Machine Learning Models for Bad Loans Prediction

M.r Alejandro Rodríguez Domínguez[1]

Abstract In this project we try to predict bad loans with Machine Learning models. We approach the problem as a Supervised Classification problem. We conduct a preprocessing step, and deal with imbalance. As a first step we apply various Machine Learning models and identify the best ones. With the remaining we try to improve the accuracy of these models with Hyperparameter Optimization and Cross Validation. Our findings suggest that these models are extremely good in predicting Bad Loans when they occur, however we see a relatively high value for False Positives. The focus of our research is in trying to reduce the False Positives for this problem. We try to infer were the problem is coming from (Oversampling techniques, preprocessing). Neither of those have nothing to do so we approach this problem with an ensemble model combining the most appropriate models to reduce False Positives.

1 Introduction

For this project we use a dataset from Kaggle called Bank Loan Status Datasets. This dataset contains debtor information for their Loans. It includes balance information as well as credit information and history of the debtor. For each Loan it includes the number of bankruptcies since the inception of the Loan. This will be our Target. The aim of the project will be a model to predict (by classification task) if a Loan will have bankruptcies or not (ie. It's a Bad Loan). This is an important problem as it can help Banks and other financial institutions to asses based on Data the quality of a debtor that has incurred a Loan with that institution. It can serve to measure the credit risk of the institution overall by studying the amount of Bad Loans in their portfolio. But also, it is useful to develop strong application procedures for new Loans.

Few works have been carried out in this dataset (five), mainly some preprocessing steps and the utilization of some Machine Learning models but with very low performance overall. This is a motivational argument for this project to try to come up with some interesting findings.

In the early XX the problem of Bad Loan prediction was performed by hand by an extensively number of professionals using Financial Ratios. With the advance of statistical analysis these professionals started using models reducing resources in terms of workforce and time. This was done with Discriminant Analysis, Logit and Probit Models as well as linear regression [1]. After that, new models were introduced such as SVM, Decision Trees and Logistic Regression. These models had a great improvement in performance and become quite popular. Further in time, Neural Networks and Deep Learning were introduced into this topic and had been challenging other models such as SVM, Decision Trees and Logistic Regression since then. Even though there has been an increase in popularity for these NN and DL techniques the previous three methods used appropriately can give similar and, in some cases, better results.

2 Research

The purpose of this project was to use Machine Learning techniques in order to come up with a systematic approach to predict Bad Loans that could improve the performance of the existing methods. Preprocessing steps are vital for this type of problem. Even if we have the most powerful model for these type of problem, if we do not feed this model with the appropiate data the results will be meaningless. Bad Loans prediction datasets are imbalanced by nature (as bankruptcy should be a rare event). As any datasets, we must deal with outliers, missing data, handle cathecorical data so that the model can get more accuratelly the overall structure of the data. We filter some models based on accuracy and we try to improve the performance of the latter models with a set of techniques that include Cross Validation, Hyperparameter Optimization. Finally we end up with few extremely good models for this problem but that have a mayor important drawback. This drawback appears in every model por this type of problem. Is the problem of False Positives.

The good news is that our models detect bankrupcies when they occurre almost all the time. However, our worry is to reduce the number of times that the model predicts that there will be bankrupcy and it does not append. We try to solve this issue by first trying to identify the source of the problem and after finding the best solution.

2.1 Problem due to imbalance in the Data

In order to deal with the imbalance of the dataset we use an oversampling technique called SMOTE [2], which consists on creating artificial data from the minority class. We firstly believed that due to this oversampling, the model was getting biased towards the minority class and the number of False Positives increased.

In order to try to solve this we used different variants of oversampling techniques. We first try to optimize the Ratio between the majority and the minority class, but we see no improvement in False Positives. We then utilize certain variants of SMOTE [3]. We use a study consisting of 85 SMOTE variants and we conclude that any of those improves the issue of False Positives. It can be found in the code, line 704, needs smote variants installation package (Link in the code).

2.2 Ensemble solution

We try to identify the problem in the preprocessing step, but we do not find a solution. We approach the solution with an ensemble model. We combine in an ensemble models that have lower number of False Positives but greater number of False Negatives with models with greater False Positives and lower False Negatives. Our intuition is that in an ensemble, characteristics of both types of models will be combine and come up with a solution to our problem. We build an ensemble model with 3 models from the Section 3 that, based on the confusion matrix, together can reduce the number of False Positives.

3 Methodology

For the project we used the following preprocessing techniques: Outlier Detection, Dataset Imbalanced, Feature Encoding, Dealing with Missing Values. We further continued with Model Selection, Hyperparameter Optimization.

