Final Group Project
ModernDay Enrollment

Dr. David Anderson

Jennie Franco

October 22, 2022

Contents

1 | Abstract

ModernDay private school seeks to improve its efforts at acquiring new students. The school serves children in preschool through grade 12 and wants to promote its programs to households most likely to enroll. ModernDay hired a team of Villanova analysts to provide a model that would meet their requirements. For this study the team utilized the CRISP-DM process for analyzing and modeling the data. Eight different models were run and tested over the span of two to three weeks. The team found that the best model with an accuracy of 92% could potentially generate a profit of about $288,000 or more. The next sections detail the steps involved in the teams data mining and modeling process.

2 | Introduction

The goal of this study is to improve ModernDay's enrollment program. Specifically, marketing to households with college educated parents earning over $100K, with children under 18 years old.

3 | Data Mining Process

In this section we explain the steps we took in processing the household datasets. We analyzed the data by looking at its properties like quantity of records, data formats, identifying relationships among the data, and validating the data's quality. First, we detected several inconsistencies in our data via the boxplots shown in Figure 3.1.

Table 3.1: Data Overview

Data	Observations	Variables	Missing Values
Household 1	10,000	33	4438
Household 2	10,000	33	4356
Household 3	8,000	33	3629

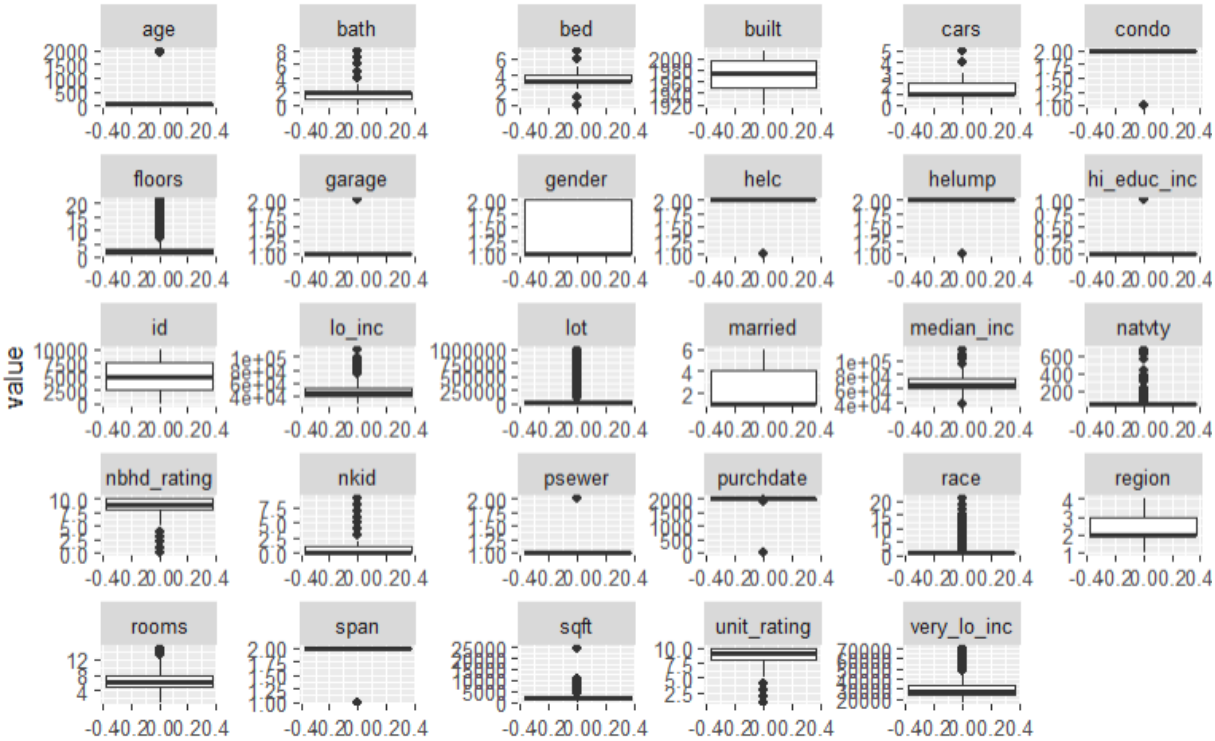
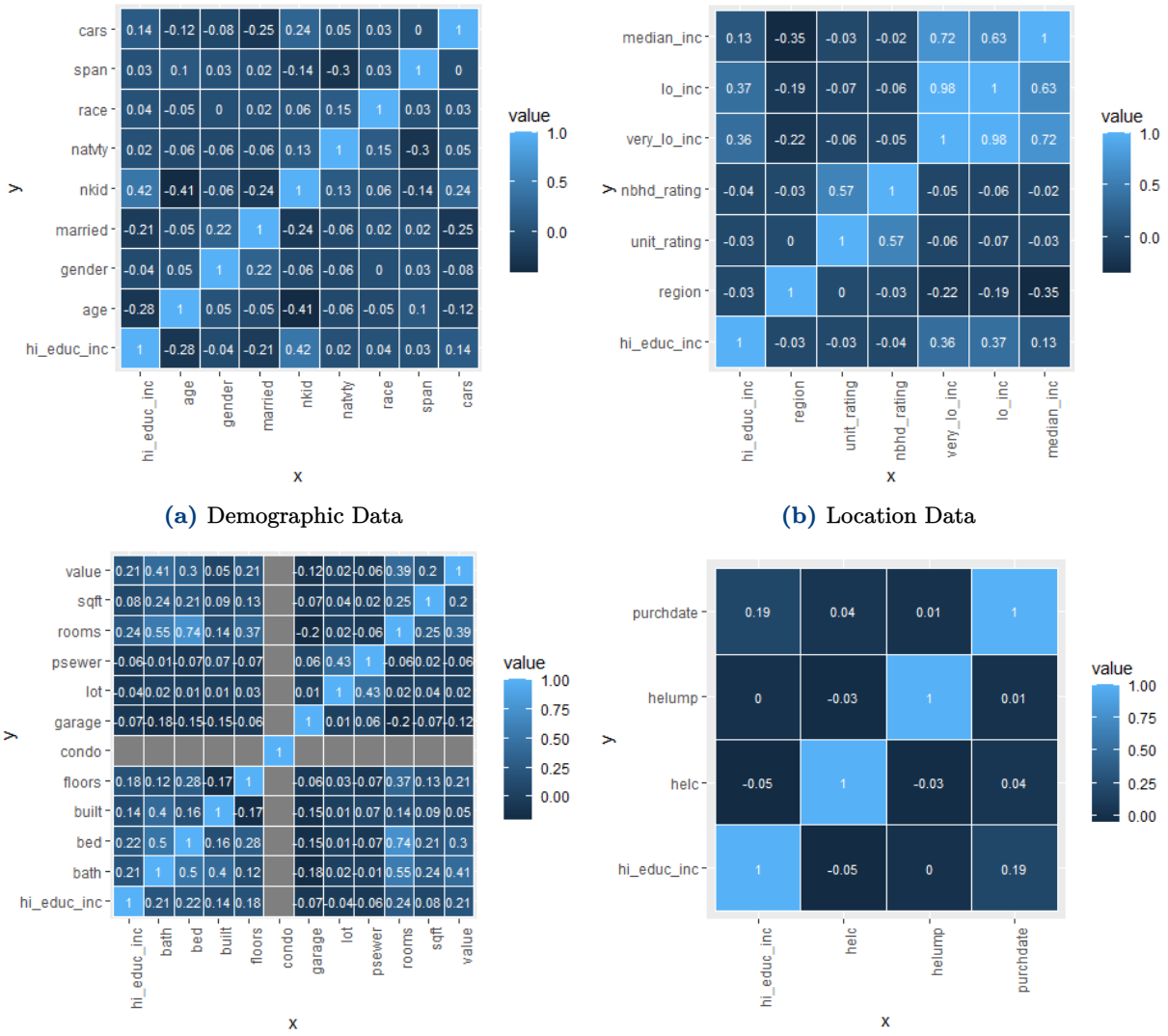


Figure 3.1: Box Plots of Joined Household Data

- age - seems to have some outliers
- purchdate - also has outliers
- lot of missing values

3.1 | Visualizations

We proceed to gain a further understanding of the data through simple visualizations. The correlation plots shown in Figure 3.2 reveal that there is a strong positive link between *very low income* and *low income*. To avoid multicollinearity, we will have to delete one of these properties.



(a) Demographic Data

(b) Location Data

(c) Dwelling Data

(d) Mortgage Data

Figure 3.2: Correlation Plots

Table 3.2: Variables with High Correlation.

