Final Report: Credit Risk Analysis

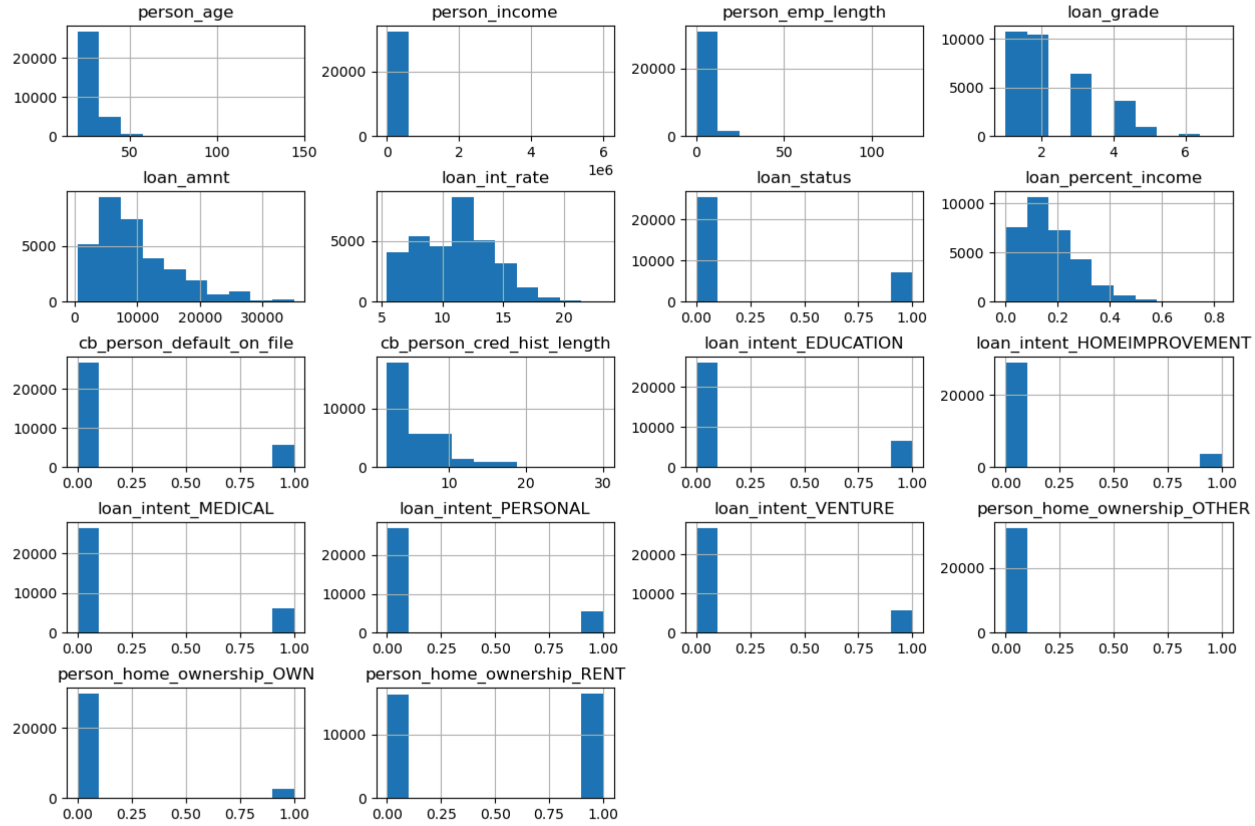
Authored by Benjamin Le

Problem statement

Financial institutions face abundant risk when giving out credit and loans to borrowers. Loan defaults directly impact profit and stability. Predicting default risk would help make these tough borrower targeted decisions. This report aims to answer how financial institutions minimize risk by predicting how likely it is that a borrower will default and which borrower and loan characteristics are most predictive of default risk.

