Experiment 03 - Classification using Rapid Miner tool

Roll No.	19
Name	Manav Jawrani
Class	D15A
Subject	Business Intelligence Lab
LO Mapped	LO3: Implement the appropriate data mining methods like classification, clustering or association mining on large data sets using open-source tools like WEKA
Grade	

DATE:

Experiment No. 3

Aim - Classification (Decision tree and Naive Bayes classification algorithms) using rapid miner tool.

Theory -

- Introduction :

Rapid Miner is a powerful and intuitive data science platform that provides a variety of tools and functionalities for data preparation, machine learning, deep learning, and predictive analysis. The Rapidminer platform is designed to help users extract valuable insights and knowledge from complex datasets, even if they have little to no coding experience.

- Classification algorithms :

Classification algorithms are a type of machine learning algorithm used to categorize or classify data into predefined classes or categories. Some of the most commonly used classification algorithms include:

1. Logistic Regression - A statistical model that uses a logistic function to model binary outcomes.
2. Decision trees - A method that uses a tree-like model to represent decisions and their possible consequences including the final outcome.
3. Naive Bayes - A probabilistic model that assumes independence between input features and calculates the probability of each class based on input.

DATE:

Metrics -

In machine learning, metrics are used to measure the performance of model or algorithm. There are various metrics used for different purposes, such as evaluating the accuracy, precision, recall and F1 score of a classification model, or the mean squared error and R-squared of a regression model. Here are some commonly used metrics, along with their definitions and formulas:

- Accuracy: measures the proportion of correctly classified instances.

  Formula:

  ~~TP+TN~~

  (TP + TN) / (TP + TN + FP + FN), where TP = True Positive, TN = True Negative, FP = False Positive, FN = False Negative.

- Precision: measures the proportion of correctly identified positive instances out of all predicted positive instances.

  Formula:

  TP / (TP + FP)

Dataset:
The dataset contains information about patients who had a stroke. It includes 5110 observations and 12 variables which are:

1. id: Unique identifier for each patient.
Attribute type: Categorical (Nominal).
2. gender: Gender of the patient.
Attribute type: Categorical (Nominal).
3. age: Age of the patient in years.
Attribute type: Continuous (Ratio).
4. hypertension: 0 if the patient doesn't have hypertension, 1 if the patient has hypertension.
Attribute type: Categorical (Nominal).
5. heart_disease: 0 if the patient doesn't have any heart diseases, 1 if the patient has a heart disease.
Attribute type: Categorical (Nominal).
6. ever_married: Whether the patient has ever been married or not.
Attribute type: Categorical (Nominal).
7. work_type: The type of work the patient does.
Attribute type: Categorical (Nominal).
8. Residence_type: The type of residence of patient.
Attribute type: Categorical (Nominal).
9. avg_glucose_level: The average glucose level of patient.
Attribute type: Continuous (Ratio).
10. bmi - The body mass index (bmi) of patient.
Attribute type: Continuous (Ratio)
11. Smoking_Status: The smoking status of patient.
Attribute type: Categorical (Nominal).

12. Stroke : whether the patient had stroke or not.
Attribute type : categorical (Nominal).

• Observation :
These are the final accuracies which we found -

	Training data	Test data
Decision Tree	64.10 %	64.91 %
Naive Bayes	65.22 %	69.21 %

• Conclusion :
Based on given final accuracies for the training and test data of Decision Tree and Naive Bayes classifiers, we can conclude the following:

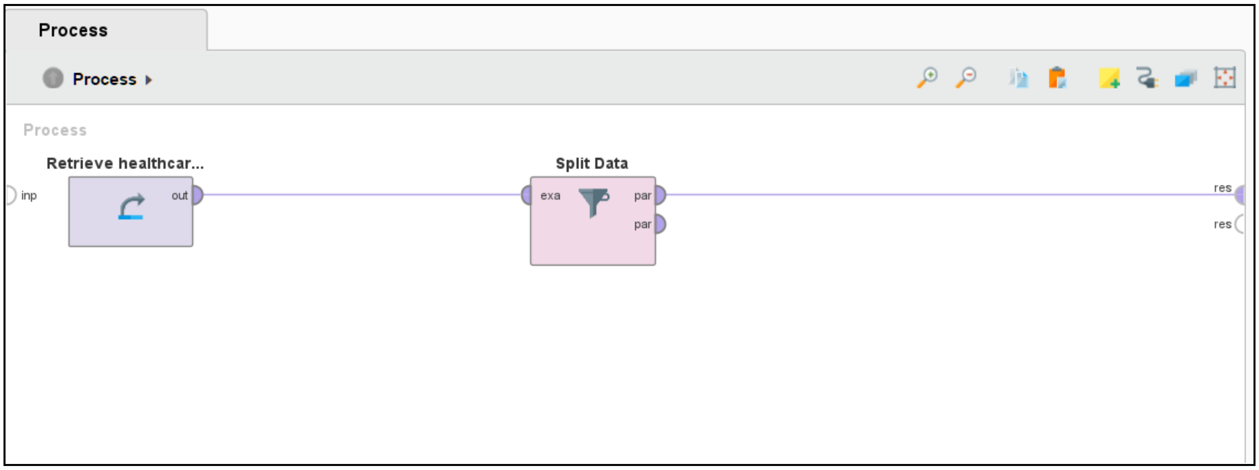
1. Both decision tree and naive bayes classifiers have similar accuracy levels on the training data.
2. Naive Bayes outperforms the decision tree classifier on test data, with a higher accuracy level of 69.21 % compared to 64.91 %.
3. Both classifiers have slightly lower accuracy levels on the test data compared to the training data, including a potential issue of overfitting.
4. Overall, Naive Bayes appears to be the better-performing classifier for this particular problem based on the test data accuracy.

Implementation:

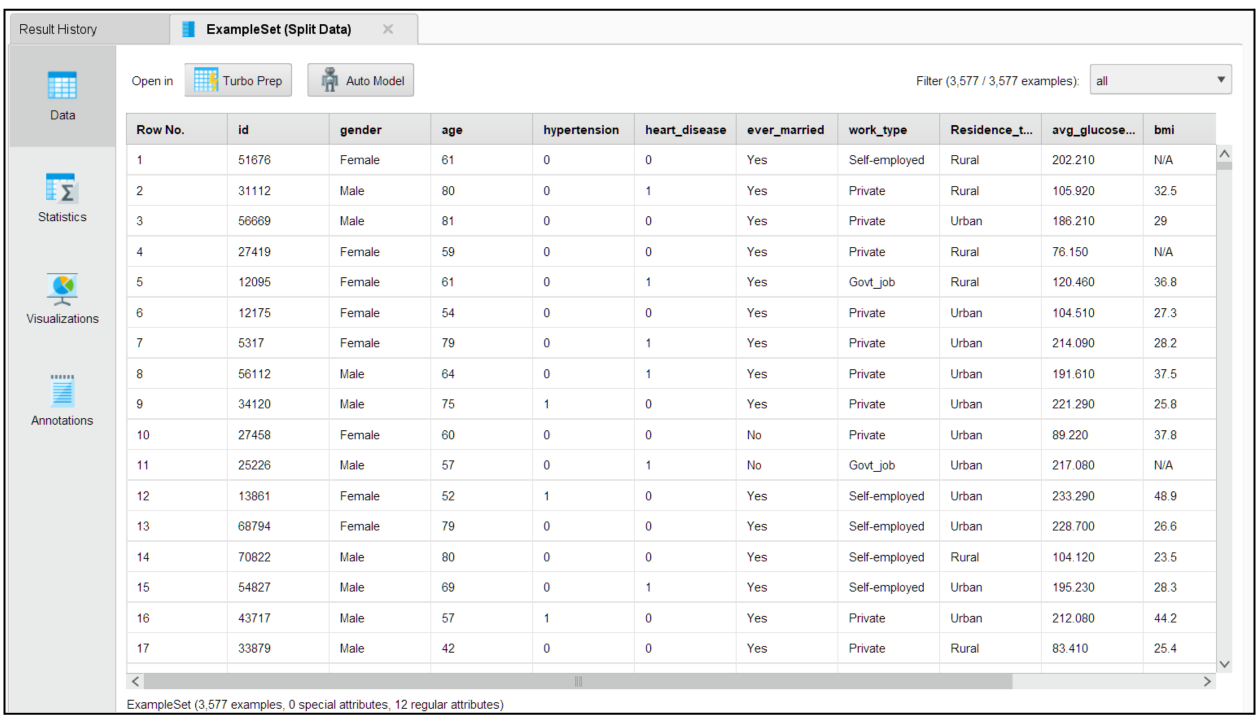
** Initially we had 5110 entries/rows **

