Machine Learning Engineer Nanodegree

Capstone Proposal

Manuel Seeger July 13th, 2019

Proposal

Domain Background

Probably more than others, players in the telco industry suffer from "churn": The move of a customer from one provider to another. In developed markets, the cost of switching providers is increasingly low for the consumer. Reasons include regulation, like transferable phone numbers and an increasingly level playing field in terms of network coverage. At the same time, overall industry growth in terms of number of addressable consumers is slowing down due to high market saturation and maturation. Differentiators like network coverage or exclusive hardware offers are becoming smaller or disappear altogether.

It costs upward of 5 times as much to acquire a new customer than it costs to retain an existing one, according to InvestP. Retaining their customer base by reducing churn is thus a very important part of a telco's customer relationship strategy. This study will seek to help predicting churn to support a churn prevention campaign.

While this study will work on data from the telecom industry, predicting and managing churn is a shared problem across most consumer-facing industry. The methodology applied in this project are expected to be transferable to similar problems cross industry, as long suitable data exist.

Personal Motivation

As a consultant for customer relationship management software, smartly segmenting customers for targeted campaigns is a recurring problem to be solved. Where data-driven prediction and segmentation tools are missing, targeting is done by intuition of marketing professionals and/or based on naive heuristics. I hope to draw insights on how to improve campaign targeting from this project.

Problem Statement

Even if preventing churn is considered cheaper customer-by-customer than acquiring a new customer, it comes at a cost. A customer retention campaign might include costly incentives for customers to reconsider churning and extend their contracts. It is most important to target these incentives at the customers most likely to churn, and not "waste" a costly incentive on customers that would have stayed with the provider in the first place.

Predicting which customers are likely to churn, and which are likely to stay thus becomes a critical component of the customer base segmentation for a retention campaign. Neslin, Gupta, et al. estimate that an increase in accuracy of churn prediction of just 0.1 percent can translate to an additional 100,000 USD of profit captured for a mid-size mobile carrier.

This project will develop different models for customer churn prediction on a customer dataset, and select the model with the best performance on predicting if a customer is likely to churn.

In machine learning terms, predicting churn is a binary classification problem. There is one target variable, Churn, which can have 1 of 2 labels: Yes - the customer is leaving this month; No - The customer is staying with the provider. Our machine learning problem solution will take information about the customer as input (such as financial, contractual, or personal), and output the probabilities that a customer belongs to the churner and the non-churner group.

Datasets and Inputs

The churn prediction will be done on a dataset of Telco customers. The dataset and its input variables is described in detail in

the appendix of V. Umayaparvathi1, K. Iyakutt in A Survey on Customer Churn Prediction in Telecom Industry: Datasets, Methods and Metrics. https://www.irjet.net/archives/V3/i4/IRJET-V3I4213.

The dataset was originally donated by an (anonymous) US carrier and published by the Fuqua School of Business at Duke University as part of a machine learning competition. The data was obtained from Kaggle [https://www.kaggle.com/jpacse/datasets-for-churn-telecom/downloads/datasets-for-churn-telecom.zip/2]

The dataset consists of 58 variables describing a customer in terms of demography, payment history, credit score, service usage pattern, and past interactions with the customer. The target variable is Churn, which denotes if the customer has churned within the past month. The training data has been selected such that churners are overrepresented compared to non-churners, to enable efficient analysis and learning. The training set has 51047 records, of which 14711 are churners, making this an un-balanced dataset.

Here are some sample records from the training set:

CustomerID	Churn	MonthlyRevenue	MonthlyMinutes	TotalRecurringCharge	...	Occupation	MaritalStatus
3053510	No	50.74	16.99	6.0	...	Retired	Yes
3302722	Yes	34.99	313.0	45.0	...	Other	Unknown
3071898	No	85.92	620.0	50.0	...	Professional	No

There is a separate test dataset which is a representative sample of customers in a given month. Labels for this data set have not been published. The project will thus apply a train-test split on the training data set to test model performance.

