ITESO, Universidad Jesuita de Guadalajara

Credit Models

Classification Model Report

Spring 2024

Emiliano Mena - 728407
Jorge Hernández - 730342

Index

1.Introduction

1.1 Project

The main objective of this project is to create a credit score model, the model receives a dataset and classifies every record on one of three possible scores (Poor, Standard and Good). This model is going to be a supervised machine learning classifier and to select the "best" model they will be compared different classifiers and finally selected the one that has a best accuracy.

1.2 Data

The data is divided on a train and test sample, the train will be subdivided on train and validation in order to measure the accuracy that the model has and the test is where the real predictions will be made. The train will have 75,000 records, the validation 25,000 and the test 50,000. The data includes 27 variables.

1.3 Questions of the project

In order to achieve the final model some of the questions and decisions that are needed to be take are:

- What variables are going to be used and why?
- What models are going to be compared and why?
- What can we conclude of the model?

2. Data Analysis

2.1 The data

The data we've initially received was urged to be processed and cleaned in function to be useful for the assignment we have done that is guessing the future also called as predict, what we have done principally is purifying the columns and rows, for example: if a column is giving us numerical information, we must guarantee that in that column there will be only numbers.

In our specific goal, that is have useful data of credit highlights from people (because is going to be introduced in a classifier model, to predict credit score), so we must pay special attention exploring the data; so we can be able to get a precise model and predictions.