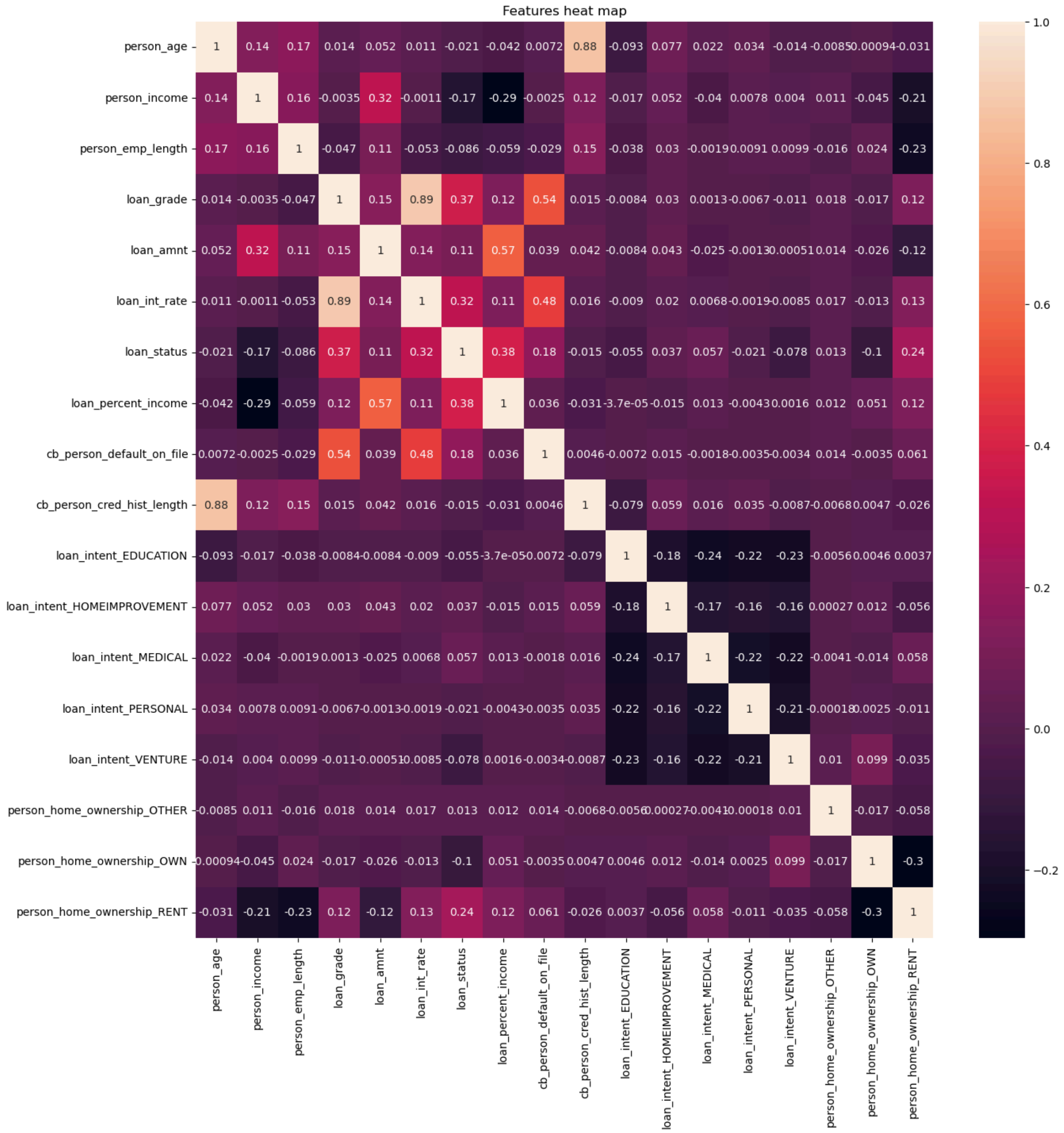
Data Wrangling

A dataset from the credit bureau was used for this analysis. The dataset contains 12 features with 32,581 entries containing data relevant to credit. After cleaning these are the distributions of the raw features:

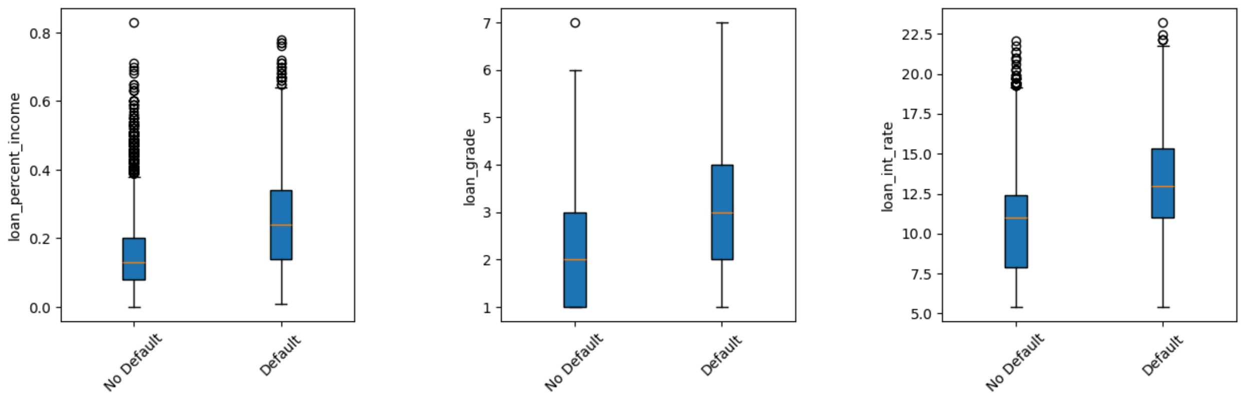


Note: there are more than 12 features now due to the one hot encoding of some variables

