**Capstone Two - Final Report**

Problem Statement

Utilizing customer segmentation techniques, can Thera Bank create a model that increases conversion rate to 20% by selectively targeting those customers most likely to accept a personal loan?

Context

Thera Bank wants to explore ways of converting its liability customers to personal loan customers. The bank ran a campaign last year for its 5,000 liability customers and had a conversion rate of 9.6%. Rather than target all of its liability customers again, Thera Bank would like to better understand the segmentation of its customers and selectively target those most likely to accept a loan offer. Selectively targeting customers will help to keep campaign costs lower and increase conversion rate.

Criteria for Success

By selectively targeting liability customers most likely to accept, we'll restrict the size of our campaign and, ideally, increase conversion rate to 20%.

Scope of Solution Space

We have all data from the last campaign for all 5,000 customers. This includes demographic information, such as age and income, their relationship with the bank, and their response to the last campaign. We'll focus our model on these existing customers rather than targeting new customers that don't have any affiliation with Thera Bank.

Constraints

The data we're using to build our model is from the last marketing campaign and may not accurately reflect changes at the customer level over the past year (increases in income, their relationship with the bank, etc.). Our assumption, however, is that features most associated with accepting a loan will have explanatory power across different sets of customers.

Stakeholders

Director of Marketing
Director of Mortgage Banking

**Data**

Collection

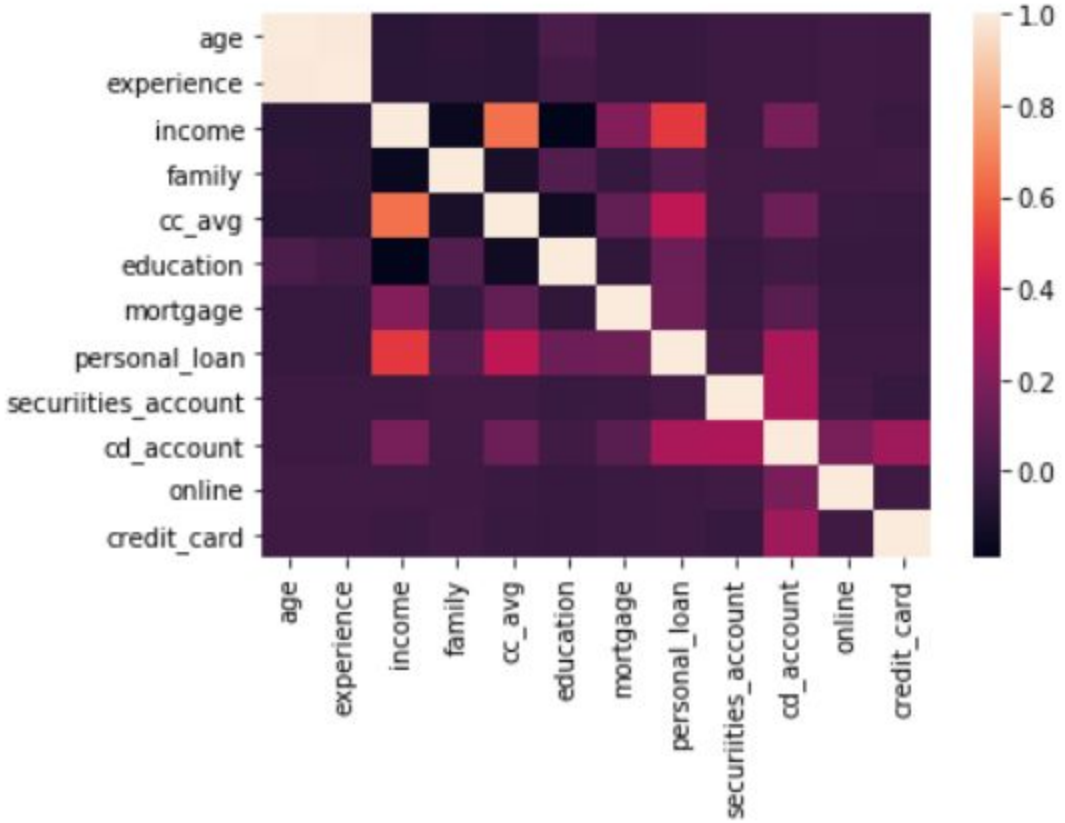
Thera Bank has all of the details from last year's campaign in a CSV file. The file contains data (numeric and categorical) on the 5,000 customers it sent offers to last year and whether or not customers accepted or rejected the loan offer. Features collected include age, number of years of working experience, income, education level, and existing accounts with Thera Bank.