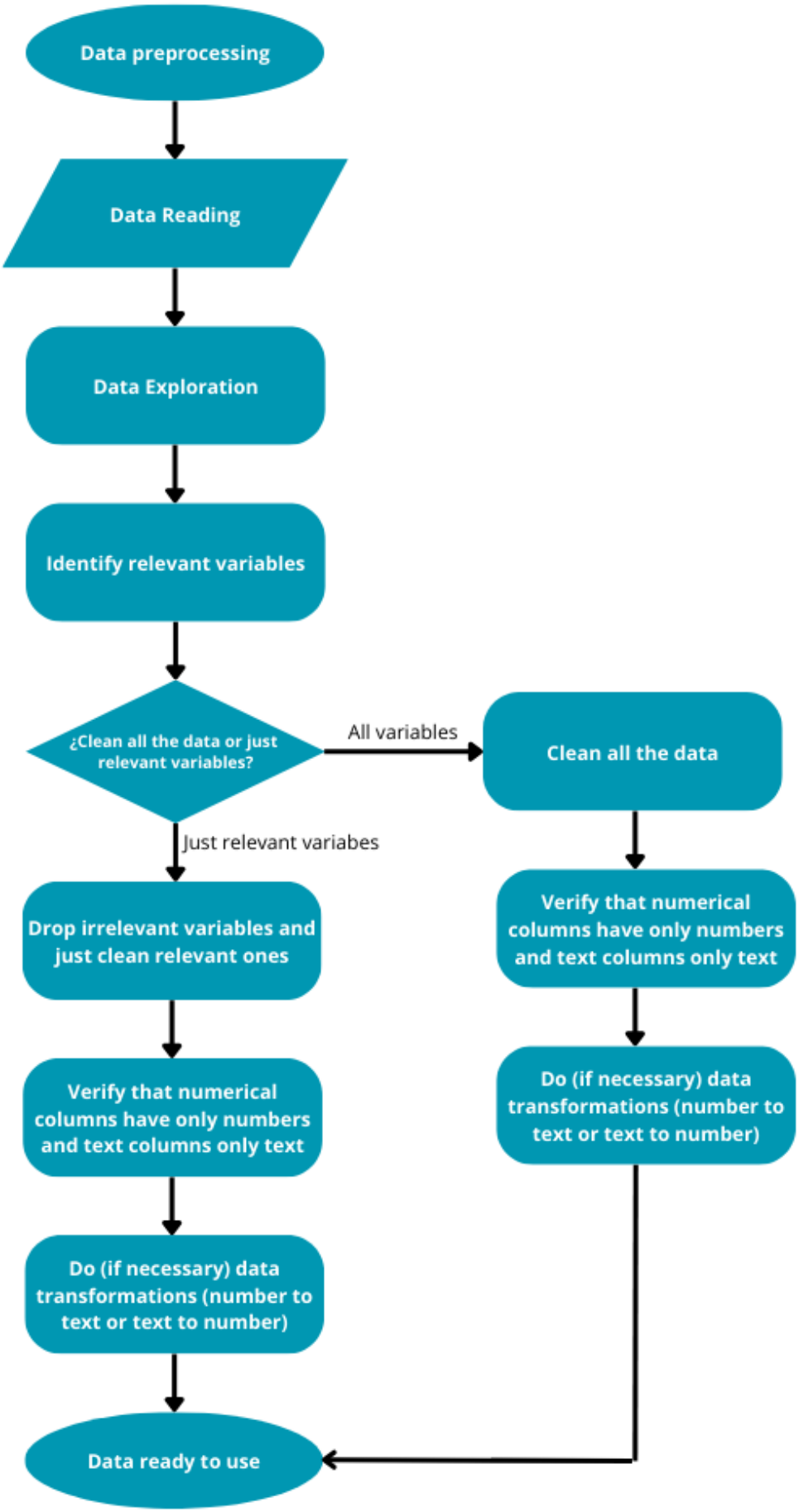

Figure 1. data preproccesing

2.1.2 Chosen features

- **Delay from due date :** indicates the elapsed time (usually days) of non-payment after the due date.
- **Number of delayed payments :** indicates how many payments the customer did, after the due date of payment
- **Credit history age :** indicates the time of your credit age, for how many time you've been holding a credit (usually given in years)
- **Number of credit inquires :** indicates how many times an entity have requested your credit history (this normally happens when you go to an entity and want to take out a loan or credit card)
- **Credit Mix :** indicates the different type of loans or credits you have contracted (mortgage, car loan, educative loan)
- **Outstanding debt :** indicates te quantity (usually monetarily) that a person or entity has not payed to another person or entity.
- **Credit utilization ratio :** indicates how much (usually indicated in percentage) of your credit line you are using. (if your credit line is 10k and you have spent 3k, your credit utilization ratio will be 30%)
- **Monthly in hand salary :** the amount each customer receives per month on his job.

These variables were selected taking as main reference the FICO score, because it is the most popular and studied credit score, taking what is already functional and adding it our own details, is a way to look for an improvement on this classifier; and basically these variables are going to be our model features.

