Data Classification Project

Team Members

Jyoti Sharma
Siyu Xu

March 21, 2020

Data Mining

Jae Yong Lee

Table of Contents

DATA MINING GOAL

The goal of this Data mining classification project is to predict the occurrence of rain in San Francisco Bay area.

The classification process will be carried out on based on information about various factors such as humidity, max/min air temperature, air speed etc.

DATASET INFORMATION

Dataset Description

The Dataset used is a monthly report generated by CIMIS (California Irrigation Management Information system) from January 2018 to January 2020. The Dataset constitutes of **732 tuples** containing **33 attributes** each.

The attributes consist of data about air temperature, wind speed, wind direction, solar radiation, relative humidity, vapor pressure, and rainfall.

Dataset Attributes

Below table consists of description of all attributes in the dataset:

S. No.	Attribute Name	Attribute Details
1	Stn Id	Id of computerized automated weather stations part of CIMS network
2	Stn Name	Name of computerized automated weather stations part of CIMS network
3	CIMIS Region	Geographic Region where Observations are made
4	Date	Date on Which Observation was taken
5	Jul	Number of Observation made in current year including the current record
6	ETo (in)	evapotranspiration recorded in in
7	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for ETO recording

8	Precip (in)	Precipitation field value indicates if precipitation occurred or not
9	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Precipitation recording
10	Sol Rad (Ly/day)	Solar radiation intensity per day
11	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for solar radiation recording
12	Avg Vap Pres (mBars)	Average Vapor pressure
13	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Average Vapor pressure recording
14	Max Air Temp (F)	Maximum Air Temp
15	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Max Air Temp recording
16	Min Air Temp (F)	Minimum Air temperature
17	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Minimum Air temperature recording
18	Avg Air Temp (F)	Average Air temperature
19	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Average Air temperature recording
20	Max Rel Hum (%)	Maximum Relative Humidity measurement
21	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Max Rel Hum recording
22	Min Rel Hum (%)	Min Relative Humidity measurement

23	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Min Rel Hum recording
24	Avg Rel Hum (%)	Average Humidity
25	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Avg Rel Hum recording
26	Dew Point (F)	Dew point temperature is the temperature to which the atmosphere must be cooled, at constant pressure and water vapor content, in order to reach saturation
27	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Dew Point recording
28	Avg Wind Speed (mph)	Average windspeed
29	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Avg Wind Speed recording
30	Wind Run (miles)	Wind Run is the total distance the wind has traveled past the weather station within a given time period
31	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Wind Run recording
32	Avg Soil Temp (F)	Average Soil temperature in F
33	qc	Quality Control to indicate if data recorded is in normal range or far out of normal range or historical average for Avg Soil Temp recording

Class Attribute

The class attribute is our prediction goal and would be the 8th attribute in the dataset. The current values populated are zero or greater than zero, the goal will be to generate this class attribute values as 0 or 1 indicating if precipitation occurred on that day or not.

Data Sources

https://cimis.water.ca.gov

DATA MING TOOLS OR ALGORITHMS INFORMATION

Attribute Selection Algorithms

As a part of preprocessing and for generating unique datasets we have decided the below attribute selection algorithms.

1) **OneRAttributeEval:**
   Evaluates the worth of an attribute by using the OneR classifier.

2) **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. Subsets of features that are highly correlated with the class while having low intercorrelation are preferred (Weka).

3) **InfoGainAttributeEval:**
   Evaluates the worth of an attribute by measuring the information gain with respect to the class (Weka).

   InfoGain(Class, Attribute) = H(Class) – H(Class | Attribute) (Weka).

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 (Weka).

5) **Manual Selection:**
   Attributes are selected by our understanding of datasets.

Classifier Algorithms

1) **Naïve Bayes:**
   In machine learning, Naïve Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naïve) independence assumptions between the features. They are among the simplest Bayesian network models (Wikipedia, 2020).

2) **SGD:**
   SGD implements stochastic gradient descent for learning various linear models

(binary class SVM, binary class logistic regression, squared loss, Huber loss and epsilon-insensitive loss linear regression).

Globally replaces all missing values and transforms nominal attributes into binary ones. It also normalizes all attributes, so the coefficients in the output are based on the normalized data. For numeric class attributes, the squared, Huber or epsilon-insensitive loss function must be used. Epsilon-insensitive and Huber loss may require a much higher learning rate (Weka).

3) **J48:**
J48 (C4.5) is an algorithm used to generate a decision tree developed by Ross Quinlan, which can be used for classification. And for this reason, J48 is often referred to as a statistical classifier (Wikipedia, 2020).

4) **Random Forest:**
Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set (Wikipedia, 2020).

5) **IBk (k = 10):**
IBk (Instance Based Learner) algorithm generates a prediction for a test instance just-in-time. It uses a distance measure to locate k "close" instances in the training data for each test instance and uses those selected instances to make a prediction (Weka).

DATA PREPROCESSING

Manual preprocessing

In the original dataset, it has 33 attributes, as a part of data pre-processing. We have removed some columns for which more than 95% values were missing.

Below are the steps used.

Firstly, we use RStudio to do the data preprocessing for our project:

1. Import our original data into RStudio.

```
> #import data
> data <- read.table(file = "daily.csv", header = TRUE, sep = ",")
> head(data)
  Stn.Id  Stn.Name      CIMIS.Region     Date Jul ETo..in. qc Precip..in. qc.1 Sol.Rad..Ly.day. qc.2
1     47 Brentwood San Francisco Bay 1/1/2018   1     0.05          0.00                  190
2     47 Brentwood San Francisco Bay 1/2/2018   2     0.03          0.00                  105
3     47 Brentwood San Francisco Bay 1/3/2018   3     0.01          0.27                   77
4     47 Brentwood San Francisco Bay 1/4/2018   4     0.03          0.15                  141
5     47 Brentwood San Francisco Bay 1/5/2018   5     0.01          0.02                   80
6     47 Brentwood San Francisco Bay 1/6/2018   6     0.06          0.00                  231
  Avg.Vap.Pres..mBars. qc.3 Max.Air.Temp..F. qc.4 Min.Air.Temp..F. qc.5 Avg.Air.Temp..F. qc.6 Max.Rel.Hum....
1                  8.3                 66.5                 38.3                 51.1                    90
2                  8.6                 60.1                 39.8                 49.1                    85
3                 10.0                 55.0                 38.5                 48.5                    99
4                 13.7                 67.3                 49.0                 55.0                    99
5                 14.2                 63.0                 47.8                 54.5                    99
6                 11.5                 65.5                 45.4                 55.0                    98
  qc.7 Min.Rel.Hum.... qc.8 Avg.Rel.Hum.... qc.9 Dew.Point..F. qc.10 Avg.Wind.Speed..mph. qc.11
1                   38                   65              39.8                        2.9
2                   59                   72              40.6                        2.2
3                   70                   86              44.5                        3.1
4                   70                   92              52.9                        3.5
5                   90                   98              54.0                        4.5
6                   57                   78              48.2                        4.7
  Wind.Run..miles. qc.12 Avg.Soil.Temp..F. qc.13
1             68.9                   47.8
2             53.6                   48.0
3             74.7                   48.6
4             85.2                   51.6
5            107.6                   52.8
6            113.8                   53.6
```

2. Remove all unnecessary columns, such as “Stn Id”, “Stn Name”, “CIMIS Region” (as entire dataset belongs to one weather station), and all “qc” columns (more than 95% values are missing).

```
> #delete unnecessary information
> data2 <- data[,c(-1:-5,-7,-9,-11,-13,-15,-17,-19,-21,-23,-25,-27,-29,-31,-33)]
```

3. Add one more column named “class”, which is based on the column of “Precip (in)”, if the value is larger than 0 it is shows in “True”, otherwise, “False”.

```
> #add one more column to clarify class attribute
> attach(data2)
> data2$class = ifelse(Precip..in. > 0, 'True', 'False')
```

4. remove “Precip (in)”, in order to keep our class attribute independent.

```
> #remove the previous class attribute "Precip (in)"
> data3 <- data2[,-2]
```

After the preprocessing from RStudio, our dataset looks like:

```
> head(data3)
  ETo..in. Sol.Rad..Ly.day. Avg.Vap.Pres..mBars. Max.Air.Temp..F. Min.Air.Temp..F. Avg.Air.Temp..F.
1     0.05              190                  8.3             66.5             38.3             51.1
2     0.03              105                  8.6             60.1             39.8             49.1
3     0.01               77                 10.0             55.0             38.5             48.5
4     0.03              141                 13.7             67.3             49.0             55.0
5     0.01               80                 14.2             63.0             47.8             54.5
6     0.06              231                 11.5             65.5             45.4             55.0
  Max.Rel.Hum.... Min.Rel.Hum.... Avg.Rel.Hum.... Dew.Point..F. Avg.Wind.Speed..mph. Wind.Run..miles.
1              90              38              65          39.8                 2.9             68.9
2              85              59              72          40.6                 2.2             53.6
3              99              70              86          44.5                 3.1             74.7
4              99              70              92          52.9                 3.5             85.2
5              99              90              98          54.0                 4.5            107.6
6              98              57              78          48.2                 4.7            113.8
  Avg.Soil.Temp..F. class
1              47.8 False
2              48.0 False
3              48.6  True
4              51.6  True
5              52.8  True
6              53.6 False
```

Secondly, we use Weka to transform our dataset from the format of CSV to ARFF.

DATA PREPROCESSING

Attribute Reduction

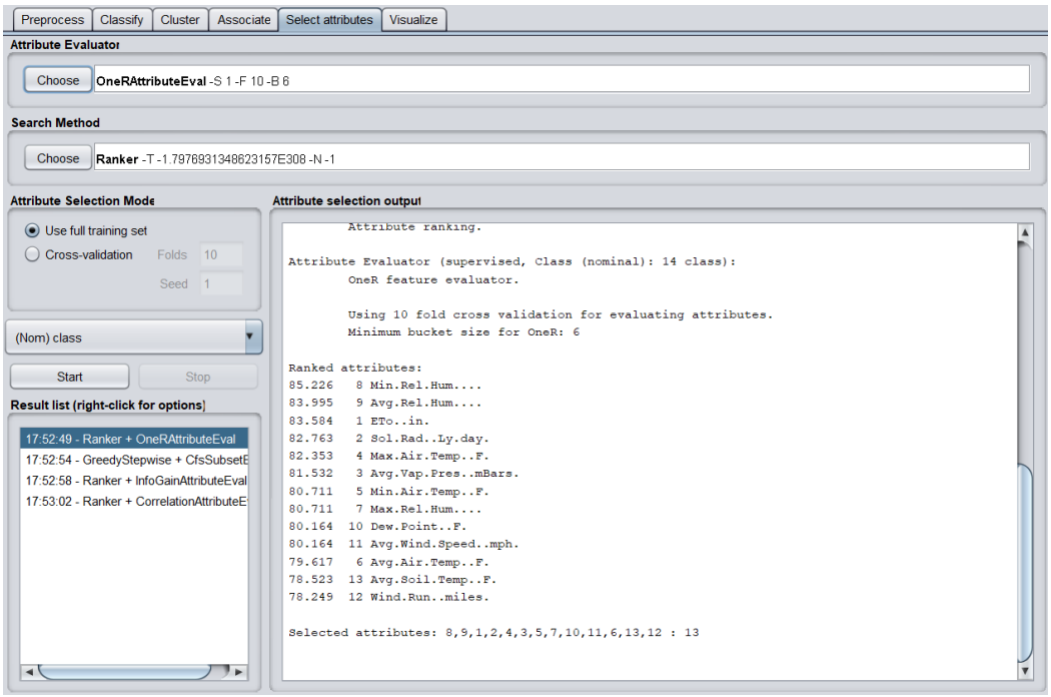
1. Dataset creation via select attributes algorithms
   We have created first four datasets using select attributes algorithms.

   Below is the mapping of dataset and the corresponding attribute evaluator and Search Method combinations.

Data set Names	Attribute Evaluator	Search Method
Daily_1	OneRAttributeEval	Ranker
Daily_2	CfsSubsetEval	GreedyStepwise
Daily_3	InfoGainAttributeEval	Ranker
Daily_4	CorrelationAttributeEval	Ranker

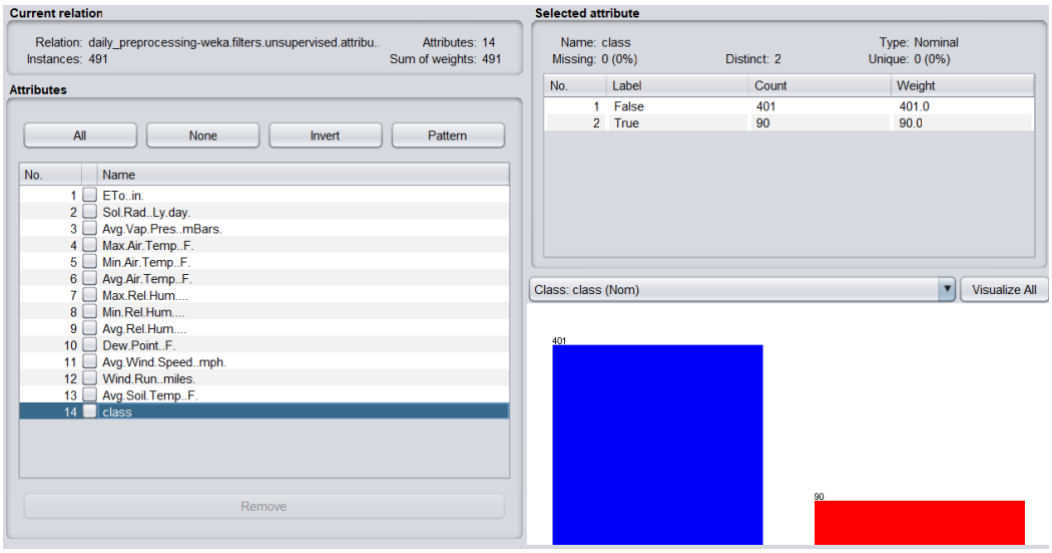
Below are the screenshots depicting steps for each attribute evaluator:

1) OneR

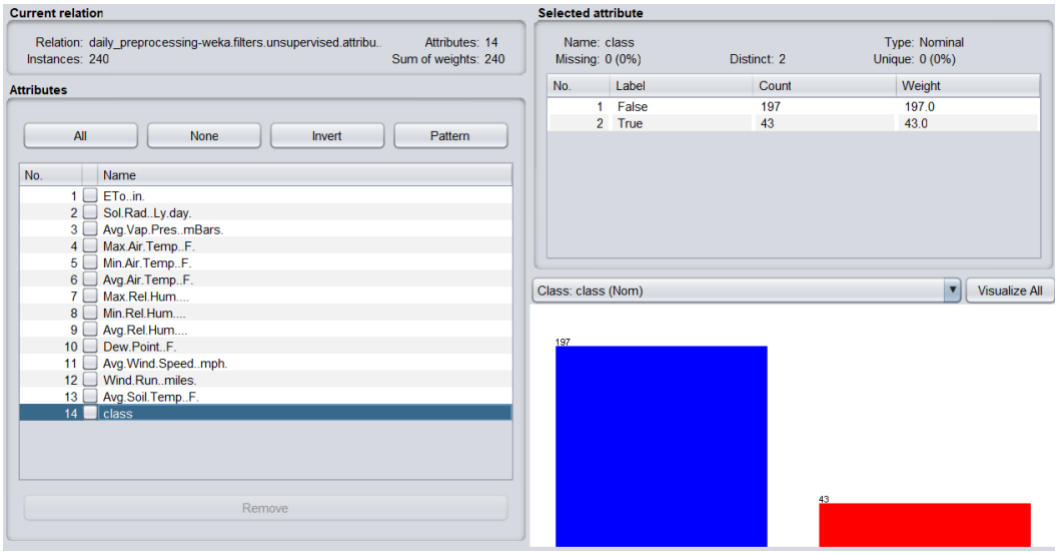


The class attribute is: 13.

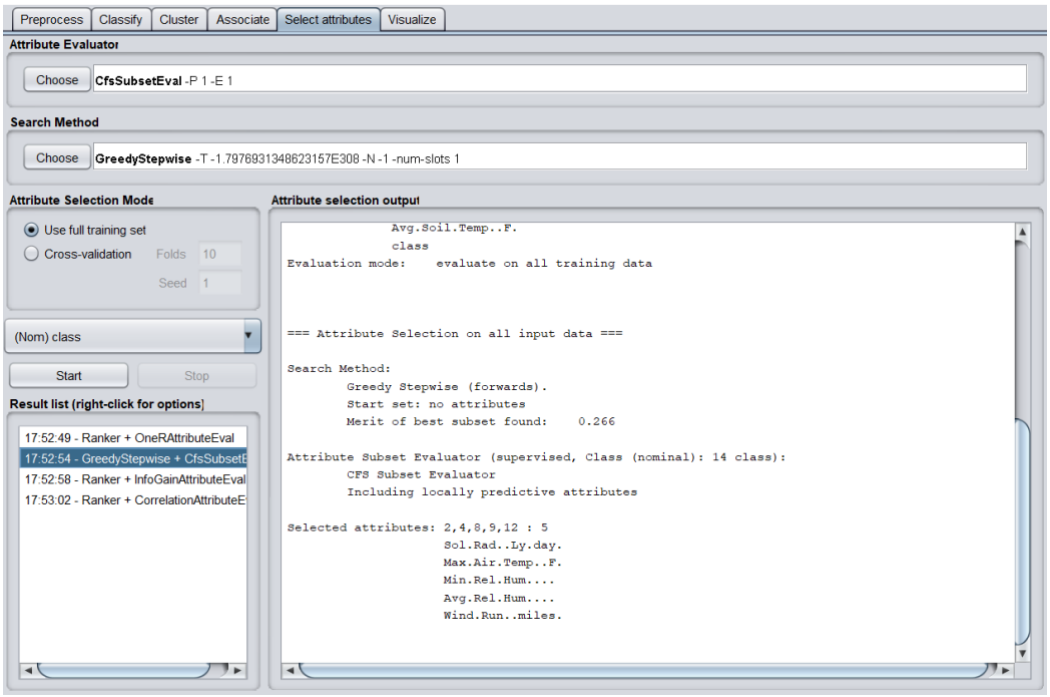
The training dataset with selected attributes:



The test dataset with selected attributes:

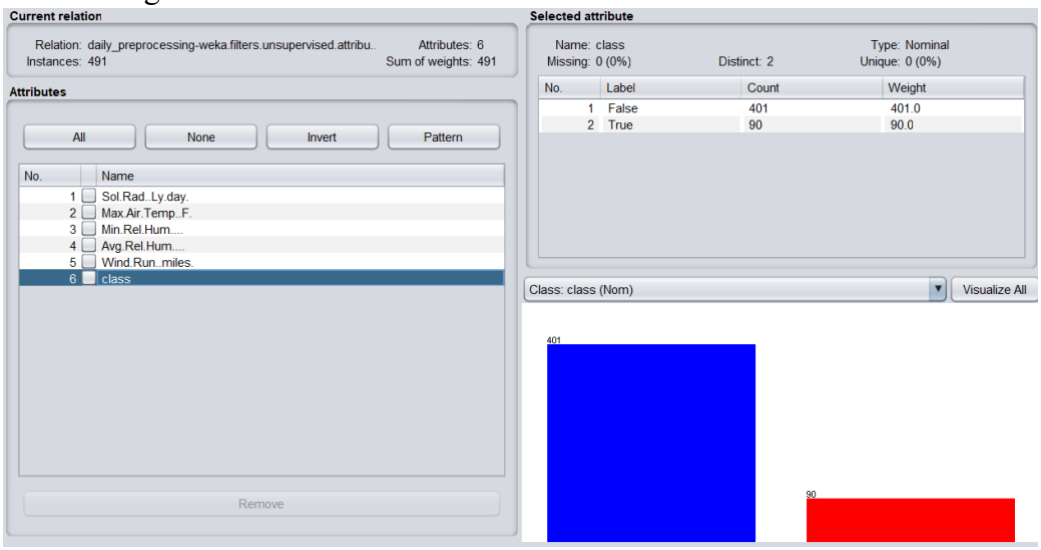


2) Cfs Subset

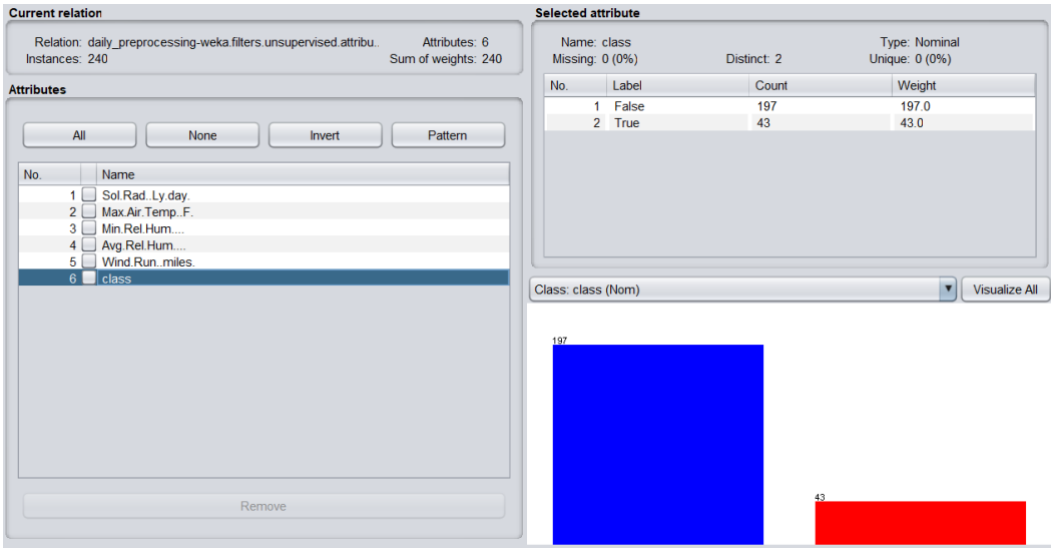


The class attribute is: 5.

