Project 1

Data mining with Rapid Miner

Team Members
Karthik Karunanithi (802827527)
Bharath Krishnan (893429449)
Shankar Tiwari (803012350)
Gargi Mrunal Kulkarni (893210922)

CPSC 483
Data Mining and Pattern Recognition
Fall, 2016

Prof: Kenytt Avery
Department of Computer Science
California State University, Fullerton
October 27, 2016

TABLE OF CONTENTS

1 INTRODUCTION

The project aims to perform different machine learning algorithms on the data set, using data mining tool and evaluate the results. The project implements the following:

Dataset: Wine Quality Dataset from the UCI Machine Learning Repository.

Data Mining Tool: Rapid Miner.

Analysis & Prediction:

- Wine quality score
- Red or white wine
- Quality as a class

Algorithms: The following algorithms were implemented in the project:

- k-Nearest Neighbor regression with different choices of k
- k-Nearest Neighbor classification with different choices of k
- Naive Bayesian classification
- Linear Regression
- Logistic Regression

2 ABOUT THE DATASET AND ALGORITHMS

2.1 DATASET

The dataset used is Wine Quality Dataset from the UCI Machine Learning Repository. This dataset is public available for research. The details are described in [Cortez et al., 2009]. There are two datasets of red and white wine samples. These datasets can be viewed as classification or regression tasks.

Below is the information of dataset:

1. Number of Observations: red wine - 1599; white wine - 4898.

2. Number of Attributes: 11 + output attribute
   Note: several of the attributes may be correlated, thus it makes sense to apply some sort of feature selection.

3. Attribute information:
   - Input variables (based on physicochemical tests):
     a. fixed acidity
     b. volatile acidity
     c. citric acid
     d. residual sugar
     e. chlorides
     f. free sulfur dioxide

g. total sulfur dioxide
h. density
i. pH
j. sulphates
k. alcohol

- Output variable (based on sensory data): quality (score between 0 and 10)

4. Missing Attribute Values: None

2.2 ALGORITHMS

The algorithms on Wine Quality Dataset are as given below :

1. *K-Nearest Neighbor*: In pattern recognition, the k-Nearest Neighbors algorithm (or k-NN for short) is a non-parametric method used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression: In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). In k-NN regression, the output is the property value for the object. This value is the average of the values of its k nearest neighbors.
2. *Naïve Bayesian Classification*: In machine learning, naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features. Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set.
3. *Linear Regression*: In statistics, linear regression is an approach for modeling the relationship between a scalar dependent variable y and one or more explanatory variables (or independent variables) denoted X. The case of one explanatory variable is called simple linear regression. For more than one explanatory variable, the process is called multiple linear regression. In linear regression, the relationships are modeled using linear predictor functions whose unknown model parameters are estimated from the data. Such models are called linear models.
4. *Logistic Regression*: In statistics, logistic regression, or logit regression, or logit model[1] is a regression model where the dependent variable (DV) is categorical. Logistic regression measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic function, which is the cumulative logistic distribution. Logistic regression can be seen as a special case of the generalized linear model and thus analogous to linear regression. The model of logistic regression, however, is based on quite different assumptions (about the relationship between dependent and independent variables) from those of linear regression.

3 ABOUT THE TOOL – RAPID MINER

RapidMiner is a software platform developed by the company of the same name that provides an integrated environment for machine learning, data mining, text mining, predictive analytics and business analytics. It is used for business and commercial applications as well as for research, education, training, rapid prototyping, and application development and supports all steps of the data mining process including data preparation, results visualization, validation and optimization. RapidMiner's "all-in-one" platform, accelerates the building of complete analytical workflows – from data prep to modeling to business deployment – in a single environment, dramatically improving efficiency and shorting the time to value for data scientists and business analysts alike. The features of Rapid Miner are as follows:

1. *Unified Platform*. One platform, one user interface, one system, support the complete analytic workflow from data prep, through model deployment to ongoing model management
2. *Visual Programming Environment*. Quick-to-learn and easy-to-use drag & drop approach accelerates end-to-end predictive analytics for improved productivity
3. *Breadth of Functionality*. More pre-defined, advanced and robustly engineered machine learning functions than any other visual platform
4. *Open Source Innovation*. Well-accepted open languages and technology, a community of 250K data science experts and a robust marketplace keeps pace with trends and extensibility needs
5. *Broad Connectivity*. More than 60 connectors (including Hadoop) provides easy access to all types of data: structured, unstructured & big data
6. *Advanced Analytics at Every Scale*. Two seamlessly integrated compute engines: In-Memory and In-Hadoop, provides the best analytics option for every size database

3.1 SYSTEM REQUIREMENTS AND INSTALLATION STEPS

The system requirement for Rapid Miner Studio have no any minimal specification. Since data analysis is a computationally intensive task—the better your hardware, the better your experience. Also, the memory should be enough to handle big data sets. Rapid Miner Studio is JAVA based, so it requires proper JRE to run. More information on system requirement is available on http://docs.rapidminer.com/studio/installation/system-requirements.html

Download the Rapid Miner Studio from Rapid Miner Official site and install the downloaded .exe file. The complete guide is available at http://docs.rapidminer.com/studio/installation/

3.2 RAPID MINER STUDIO OVERVIEW

RapidMiner Studio is a code-free environment for designing advanced analytic processes with machine learning, data mining, text mining, predictive analytics and business analytics. It is a powerful visual programming environment for rapidly building complete predictive analytic workflows. This all-in-one tool features hundreds of pre-defined data preparation and machine learning algorithms to efficiently support all data science needs. The studio GUI looks like below figure.

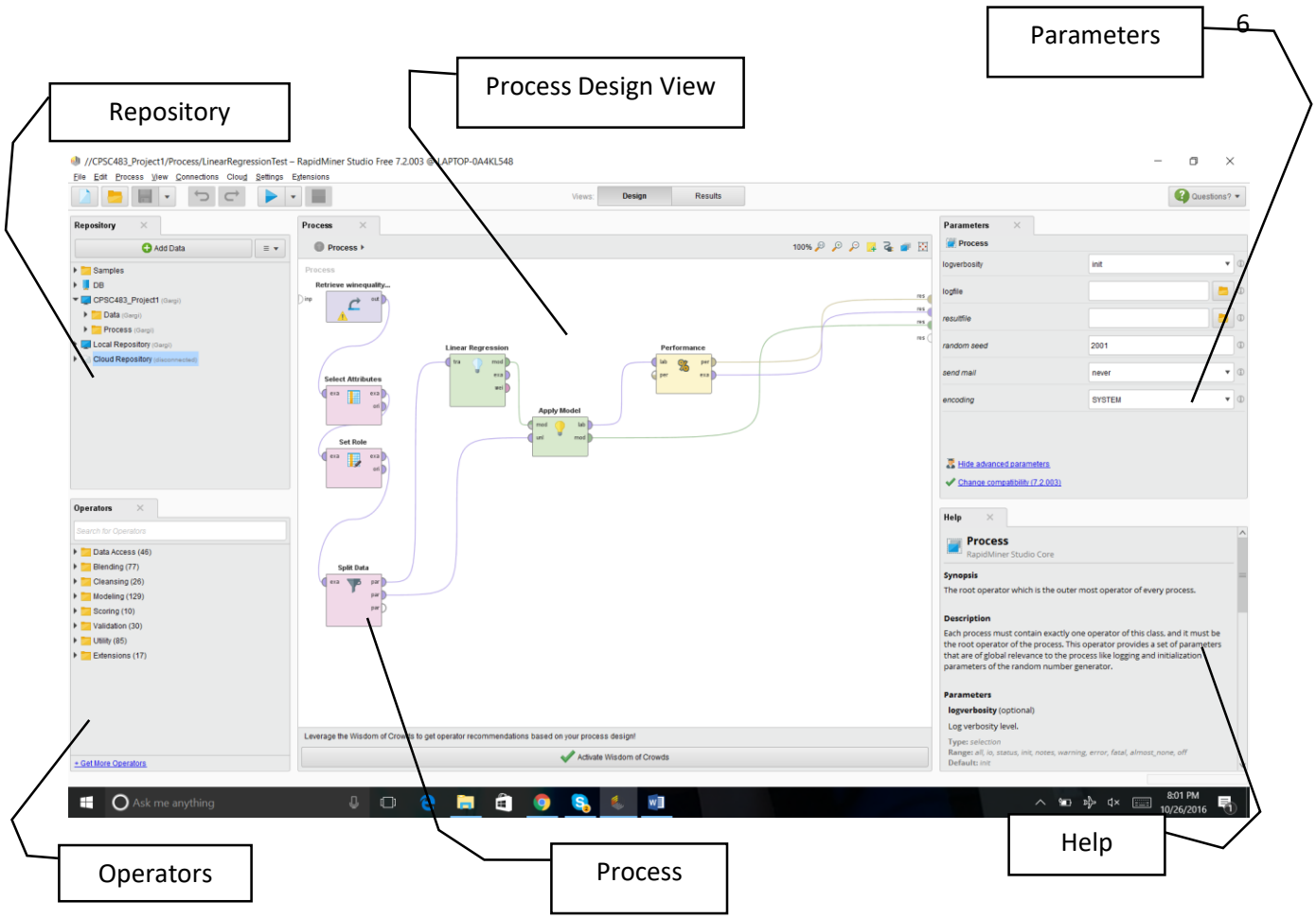


Figure 1 : Rapid Miner Studio UI

The following is the description of the panels of Rapid Miner UI:

- *Repository*: The storage mechanism for data and RapidMiner processes. Best practice recommends you use the repositoryfor data storage instead of reading directly from a file or database. If you use a Read operator, meta data will not be available to RapidMiner, limiting the available functions. By default, RapidMiner Studio comes configured with a variety of sample data sets and process in the Samples directory of your repository. From the Repository panel you can also access the Cloud Repository.
- *Operators*: The building blocks, grouped by function, used to create RapidMiner processes. An *operator* has input and output ports; the action performed on the input ultimately leads to what is supplied to the output. Operator parameters control those actions. There are more than 1500 operators available in RapidMiner. Operators, in the **Operators** panel of the **Design** view, are both browsable and searchable.
- *Process*: A set of interconnected operators represented by a flow design, where each operator manipulates your data. A process might, for example, load a data set, transform the data, compute a model, and apply the model to another data set.
- *Process View:* The working area for building processes. This is the canvas in the **Design** view where you drag operators or where, when you double-click a process, the operators of that process appear. The shown UI is of design view where the process is designed. The results of the designed process can be viewed in **Result** view.

