A comparison of classification models and re-sampling methods on imbalanced data

Group PI: Peter Frankild, Filip Soudakov, Jonas Rossen and Noel Pedersen

May 25, 2023

Abstract

In this article three different classification models, random forest [17], deep neural network [16] and linear logistic regression [15], are compared in accurately predicting good and bad creditors. The aim is to determine which classification model performs best and which hyperparameters and methods can provide optimal results in terms of accurately predicting if a person is a bad or good creditor. The different models are analysed through the use of grid-search [14] to determine the best hyperparameters, and different re-sampling methods such as random oversampling [6], random under sampling [8] and SMOTE [7] to determine how the models is performing on a balanced dataset. Through this analysis it was found that the combination of SMOTE and the random forest classifier it is possible to get an near perfect accuracy score, indicating that this model performs significantly better on a balanced dataset.

Introduction

The COVID-19 pandemic has had a significant impact on the financial stability of individuals and businesses worldwide, resulting in an increased risk of loan default. This has had a ripple effect effect on the banking sector, which has had to grapple with the challenges of managing an increasing number of defaulting loans. [11]

Credit scoring which means classifying customers by their credit risk level has been an important mechanism in the financial sector for a long time and different techniques have been used for this. [2]

The difficulties in obtaining large, balanced and representative data in the banking sector makes it crucial to apply the right predictive models designed to deal with unbalanced datasets, such as using methods like random forests, which is known to work well with smaller datasets. [19]

To ensure that the models are accurate, it is important to use techniques such as data normalisation, feature engineering and data cleaning to guarantee that the data is representative of the population being modelled.

In our research, we tested three different classification models, a random forest model, linear logistic regression, also known as logit in Sklearn, and a deep neural network, on a dataset of 1000 individuals with 20 features and one class describing whether they are likely to repay the loan back or not. The dataset is imbalanced in favour of people with a good credit risk by 70/30 percentage division, making it necessary to create a strategy to handle imbalanced datasets. The dataset is from Kaggle, an online community for machine learning practitioners [12]

Problem

Which of the three models is the best model to solve a binary classification problem in terms of accuracy, while dealing with imbalanced data of bank customers and their credibility on loans, and what re-sampling methods should be used to balance the data when working with such models?

Methods

Data preparation

The data was prepared for the three different classification models with techniques such as one-hot encoding [21] to convert categorical variables into numeric values and normalization [20] and feature scaling such as MinMax scaling [18] to standardize the numerical values. Data was also checked for null-values. Data visualization was used both on the categorical and numerical data in the form of diagrams to identify anomalies or patterns in the data distribution.

The models researched were chosen for various reasons. Logistic regression is the most commonly used classifier used in credit scoring [2]. Random forest has previously outperformed other classifiers in credit scoring on several metrics [22]. Previous research on credit classification models indicates that deep neural networks are not optimal due to their computational requirements, lower performance compared to other models, and the fact that they are "black-box" models meaning that it is difficult to interpret why they reach a certain outcome [4]. Still, a DNN was chosen as an experimental model as an attempt to test if it can be a viable solution given the right parameters. We optimize the models by tuning hyperparameters, both manually and using grid-search, to achieve the best performance on the data [23].

Grid search is a hyperparameter optimization technique that involves testing different combinations of hyperparameters to find the optimal set of hyperparameters that maximizes the performance of the model. The technique involves defining a grid of hyperparameters, and then training and evaluating the model for each combination of hyperparameters in the grid. The grid-search was set to determine the best parameters for the model by accuracy.

For the random forest classifier, the parameters was tested by varying the amount of trees in the forest, the amount of splits each tree is allowed to make and the minimum amount of samples required for a leaf to be considered a leaf[17].

To deal with imbalanced data, techniques such as Random Oversampling [6], Random Undersampling [8] and SMOTE [7] were used as they have been giving good results in previous research done on imbalanced data [10]. Also, a Stratified Shuffle Split technique [**?**] was used to divide the dataset into three separate splits where the distribution of the classes was the same in each split, so that we could achieve comparable results.

