Data Mining - 1
FSS2023, Project Report

Team 7

Aakriti Istwal, 1871754
Amela Medolli 1729253
Jona Frroku, 1951865
Jorida Jolla, 1871857
Priyanka Roy, 1933097
Miger Shkrepa, 1959025

University of Mannheim, Germany

Abstract. This project focuses on analysing the dynamics of SBA(Small Business Administration) loan approvals, defaults and associated risks as a classification problem. Using data analysis and machine learning technologies, we are developing a predictive classification model that enables banks to better understand loan outcomes and effective risk management. Extensive data analysis, pre-processing, modelling adaptation, feature selection and hyper-parameter optimization are the methods we have used to approach this binary classification problem on the SBA dataset referred from Kaggle. The MIS_Status variable, which shows whether the loan has been fully repaid or not, is the target variable. In the end, Macro-F1 scores are used as a criterion for evaluation. This in turn contributes to building a positive ecosystem in which lenders can safely finance small and medium-sized enterprises while successfully managing risks.

Keywords: ML approaches, Classification, Predictive Analysis, Loan approvals.

1 Problem Description

Although the transformational effects of SBA loans are highlighted by success stories like FedEx and Apple Computer, there have also been cases where startups and small businesses have defaulted on SBA-guaranteed loans. Since the SBA's guarantee covers a portion of the loan, it does not extend to the full loan amount. Consequently, banks still face potential losses if a borrower defaults. The risk of default presents a significant challenge for banks when determining whether to approve loans and extend financial support to startups and small firms. In this project, we aim to analyse a historical data on SBA loan approvals and defaults, delving into the factors that contribute to successful loan outcomes, as well as the challenges faced by banks in mitigating risk.

Through comprehensive data analysis, we will explore various dimensions such as industry sectors, loan amounts, borrower characteristics and economic conditions to gain a deeper understanding of the risks and rewards. Using different data mining methods, we will examine the patterns and trends in loan approvals and defaults and identify key indicators that will help predict loan success and failure. In turn, this should help the banks make informed decisions and foster an environment where startups and small firms can thrive while managing risks effectively.

2 Data Understanding

2.1 Dataset

The dataset is provided by the U.S Govt. body "SBA", which aids small businesses in the American Credit Market. It contains information about approved and rejected loans in a CSV file format spanning historical data from 1987 to 2014.

2.2 Data Exploration

The dataset has 779587 unique values which are distributed across 27 attributes. Name, datatype, information about missing values and a succinct description are provided for each of these aspects.

Although the descriptions are clear-cut for the majority of the variables, some variables such as NewExist, LowDoc, MIS_Status and NAICS require a bit more in-depth explanation:

- **NAICS** stands for North American Industry Classification System. This is a 2-through 6-digit hierarchical classification system described in the U.S. economy. The first 2 digits of the NAICS classification represent the economic sector.
- **NewExist** represents whether the business is in existence for more than 2 years (existing business=1) or not (new business=2).
- **LowDoc** represents a "LowDoc Loan" program that was implemented to process loans under $150,000.
- **MIS_Status** simply tells us the status of the loan i.e., whether the loan is in default/charged-off state (CHGOFF) or has been paid successfully (PIF).

We examine each variable to understand its significance and relevance to the issue at hand, as well as how we may utilize it to identify the predictor variable MIS_Status using a variety of data analysis techniques. Our initial findings on the data quality revealed that many attributes had missing values, with "ChgOffDate" leading others with 82% NA values. The distribution of unique values also reveals interesting insights about data entry errors which need to be handled during preprocessing. Figure 1 shows the correlation heat map where “Term” seems to be the most positively correlated attribute w.r.t target. On the other hand, “ChgOffPrinGr” seems to be the most negatively correlated feature. There was no such feature which showed a very strong correlation with the label, hence no strong dependencies can be inferred. We selected the time period (1990–2006) from the loan approvals financial year to complete the analysis because it corresponds to a time without significant economic crises in the history of the US economy. This choice is supported by a loan analysis from figure 2 that includes data from the *Great Recession Period* (2007–2008). Insightful evidence of the financial crisis may be drawn from the statistics, which show a substantial decline in the volume and number of granted loans from 2007 to 2008. As a result, the charged-off amount for this time series exhibited an upward trend.