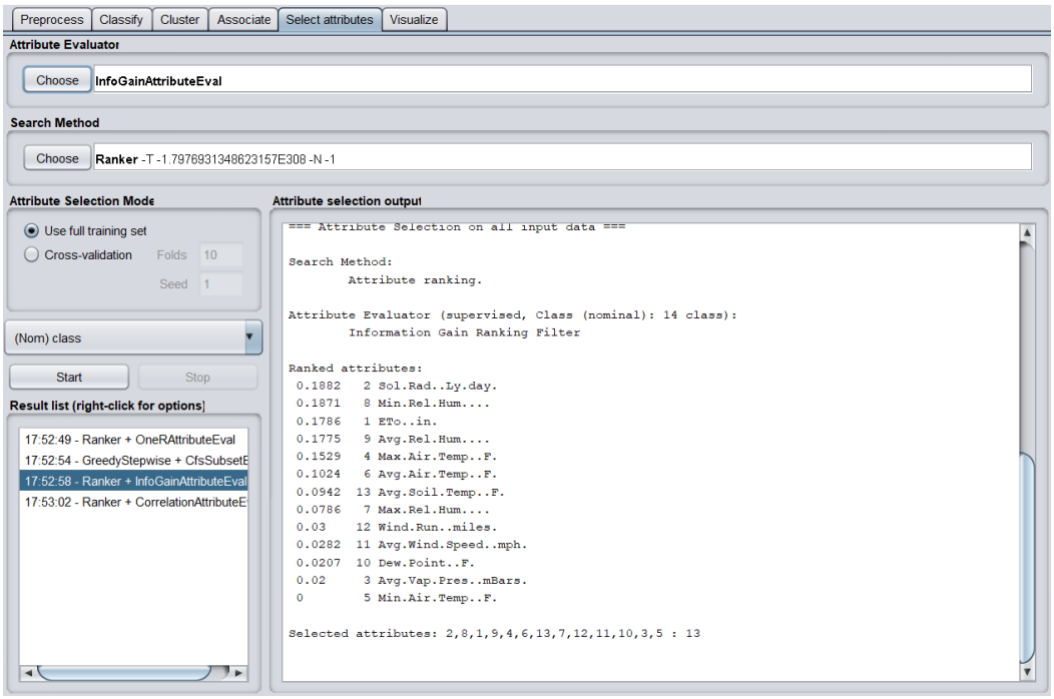
The training dataset with selected attributes:



The test dataset with selected attributes:

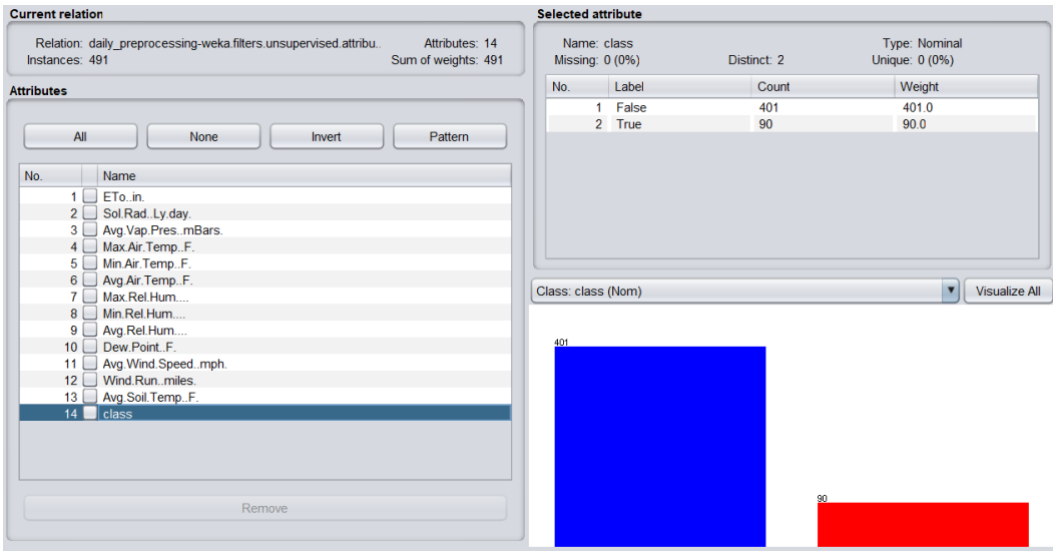


3) Information Gain

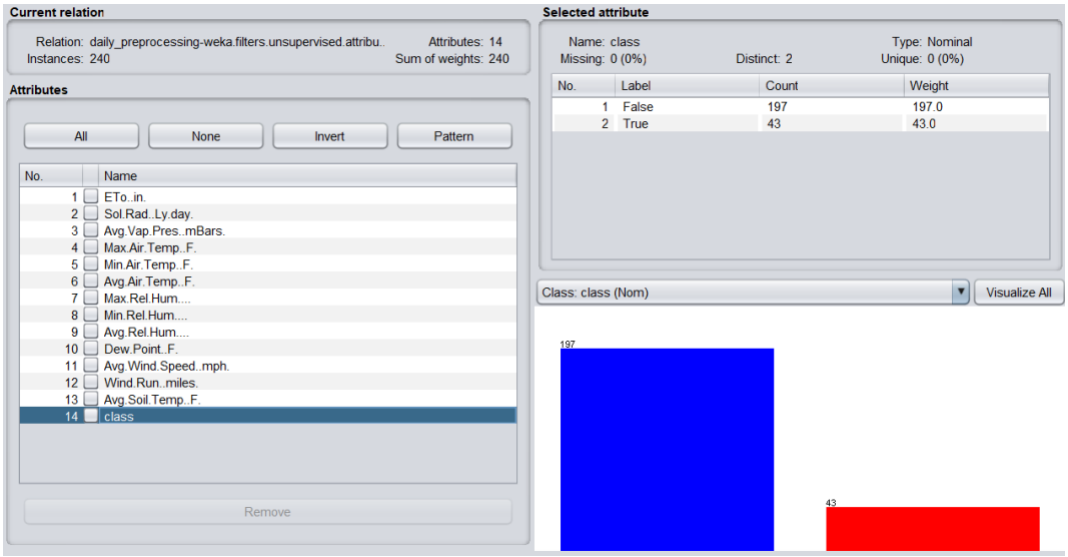


The class attribute is: 13.

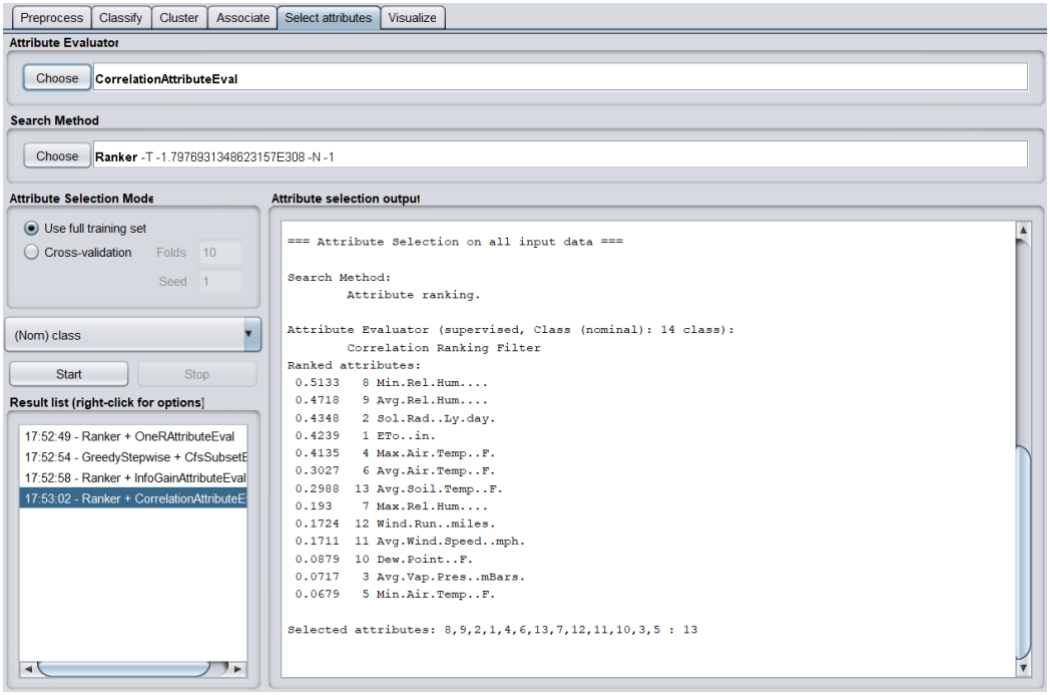
The training dataset with selected attributes:



The test dataset with selected attributes:

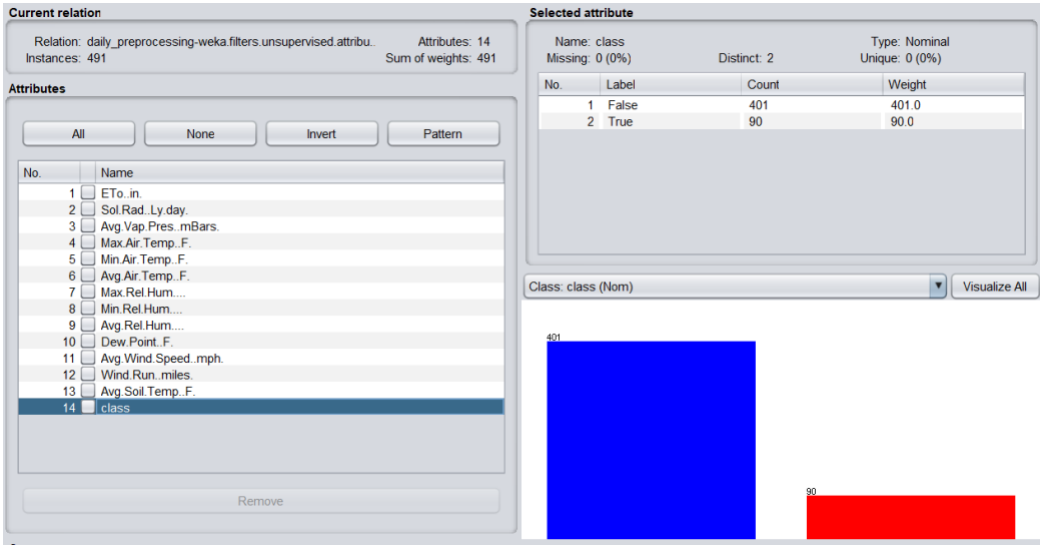


4) Correlation

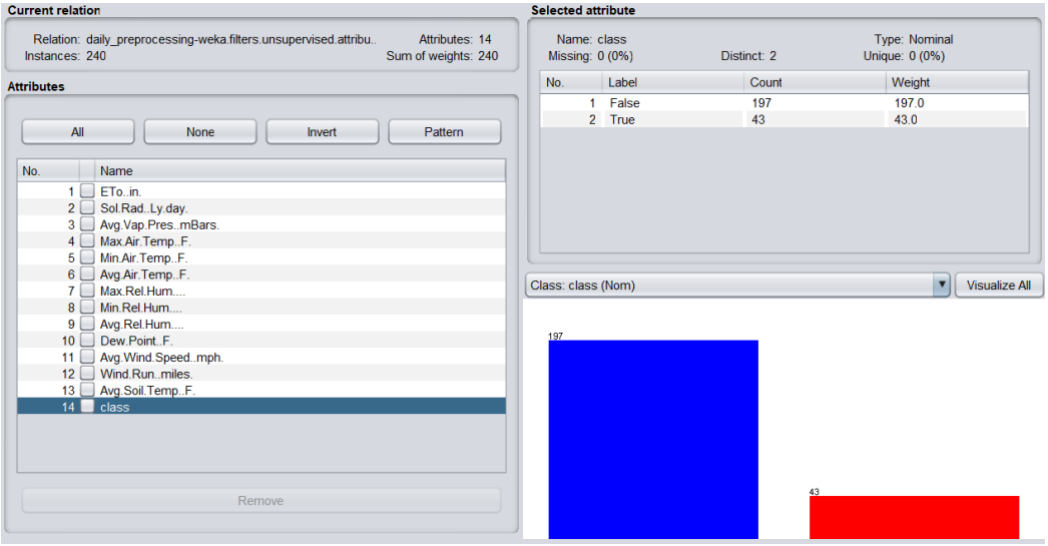


The class attribute is: 13.

The training dataset with selected attributes:



The test dataset with selected attributes:



2. Dataset creation manually

Manual Selection

Dataset Daily_5 is created by reducing attributes manually. Based on our understanding we have removed values:

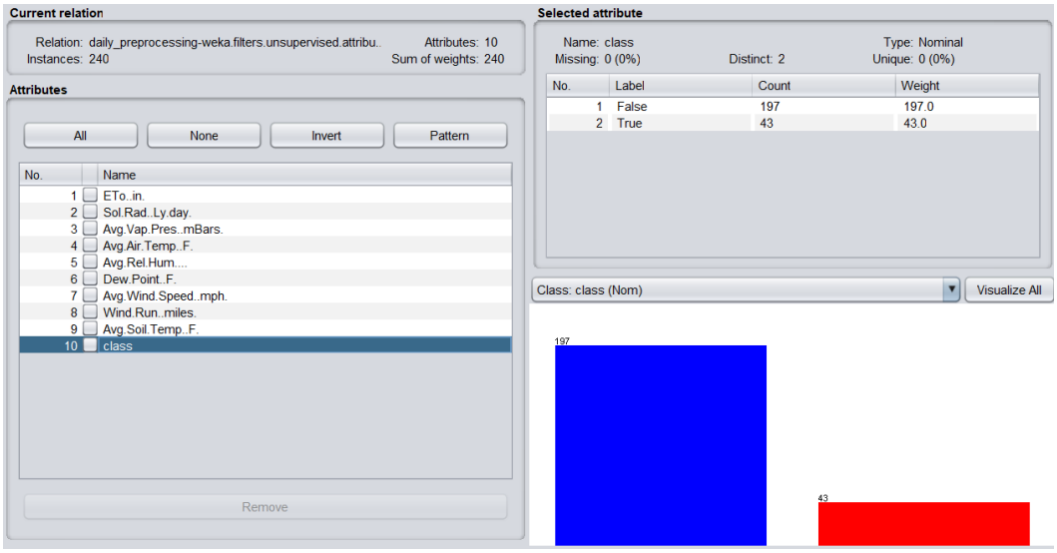
Max Rel Hum (%)	**Min Rel Hum (%)**	**Max Air Temp (F)**	**Min Air Temp (F)**

Because all of them have average values of these attributes already present in the dataset: **Avg Rel Hum (%)** and **Avg Air Temp (F)**.

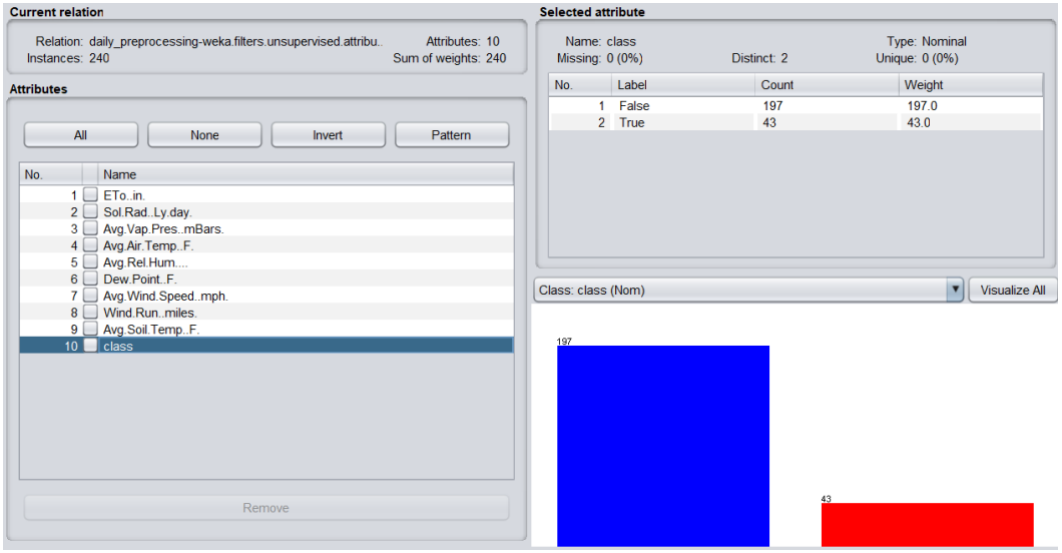
The attributes we keep are: **ETo (in)**, **Sol Rad (Ly/day)**, **Avg Vap Pres (mBars)**, **Avg Air Temp (F)**, **Avg Rel Hum (%)**, **Dew Point (F)**, **Avg Wind Speed (mph)**, **Wind Run (miles)**, **Avg Soil Temp (F)**, and class attribute **class**.

The class attribute is: 10.

The training dataset with selected attributes:



The test dataset with selected attributes:

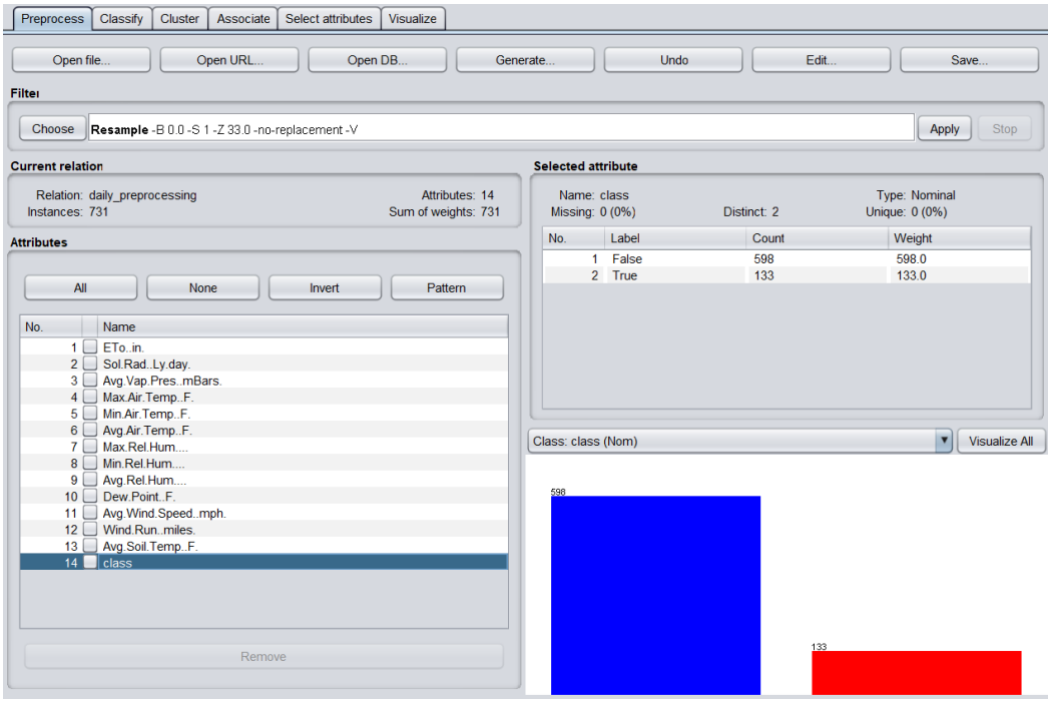


Attribute Reduction

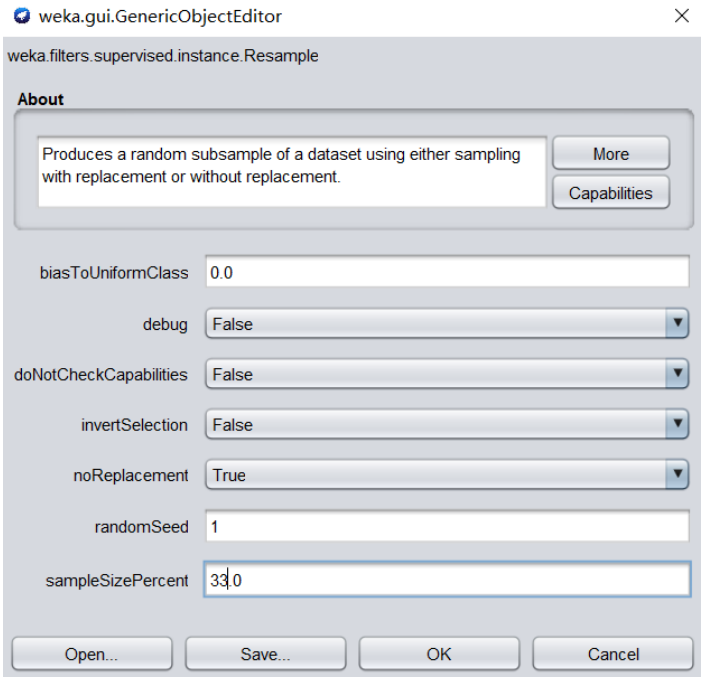
From the previous part we split all our dataset into training dataset and test dataset in a ratio of 67:33 using Resample attribute.

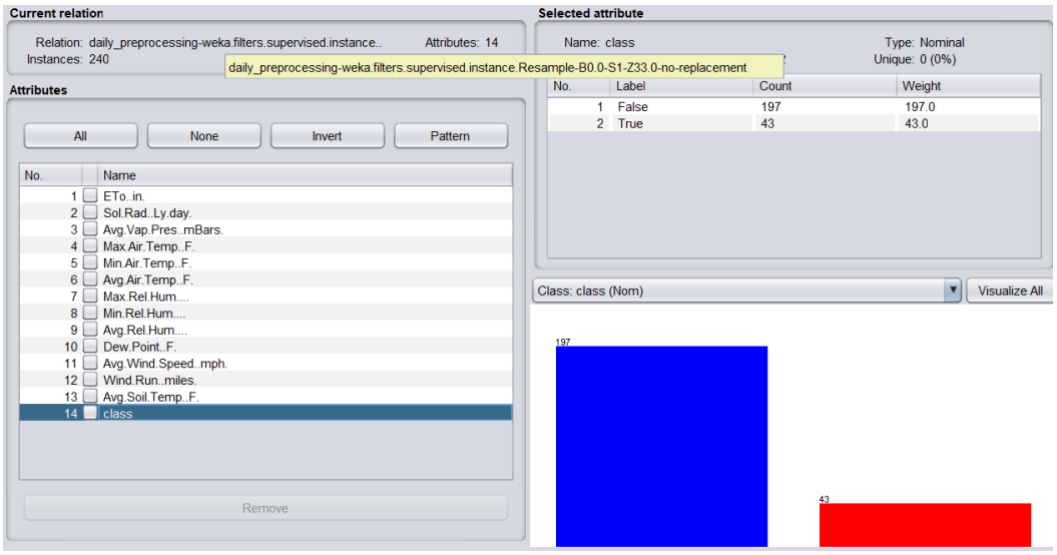
Screenshots of the dataset splitting process:

1. Opening the dataset in Weka

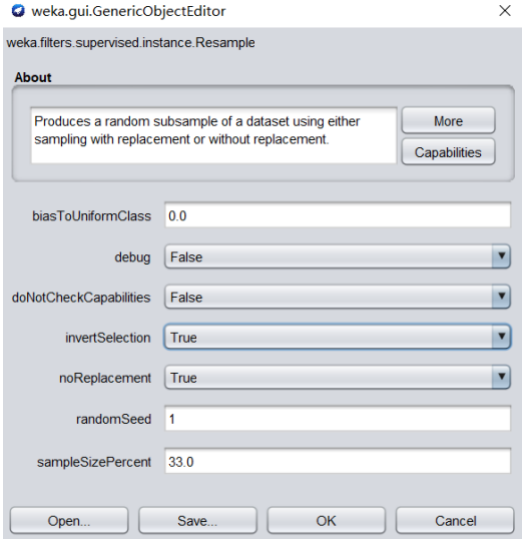


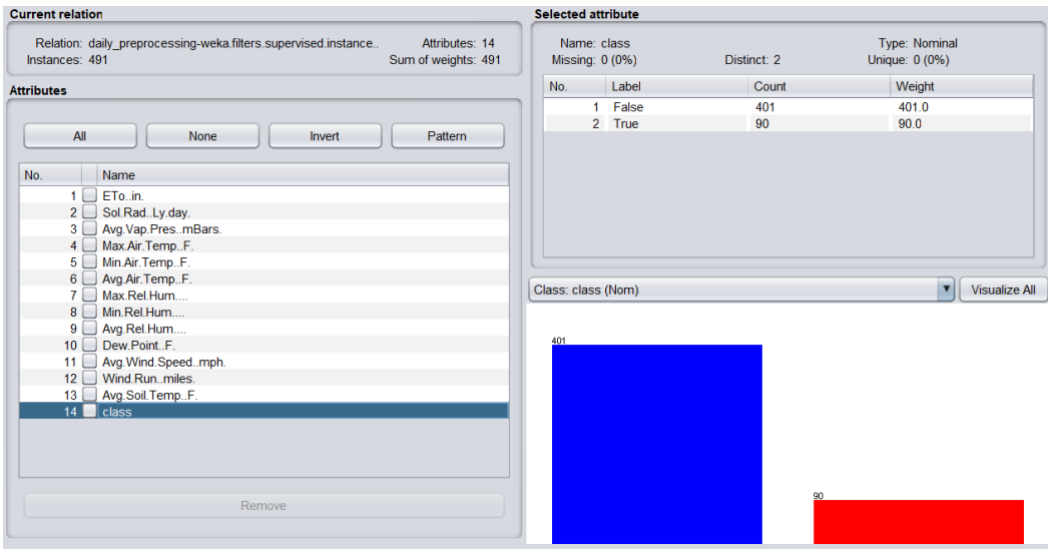
2. Creation of Test dataset



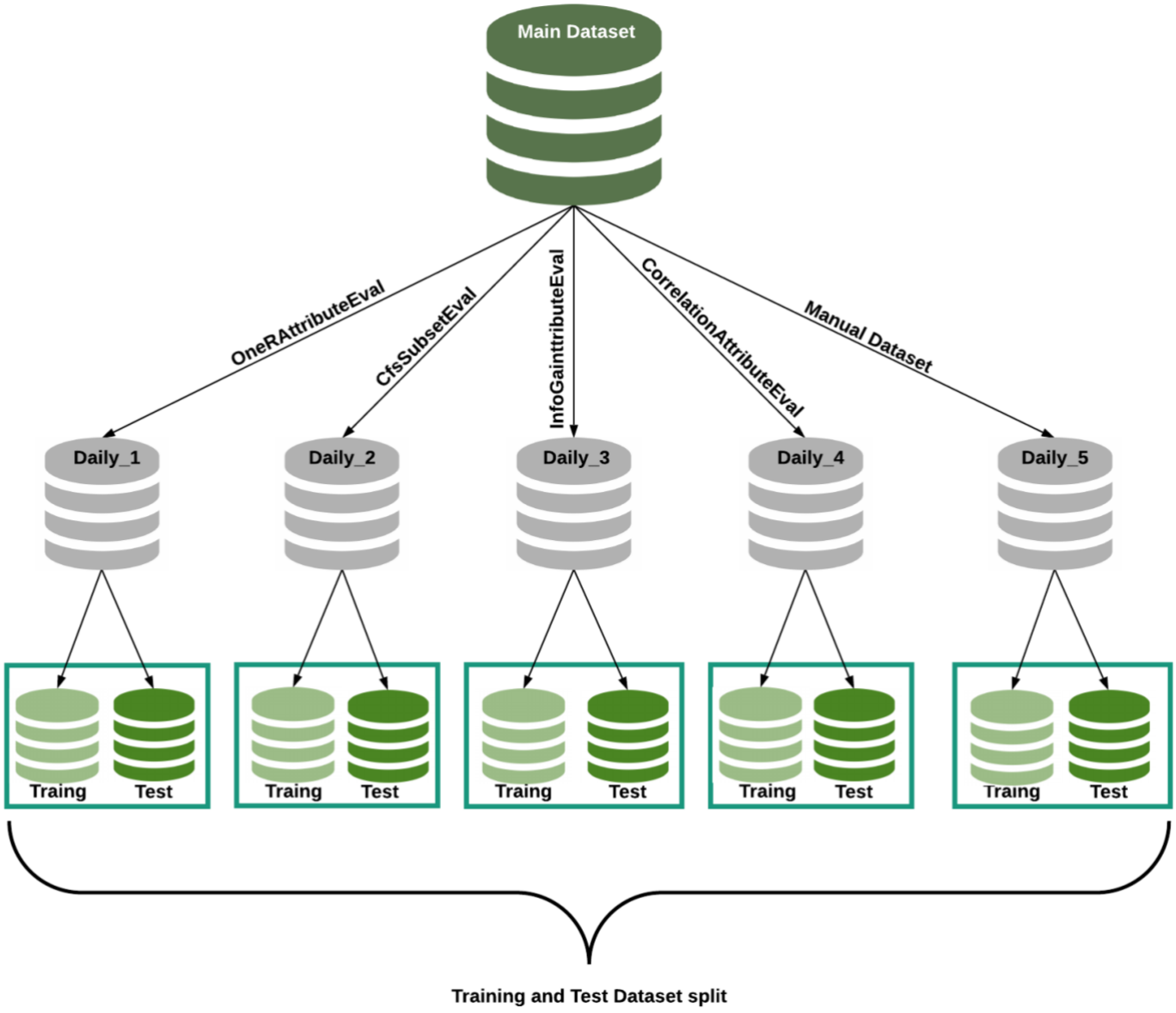


3. Creation of Training dataset





Mapping of Datasets created:



DATA MINING RESULT AND EVALUATION

To figure out the best model for classification process, we will be performing classification process using below classifiers:

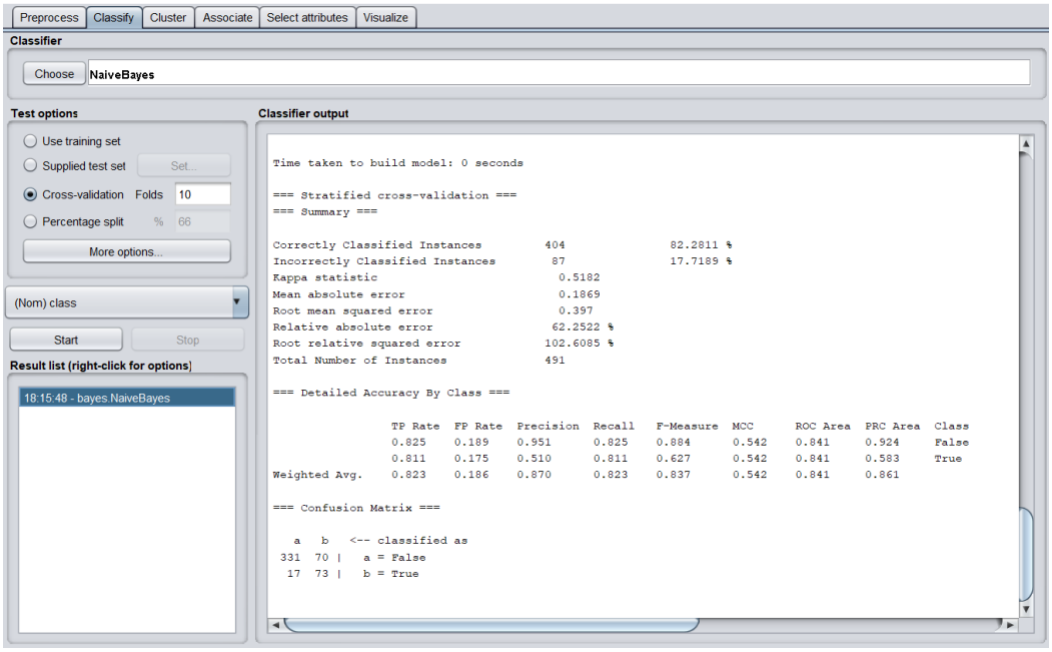
NaiveBayes
SGD
J48
Random Forest
IBK(K=10)

We will be building models using Training datasets and testing them using the test datasets created after splitting the datasets in 67:33 ratio.

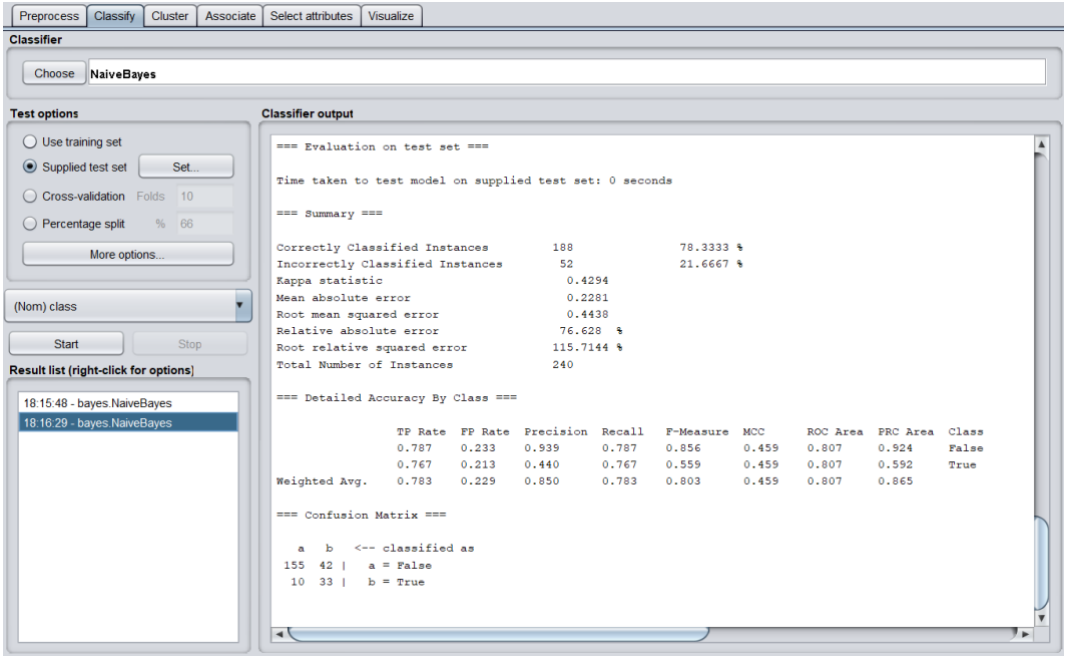
Test Performed on Daily_1 Dataset

1. **Naïve Bayes**

   Build the model with training dataset:

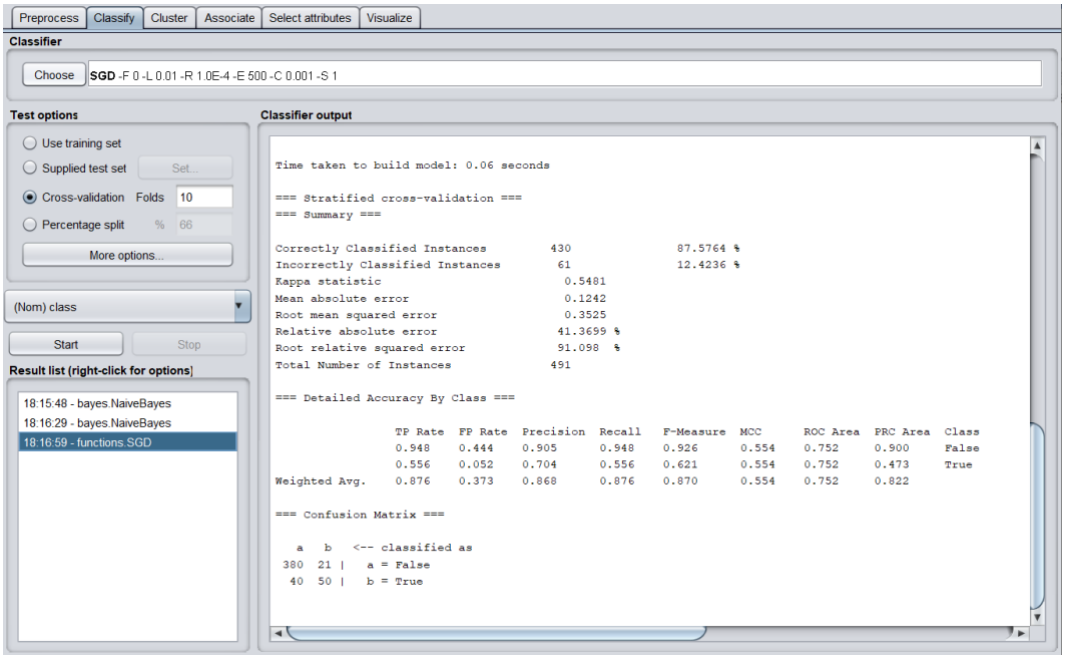


Test it with test dataset:

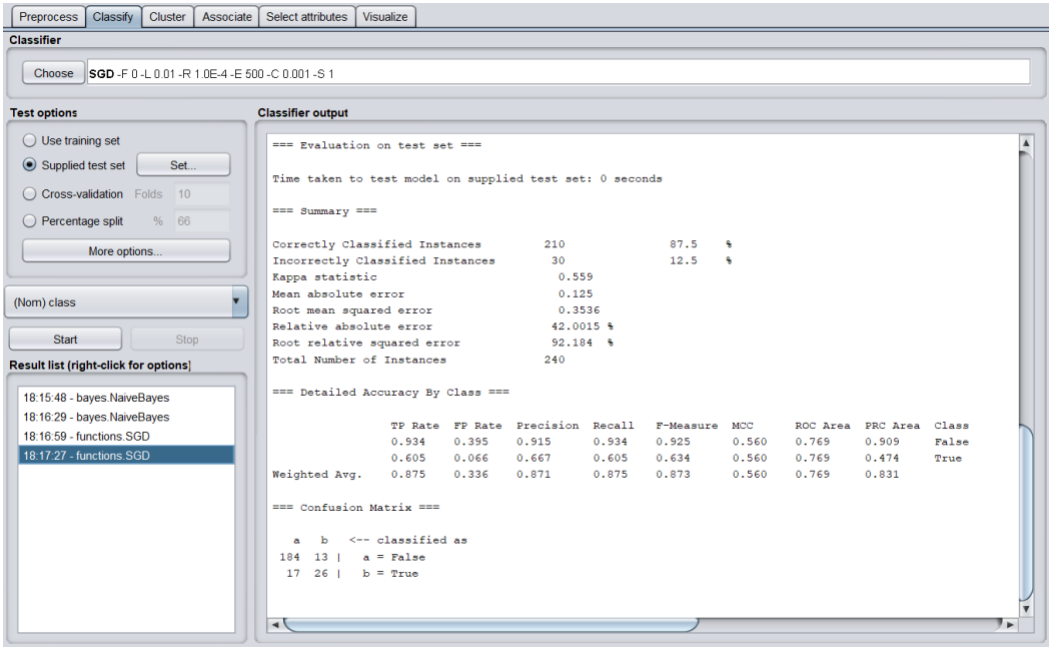


2. SGD

Build the model with training dataset:

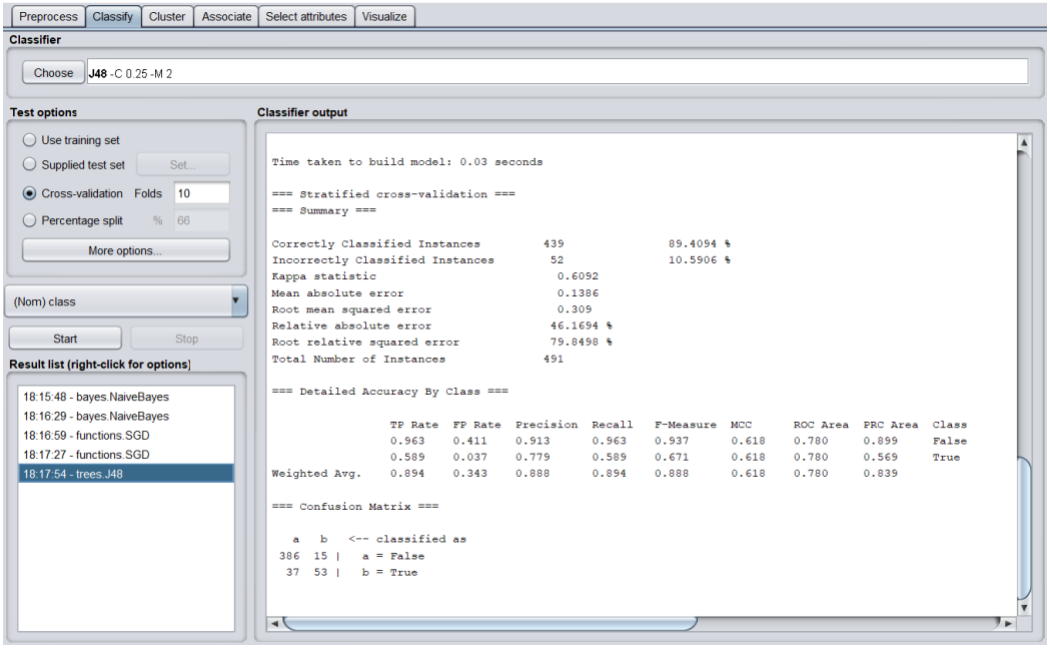


Test it with test dataset:

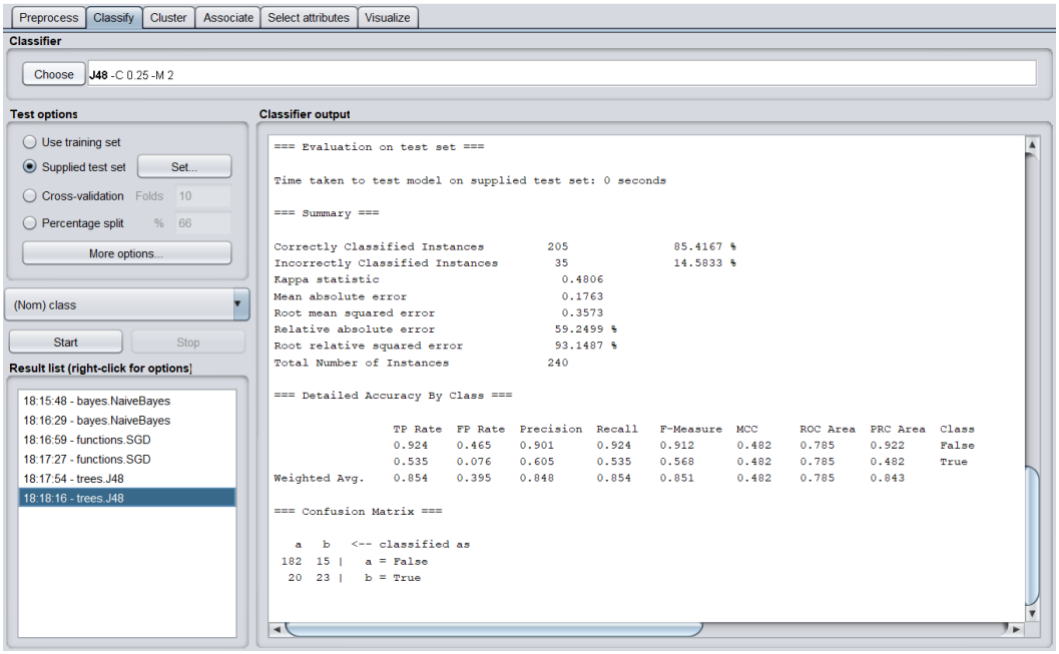


3. J48

Build the model with training dataset:

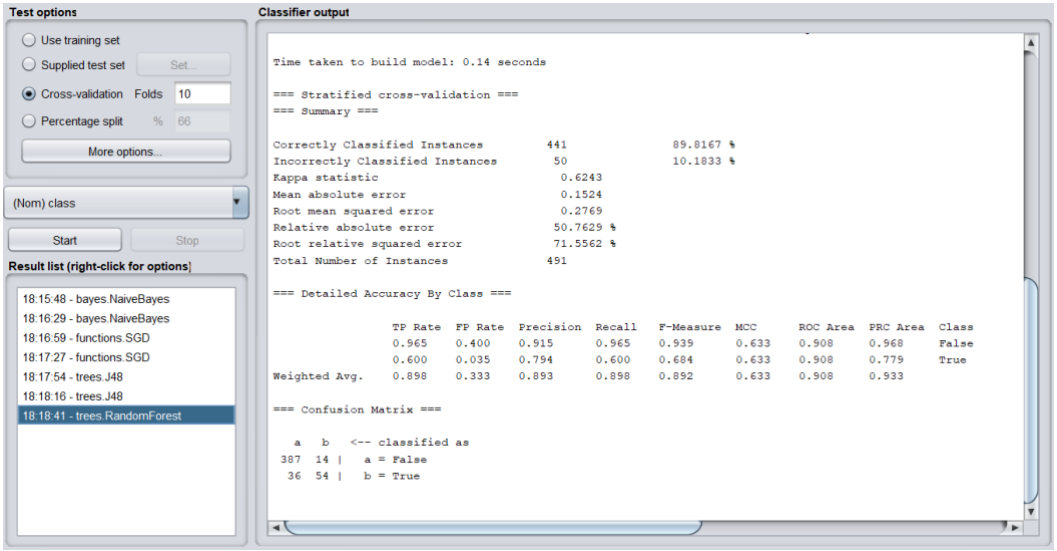


Test it with test dataset:

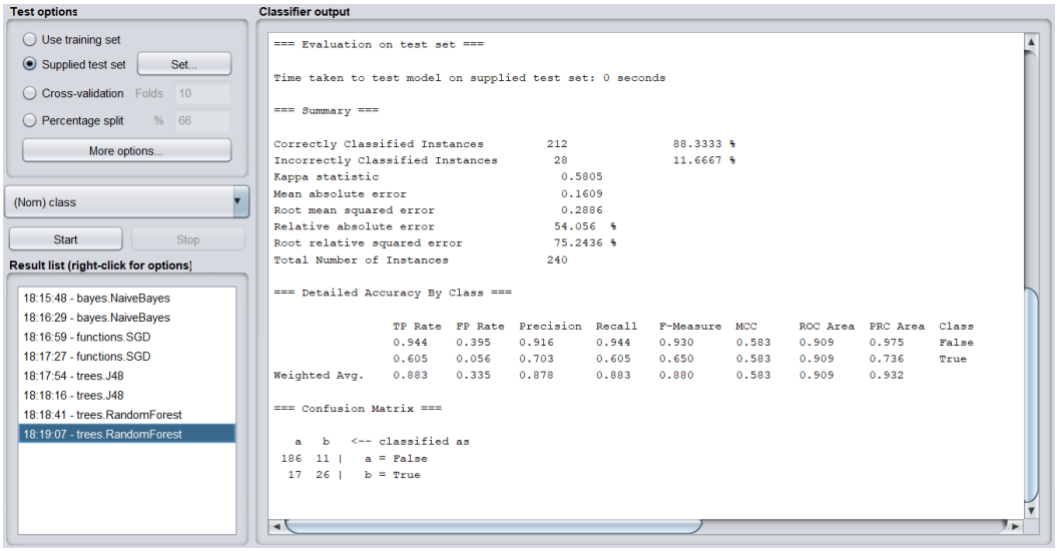


4. Random Forest

Build the model with training dataset:

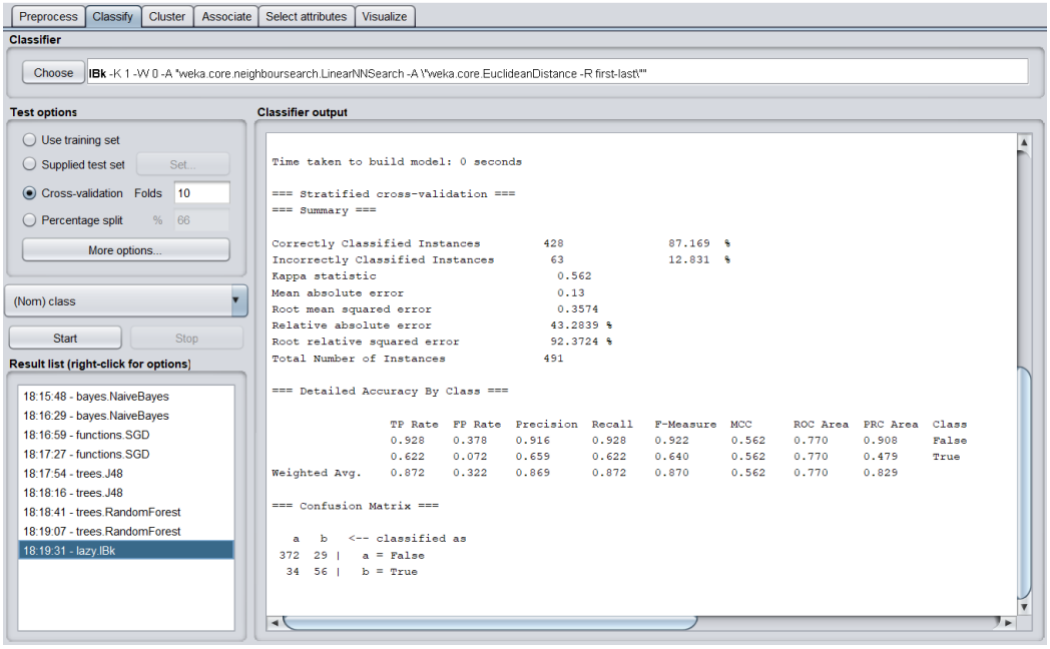


Test it with test dataset:

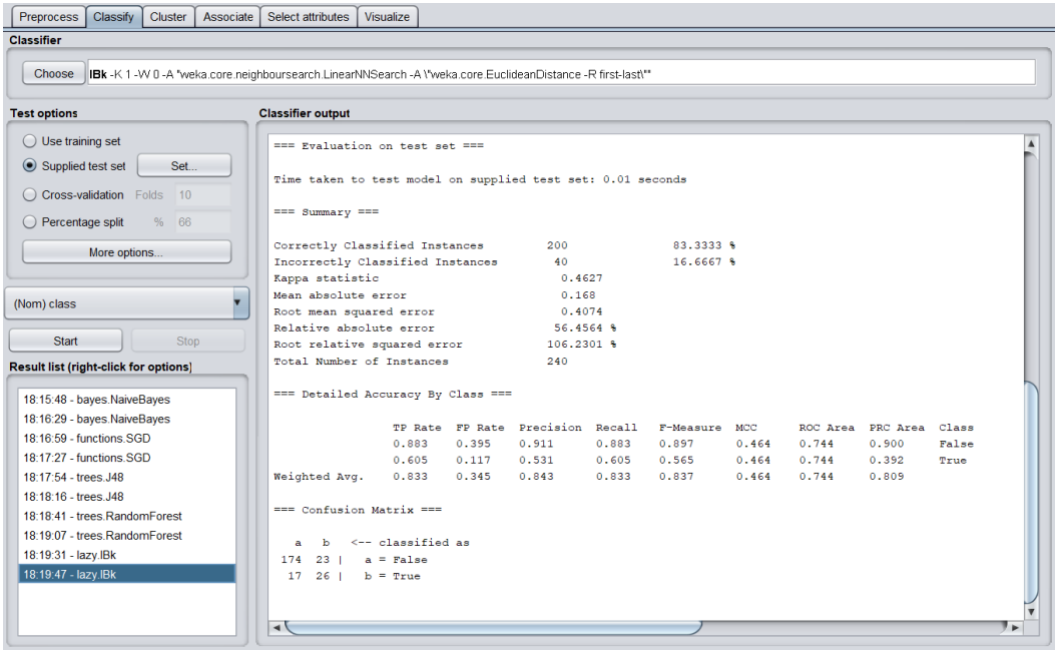


5. IBk (k = 10)

Build the model with training dataset:



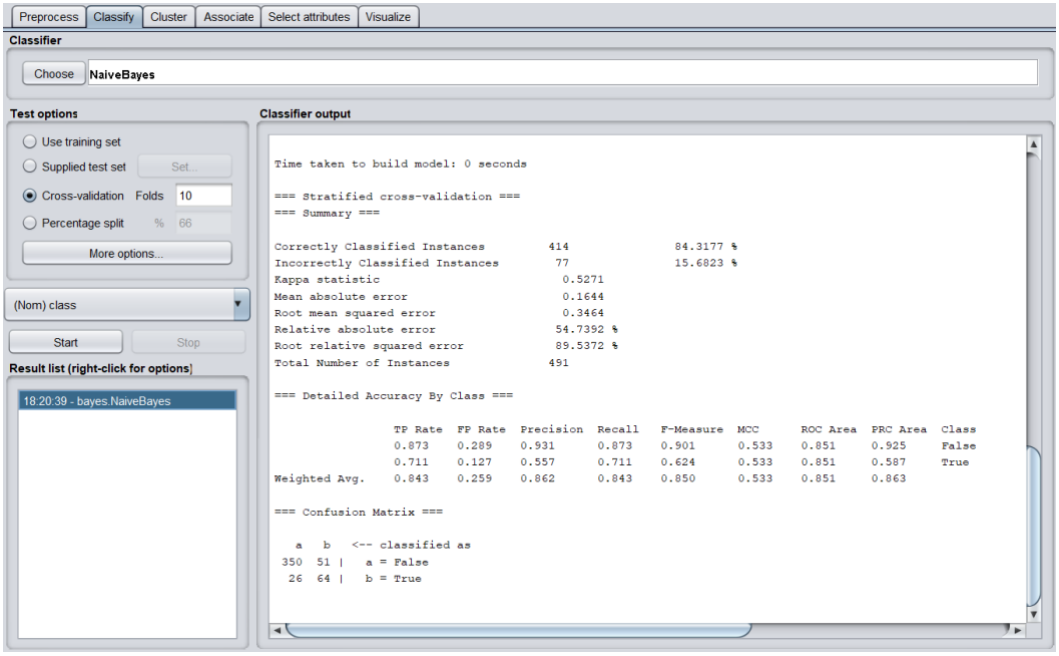
Test it with test dataset:



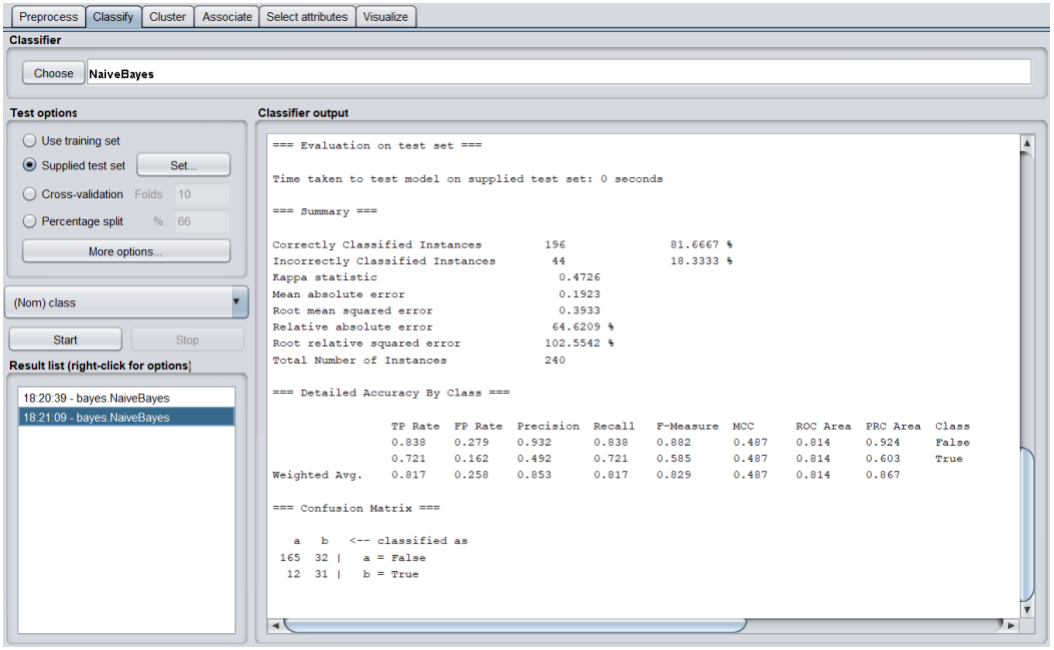
Test Performed on Daily_2 Dataset

1. Naïve Bayes

Build the model with training dataset:

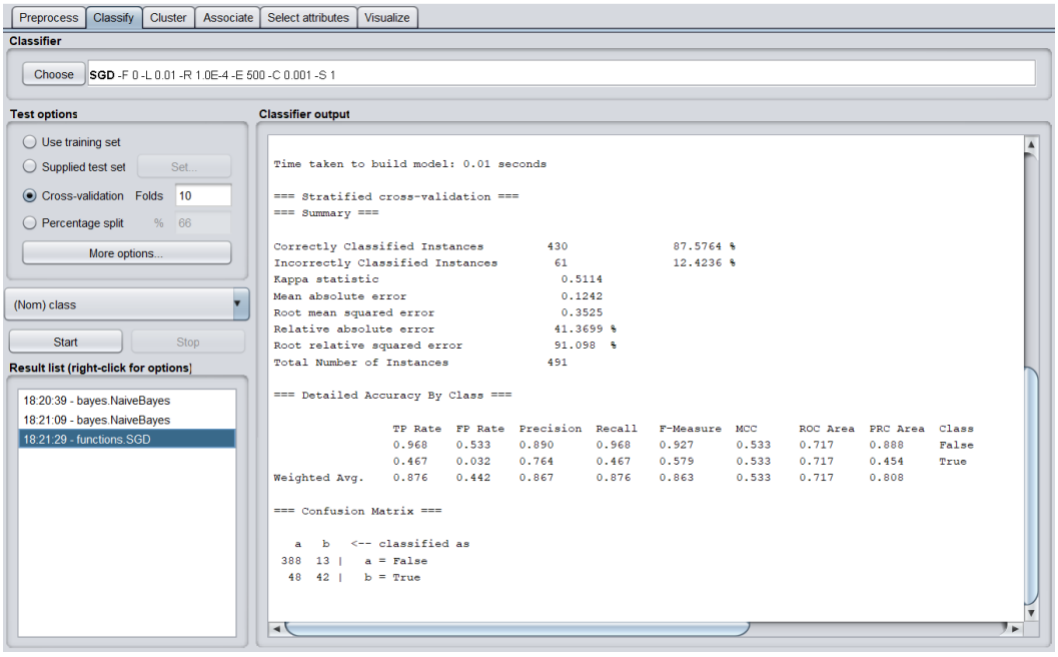


Test it with test dataset:

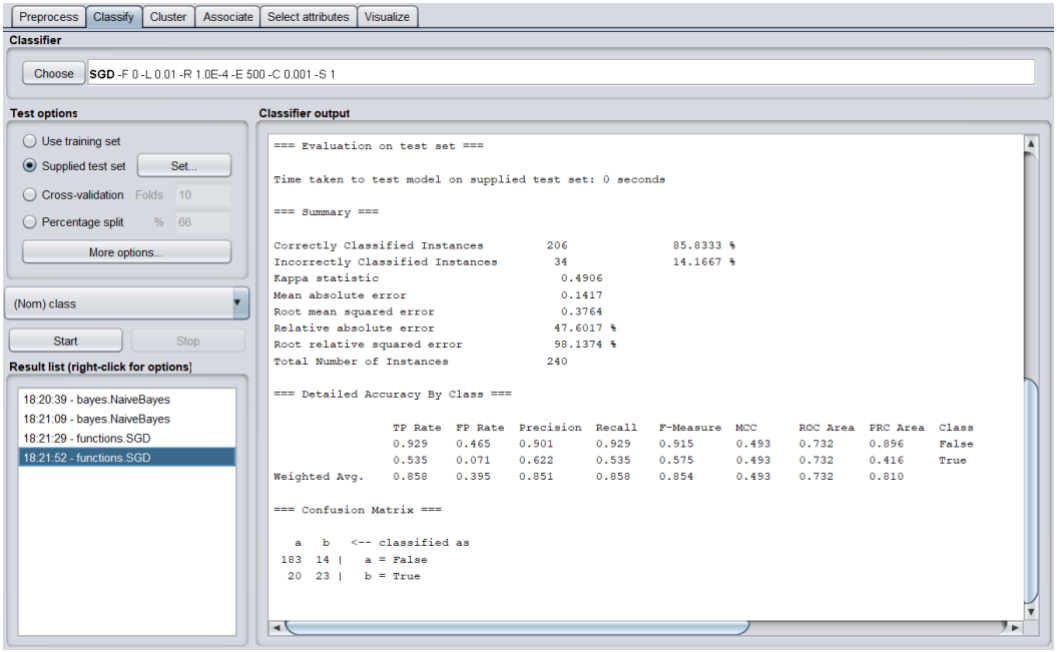


2. SGD

Build the model with training dataset:

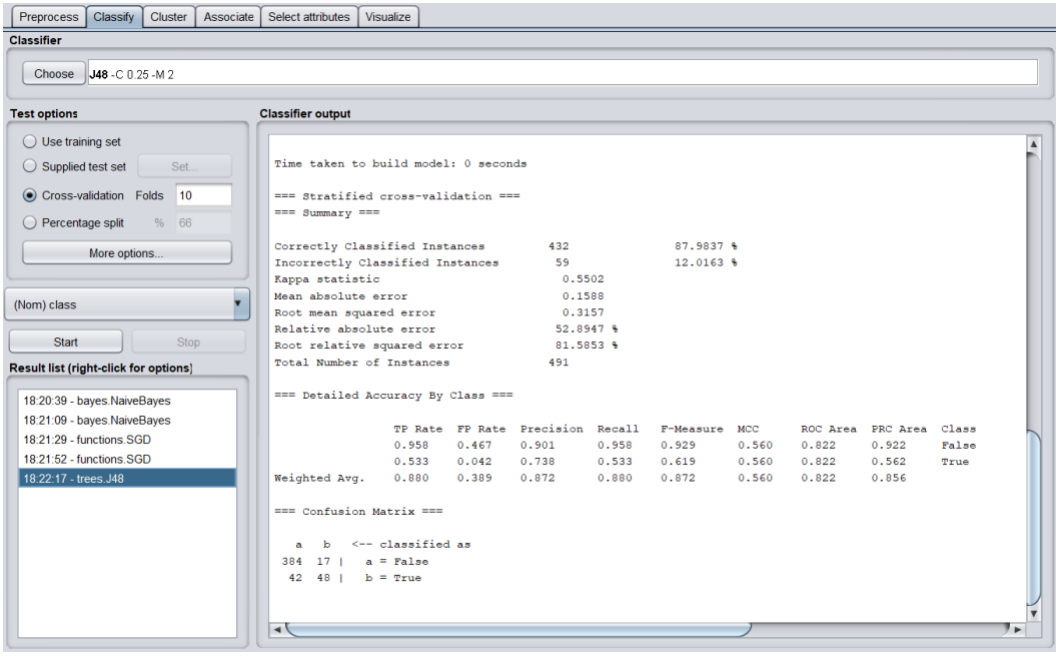


Test it with test dataset:

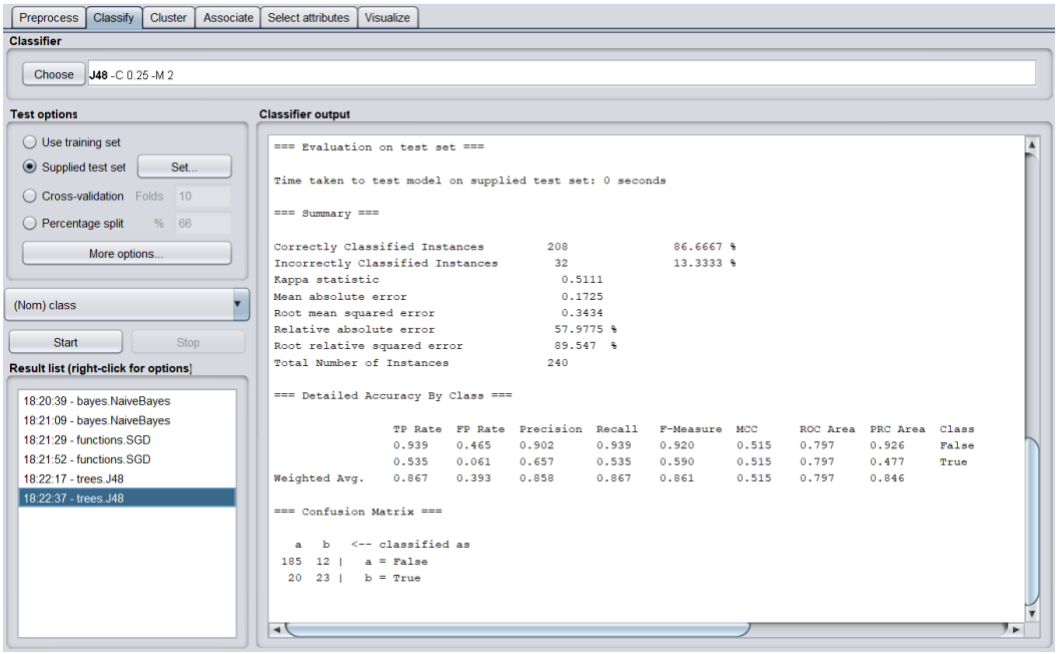


3. J48

Build the model with training dataset:

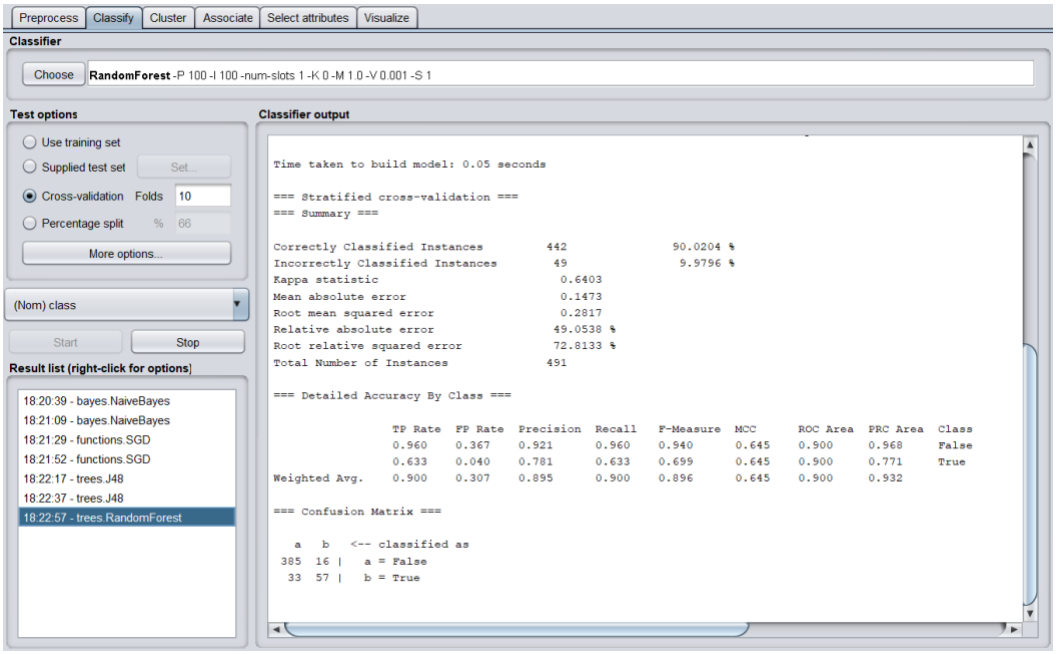


Test it with test dataset:

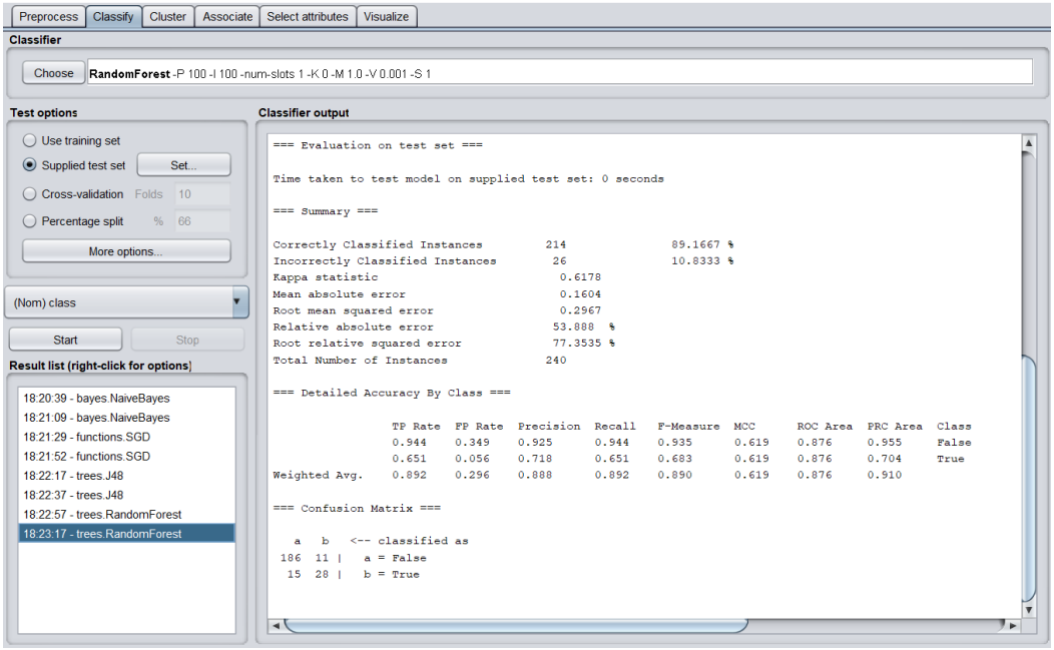


4. Random Forest

Build the model with training dataset:

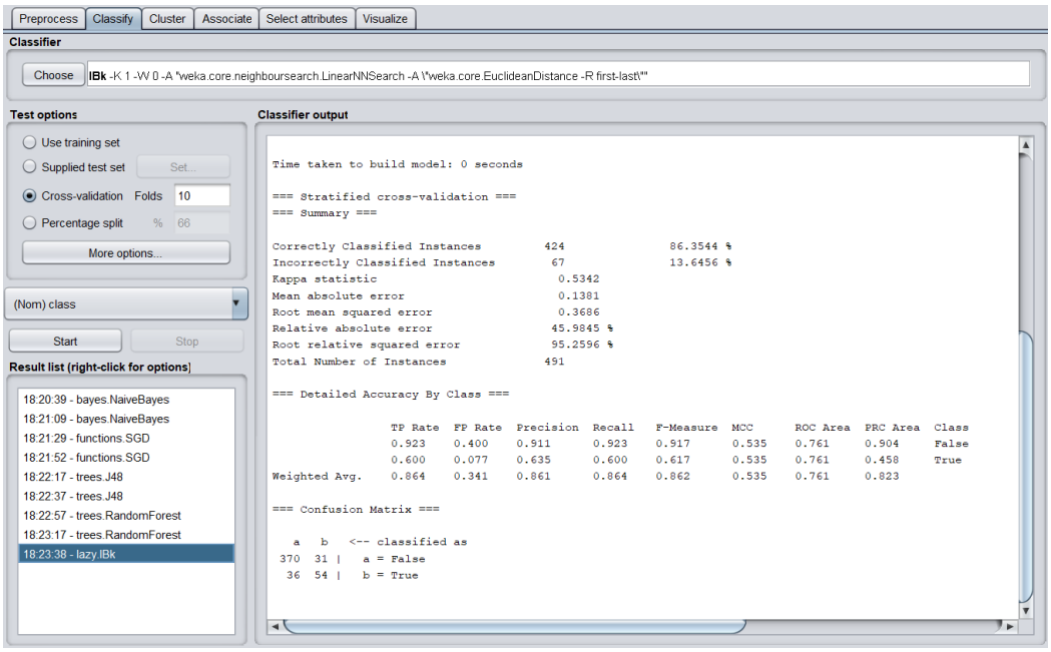


Test it with test dataset:

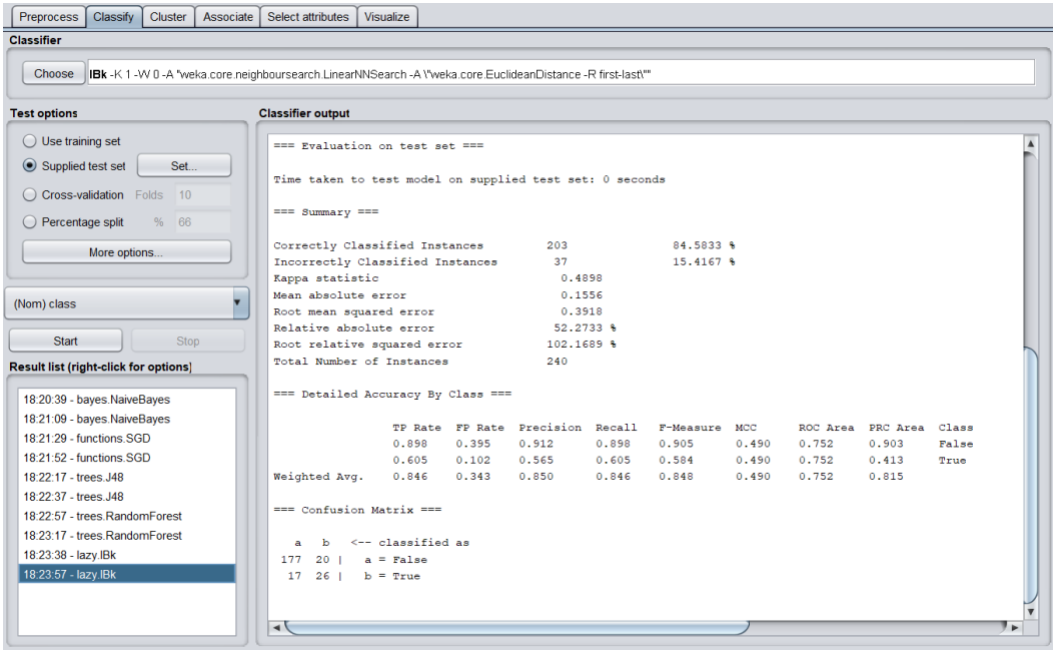


5. IBk (k = 10)

Build the model with training dataset:



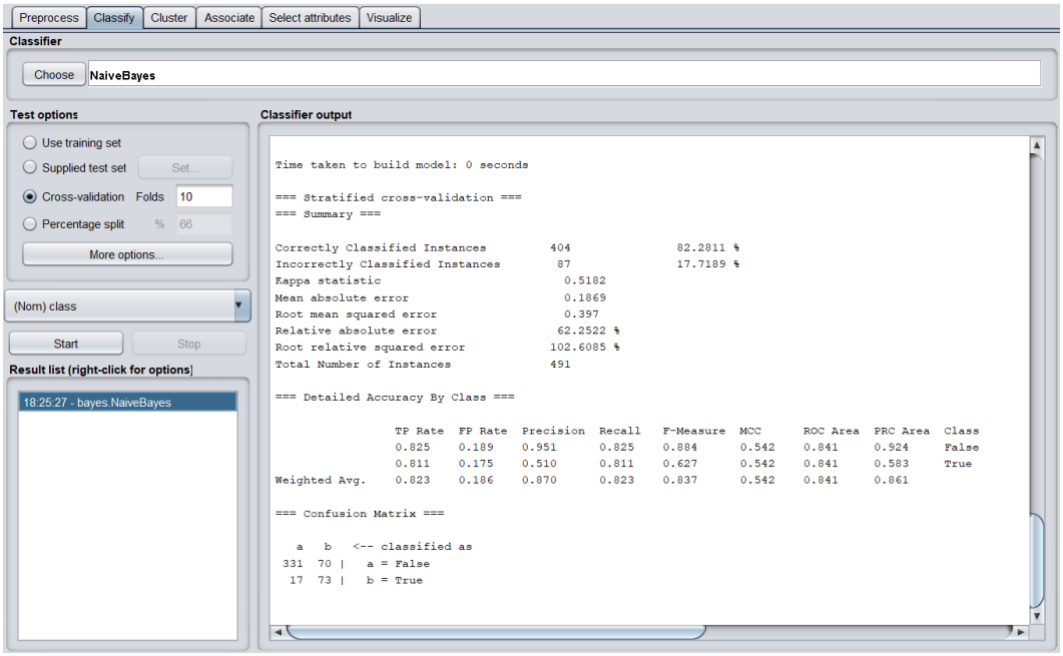
Test it with test dataset:



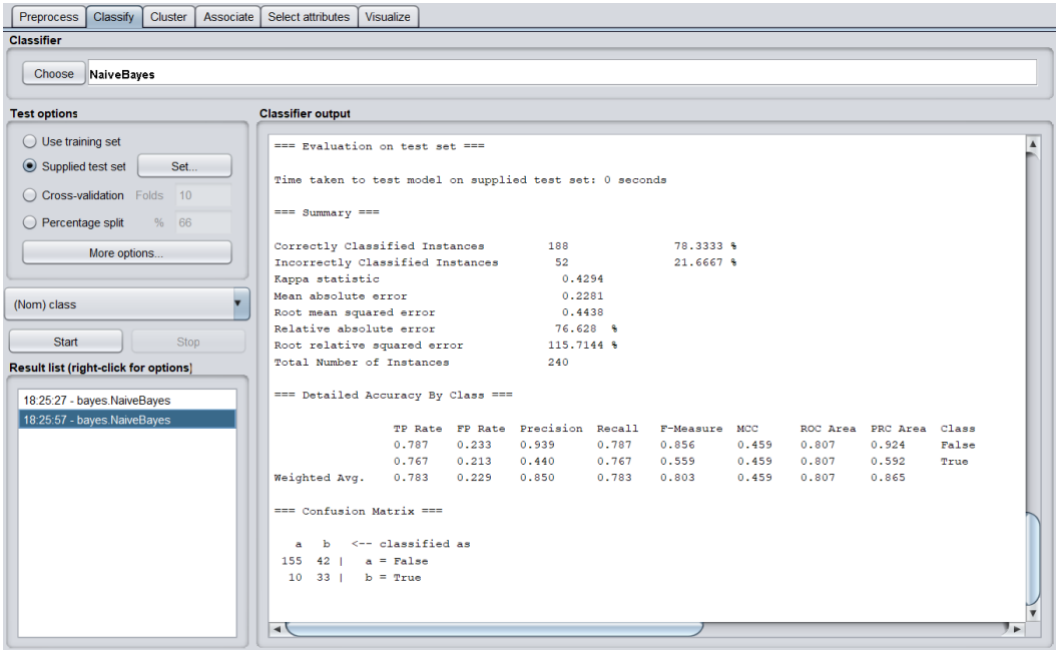
Test Performed on Daily_3 Dataset

1. Naïve Bayes

Build the model with training dataset:

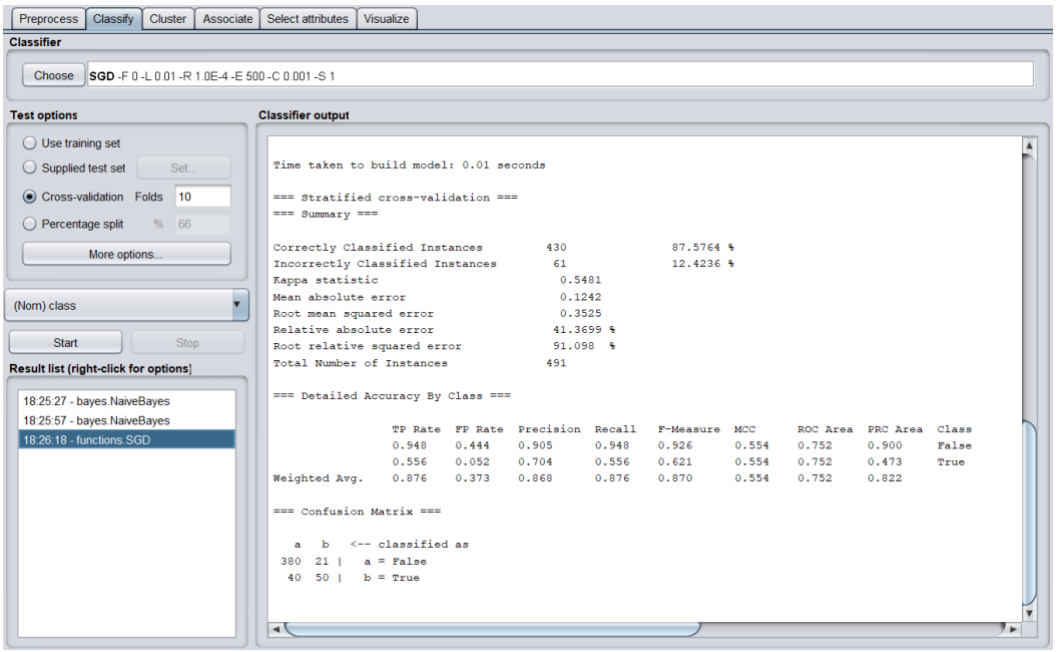


Test it with test dataset:

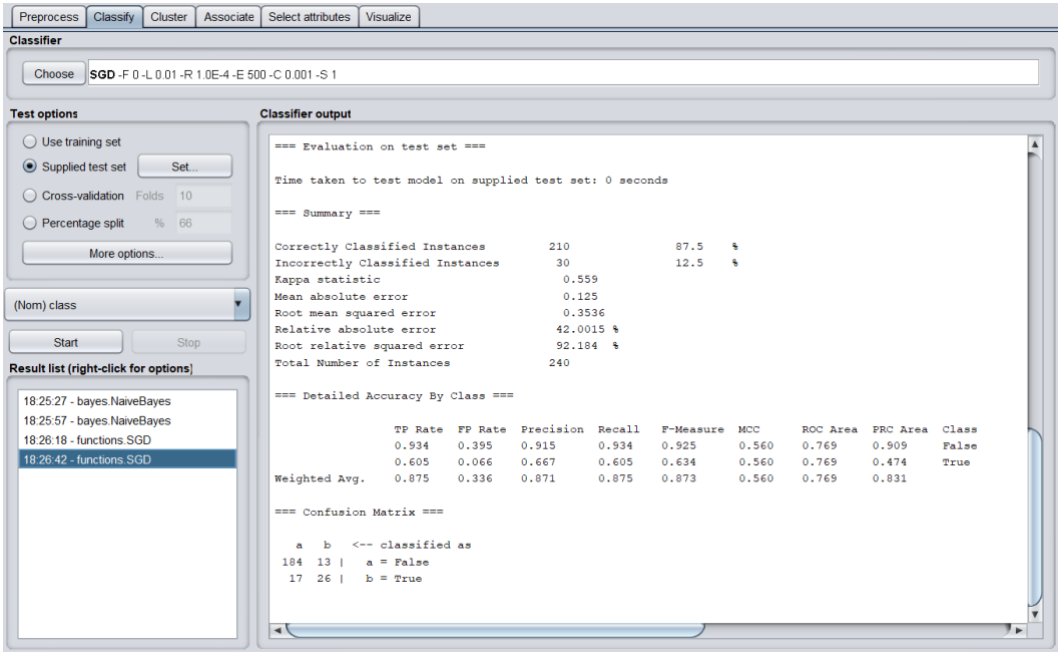


2. SGD

Build the model with training dataset:

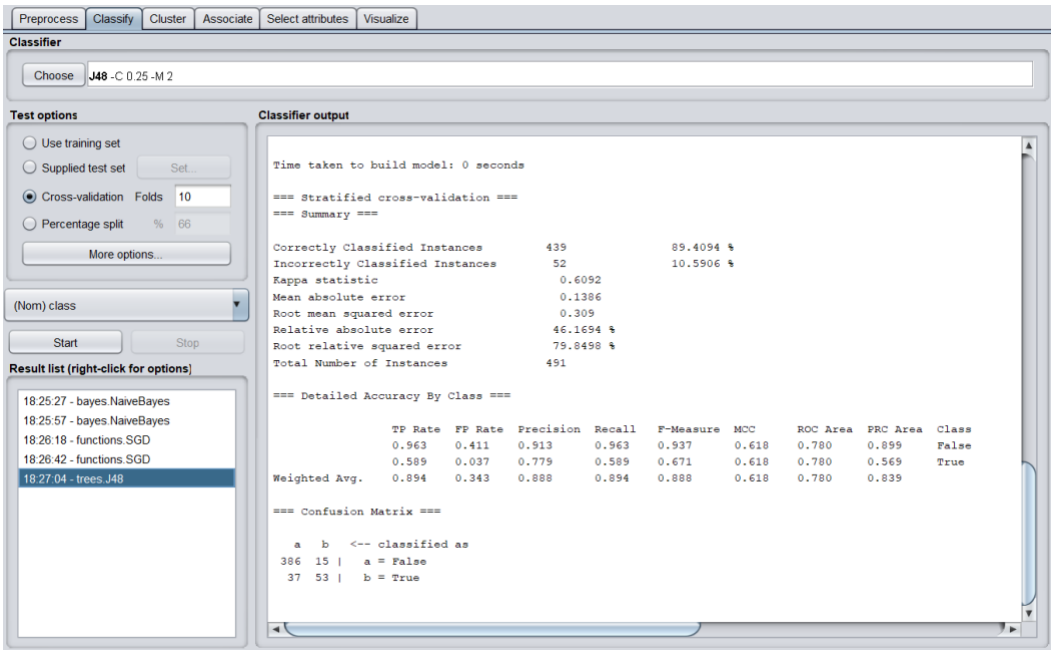


Test it with test dataset:

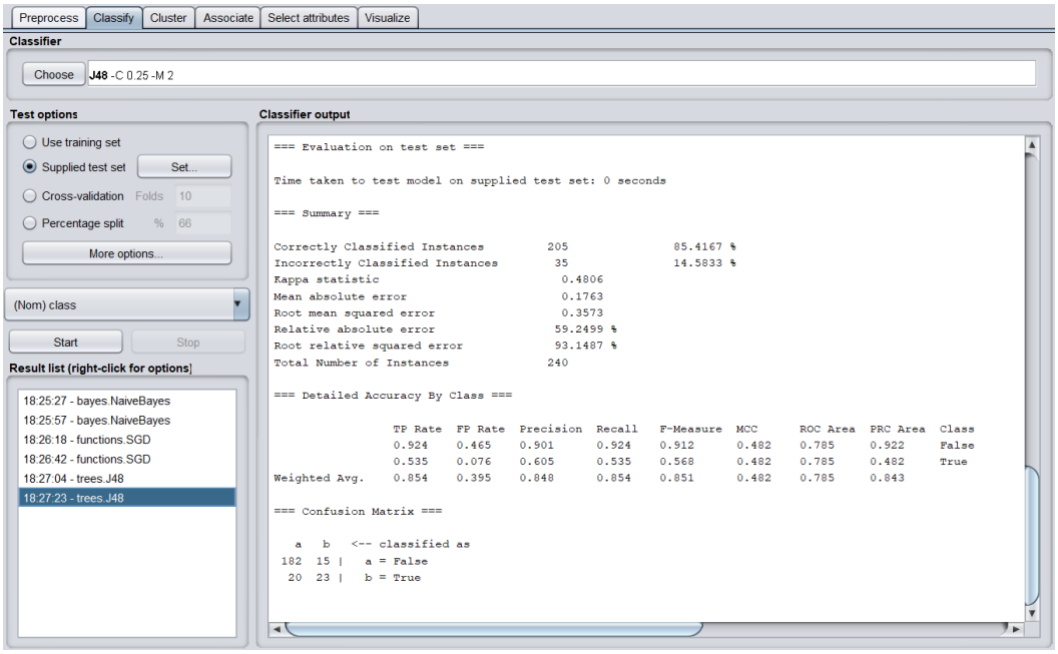


3. J48

Build the model with training dataset:

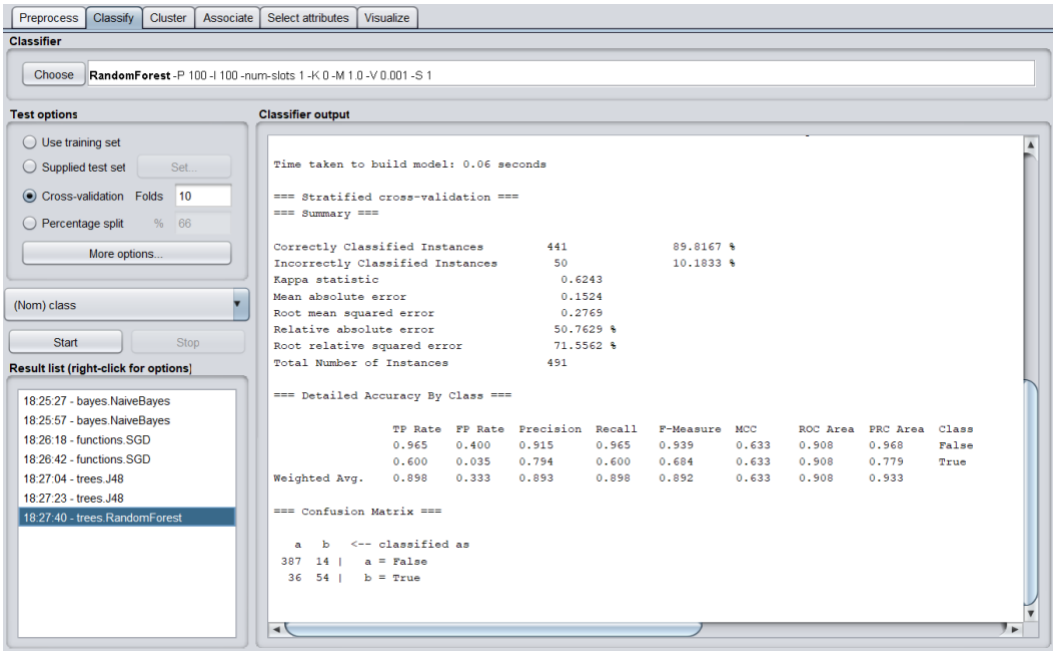


Test it with test dataset:

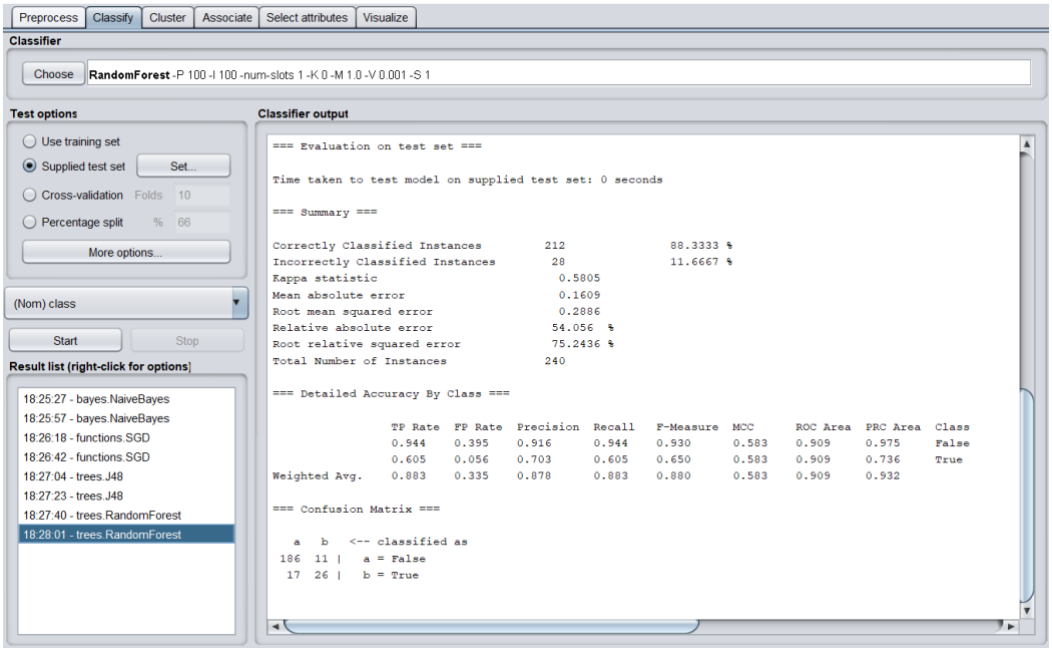


4. Random Forest

Build the model with training dataset:

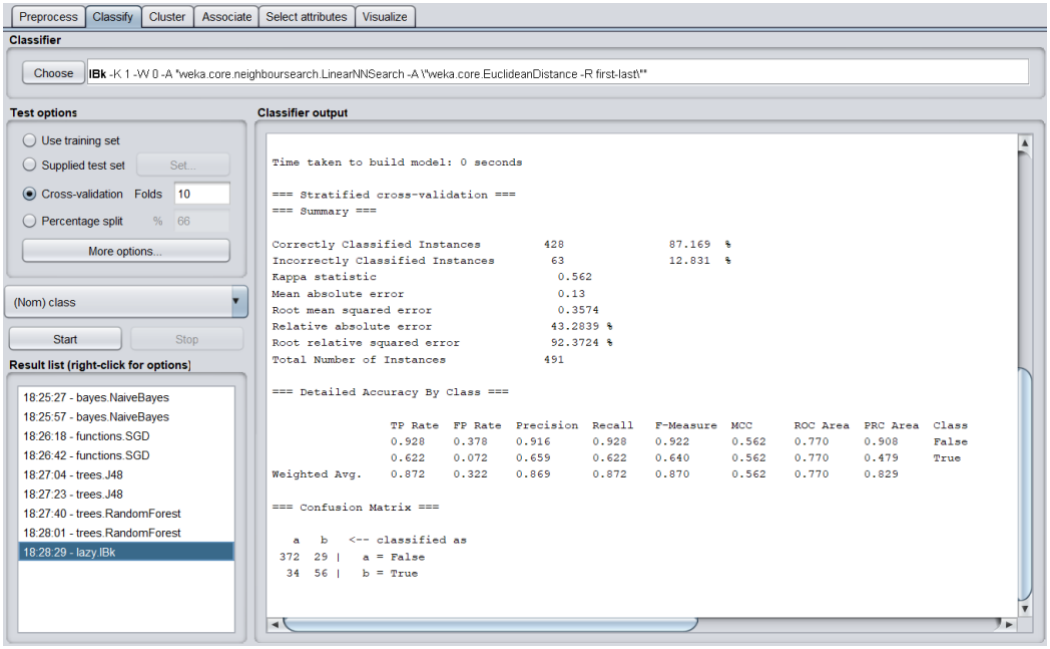


Test it with test dataset:

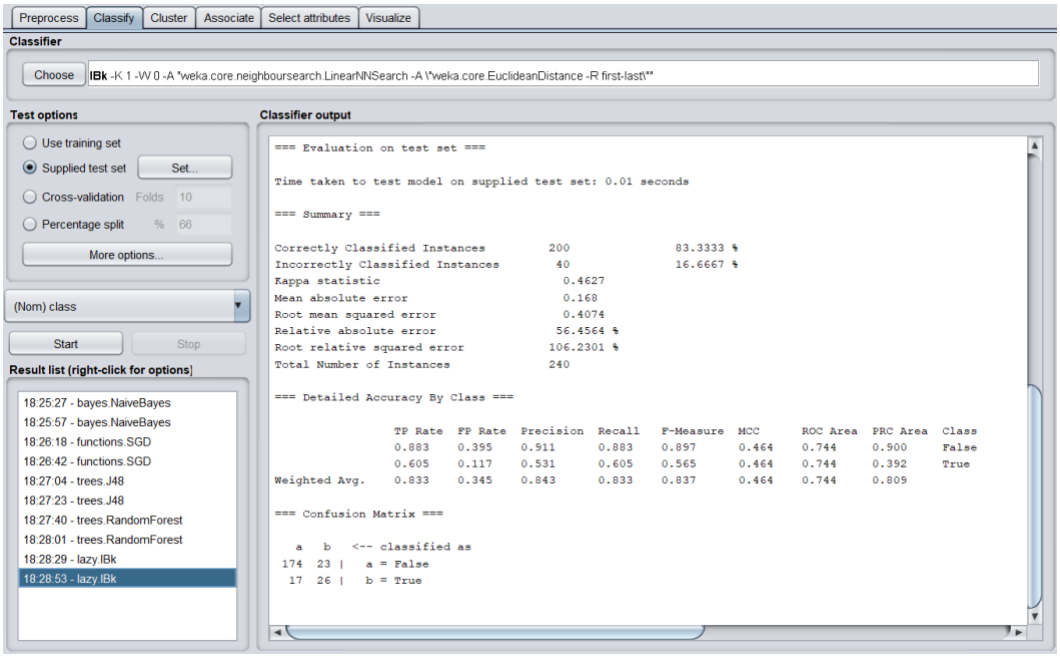


5. IBk (k = 10)

Build the model with training dataset:



