Loan Default Prediction

Capstone Project Presentation

Problem Definition

- **Banks make can make substantial profits on home equity loans.**
- **Opportunity in the market segment.**
- **When a customer defaults, it's costly to the bank and negatively impacts profits.**

How can a bank minimize losses due to loan default?

Objectives

- Make better predictions on customer's risk for loan default.
- Build a predictive model based on empirical data and evidence.
- Discover drivers of loan default and increase business intelligence.
- Define actions to reduce default rates.

Solution Model

- Home Equity dataset (HMEQ)
- 5,960 records of recent home equity loans
- 20% default rate

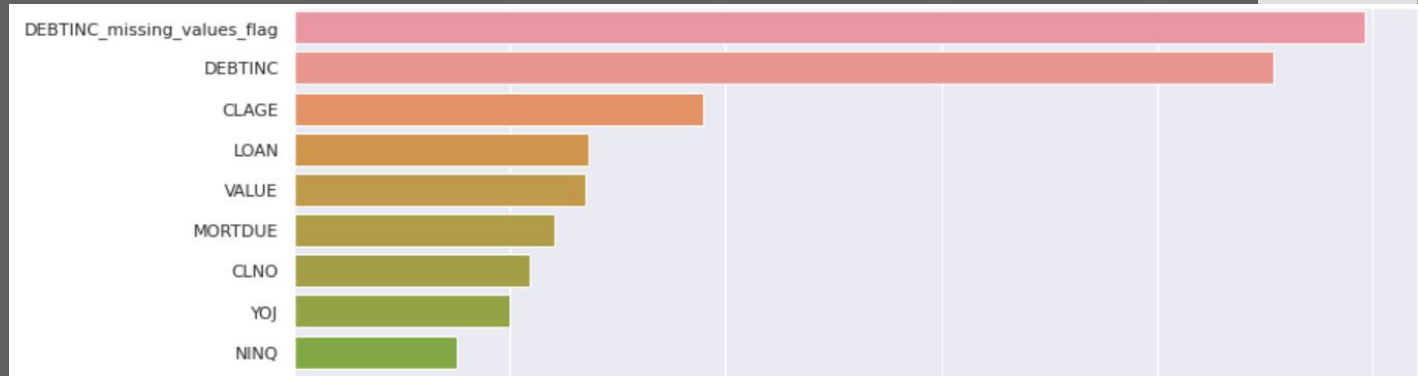
Best predicting model on test set:

- Accuracy score of 89%
- Recall score of 76% (false-negative errors)
- Precision score of 72% (false-positives errors)

Evidence and Findings

- Debt to income ratio is the most important driver of loan default.
- The age of a customer's oldest credit line and number of existing credit line were also top drivers.
- Loan amount, current property value and amount of existing mortgage due are highly correlated and key features for loan default prediction (opportunity for future feature engineering).
- Use common attributes to develop customer profiles.

Random forest list of Features of Importance in order



Evidence and Findings (Continued)

- Missing information greatly affects the prediction because the model is missing it's top predictors.
- For example, customers who work in sales are twice as likely to default on a loan when compared to someone who doesn't work in sales (only made up 1.8% of observations).

VALUE_missing_values_flag	20.738610
DEBTINC_missing_values_flag	14.759061
CLNO_missing_values_flag	3.038353
CLAGE_missing_values_flag	2.606840
JOB_Sales	2.062427
MORTDUE_missing_values_flag	1.556343
REASON_missing_values_flag	1.404438

Logistic regression list of odds in order, representing the impact of each variable on the odds ratio of loan default.

Recommendations

- Be aware of top drivers of default such as a customer's debt to income ratio, which should be less than about 43% at the most and customer's oldest line of credit should be greater than about 14 years. Number of existing credit lines less than 2.5.
- We can use common features in the data to create customer profiles for those less likely to default and use the information for targeted marketing campaigns.
- Ensure customers have thoroughly completed applications for the best prediction of default.
- Data collection-replace delinquent lines of credit and derogatory credit reports information with credit score (which encompases each) to see if it improves model performance.
- Data collection-create more options for job type 'other;' try to keep total job field options to 9 or less (Hospitality, Healthcare, Education).

Implementation Summary

Next Steps:

- Integrate new prediction model into loan approval procedures.
- Review and modify data collection process.
- Conduct regular model maintenance, especially as new data comes in.
- Implement recommendations.

Thank you

•••

Joy Mock
joy23@live.com

Appendix

Model Comparison

Why machine learning?

We use data science techniques to manage and process data for machine learning algorithms that build, train and test models to answer a particular problem about the data. The classification models used on this project are well suited for prediction problems.

During the project, we tested variations of logistic regression and tree based models. The final model is #7.

	Model	Train_Accuracy	Test_Accuracy	Train_Recall	Test_Recall	Train_Precision	Test_Precision
0	LogisticRegression(class_weight={0: 0.2, 1: 0....	0.846896	0.856544	0.777896	0.737903	0.584198	0.633218
1	LinearDiscriminantAnalysis()	0.858641	0.867450	0.670563	0.653226	0.634171	0.692308
2	QuadraticDiscriminantAnalysis()	0.829279	0.827181	0.650372	0.645161	0.557885	0.575540
3	DecisionTreeClassifier()	1.000000	0.853188	1.000000	0.592742	1.000000	0.665158
4	DecisionTreeClassifier()	1.000000	0.853188	1.000000	0.592742	1.000000	0.665158
5	DecisionTreeClassifier(class_weight={0: 0.17, ...	0.850461	0.841443	0.814028	0.737903	0.587423	0.596091
6	(DecisionTreeClassifier(max_features='auto', r...	1.000000	0.901846	1.000000	0.661290	1.000000	0.832487
7	(DecisionTreeClassifier(max_features='auto', m...	0.945050	0.889262	0.969182	0.762097	0.796507	0.721374