3.1 Preprocessing the Dataset

For the Dataset we had to carry out some preprocessing techniques. We have two files for Training and Test sets, so we carry the same preprocessing techniques in both files, like if they were part of the same file. First, we dealt with outliers.

For outliers we used the Inter Quantile Range (IQR), which is calculated as the difference between the Third (Q3) and the First (Q1) quartile. We extract all instances that have numerical values outside of a Range. This Range is calculated by adding to Q3 the IQR multiplied by a factor and subtracting the IQR to Q1 multiplied by the same factor. This factor is optimized so that at the end we have numerical features with a distribution with no fat tails. We check this using Boxplots.

We then deal with missing values. First we identify the number of missing values and we eliminate the Feature:"Month Since Last Delinquency" as it has almost 50% of values missing. For the remaining features ´missing values we use the Iterative Imputer from Sklearn.[4]. We use the parameters by default that uses Bayesian Ridge. In order to use the Iterative Imputer, we must preprocess some categorical features with a Label Encoder. The iterative imputer will estimate the value for the missing values running regression with the remaining features. For that we need to convert string features into numerical categorical features. We run the same technique for handling categorical data for the Iterative Imputer than we use for the future Machine Learning models.

For the case of "Years in the current Job" we use 10 numerical categories to replace the initial ones (From less than 1 year to more than 10 years). The rest of the features either are numerical or in case they are categorical with words we convert them to numbers. Once the Iterative Imputer has estimated the new values for the missing values, we do the following further prune of data.

- For new estimated categorical values that do not fall in the initial categories we eliminate those instances (less than 0.01% of the original dataset)
- For those numerical features such as Bankruptcies and Tax Liens which have values in a small interval of integer numbers, we eliminate those predictions that fall outside those integers (less than 0.01% of the original dataset)

The target is bankruptcies and it ranges from 0 to 7. Almost 99.9% of the Target in all the dataset contains 0 and 1. The remaining lies between 2 and 7. We convert all values greater than 1 to 1 so we can have a binary classification problem.

The next step is scaling the data so that we can use it in a machine learning model. We use two type of scaling, standardization and Min Max scaler. For those numerical features (Current Loan Amount, Credit Scoring, Annual Income) that have a large range of values we use standardization. For the rest we use Min Max scaler.

We decide to not extract any feature as we do not have a large amount, however we show below by a regression analysis the predictive power of each feature on the Target.

Score: 96.35654151530119 for feature Loan Status
Score: 3835.006326711485 for feature Current Loan Amount
Score: 230.934398528134 for feature Term
Score: 25.102646282216362 for feature Credit Score
Score: 1043.8850168840727 for feature Annual Income
Score: 374.66703071289805 for feature Years in current job
Score: 147.70891038945135 for feature Home Ownership
Score: 244.63758882527466 for feature Purpose
Score: 1923.38116653727 for feature Monthly Debt
Score: 2422.353386354031 for feature Years of Credit History
Score: 37.8216278625515 for feature Number of Open Accounts
Score: 342636.2596188897 for feature Number of Credit Problems
Score: 15176.23570472835 for feature Current Credit Balance
Score: 12861.837452468722 for feature Maximum Open Credit
Score: 278.2081604645467 for feature Tax Liens

We end the preprocessing step with the treatment of the imbalance of the dataset. We only deal with imbalance in the training set (Training file). For this process we use SMOTE. As it is mentioned in the Research section of this paper, we have two problems, firstly model selection and secondly the research topic which consist on dealing with the False Positives issue. In the Research section we deal with the research topic but in this section, we focus on selecting the best models for the problem. Both sections are related because the models selected in the Methodology section will be used for the Research section to deal with the research topic.

For this section, in order to deal with imbalance, we use the standard SMOTE from Sklearn. Before SMOTE we have 8516 ones and 73670 zeros. We artificially generate instances with target one to end up with a balance of 73670 instance of each class.

3.2 Model Selection

We run a list of Machine Learning models so that we can get an initial idea of which models perform better than others for this specific problem. For that we use the default parameters from Sklearn. In order to asses the performance of each

model we look at the confusion matrix which gives more information than simply the accuracy for this type of imbalanced dataset.

We have the following results for the different models:

	TP	FP	FN	TN
Logistic Regression	8419	208	0	946
SVM	8445	182	1	945
SGD	8326	301	0	946
Decission Tree	8519	108	45	919
Random Forest	8518	109	27	919
Ada Boost	8355	272	12	934
Gradient Tree Boosting	8517	110	21	925
Neural Networks	8409	218	0	946

From this study we end up with four favorite models to continue our study: Logistic Regression, Support Vector Machine, Random Forest and Gradient Boosting. This selection has been done by penalizing more False Negatives and less False Positives, as it is more important that the model at least predicts well when there is bankruptcy. We have added other two models that have more False Negatives but that reduce considerably the False Positives.