To deal with overfitting in deep neural networks, both dropout layers and weight regularization was used. Dropout is a method that randomly removes a certain fraction of the hidden nodes from the network, thus making the nodes less dependant on each other and perform better on new data. L2 weight regularization is used to keep the size of the weights small while still maintaining all of the learning features. [5]

Each model is evaluated on a training set and validation set to understand if the model is under- or overfitted, and at last each model is tested on a previously unseen test set. The models are then compared by the accuracy and f1-score (except DNN) to determine the best performing model. Accuracy was chosen as a performance measure as it is easy to interpret, meaning the total fraction of true predictions, while a commonly used f1 gives an aggregate score that balances between precision and recall [9]. Both of the scores scale from 0 to 1. In further research we would also compare the models' performance based on metrics that address imbalanced datasets, such as Youden's index [1]

Analysis

In the analysis of the three different models the first analysis was performed without any modifications to their default settings and parameters. This step makes it possible to understand the baseline performance of each model and to determine which modifications had the most significant effect on each model's performance.

Random forest analysis

The random forest classifier was built using the Sklearn ensemble model. In the first analysis of the random forest classifier, the default model was fitted and trained on a dataset where the data had not been balanced or stratified. After the first analysis, the model was fine-tuned by using grid-search to determine the best hyperparameters for the given dataset. The maximum depth and minimum samples per leaf were tested with relatively small values, but not less than two as this could lead to overfitting of the models [3]. The number of estimators were tested with both low and high numbers to determine the optimal size of the forest. The results were then used to compare how new modifications to both the model and the data affected the models performance. The result of this analysis was an accuracy score of 0.87 on the training set and 0.73 on the validation set, indicating that the model was overfitted to the training set and not good at generalizing. This is a general issue with tree classifiers as they try to memorize rather than generalize [13].

Three different re-sampling techniques were then used to test how each re-sampling method affected the model. The model was first tested on a dataset where the Random oversampling method was used to balance the data, by synthesizing additional data points to the minority class. The model was tested with the default settings of the random forest classifier and then the hyperparameters were fine-tuned using the grid-search technique. The results of this analysis was a score of 0.99 on the training set and 0.73 on the validation set, indicating an even higher overfitted model than in the first analysis.

The model was then tested on a dataset where the random undersampling method was used to balance the data by removing data points from the majority class. This resulted in a score of 0.99 on the training set and 0.67 on the validation set, indicating that this affects the models performance even worse than the random oversampling method.

The stratified shuffle split was then used on the dataset to compensate for the overfitting discovered in the previous analysis. The results of the analysis still indicated that the model was overfitted to the training set as the results was 0.91 on the training set and 0.77 on the validation set, but it also showed that the model had a slightly better ability to generalize from the data when the data was stratified.

At last the random forest classifier was tested on a dataset where the re-sampling method SMOTE was used. SMOTE is also an over sampling method but different from the random oversampling method, as it synthesizes data-points to both the minority and the majority class. By using this method an equal amount of classes was achieved in the training set. When analysing and fine-tuning the model to this dataset it was found that the model was able to perform much better and was able to generalize from the training set, eliminating the issue of overfitting when tested on a validation set that was not re-sampled. The results were a 98% accuracy on the training set and 95% accuracy on the validation set which indicates a very well performing model in general.

Logistic regression analysis

The logistic regression model was built using Sklearn. Similarly to random forest the model was first trained on a dataset that had not been balanced. After the first analysis a grid-search was then performed to determine the best parameters to use on the dataset. A lib-linear solver and a penalty of l1 was suggested but these parameters did not improve the performance.

The three previously used re-sampling techniques were again implemented in the logistic regression model. Comparing these results of the three different re-sampling methods it appears that the model performed better on the original imbalanced dataset according to f1 score which was 0.82 and with no significant difference in accuracy which was 0.73 on all models except the one using stratified data where the score was 0.66. However, the results on the SMOTE dataset indicate that the model performs just as good as the original dataset.

