DMBI Lab

EXPERIMENT NO. 5

Name : Kaushik Kotian **Roll no.** 30 **Div :** D15B **Batch:**B

AIM : To explore Rapid Miner and implement classification models like Decision Tree and Naive Bayes.

1. Preprocess data. Split data into train and test set
2. Build Classification model using Rapid miner on training data (use decision tree and naive Bayes method)
3. Calculate metrics based on test data
4. Compare the models based on metrics and find which model is best suited for the dataset.

Link to the dataset :
https://www.kaggle.com/datasets/bhavikjikadara/loan-status-prediction?resource=downl oad

THEORY :

RapidMiner is a comprehensive open-source platform for data science and machine learning, providing a user-friendly interface for building, deploying, and maintaining predictive models. It offers a visual workflow environment where users can design and execute data analysis tasks without needing extensive programming knowledge. RapidMiner supports various data preprocessing techniques, feature engineering, model training using diverse algorithms such as decision trees and neural networks, and model evaluation with metrics like accuracy and F1-score. It also enables model deployment for making predictions on new data or integrating models into production systems. With its intuitive interface and robust functionality, RapidMiner empowers data scientists and analysts to explore data, create powerful machine learning models, and derive valuable insights from complex datasets.

A **classification model** is a type of machine learning model that is used to categorize input data into predefined classes or categories. The goal of a classification model is to learn the mapping between input features and the corresponding output labels, enabling it to accurately predict the class of new, unseen data points.

Key components of a classification model include:

1. Input Data: The data used to train and test the classification model typically consists of features or attributes that describe the input, along with the corresponding output labels or classes.

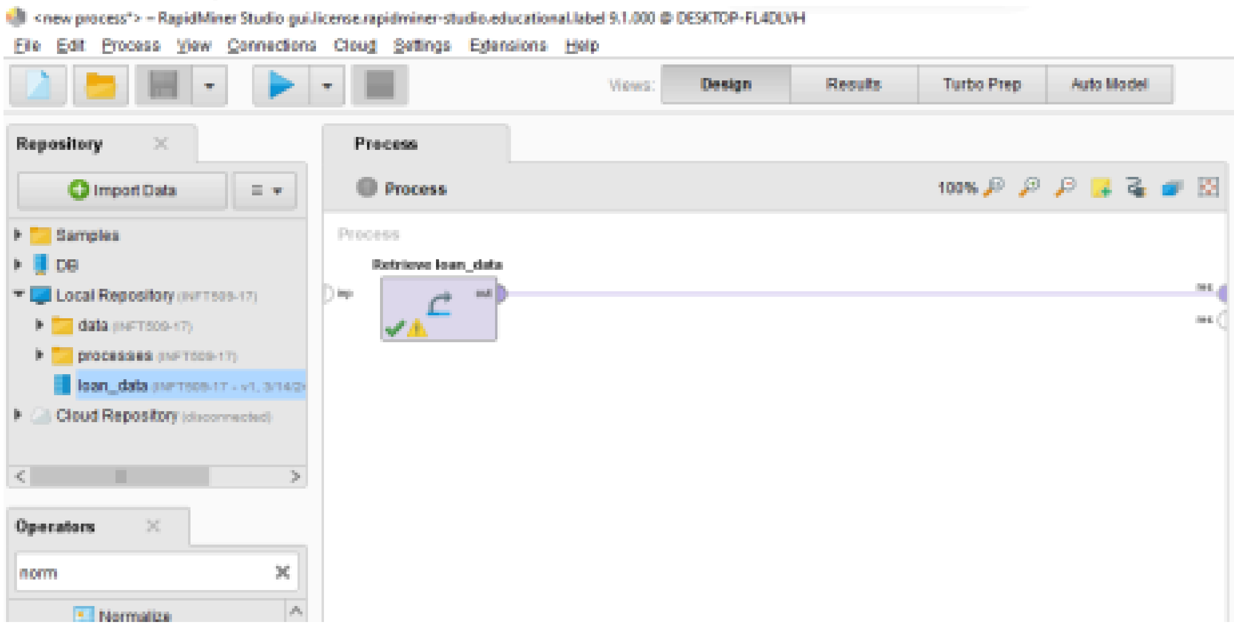
2. Training Phase: During the training phase, the classification model learns from the input data by adjusting its internal parameters based on the provided input-output pairs. Different algorithms, such as decision trees, logistic regression, support vector machines (SVM), k-nearest neighbors (KNN), and neural networks, can be used for training classification models.

