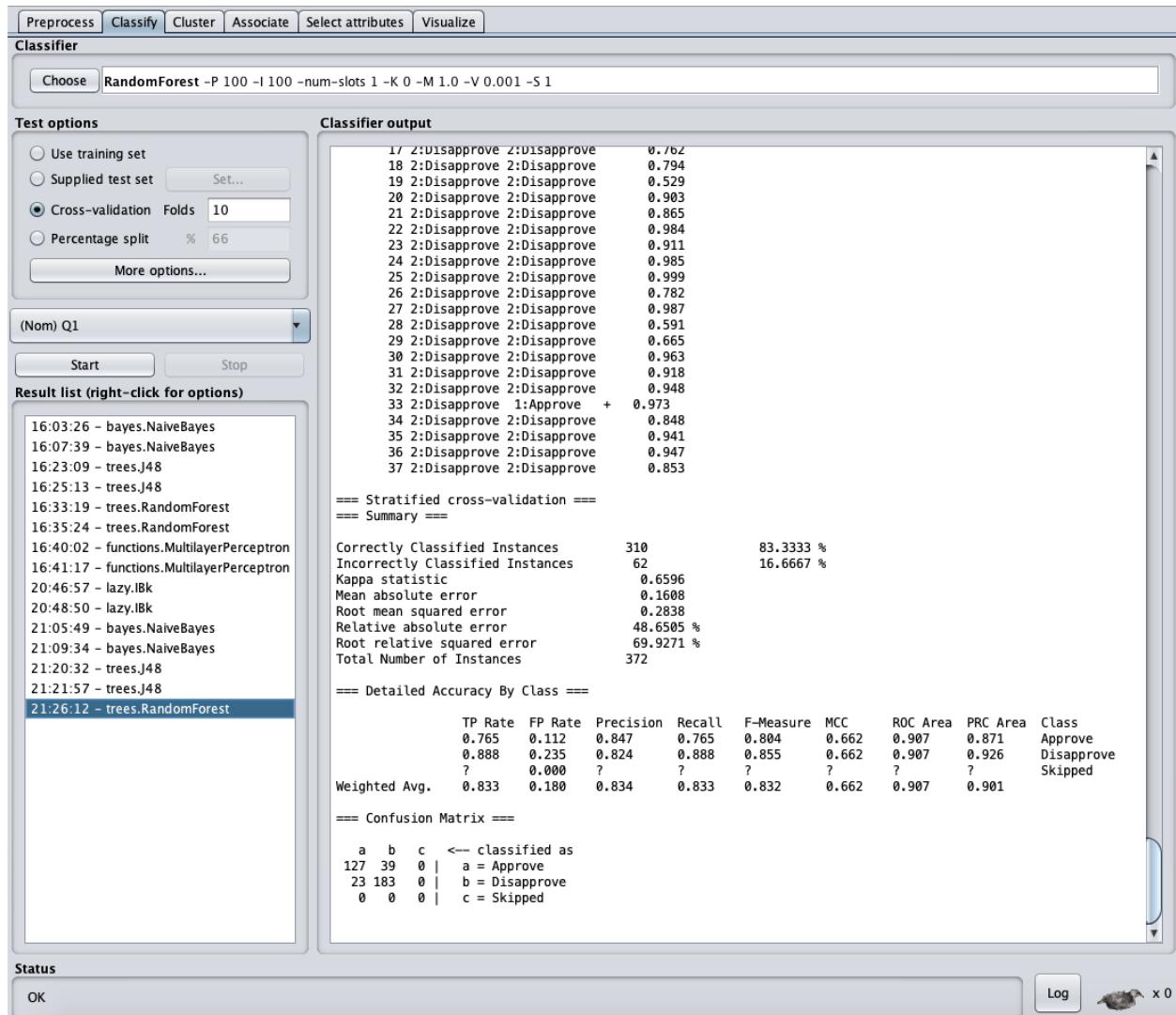
OK x 0

J48 - Cross Validation (Set 5)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
84.14%	82.5%	14.6%	87.7%	Approve
	85.4%	17.5%	87.7%	Disapprove

J48 - Test set (Set 5)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
82.91%	80.3%	14.9%	88.2%	Approve
	85.1%	19.7%	88.2%	Disapprove

3. Random Forest

- Step 1: Choose RandomForest and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set5_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Classifier