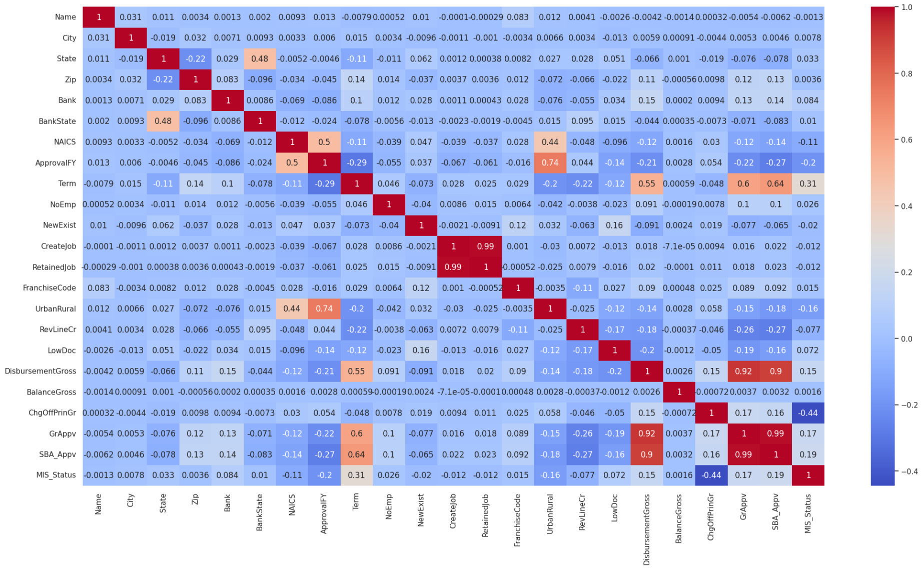

Fig. 1: Feature Correlation - Heatmap

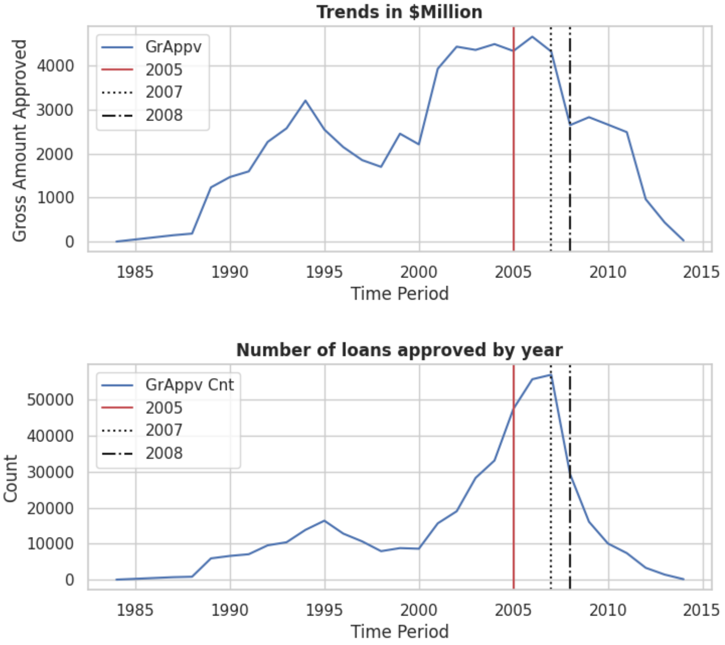

Fig. 2: Gross Amount Analysis w.r.t Time Period

2.3 Target Variable Analysis

The target variable is MIS_Status which indicates whether the loan was Charged off (CHGOFF) or Paid in full (PIF). Charged off loan is usually considered to be in default and PIF indicates that the borrower has successfully repaid the entire outstanding balance of the loan, including any interest and fees. Figure 3 clearly illustrates the imbalance in the class distribution for this variable.