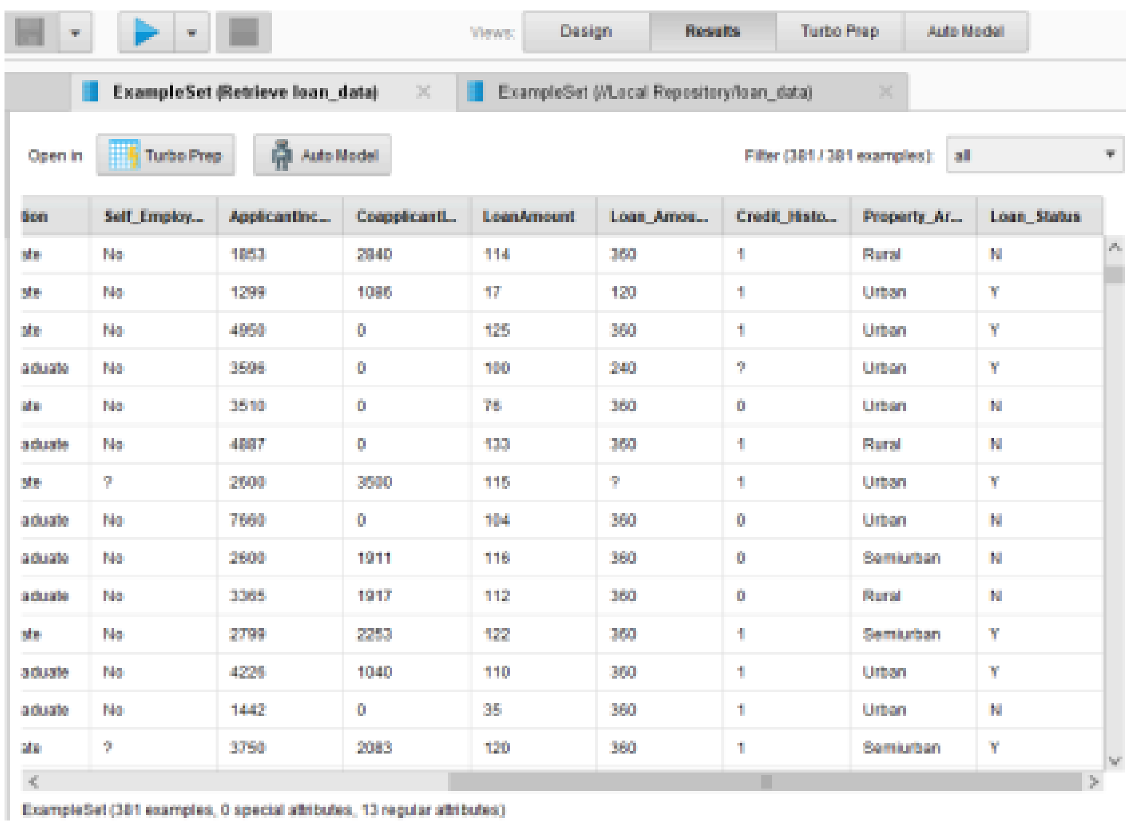
3. Prediction: Once the model is trained, it can be used to predict the class labels of new, unseen data points. The model applies the learned mapping to the input features of the new data and outputs the predicted class or probability distribution over classes.

4. Evaluation: The performance of a classification model is evaluated using metrics such as accuracy, precision, recall, F1-score, and area under the receiver operating characteristic curve (ROC-AUC). These metrics assess how well the model generalizes to unseen data and its ability to correctly classify instances into their respective classes.

Steps :