- *Parameter:* The setting(s) whose value(s) determine the characteristics or behavior of an operator. RapidMiner presents parameters in the Parameters panel of the Design view. There are regular parameters and expert parameters. The expert parameters are indicated by italic names and are displayed or hidden by clicking the Show/Hide advanced parameters link at the bottom of the panel.
- *Help*: It gives the description about the operator selected in process. The complete documentation can be found in help panel for an operator selected.

4 PROJECT IMPLEMENTATION

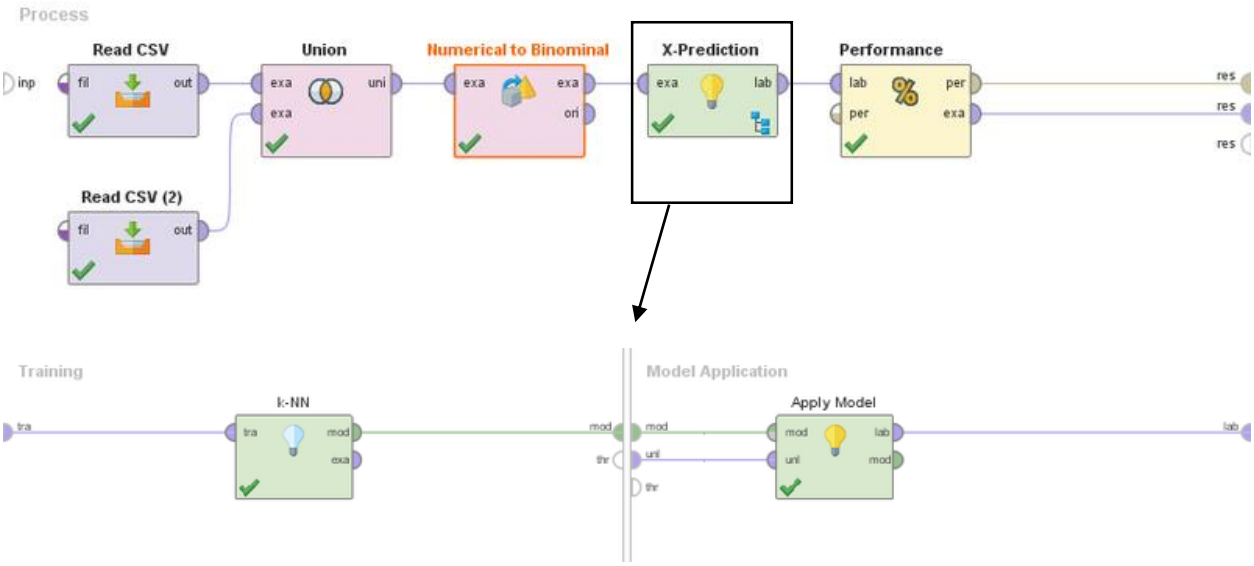
In this project the algorithms KNN classification and regression, Naïve Bayes classification, Linear regression and Logistic regression are implemented in Rapid Miner. The dataset used is wine dataset. The aim is to predict wine quality score, to classify red and white wine and to classify wine depending on score.

4.1 KNN CLASSIFICATION

The k-NN classification is implemented to classify wine depending on quality score and wine type.

4.1.1 Classifying the wine depending on quality score

4.1.1.1 Process Design

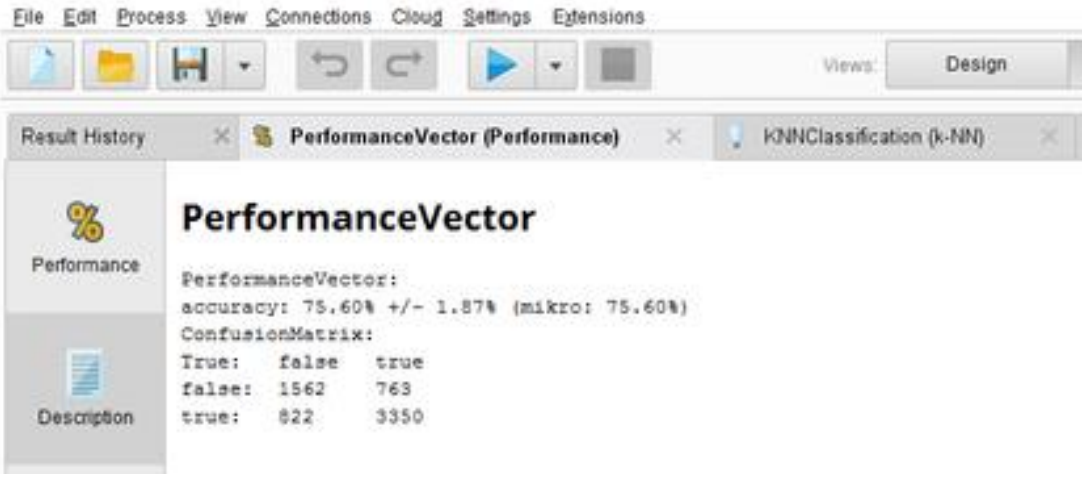


Following operators were used to implement the process:

- **Read CSV** operator is used to read the dataset.
- **Union operator** is used to combine two datasets.
- **Numerical to Binomial operator** is used to convert the quality attribute into binary type. If quality is less than 5 then it is set to false otherwise true.

- **X-Prediction operator** performs the cross validation on data set. It is a nested operator. It has two subprocesses: a training subprocess and a testing subprocess. The training subprocess is used for training a model. The trained model is then applied in the testing subprocess. The testing subprocess returns a labeled ExampleSet.
- **k-NN operator** implements the k-NN Algorithm and is trained by X-predictor.
- **Apply Model operator** applies the trained k-NN model on test set from X-predictor.
- **Performance operator** is of classification type. It evaluates the model and output the performance vector containing accuracy, confusion matrix etc.

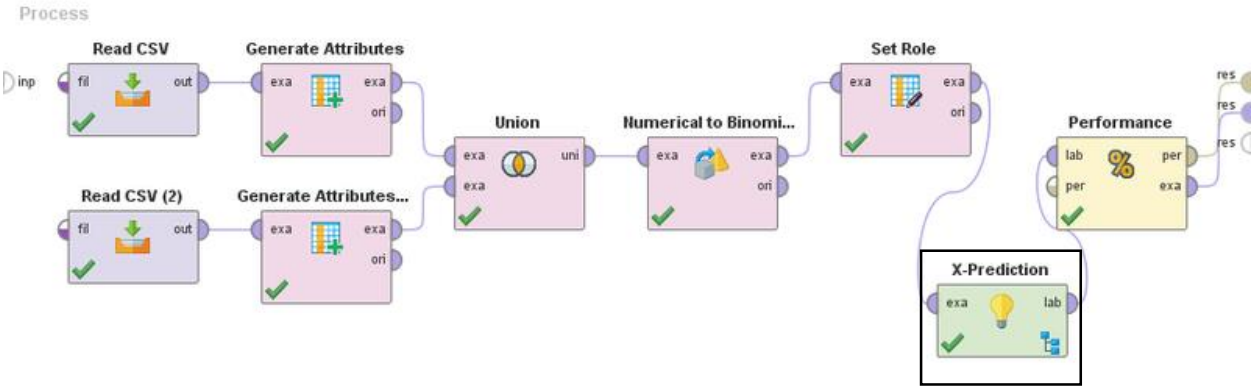
4.1.1.2 Results

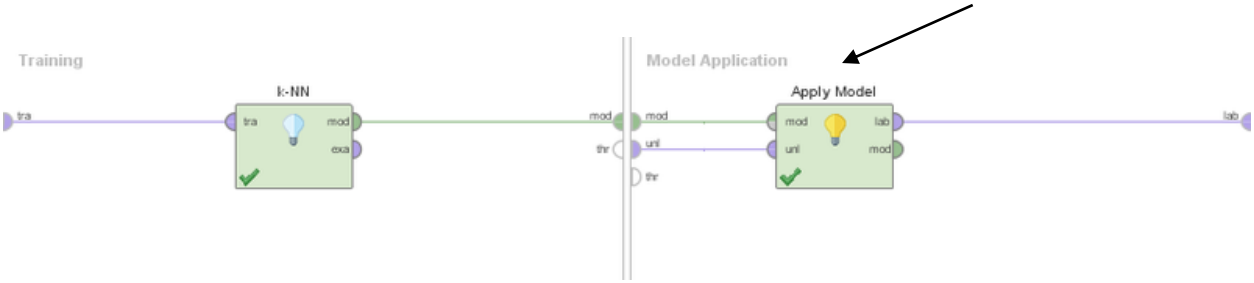


The performance vector shows the accuracy of the classifier and the confusion matrix. The classification is performed for different values of *k* and k=1 gave optimal results.

4.1.2 Classifying the wine depending on type- Red or White wine

4.1.2.1 Process Design

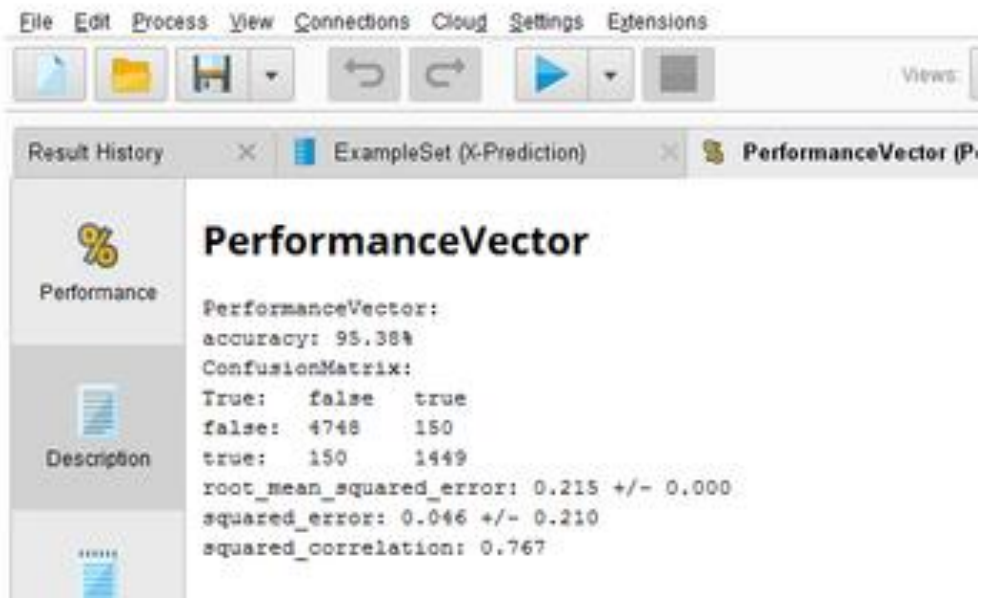




Following operators were used to implement the process:

- **Read CSV** operator is used to read the dataset.
- **Generate Attributes** is used to generate new attribute *wine* for wine type-red or white wine.
- **Union operator** is used to combine two datasets.
- **Numerical to Binomial operator** is used to convert the wine attribute into binary type. If wine type is red the wine is set false otherwise true.
- **Set Role operator** is used to set the role for wine attribute as label.
- **X-Prediction operator** performs the cross validation on data set. It is a nested operator. It has two subprocesses: a training subprocess and a testing subprocess. The training subprocess is used for training a model. The trained model is then applied in the testing subprocess. The testing subprocess returns a labeled ExampleSet.
- **k-NN operator** implements the k-NN Algorithm and is trained by X-predictor.
- **Apply Model operator** applies the trained k-NN model on test set from X-predictor.
- **Performance operator** is of classification type. It evaluates the model and output the performance vector containing accuracy, confusion matrix etc.