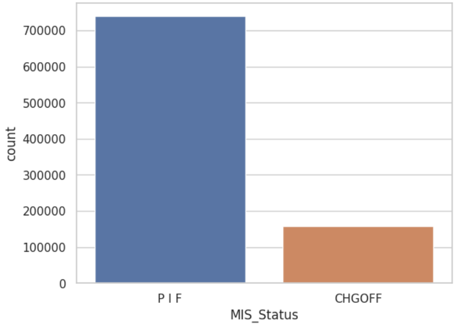

Fig. 3: Distribution of target variable

The variation in loan default rates may be explained by a number of factors that repeatedly show up as risk indicators. The following discussion covers attributes, including Location (State), Gross Disbursement, New versus Existing Businesses and SBA's Guaranteed Portion of the approved Loan, along with some exploratory analysis.

3 Methodology

3.1 Data Preprocessing

We use a number of preprocessing procedures after importing the dataset to confirm the quality and convert it into a representation that is appropriate for classification.
Missing Values: Our initial data quality observations showed that 11 out of 27 features had missing values. To deal with it, each feature is examined and explained why they are so sparse and whether they require further processing. For example *ChgOffDate* contains information of the date when a loan is declared to be in default but more than 80% of its value accounted for being undefined, hence, this was dropped off. Similarly, we also imputed the values of the attributes *City* & *State* by using *ZIP* code values since it had no null values.
Data Consistency: Upon further analysis, we discovered that some columns included values that do not conform to the expected or specified data description and format.

- ApprovalFY: Represented the financial year of the loan approval date but it also recorded the value "1976A", so we eliminated such values because they only represented a very small portion of the data.
- NewExist: The value 0 in this column does not adhere to the dataset description and doesn't cover less than 5% of the total distribution hence, we decided to remove these values.
- LowDoc: The same problem was encountered with this column as well, where the values are supposed to be specified as {Y,N} instead, included values such as {C,1,S,R,A,0}, which were inconsistent with the definition and interpretation of this column. Based on the contribution proportion of the values, it was safe to eliminate these as well.
- RevLineCr: Undergoes the same preprocessing as the *LowDoc* column, where other values that didn't fall under the {Y,N} range were also dropped.

Outliers: Graphical techniques like histograms and box plots (figure 4) have been utilized to visually inspect the data sets for outliers. *ApprovalFY* seems to have outliers and also displays skewness. Additionally, it appears that the distributions of *Term, DisbursementGross, ChgOffPrinGr, GrAppv*, and *SBA_Appv* are all heavily concentrated on a certain range of values.

Feature Generation:

- Binarization: *MIS_Status, LowDoc, RevLineCr* & *FranchiseCode* are all encoded as binary values.
- Label Encoding: Approximately 60 encoded values are generated for *State* and *BankState.*
- Dealing with amounts: The columns *DisbursementGross*, *BalanceGross*, *GrAppv*, *ChgOffPrinGr*, and *SBA_Appv* are by default represented as an object type eg: "$60,000.00". These values are processed by removing **$** and **,** and transforming them to float values.