With these models we continue carrying out Hyperparameter Optimization. For this we use Grid Search Cross Validation from Sklearn. For each model we built a grid with a range of hyperparameters and optimization algorithms. Grid Search will come up with the best combination of these two by performing Cross Validation on each combination and taking the best performing one in terms of accuracy. Below we show the best models for the task of bankruptcy prediction for this dataset.

4 Evaluation and Conclusion

We divide this section in two, firstly, we evaluate our results of the Methodology section (Section 3) and then we focus on the Research section (Section 2). After Hyperparameter Optimization with Grid Search CV we obtain 4 models that do not improve substantially from the standard parameters chosen by Sklearn. The following models are selected with their respective parameters and confusion matrix as being an imbalanced dataset its the best measure to asses performance for this problem :

	TP	FP	FN	TN
Logistic Regression	8445	182	0	946
SVM	8445	182	1	945
Gradient Tree Boosting	8517	110	21	925
Random Forest	8374	253	0	946

The final models are :

- Logistic Regression : {'C': 10.0, 'penalty': 'l2'}
- Random Forest: {'criterion': 'entropy', 'max_depth': 8, 'max_features': 'auto', 'n_estimators': 200}
- SVM:{'kernel':('linear'),'C':(1),'gamma':(,'auto'),'decision_function_shape' :('ovr'),'shrinking':(True)}
- Gradient Tree Boosting: {"loss":["deviance"], "learning_rate": [0.1], "min_samples_split":2, "min_samples_leaf": 1, "max_depth":[3], "criterion": ["friedman_mse""],"n_estimators":[10]

For the Research Section (Section 2), we build an ensemble model with models from section 3 that we understand based on what we stated in that section can solve the issue of False Positives, these models are : Logistic Regression, SVM and Gradient Tree Boosting.

We build an ensemble model with the Logistic Regression, SVM and Gradient Tree Boosting of above after the process of Hyperparameter Optimization that we carried out to come up with the best models. We try assigning different weights to each model during the ensemble and for each combination we get different values of FP and FN. The final combination we use is : 13% for both SVM and Logistic Regression and 74% for Gradient Boosting. With this we obtain the following matrix in the test set :

	TP	FP	FN	TN
Ensemble	8503	124	18	928

As we can see from the results, we can improve the problem of False Positives at least partially but we can not come with a extremely good solution for this problem with an ensemble model.

However, we can conclude that depending on what the user wants to penilize, either False Positives or False Négatives the user should use one model or the other. Best models for this problem that reduce the number of False Positves are Random Forest and Gradient Tree Boosting. These models will fail a bit more in predicting bankruptcies, however they will predict less bankruptcies when they do not occur. On the other hand, if you want to have the highest accuracy in predicting bankruptcies when they occur at expense of having predicted some bankrupcies when they do not occur you should use : Logistic Regression or Support Vector Machine.

The final models develped in this paper have a very good performance for bankruptcy prediction in Bad Loans. We had focus in reducing False Positives because its the only failure one could mention regarding these models. Areas of future work that would be :

- Implement an oversampling method that do not generate instances that are closed to the boundaries between class Targets. We tried to modify the ratio of Mayority/Minory class in the SMOTE generation so that this could be applied but that did not solve the issue with False Positives. We think that with further research there must be a way to generate artificial instances that avoid False Positives in the test set.
- With further research we can improve the ensemble model to penalize for False Positives. This could be approach by adding more than three models to the ensemble and optimizing the weights.

Apart from the suggested solution, that the user should decide which model to use depending if they want to penalize FP or FN, we have not find a specific solution for this issue. However, this paper have ilustrated many stages of the application of Machine Learning models with imbalanced datasets where the problem is not coming from and therefore we have narrowed the fields of study in order to solve the problem of False Positives.

References [2]

1. Uzair Aslam, Hafiz Ilyas Tariq Aziz, Asim Sohail, and Nowshath Kadhar Batcha∗: An Empirical Study on Loan Default Prediction Models. (2019).

2. N. V. Chawla, K. W. Bowyer, L. O.Hall, W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," Journal of artificial intelligence research, 321-357, 2002.

3. György Kovács. An empirical comparison and evaluation of minority oversampling techniques on a large number of imbalanced datasets. (2019)

4. Stef van Buuren, Karin Groothuis-Oudshoorn "mice: Multivariate Imputation by Chained Equations in R". (2011).