4.1.2.2 Results



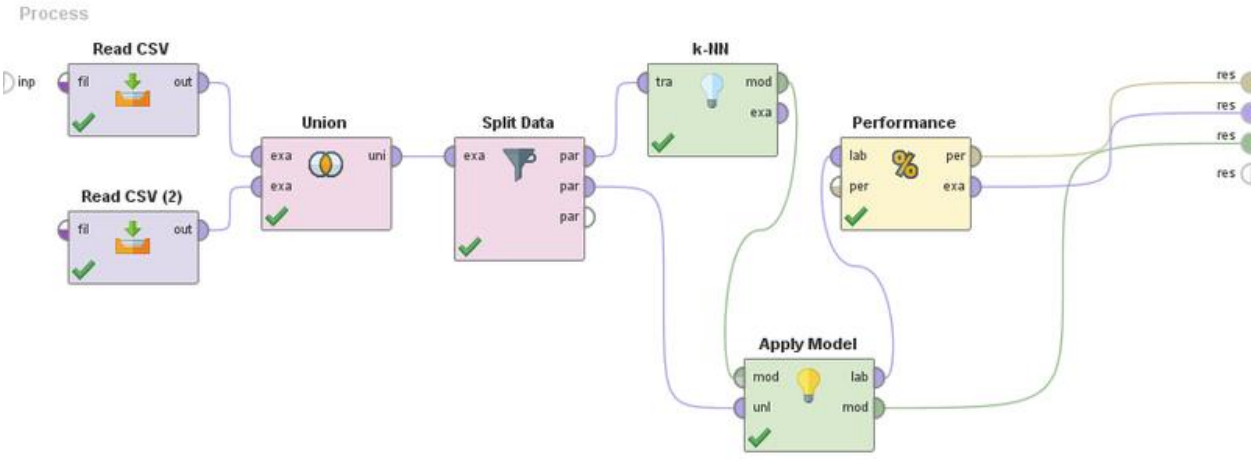
The performance vector shows the accuracy of the classifier, the confusion matrix etc. The classification is performed for different values of *k* and k=1 gave optimal results.

4.2 KNN REGRESSION

The k-NN Regression is performed on red and wine data sets combined to predict the wine quality score.

4.2.1 KNN regression with Set Validation

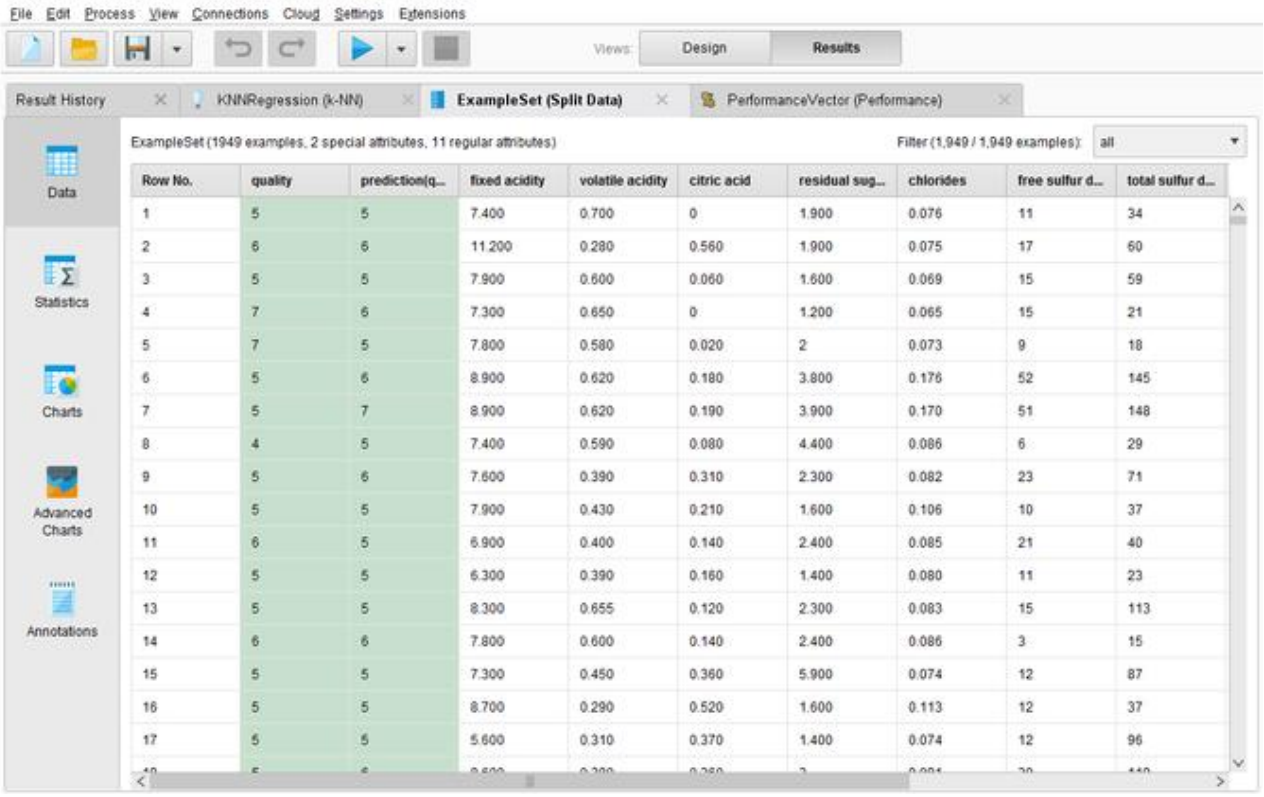
4.2.1.1 Process Design



Following operators were used to implement the process:

- **Read CSV** operator is used to read the dataset.
- **Union operator** is used to combine two datasets.
- **Split Data operator** is used to perform validation by splitting data into 70% training set and 30% test set.
- **k-NN operator** implements the k-NN Algorithm and is trained using training partition of spilt data output.
- **Apply Model operator** applies the trained k-NN model on test partition of the split data output.
- **Performance operator** is of regression type. It evaluates the model and output the performance vector containing root squared error, prediction error, absolute error etc.

4.2.1.2 Results



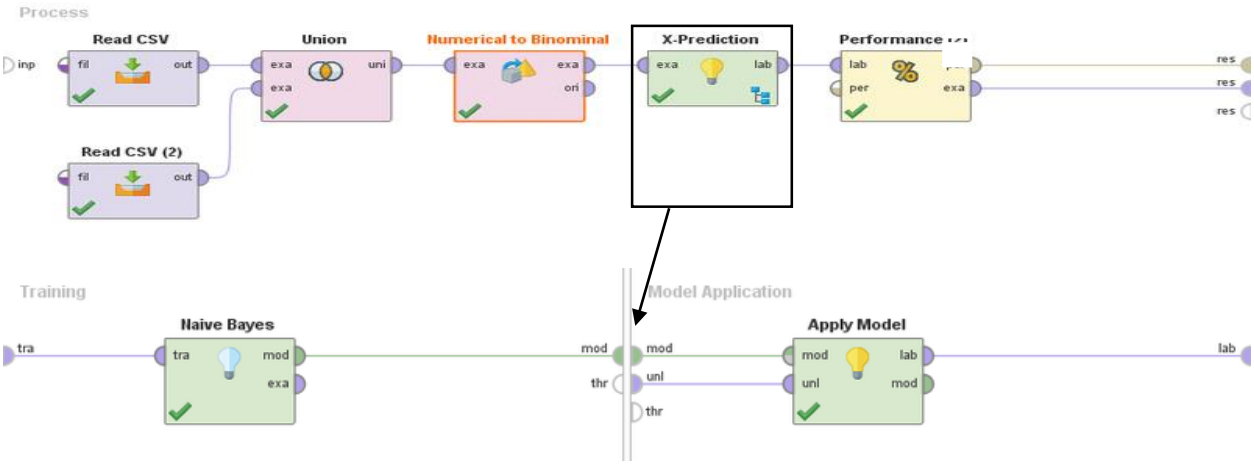
The results show the performance vector with root mean squared error, squared error and squared correlation. The example set shows the true and predicted values for wine quality. The k-NN regression was performed for different values of *k* and k~17 it gave optimal performance.

4.3 NAÏVE BAYES CLASSIFICATION

The Naïve Bayes classification is implemented to classify wine depending on quality score and wine type.