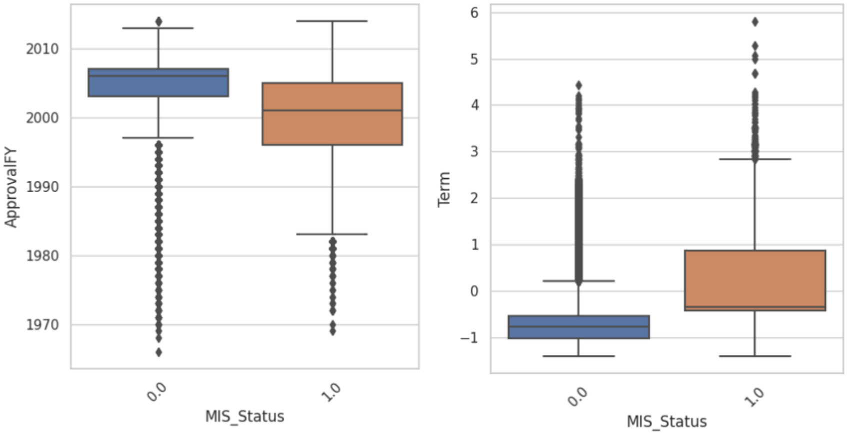

Fig. 4: Boxplot and Histogram analysis

- Dealing with dates: The date columns are converted from string object to datetime object by specifying the format of input date in order to avoid false conversions due to the Y2K format. While creating a more meaningful feature from dates called *DisbursementDuration_days* we observed some values where the disbursement date was before the approval date which is a clear error and hence removed these entries from the dataset.
- Default Days: Created a new feature *NrOfDefaultDays*, which calculates how long it would take a loan to be declared as default from the time it was disbursed. We analysed the effect of a business being new or existing on this with an assumption that a new business would have an average of less number of default days due to its potential instability but it shows that no such effect exists.
- NAICS: The first two digits of the values in this column express general industry so we assigned the appropriate industries to each of them. We then plotted these values to understand the distribution and noticed that most of them fall under the “Retail trade” industry.
- Same State: This feature is created by comparing the borrowers state and bank state. From this, we understood that there is a higher chance to get the loan approved if the lenders and borrowers are residing in the same state.
- Term: We see that the most popular loan terms are around the 100, 250 and 300 month marks with that first value being the most dominant one. Since the term of the loan is a function of the expected lifetime of the assets, loans backed by real estate will have terms 20 years or greater (240 months) and are the only loans granted for such a long term, whereas loans not backed by real estate will have terms less than 20 years.
- SBA Guranteed Percent: This feature is created to observe the ratio of the amount guaranteed by SBA out of the Gross loan amount. On further analy-

sis, it shows that it usually falls under the 40% and 80% of the gross amount approved.

3.2 Modelling

To train our classification models we will be using pipelines with StandardScaler that include the following algorithms:
Dummy Classifier generates predictions using straightforward principles without attempting to identify patterns in the data and serves as a benchmark & baseline for the evaluation.
Random Forests mixes multiple decision trees to make predictions by identifying patterns and connections between input features and loan approval. The hyperparameters tuned in our case include max depth, min samples leaf, and min samples split.
Decision Trees can categorize loans by executing sequential tests on the features at each intermediate node and then sending examples down a path from the root to a leaf node each reflecting the assigned label.
KNN predicts loan categories by comparing attributes to training data and selecting the label with the highest frequency among neighbouring instances.
Support Vector Machines (SVM) The algorithm's goal is to find a line or hyperplane that divides the feature space of loans into approved and not approved regions while optimizing the margin to the nearest occurrences.
Gradient Boosting Classifier is a type of ensemble learning method that combines multiple weak learners (typically decision trees) to create a strong predictive model. Depending on a variety of applicant and loan characteristics, categorization duties entail deciding whether a loan application should be authorized or denied.
XGBoost stands for Extreme Gradient Boosting and uses the combination of multiple decision trees which are trained through gradient boosting. These trees' predictions are combined using a weighted average which allows the algorithm to understand which trees should be given more weight in the final prediction. For hyperparameter tuning we used n_estimators, booster, colsample_bytree, learning_rate, max_delta_step, max_depth, min_child_weight, objective, reg_lambda, scale_pos_weight, subsample, verbosity.
Logistic Regression is a statistical and parametric classification model that uses the Logistic function to estimate the probability of an event occurring. It uses independent variables to predict the outcome of the dependent variable which will always be categorical.