1. Data Preparation



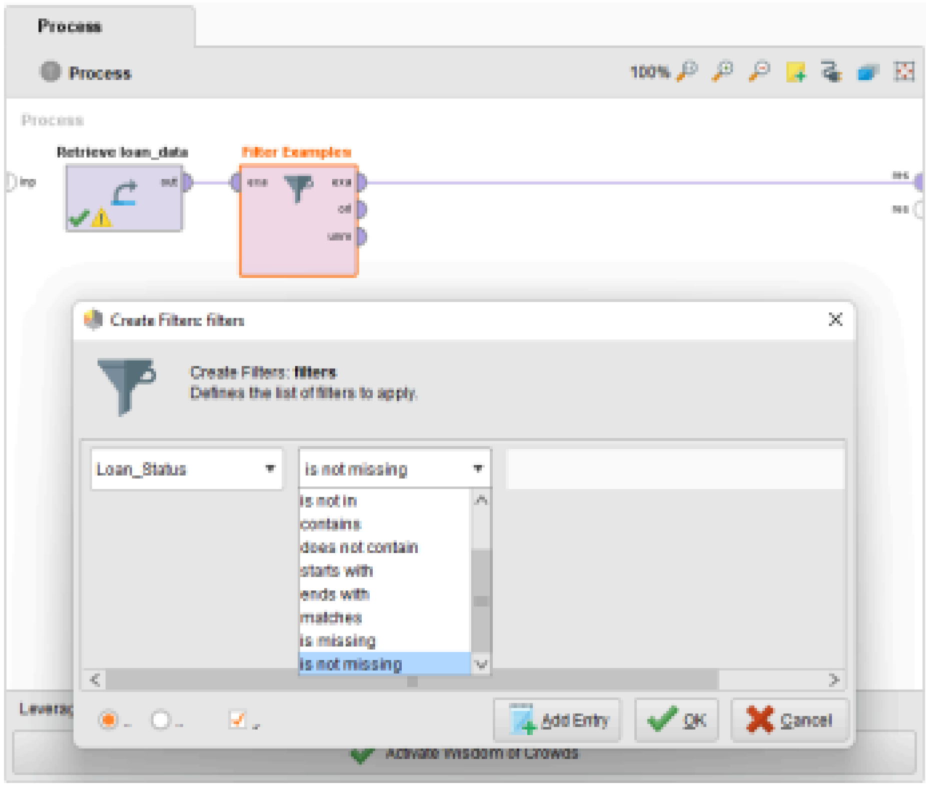


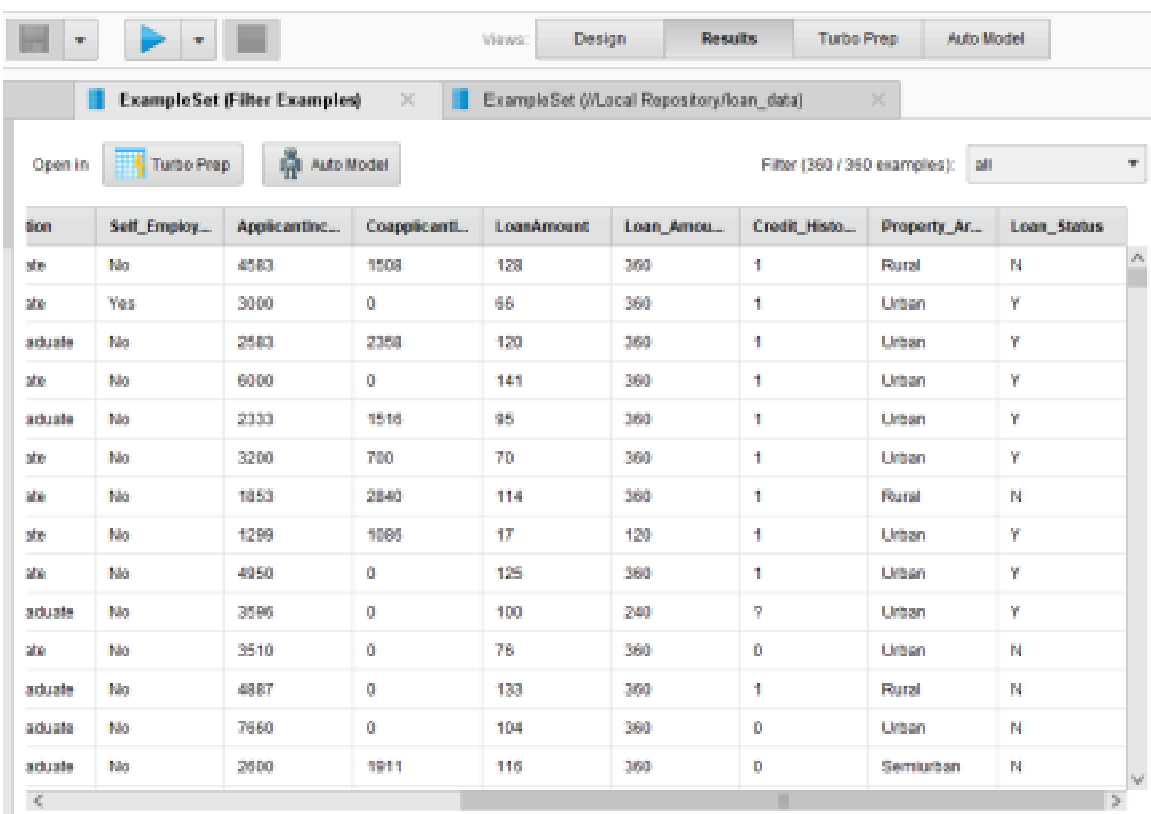
tion	Self_Employ...	ApplicantInc...	CoapplicantI...	LoanAmount	Loan_Amou...	Credit_Histo...	Property_Ar...	Loan_Status
ate	No	1853	2840	114	360	1	Rural	N
ate	No	1299	1086	17	120	1	Urban	Y
ate	No	4950	0	125	360	1	Urban	Y
aduate	No	3596	0	100	240	?	Urban	Y
ate	No	3510	0	76	360	0	Urban	N
aduate	No	4887	0	133	360	1	Rural	N
ate	?	2600	3500	115	?	1	Urban	Y
aduate	No	7660	0	104	360	0	Urban	N
aduate	No	2600	1911	116	360	0	Semiurban	N
aduate	No	3365	1917	112	360	0	Rural	N
ate	No	2799	2253	122	360	1	Semiurban	Y
aduate	No	4226	1040	110	360	1	Urban	Y
aduate	No	1442	0	35	360	1	Urban	N
ate	?	3750	2083	120	360	1	Semiurban	Y

ExampleSet (381 examples, 0 special attributes, 13 regular attributes)

2.