4.3.1 Classifying the wine depending on quality score

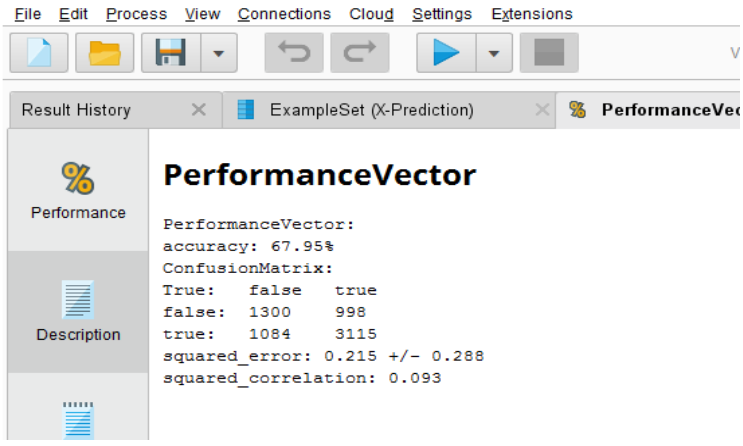
4.3.1.1 Process Design

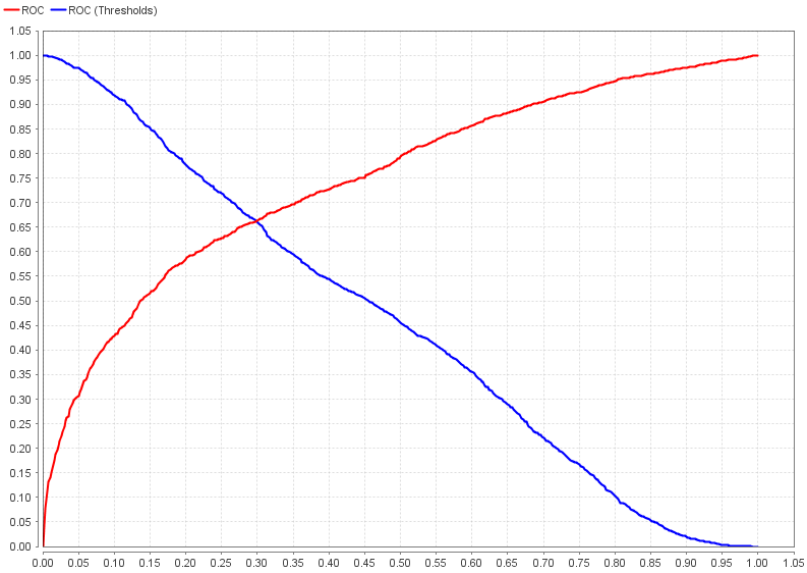


Following operators were used to implement the process:

- **Read CSV** operator is used to read the dataset.
- **Union operator** is used to combine two datasets.
- **Numerical to Binomial operator** is used to convert the quality attribute into binary type. If quality is less than 5 then it is set to false otherwise true.
- **X-Prediction operator** performs the cross validation on data set. It is a nested operator. It has two subprocesses: a training subprocess and a testing subprocess. The training subprocess is used for training a model. The trained model is then applied in the testing subprocess. The testing subprocess returns a labeled ExampleSet.
- **Naïve Bayes operator** implements the Naïve Bayes Algorithm and is trained by X-predictor.
- **Apply Model operator** applies the trained Naïve Bayes model on test set from X-Predictor.
- **Performance operator** is of classification type. It evaluates the model and output the performance vector containing accuracy, confusion matrix etc.

4.3.1.2 Results





Result History | ExampleSet (X-Prediction) | PerformanceVector (Performance (2))

Data | Statistics | Charts | Advanced Charts | Annotations

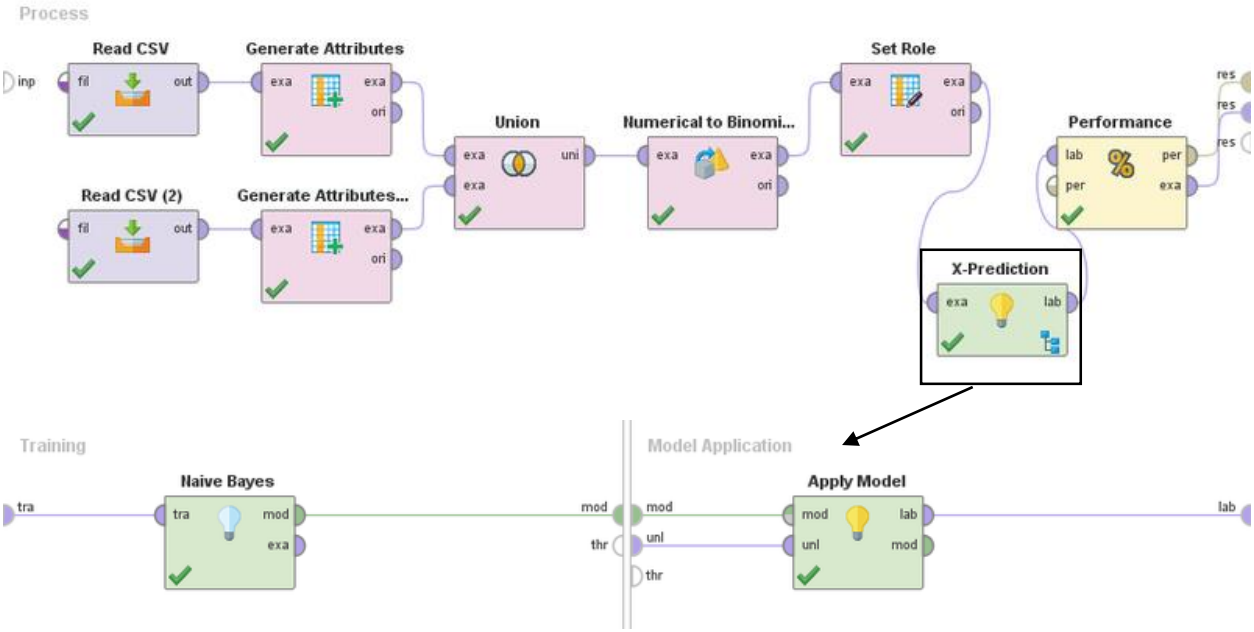
ExampleSet (6497 examples, 4 special attributes, 11 regular attributes) Filter (6,497 / 6,497 examples): all

Row No.	quality	prediction(q...	confidence(f...	confidence(t...	fixed acidity	volatile acidity	citric acid	residual sug...	chlorides
1	true	false	0.560	0.440	7	0.270	0.360	20.700	0.045
2	true	true	0.278	0.722	6.300	0.300	0.340	1.600	0.049
3	true	true	0.238	0.762	8.100	0.280	0.400	6.900	0.050
4	true	true	0.362	0.638	7.200	0.230	0.320	8.500	0.058
5	true	true	0.362	0.638	7.200	0.230	0.320	8.500	0.058
6	true	true	0.235	0.765	8.100	0.280	0.400	6.900	0.050
7	true	true	0.384	0.616	6.200	0.320	0.160	7	0.045
8	true	false	0.544	0.456	7	0.270	0.360	20.700	0.045
9	true	true	0.260	0.740	6.300	0.300	0.340	1.600	0.049
10	true	true	0.069	0.931	8.100	0.220	0.430	1.500	0.044
11	false	true	0.006	0.994	8.100	0.270	0.410	1.450	0.033
12	false	true	0.289	0.711	8.600	0.230	0.400	4.200	0.035
13	false	true	0.054	0.946	7.900	0.180	0.370	1.200	0.040
14	true	true	0.001	0.999	6.600	0.160	0.400	1.500	0.044
15	false	false	0.794	0.206	8.300	0.420	0.620	19.250	0.040
16	true	true	0.013	0.987	6.600	0.170	0.380	1.500	0.032
17	true	false	0.532	0.468	6.300	0.480	0.040	1.100	0.046

The performance vector shows the accuracy of the classifier, the confusion matrix etc. The ROC curve tells about the classifier performance. The Example set table shows the quality score true and predicted values.

4.3.2 Classifying the wine depending on type- Red or White wine

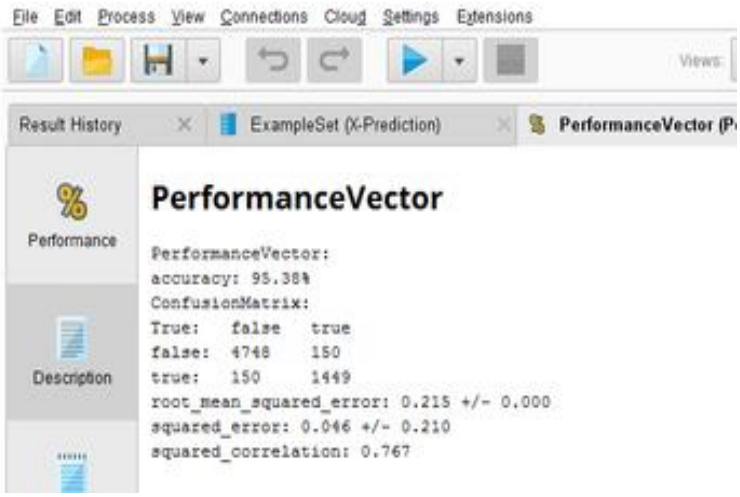
4.3.2.1 Process Design

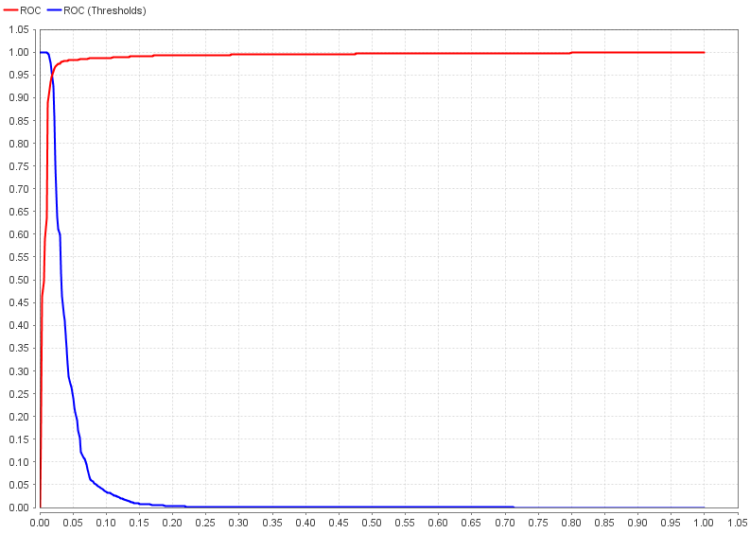


Following operators were used to implement the process:

- **Read CSV** operator is used to read the dataset.
- **Generate Attributes** is used to generate new attribute *wine* for wine type-red or white wine.
- **Union operator** is used to combine two datasets.
- **Numerical to Binomial operator** is used to convert the wine attribute into binary type. If wine type is red the wine is set false otherwise true.
- **Set Role operator** is used to set the role for wine attribute as label.
- **X-Prediction operator** performs the cross validation on data set. It is a nested operator. It has two subprocesses: a training subprocess and a testing subprocess. The training subprocess is used for training a model. The trained model is then applied in the testing subprocess. The testing subprocess returns a labeled ExampleSet.
- **Naïve Bayes operator** implements the Naïve Bayes Algorithm and is trained by X-predictor.
- **Apply Model operator** applies the trained Naïve Bayes model on test set from X-Predictor.
- **Performance operator** is of classification type. It evaluates the model and output the performance vector containing accuracy, confusion matrix etc.