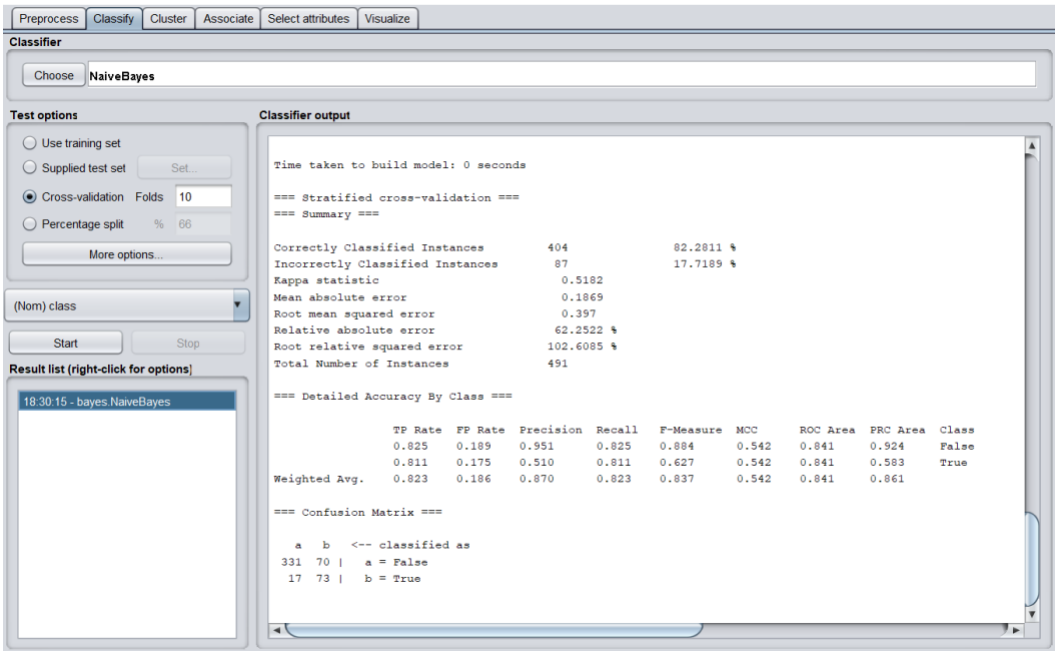
Test it with test dataset:



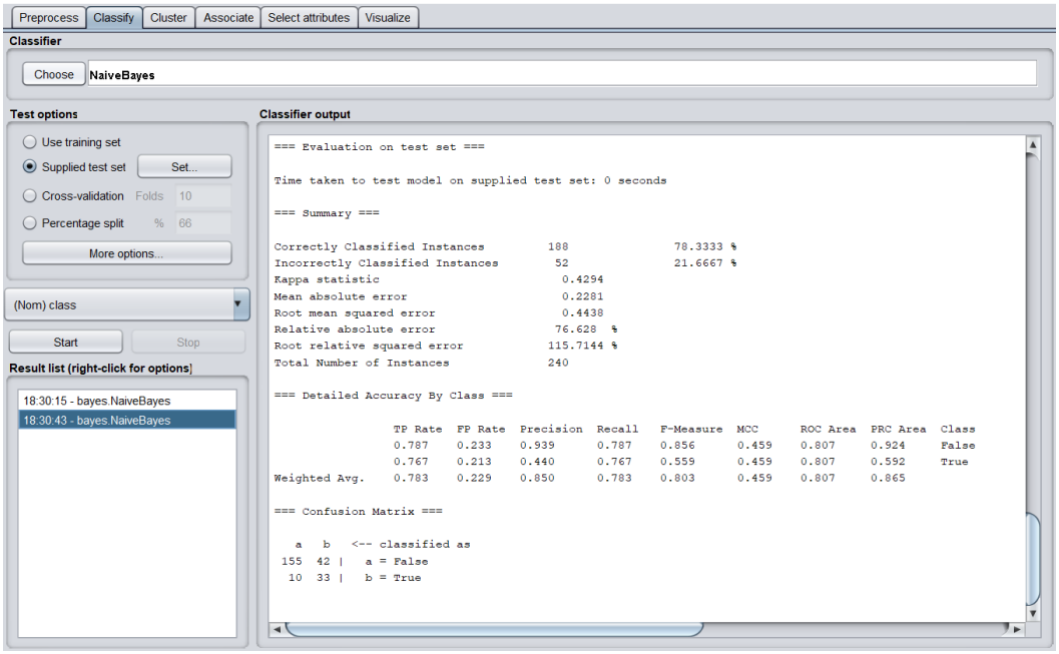
Test Performed on Daily_4 Dataset

1. Naïve Bayes

Build the model with training dataset:

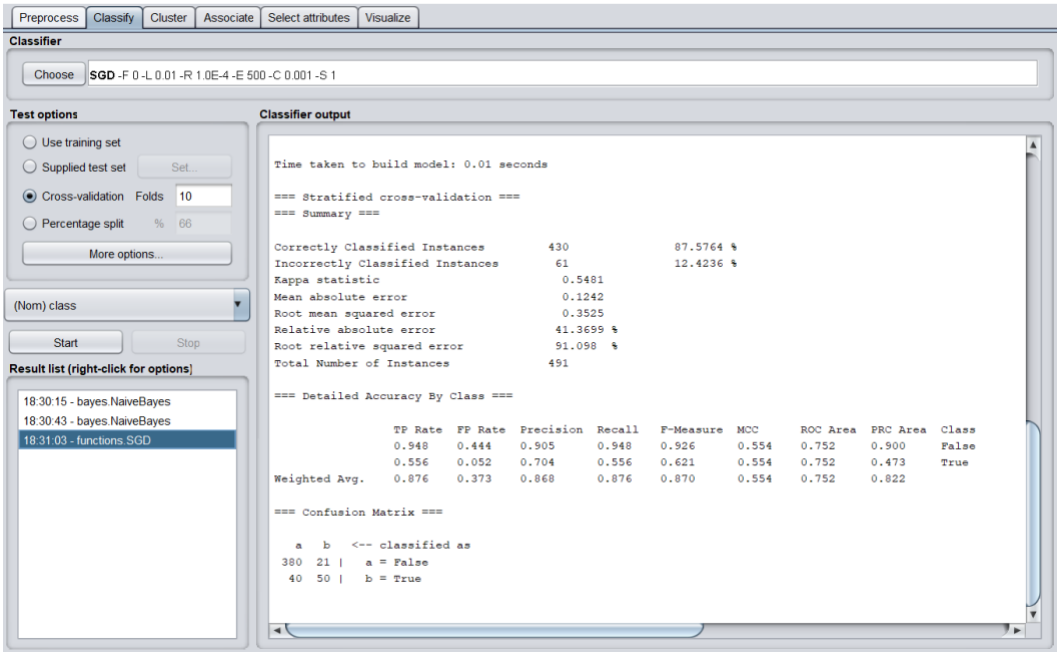


Test it with test dataset:

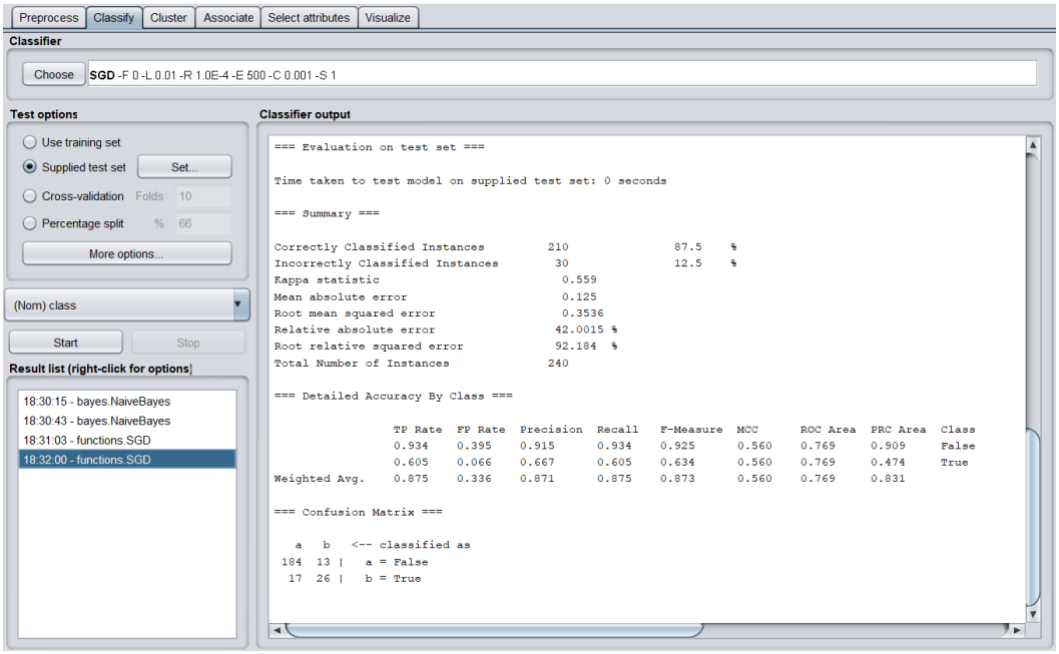


2. SGD

Build the model with training dataset:

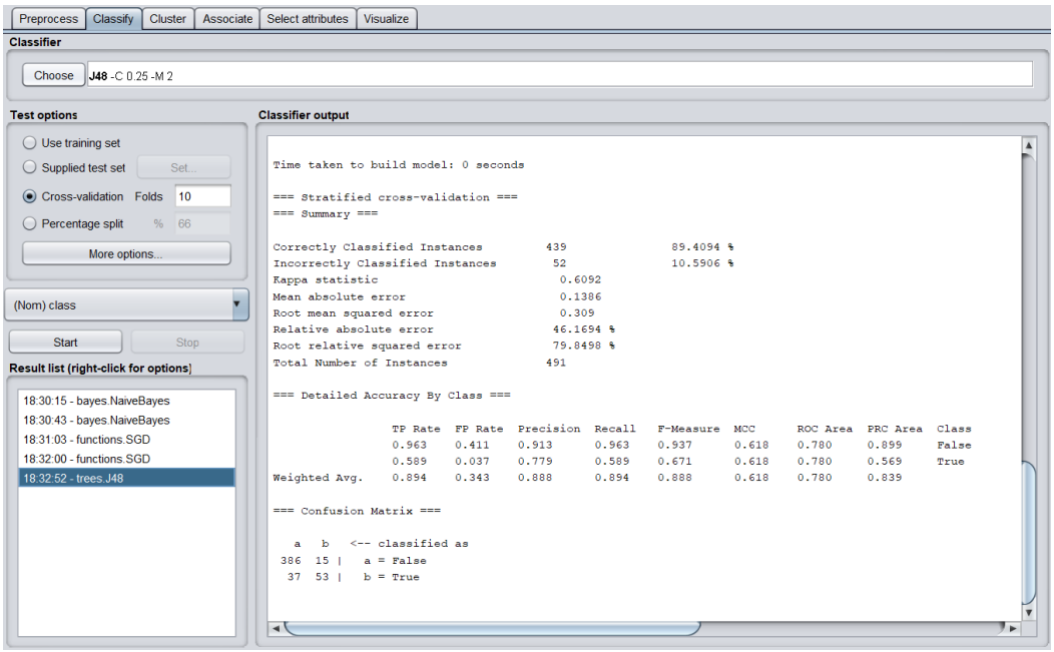


Test it with test dataset:

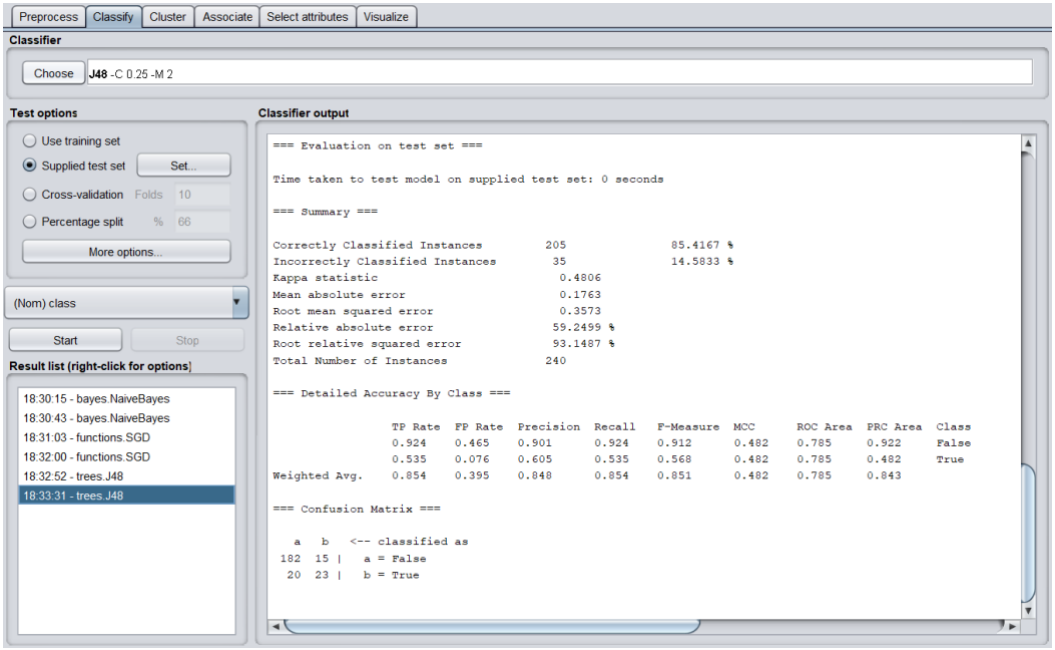


3. J48

Build the model with training dataset:

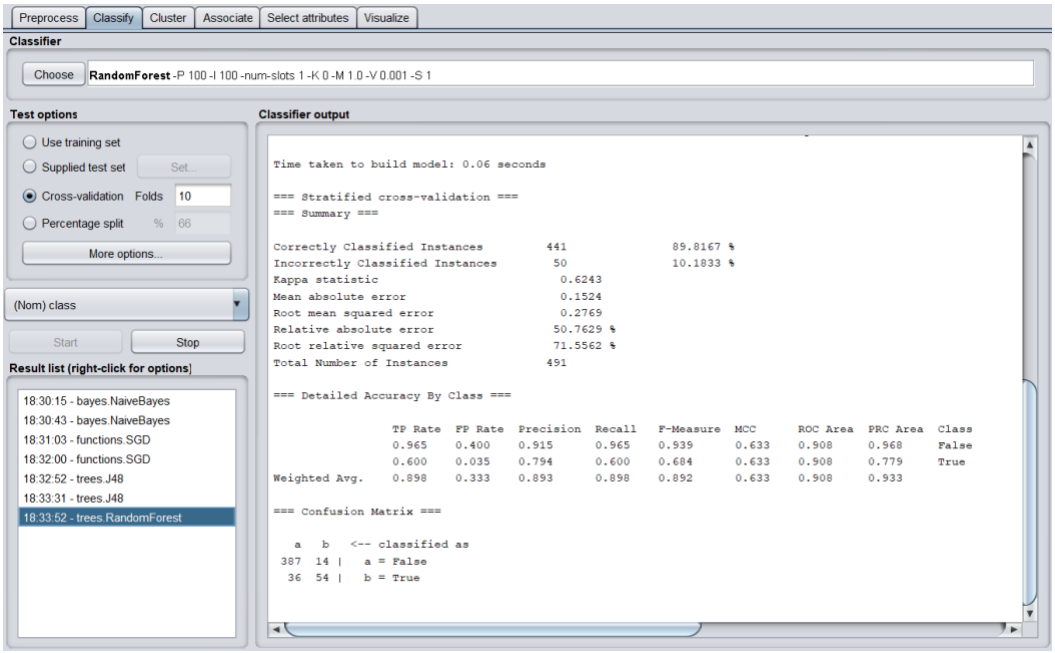


Test it with test dataset:

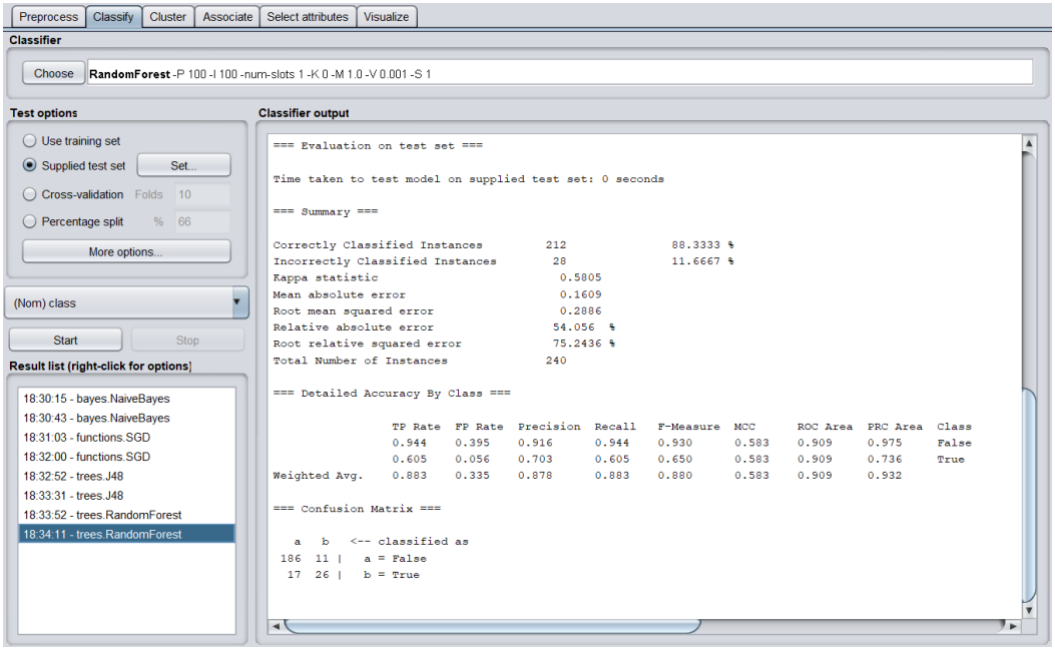


4. Random Forest

Build the model with training dataset:

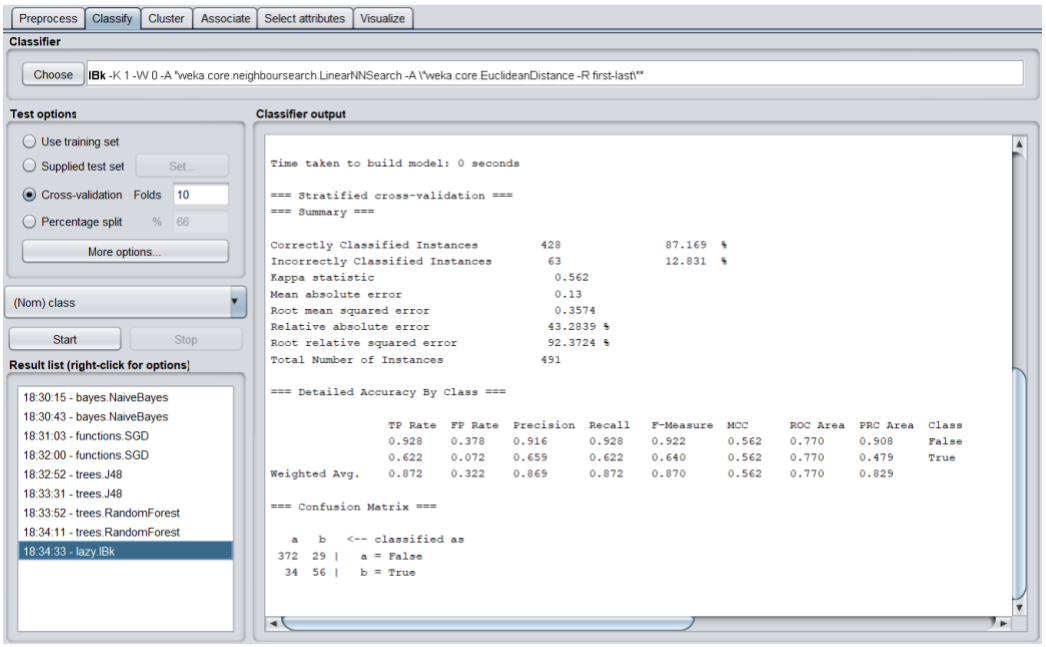


Test it with test dataset:

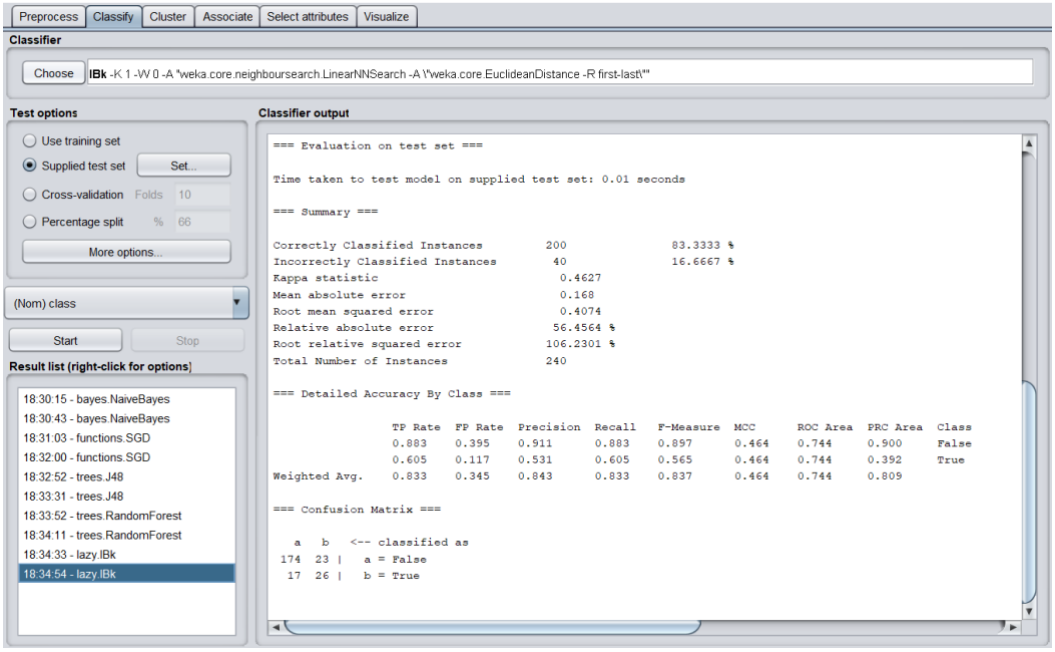


5. IBk (k = 10)

Build the model with training dataset:



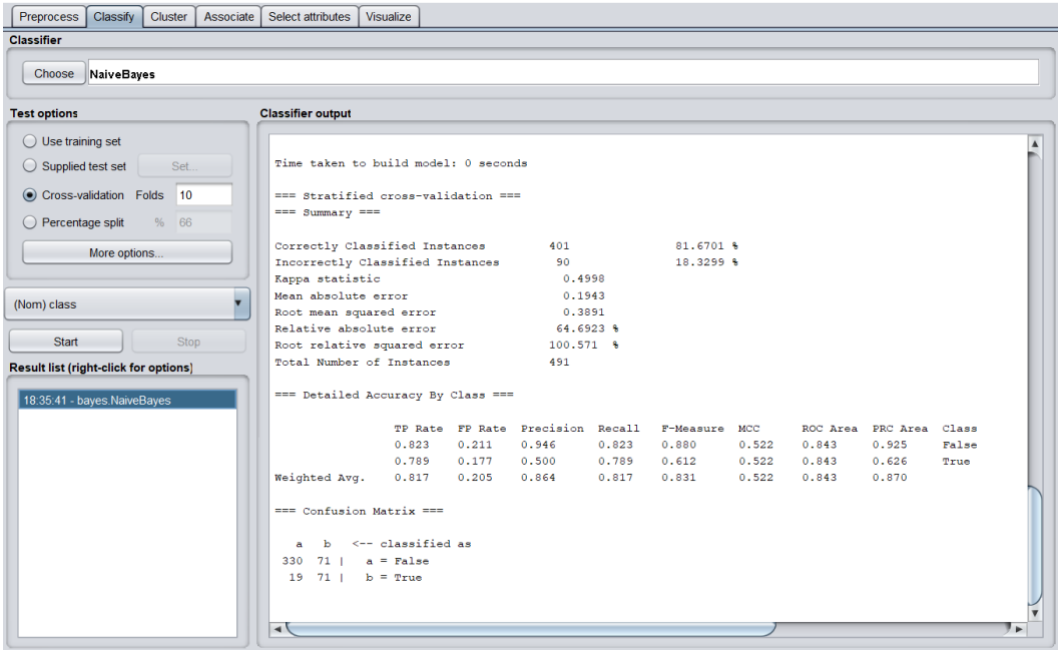
Test it with test dataset:



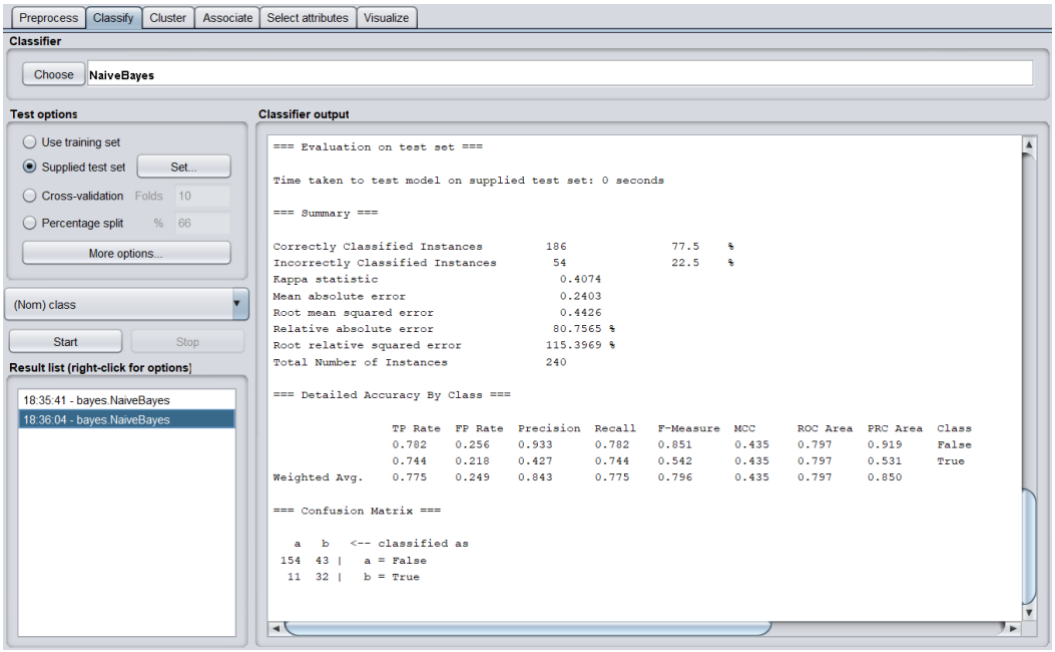
Test Performed on Daily_5 Dataset

1. Naïve Bayes

Build the model with training dataset:

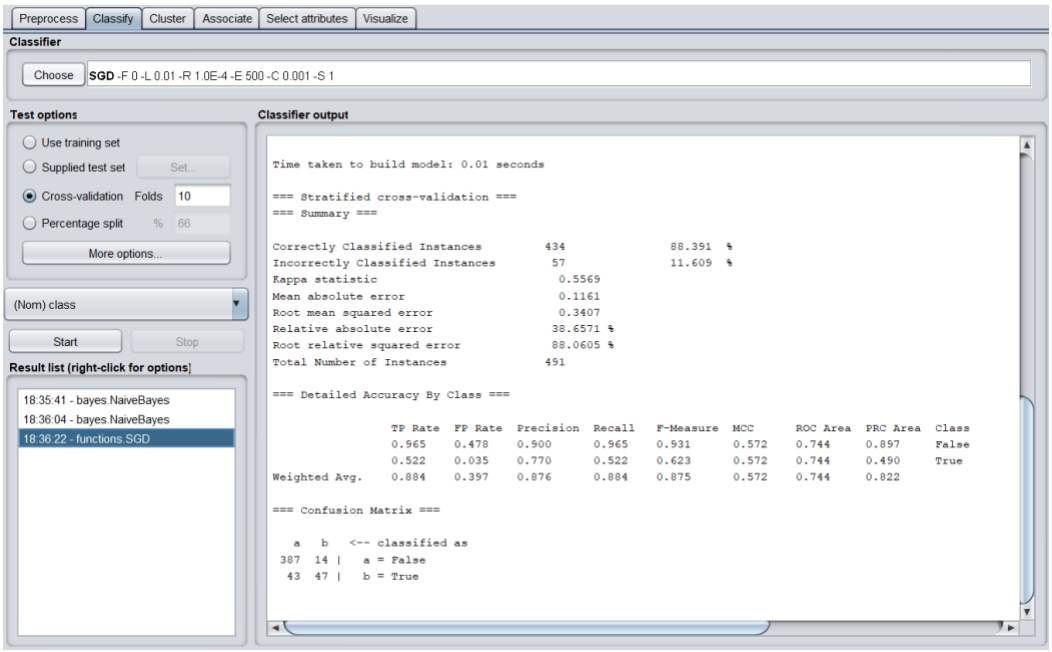


Test it with test dataset:

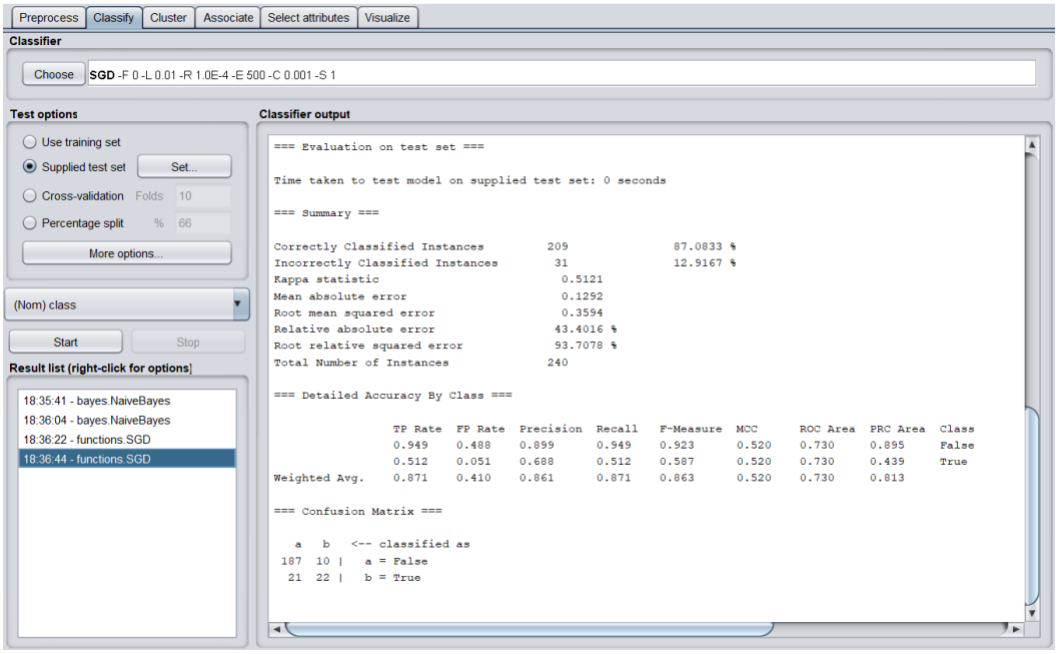


2. SGD

Build the model with training dataset:

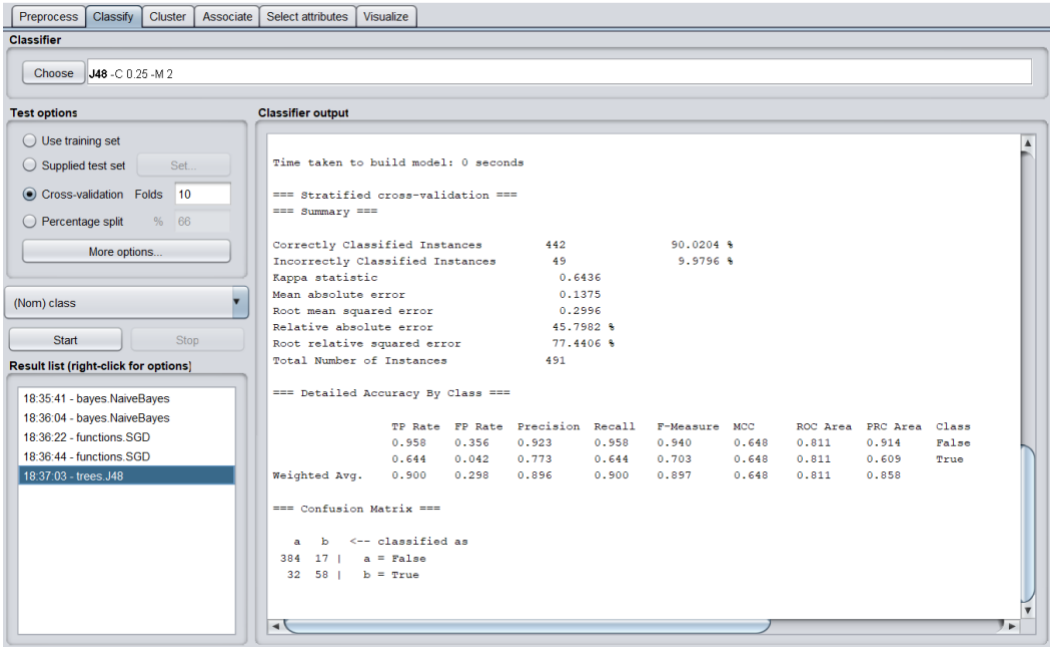


Test it with test dataset:

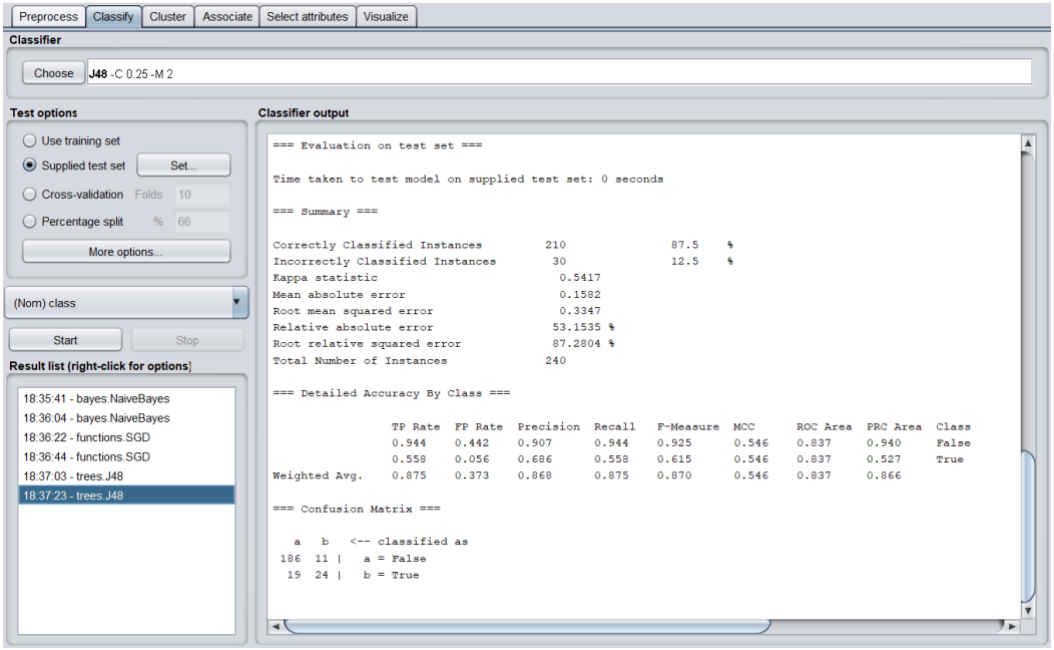


3. J48

Build the model with training dataset:

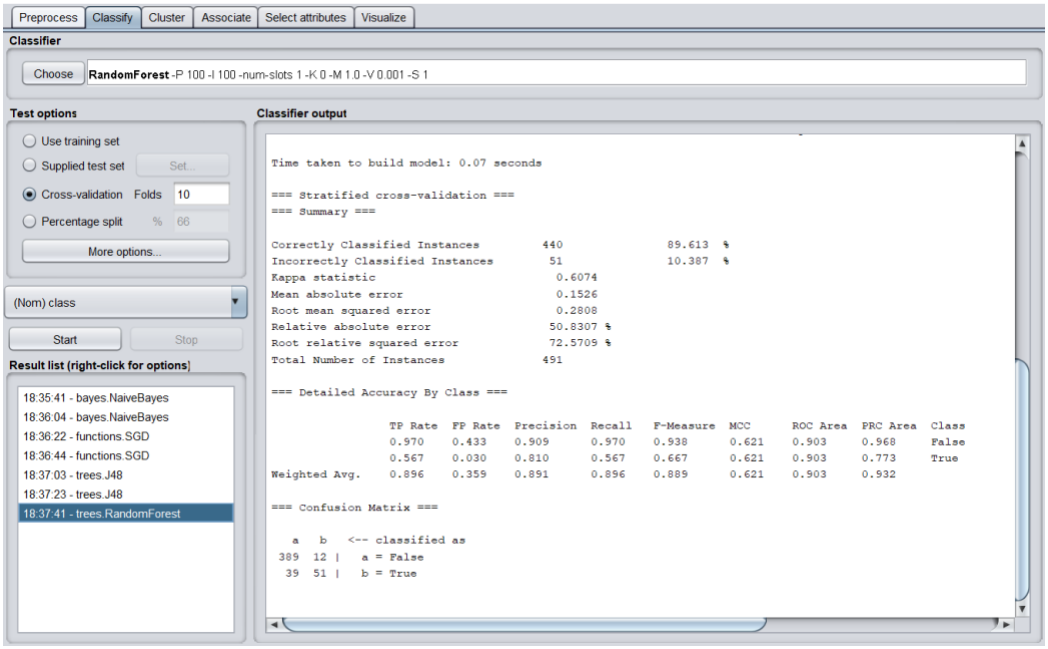


Test it with test dataset:

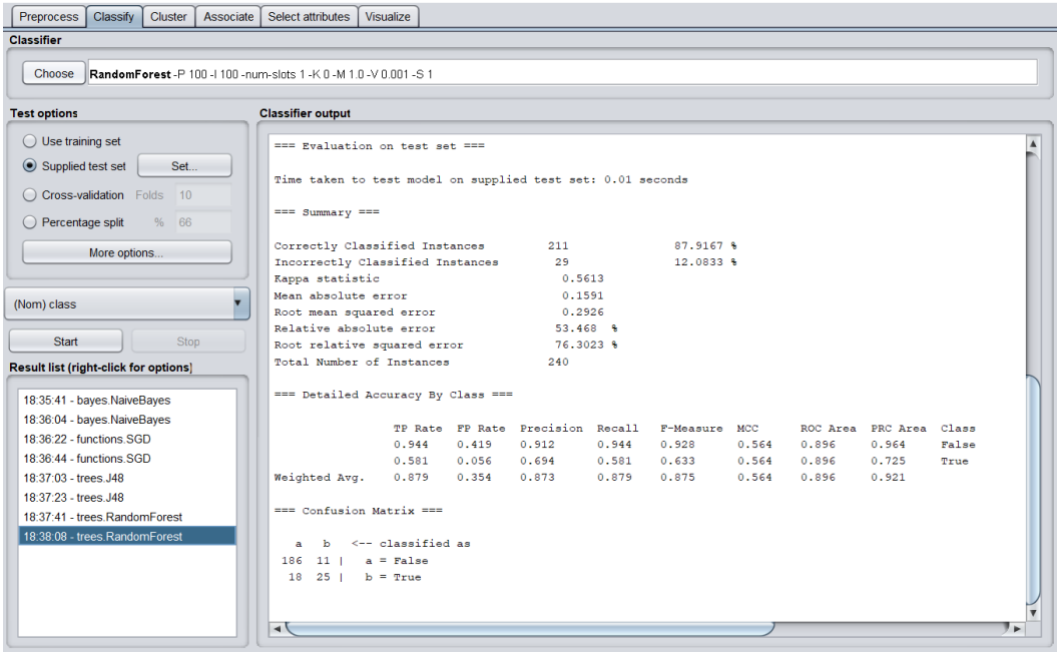


4. Random Forest

Build the model with training dataset:

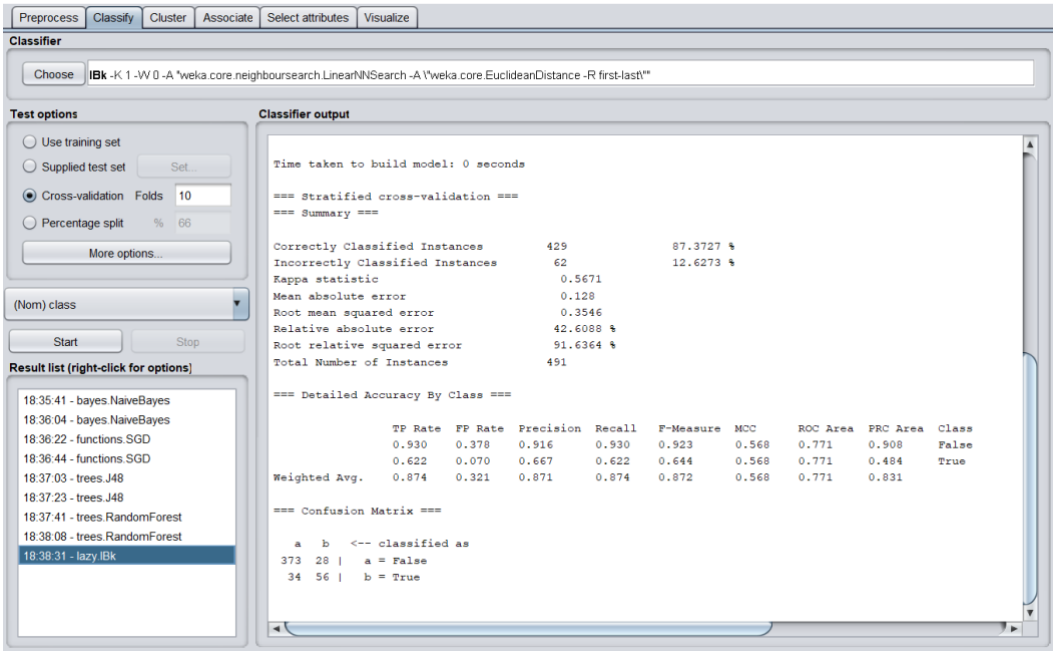


Test it with test dataset:

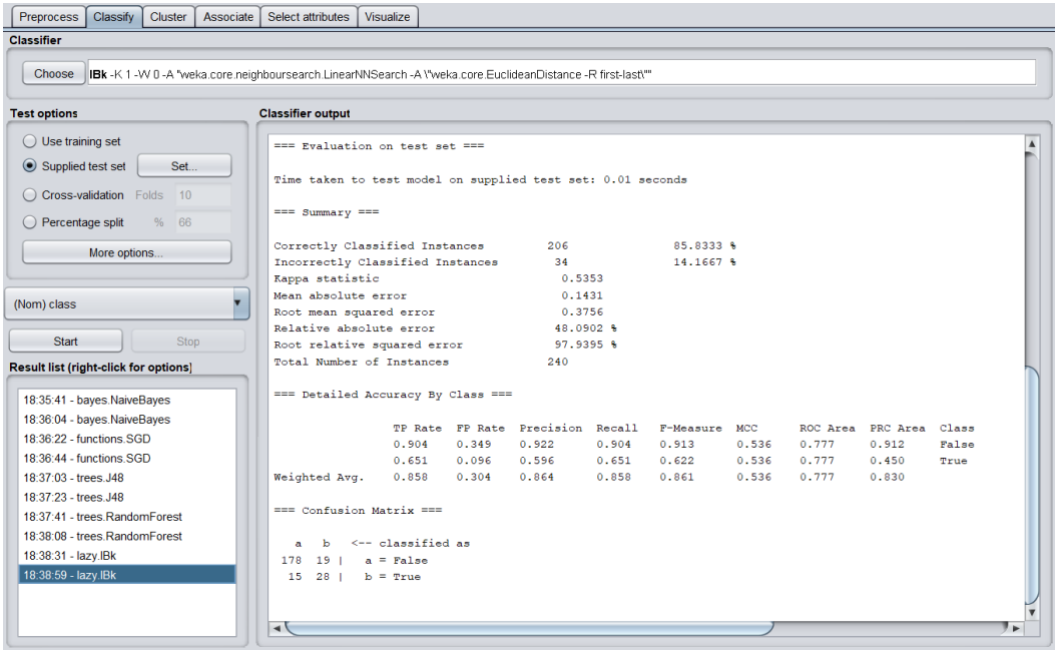


5. IBk (k = 10)

Build the model with training dataset:



Test it with test dataset:



Justification of Best Model

1. Analysis

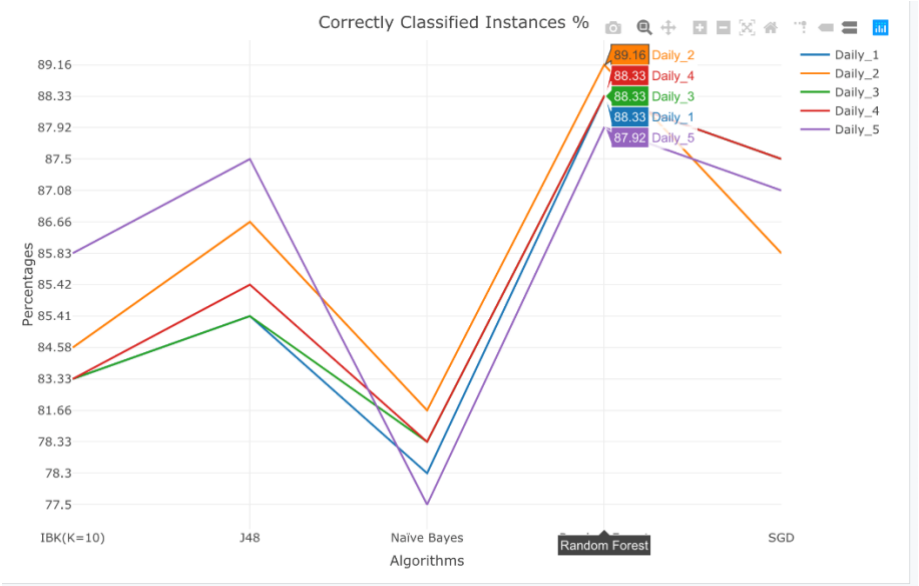
After performing the test on all the datasets created using all the classification model build, we decided to choose Accuracy(%), TP rate, FP Rate, ROC Area and F-Measure as the appropriate metrics for performing comparison. Also since Test statistics help us in deciding how useful a model is in classifying new instances correctly, we will be performing comparison on basis of metrics observed on test datasets.