2.1.2 Chosen features

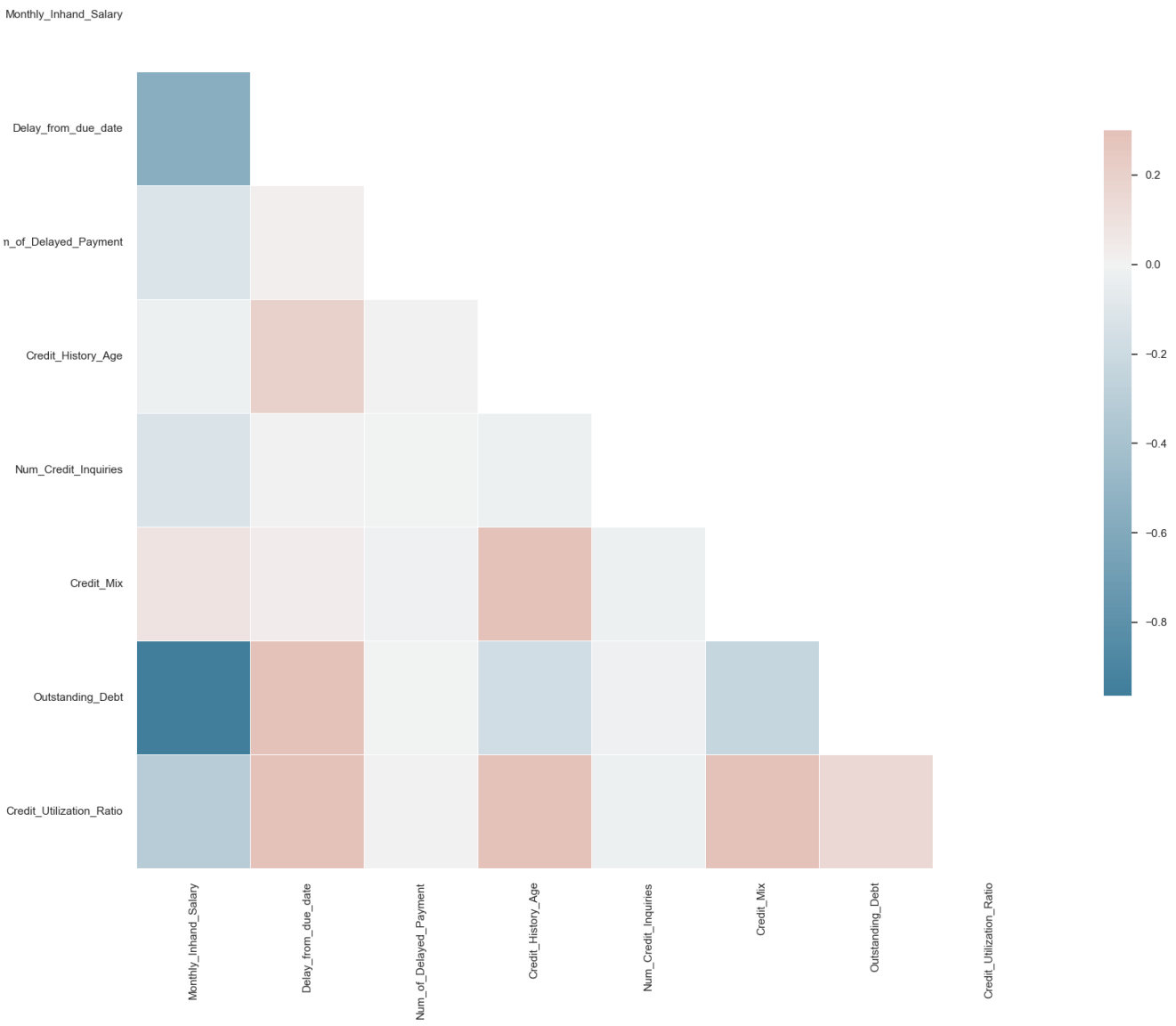

Figure 2. features correlation heatmap

We can observe that this is a correlation matrix, that includes all of our chosen relevant variables, here with colors of a heatmap and squares is represented the relation or dependence between each variable.

Red color indicates high relation (strong red = high relation) and blue color indicates poor relation (strong blue = poor relation), so, in our specific case we can se a majority of white squares (neutral relation), then red squares, and fewer blue squares; summary, we can say that our variables are in some way really related, could be possibly caused by natural relation between them.

Model outputs

Is important to remember that the output (Credit_Score) of the model is multiclass:

- Poor: The customer is not likely to receive credits
- Standard: The customer can receive new credits.
- Good: The customer will receive option of new and better credits.

Time to apply the Machine Learning

Now that we have already selected the variables that will be part of our final model, we need to start the search for the best model. In this part we are going to try 5 diferent models and stay with the one that has the best accuracy.

2.2 Methods

-KNeighbors Classifier:

It is an algorithm used in Machine Learning for regression and classification, in the case of the classification the labels are assigned on the class with more frecuency on the train and then starts to fill the other classes approaching to each one.

-Gradient Boosting:

This classifier is an ensemble method and the way it works looks like this: combines the predictions of multiple weak learners to create a single, more accurate strong learner.

-Random Forest:

This algorithm is one of the different ensemble methods that exist for supervised learning . What it does is to combine the output of multiple Decision Trees (method that splits the data in leafs to clasify the data) to finally reach a final result.

2.2 Methods

-Extremely Randomized Trees Classifier:

The main difference between this method and random forest occurs as (Extra Trees) execute more random splits on how data is computed, basically in RF the threshold is selected in order to the best split between the features, while in ERTC look to randomness splits instead of best ones.

-Stacking Classifier:

This ensemble method allows to combine any other method. The form it works is that it takes the outputs of every method it has and at the end it goes thru the final method selected and gives the final result. It is really powerfull and it improves the accuracy that the other models can do on their own.

2.3 Analysis

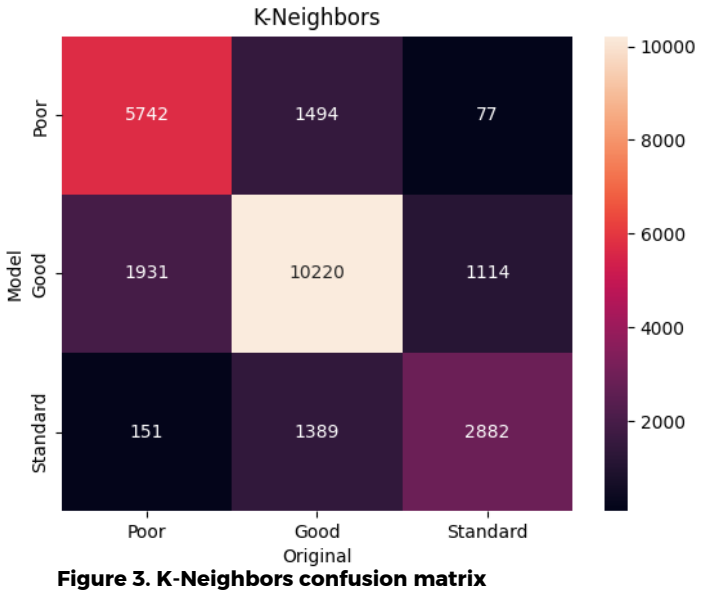

Figure 3. K-Neighbors confusion matrix

Method: K-neighbors

Accuracy: 75.38%

Cross Validation: 61.29%

Analysis: Is the model with the lowest metrics.