4.3.2.2 Results





Row No.	Wine	prediction(...	confide...	confid...	volatile acidity	fixed acidity	citric acid	residual sug...	chlorides	free sulf
1	false	false	1	0	0.270	7	0.360	20.700	0.045	45
2	false	false	0.999	0.001	0.300	6.300	0.340	1.600	0.049	14
3	false	false	1.000	0.000	0.280	8.100	0.400	6.900	0.050	30
4	false	false	1.000	0.000	0.230	7.200	0.320	8.500	0.058	47
5	false	false	1.000	0.000	0.230	7.200	0.320	8.500	0.058	47
6	false	false	1.000	0.000	0.280	8.100	0.400	6.900	0.050	30
7	false	false	1.000	0.000	0.320	6.200	0.160	7	0.045	30
8	false	false	1	0	0.270	7	0.360	20.700	0.045	45
9	false	false	0.999	0.001	0.300	6.300	0.340	1.600	0.049	14
10	false	false	1.000	0.000	0.220	8.100	0.430	1.500	0.044	28
11	false	false	0.994	0.006	0.270	8.100	0.410	1.450	0.033	11
12	false	false	0.980	0.020	0.230	8.600	0.400	4.200	0.035	17
13	false	false	0.970	0.030	0.180	7.900	0.370	1.200	0.040	16
14	false	false	1.000	0.000	0.160	6.600	0.400	1.500	0.044	48
15	false	false	1	0	0.420	8.300	0.620	19.250	0.040	41
16	false	false	1.000	0.000	0.170	6.600	0.380	1.500	0.032	28
17	false	false	0.949	0.051	0.480	6.300	0.040	1.100	0.046	30

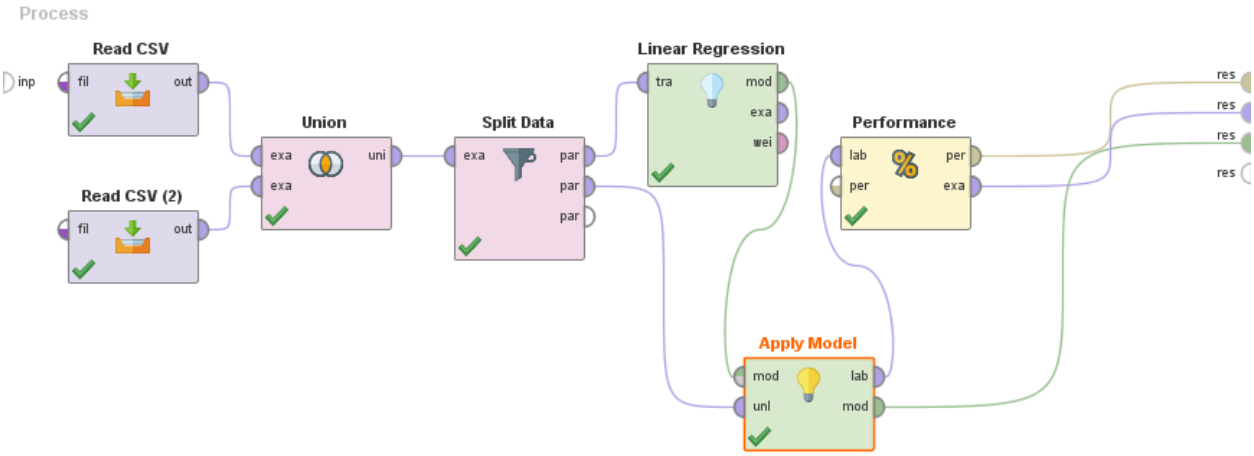
The performance vector shows the accuracy of the classifier, the confusion matrix etc. The ROC curve tells about the classifier performance. The Example set table shows the quality score true and predicted values.

4.4 LINEAR REGRESSION

The Linear Regression is performed on red and wine data sets combined to predict the wine quality score.

4.4.1 Linear regression with Set Validation

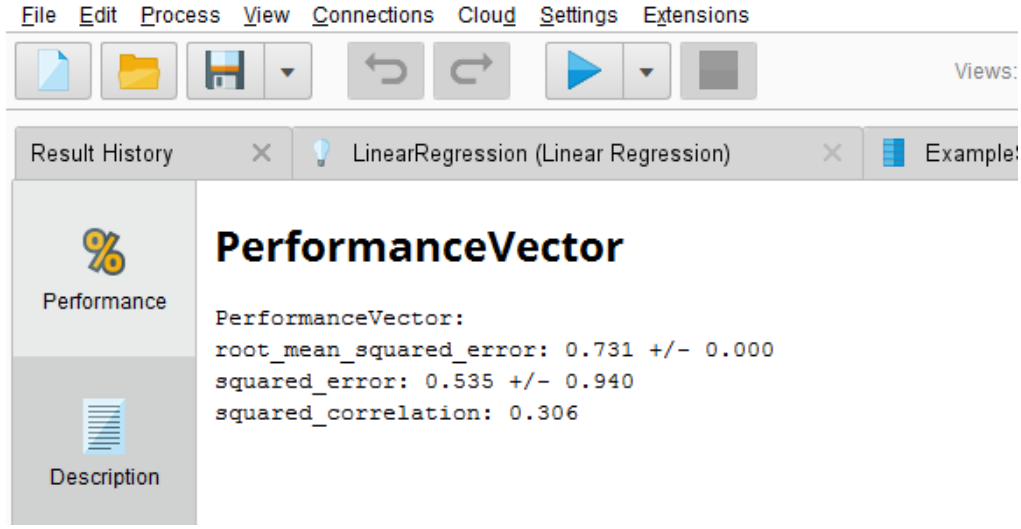
4.4.1.1 Process Design

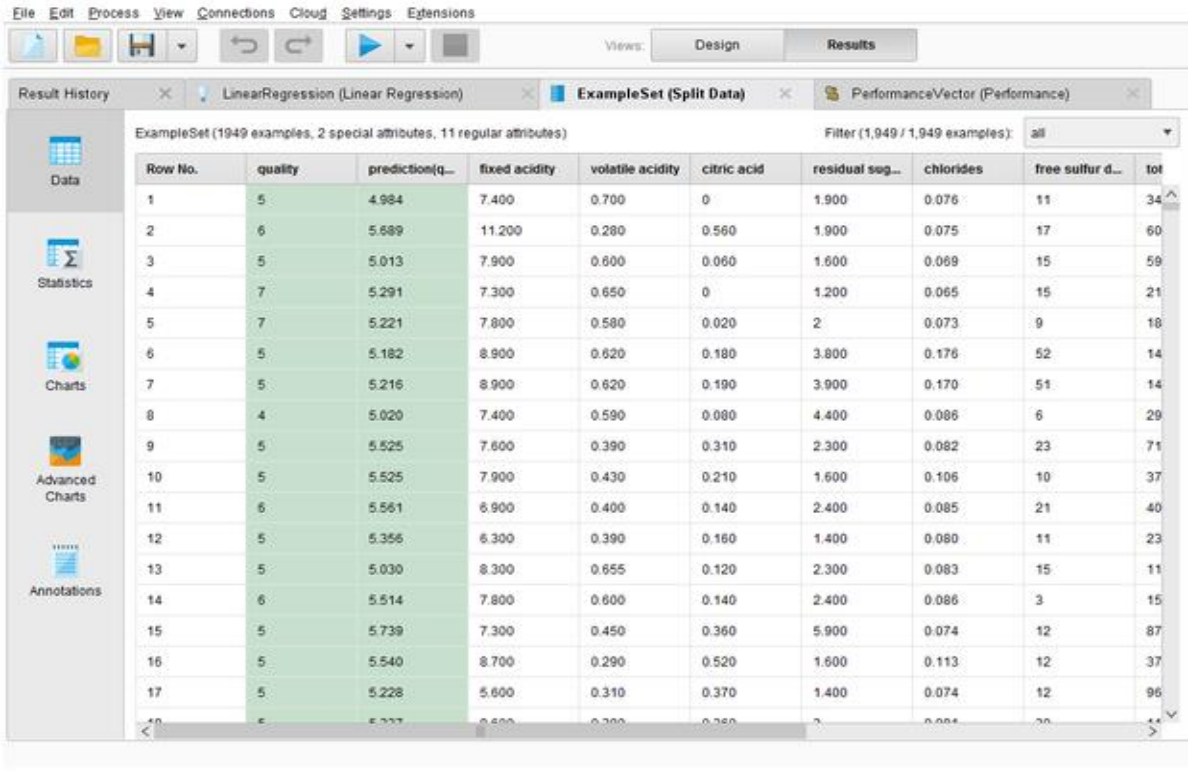


Following operators were used to implement the process:

- **Read CSV** operator is used to read the dataset.
- **Union operator** is used to combine two datasets.
- **Split Data operator** is used to perform validation by splitting data into 70% training set and 30% test set.
- **Linear Regression operator** implements the Linear Regression Algorithm and is trained using training partition of spilt data output.
- **Apply Model operator** applies the trained Linear Regression model on test partition of the split data output.
- **Performance operator** is of regression type. It evaluates the model and output the performance vector containing root squared error, prediction error, absolute error etc.