Choose: RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1

Test options		Classifier output																																																			
<input type="radio"/> Use training set		141 2:Disapprove 2:Disapprove 0.77																																																			
<input checked="" type="radio"/> Supplied test set	Set...	142 2:Disapprove 2:Disapprove 0.854																																																			
<input type="radio"/> Cross-validation	Folds 10	143 2:Disapprove 2:Disapprove 0.991																																																			
<input type="radio"/> Percentage split	% 66	144 2:Disapprove 2:Disapprove 0.88																																																			
More options...		145 2:Disapprove 2:Disapprove 0.996																																																			
		146 2:Disapprove 2:Disapprove 0.977																																																			
		147 2:Disapprove 2:Disapprove 0.977																																																			
		148 2:Disapprove 2:Disapprove 0.946																																																			
		149 2:Disapprove 2:Disapprove 0.697																																																			
		150 2:Disapprove 2:Disapprove 0.792																																																			
		151 2:Disapprove 2:Disapprove 0.859																																																			
		152 2:Disapprove 2:Disapprove 0.848																																																			
		153 2:Disapprove 1:Approve + 0.624																																																			
		154 2:Disapprove 2:Disapprove 1																																																			
		155 2:Disapprove 2:Disapprove 0.962																																																			
		156 2:Disapprove 1:Approve + 0.773																																																			
		157 2:Disapprove 2:Disapprove 0.851																																																			
		158 2:Disapprove 2:Disapprove 0.722																																																			
==== Evaluation on test set ====																																																					
Time taken to test model on supplied test set: 0.02 seconds																																																					
==== Summary ====																																																					
<table border="1"> <thead> <tr> <th>Correctly Classified Instances</th> <th>132</th> <th>83.5443 %</th> </tr> </thead> <tbody> <tr> <td>Incorrectly Classified Instances</td> <td>26</td> <td>16.4557 %</td> </tr> <tr> <td>Kappa statistic</td> <td>0.6675</td> <td></td> </tr> <tr> <td>Mean absolute error</td> <td>0.164</td> <td></td> </tr> <tr> <td>Root mean squared error</td> <td>0.2845</td> <td></td> </tr> <tr> <td>Relative absolute error</td> <td>49.6163 %</td> <td></td> </tr> <tr> <td>Root relative squared error</td> <td>70.0486 %</td> <td></td> </tr> <tr> <td>Total Number of Instances</td> <td>158</td> <td></td> </tr> </tbody> </table>				Correctly Classified Instances	132	83.5443 %	Incorrectly Classified Instances	26	16.4557 %	Kappa statistic	0.6675		Mean absolute error	0.164		Root mean squared error	0.2845		Relative absolute error	49.6163 %		Root relative squared error	70.0486 %		Total Number of Instances	158																											
Correctly Classified Instances	132	83.5443 %																																																			
Incorrectly Classified Instances	26	16.4557 %																																																			
Kappa statistic	0.6675																																																				
Mean absolute error	0.164																																																				
Root mean squared error	0.2845																																																				
Relative absolute error	49.6163 %																																																				
Root relative squared error	70.0486 %																																																				
Total Number of Instances	158																																																				
==== Detailed Accuracy By Class ====																																																					
<table border="1"> <thead> <tr> <th></th> <th>TP Rate</th> <th>FP Rate</th> <th>Precision</th> <th>Recall</th> <th>F-Measure</th> <th>MCC</th> <th>ROC Area</th> <th>PRC Area</th> <th>Class</th> </tr> </thead> <tbody> <tr> <td>0.817</td> <td>0.149</td> <td>0.817</td> <td>0.817</td> <td>0.817</td> <td>0.817</td> <td>0.667</td> <td>0.901</td> <td>0.883</td> <td>Approve</td> </tr> <tr> <td>0.851</td> <td>0.183</td> <td>0.851</td> <td>0.851</td> <td>0.851</td> <td>0.851</td> <td>0.667</td> <td>0.901</td> <td>0.910</td> <td>Disapprove</td> </tr> <tr> <td>?</td> <td>0.000</td> <td>?</td> <td>?</td> <td>?</td> <td>?</td> <td>?</td> <td>?</td> <td>?</td> <td>Skipped</td> </tr> <tr> <td>Weighted Avg.</td> <td>0.835</td> <td>0.168</td> <td>0.835</td> <td>0.835</td> <td>0.835</td> <td>0.667</td> <td>0.901</td> <td>0.898</td> <td></td> </tr> </tbody> </table>					TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class	0.817	0.149	0.817	0.817	0.817	0.817	0.667	0.901	0.883	Approve	0.851	0.183	0.851	0.851	0.851	0.851	0.667	0.901	0.910	Disapprove	?	0.000	?	?	?	?	?	?	?	Skipped	Weighted Avg.	0.835	0.168	0.835	0.835	0.835	0.667	0.901	0.898	
	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class																																												
0.817	0.149	0.817	0.817	0.817	0.817	0.667	0.901	0.883	Approve																																												
0.851	0.183	0.851	0.851	0.851	0.851	0.667	0.901	0.910	Disapprove																																												
?	0.000	?	?	?	?	?	?	?	Skipped																																												
Weighted Avg.	0.835	0.168	0.835	0.835	0.835	0.667	0.901	0.898																																													
==== Confusion Matrix ====																																																					
<table border="1"> <thead> <tr> <th>a b c</th> <th><-- classified as</th> </tr> </thead> <tbody> <tr> <td>58 13 0</td> <td> a = Approve</td> </tr> <tr> <td>13 74 0</td> <td> b = Disapprove</td> </tr> <tr> <td>0 0 0</td> <td> c = Skipped</td> </tr> </tbody> </table>				a b c	<-- classified as	58 13 0	a = Approve	13 74 0	b = Disapprove	0 0 0	c = Skipped																																										
a b c	<-- classified as																																																				
58 13 0	a = Approve																																																				
13 74 0	b = Disapprove																																																				
0 0 0	c = Skipped																																																				

Status: OK

RandomForest - Cross Validation (Set 5)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
83.33%	76.5%	11.2%	90.7%	Approve
	88.8%	23.5%	90.7%	Disapprove

RandomForest - Test set (Set 5)				
Accuracy	TP Rate	FP Rate	ROC Area	Class
83.54%	81.7%	14.9%	90.1%	Approve
	85.1%	18.3%	90.1%	Disapprove

4. MultiLayerPerceptron

- Step 1: Choose MultilayerPerceptron and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start

Choose MultilayerPerceptron -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a

Test options

- (Nom) Q1
- Start Stop

Result list (right-click for options)

```

16:03:26 - bayes.NaiveBayes
16:07:39 - bayes.NaiveBayes
16:23:09 - trees.J48
16:25:13 - trees.J48
16:33:19 - trees.RandomForest
16:35:24 - trees.RandomForest
16:40:02 - functions.MultilayerPerceptron
16:41:17 - functions.MultilayerPerceptron
20:46:57 - lazy.IBk
20:48:50 - lazy.IBk
21:05:49 - bayes.NaiveBayes
21:09:34 - bayes.NaiveBayes
21:20:32 - trees.J48
21:21:57 - trees.J48
21:26:12 - trees.RandomForest
21:26:54 - trees.RandomForest
21:30:32 - functions.MultilayerPerceptron
  