Cleaning

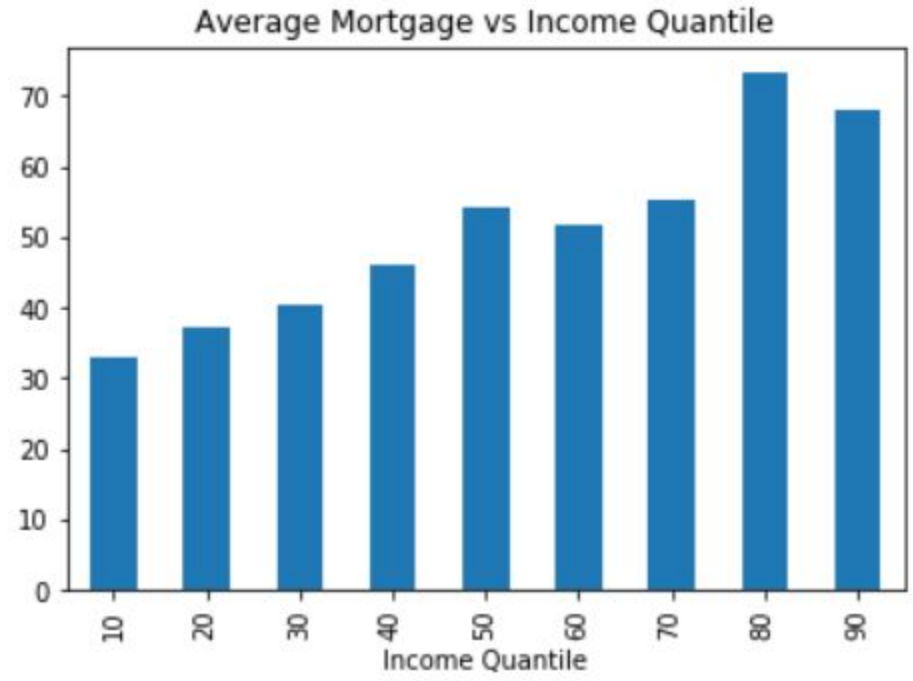
While the data was primarily clean and ready for use, some cleaning was required before proceeding. Zip code had been loaded as an integer, so to prevent future models from interpreting that feature as ordinal, its data type was converted to object type. Some observations in the data had negative years of experience, which is impossible: the minimum number of years anyone could have worked would be 0. We therefore dropped rows where experience was negative.

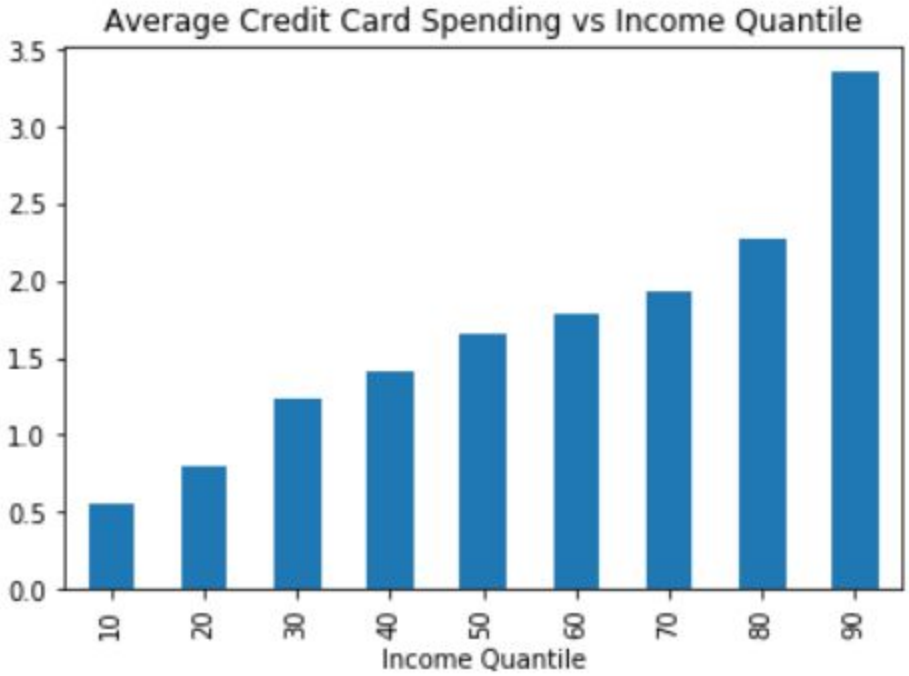
Exploratory Data Analysis

After removing observations with negative experience values, our resulting dataframe 4,948 observations. We used boxplots to visualize the distribution of values for each of the numeric features, and we also used Seaborn pairplot and heatmap to examine the bivariate relationships among numeric features.