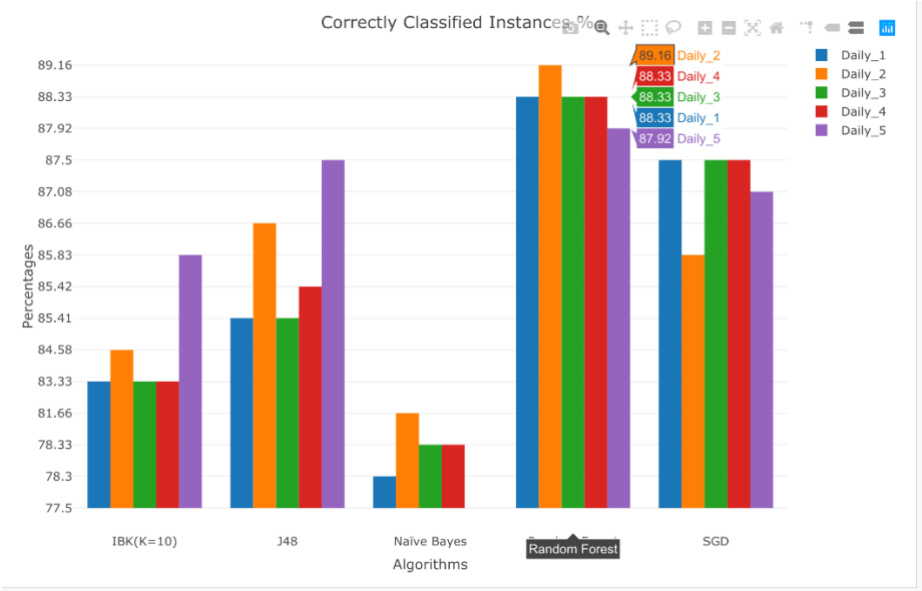
Dataset	*Classifier*	*Accuracy (%)*	*TP Rate*	*FP Rate*	*Roc Area*	*F-Measure*
Daily_1	Naïve Bayes	78.33%	0.767	0.213	0.807	0.559
	SGD	87.50%	0.605	0.066	0.769	0.634
	J48	85.42%	0.535	0.076	0.785	0.568
	Random Forest	88.33%	0.605	0.056	0.909	0.65
	IBK(K=10)	83.33%	0.605	0.117	0.744	0.565
Daily_2	Naïve Bayes	81.67%	0.721	0.162	0.814	0.585
	SGD	85.83%	0.535	0.071	0.732	0.575
	J48	86.67%	0.535	0.061	0.797	0.59
	Random Forest	89.17%	0.651	0.056	0.876	0.683
	IBK(K=10)	84.58%	0.605	0.102	0.752	0.584
Daily_3	Naïve Bayes	78.33%	0.767	0.213	0.807	0.559
	SGD	87.50%	0.605	0.066	0.769	0.634
	J48	85.42%	0.535	0.076	0.785	0.568
	Random Forest	88.33%	0.605	0.056	0.909	0.65
	IBK(K=10)	83.33%	0.605	0.117	0.744	0.565
Daily_4	Naïve Bayes	78.33%	0.767	0.213	0.807	0.559
	SGD	87.50%	0.605	0.066	0.769	0.634
	J48	85.42%	0.535	0.076	0.785	0.568
	Random Forest	88.33%	0.605	0.056	0.909	0.65
	IBK(K=10)	83.33%	0.605	0.117	0.744	0.565
Daily_5	Naïve Bayes	77.50%	0.744	0.218	0.797	0.542
	SGD	87.08%	0.512	0.051	0.73	0.587
	J48	87.50%	0.558	0.056	0.837	0.615
	Random Forest	87.92%	0.581	0.056	0.896	0.633
	IBK(K=10)	85.83%	0.651	0.096	0.777	0.622

Metrics Observed on Test Dataset(For TRUE Class)

2. Results

1) Correctly Classified Instances (Accuracy (%))

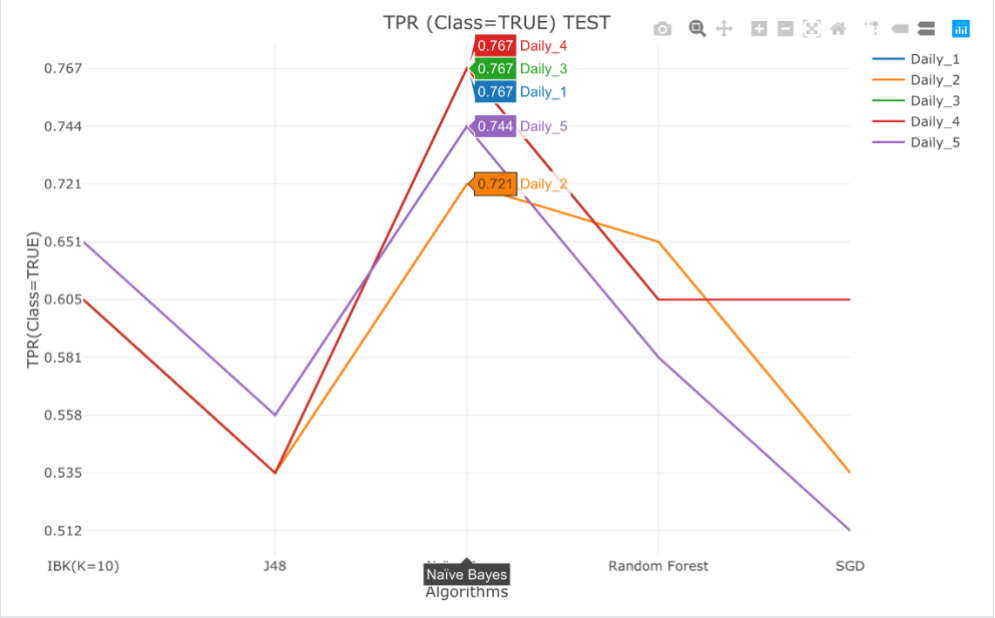


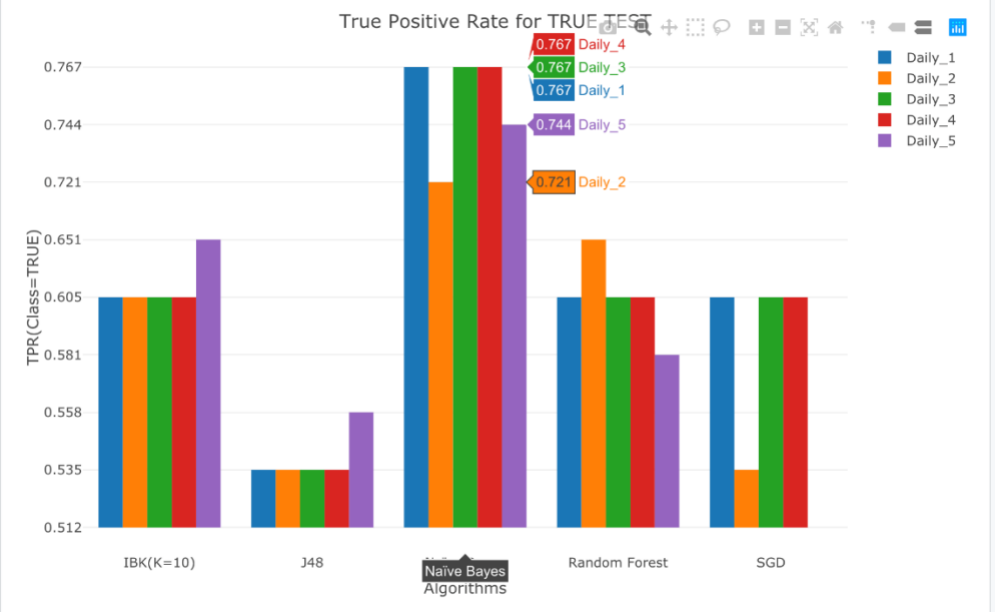


As we can see clearly that the test model of **Random Forest** tested on dataset **Daily_2** has the highest accuracy, which is 89.1667%.

2) TP Rate for TRUE

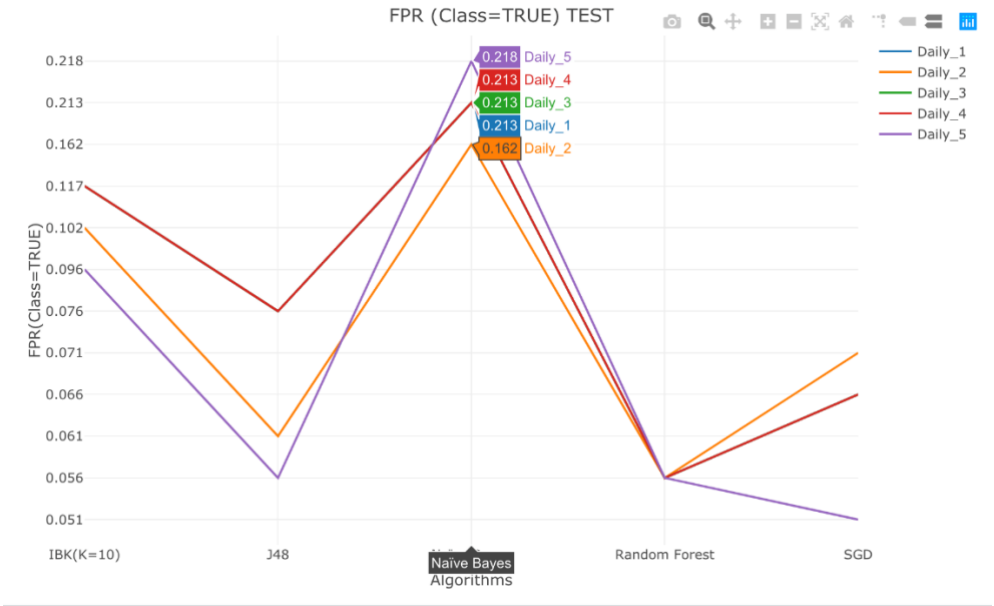
For our model classifying the occurrence is crucial, that's why we will be comparing TP Rate for TRUE class values only.

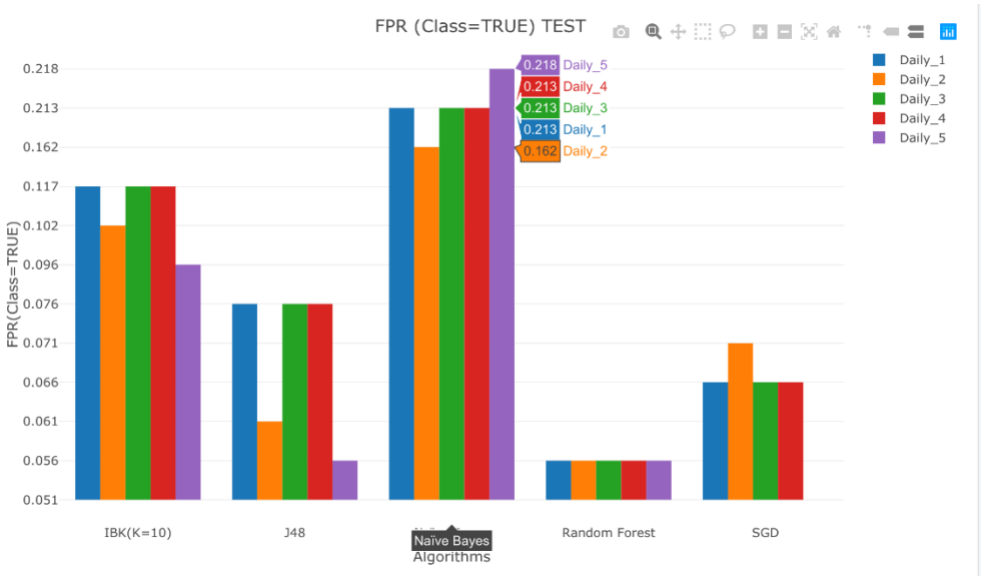




As we can see clearly that the test model of **Naïve Bayes** tested on datatset **Daily_1, Daily_3, and Daily_4** have the highest TP Rate for Class TRUE, which is 0.767.

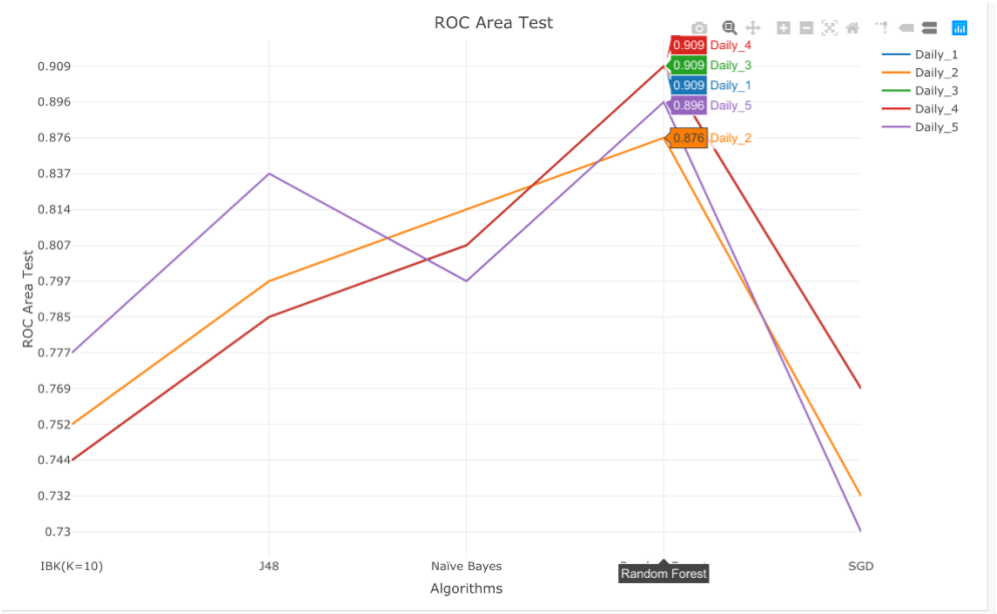
3) FP Rate for TRUE

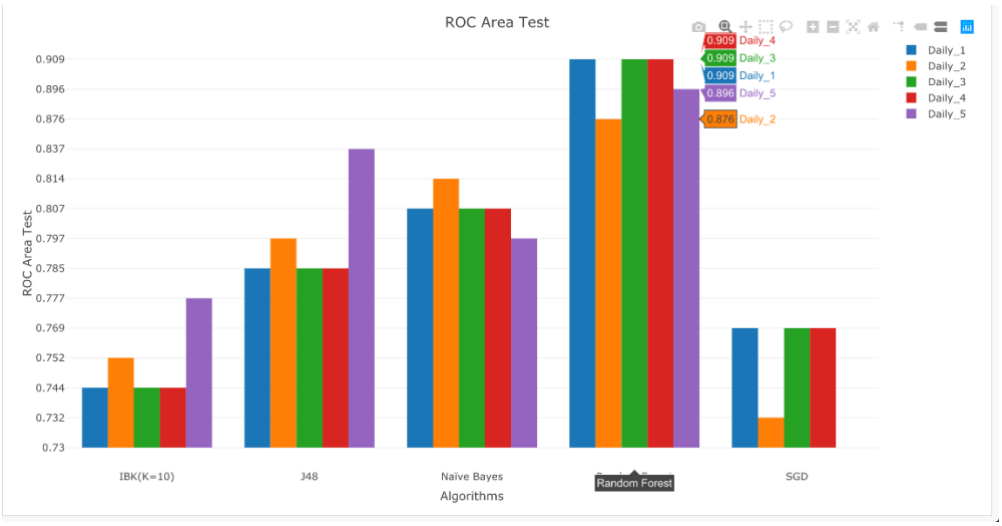




As we can see clearly that the test model of **Naïve Bayes** tested on **Daily_5** has the highest FP Rate for Class TRUE, which is 0.218.

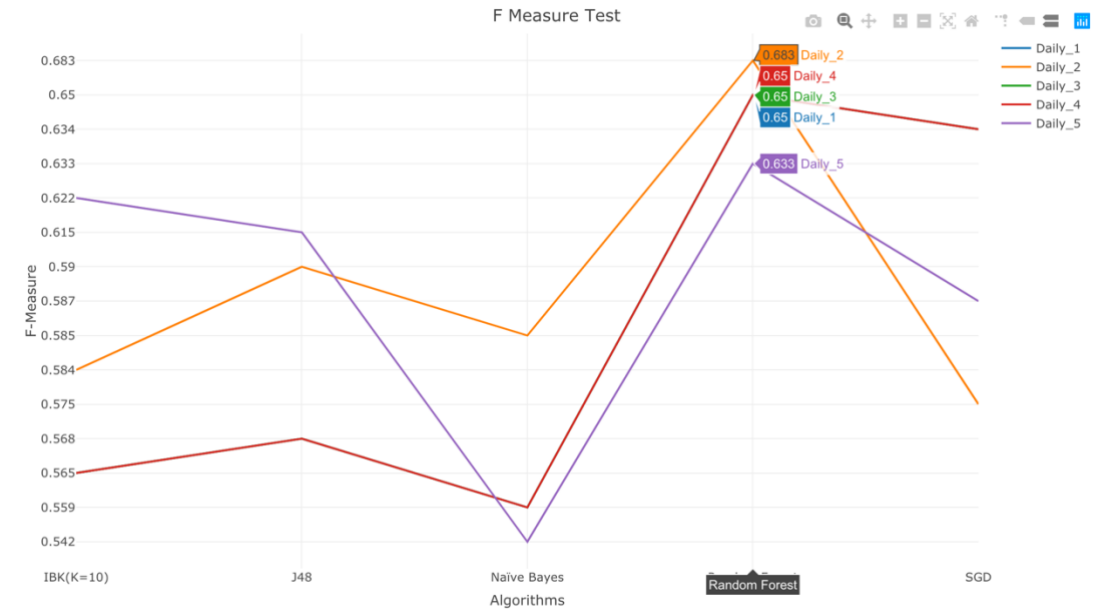
4) Roc Area

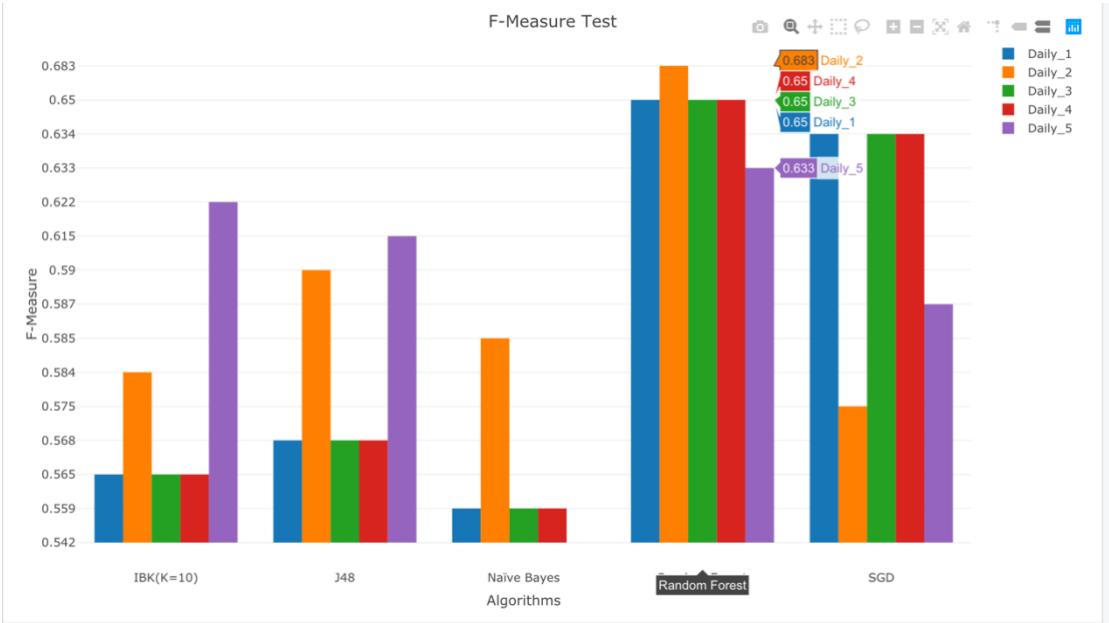




As we can see clearly that the test model of **Random Forest** tested on **Daily_1, Daily_3, and Daily_4** have the largest Roc Area, which is 0.909.

5) F-Measure





As we can see clearly that the test model of **Random Forest** tested on **Daily_2** have the largest F-Measure which is 0.683.

3. Conclusion

After comparing all the metrics together, we think the **best classifier model** would be **Random Forest** built on **dataset Daily_2**. It has the highest accuracy 89.1667%, with TP Rate of 0.651 for class TRUE, TP Rate of 0.944 for class FALSE, FP Rate of 0.056 for class TRUE, FP Rate of 0.349 for class FALSE, 0.876 Roc Area and highest F-Measure 0.683.

SUMMARY

For this data mining project, we start with a raw dataset about weather information of Brentwood at San Francisco Bay area from January 1, 2018 to January 1, 2020. Our goal is to predict the occurrence of rain in this area.

We struggled a bit at which attributes should be eliminated manually in the dataset as a part of preprocessing, based on our testing, different columns reductions have different impact on our results. After comparing different reduction methods, we choose to only keep the attributes which are correlated with our data mining goal and remove the attributes having more than 95% of the values missing. We learned about our dataset and accordingly took the decision.

We also learned how to use the attribute selection algorithms and classification algorithms in Weka, as well as splitting dataset. And we understand the difference between "Use training…", "Supplied test…", and "Cross-validation…". "Use training…" is not quite useful, as we observed significant difference in accuracy(%) , moreover it was unable to evaluate the metrics for TRUE class values. Validation dataset is predominately used to describe the evaluation of models when tuning hyperparameter and data preparation, and the Test dataset is predominately used to describe the evaluation of a final tuned model when comparing it to other final models. This is the reason why we choose the test dataset to process our model comparison.

For this project, both of group members had contributed a considerable amount of time. We generally split our project into two part, Jyoti takes care of the first three attribute select algorithms as well as its' classification algorithms, the dataset splitting explanation, and the model justification graphs. Siyu takes care of the last two attribute select algorithms as wells its' classification algorithms, the dataset preprocess, and wrapped all of them up and finish the model justification and conclusion. We have several versions of our report, and both of group members modified it and made it better.

Bibliography

CIMIS (2020). CIMIS Station Reports. Retrieved from https://cimis.water.ca.gov.

Wikipedia (February 29, 2020). C4.5 algorithm. Retrieved from https://en.wikipedia.org/wiki/C4.5_algorithm.

Wikipedia (March 8, 2020). Naïve Bayes classifier. Retrieved from https://en.wikipedia.org/wiki/Naive_Bayes_classifier.

Wikipedia (March 12, 2020). Random forest. Retrieved from https://en.wikipedia.org/wiki/Random_forest.

Wikipedia (Octover 26, 2019). Multilayer perceptron. Retrieved from https://en.wikipedia.org/wiki/Multilayer_perceptron.