Deep neural network analysis

The DNN model was built using a Tensorflow Sequential model which required the data to be preprocessed and concatenated in a single Tensor layer. The categorical data was one-hot encoded and the numeric data was normalized. To address issues of an imbalanced dataset, each of the dataset splits were stratified to ensure that the distribution of the two classes had the same ratio in each of them.

Five different variations of the model and hyperparameters were tested. The only performance metric used on the DNN was accuracy due to Tensorflows limitations, and it should be addressed in further research by an alternative method such as manual calculation of f1 for better comparison with other models. A loss function was used to determine the error of the predictions.

The models were all trained for not more than 250 epochs as the cost curve of the loss and accuracy showed that the model was not learning further (1). After the first baseline run the result showed a considerable loss of about 45% and no improvement after 10 epochs. Another model with ReLU and Sigmoid activation functions was then implemented; this time the model quickly reached a perfect accuracy on the training set but a considerably lower score on the validation set. This model was then further regularized with dropout layers taking half of the nodes between each hidden layer. This resulted in much worse training score and a wide fluctuation in learning curve. To try to control this, only one dropout layer was left while the dropout fraction was lowered to 20%. Learning rate was also halved for more stable learning. The model performed again much better on the training set but still poorly on the validation set. Further weight regularization did not improve the performance of the model either.

As the results with oversampled dataset using SMOTE in random forest were good, in future research this methods could also be used to train the neural network for better comparison.

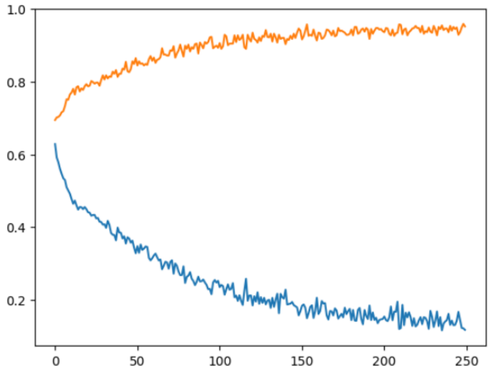


Figure 1: Cost (blue) and accuracy (orange) curves after training a DNN with single dropout layer over 250 epochs.

Findings

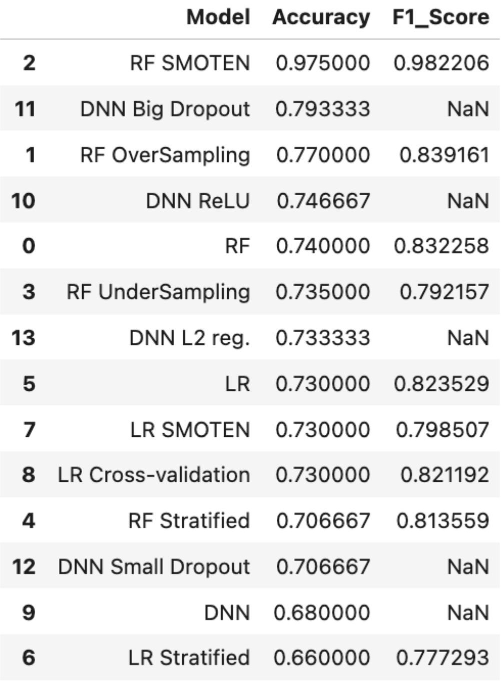
Through the analysis of the different models it was found that using the random forest classifier on a dataset synthesized through the re-sampling method SMOTE, made significant improvements to that models ability to generalize and predict with a near perfect accuracy score. It was also found that other methods such as Random Undersampling and Oversampling did not make any significant improvements to the models performance while the method of Stratified Shuffle Split made minor improvements to the models ability to generalize.

Looking at the table (2) the best accuracy score achieved on the logistic regression model and deep neural network were 0.73 and 0.79 respectively, while the random forest model scored 0.97. All logistic regression models had a similar performance no matter what parameters or re-sampling methods were used. On the other hand, using several dropout layers had a significant improvement in test performance on the DNN, indicating that it generalized better.