The heatmap identified some feature relationships worth exploring. We see strong positive correlation between age and experience, but this is trivial. We also see positive correlation between cc_avg and income, personal_loan and income, as well as mortgage and income. To help visualize these relationships more clearly, we divide income into 10 separate quantiles and calculate average values for each quantile.

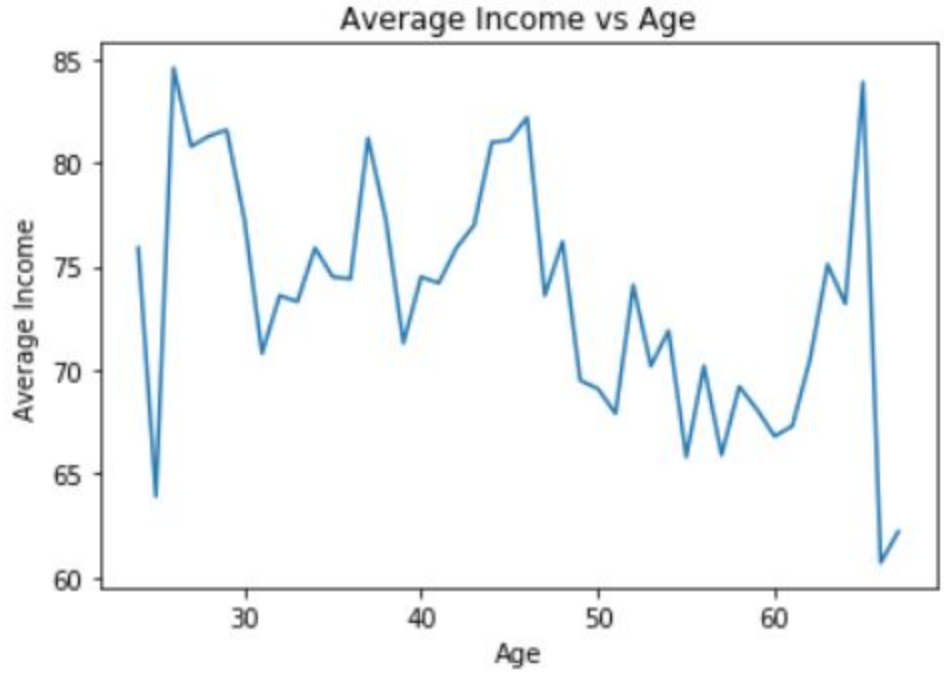


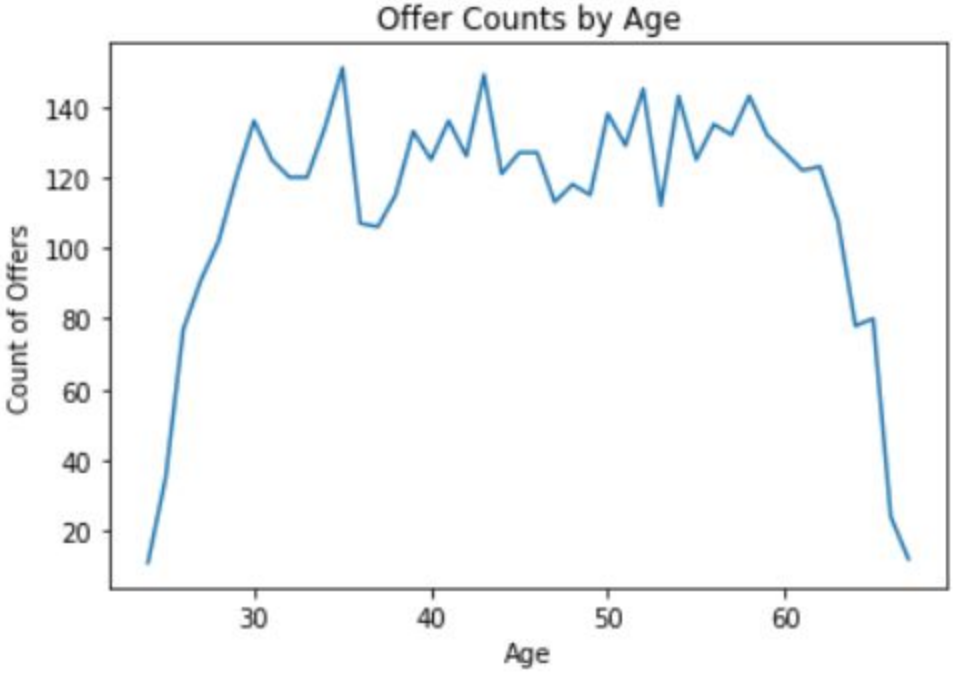




Higher incomes tend to have higher mortgages and credit card spending. Another feature worth exploring was zip code, specifically whether or not certain zip codes experienced significant conversion rates on offers from last year's campaign.