Exploratory Data Analysis



This heat map shows the full combination of correlations between all the features. When evaluating the feature of interest, loan_status, it can be seen that the features ‘loan_percent_income’, ‘loan_grade’, ‘loan_int_rate’ had the most significant correlation.



Boxplots of correlation with target variable, ‘loan_status’, with the features with the highest correlation, ‘loan_percent_income’, ‘loan_grade’, ‘loan_int_rate’.

The following is a full description of the features:

person_age: Age
person_income: Annual Income
person_home_ownership: Home ownership
person_emp_length: Employment length
loan_intent: Loan intent
loan_grade: Loan grade
loan_amnt: Loan amount
loan_int_rate: Interest rate
loan_status: Loan status (0 is non default 1 is default)
loan_percent_income: Loan's percent income
cb_person_default_on_file: Historical default
cb_preson_cred_hist_length: Credit history length

Modeling

The dataset was first split into training and test splits of 70/30 with the 'loan_status' feature as the target variable.

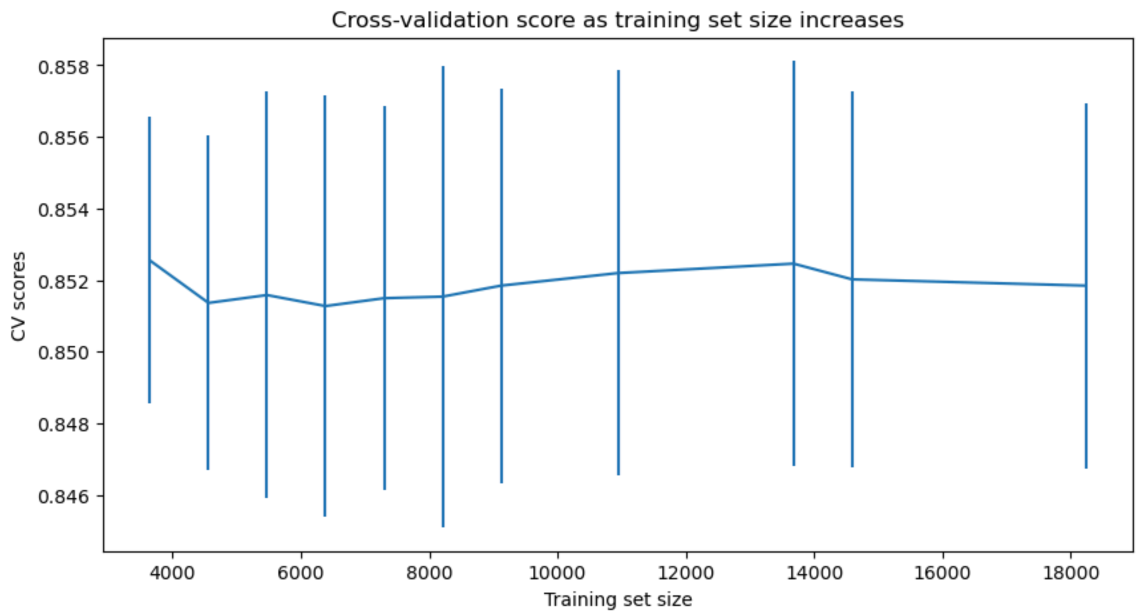
Models evaluated:

- Logistic Regression model
- Random Forest model

GridSearchCV was used for hyperparameter tuning and cross-validation. The Random Forest model was computationally expensive for a traditional GridSearch so RandomizedSearch was used instead.

For the Logistic Regression model, a pipeline of StandardScaler, SelectKBest, and the LogisticRegression object was utilized. StandardScaler scaled the data and SelectKBest selected the k best features.

Similarly for Random Forest, a similar pipeline was made of SelectKBest and the RandomForest object. StandardScaler was not included as scaling is not necessary for a Random Forest model.



This graph is to test if there was enough data by testing how much the CV score changes as the training set size increases. It can be seen that there is not much change in CV score and its variance meaning there is sufficient data.