Var1	Var 2	Frequency
very low income	lwo income	0.979
bed	rooms	0.741
very low income	median income	0.737
nkid	low income	0.653
nkid	hi educ inc	0.412

The distribution of our predictor variable (hi educ inc) was next examined in Figure 3.3. It is clearly evident that 85% of the data is in one class and the remaining 15% is in another class. In this case, we are focused in accurately predicting the true positives (class 1) and reducing the false positives. If unseen data does not have the same distribution as the data we trained on, it could introduce bias into our model. To handle the class imbalance we reshaped our data.

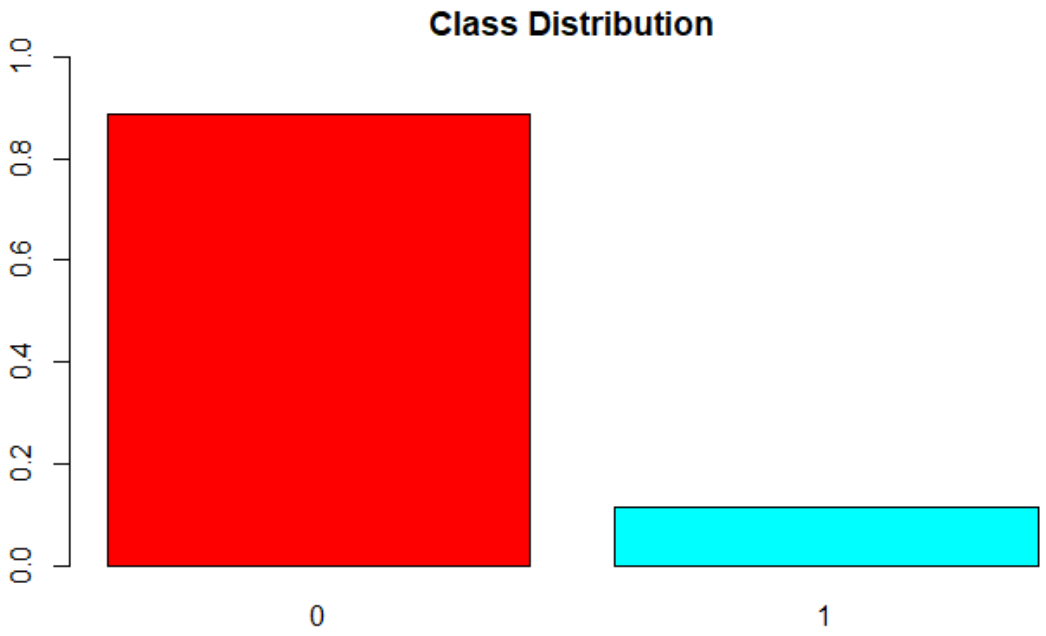


Figure 3.3: Class Distribution Plot

3.2 | Data Cleaning

We move on to clean and prepare the final dataset(s) for modeling. We had to address both the outliers and missing values of our joined data sets. Rather than deleting the rows altogether, we transformed the outliers to NAs. This is done to preserve the remaining feature data.

We then dealt with the missing values. Missing values can cause bias and affect the efficiency of how our models perform. So, here we explored methods to handle that. We can deal with incomplete data in many different ways, for example, pairwise or listwise deletion. In our case we wanted to preserve a bigger sample size, so more sophisticated methods such as missing data imputation were applied.

3.3 | Data Imputation by Classification & Regression trees

We opted to use classification and regression trees to fill in the missing data. The mice package in R is used to run the back-end algorithm. According to the R documentation, the process is as follows:

1. Use recursive partitioning to fit a classification or regression tree;
2. Determine the terminal node for each ymis based on the fitted tree;
3. Make a random draw among the member in the node, and take the observed value from that draw as the imputation

The final step in our data wrangling was to drop highly correlated variables. Recall that very low income and low income had a correlation coefficient of 0.98, thus the team chose to drop the low income feature.

4 | Model Deployment

After running the mice package to take care of the missing values and dropping unnecessary variables, we proceeded to prep our data for modeling. We divided the data in an 80/20 split. Then, using the training data, we ran five models; a classification tree, forest tree, ada boost, xgboost, and xgboost Tuned.

Although xgBoost handles class imbalance well, the team decided to run a few more models to see how they would perform using popular techniques. According to the R-Blogger site below are a few methods to handle class imbalance:

- Over-sampling: randomly replicates instances in the minority class
- Under-sampling: randomly removes instances in the majority class
- Synthetic minority sampling technique (SMOTE): down samples the majority class and synthesizes new minority instance by interpolating between existing ones

4.1 | Model Evaluation

We then put our three extra models to the test. The ROC curves and AUC values are shown in Figure 4.1 and Table 4.1. Furthermore, for each of our models, we run confusion matrices to predict the total profit.