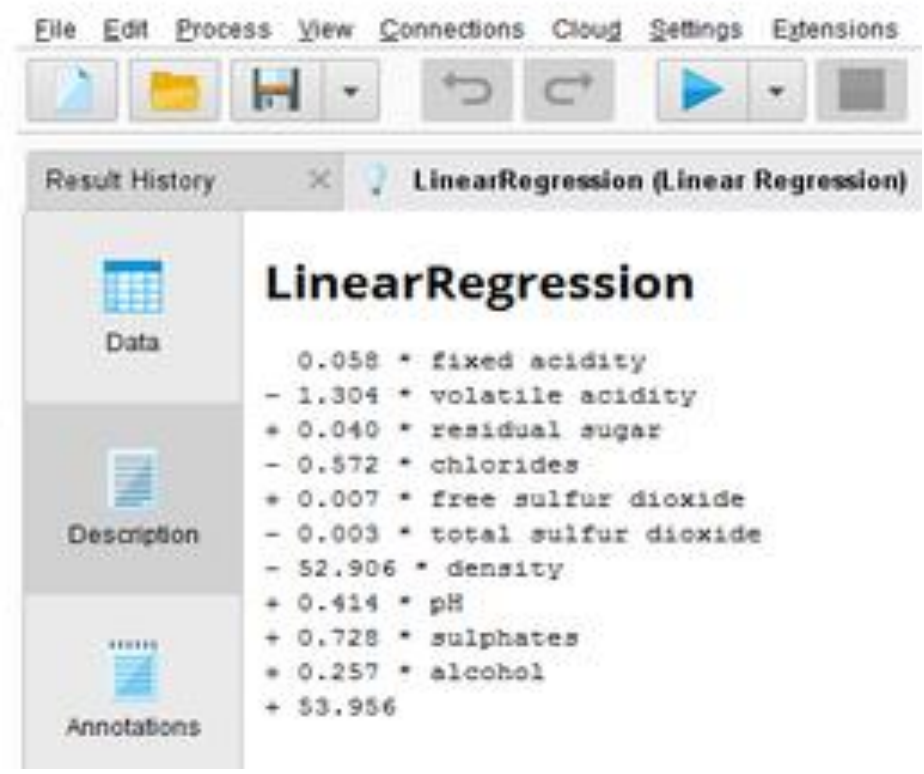
4.4.1.2 Results





LinearRegression (Linear Regression) | ExampleSet (Split Data) | PerformanceVector (Performance)

Attribute	Coefficient	Std. Error	Std. Coefficient	Tolerance	t-Stat	p-Value	Code
fixed acidity	0.058	0.018	0.088	0.967	3.245	0.001	***
volatile acidity	-1.304	0.086	-0.244	0.937	-15.128	0	****
residual sugar	0.040	0.006	0.218	0.981	6.397	0.000	****
chlorides	-0.572	0.400	-0.023	0.880	-1.430	0.153	
free sulfur dioxide	0.007	0.001	0.139	1.000	7.596	0.000	****
total sulfur dioxide	-0.003	0.000	-0.167	0.998	-7.811	0.000	****
density	-52.906	14.308	-0.183	0.698	-3.698	0.000	****
pH	0.414	0.107	0.077	1.000	3.857	0.000	****
sulphates	0.728	0.090	0.125	0.992	8.090	0.000	****
alcohol	0.257	0.020	0.353	0.503	13.169	0	****
(Intercept)	53.956	14.014	?	?	3.850	0.000	****



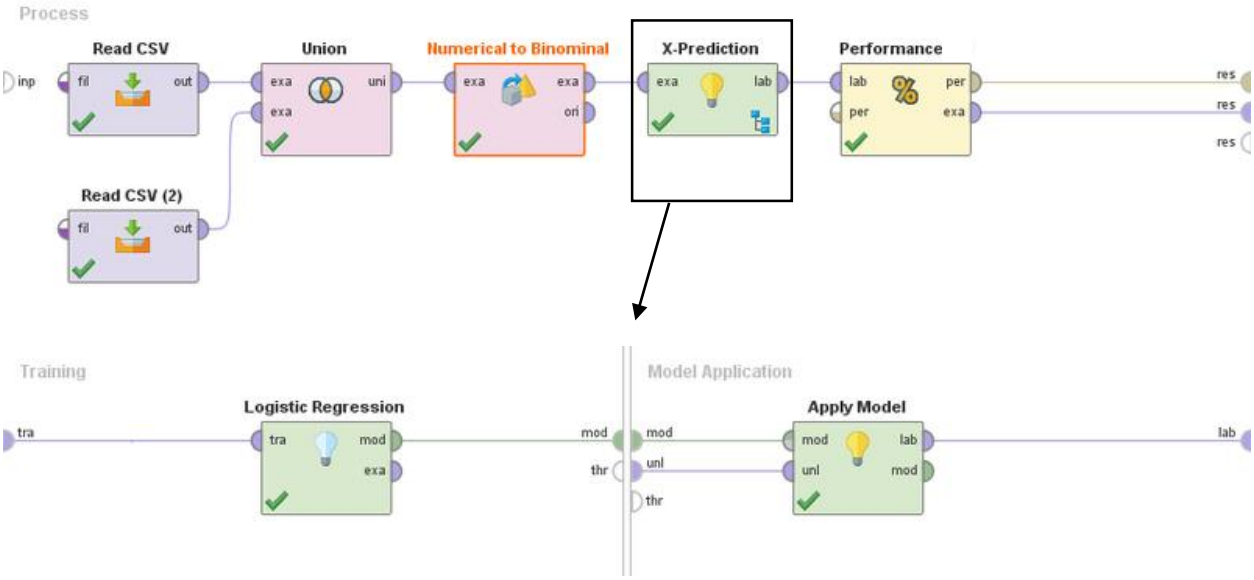
The results show the performance vector with root mean squared error, squared error and squared correlation. The example set shows the true and predicted values for wine quality. The Linear Regression table shows the predictors with coefficient values, p-values etc. The function for predicting quality from predictors is displayed in last screen shot.

4.5 LOGISTIC REGRESSION

The Logistic Regression is implemented to classify wine depending on quality score and wine type.

4.5.1 Classifying the wine depending on quality score

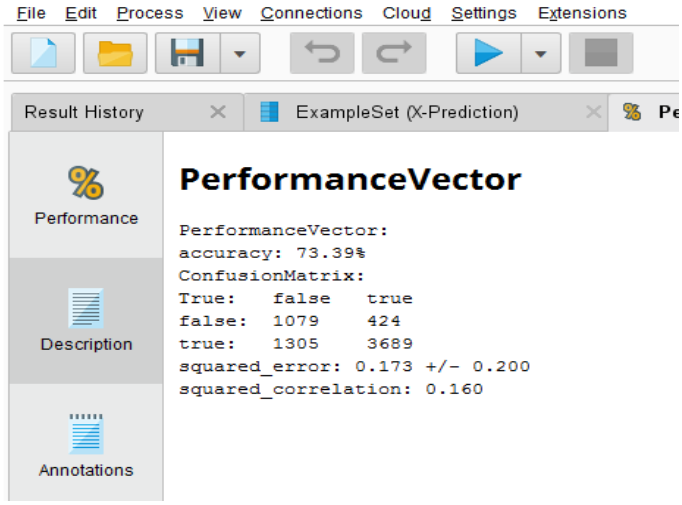
4.5.1.1 Process Design

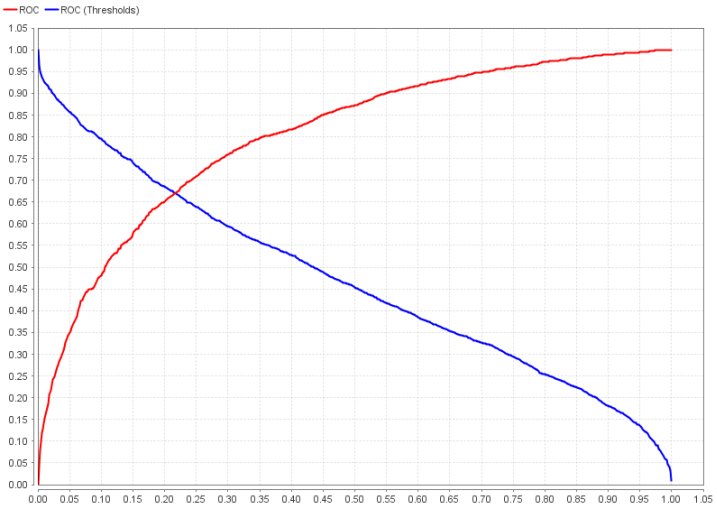


Following operators were used to implement the process:

- **Read CSV** operator is used to read the dataset.
- **Union operator** is used to combine two datasets.
- **Numerical to Binomial operator** is used to convert the quality attribute into binary type. If quality is less than 5 then it is set to false otherwise true.
- **X-Prediction operator** performs the cross validation on data set. It is a nested operator. It has two subprocesses: a training subprocess and a testing subprocess. The training subprocess is used for training a model. The trained model is then applied in the testing subprocess. The testing subprocess returns a labeled ExampleSet.
- **Logistic Regression operator** implements the Logistic Regression Algorithm and is trained by X-predictor.
- **Apply Model operator** applies the trained Logistic Regression model on test set from X-Predictor.
- **Performance operator** is of classification type. It evaluates the model and output the performance vector containing accuracy, confusion matrix etc.