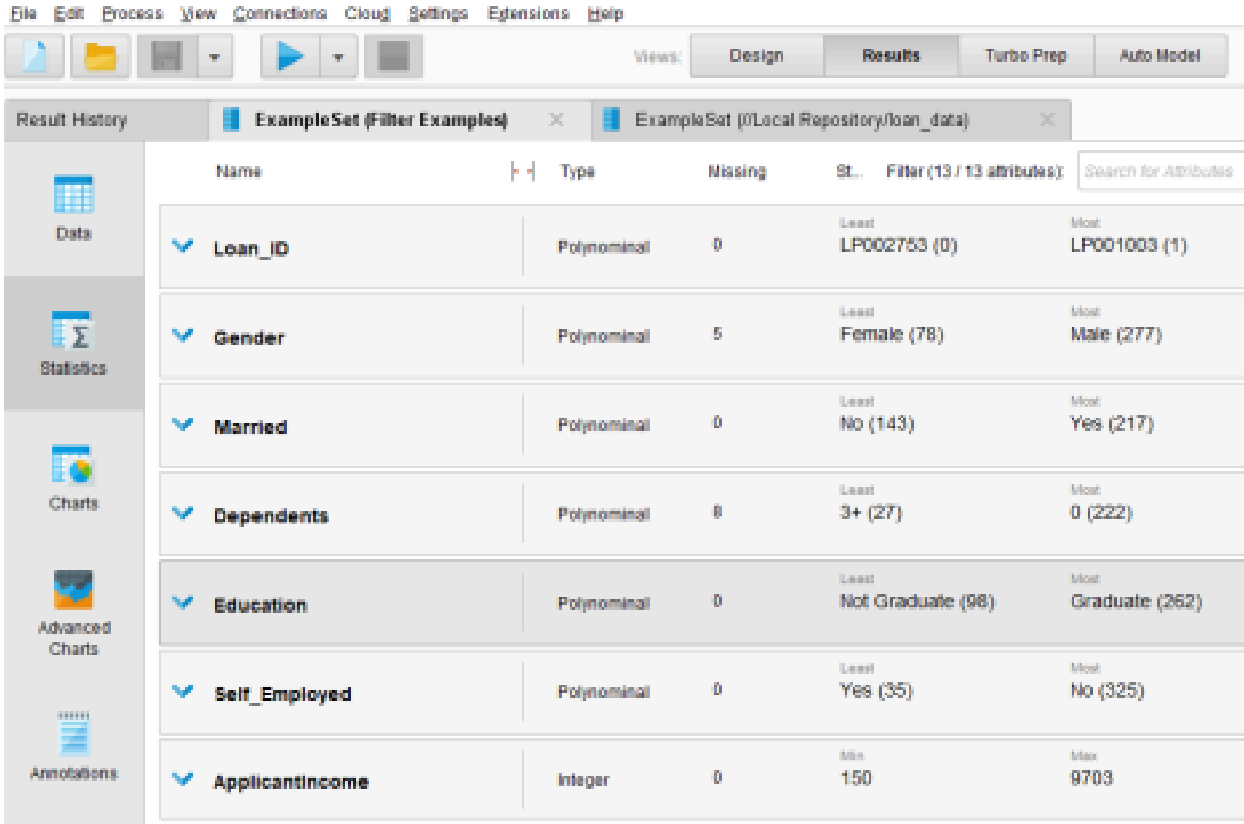
Filter the data to eliminate missing values


Process
Process
100%
Process
Retrieve loan_data
Filter Examples
Create Filters: filters
Create Filters: filters
Defines the list of filters to apply.
Loan_Status
is not missing
is not in
contains
does not contain
starts with
ends with
matches
is missing
is not missing
Add Entry
OK
Cancel
Activate Wisdom of Crowds