Solution Statement

Predicting Churn from the proposed dataset presents a supervised classification problem. The solution will apply the following supervised classification algorithms on the data to find the model that best predicts churn for the features selected:

- Logistic Regression
- Decision Trees (in the form of random forest)
- Neural Nets
- Boosted decision trees (in the form of XGBoost)

The input variables will be analyzed for their power in predicting the target variable Churn and pre-processed where required. The different models will then be trained on the prepared input data.

Performance metrics of the models will be collected to determine the best model for churn prediction. The model best performing under the evaluation metric discussed further below will be selected as the solution.

Benchmark Model

Each solution candidate model will be benchmarked against a baseline model. For a baseline model, the data will be split into churners and non-churners by a kNN-classifier with k=2. A kNN classifier is unlikely to capture complex interactions well compared to the candidate models, but will give a good baseline for performance which the candidate models can improve on.

To keep benchmarking consistent, the kNN-classifier will be run on the same training data as the solution models, after the data has been cleaned for obvious quality flaws (like missing values, etc), but before the full pre-processing pipeline is built. Within one benchmarking run, the train and test split, as well as other functionality based on pseudo-randomness will be kept constant and reproducible by explicitly seeding the implementations random generator.

Evaluation Metrics

We look at our model's performance in terms of the following confusion matrix:

-	Predicted churn	Predicted Retention
Churning customer	true positive	false negative
Retained customer	false positive	true negative

Our model's predictions should have as few false negatives as possible, as those would be customers we likely lose because we don't target them with a retention campaign. We are not as much concerned with false positives, as a retention campaign is considered much less costly than a (re-)acquisition campaign. In other words, we can afford some false positives, as long as we miss as few as possible true churning customers.

In a real-world example, the distribution of churners and non-churners is highly skewed. Neslin, Gupta, et al. estimate around 2% of customers churn monthly for a typical US carrier.

We are thus mostly concerned in keeping false negatives low, and in catching as many churning customers as possible. In terms of performance metrics, we will model this evaluation as a high-recall f-beta score:

fβ-score with β = 2

This will be the main metric to evaluate our models' performance. During the project the value for β will be reviewed and fine tuned.

Project Design

Initial Data Analysis

In a first step, the dataset will be viewed for quality. Descriptive statistics on the data features will be collected, columns checked for missing values, normality, and skew. Data will be cleaned and transformations applied where necessary.

Benchmark model

After data has been cleaned for quality shortcomings, the benchmark kNN classifier will be applied to the data. The performance metric of the benchmark classifier will be the baseline for the solution model to improve on.

Exploratory data analysis

In this section, the data will be reviewed for fitness for the prediction task, mainly using visualization. The features will be tested for correlation and how well they appear to match assumptions based on industry knowledge.

Features will be semantically grouped where applicable. Apparent important of the features for the predicion task will be discussed. Features unlikely to contribute to the predictive power of a model will be removed.

Data preprocessing

Based on the results of the initial and exploratory data analysis, the data will be pre-processed to be better usable for model building. Ranges will be normalized, outliers will be removed where necessary, categorical variables will be one-hot encoded. A pipelines will be built to feed the pre-processed data into the models.

Model building

Different algorithms will be implemented to give an overview which algorithm performs best on the problem of predicting churn. Algorithms of the following types will be implemented and tested on the problem: Logistic Regression, Tree/forest

classifier, boosted classifier, as well as a MLP/neural net classifier. The proposed algorithms will be implemented in different models and trained on a training subset of the dataset.

Where the algorithm takes hyper parameters as input, grid search will be used with k-fold cross validation to determine the most fitting hyper parameters, per model. Examination of the learning curves of each algorithm will be done to spot possible over- or under fitting.

Model performance evaluation

The models fitted on the data will be evaluated based on the f-beta score discussed above. Additionally, the AUC-ROC curves of the models will be examined to see how the models compare against each other. A best performance model will be selected. The performance of all models will be compared against the benchmark model. Suitability of a machine learning predictor over a naive heuristic will be discussed.

Outlook

The proposed model for churn prediction will be put into context of the domain background. Possible improvements to the problem solution will be discussed. A strategy for implementing a predictor in a productive environment will be proposed.