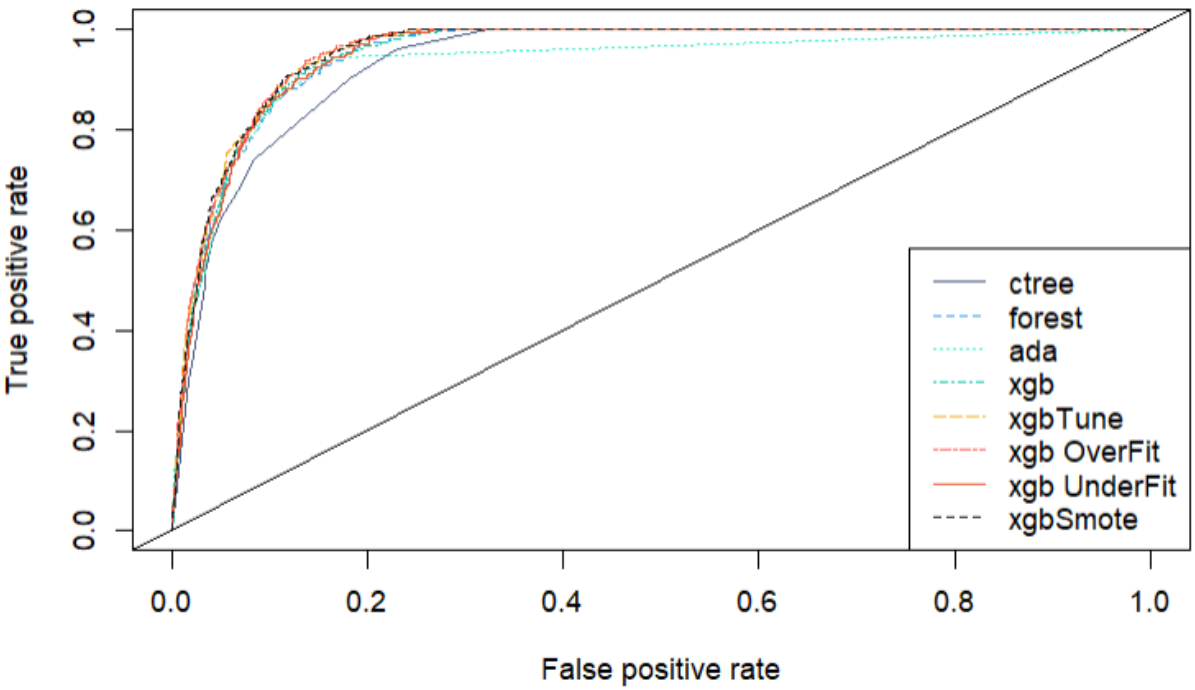


Figure 4.1: ROC Curves

4.1.1 | Total Profit Predictions

The predicted profits of our test data using a cutoff value of 0.4.

Predicted Profit

Table 4.1: Confusion matrix & profit for all models

Model	TP	TN	FP	FN	Revenue	Cost	Profit	Accuracy
Class Tree	240	3413	227	120	360,000	233,500	$126,500	0.91
Forest Tree	313	3358	154	175	469,500	233,500	$236,000	0.92
Ada Boost	303	3362	164	171	454,500	233,500	$221,000	0.92
Xgb	311	3355	156	178	466,500	233,500	$233,000	0.92
Xgb Tuned	325	3350	142	183	487,500	233,500	$254,000	0.91
Over	447	2973	20	560	670,500	233,500	$437,000	0.84
Under	443	2908	24	625	664,500	233,500	$431,000	0.84
Smote	348	2890	119	221	522,000	233,500	$288,500	0.92

We then ran our models via the leadership board testing on household 2 data. The top models were xgbTuned and xgbTuned SMOTE. Because SMOTE performed slightly better we use this model to run our predictions on household 3 data.

5 | Results

ModernDay private school would benefit substantially from running the selected XgBoost Smote model to predict the households to market to, which could generate a profit of about $280,000 with a model accuracy of 92%.

5.1 | Reference Links

- Mice Package for Data Imputation
- Handling Class Imbalance
- CRISP-DM