Contents

[Introduction 2](#_Toc57462551)

[PART A 2](#_Toc57462552)

[Logistic Elastic-Net Regression 2](#_Toc57462553)

[Classification Tree 2](#_Toc57462554)

[PART B 3](#_Toc57462555)

[Optimized parameters for elastic-net 3](#_Toc57462556)

[Classifier Tree 3](#_Toc57462557)

[Confusion Matrix on Spitted Training Set 3](#_Toc57462558)

[PART C 4](#_Toc57462559)

[Confusion Matrix on real dataset 4](#_Toc57462560)

[Comparison 4](#_Toc57462561)

[Recommendation 5](#_Toc57462562)

[References 6](#_Toc57462563)

# Introduction

In current information technology period Data Analysis and visualization is a growing and on-demand field. Not only IT but also fields like banking, marketing is significantly utilizing data analysis for their operations. In cyber security, the fusion is emerging, many applications such as phishing detection, malware detection are coming to market.

This report consists the analysis of given algorithms to detect the spam emails. The algorithms that were allocated here by the program are logistic elastic net regression, and classifier tree. After a brief about the given algorithms this report goes through the training stage, then briefs about the tuning, after that explains the testing of trained models against the real-world test dataset. Then it briefs on the results comparison with confusion matrix, and concluding with suggested model to deploy.

# PART A

## Logistic Elastic-Net Regression

Logistic elastic-Net is a classification algorithm that combines the features of both Ridge, and Lasso Regression (Dhruve, 2020). The primary advantage of Elastic-net is due to its hybrid (CFI, n.d.) nature it overcomes the limitations of both the ridge and lasso by using their L1 and L2 regularization. To overcome the limitations of Lasso, this elastic includes a quadratic expression. In elastic net two steps are involved to find out the estimator. The first step is to find out the ridge expression co-efficient, and secondly using the Lasso’s shrinkage of the co-efficient. The primary advantage of elastic-net is, it performs both variable selections, and then regularization at the same time.

## Classification Tree

It is also known as a Decision Tree. These are multipurpose algorithms that are capable to perform both classification and regression operations. These are widely used in complex datasets. Also, classification trees are the base of random forest algorithms. The primary advantage (Dhiraj, 2019) of the classification tree is it does not require data scaling or normalization. Also, missing data does not affect the process.

# PART B

## Optimized parameters for elastic-net

The fitting for elastic-net has been done using glmnet (Trevor, n.d.) library which uses penalized maximum likelihood to fit a generalized linear model. Hyperparameters to tune were found using

the function best tunes with the combination of Grid search. The model was trained with repeatedCV with 10 folds and 5 repeats, the net models (𝜆, 𝛼) values for this model are given below,

|  |  |  |
| --- | --- | --- |
| Alpha | Lambda | Optimized Hyperparameters |
| 0.1 | 0.00231013 | 7 |

## Classifier Tree

The classifier tree was trained using the rpart library, with 10-fold cross, and the validation repeated for 3 times. And to tune the CP parameter in Rpart 15 values has been used. The highest accuracy recorded was 0.7976487 (which gives 80% when rounds).

## Confusion Matrix on Spitted Training Set

|  |  |
| --- | --- |
| **Elastic regression** | **Classifier Tree** |
| Confusion Matrix and Statistics  Yes No  Yes 671 131  No 344 853  Accuracy : 0.7624  95% CI : (0.7431, 0.7809)  No Information Rate : 0.5078  P-Value [Acc > NIR] : < 2.2e-16    Kappa : 0.5262    Mcnemar's Test P-Value : < 2.2e-16    Sensitivity : 0.6611  Specificity : 0.8669  Pos Pred Value : 0.8367  Neg Pred Value : 0.7126  Prevalence : 0.5078  Detection Rate : 0.3357  Detection Prevalence : 0.4012  Balanced Accuracy : 0.7640    'Positive' Class : Yes | Confusion Matrix and Statistics  Yes No  Yes 762 130  No 253 854  Accuracy : 0.8084  95% CI : (0.7905, 0.8254)  No Information Rate : 0.5078  P-Value [Acc > NIR] : < 2.2e-16    Kappa : 0.6174    Mcnemar's Test P-Value : 4.549e-10    Sensitivity : 0.7507  Specificity : 0.8679  Pos Pred Value : 0.8543  Neg Pred Value : 0.7715  Prevalence : 0.5078  Detection Rate : 0.3812  Detection Prevalence : 0.4462  Balanced Accuracy : 0.8093    'Positive' Class : Yes |