Conclusion: It is a fast model but it doesn´t give us the accuracy we are looking.

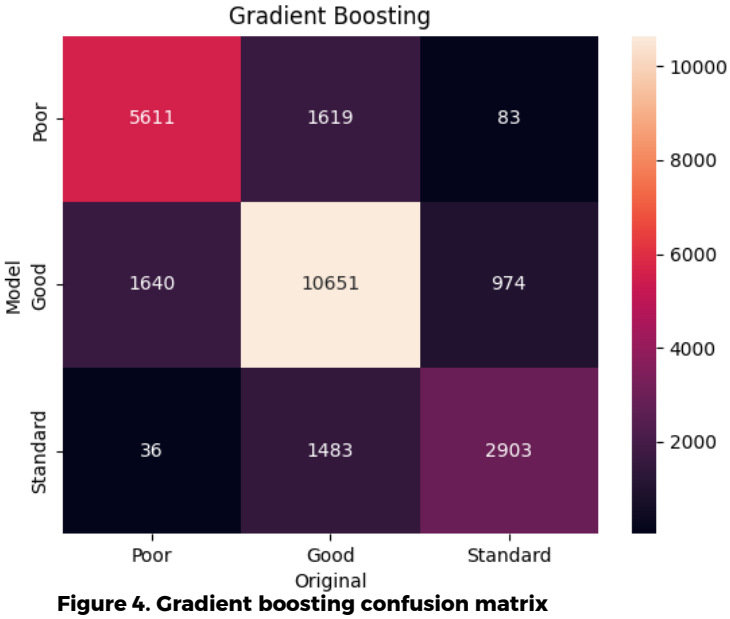

Figure 4. Gradient boosting confusion matrix

Method: Gradient Boosting

Accuracy: 76.66%

Cross Validation: 66.71%

Analysis: The cross validation is a lower than the accuracy.

Conclusion: From the not ensemble models is the best one but is not that good to be the final model.

2.3 Analysis

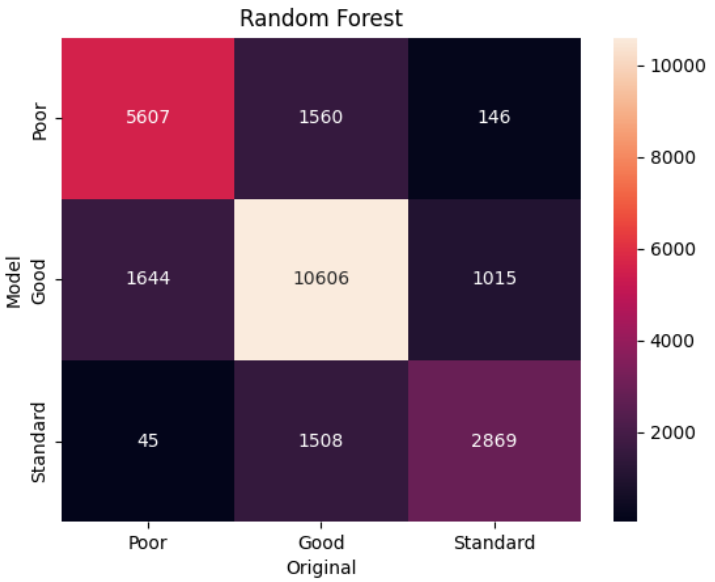

Figure 5. Random Forest confusion matrix

Method: Random forest

Accuracy: 76.33%

Cross Validation: 70.03%

Analysis: From the ensemle models is the one with the lowest metrics.

Conclusion: It has a cross validation that gets closer to the accuracy but is not going to be the final model.