```

Classifier output

```

18 2:Disapprove 2:Disapprove 0.962
19 2:Disapprove 2:Disapprove 0.935
20 2:Disapprove 2:Disapprove 0.997
21 2:Disapprove 2:Disapprove 1
22 2:Disapprove 2:Disapprove 1
23 2:Disapprove 2:Disapprove 0.994
24 2:Disapprove 2:Disapprove 0.986
25 2:Disapprove 2:Disapprove 1
26 2:Disapprove 2:Disapprove 0.823
27 2:Disapprove 2:Disapprove 0.996
28 2:Disapprove 2:Disapprove 0.917
29 2:Disapprove 2:Disapprove 1
30 2:Disapprove 2:Disapprove 0.671
31 2:Disapprove 2:Disapprove 1
32 2:Disapprove 2:Disapprove 0.933
33 2:Disapprove 1:Approve + 1
34 2:Disapprove 2:Disapprove 1
35 2:Disapprove 2:Disapprove 1
36 2:Disapprove 2:Disapprove 1
37 2:Disapprove 2:Disapprove 0.918

== Stratified cross-validation ==
== Summary ==
Correctly Classified Instances 296 79.5699 %
Incorrectly Classified Instances 76 20.4301 %
Kappa statistic 0.5861
Mean absolute error 0.1386
Root mean squared error 0.3401
Relative absolute error 41.9261 %
Root relative squared error 83.7785 %
Total Number of Instances 372

== Detailed Accuracy By Class ==


|               | TP Rate | FP Rate | Precision | Recall | F-Measure | MCC   | ROC Area | PRC Area | Class      |
|---------------|---------|---------|-----------|--------|-----------|-------|----------|----------|------------|
| 0.765         | 0.180   | 0.774   | 0.765     | 0.770  | 0.586     | 0.861 | 0.805    | 0.805    | Approve    |
| 0.820         | 0.235   | 0.813   | 0.820     | 0.816  | 0.586     | 0.861 | 0.875    | 0.875    | Disapprove |
| ?             | 0.000   | ?       | ?         | ?      | ?         | ?     | ?        | ?        | Skipped    |
| Weighted Avg. | 0.796   | 0.210   | 0.795     | 0.796  | 0.796     | 0.586 | 0.861    | 0.844    |            |


== Confusion Matrix ==


|     |     |   | <-- classified as |
|-----|-----|---|-------------------|
| a   | b   | c | a = Approve       |
| 127 | 39  | 0 | a = Approve       |
| 37  | 169 | 0 | b = Disapprove    |
| 0   | 0   | 0 | c = Skipped       |


```

Status

OK Log x 0

- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set5_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose MultilayerPerceptron -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a

Test options

- Use training set
- Supplied test set Set...
- Cross-validation Folds 10
- Percentage split % 66
- [More options...](#)

(Nom) Q1

[Start](#) [Stop](#)

Result list (right-click for options)

```

16:03:26 ~ bayes.NaiveBayes
16:07:39 ~ bayes.NaiveBayes
16:23:09 ~ trees.J48
16:25:13 ~ trees.J48
16:33:19 ~ trees.RandomForest
16:35:24 ~ trees.RandomForest
16:40:02 ~ functions.MultilayerPerceptron
16:41:17 ~ functions.MultilayerPerceptron
20:46:57 ~ lazy.IBk
20:48:50 ~ lazy.IBk
21:05:49 ~ bayes.NaiveBayes
21:09:34 ~ bayes.NaiveBayes
21:20:32 ~ trees.J48
21:21:57 ~ trees.J48
21:26:12 ~ trees.RandomForest
21:26:54 ~ trees.RandomForest
21:30:32 ~ functions.MultilayerPerceptron
21:31:46 ~ functions.MultilayerPerceptron

```

Classifier output

142	2:Disapprove	2:Disapprove	0.998
143	2:Disapprove	2:Disapprove	1
144	2:Disapprove	2:Disapprove	0.75
145	2:Disapprove	2:Disapprove	1
146	2:Disapprove	2:Disapprove	1
147	2:Disapprove	2:Disapprove	0.999
148	2:Disapprove	2:Disapprove	0.996
149	2:Disapprove	2:Disapprove	0.997
150	2:Disapprove	2:Disapprove	0.993
151	2:Disapprove	2:Disapprove	1
152	2:Disapprove	2:Disapprove	1
153	2:Disapprove	1:Approve	+ 0.99
154	2:Disapprove	2:Disapprove	1
155	2:Disapprove	2:Disapprove	0.999
156	2:Disapprove	1:Approve	+ 1
157	2:Disapprove	2:Disapprove	1
158	2:Disapprove	2:Disapprove	0.997

== Evaluation on test set ==

Time taken to test model on supplied test set: 0.01 seconds

== Summary ==

Correctly Classified Instances	134	84.8101 %
Incorrectly Classified Instances	24	15.1899 %
Kappa statistic	0.6931	
Mean absolute error	0.1054	
Root mean squared error	0.3005	
Relative absolute error	31.8848 %	
Root relative squared error	73.992 %	
Total Number of Instances	158	

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.831	0.138	0.831	0.831	0.831	0.693	0.908	0.859	0.911	Approve
0.862	0.169	0.862	0.862	0.862	0.693	0.913	0.931	?	Disapprove
?	0.000	?	?	?	?	?	?	?	Skipped
Weighted Avg.	0.848	0.155	0.848	0.848	0.848	0.693	0.911	0.899	

== Confusion Matrix ==

a	b	c	<-- classified as
59	12	0	a = Approve
12	75	0	b = Disapprove
0	0	0	c = Skipped

Status

OK

[Log](#)  x 0

MultilayerPerceptron - Cross Validation (Set 5)