With a 97% accuracy, the random forest model in combination with SMOTE re-sampling method makes the model perform the best in comparison to the other models with near perfection without overfitting. Still, further research needs to be done to compare models with metrics that address imbalanced data such as Youden's index to get a more clear picture of how they perform on the minority class.

Conclusion

The findings indicate that while the random forest is able to score a near perfect accuracy score and significantly higher than the other two models, it is only able to do so in combination with the SMOTE re-sampling method. This means it is not possible to confirm that the random forest is able to outperform the other models on a imbalanced dataset but is able to do so when the dataset has been balanced by the SMOTE re-sampling method.



	Model	Accuracy	F1_Score
2	RF SMOTEN	0.975000	0.982206
11	DNN Big Dropout	0.793333	NaN
1	RF OverSampling	0.770000	0.839161
10	DNN ReLU	0.746667	NaN
0	RF	0.740000	0.832258
3	RF UnderSampling	0.735000	0.792157
13	DNN L2 reg.	0.733333	NaN
5	LR	0.730000	0.823529
7	LR SMOTEN	0.730000	0.798507
8	LR Cross-validation	0.730000	0.821192
4	RF Stratified	0.706667	0.813559
12	DNN Small Dropout	0.706667	NaN
9	DNN	0.680000	NaN
6	LR Stratified	0.660000	0.777293

Figure 2: Results of all evaluated models sorted after accuracy score.

References

[1] Mohamed Bekkar, Hassiba Kheliouane Djemaa, and Taklit Akrouf Alitouche. Evaluation measures for models assessment over imbalanced data sets. *J Inf Eng Appl*, 3(10), 2013.

[2] Christine Bolton et al. *Logistic regression and its application in credit scoring.* PhD thesis, University of Pretoria, 2010.

[3] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[4] Björn Rafn Gunnarsson, Seppe Vanden Broucke, Bart Baesens, María Óskarsdóttir, and Wilfried Lemahieu. Deep learning for credit scoring: Do or dont? *European Journal of Operational Research*, 295(1):292–305, 2021.

[5] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[6] imbalanced learn. Over-sampling, 2022.

[7] imbalanced learn. Smote, 2023.

[8] imbalanced learn. Under-sampling, 2023.

[9] Jake Lever. Classification evaluation: It is important to understand both what a classification metric expresses and what it hides. *Nature methods*, 13(8):603–605, 2016.

[10] Forhad An Naim, Ummae Hamida Hannan, and Md Humayun Kabir. Effective rate of minority class over-sampling for maximizing the imbalanced dataset model performance. In *Proceedings of Data Analytics and Management: ICDAM 2021, Volume 2*, pages 9–20. Springer, 2022.

[11] Asror Nigmonov and Syed Shams. Covid-19 pandemic risk and probability of loan default: evidence from marketplace lending market. *Financial Innovation*, 7(1):1–28, 2021.

[12] Srihari P. Credit risk customers, Apr 2023.

[13] Lior Rokach and Oded Maimon. Top-down induction of decision trees classifiers-a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 35(4):476–487, 2005.

[14] scikit learn. Gridsearchcv, 2023.

[15] scikit learn. Logistic regression, 2023.

[16] scikit learn. Neural network models, 2023.

[17] scikit learn. Random forest classifier, 2023.

[18] scikit learn. sklearn.preprocessing.minmaxscaler, 2023.

[19] Toose Speiser, Miller and Ip. A comparison of random forest variable selection methods for classification prediction modeling. 2019.

[20] TensorFlow. tf.keras.layers.normalization.

[21] TensorFlow. tf.one$_{h}ot$.

[22] Yuelin Wang, Yihan Zhang, Yan Lu, and Xinran Yu. A comparative assessment of credit risk model based on machine learninga case study of bank loan data. *Procedia Computer Science*, 174:141–149, 2020.

[23] Ni Wayan Surya Wardhani, Masithoh Yessi Rochayani, Atiek Iriany, Agus Dwi Sulistyono, and Prayudi Lestantyo. Cross-validation metrics for evaluating classification performance on imbalanced data. In *2019 International conference on computer, control, informatics and its applications (IC3INA)*, pages 14–18. IEEE, 2019.