
Department of Computer Science

CPCS331: Artificial Intelligence I

Fall 2022

**induvial project**

# Applying Machine Learning algorithms on Bank Marketing Dataset

**Anfal sultan alshehri**

**Kawther kamal**

**Section: B2A**

**Instructeur: Dr. Hajer**

**Date:11/6/2022**

# Contents

## Task Assignment

| Name | Task |
| --- | --- |
| Anfal sultan | **Support vector machine** |
| Kawther kamal | **Random forest** |

All team members work together except in the technical part.

## Problem Explanation

We will use a dataset related to marketing campaigns (phone calls) of a Portuguese banking institution. The classification goal is to predict if the client will subscribe to a term deposit (variable y). We will use a different type of model and see which model gives the highest accuracy.

## Purpose of the project:

The purpose of this project is to apply the machine learning algorithm using two software on the chosen Dataset and calculate its accuracy. The goal is to explore how different validations perform with the same algorithm on the same data to produce different outcomes.

## Outline the Approach:

1. Chose the Dataset (Bank Marketing), ML algorithm to classify.
2. Use Software (Weka, RapidMiner) for testing and Visuals of the result.
3. Compare the accuracy between the two software (Weka, RapidMiner).

## Technical Description

### Selected Dataset

Bank Marketing Dataset is the selected Dataset in this project. It has 45211 instances and 17 attributes. This Dataset is created to predict if the client of the bank will subscribe to a term deposit or not (y).

### ML algorithms

In this experiment, the algorithm that will be used is the Support Vector Machine. This algorithm is used for Classification and same times for Regression problems. The algorithm classifies data using Hyperplane. A Hyperplane is a decision boundary that differentiates the two classes in SVM.

The SVM(Support Vector Machine) algorithm has to type :

1- Linear SVM: for the binary class classifier.
2- Non-Linear SVM: multiple class classifier.

One of the applications of the SVM algorithm are **Face detection, image classification, text categorization**

Linear SVM will be used in this experiment since we have two classes (the client will subscribe to a term, and the client will not subscribe to a term).

## Result

### Experiment Result

| Percentage | Weka Accuracy | RapidMiner Accuracy |
|---|---|---|
| 66% Data, 34% Test | 89.9029% | 90.36% |
| 77% Data, 23% Test | 88.3036% | 90.41% |
| 88% Data, 12% Test | 90.2893% | 90.28% |
| Cross Validation, 10 folds | 80.7395% | 89.99% |

*Table 1: Result Table*

### Analyze Result

Form Experiment Result Table, RapidMiner, and Weka have close accuracy percentages between 91%-80%. However, the RapidMiner tool has higher accuracy than Weka in all validations. The average of the result is 88%.

## Conclusion

After we finished our experiment we measured the difference between the results from each algorithm. In conclusion, The Random Forest Algorithm has the highest accuracy than the Support Vector Machine. Moreover, it's more stable than the SVM algorithm. Also has a greater average than the SVM. In this experiment, we find out that the Random Forest is better than the SVM algorithm because the average accuracy is higher than the SVM algorithm.

## References

*Bank Marketing Data Set*. UCI Machine Learning Repository . (n.d.). Retrieved November 5, 2022, from https://archive.ics.uci.edu/ml/datasets/bank+marketing

GmbH, R. (n.d.). *Support vector machine*. RapidMiner Documentation. https://docs.rapidminer.com/latest/studio/operators/modeling/predictive/support_vector_machines/support_vector_machine.html

Sugiyama, M. (2016). Support vector classification. *Introduction to Statistical Machine Learning*, 303-320. https://doi.org/10.1016/b978-0-12-802121-7.00038-8