1. Splitting the dataset in training data

Process -



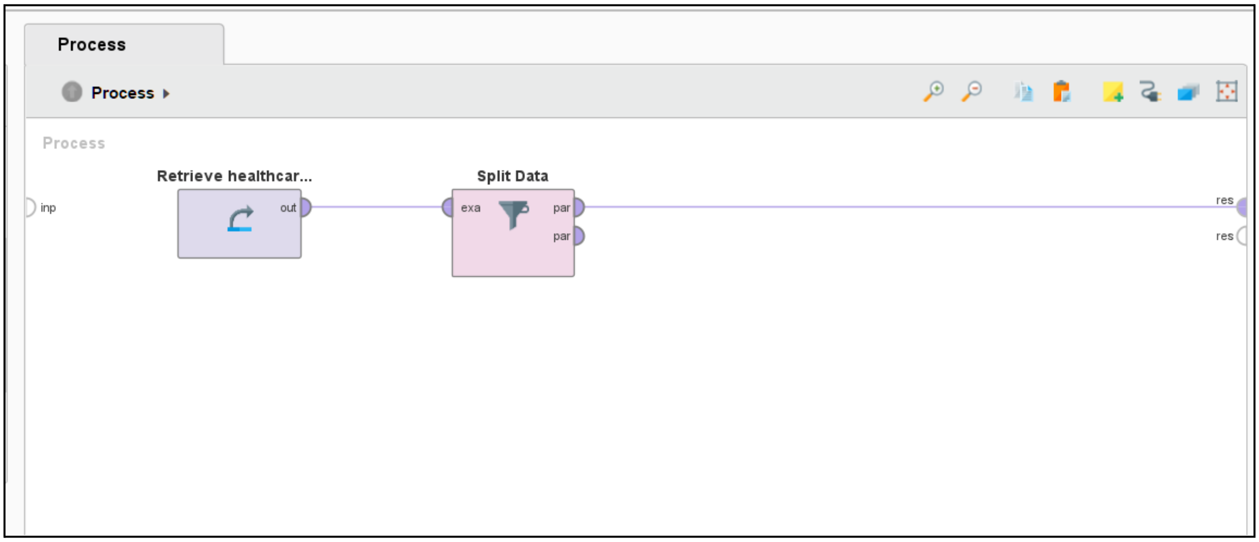
Dataset -



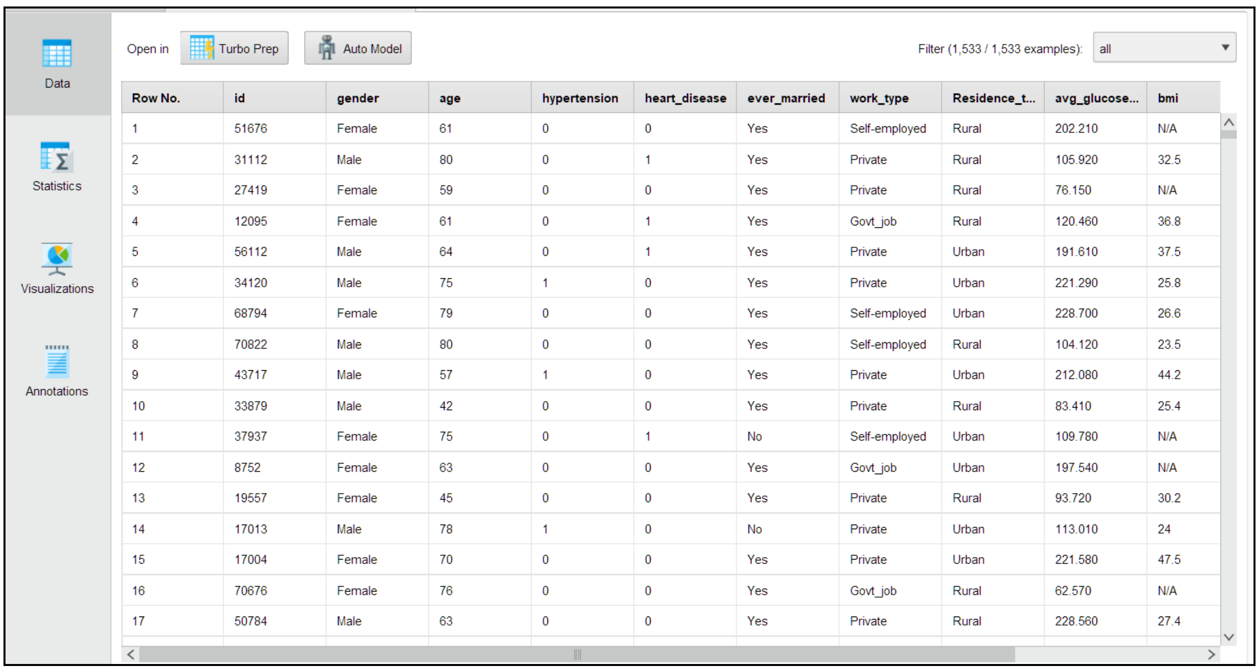
Row No.	id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_t...	avg_glucose...	bmi
1	51676	Female	61	0	0	Yes	Self-employed	Rural	202.210	N/A
2	31112	Male	80	0	1	Yes	Private	Rural	105.920	32.5
3	56669	Male	81	0	0	Yes	Private	Urban	186.210	29
4	27419	Female	59	0	0	Yes	Private	Rural	76.150	N/A
5	12095	Female	61	0	1	Yes	Govt_job	Rural	120.460	36.8
6	12175	Female	54	0	0	Yes	Private	Urban	104.510	27.3
7	5317	Female	79	0	1	Yes	Private	Urban	214.090	28.2
8	56112	Male	64	0	1	Yes	Private	Urban	191.610	37.5
9	34120	Male	75	1	0	Yes	Private	Urban	221.290	25.8
10	27458	Female	60	0	0	No	Private	Urban	89.220	37.8
11	25226	Male	57	0	1	No	Govt_job	Urban	217.080	N/A
12	13861	Female	52	1	0	Yes	Self-employed	Urban	233.290	48.9
13	68794	Female	79	0	0	Yes	Self-employed	Urban	228.700	26.6
14	70822	Male	80	0	0	Yes	Self-employed	Rural	104.120	23.5
15	54827	Male	69	0	1	Yes	Self-employed	Urban	195.230	28.3
16	43717	Male	57	1	0	Yes	Private	Urban	212.080	44.2
17	33879	Male	42	0	0	Yes	Private	Rural	83.410	25.4