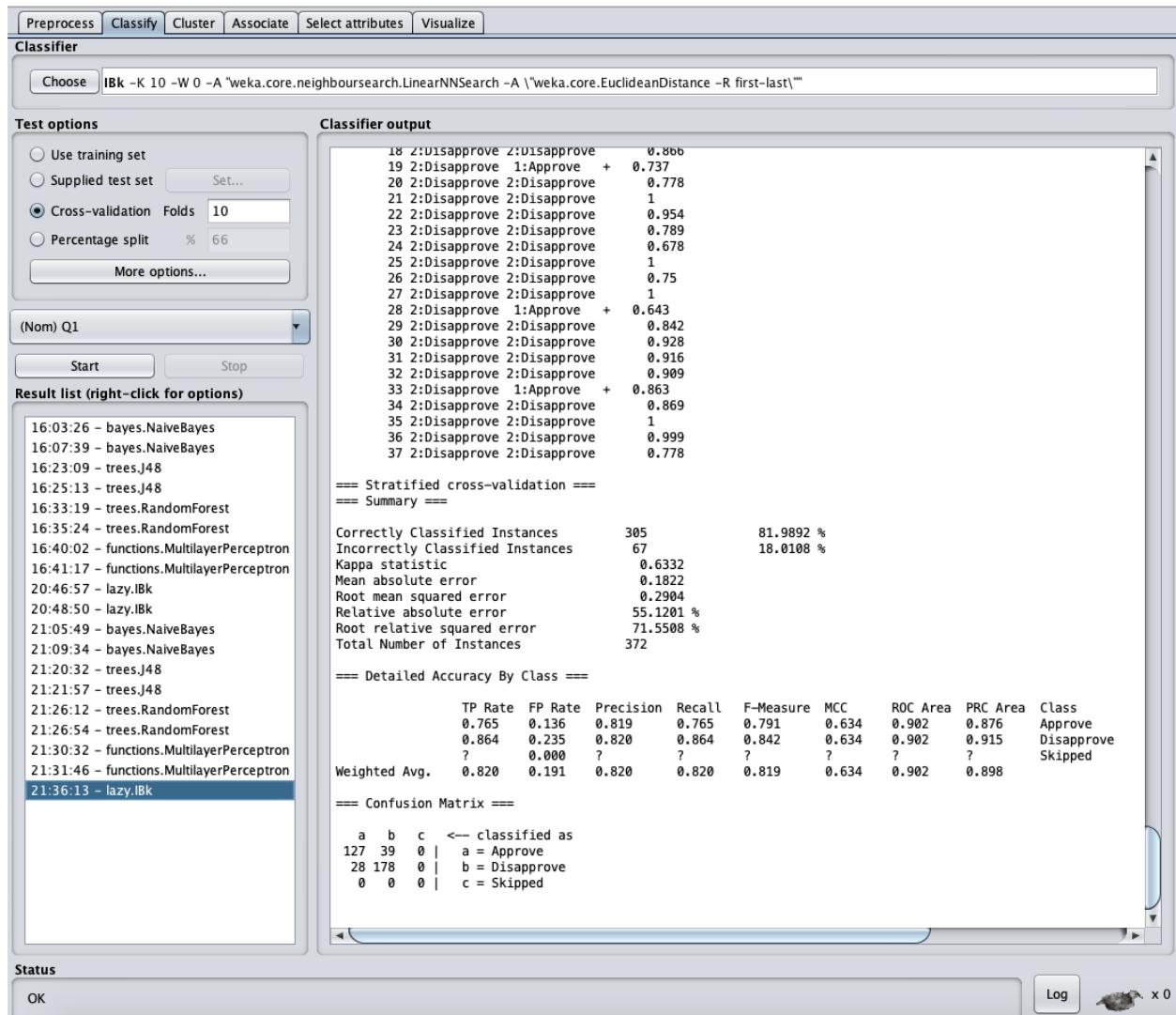
Accuracy	TP Rate	FP Rate	ROC Area	Class
79.57%	76.5%	18.0%	86.1%	Approve
	82.0%	23.5%	86.1%	Disapprove

MultilayerPerceptron - Test set (Set 5)

Accuracy	TP Rate	FP Rate	ROC Area	Class
84.81%	83.1%	13.8%	90.8%	Approve
	86.2%	16.9%	91.3%	Disapprove

5. K-nearest Neighbor (IBK) with K = 10

- Step 1: Choose IBK and select Cross-validation with 10 Folds and Q1 as Class attribute -> Start



- Step 2: re-evaluate the model with the test dataset. Select 'Supplied test set' and click 'Set' -> Open file, open "set5_testing" with Q1 as Class and close -> More options -> choose PlainText as Output prediction and OK -> Start

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose IBk -K 10 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

Test options

- Use training set
- Supplied test set Set...
- Cross-validation Folds 10
- Percentage split % 66
- More options...

(Nom) Q1

Start Stop

Result list (right-click for options)

```

16:03:26 - bayes.NaiveBayes
16:07:39 - bayes.NaiveBayes
16:23:09 - trees.J48
16:25:13 - trees.J48
16:33:19 - trees.RandomForest
16:35:24 - trees.RandomForest
16:40:02 - functions.MultilayerPerceptron
16:41:17 - functions.MultilayerPerceptron
20:46:57 - lazy.IBk
20:48:50 - lazy.IBk
21:05:49 - bayes.NaiveBayes
21:09:34 - bayes.NaiveBayes
21:20:32 - trees.J48
21:21:57 - trees.J48
21:26:12 - trees.RandomForest
21:26:54 - trees.RandomForest
21:30:32 - functions.MultilayerPerceptron
21:31:46 - functions.MultilayerPerceptron
21:36:13 - lazy.IBk
21:37:14 - lazy.IBk