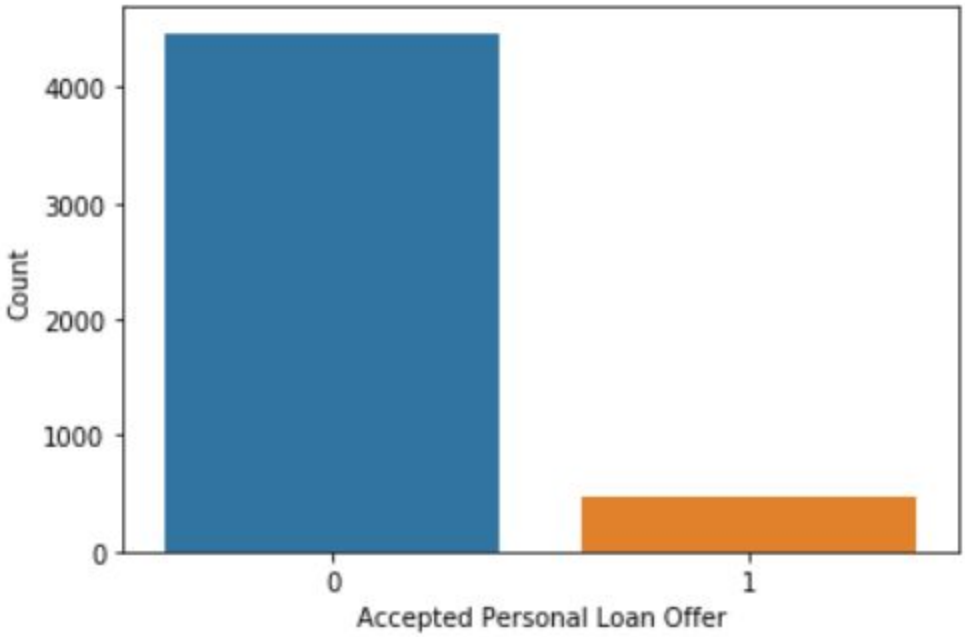
| | zip | loan_offers | loan_acceptance |
|---|---|---|---|
| **0** | 96008 | 3 | 0.67 |
| **1** | 95135 | 3 | 0.67 |
| **2** | 94705 | 4 | 0.50 |
| **3** | 94108 | 4 | 0.50 |
| **4** | 92056 | 6 | 0.50 |
| **5** | 91129 | 2 | 0.50 |
| **6** | 90059 | 4 | 0.50 |
| **7** | 90016 | 2 | 0.50 |
| **8** | 93022 | 5 | 0.40 |
| **9** | 95192 | 3 | 0.33 |

Zip code likely isn't explaining many loan offer conversions, so this can be dropped during modeling. The last relationships we explore in EDA are what the average income by age looks like for bank customers, and the amount of offers the bank made to each age group in last year's campaign.






Before we move into the modeling of our data, we need to understand the class imbalance in our predicted variable, personal_loan.




**Pre-Processing**

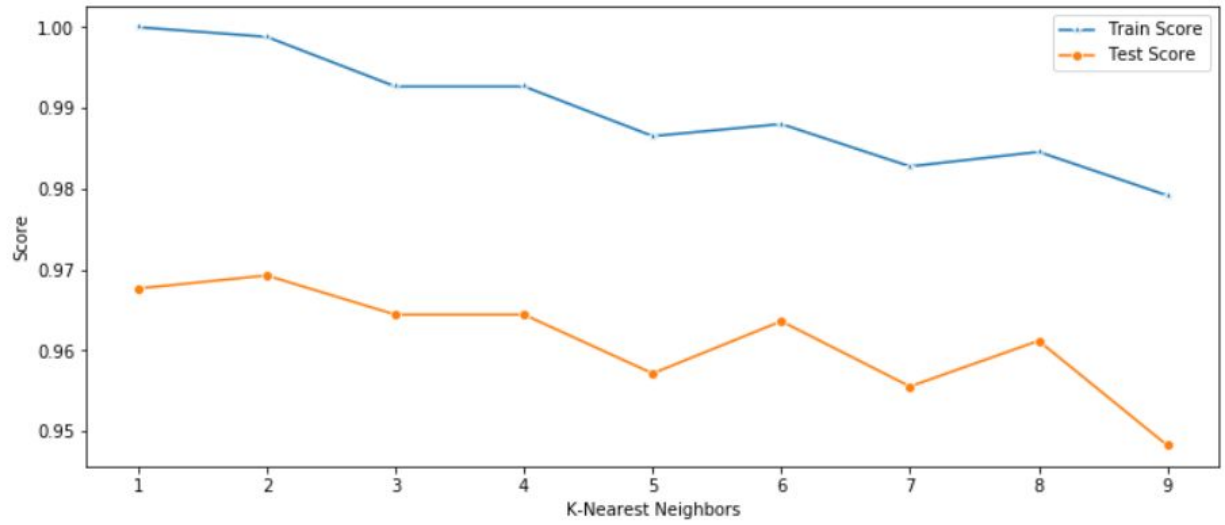
We drop the zip code feature after examining it in EDA. We split the resulting dataframe (4,948 rows, 12 columns) into X and y, where y is our dependent variable (personal_loan), and X are our independent features (the other 11 columns). We use StandardScaler to fit and transform X before splitting our data into training and testing sets with a 75/25 split.