4 Results

4.1 Evaluation Framework

A two-step approach is followed for evaluating the results:

- Without hyper-parameter tuning

- With hyper-parameter tuning

The preprocessed dataset is split up using the train-valid-test split method from the fast-ml module in a ratio of 80:10:10, which inherently does a stratified split on the target label. We used macro-F1 since the data is imbalanced, although we did incorporate this into account and ran the models by setting the class weights as "balanced". As part of feature selection, we use the Select K-best algorithm to select the top-scoring attributes and discard the rest.

4.2 Results

Without Hyper-parameter tuning: The pre-processed dataset is modelled using all primary classification classifiers such as Random Forest, Decision Tree, KNN, Gaussian-Naive Bayes, SVM, Logistic Regression, Gradient Boosting, and XGBoost. Furthermore, experimentation on model fitting is done by using the following approaches:
Approach 1: Without selecting K-best features and without balancing the class
Approach 2: Without selecting K-best features but balancing the class
Approach 3: Selecting K-best features and balancing the class

- Approach 1: As observed from table 1, the best-performing model comes out to be Random Forest with an F1-Score of 0.9894, although XGBoost closely follows this score along with Gradient Boosting classifier.
- Approach 2: Balanced fitting on the classifiers lead to an increase in F1-scores for both val and test and the best model turns out to be XGBoost.
- Approach 3: Multiple features with a score of 700 and less were dropped as they were very less in comparison to others and modelling is done using a balanced weight setting. There were still a few steps of improvement as can be observed from table 1 but XGBoost still performs the best on the test set.

In comparison to baseline models performance from table 1, the rest of the classifiers seem to perform exceptionally well by generalizing better to the test set.

With Hyperparameter tuning: The best-performing models namely Random Forest and XGBoost are selected for hyperparameter optimization using Halving Grid Search. The common approach for this process included selecting the K-best features and defining the value sets for the hyperparameter combinations. As part of the evaluation purpose Repeated Stratified K-fold strategy is used, where K = 5 and a number of repeats = 2; this method proves to be efficient in cases of imbalanced datasets by maintaining the class distribution and reducing the variability in results. Finally, according to table 2, both random forest and XGBoost perform equally well with an F1-Score of 0.99.

4.3 Error Analysis

The original dataset had a lot of discrepancies that were resolved using multiple approaches of exploration, pre-processing and feature engineering. However, the

Model	Approach 1		Approach 2		Approach 3	
	Val	Test	Val	Test	Val	Test
Baseline	0.447	0.447	0.449	0.448	0.447	0.447
Random Forest	0.98941	0.98944	0.99031	0.98927	0.99033	0.98980
Decision Tree	0.97782	0.97694	0.97961	0.97783	0.97761	0.97793
KNN	0.94750	0.94729	0.94819	0.94868	0.970335	0.96975
SVM	0.98169	0.98230	0.986217	0.98470	0.98761	0.98606
Gradient Boost	0.98937	0.98940	0.99037	0.989218	0.99020	0.9898
Logistic Regression	0.97716	0.97656	0.986379	0.986181	0.9865771	0.98615
XGBoost	0.98929	0.98900	0.98909	0.99028	0.98934	0.99065

Table 1: F1 Score : W/O Hyperparam Optimization for Validation/Test

Model	Val	Test
Random Forest	0.99	0.99
XGBoost	0.99	0.99

Table 2: F1 Score: With Hyper-parameter Optimization

model still seems to misclassify some of the data points specifically from the majority class (MIS_Status = 1). The confusion matrix fig 5 shows the number of miss-classifications is 344 in the case of Random Forest tuning and 335 for XGBoost, which is good considering total data points.