```

Classifier output

```

142 2:Disapprove 2:Disapprove 0.875
143 2:Disapprove 2:Disapprove 0.999
144 2:Disapprove 2:Disapprove 0.95
145 2:Disapprove 2:Disapprove 0.95
146 2:Disapprove 2:Disapprove 0.97
147 2:Disapprove 2:Disapprove 0.999
148 2:Disapprove 2:Disapprove 0.826
149 2:Disapprove 2:Disapprove 0.692
150 2:Disapprove 2:Disapprove 0.75
151 2:Disapprove 2:Disapprove 0.812
152 2:Disapprove 2:Disapprove 0.846
153 2:Disapprove 2:Disapprove 0.615
154 2:Disapprove 2:Disapprove 0.9
155 2:Disapprove 2:Disapprove 0.778
156 2:Disapprove 1:Approve + 0.84
157 2:Disapprove 2:Disapprove 0.812
158 2:Disapprove 2:Disapprove 0.806

```

== Evaluation on test set ==

Time taken to test model on supplied test set: 0.02 seconds

== Summary ==

Correctly Classified Instances	131	82.9114 %
Incorrectly Classified Instances	27	17.0886 %
Kappa statistic	0.6533	
Mean absolute error	0.1832	
Root mean squared error	0.2937	
Relative absolute error	55.4134 %	
Root relative squared error	72.3221 %	
Total Number of Instances	158	

== Detailed Accuracy By Class ==

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
0.789	0.138	0.824	0.789	0.806	0.654	0.897	0.838	0.896	Approve
0.862	0.211	0.833	0.862	0.847	0.654	0.896	0.924	0.887	Disapprove
?	0.000	?	?	?	?	?	?	?	Skipped
Weighted Avg.	0.829	0.178	0.829	0.829	0.654	0.897	0.885		

== Confusion Matrix ==

a	b	c	<-- classified as
56	15	0	a = Approve
12	75	0	b = Disapprove
0	0	0	c = Skipped

Status OK Log x 0

IBK - Cross Validation (Set 5)

Accuracy	TP Rate	FP Rate	ROC Area	Class
81.99%	76.5%	13.6%	90.2%	Approve
	86.4%	23.5%	90.2%	Disapprove

IBK - Test set (Set 5)

Accuracy	TP Rate	FP Rate	ROC Area	Class
82.91%	78.9%	13.8%	89.7%	Approve
	86.2%	21.1%	89.6%	Disapprove

Performance Comparisons

To compare and rank the classifiers, we performed t-test comparison on all 5 attribute sets for 4 comparison fields - accuracy, FP Rate, TP Rate, ROC curve. First, the accuracy comparison is conducted as below:

- Step 1: Open ‘Experimenter’ -> Click ‘New’ -> Click ‘Add new’ under Dataset -> Select all of the five training datasets that are previously created and click ‘Open’ -> Click ‘Add new’ under Algorithms -> Click ‘Choose’ and select NaiveBayes and click OK. Do the same for the other four algorithms (make sure k = 10 for IBk algorithm)
- Step 2: Go to ‘Run’ tab and click “Start”. It will take an hour to finish the process with 0 error
- Step 3: Go to ‘Analyse’ tab and select ‘Paired T-Tester’ for ‘Testing with’, ‘Percent_correct’ for ‘Comparison field’ and significance of 0.05. Click the ‘Select’ next to ‘Test base’, select NaiveBayes algorithm and click ‘Perform test’ under ‘Actions’

Below is the result of the t-test based on accuracy.

The screenshot shows the Weka Experimenter interface with the 'Analysse' tab selected. The 'Actions' panel includes 'Perform test', 'Save output', and 'Open Explorer...'. The 'Configure test' panel shows settings for 'Testing with' (Paired T-Tester), 'Comparison field' (Percent_correct), 'Significance' (0.05), and 'Test base' (Select). The 'Test output' panel displays the command run: 'weka.experiment.PairedTTester -G 4,5,6 -D 1 -R 2 -S 0.05 -result-matrix ...'. It also lists datasets (5), resultsets (5), confidence (0.05 two tailed), and date (11/9/20, 3:30 PM). The 'Dataset' section shows a table of accuracy values for various classifiers across different datasets. The 'Result list' panel at the bottom shows log entries for available resultsets and specific t-test runs.

Dataset	(1) bayes.Na (2) trees (3) trees (4) funct (5) lazy.
'R_data_frame-weka.filter(100)	51.34 54.43 v 53.90 v 51.77 54.65 v
'R_data_frame-weka.filter(100)	52.82 53.95 v 52.58 52.43 53.23
'R_data_frame-weka.filter(100)	49.92 53.49 v 39.44 * 46.12 * 53.17 v
'R_data_frame-weka.filter(100)	55.34 51.96 * 47.13 * 46.32 * 54.06
'R_data_frame-weka.filter(100)	51.41 46.71 * 45.57 * 43.57 * 46.40 *

Key:
(1) bayes.NaiveBayes '' 5995231201785697655
(2) trees.J48 '-C 0.25 -M 2' -217733168393644444
(3) trees.RandomForest '-P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1' 11168394
(4) functions.MultilayerPerceptron '-L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a' -599066
(5) lazy.IBk '-K 10 -W 0 -A "\weka.core.neighboursearch.LinearNNSearch -A \\\weka.co

The test was performed with significance of 0.05 and Naive Bayes as the Test base.

- J48 performed 3 times better than Naive Bayes and 2 times worse.
- Random Forest performed better once, 3 times worse, and 1 time equal to Naive Bayes.
- MultiLayer Perceptron performed 3 times worse than Naive Bayes, and none better.
- K-nearest Neighbor performed 2 times better and once worse.

To determine the rank between K-nearest Neighbor and J48, we redid the test with K-nearest Neighbor as Test base.

The screenshot shows the Weka interface with the 'Paired T-Tester' configuration and its output.