To handle class imbalance in our dependent variable, we use SMOTE to create synthetic samples with accepted loan offers. The training set before SMOTE had 373 observations with accepted loan offers. After SMOTE, this was increased to 3,338 observations, the same number of observations in the set with a rejected loan offer.

**Modeling**

This is a supervised learning classification problem that we model in three different ways: K-Nearest Neighbors (KNN), Logistic Regression, and Gradient Boosting. The primary metric in evaluating these different models is accuracy, though we pay particular attention to the number of True Positives and False Negatives when evaluating the testing set.

In KNN, we see that the optimal number of neighbors is 2, and our model performs with 96.9% accuracy.




| | 0 | 1 |
|---|---|---|
| 0 | 1118 | 26 |
| 1 | 12 | 81 |

The drawback to the KNN model can be seen in the confusion matrix above. While we do a very strong job of identifying those that will reject the personal loan (1,118 of 1,130), we miss out on a lot of potential loan offers being accepted by only predicting 81 of 107 accepted loan offers. After hyperparameter tuning the model, the optimal KNN remains 2, and our results stay the same.

In the baseline Logistic Regression model, we do a much  better job of correctly identifying accepted loan offers.

| | 0 | 1 |
|---|---|---|
| 0 | 1031 | 11 |
| 1 | 99 | 96 |

Here, we correctly identified 96 of 107 accepted loan offers. However, we also misclassify quite a few rejected loan offers as predicted to be accepted. To reduce the number of false positives (99), we turn our attention to hyperparameter tuning and cross validation. After establishing the optimal regularization parameter and refitting the model, our model improves in accuracy, but at the cost of reduction in recall.

| | 0 | 1 |
|---|---|---|
| 0 | 1120 | 29 |
| 1 | 10 | 78 |

We see a significant increase in correctly identifying rejected loan offers, but we'd now only identify 78 accepted loan offers versus the 96 our baseline identified.

The last model we develop is gradient boosting. The baseline performs very strongly, identifying 6 more accepted loan offers than our baseline logistic regression, and also correctly classifying 81 more rejected offers.

| | 0 | 1 |
|---|---|---|
| 0 | 1112 | 5 |
| 1 | 18 | 102 |

Similar to tuning the Logistic Regression model, when we tune the gradient boosting model, we see an overall increase in accuracy (much more modest given the model's already strong performance) at the expense of identifying accepted personal loans.

| | 0 | 1 |
|---|---|---|
| 0 | 1117 | 8 |
| 1 | 13 | 99 |

The baseline gradient boosting model is the model we'd select for our next marketing campaign. Based on the results from our testing set, we could expect a loan offer conversion rate of approximately 85% (102/120).

**Conclusions**

We set out to increase conversion rate for our marketing campaign from 9.6% to 20%. Through gradient boosting, our expected conversion rate is approximately 85%. Thera Bank stands to benefit from this in a couple of ways. First, by only targeting those customers most likely to accept a personal loan, Thera Bank significantly reduces the cost of the campaign and increases return on investment. Secondly, Thera Bank maintains a positive image with those customers unlikely to accept an offer by not sending them offers they'll find irrelevant.

It's worth noting that the data for this exercise was the data collected from last year's campaign. There are likely to be changes that occur year-over-year that impact individual customers, so it's unlikely training/testing data will perform exactly the same this year. Regardless, Thera Bank can continue to promote similar machine learning exercises for customer segmentation in order to grow its business.