AI Bootcamp
Project 2

CREDIT CARD DELINQUENCY PREDICTI0NS

Presented by:
Bo Kimball | Stephen Singletary | JB Graham | Zak Hagberg

Agenda

- *Project Overview*
- *Process*
- *Chosen Models*
- *Imbalanced Data*
- *Future Research*
- *Summary & Results*

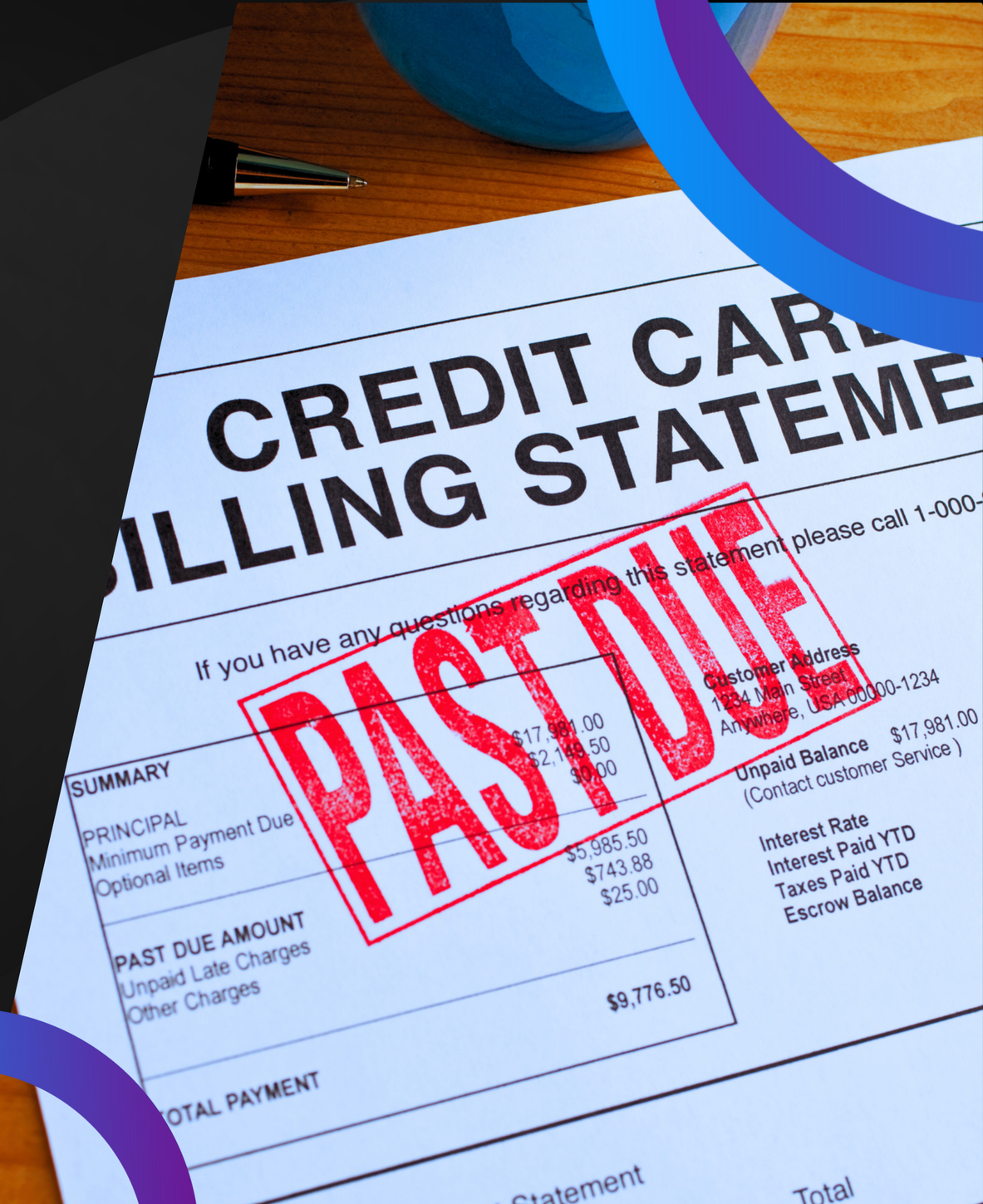
Project Overview

For Project 2, we crushed it by diving into the world of credit card defaults, armed with a killer dataset packed with user attributes, and their balance and payment deets.

We unleashed a barrage of data models into the mix, honing in on those telltale signs that flag a user as a likely defaulter. Our mission? Predict who's going to miss their payments before it even happens. Boom!