Configure test:

- Testing with: Paired T-Tester
- Select rows and cols: Rows, Cols, Swap
- Comparison field: Percent_correct
- Significance: 0.05
- Sorting (asc.) by: <default>
- Test base: Select
- Displayed Columns: Select
- Show std. deviations:
- Output Format: Select

Test output:

```

Tester: weka.experiment.PairedTTester -G 4,5,6 -D 1 -R 2 -S 0.05 -result-matrix "v
Analysing: Percent_correct
Datasets: 5
Resultsets: 5
Confidence: 0.05 (two tailed)
Sorted by: -
Date: 11/9/20, 3:33 PM

Dataset      (5) lazy.IBk | (1) bayes (2) trees (3) trees (4) funct
'R_data_frame-weka.filter(100)' 54.65 | 51.34 * 54.43 53.90 * 51.77 *
'R_data_frame-weka.filter(100)' 53.23 | 52.82 53.95 52.58 * 52.43
'R_data_frame-weka.filter(100)' 53.17 | 49.92 * 53.49 39.44 * 46.12 *
'R_data_frame-weka.filter(100)' 54.06 | 55.34 51.96 * 47.13 * 46.32 *
'R_data_frame-weka.filter(100)' 46.40 | 51.41 v 46.71 45.57 43.57 *

(v/ /*) | (1/2/2) (0/4/1) (0/1/4) (0/1/4)

Key:
(1) bayes.NaiveBayes '' 5995231201785697655
(2) trees.J48 '-C 0.25 -M 2' -217733168393644444
(3) trees.RandomForest '-P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1' 11168394
(4) functions.MultilayerPerceptron '-L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a' -599066
(5) lazy.IBk '-K 10 -W 0 -A \"weka.core.neighboursearch.LinearNNSearch -A \\"\\weka.co

```

Result list:

```

12:52:35 - Available resultsets
12:52:36 - Percent_correct - bayes.NaiveBayes " 5995
12:52:58 - Percent_correct - trees.J48 '-C 0.25 -M 2'
12:53:09 - Percent_correct - bayes.NaiveBayes " 5995
14:20:36 - Percent_correct - bayes.NaiveBayes " 5995
15:30:00 - Available resultsets
15:30:12 - Percent_correct - bayes.NaiveBayes " 5995
15:33:26 - Percent_correct - lazy.IBk '-K 10 -W 0 -A '

```

J48 performed once worse than K-Nearest Neighbor and 4 times equally the same. Therefore, the final ranking based on accuracy from high to low is **KNN, J48, Naive Bayes, Random Forest, Multilayer Perceptron.**

Second, TP Rate as the comparison field is performed as below.

Configure test

- Testing with: Paired T-Tester
- Select rows and cols: Rows, Cols, Swap
- Comparison field: True_positive_rate
- Significance: 0.05
- Sorting (asc.) by: <default>
- Test base: Select
- Displayed Columns: Select
- Show std. deviations:
- Output Format: Select

Test output

```

Tester: weka.experiment.PairedTTester -G 4,5,6 -D 1 -R 2 -S 0.05 -result-matrix "C:\Users\DELL\OneDrive\Desktop\Weka\output\True_positive_rate.result"
Analysing: True_positive_rate
Datasets: 5
Resultsets: 5
Confidence: 0.05 (two tailed)
Sorted by: -
Date: 11/9/20, 3:54 PM

```

Dataset	(1) bayes.N (2) tree (3) tree (4) func (5) lazy
'R_data_frame-weka.filter(100)	0.15 0.07 * 0.08 * 0.14 0.06 *
'R_data_frame-weka.filter(100)	0.18 0.07 * 0.15 * 0.19 0.13 *
'R_data_frame-weka.filter(100)	0.22 0.00 * 0.37 v 0.34 v 0.13 *
'R_data_frame-weka.filter(100)	0.24 0.15 * 0.34 v 0.40 v 0.37 v
'R_data_frame-weka.filter(100)	0.72 0.76 v 0.62 * 0.56 * 0.89 v

(v/ /*) | (1/0/4) (2/0/3) (2/2/1) (2/0/3)

Key:

- (1) bayes.NaiveBayes '' 5995231201785697655
- (2) trees.J48 '-C 0.25 -M 2' -217733168393644444
- (3) trees.RandomForest '-P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1' 11168394
- (4) functions.MultilayerPerceptron '-L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a' -599066
- (5) lazy.IBk '-K 10 -W 0 -A \"weka.core.neighboursearch.LinearNNSearch -A \\"weka.co

Result list

```

12:52:36 - Percent_correct - bayes.NaiveBayes "5995231201785697655"
12:52:58 - Percent_correct - trees.J48 '-C 0.25 -M 2' -217733168393644444
12:53:09 - Percent_correct - bayes.NaiveBayes "5995231201785697655"
14:20:36 - Percent_correct - bayes.NaiveBayes "5995231201785697655"
15:30:00 - Available resultsets
15:30:12 - Percent_correct - bayes.NaiveBayes "5995231201785697655"
15:33:26 - Percent_correct - lazy.IBk '-K 10 -W 0 -A \"weka.core.neighboursearch.LinearNNSearch -A \\"weka.co
15:54:34 - True_positive_rate - bayes.NaiveBayes "5995231201785697655"

```

From TP Rate perspective,

- J48 performed once better than Naive Bayse, and 4 times worse.
- Random Forest performed twice better, and 3 times worse.
- MultiLayer Perceptron performed 2 times better, 2 times equally, and once worse than Naive Bayes.
- KNN performed twice better, and 3 times worse than Naive Bayes.

If we change our Test base to Multilayer Perceptron, the result returns as below.

Setup Run Analyse

Source

Got 2500 results

Actions

Perform test Save output Open Explorer...