2. Splitting the dataset in test data

Process



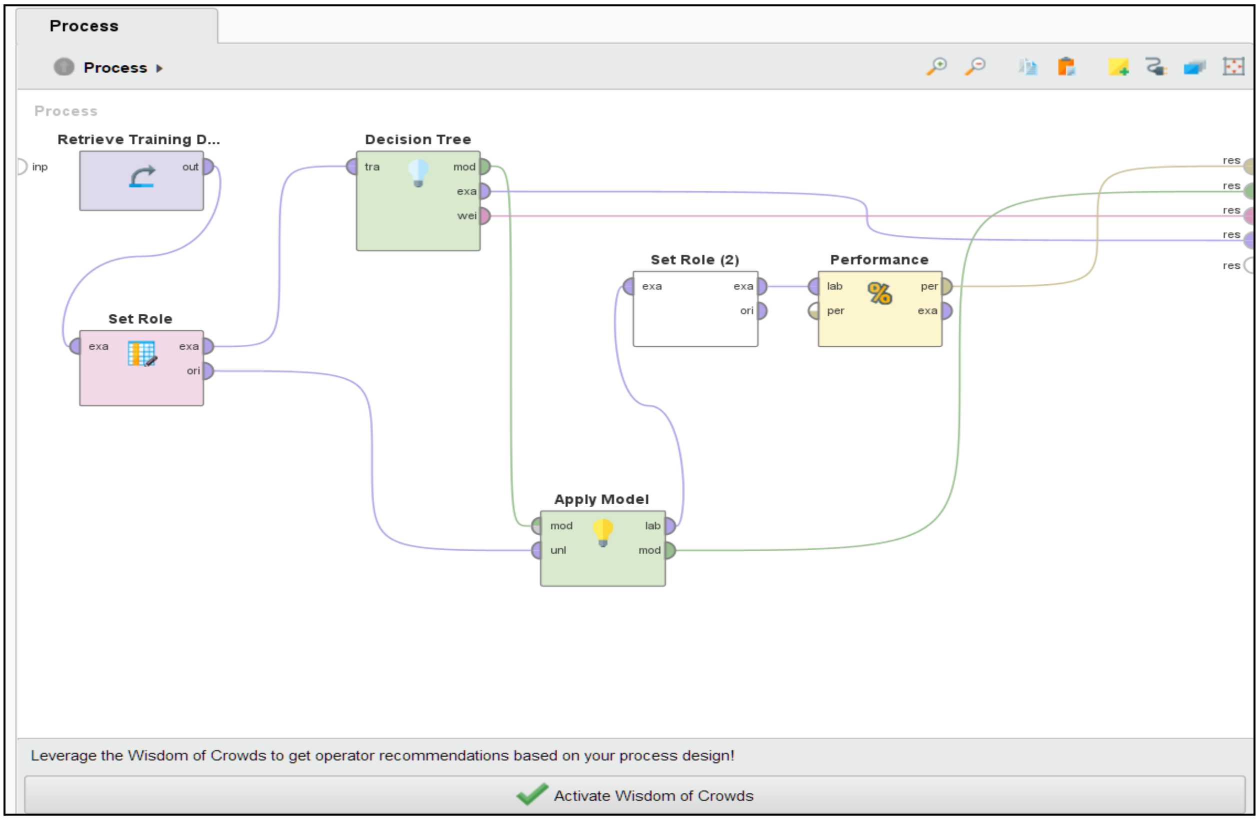
Dataset -



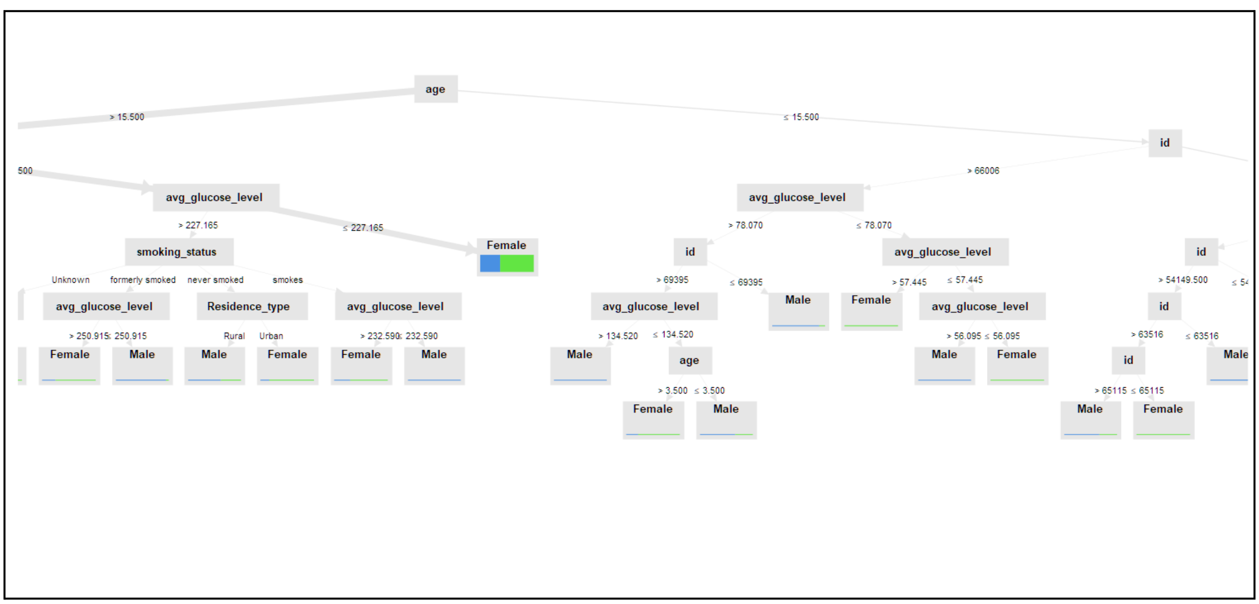
Row No.	id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_t...	avg_glucose...	bmi
1	51676	Female	61	0	0	Yes	Self-employed	Rural	202.210	N/A
2	31112	Male	80	0	1	Yes	Private	Rural	105.920	32.5
3	27419	Female	59	0	0	Yes	Private	Rural	76.150	N/A
4	12095	Female	61	0	1	Yes	Govt_job	Rural	120.460	36.8
5	56112	Male	64	0	1	Yes	Private	Urban	191.610	37.5
6	34120	Male	75	1	0	Yes	Private	Urban	221.290	25.8
7	68794	Female	79	0	0	Yes	Self-employed	Urban	228.700	26.6
8	70822	Male	80	0	0	Yes	Self-employed	Rural	104.120	23.5
9	43717	Male	57	1	0	Yes	Private	Urban	212.080	44.2
10	33879	Male	42	0	0	Yes	Private	Rural	83.410	25.4
11	37937	Female	75	0	1	No	Self-employed	Urban	109.780	N/A
12	8752	Female	63	0	0	Yes	Govt_job	Urban	197.540	N/A
13	19557	Female	45	0	0	Yes	Private	Rural	93.720	30.2
14	17013	Male	78	1	0	No	Private	Urban	113.010	24
15	17004	Female	70	0	0	Yes	Private	Urban	221.580	47.5
16	70676	Female	76	0	0	Yes	Govt_job	Rural	62.570	N/A
17	50784	Male	63	0	0	Yes	Private	Rural	228.560	27.4