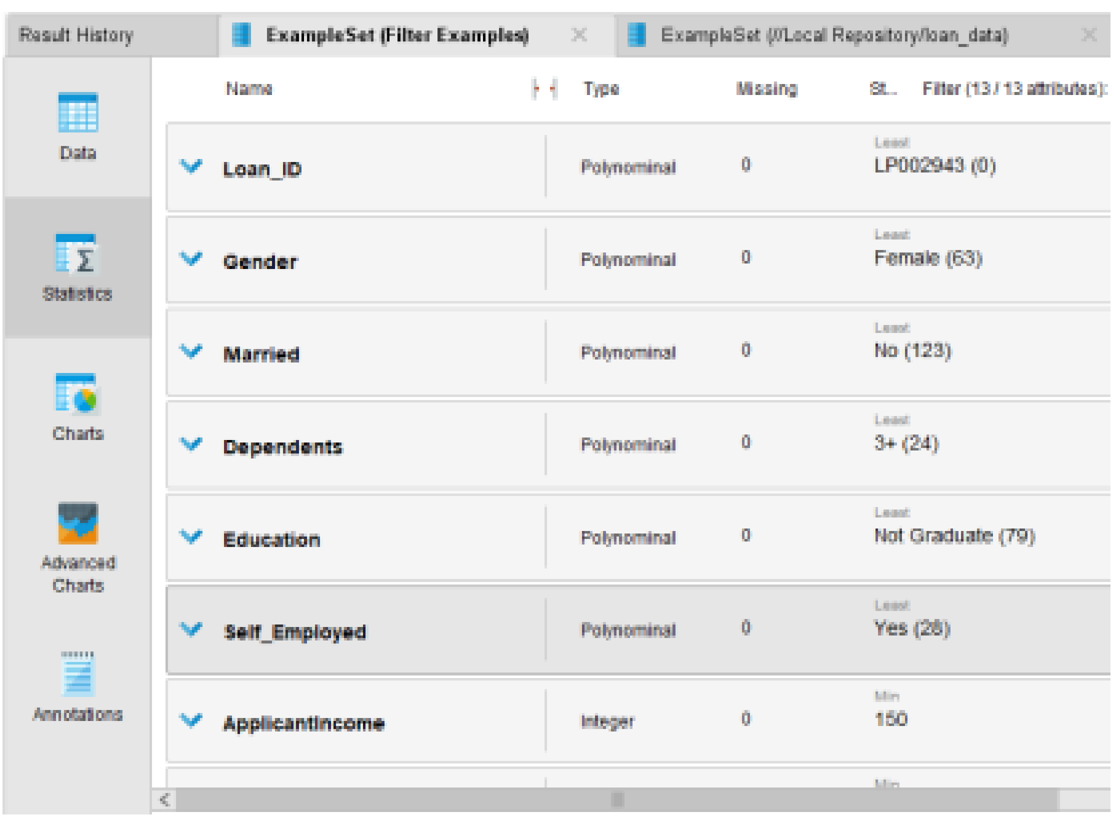

Views:
Design
Results
Turbo Prep
Auto Model
ExampleSet (Filter Examples)
ExampleSet (//Local Repository/loan_data)
Open in
Turbo Prep
Auto Model
Filter (360 / 360 examples):
all
Self_Employ...
ApplicantInc...
CoapplicantI...
LoanAmount
Loan_Amou...
Credit_Histo...
Property_Ar...
Loan_Status

3.

Check the statistics tab in results, and eliminate missing values from entire dataset

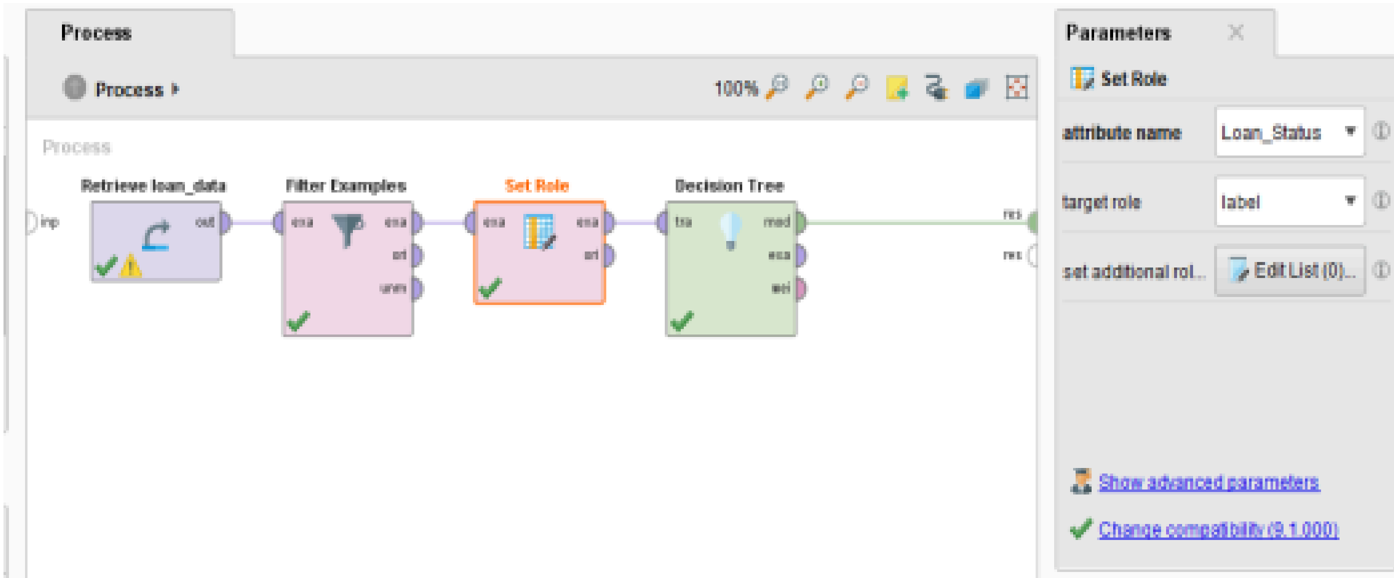


After removing all missing values:

