CreditClassify: Advanced Credit Score Classification Model

Axel Luca, Preet Patel, Xiaojun Liu

What is exactly the function of your tool (or a method)? That is, what will it do?

Our goal is to create a model based on a combination of pre-existing ones such as the Support Vector Machine and Gradient-Boosting classifier that is able to classify credit scores as accurately as possible.

Why would we need such a tool (or a method) and who would you expect to use it and benefit from it?

We need this kind of tool because it can help banks, financial institutions, and credit rating agencies assess the credit risk of individuals or businesses more accurately. This allows for smarter decisions in financial services like loans and credit card issuance. Expected users include credit rating agencies, risk management experts, financial analysts, loan officers, policymakers, and academic researchers. These users can use this tool to better predict credit default risks, optimize resource allocation, and reduce the risk of bad loans. Also, a more accurate credit scoring model also helps promote fairness in the financial system and improves customer satisfaction.

Do these kinds of tools/methods already exist? If similar tools/methods exist, how is your tool/method different from them? Would people care about the difference? How hard is it to build such a tool/algorithm? What is the challenge?

Yes, similar tools and methods already exist. Machine learning and artificial intelligence technologies are widely used in credit scoring. Models like Support Vector Machines (SVM), Random Forests, and Gradient Boosting are used in the finance industry to assess credit risk.

Our tool may differ in the following ways:

· Intended higher accuracy: By combining the strengths of different models such as the stacking of Support Vector Machines which help with datasets with a large amount of features and Gradient-Boosting classifier which can model complex relationships between features in a dataset, we intend to improve credit score classification accuracy. This specific combination of classification algorithms is not very common as we were not able to find many papers that have tested this before so we would like to implement it ourselves and see the results.

· Intent to have better feature selection: We use advanced feature selection techniques like Elastic Net which combines the benefits of both Lasso and Ridge regressions to make sure that the model only selects the features of the dataset that have the highest effect on credit score classification. We suspect that multicollinearity exists and Elastic Net helps deal with the features in the dataset that are highly correlated. From papers we have seen, while Lasso and Ridge regressions seem to be common ways to select features that are important to machine learning models' predictions and classifications, not many used Elastic Net for such a task.

· Intent to improve data quality: We ran a small experiment on the data already and we found a clear imbalance in the labels used for classification. Therefore, by using Synthetic Minority Oversampling Technique (SMOTE) we can deal with this imbalance.

People will care about the difference between them. Financial institutions and consumers both want credit scoring systems to be more accurate and fairer. Improving the model's accuracy and quality can increase users' trust in the system and reduce the risk of incorrect classifications. We plan to provide this experience by, from what we have seen, uncommon techniques in creating our classification model.

Building such a tool has certain challenges, including:

· Data quality and quantity: As not all of the features are numerical data, we will need some way to make sure that all of our data ends up being in the form of numerical data so that it can be used for our model.

· Data quality and quantity: We need a lot of high-quality data to train the model and deal with any imbalance in the data.

· Model optimization: Tuning and optimizing the model to get the best performance requires repeated experiments with potentially different model hyperparameters and validation methods such as K-Fold .

How do you plan to build it? You should mention the data you will use and the core algorithm that you will implement (either existing algorithm for tools or new algorithm for methods).

First, we will clean and prepare the Kaggle data, using methods like SMOTE to balance it if needed. We will then make sure that all the data has successfully been converted to numerical data. Next, we will look at the data to pick important features using Elastic Net. Then, we will build the model that stacks Support Vector Machines and Gradient Boosting classifiers. Finally, we will adjust the settings of our models and test them to make sure they work well on new data. We will also make sure our models' decisions are easy to understand for the users.

What existing resources can you use?

Data Resources: Credit score classification | Kaggle

Programming Languages:

Python: Widely used for machine learning with rich libraries and community support.

Libraries and Frameworks:

Scikit-learn: For implementing SVM, Gradient Boosting, and other machine learning algorithms.

Imbalanced-learn: Compatible with Scikit-learn and provides additional tools like the ability to use SMOTE on datasets with an imbalance amount of labels used for classification.

Pandas and NumPy: For data manipulation and analysis.

Matplotlib: For data visualization.

Platforms:

Jupyter Notebook / Colab: For interactive coding and data exploration.

Github: Used to share and store code.

How will you demonstrate the usefulness of your tool/method?

The most important metric for us to use to test our model is accuracy as the main goal of our project is to produce a machine learning model that can classify credit scores as accurately as possible. We would like to compare the performance of our tool against existing credit scoring models like logistic regression, decision trees, and other machine learning methods.