3. Decision tree of training data

Process -



Tree -



Performance matrix -

Criterion

accuracy

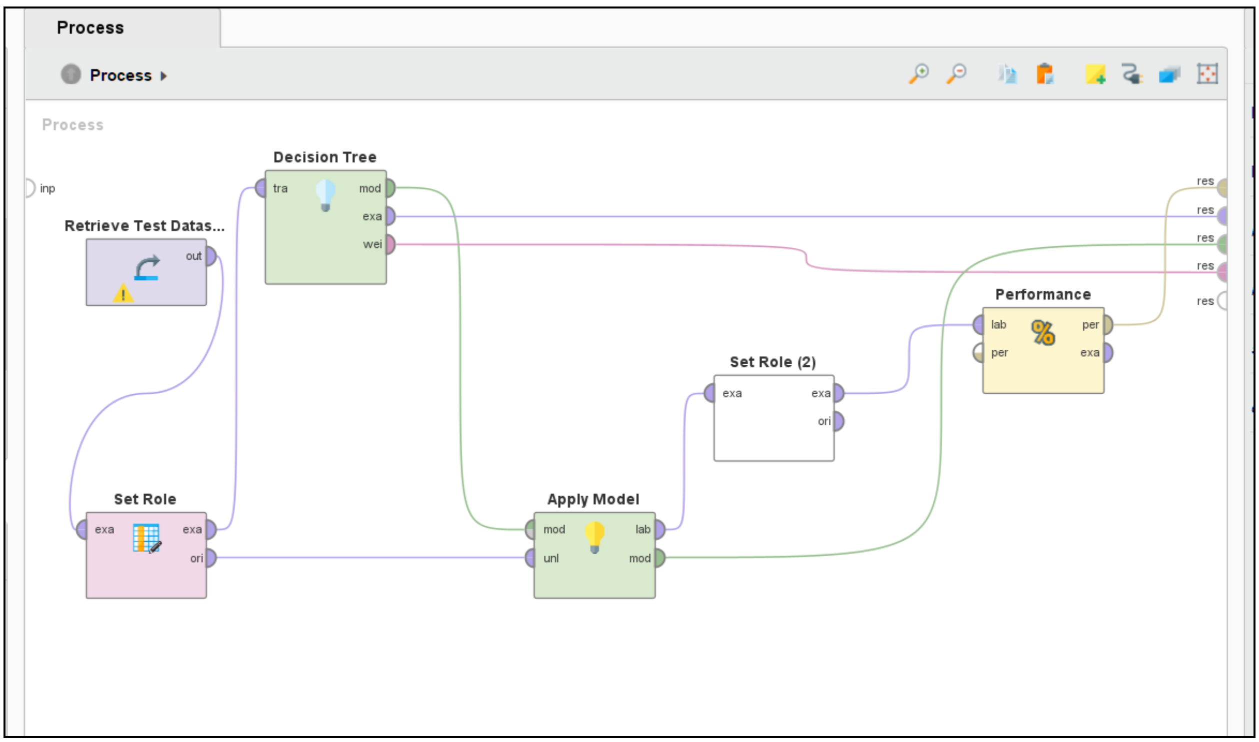
Table View Plot View

accuracy: 64.10%

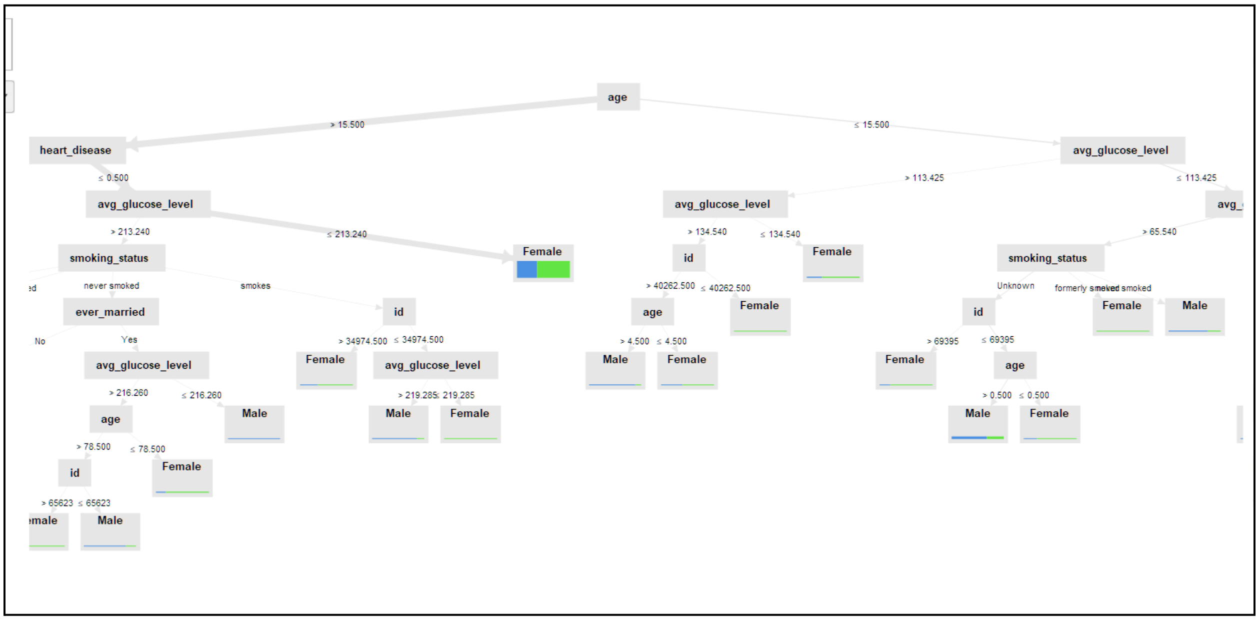
	true Male	true Female	true Other	class precision
pred. Male	409	208	0	66.29%
pred. Female	1075	1884	1	63.65%
pred. Other	0	0	0	0.00%
class recall	27.56%	90.06%	0.00%	