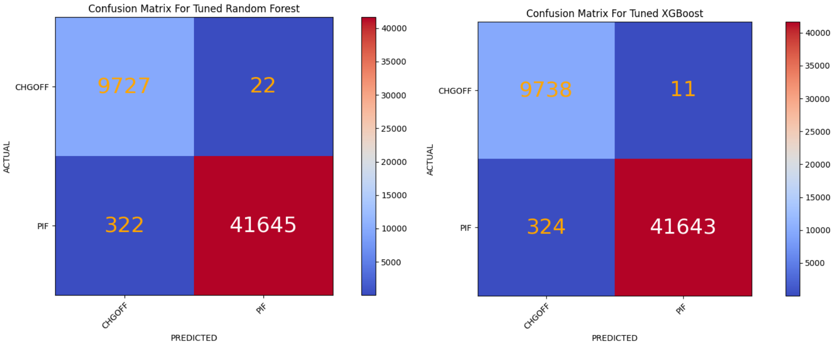

Fig. 5: Confusion Matrix on tuned RF and XGBoost

Furthermore, we wish to analyse the miss-classified data to understand if there is any underlying pattern due to which the error occurs. Both models seem to consider *ChgOffPrinGr* as the most important attribute, but it seems the explanation for the error does not seem to be intuitive from the important features. Huge oscillations from these features do not depict any clear trend observed for the miss-classified data; as seen from figure 6.

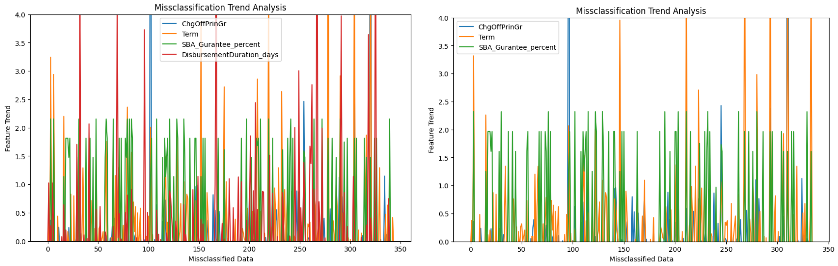

Fig. 6: Feature Trend vs Miss-classified data on tuned RF and XGBoost

5 Conclusion

For financial institutions, machine learning approaches are profitable and can forecast loan acceptance with consistency by using applicant profiles and financial histories. We investigated numerous machine learning techniques in our study and significantly improved over the baseline. Notably, the Random Forest Classifier and XGBoost outperformed all other models including the baseline with a macro F1 score of 0.99 on the test set.

Incorporating gradient boosting methods like Ada Boost, and LGBM may be helpful in the future to improve our models even more as these deal with the problem of excessive bias in machine learning models by employing a sequential strategy to integrate numerous weak learners and incrementally improve predictions. Additionally, adding detailed application data to the dataset, such as employment histories, credit usage trends, and long-term payment histories, may enhance the performance of the models even more. Financial organizations can improve their loan approval procedures and make sure that efficient and ethical lending standards are followed by iteratively improving these models and incorporating more diverse and thorough application data.

References

1. Kaggle Dataset
https://www.kaggle.com/datasets/mirbektoktogaraev/should-this-loan-be-approved-or-denied
2. Scikit Learn Label Encoder
https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html
3. Scikit Learn Scalers
https://scikit-learn.org/stable/modules/preprocessing.html
4. Scikit Learn
https://scikit-learn.org/stable
5. Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011
6. Repeated-k-fold-cross-validation
https://machinelearningmastery.com/repeated-k-fold-cross-validation-with-python/
7. Train-Val-Test Split
https://pypi.org/project/fast-ml/
8. Datetime Conversion
https://docs.python.org/3/library/datetime.html
9. Scikit Learn Pipeline
https://scikit-learn.org/stable/modules/generated/sklearn.pipeline.Pipeline.html
10. XGBoost Classifier
https://towardsdatascience.com/beginners-guide-to-xgboost-for-classification-problems-50f75aac5390