[Github Project 2 Repository]

[Kaggle: Credit Card Delinquency Data]



Our Process

01

Identified dataset (URL) that focuses on credit card defaults from Kaggle.com

02

Cleaned data to only include rows and columns that we wanted to leverage for our models

03

Model Exploration & Identification

Logistic regression
Decision Tree
Random Forest
Linear Regression
and more...

04

Each member created built assigned models and group worked through findings together

05

Synthesized findings and presented findings to class

Data Overview

Collection

We spent time exploring numerous data sets on kaggle.com that included credit card fraud, the price of bananas, and tortilla prices. Ultimately we landed on a credit card default dataset for the project

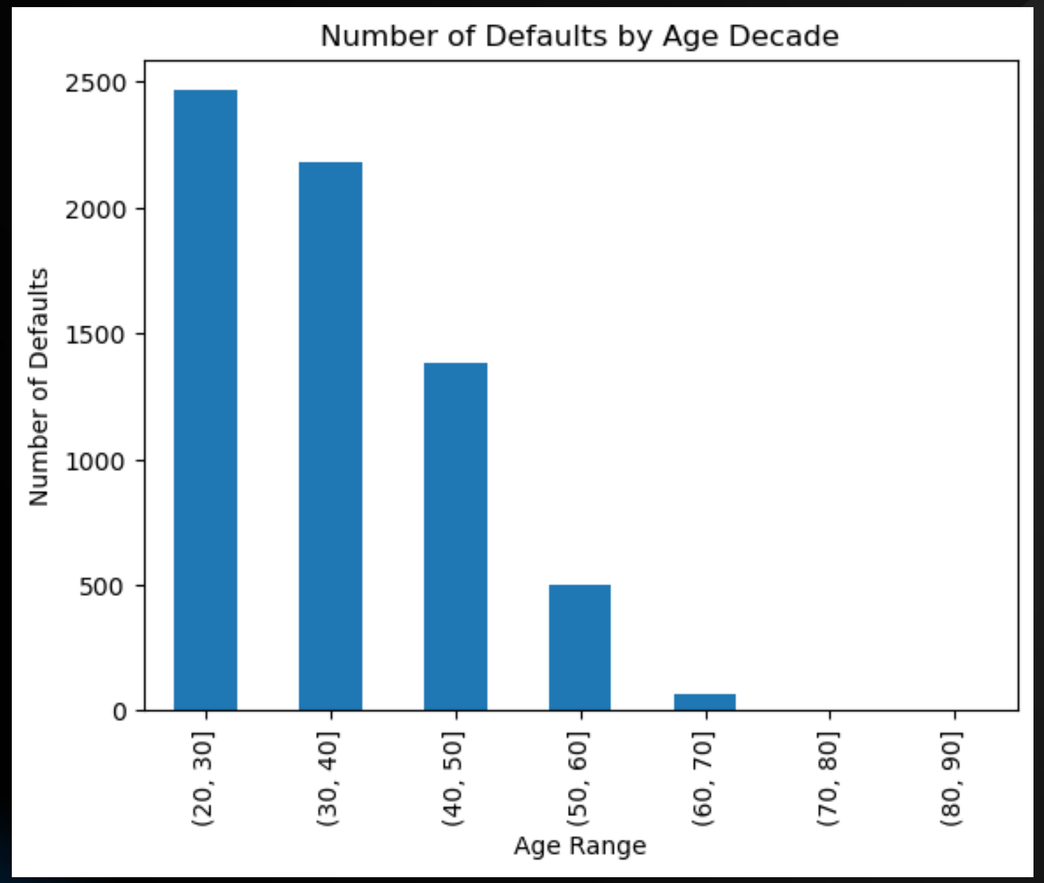
Clean Up

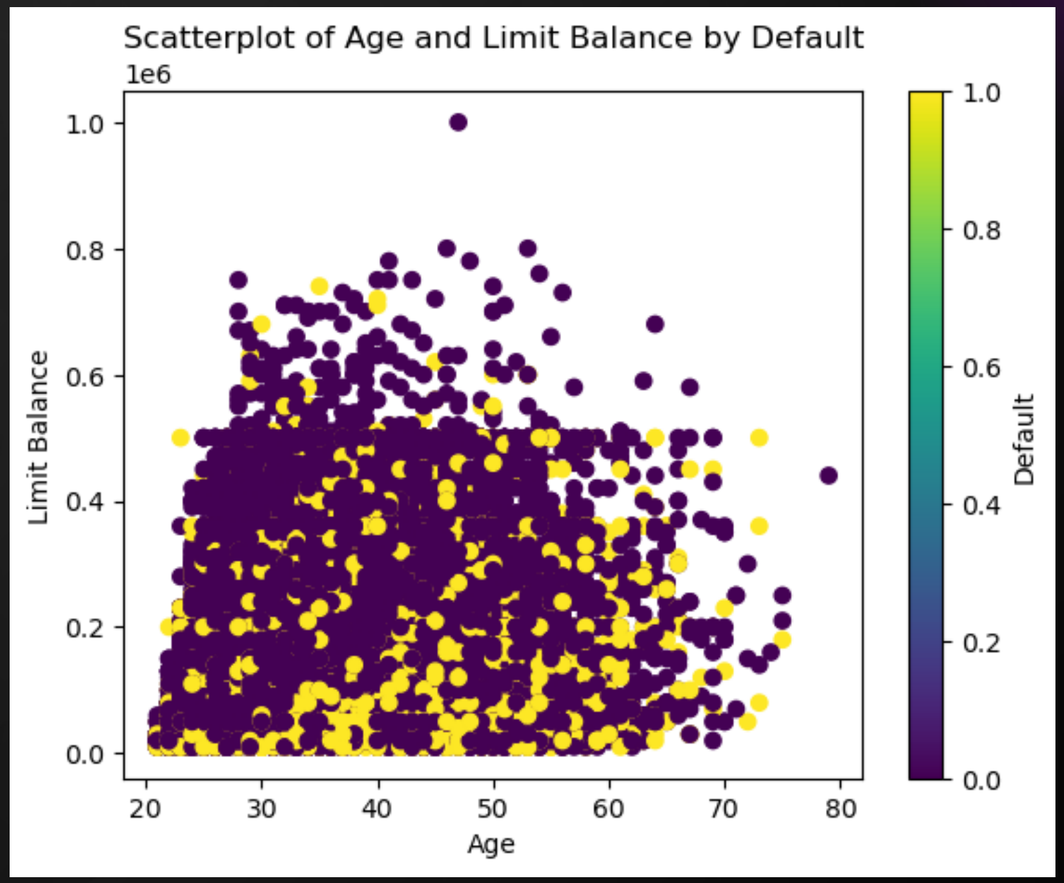
We found inconsistencies in variable columns, such as unexpected values like 0 in the marriage status column which should've only contained 1,2 and 3. We removed rows with unknown values and eliminated the customer ID column to prevent potential model disruptions.

Exploration

We explored the dataset by leveraging supervised learning models which included Decision Tree, Logistic Regression, Extra Trees Classifier, Gradient Boosting, and more

Data Exploration





Random Forest

Model Overview

Random Forest models can be used for both classification and regression tasks. Since we are trying to predict the likelihood of default, we should use a classifier, as the final output is a discrete label/category.

Hypothesis

Random Forest modeling generally provides high accuracy even when using diverse or incomplete datasets. Because it uses random feature selection, it should resist overfitting data as well.

Findings