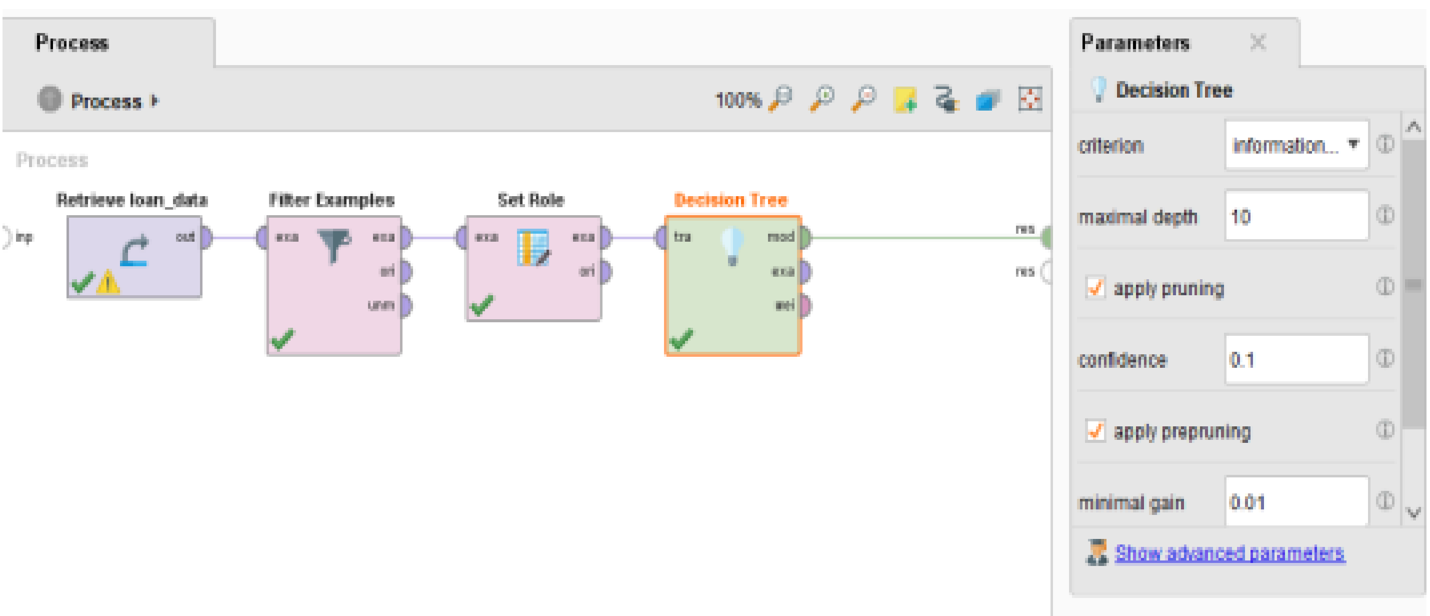


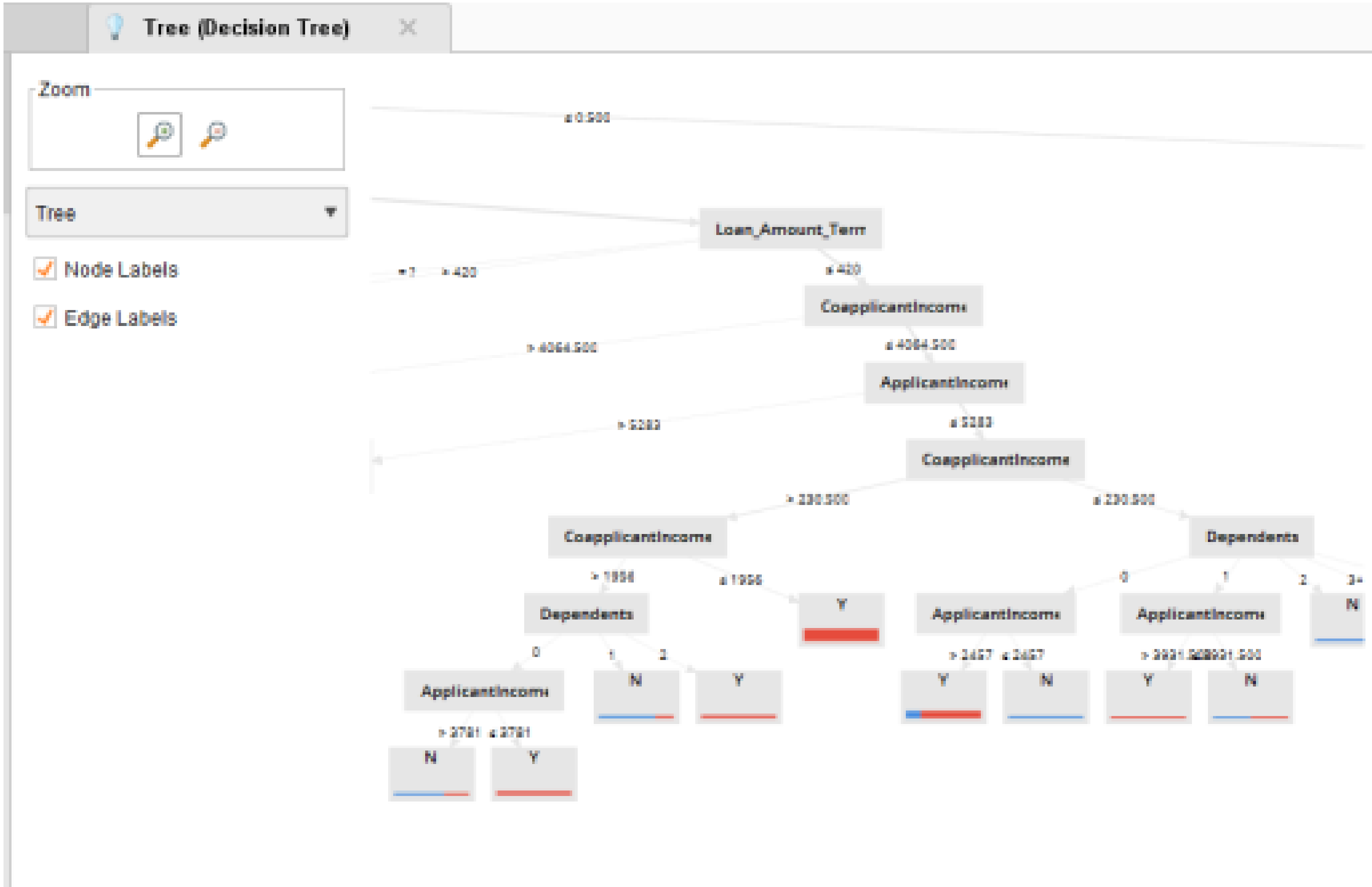
4. Building decision tree :