Configure test

Testing with Paired T-Tester

Select rows and cols Rows Cols Swap

Comparison field True_positive_rate

Significance 0.05

Sorting (asc.) by <default>

Test base Select

Displayed Columns Select

Show std. deviations

Output Format Select

Test output

```

Tester: weka.experiment.PairedTTester -G 4,5,6 -D 1 -R 2 -S 0.05 -result-matrix "v
Analysing: True_positive_rate
Datasets: 5
Resultsets: 5
Confidence: 0.05 (two tailed)
Sorted by: -
Date: 11/9/20, 3:58 PM

```

Dataset	(4) functio	(1) baye	(2) tree	(3) tree	(5) lazy
'R_data_frame-weka.filter(100)	0.14	0.15	0.07 *	0.08 *	0.06 *
'R_data_frame-weka.filter(100)	0.19	0.18	0.07 *	0.15 *	0.13 *
'R_data_frame-weka.filter(100)	0.34	0.22 *	0.00 *	0.37	0.13 *
'R_data_frame-weka.filter(100)	0.40	0.24 *	0.15 *	0.34 *	0.37 *
'R_data_frame-weka.filter(100)	0.56	0.72 v	0.76 v	0.62 v	0.89 v

(v/ /*) | (1/2/2) (1/0/4) (1/1/3) (1/0/4)

Key:

- (1) bayes.NaiveBayes '' 5995231201785697655
- (2) trees.J48 '-C 0.25 -M 2' -217733168393644444
- (3) trees.RandomForest '-P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1' 11168394
- (4) functions.MultilayerPerceptron '-L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a' -599066
- (5) lazy.IBk '-K 10 -W 0 -A \'weka.core.neighboursearch.LinearNNSearch -A \\'\"weka.co

The final ranking based on the true positive rate from high to low is **Multilayer Perceptron, Naive Bayes, Random Forest, KNN, and J48**.

Third, the comparison field of FP Rate implemented as below.

The screenshot shows the Weka interface with the following configuration:

- Testing with:** Paired T-Tester
- Select rows and cols:** Rows, Cols, Swap
- Comparison field:** False_positive_rate
- Significance:** 0.05
- Sorting (asc.) by:** <default>
- Test base:** Select
- Displayed Columns:** Select
- Show std. deviations:**
- Output Format:** Select

Test output:

```

Tester: weka.experiment.PairedTTester -G 4,5,6 -D 1 -R 2 -S 0.05 -result-matrix "v
Analysing: False_positive_rate
Datasets: 5
Resultsets: 5
Confidence: 0.05 (two tailed)
Sorted by: -
Date: 11/9/20, 4:09 PM

Dataset (1) bayes.N | (2) tree (3) tree (4) func (5) lazy
'R_data_frame-weka.filter(100) 0.17 | 0.05 * 0.06 * 0.13 * 0.04 *
'R_data_frame-weka.filter(100) 0.15 | 0.06 * 0.13 * 0.16 0.11 *
'R_data_frame-weka.filter(100) 0.22 | 0.08 * 0.38 v 0.32 v 0.11 *
'R_data_frame-weka.filter(100) 0.15 | 0.13 * 0.26 v 0.31 v 0.24 v
'R_data_frame-weka.filter(100) 0.20 | 0.27 v 0.25 v 0.23 v 0.44 v
----- (v/ /*) | (1/0/4) (3/0/2) (3/1/1) (2/0/3)

Key:
(1) bayes.NaiveBayes '' 5995231201785697655
(2) trees.J48 '-C 0.25 -M 2' -217733168393644444
(3) trees.RandomForest '-P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1' 11168394
(4) functions.MultilayerPerceptron '-L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a' -599066
(5) lazy.IBk '-K 10 -W 0 -A \'weka.core.neighboursearch.LinearNNSearch -A \\\\"weka.co

```

Result list:

```

15:30:12 - Percent_correct - bayes.NaiveBayes "5995
15:33:26 - Percent_correct - lazy.IBk '-K 10 -W 0 -A \
15:54:34 - True_positive_rate - bayes.NaiveBayes "59
15:57:29 - True_positive_rate - lazy.IBk '-K 10 -W 0 -
15:58:24 - True_positive_rate - functions.MultilayerPer
15:59:00 - True_positive_rate - lazy.IBk '-K 10 -W 0 -
16:00:09 - True_positive_rate - trees.RandomForest '-
16:09:17 - False_positive_rate - bayes.NaiveBayes "5

```

As shown above, Random Forest and Multilayer Perceptron have higher false positive rate than that of Naive Bayes 3 times, and KNN's false positive rate is higher than Naïve Bayes' twice. Meanwhile, J48 performed better 4 times.

To compare MultiLayer Perceptron and Random Forest, we have to change the Test base to Random Forest.

The screenshot shows the Weka interface with the 'Analyse' tab selected. In the 'Test output' pane, a Paired T-Test is being run on 2500 results. The configuration includes:

- Testing with: Paired T-Tester
- Select rows and cols: Rows, Cols, Swap
- Comparison field: False_positive_rate
- Significance: 0.05
- Sorting (asc.) by: <default>
- Test base: Select
- Displayed Columns: Select
- Show std. deviations:
- Output Format: Select

The 'Test output' pane displays the results of the Paired T-Test, including the tester used, datasets, result sets, confidence, sorted by, and date. It also shows a dataset comparison table and a key for classifier names.

Dataset	(3) trees.R	(1) baye	(2) tree	(4) func	(5) lazy
'R_data_frame-weka.filter(100)	0.06	0.17 v	0.05 *	0.13 v	0.04 *
'R_data_frame-weka.filter(100)	0.13	0.15 v	0.06 *	0.16	0.11 *
'R_data_frame-weka.filter(100)	0.38	0.22 *	0.00 *	0.32 *	0.11 *
'R_data_frame-weka.filter(100)	0.26	0.15 *	0.13 *	0.31 v	0.24 *
'R_data_frame-weka.filter(100)	0.25	0.20 *	0.27 v	0.23 *	0.44 v