# Appendix

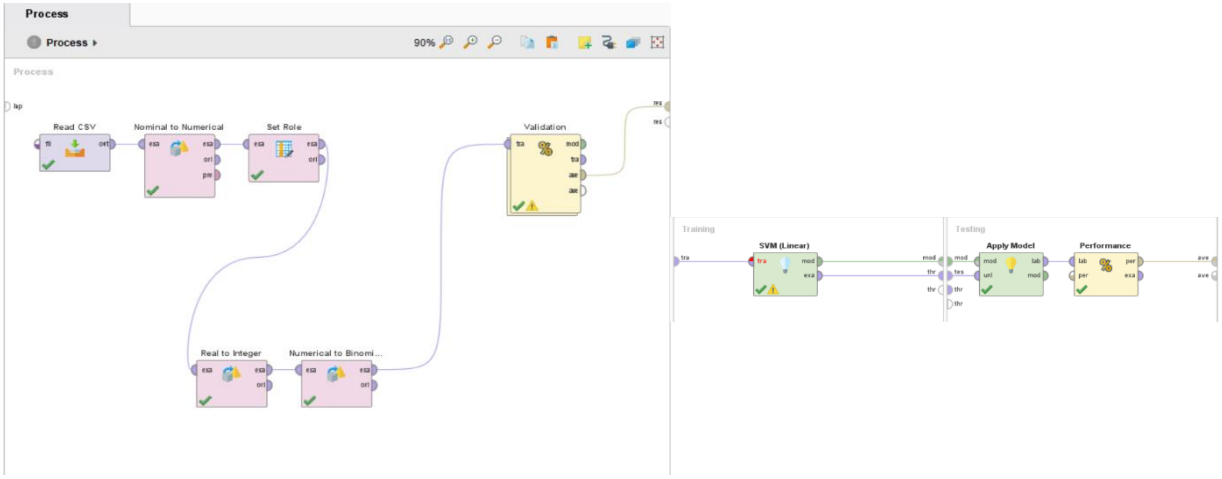


*Figure : RapidMiner Split Validation*

**accuracy: 90.36%**

| | true false | true true | class precision |
|---|---|---|---|
| pred. false | 14319 | 1289 | 91.74% |
| pred. true | 300 | 567 | 65.40% |
| class recall | 97.95% | 30.55% | |

*Figure :RapidMiner Split Validation 66%*

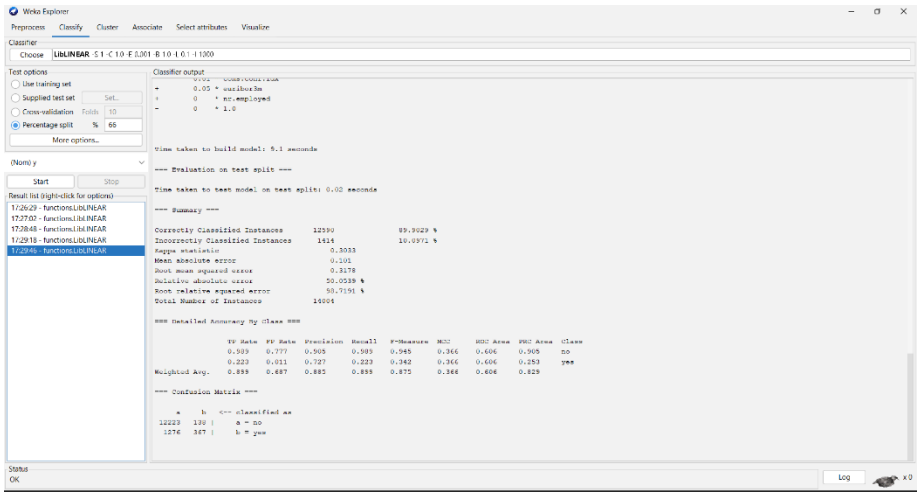


*Figure :Weka with split 66%*

**accuracy: 90.41%**

| | true false | true true | class precision |
|---|---|---|---|
| pred. false | 10748 | 969 | 91.73% |
| pred. true | 216 | 423 | 66.20% |
| class recall | 98.03% | 30.39% | |

*Figure :RapidMiner with spilt 77%*

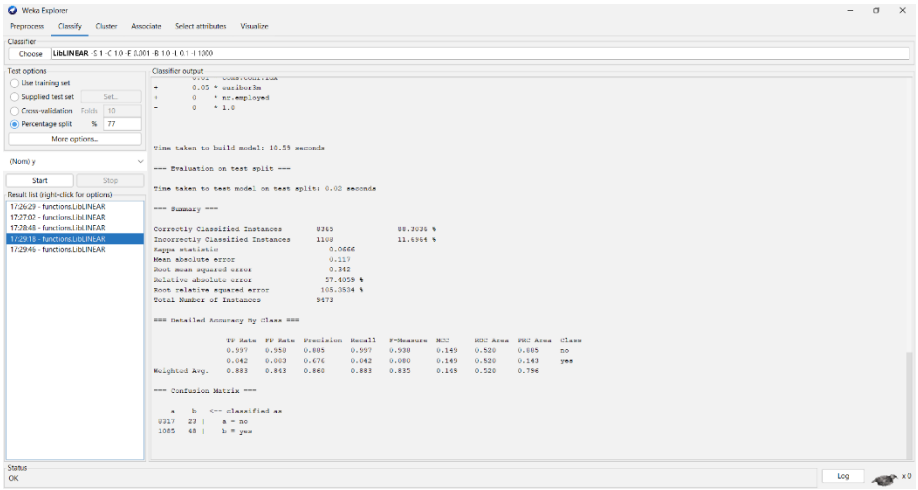


*Figure :Weka with split 77%*

accuracy: 90.28%

| | true false | true true | class precision |
|---|---|---|---|
| pred. false | 7163 | 654 | 91.63% |
| pred. true | 147 | 274 | 65.08% |
| class recall | 97.99% | 29.53% | |