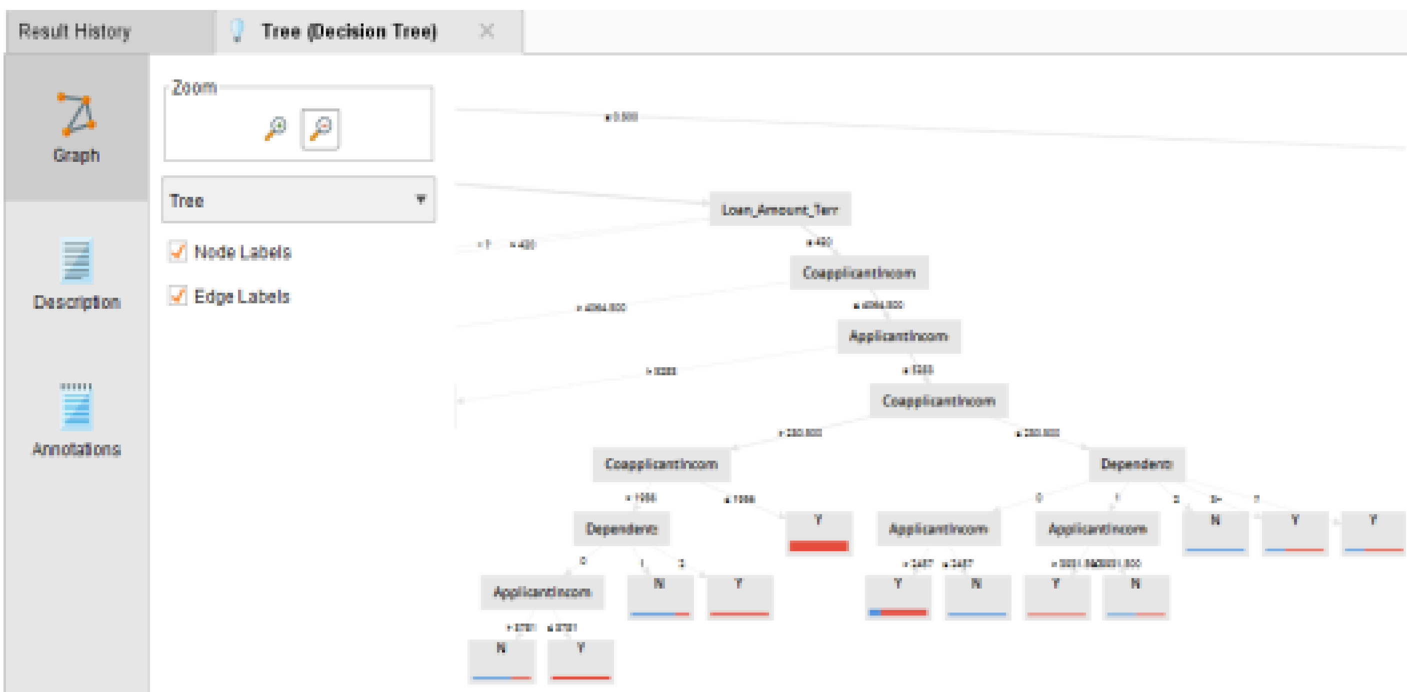
Set label attribute



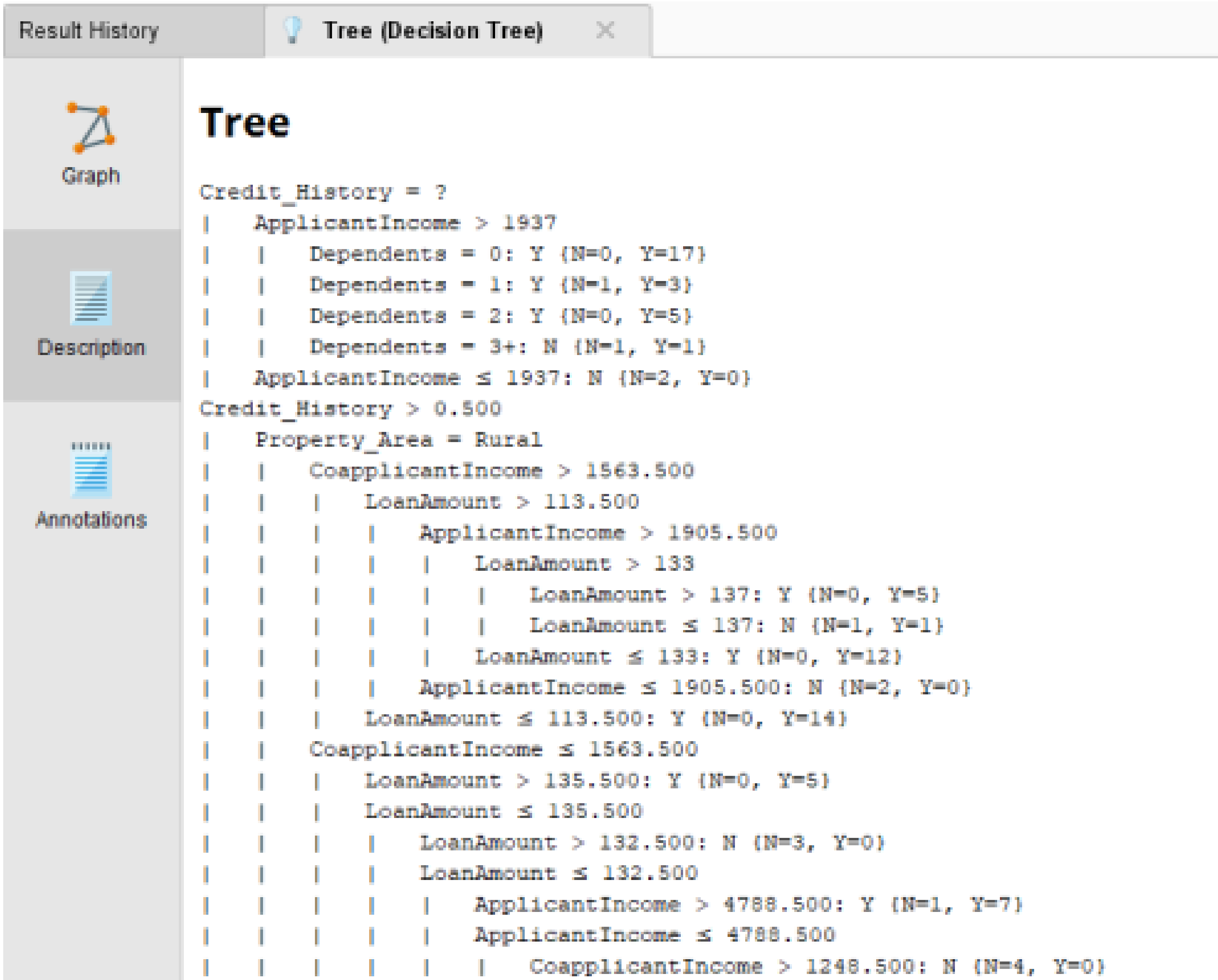
Set criterion (information gain/ gini index) :




Tree (Decision Tree)
Zoom
Tree
Node Labels
Edge Labels
Loan_Amount_Term
CoapplicantIncome
ApplicantIncome
CoapplicantIncome
CoapplicantIncome
Dependents
Dependents
ApplicantIncome
ApplicantIncome
ApplicantIncome
Y
N


Result History
Tree (Decision Tree)
Graph
Description
Annotations
Zoom
Tree
Node Labels
Edge Labels
Loan_Amount_Term
Y
N

Numeric description of the decision tree



Conclusion : Hence we have understood and implemented a classification model using Rapid Miner software and performed data pre processing , data cleaning, and built the decision tree.