# PART C

## Confusion Matrix on real dataset

|  |  |
| --- | --- |
| **Elastic Regression** | **Classifier Tree** |
| Confusion Matrix and Statistics  Yes No  Yes 5841 3379  No 1591 39189  Accuracy : 0.9006  95% CI : (0.8979, 0.9032)  No Information Rate : 0.8514  P-Value [Acc > NIR] : < 2.2e-16  Kappa : 0.6427  Mcnemar's Test P-Value : < 2.2e-16    Sensitivity : 0.7859  Specificity : 0.9206  Pos Pred Value : 0.6335  Neg Pred Value : 0.9610  Prevalence : 0.1486  Detection Rate : 0.1168  Detection Prevalence : 0.1844  Balanced Accuracy : 0.8533  'Positive' Class : Yes | Confusion Matrix and Statistics  Yes No  Yes 6305 251  No 1127 42317  Accuracy : 0.9724  95% CI : (0.971, 0.9739)  No Information Rate : 0.8514  P-Value [Acc > NIR] : < 2.2e-16  Kappa : 0.8855  Mcnemar's Test P-Value : < 2.2e-16    Sensitivity : 0.8484  Specificity : 0.9941  Pos Pred Value : 0.9617  Neg Pred Value : 0.9741  Prevalence : 0.1486  Detection Rate : 0.1261  Detection Prevalence : 0.1311  Balanced Accuracy : 0.9212  'Positive' Class : Yes |

## Comparison

When comes to the data analysis most used parameters that define the value of the result are accuracy, specificity, and sensitivity (Irizarry, 2019). Accuracy is defining the ratio of accurately labeled. Sensitivity is generally the true positive (or recall) (I.e identifying the actual positive rate that the model detected as positive). The more the sensitivity is higher it shows that the model is good at determining the positive values accurately. Specificity defines the True Negative (I.e. Identifying the exactly identifying the negative values as negative) the more the specificity is higher it shows that the model is good at detecting the negative values accurately.

In this case, all the three primary factors indicating that the classifier tree is providing better performance than the elastic-net.

## Recommendation

The final model that I would deploy would be the classifier tree. There are 4 reasons. Firstly, the working mechanism of the classifier tree with CP parameter. A classifier tree does not require balanced, and/or pre-processed data. So even if some data is missing the classifier tree will not get any downgrades in terms of performance and efficiency. Secondly, the results that I have received during the test clearly showing the three main factors (accuracy, sensitivity, and specificity) that define the quality of a model is better than comparing with elastic-net.

Also, in general, during the decision process, the classifier tree divides the space into multiple smaller places which provides a wider perspective, whereas elastic-regression separates the space by a single line. When considering the nature application (email spam detection in this case) the volume of data would be higher, so using the classifier tree make the division better and increase the accuracy. Fourthly, easy to use for this specific email spam application, classifier tree requires simple math and doesn’t need specific complex math algorithms, so when the email spam detection application encounters any programming logic errors in mid of operations troubleshooting becomes easier which is a good thing for business continuity.

# References

CFI. Elastic Net - Overview, Geometry, and Regularization. Retrieved 25 November 2020, from https://corporatefinanceinstitute.com/resources/knowledge/other/elastic-net/

Dhiraj, K. (2019). Top 5 advantages and disadvantages of Decision Tree Algorithm [Blog]. Retrieved from https://dhirajkumarblog.medium.com/top-5-advantages-and-disadvantages-of-decision-tree-algorithm-428ebd199d9a

Dhruve. (2020). Elastic Net Regression in R Programming - GeeksforGeeks. Retrieved 25 November 2020, from https://www.geeksforgeeks.org/elastic-net-regression-in-r-programming/

Trevor, H. glmnet function | R Documentation. Retrieved 25 November 2020, from https://www.rdocumentation.org/packages/glmnet/versions/4.0-2/topics/glmnet

Irizarry, R. (2019). Introduction to data science (1st ed., p. Chapter 27). Chapman and Hall/CRC; 1st edition.

Le, J (2018) Decision Trees in R. Retrieved from, https://www.datacamp.com/community/tutorials/decision-trees-R

Oleszak.M (2019) Regularization: Ridge, Lasso and Elastic Net. Retrieved from

https://www.datacamp.com/community/tutorials/tutorial-ridge-lasso-elastic-net