4.5.1.2 Results



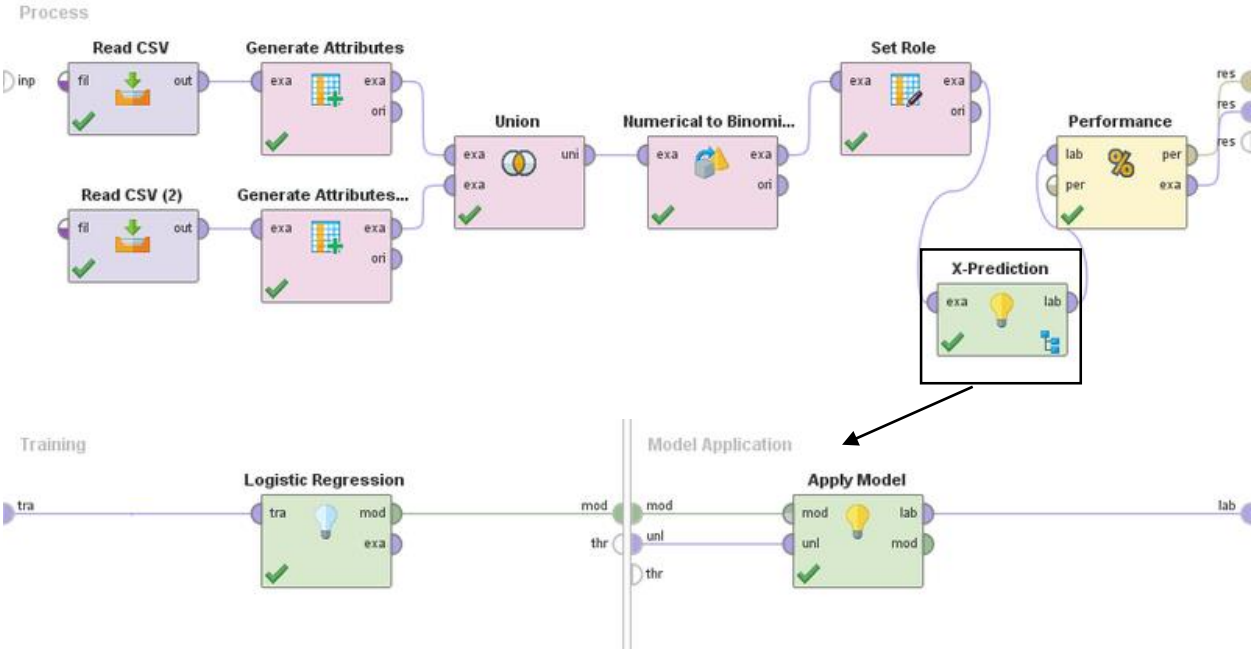


Row No.	quality	prediction(q...	confidence(f...	confidence(t...	fixed acidity	volatile acidity	citric acid	residual sug...	chlorides
1	true	true	0.482	0.518	7	0.270	0.360	20.700	0.045
2	true	false	0.680	0.320	6.300	0.300	0.340	1.600	0.049
3	true	true	0.305	0.695	8.100	0.280	0.400	6.900	0.050
4	true	true	0.407	0.593	7.200	0.230	0.320	8.500	0.058
5	true	true	0.407	0.593	7.200	0.230	0.320	8.500	0.058
6	true	true	0.319	0.681	8.100	0.280	0.400	6.900	0.050
7	true	true	0.530	0.470	6.200	0.320	0.160	7	0.045
8	true	true	0.482	0.518	7	0.270	0.360	20.700	0.045
9	true	false	0.680	0.320	6.300	0.300	0.340	1.600	0.049
10	true	true	0.248	0.752	8.100	0.220	0.430	1.500	0.044
11	false	true	0.093	0.907	8.100	0.270	0.410	1.450	0.033
12	false	true	0.433	0.567	8.600	0.230	0.400	4.200	0.035
13	false	true	0.140	0.860	7.900	0.180	0.370	1.200	0.040
14	true	true	0.037	0.963	6.600	0.160	0.400	1.500	0.044
15	false	true	0.401	0.599	8.300	0.420	0.620	19.250	0.040
16	true	true	0.106	0.894	6.600	0.170	0.380	1.500	0.032
17	true	false	0.740	0.260	6.300	0.480	0.040	1.100	0.046

The performance vector shows the accuracy of the classifier, the confusion matrix etc. The ROC curve tells about the classifier performance. The Example set table shows the quality score true and predicted values.

4.5.2 Classifying the wine depending on type- Red or White wine

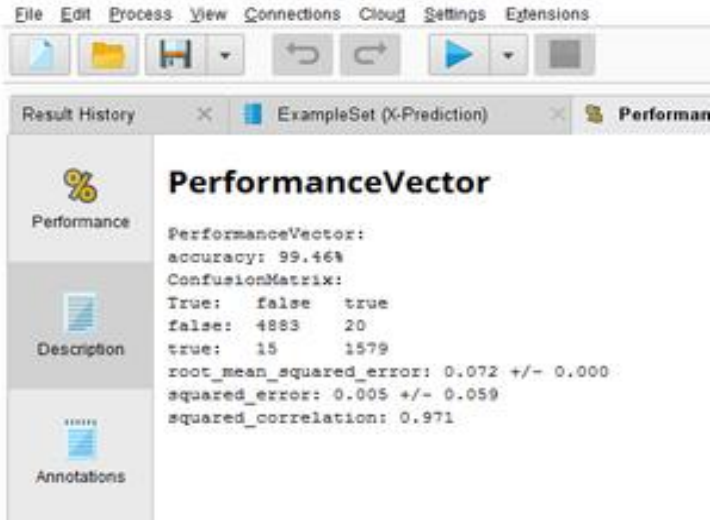
4.5.2.1 Process Design

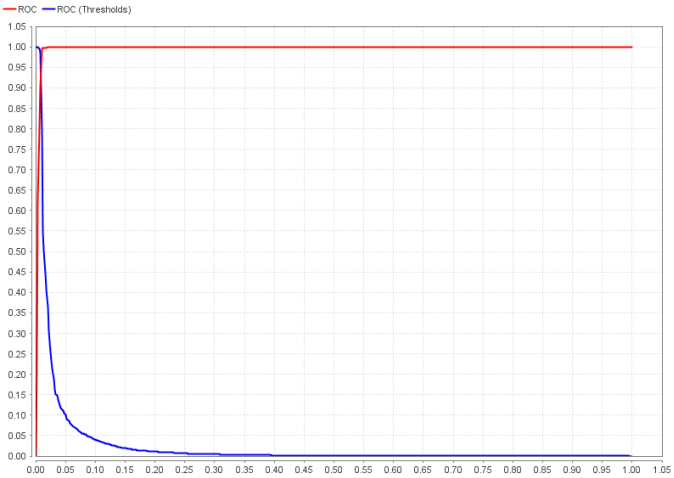


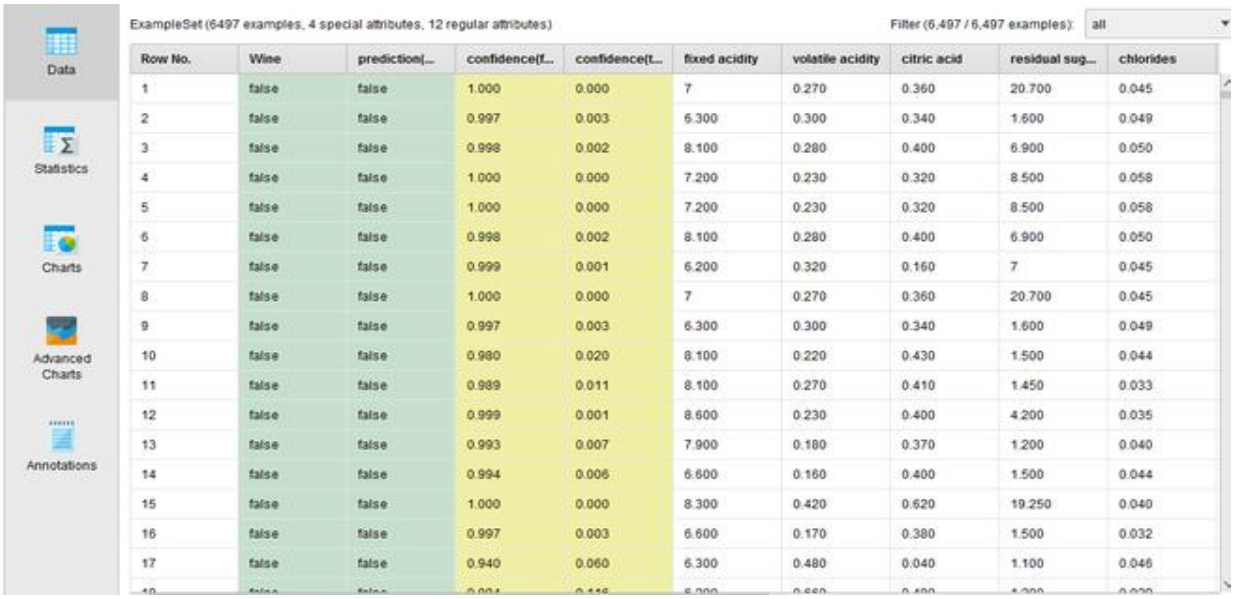
Following operators were used to implement the process:

- **Read CSV** operator is used to read the dataset.
- **Generate Attributes** is used to generate new attribute *wine* for wine type-red or white wine.
- **Union operator** is used to combine two datasets.
- **Numerical to Binomial operator** is used to convert the wine attribute into binary type. If wine type is red the wine is set false otherwise true.
- **Set Role operator** is used to set the role for wine attribute as label.
- **X-Prediction operator** performs the cross validation on data set. It is a nested operator. It has two subprocesses: a training subprocess and a testing subprocess. The training subprocess is used for training a model. The trained model is then applied in the testing subprocess. The testing subprocess returns a labeled ExampleSet.
- **Logistic Regression operator** implements the Logistic Regression Algorithm and is trained by X-predictor.
- **Apply Model operator** applies the trained Logistic Regression model on test set from X-Predictor.
- **Performance operator** is of classification type. It evaluates the model and output the performance vector containing accuracy, confusion matrix etc.

4.5.2.2 Results







Row No.	Wine	prediction(...	confidence(f...	confidence(t...	fixed acidity	volatile acidity	citric acid	residual sug...	chlorides
1	false	false	1.000	0.000	7	0.270	0.360	20.700	0.045
2	false	false	0.997	0.003	6.300	0.300	0.340	1.600	0.049
3	false	false	0.998	0.002	8.100	0.280	0.400	6.900	0.050
4	false	false	1.000	0.000	7.200	0.230	0.320	8.500	0.058
5	false	false	1.000	0.000	7.200	0.230	0.320	8.500	0.058
6	false	false	0.998	0.002	8.100	0.280	0.400	6.900	0.050
7	false	false	0.999	0.001	6.200	0.320	0.160	7	0.045
8	false	false	1.000	0.000	7	0.270	0.360	20.700	0.045
9	false	false	0.997	0.003	6.300	0.300	0.340	1.600	0.049
10	false	false	0.980	0.020	8.100	0.220	0.430	1.500	0.044
11	false	false	0.989	0.011	8.100	0.270	0.410	1.450	0.033
12	false	false	0.999	0.001	8.600	0.230	0.400	4.200	0.035
13	false	false	0.993	0.007	7.900	0.180	0.370	1.200	0.040
14	false	false	0.994	0.006	6.600	0.160	0.400	1.500	0.044
15	false	false	1.000	0.000	8.300	0.420	0.620	19.250	0.040
16	false	false	0.997	0.003	6.600	0.170	0.380	1.500	0.032
17	false	false	0.940	0.060	6.300	0.480	0.040	1.100	0.046

