MGT 6203 - Data Analytics for Business

Churn Prediction and Preventive Action

Team 1

Forrest Willoughby
Jeh Lokhande
John Abraham
Nirmit Chetwani
Raghav Tandon

Contents

Introduction

Acquiring new customers is often more expensive than retaining existing ones. Given the increase in competition, customers have lucrative alternatives and do not shy from making the switch. Companies have started focusing on retaining existing customers to maintain their market share and prevent other companies from gaining in the market. The best way to solve this problem is personalizing the coupons for each customer.

The question that we decided to explore was to understand the trend of churn in a retail chain and figure out a way to reduce it by recommending coupons to incentivize the churning customers to stay.

Churn analysis is a vague problem with a lot of variability in how it is framed. How do we define if a customer has churned or not? Are no purchases made in a period of 3 weeks a good indicator of churn? Or is it too late to act to retain these customers? If it is not too late, is it too early to call them churned and is spending money on them just an additional cost to the company?

Once we determine the specifics of churn we need to incentivize those customers not to churn. The question that we attempt to answer is 'How do we do that'? Do we recommend products that they have purchased in the past and provide offers on those products? Do we provide coupons for similar products? Or do both?

Literature Review

Churn is a problem across industries. Being able to retain customers instead has been recognized as a way make money and lots of research has gone into identifying the customers who are at risk of churning.

Many methods have been applied to this classification problem. This includes rule based, decision trees, neural networks, nearest neighbor, and ensemble methods [2]. The goodness of these methods is tested using AUC (area under the receiver operating curve) and looking at the lift chart of the top decile [2]. An important part of implementing these models is input/feature selection. The model is only as good as the data that it is built on [3]. Different types of datasets have been used to answer this question. There can be industry specific datasets that can provide better insight into the churn of customers such as customer usage habits with their ISP [3]. Recency, Frequency, Monetary variables have been shown to be good indicators of churn [3].

Method

Data

The Dunnhumby dataset contains household level transactional data for over two years from 2500 households, who are shoppers at a retailer. The transactional data not only contains temporal and product level data, but also has foreign keys that can be used to join with tables containing demographic information, marketing campaign details, and income data for the corresponding customers.

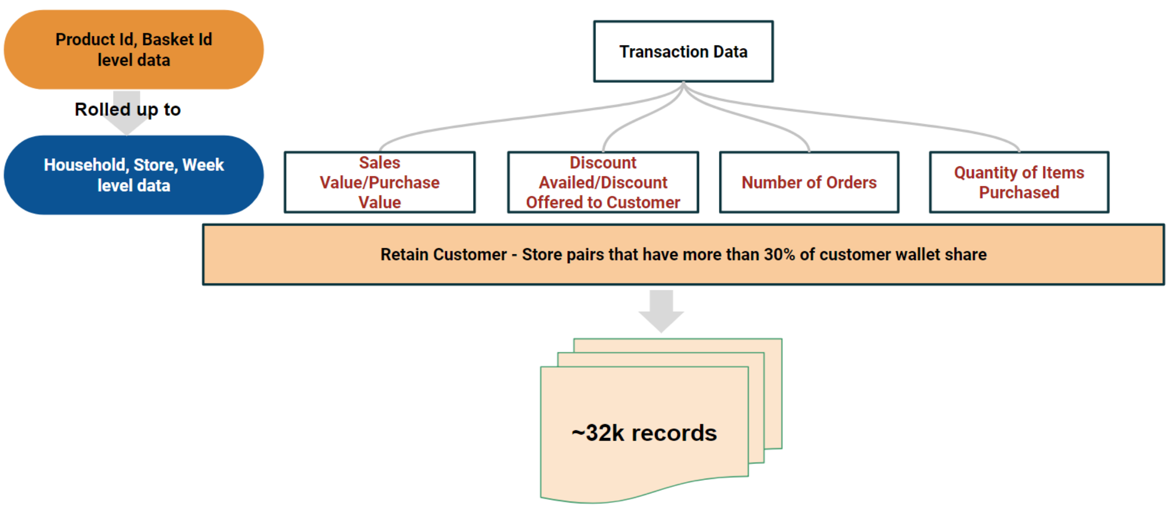
The list of tables with a short description are mentioned below:

Table	Description
Demographic	This table contains data with demographic information for each of the households that shop at the different stores
Transaction	This is the fact table that contains transactional data at a product level with links to all the other dimension tables such as coupon, household, product
Income	This table links to the household table and has data regarding the income of each household
Product	This table has product specific data for products that appear in the transaction_data table
Coupon	This table has coupon level information for coupons provided to the different households under different campaigns
Causal	This table has data about the marketing effort by the company through mailers and displays
Campaign data	This table has temporal information about the campaigns that were run by the retailer
Campaign table	This table links the campaign data table to the household table with household level information about the campaign
Coupon_redempt	This table contains data about coupon redemption with data at a household - day level

Approach

Transaction data for the retailer has been aggregated to a household, store, and week level since that is the relevance we require to develop a churn model. All the metrics such as purchase value, number of orders, etc. have been summed up while aggregating. Demographic variables were also tried, but since they were sparse and did not add show any correlations with churn, they were dropped.
An important point is transaction data is often noisy, with some customers only being one-time customers. Marking these customers as churned and giving them coupons is not how we wanted the model to function and is a challenge we needed to tackle. We decided to only include households that have a frequency of purchase above a certain threshold in the given time-period. Furthermore, the transactional data has multiple stores from which a household might have shopped. To ensure uniformity in the data, we decided to further filter on only customer - store pairs that have more than 30% of the customers' wallet share. Wallet share is how much of a customer's spending they do at a specific store.

