Telecommunications

Predicting Customer Engagement

Modern marketing campaigns rely heavily on customer relationship management. This approach is integral in the telecommunications sector where massive profits hang in the balance. Developing a loyal customer base is imperative in order to remain competitive in this industry that is dominated by giant companies like AT&T and Verizon. Telecommunications companies generate big amounts of data from customer's activities on the company's internet and mobile networks. With systematic cleaning, analyses, and modeling of these data, customers who are more open to purchasing new or additional products or services can be identified. Accordingly, here I aim to build a model to predict the customers most likely to respond positively to upsell proposals to buy additional products. The client is a telecommunications company, like AT&T.

The dataset used here is real customer marketing data, that has been anonymized, from the French telecom company Orange. These data were previously used in a KDD cup competition. The available target data are a binary indicator of whether the customer is open to upsell proposals and will be referred to as the upsell target. 50,000 customers are included in the dataset with a slew of numerical and categorical data. The dependent predictor variables are not labeled according to what they represent, which places a higher reliance on quantitative feature selection and engineering methods.

Data Wrangling

The data are stored in tab-delimited text files, so I loaded them into a pandas dataframe using the pandas "read_table" function. I discovered the upsell target variable was imbalanced, which will be accounted for in each machine learning algorithm by having scikit-learn assign a balanced weight

to the positive and negative outcomes. There are 190 numerical (float) variables and 40 categorical variables. Initial exploration of the data revealed that the matrix is quite sparse. The numerical variables are the sparsest, with only 42 numerical features that have values for at least 10,000 customers. Comparably, 32 categorical features have at least 10,000 values. I focused first on the numerical features where the demands on missing value imputation was high.

After normalizing the features with their mean and standard deviation, I initially imputed a large negative value (-99999) for the missing cells. I then trained a basic random forest algorithm on the full dataset to obtain an initial measure of how informative the features were for predicting upsell. Based on the random forest feature importances, only 40 features offered meaningful predictive information that the random forest could use. Thus, more complex data imputation methods were needed.

As a result, I decided to test three additional imputation approaches. I found a popular library on github, called “fancyimpute,” with modules for computing some of the most effective advanced imputation methods. A brief description of the data imputation techniques can be found below.

- `SoftImpute`: Matrix completion by iterative soft thresholding of SVD decompositions. Inspired by the softImpute package for R, which is based on Spectral Regularization Algorithms for Learning Large Incomplete Matrices by Mazumder et. al.
- `IterativeSVD`: Matrix completion by iterative low-rank SVD decomposition. Should be similar to SVDimpute from Missing value estimation methods for DNA microarrays by Troyanskaya et. al.
- `MatrixFactorization`: Direct factorization of the incomplete matrix into low-rank U and V, with an L1 sparsity penalty on the elements of

> U and an L2 penalty on the elements of V. Solved by gradient descent.

I wrote a custom function, “testImputers,” to assess which method best represented the data in order to maximize upsell predictive power in a random forest algorithm. To reduce overfitting and optimize generalizability, I trained and tested a 500 tree random forest with 10-fold cross-validation (CV). Since my goal is to obtain accurate probabilities for each customer (or sample), the performance metric used was the negative log loss.

The algorithm trained on the data imputed with the softimpute technique produced the highest mean CV negative log loss of -0.47. However, this approached also produced the highest variability in performance across CV folds, suggesting softimpute worked well for some customers but poorly for others. This hypothesis and more will be explored in future analyses.

Future Analyses

In these future explorations, I will train and test a variety of algorithms covering naive bayes, random forests, SVM, gradient boosting, and neural networks, using 10-fold cross validation on the most recent season of the data. Depending on the performance of the individual algorithms, I may ensemble a diverse set of these algorithms using blended stacking in order to maximize predictive power. Additionally, I will explore several scikit-learn feature selection algorithms in aims of narrowing down the features to those that are maximally relevant and minimally redundant. The end model will provide a score for each customer that indicates the likelihood that one will positively respond to upsell attempts, which the telecom company can use to optimize future marketing campaigns.