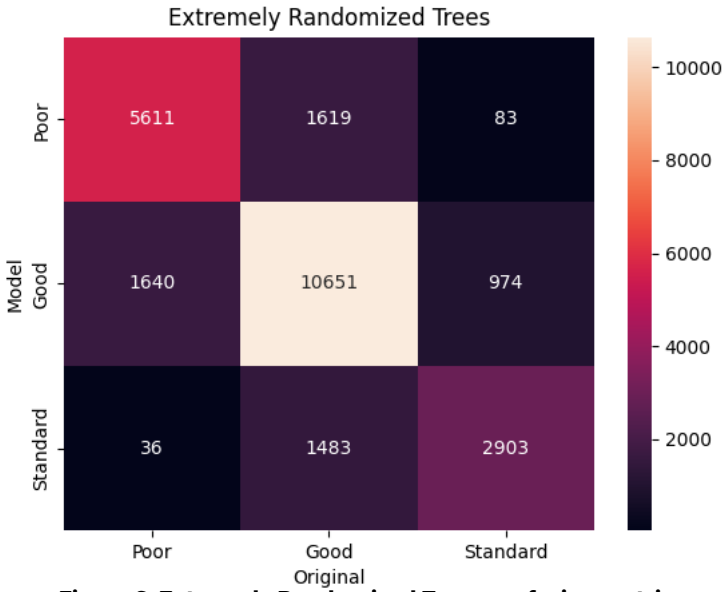

Figure 6. Extremely Randomized Trees confusion matrix

Method: Extremely Randomized Trees

Accuracy: 76.66%

Cross Validation: 70.29%

Analysis: Almost the same metrics of the Random Forest.

Conclusion: Is not the accuracy we are looking for.

2.3 Analysis

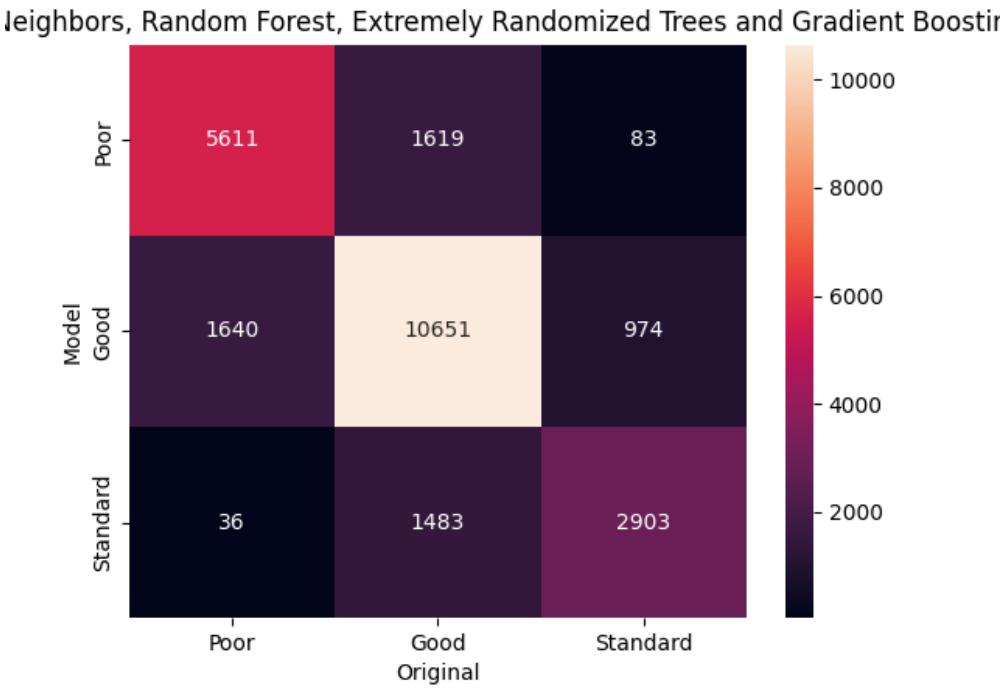

Figure 7. Stacking classifier confusion matrix

Method: Stacking Classifier

Components: K-Neighbors, Gradient Boosting, Random Forest, Extremely Randomized Trees

Accuracy: 78.66%

Cross Validation: 70.88%

Analysis: The cross validation is almost the same of the other ensemble models, but the accuracy is better and is near the 79%. From all the models we tried is the best one.

Conclusion: The Stack model is going to be the choosed as the final model.