As predicted, we were able to attain a reasonable degree of accuracy, but it was not precise enough for our needs. Even adjusting the max depth of the trees, the number of estimators, and the minimum number of samples required for a split failed to improve performance.

Supporting Data

0.665

Balanced Accuracy Score

Decision Tree

Model Overview

The Decision Tree model was one of the top performing models with a 72.41% Accuracy Score. Initially this was the highest performing model.

Exploration

We tried a multitude of techniques to get the accuracy score higher. Examples of this include resampling using SMOTE, testing scaled vs. unscaled data and removing a variety of variables.

Findings

Ultimately, this model wasn't performing quite up to our standards. All of the techniques we used could not bring the accuracy score above a 0.7241.

Supporting Data

.7241

Balanced Accuracy Score

Logistic Regression

Model Overview

The Logistic Regression model was one of our weakest models with an ROC AUC Score of 60%, which ran on scaled/cleaned data.

Hypothesis

Initially, we did not have strong confidence in the model due to the hypothesis that the feature set may be too complex.

Findings

The hypothesis proved true when analyzing the score, pointing us to explore more advanced models like the decision trees, random forests and gradient boosting classifers.

Supporting Data

.60

ROC AUC Score

Extra Trees Classifier

Model Overview

The Extra Trees Classfier performed poorly out of the gate, coming in severely overfit with a 99.9% training score. While the model performed much better after the SMOTE technique was applied, it was still not our top performing model with this dataset.

Hypothesis

Initially we assumed the model would perform well due to the fact that it is generally good with noisy or high-dimensional data.