Findings

The random forest model performs significantly better across all metrics:

Model: Logistic Regression

Accuracy: 0.849

Precision: 0.731

Recall: 0.512

F1 Score: 0.602

ROC-AUC: 0.867

Model: Random Forest

Accuracy: 0.935

Precision: 0.97

Recall: 0.732

F1 Score: 0.835

ROC-AUC: 0.938

The best hyperparameters for the Random Forest model found by the grid search were:

bootstrap: False

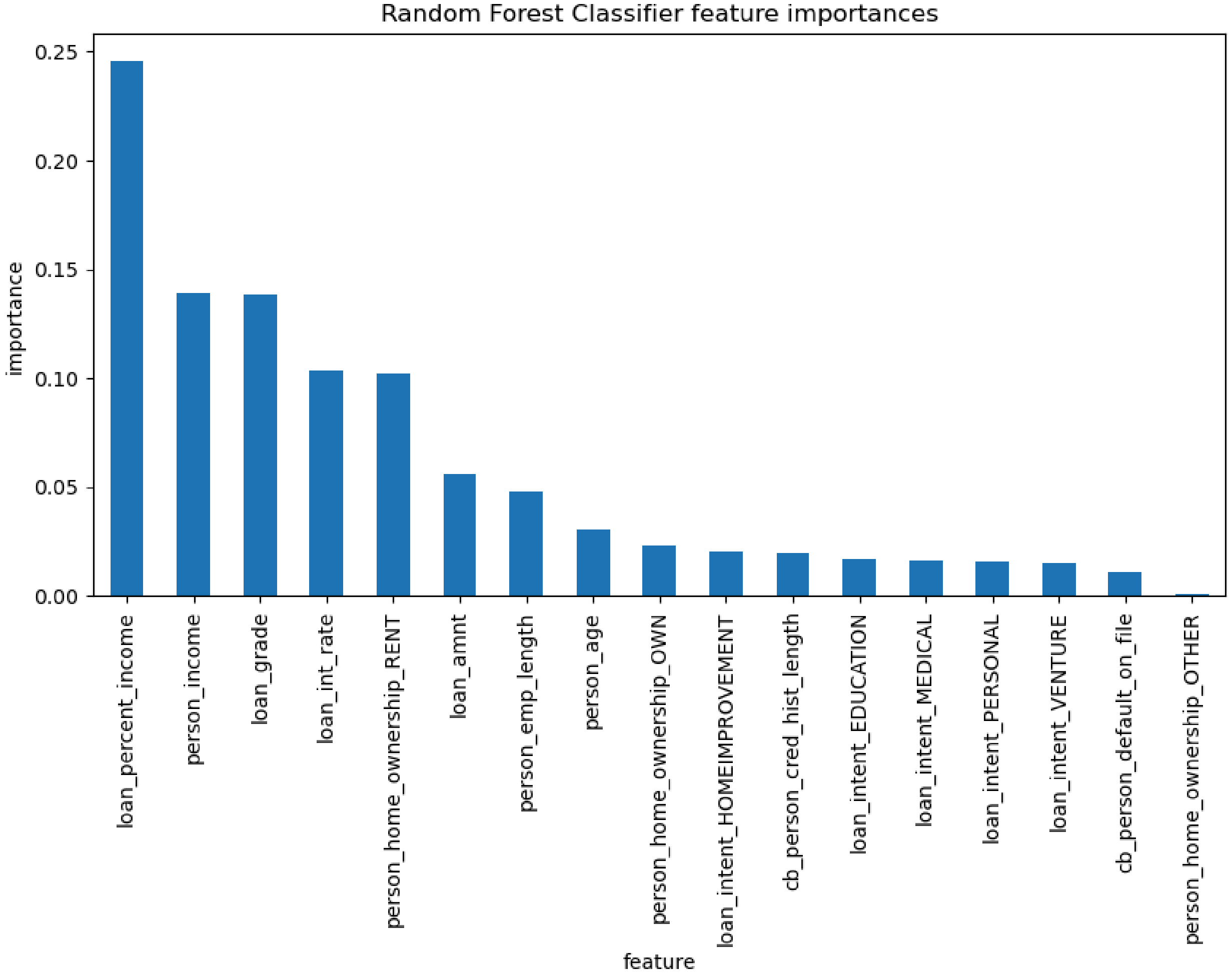
max_depth: 19

min_samples_leaf: 2

min_samples_split: 6

n_estimators: 209

n_features: 17



This shows the features and their importance according to the best Random Forest model.

Recommendations and Further Research

- Create risk mitigation for borderline cases
- Develop risk-based pricing strategies
- Deploy Random Forest model for risk assessment
- Test different models such as gradient boosting or naive bayes