3. Results

	Name	Credit_Score
0	Aaron Maashoh	Good
1	Aaron Maashoh	Good
2	Aaron Maashoh	Good
3	Aaron Maashoh	Good
4	Rick Rothackerj	Good
5	Rick Rothackerj	Good
6	Rick Rothackerj	Good
7	Rick Rothackerj	Good
8	Langep	Standard
9	Langep	Good

Table 1. Test predictions with the Stack model

So after we selected de Stacking model as the final model, we now need to test it and predict the labels for the test sample. We can see the first 10 customers of the data with the credit score predicted.

The predicted labels of the model were divided the next way. The most popular classification was th Standard with almost 50% of the predictions.

Credit_Score	Number of Predicted Labels	Percentage
Good	9723	19.45
Poor	16272	32.54
Standard	24005	48.01

Table 2. Classification of the Testl

3. Results

If we compare the classification our model made with the original classification of the train and validation, it is similar. On the three we have the Standard classification as the one with more values, followed by the Poor and at the final the Good.

	Number of Predicted Labels	Percentage
Credit_Score		
Good	4422	17.69
Poor	7313	29.25
Standard	13265	53.06

Table 2. Classification of the Train

	Number of Predicted Labels	Percentage
Credit_Score		
Good	13406	17.87
Poor	21685	28.91
Standard	39909	53.21

Table 2. Classification of the Validation

3. Results

To end this analysis of the results we have the pie graphics of these data we just saw, so we can appreciate better the way the three data samples were classified almost with the same values on every classification.

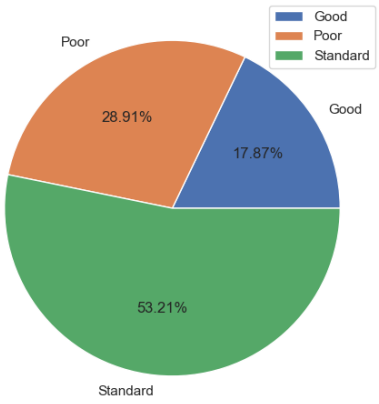

Figure 8. Classification of Train sample

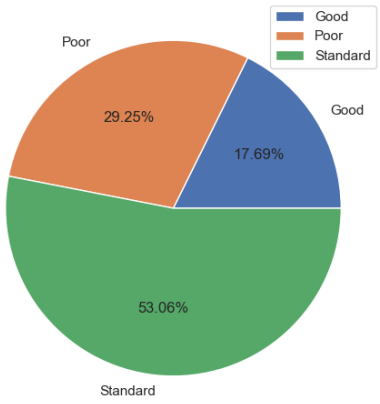

Figure 9. Classification of Validation sample

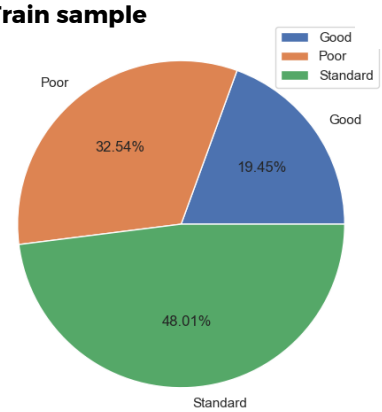

Figure 10. Classification of Test sample

4. Conclusions

After going thru all the required processes and executing different classifiers we observed that Stacking classifier model was the best between our different nominees we could find; In order to reach a best optimization of the model we could consider another variables.

If we think about considering different variables we could fall doing a model that is unethical because, if we start considering physical, ethnical, and educational aspects, could certainly improve the model but it would be on the real life harmful because you could damage the life of the people just because they work on a specific area, have certain skin color or their age they are, just to give examples.

So, what we have is our own model, based on what already exists and also what we believe are important aspects to do this kind of evaluation, certainly we did not reach a high percentage of success in our best model, but it's a decent approach to these kind of models; considering it's our first time doing this.

Thank you for reading this report.

5. References

- **¿Qué es KNN? | IBM. (s. f.). https://www.ibm.com/mx-es/topics/knn**
- **Sharmasaravanan. (2023, 25 agosto). Understanding Stacking Classifiers: A Comprehensive guide. Medium. https://sharmasaravanan.medium.com/understanding-stacking-classifiers-a-comprehensive-guide-195bfab58e48**
- **What is Random Forest? | IBM. (s. f.). https://www.ibm.com/topics/random-forest#:~:text=Random%20forest%20is%20a%20commonly,Decision%20trees**
- **Tuychiev, B. (2023, 27 diciembre). A Guide to The Gradient Boosting Algorithm. https://www.datacamp.com/tutorial/guide-to-the-gradient-boosting-algorithm**