Development of a churn prevention solution for a global management consulting firm

Central European University, Department of Economics and Business

Capstone Project summary

2022.06.10

Daniel Gyebnar

Table of Contents

1. Client introduction

My client for my capstone project was a global management consulting firm with a presence around the globe. Top management consulting firms are more and more focusing on building up their data analytics capabilities and offering stand-alone data analytics project to clients, as well as enhancing traditional management consulting projects with advanced analytics. These capabilities are starting to become a key differentiator for an increasing number of management consulting projects in our data driven world. My client's Data Analytics Team is aiming to develop off-the-shelf Software as a Service solutions that can be easily custom tailored to clients in the future. One of such tools to be developed is a churn prevention solution that my client plans to offer to subscription based, B2C focused service providers. The churn prevention solution would help the service providers identify potential churners and help focus resources on retaining customers.

2. Project description

Subscription based businesses have always struggled with churn prevention and have tried to decrease churn rates ever since. Offering discounts and bonuses to gain customer loyalty have been one of the main tools to prevent switching to other service providers. However, identifying churning clients and efficiently allocating resources to retain them is not an easy task, as depending on the industry only a small, 2-10% fraction of clients tend to churn yearly. My client aims to offer the churn prevention solution to help service providers identify potential churners and allow them to focus their resources on clients of top priority. This would create a direct positive P&L impact through retaining revenue and even decreasing retention costs.

The aim of my capstone project was to support my client in developing a prototype of the Churn prevention solution. The aim of the prototype was twofold. First, to create the environment and the workflow already setup with the predicting model and the countermeasures implemented, and results visualized in a dashboard, making it easy to roll-out and custom tailor for various clients in the future. Second, to demonstrate a Proof of Concept that can be used for business development, presenting how the solution would work and showcase a dashboard that visualizes the predictions and also demonstrates the effects of potential countermeasures applied.

The delivery goal of the project was to have a prototype ready that can preprocess the client data, fit the model, present predictions on potential churners on a dashboard, and also allows to apply countermeasures by calculating their effects and visualizes their impact, including costs and revenue gains. Therefore, the prototype had to be built with keeping the productional aspects of the software solution as the top priority, meaning that all functionalities that can be generalized should be called with generic functions, and create a fixed pipeline making the solution easily applicable to any client data for future projects, limiting required coding and software engineering capacities during roll-out.

3. Project summary

During the project I had the full responsibility of delivering the data science and analytics related workpackages to the operational team, who were responsible for the software engineering aspects of the development. The project was divided into two phases. Phase 1 (Research and setup) included preliminary research where I was responsible summarizing existing literature on churn prediction with insights into features needed, feature engineering, and most common modeling techniques applied, and plan the model according to these best practices. Phase 2 (Model development) included the modeling and development process to create the working prototype, where I was responsible for workpackages executing the data preprocessing, feature engineering, clustering and feature selection, the core model fitting and evaluation including a hyperparameter tuning tool, the implementation of potential countermeasures and their effects on churners, and finally the plotting of key charts that were used for the interactive dashboard.

4. Key results per workpackage delivered

I have delivered 6 workpackages which covered the data analytics elements of the churn prevention solution. The result of my deliverables was a working prototype that executed basic preprocessing and feature engineering, fitted the model, plotted the results on a dashboard, and dynamically visualized the effect of measures applied. The workpackages reduce the data science and coding knowledge needed for future live roll-outs, leaving only the very specific tasks and custom tailoring to be performed on-site.

4.1 Research on best practices for churn prevention

I delivered a literature review that summarizes the key insights of 10+ publications. My research gave valuable insights into best practices to follow, helped plan and structure the workpackages, and reassured my client on the right content to implement.

During my research I examined the type of variables used and deemed as most important and demonstrated the advantages and disadvantages of using static vs. time dependent data. My research pinpointed the most important and common features to include. the importance of certain elements of data preprocessing such as SMOTE, clustering of customers before fitting models, section of features using CART and feature importance metrics, and furthermore the type of models usually applied with the best results. Following the more common approach of using static data for simplicity was a key outcome of the research.

4.2 Data preparation and feature engineering

I delivered a workpackage that takes the raw data as an input and creates a data cleaning workflow with most preprocessing elements automatized, creates customer clusters, supports feature selection using CART and gives plots and feedback on results of each process element.