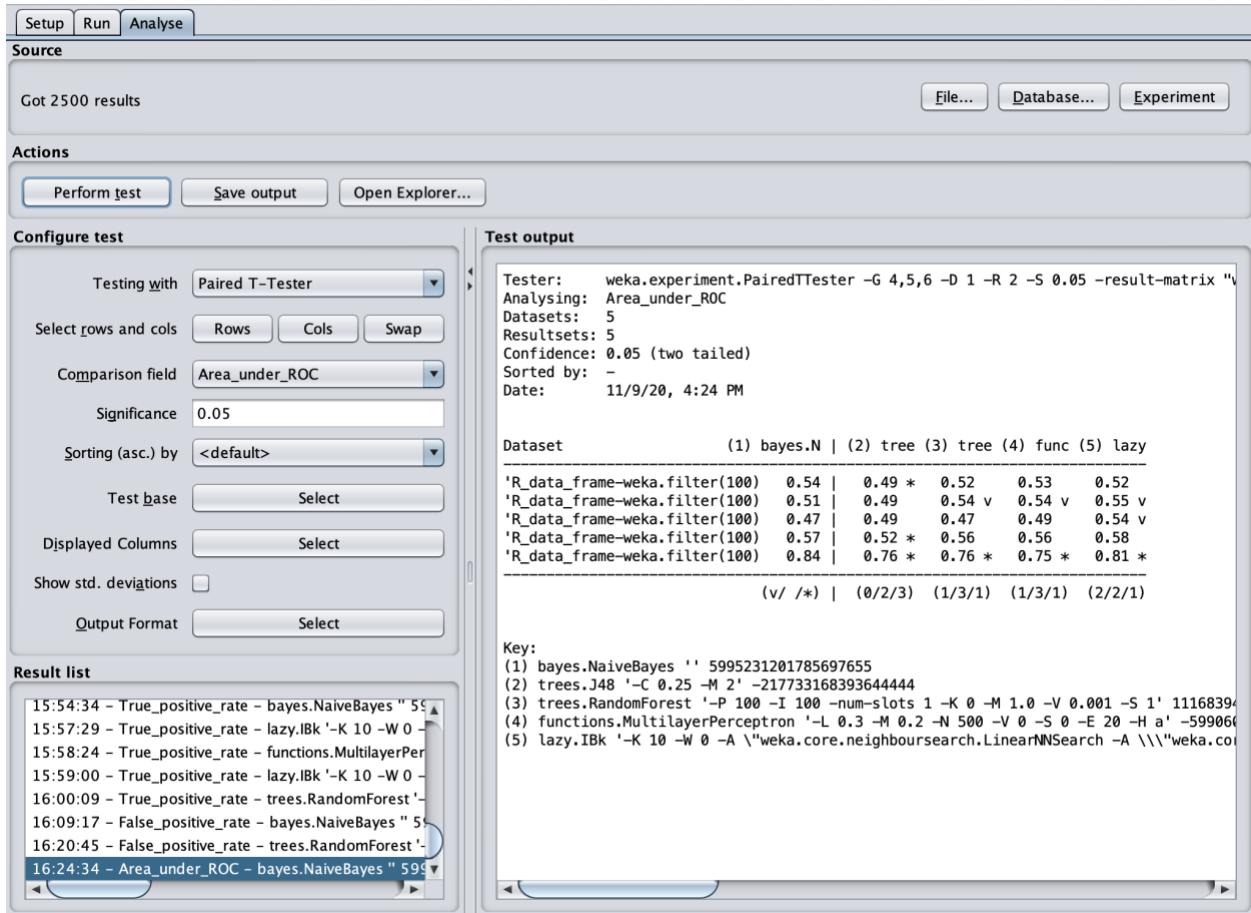
Key:

- (1) bayes.NaiveBayes '' 5995231201785697655
- (2) trees.J48 '-C 0.25 -M 2' -217733168393644444
- (3) trees.RandomForest '-P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1' 11168394
- (4) functions.MultilayerPerceptron '-L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a' -599066
- (5) lazy.IBk '-K 10 -W 0 -A \'weka.core.neighboursearch.LinearNNSearch -A \\'\"weka.co

As we can see, MultiLayer Perceptron has performed better twice, twice worse than, and once equally to Random Forest.

In sum, the final ranking based on the false positive rate from best to worst (best indicating the classifier with lowest FP rate) is **J48, KNN, Naïve Bayes, Random Forest, Multilayer Perceptron**.

Lastly, we want to compare the performances based on the area under the ROC curve.



With Naive Bayes as our Test base,

- J48 performed 3 times worse, never better.
- Random Forest and MultiLayer Perceptron both performed equally 3 times, once better and once worse than Naive Bayes.
- KNN performed twice better, once worse, and once equal to Naive Bayes.

Therefore, based on the area under the ROC curve, we can rank the models from best to worst as follows - **KNN, Naive Bayes, MultiLayer Perceptron, Random Forest, J48**.

Discussion and Conclusion

From the various attribute selection methods, we found that the most affective factors were QPID - political party the participant's support, and the response to question 3 - whether or not to open the country. This implies that the political orientation and the opinions in regards to when to reopen the US economy played more significant roles than any other factors (ex. ethnicity, age, education level, etc.) for the assessment of the Trump Administration's response.

Based on the results of the t-test we performed, each approach has come up with different results in terms of what algorithm works best. For example, KNN performed the best accuracy and highest area under ROC, while Multilayer Perceptron was the top model for TP rate and FP rate. However, KNN was second to the worst for the TP rate, while Multilayer Perceptron was the worst for accuracy.

Even though there's no absolute winner, we conclude that **K-Nearest Neighbors** is the best classifier in this case. It was wishy-washy for TP rate, but the result may potentially be better with different k values (i.e., different number of Neighbors). Moreover, it is true that accuracy is not always the best model evaluation metric. It can convey the health of a model well only when all the classes have similar prevalence in the data. However, since we split the dataset into train/test datasets with the same distribution of the class label, we have symmetric datasets (FN & FP counts are close). Since KNN was the best for the accuracy and area under ROC, we've ultimately chosen it as the best classifier model.