4 . Decision tree of test data

Process -



Tree -



Performance matrix -

Criterion

accuracy

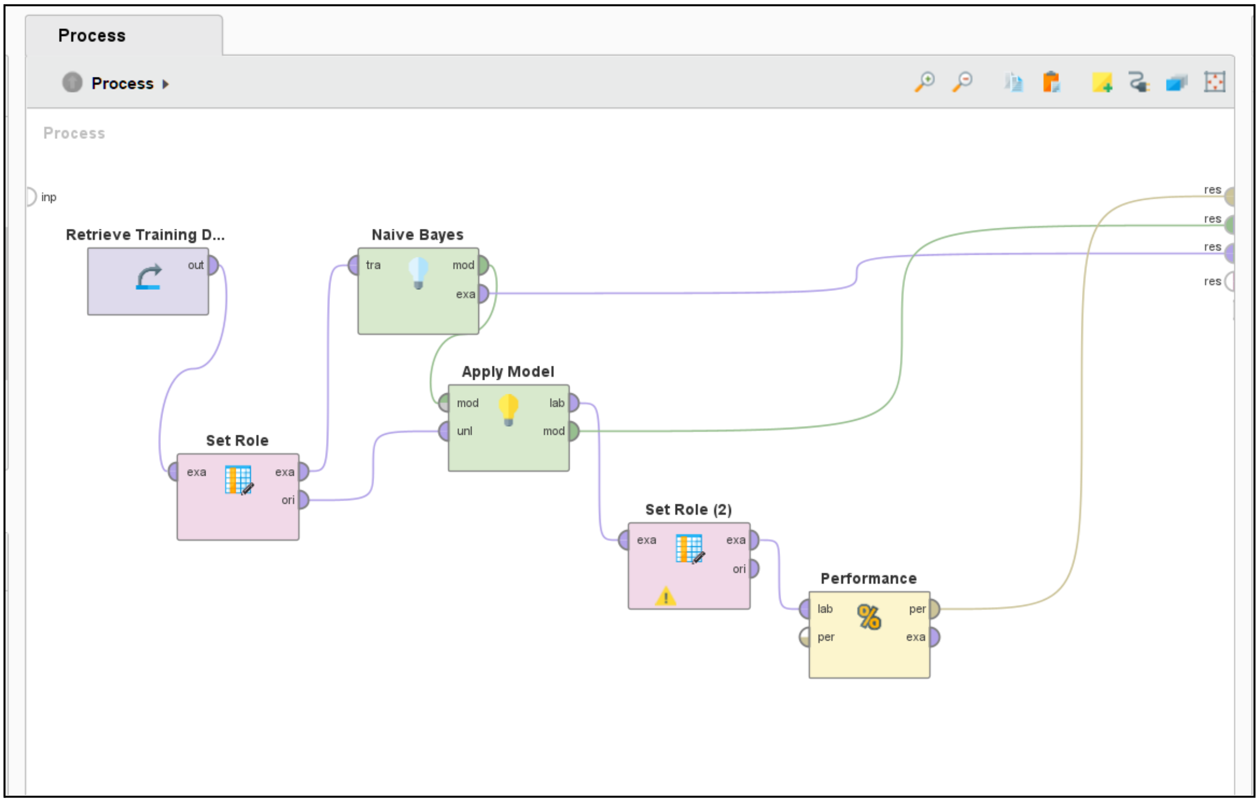
Table View Plot View

accuracy: 64.91%

	true Male	true Female	true Other	class precision
pred. Male	187	74	0	71.65%
pred. Female	464	808	0	63.52%
pred. Other	0	0	0	0.00%
class recall	28.73%	91.61%	0.00%	

5. Naive Bayes of training data

Process -



Simple Distribution -

SimpleDistribution

```
Distribution model for label attribute gender


Class Male (0.415)
11 distributions

Class Female (0.585)
11 distributions

Class Other (0.000)
11 distributions
```

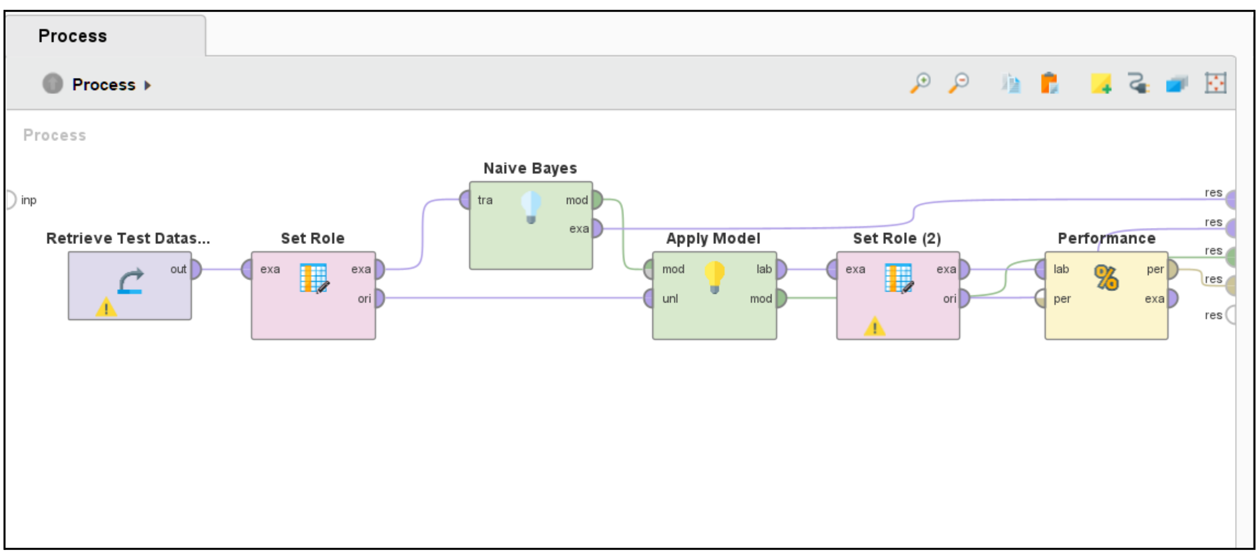
Performance matrix -

PerformanceVector

```
PerformanceVector:
accuracy: 65.22%
ConfusionMatrix:
True:    Male    Female  Other
Male:    545     305     0
Female:  939     1787    0
Other:   0       0       1
kappa: 0.237
ConfusionMatrix:
True:    Male    Female  Other
Male:    545     305     0
Female:  939     1787    0
Other:   0       0       1
```

6. Naive Bayes of test data

Process -



Simple Distribution -

SimpleDistribution

```
Distribution model for label attribute gender


Class Male (0.425)
11 distributions

Class Female (0.575)
11 distributions

Class Other (0.000)
11 distributions
```

Performance matrix -

PerformanceVector

```
PerformanceVector:
accuracy: 69.21%
ConfusionMatrix:
True:    Male    Female  Other
Male:    325     146     0
Female:  326     736     0
Other:   0       0       0
kappa: 0.346
ConfusionMatrix:
True:    Male    Female  Other
Male:    325     146     0
Female:  326     736     0
Other:   0       0       0
```