Th workpackage contains generic functions that take care of most common data preprocessing tasks: finds and helps deal with missing values, alerts for alterations between actual and expected data types per variable, performs most common cleaning tasks, plots distributions, implements one hot encoding, winsorizing and normalizing, and creates a variable that flags zero values above a given threshold. Furthermore, it includes more advanced tools as well: It performs KNN with PCA analysis to cluster customers into arbitrary set segments, offering the possibility to train separate models for each segment. It also fits a basic CART model to calculate permutated feature importance, and plots AUC as a function of the number of predictors, supporting decisions to reduce unnecessary complexity and save calculation time.

4.3 Analysis and evaluation

I delivered a workpackage that takes the cleaned data as an input and prepares the training data, offers support for hyperparameter tuning and feature selection, fits and evaluates model and gives automatic feedback, and implements final prediction and classification into the dataframe.

The workpackage performs a train-test split, uses SMOTE to resolve the rare event problem by balancing the training dataset, and also contains a hyperparameter tuning tool. This tool helps find the right range of preliminary grid parameters for the GridSearchCV algorithm within the solution, plotting the evaluation metric (AUC) as a function of a given grid parameter when all the other parameters are set as the optimal parameter based on the grid search algorithm. The workpackage evaluates the model using 5-fold cross validation, refits for the best parameters, evaluates on the holdout set, and implements predictions, automatically plotting feedback on key results such as ROC curve, AUC, and histogram of predicted probabilities. Classification is done via a custom cost function, iterating through possible thresholds, and plotting various evaluation metrics for additional information. After approving the best threshold, the workpackage updates the dataframe with the predicted probabilities and classifications, and plots the confusion matrix and the feature importance plot.

4.4 Counter-measure implementation

I delivered a workpackage that takes as input the dataframe with predictions, applies selected countermeasures to customers, calculates the effect of applied countermeasures, reclassifies the customers based on the impact of measures applied, and calculates the net effect and effectivity of countermeasures.

I defined possible countermeasures based on research and expert discussions and determined their potential impact and cost per customer. I implemented functions that can be used to easily apply the countermeasures on the dataframe given an input series, and calculate the effect of the countermeasure per each client: the cost of implementation, the expected effect of reducing churn probability in percentage points, and the reclassification of customers after countermeasure was applied. The workpackage also summarizes the net effect of each countermeasure, calculating the net revenue retained, and also allocating it to each measure. The final effectivity of the measures was defined as the ratio of revenue retention capability of a given measure with comparison to the total revenue retained.

4.5 Visualization of results

I delivered a workpackage that takes the final outputs, includes most important charts that can be easily customized to any data, and contains aggregate functions that create instantly plottable dataframes to simplify the creation of any new charts. I plotted 10 key charts which were implemented in the final dashboard.

The final dashboard was prepared using figma, and I created functions using Plotly to visualize key results. To enrich the prototype and to demonstrate the possibilities with the dashboard, I generated random client attributes in addition, and implemented functions in such a way that is easily reproducible for any new dataset during future roll-outs: the functions take an aggregation function that creates an input dataframe for the plot in the structure according to the selected plot type. It aggregates according to either sum, mean or count for the selected numeric column or columns, segments it according to selected categorical feature columns, and creates a dataframe that can be instantly plotted into bar, stacked bar, grouped bar, or scatter plots.

4.6 Next steps: Dynamic churn prevention framework

The current prototype used static data for a single time period, with transactional data aggregatet. However, for future improvements, my client aims to implement a dynamic churn prediction model, based on the Multi Period Training Data framework (Özden Gür Ali and Umut Aritürk, 2014). The Multi Period Training Data proposes a framework for generating multiple observations per customer from different time periods and hence including customers who churned earlier as well. The framework also allows to take into account the variation in external macro-economic indicators such a quarterly data on GDP, monthly data on unemployment, and yearly data on inflation by including multiple time periods into the data. Such time-dependent multi-period data allows to predict churn for each period in advance, while increasing the sample size for training.

To prepare such next steps, I delivered a workpackage that takes the raw customer data, joins it with any time-dependent economic inputs to a dataframe, and creates training and holdout data according to selected method. The workpackage takes care of transforming the cleaned raw data into a multi-index xt data, automatically joins it with collected macro-economic indicators with differing frequency, and as an output separates the training and holdout period and structures the dataset accordingly, creating binary target variables for churn in custom set multiple periods in advance.