*Figure :RapidMiner with Split 88%*

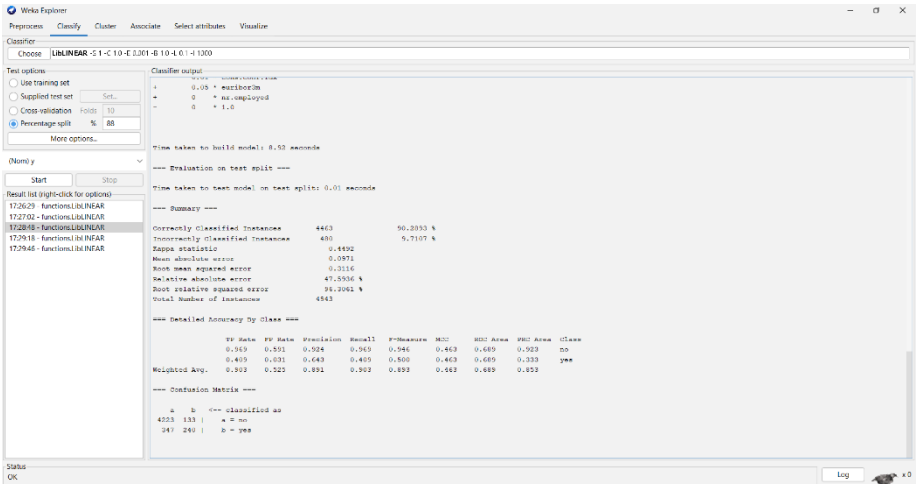


*Figure : Weka With Split 88%*

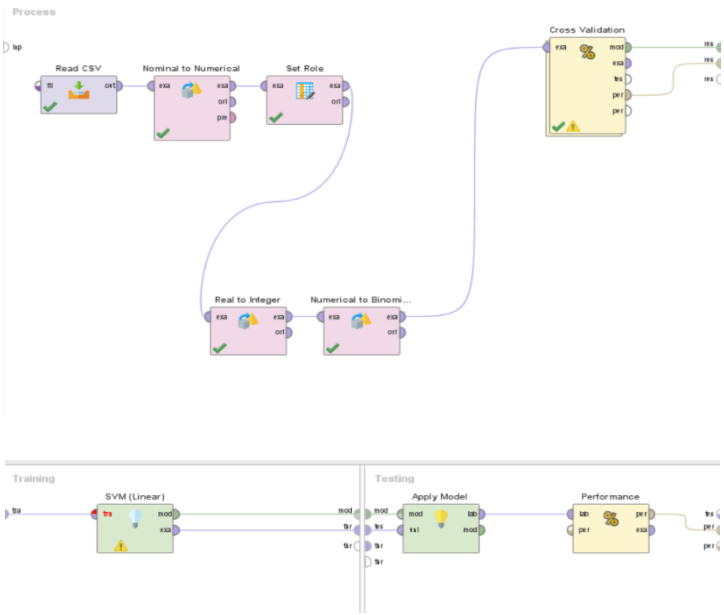


*Figure :RapidMiner Cross validation*

**accuracy: 89.99% +/- 0.45% (micro average: 89.99%)**

| | true false | true true | class precision |
|---|---|---|---|
| pred. false | 35878 | 3453 | 91.22% |
| pred. true | 670 | 1187 | 63.92% |
| class recall | 98.17% | 25.58% | |

*Figure :RapidMiner With cross Validation 10 folds*

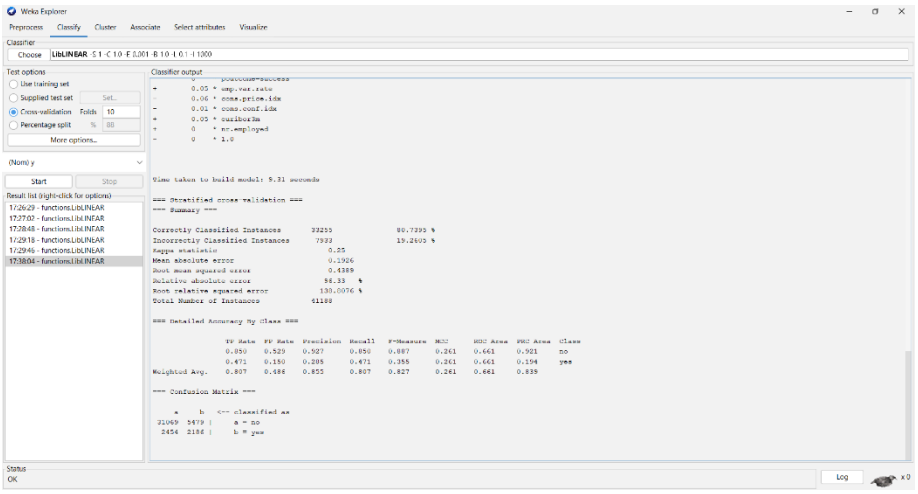


*Figure :Weka with Cross Validation 10 folds*