Findings

Overall with this dataset, the Extra Trees Classifier stumbled initially with the imbalanced target data and still showed some weakness with the SMOTE technique due to the reletively high dimensional data.

Supporting Data

0.999
Training Score

0.814
Training Score - SMOTE

0.815
Testing Score

0.820
Testing Score - SMOTE

Adapted Boosting Classifier

Model Overview

The AdaBoost model was deployed with scaled data from the original cleaned dataset.

Hypothesis

With fewer hyperparameters to tune and the model less prone to overfitting, the hypothesis was that the model should perform well.

Findings

Overall, the model performed well and the hypothesis proved to be true. The AdaBoost model improved our accuracy and seemed to handle the imbalanced data much better than others explored.

Supporting Data

0.814

Training Score

0.820

Testing Score

Gradient Boosting

Model Overview

The Gradient Boosting model was deployed with scaled and cleaned data from our dataset.

Hypothesis

The hypothesis for the Gradient Boosting model was similar to the thoughts of deploying the AdaBoost model due to the general principles that it should be less prone to overfitting and handles mixed data types well.

Findings

Overall the Gradient Boost model was our highest performing model with both the Training and Testing Scores, proving our hypothesis true when looking at our dataset and the benefits of the model framework.

Supporting Data

0.823

Training Score

0.826

Testing Score

Further Analysis & Questions

Regularization

Possibly apply Ridge technique to the extra trees model to prevent overfitting and improve the generalization performance of the model.

Neural Networks

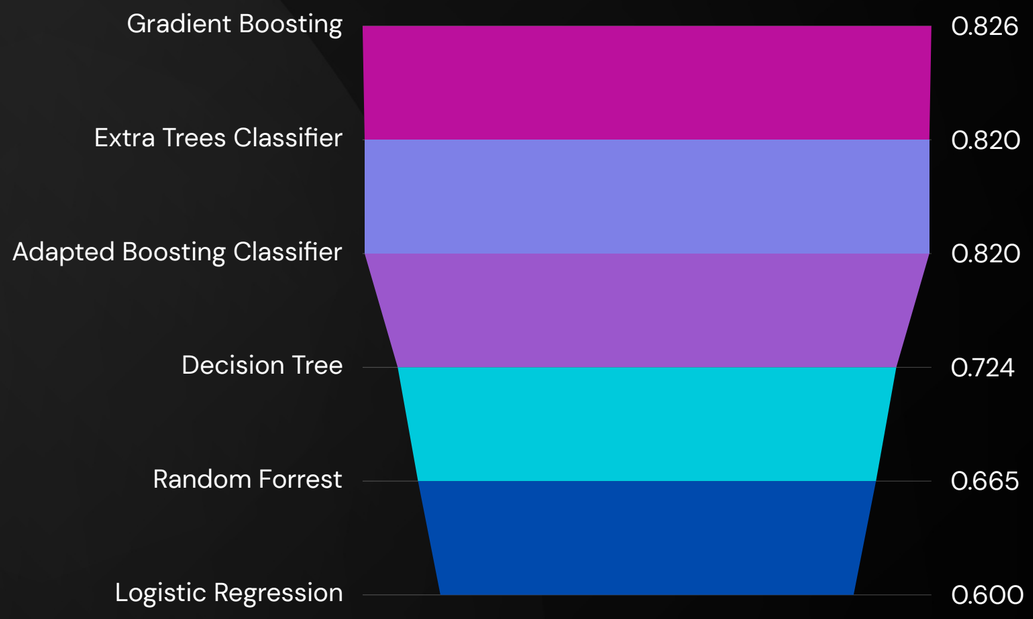
Apply neural networks to more accurately predict who has a high chance of defaulting due to their ability to excel at capturing complex, non-linear relationships in data, enhancing their ability to classify intricate patterns.

New Data

Having fresh, new data that the model has never seen to test our model on in a real life scenario.

Results

In this project, we evaluated several supervised learning models, including Decision Tree, Logistic Regression, Extra Trees Classifier, and Gradient Boosting. Among these, the Gradient Boosting model stood out, achieving the highest performance with a training score of 0.823 and a testing score of 0.826. This indicates that Gradient Boosting is well-suited for this dataset and outperformed other models in terms of classification accuracy.



Conclusion

The results suggest that Gradient Boosting is a powerful technique for classification tasks, offering several benefits. It excels at capturing complex patterns in the data, as evidenced by its superior performance compared to other models. Additionally, Gradient Boosting is robust against overfitting and can handle high-dimensional datasets effectively. Overall, these findings highlight the benefits of using Gradient Boosting for classification tasks, especially when dealing with complex datasets where accurate predictions are crucial.