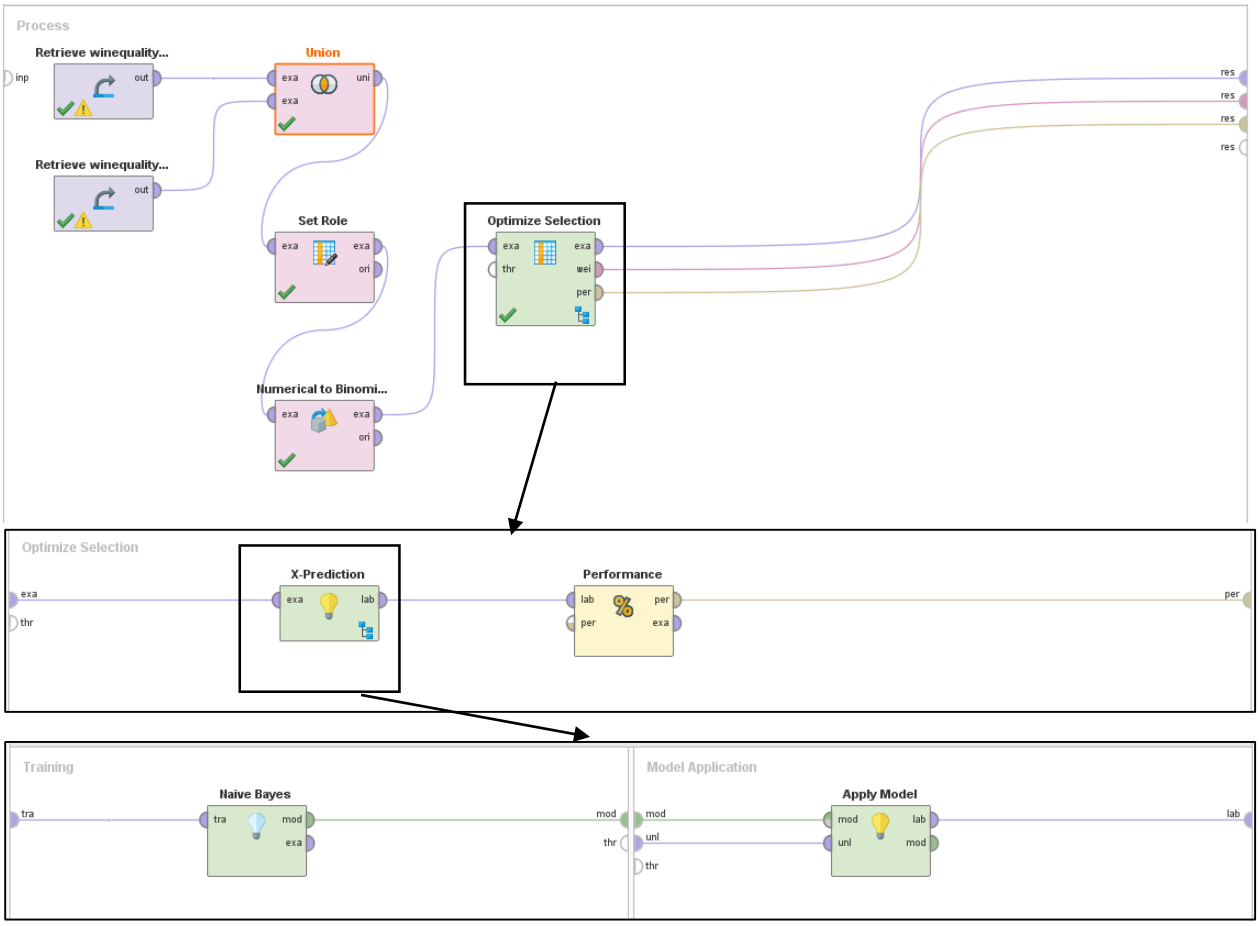
The performance vector shows the accuracy of the classifier, the confusion matrix etc. The ROC curve tells about the classifier performance. The Example set table shows the quality score true and predicted values.

5 OPTIMIZATION USING FEATURE SELECTION

Feature selection improves the performance. The optimization using feature selection is performed on all the Classifier for classifying wine depending on the quality score.

5.1 CLASSIFYING THE WINE DEPENDING ON QUALITY SCORE USING FEATURE SELECTION

5.1.1 Process Design



The above diagram shows the process design for optimizing the Naïve Bayes classifier using feature selection. The same can be modified for different classifiers and regression models.

Following operators were used to implement the process:

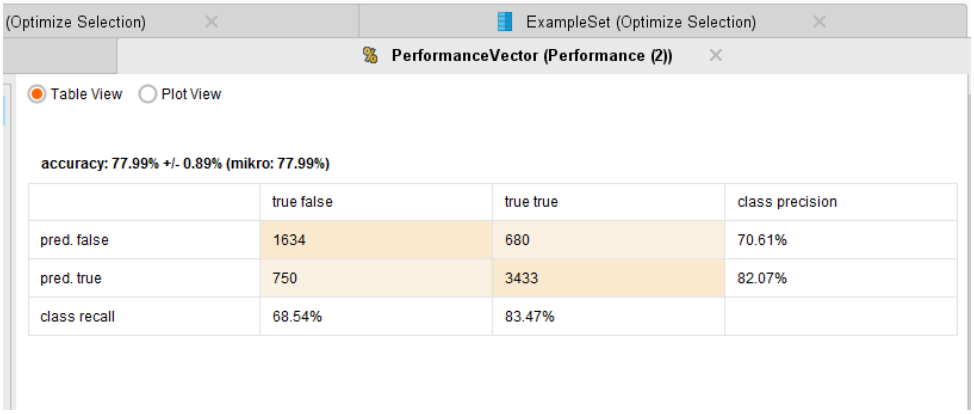
- **Retrieve** operator is used to read the dataset from repository.
- **Union operator** is used to combine two datasets.

- **Set Role operator** is used to set role of quality attribute as label.
- **Optimize selection** This operator selects the most relevant attributes of the given ExampleSet. Two deterministic greedy feature selection algorithms 'forward selection' and 'backward elimination' are used for feature selection. Here we used backward elimination.
- **Numerical to Binomial operator** is used to convert the quality attribute into binary type. If quality is less than 5 then it is set to false otherwise true.
- **X-Prediction operator** performs the cross validation on data set. It is a nested operator. It has two subprocesses: a training subprocess and a testing subprocess. The training subprocess is used for training a model. The trained model is then applied in the testing subprocess. The testing subprocess returns a labeled ExampleSet.
- **Naïve Bayes operator** implements the Naïve Bayes Algorithm and is trained by X-predictor.
- **Apply Model operator** applies the trained Naïve Bayes model on test set from X-Predictor.
- **Performance operator** is of classification type. It evaluates the model and output the performance vector containing accuracy, confusion matrix etc.

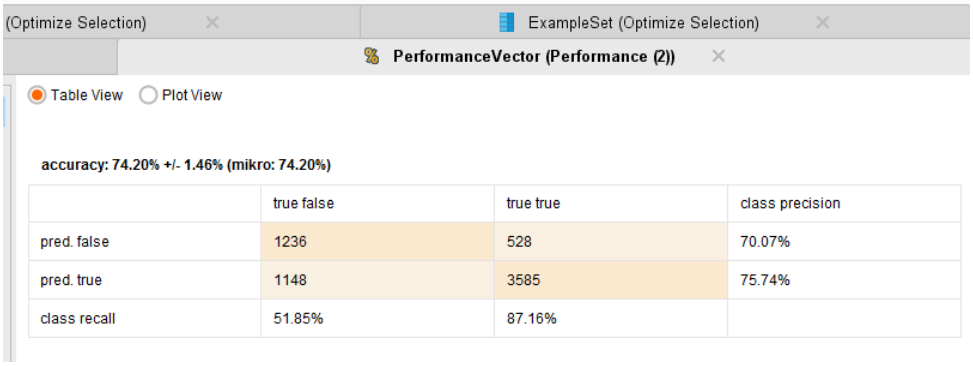
5.1.2 Results

Below are the results of feature selection for different models:

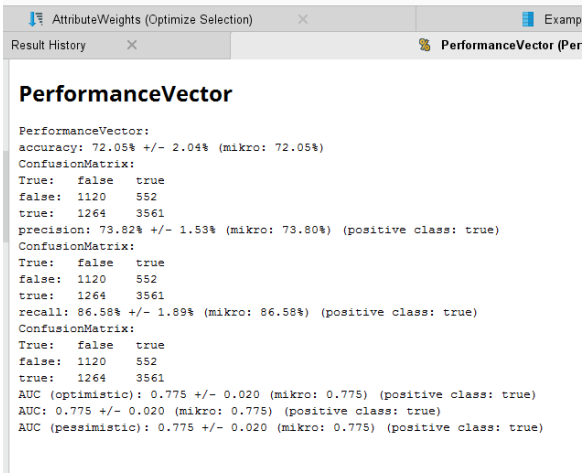
1. k-NN Classification:



2. Logistic Regression:



3. Naïve Byes Classification:



6 CONCLUSION

In this project different algorithms were implemented in Rapid Miner data mining tool. The following conclusions are drawn from analysis and prediction performed on wine data set:

- Predicting the Quality score using algorithms:

Algorithm	**Linear Regression**	**k-NN Regression**
Root Mean Squared error	0.731	0.807 for k=17

Thus Linear Regression performs better than k-NN Regression.

- Classifying the wine dataset into quality<5 and quality>=5 using algorithms:

Algorithm	**Logistic Regression**	**Naïve Bayes Classifier**	**k-NN Classification**
Accuracy	73.39%	67.95%	75.60% for k=1

The Accuracy of Logistic Regression is the high. Thus it gives better classification.

- Classifying the data into Red or White Wine using algorithms:

Algorithm	**Logistic Regression**	**Naïve Bayes Classifier**	**k-NN Classification**
Accuracy	73.39%	67.95%	75.60% for k=1

The Accuracy of Logistic Regression is the high. Thus it gives better classification.

The optimization for the algorithms is performed by feature selection. The results were improved because of feature selection.

Algorithm	**Logistic Regression**	**Naïve Bayes Classifier**	**k-NN Classification**
Accuracy before optimization	73.39%	67.95%	75.60% for k=1
Accuracy before optimization	74.20%	72.05%	77.99% for k=1

7 REFERENCES

[1] https://rapidminer.com/

[2] https://www.wikipedia.org/

[3] P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Modeling wine preferences by data mining from physicochemical properties. In Decision Support Systems>, Elsevier, 47(4):547-553. ISSN: 0167-9236.