The below flowchart describes our approach to prepare the data.

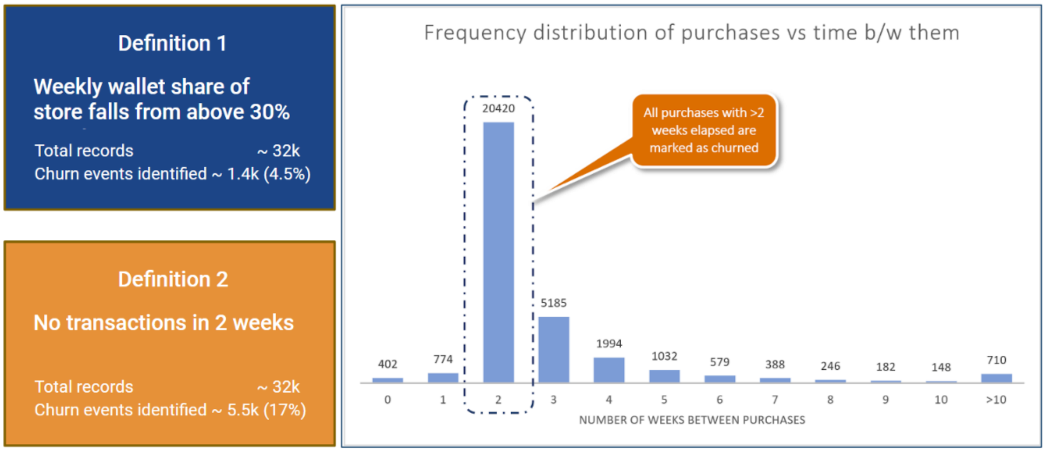


Churn Analysis

The most important point addressed is how to define churn. We had 2 hypotheses:

1. Weekly wallet share of the store for the customer falls significantly below 30%
2. No transaction for the customer in the past 2 weeks

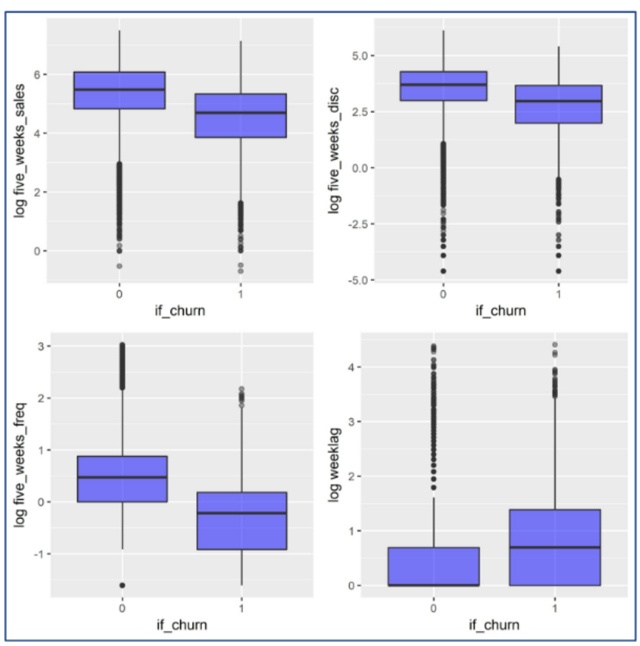
Churn definitions



Definition 1 created a very sparse list for churned customers and was not suitable for the given data. We decided to move on with definition 2 [as discussed with Prof. Hu] which marked around 17% customers as churned. The frequency distribution also indicates that a lot of people visit stores once every two weeks.

Two sets of variables were created to model churn, one aggregating spend for households over past five weeks while the other aggregating spend over past three weeks. The below box plots describe how log (transformation used for final model) of five week-spend variables and week-lag [that depicts the number of weeks before which the last purchase was made] have different distribution of values for churned and non-churned customers.

Box plots showing the spend and frequency data for cases of no churn and churn



Primitive analysis from the above charts;

1. The lesser the average purchase amount, the greater the likelihood to churn
2. The greater the discount offered, the lesser the likelihood to churn
3. The higher the order frequency, the lesser the likelihood to churn
4. The greater the time since the last purchase, the greater the likelihood to churn

We did a similar analysis on the 'Three week' variables and concluded that they display similar behavior to the 'Five week' variables. The correlation matrix below also signifies this high correlation between these two sets of variables.

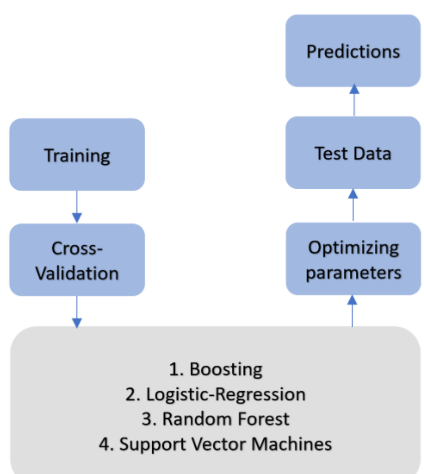
Correlation Matrix

	overall_weekly_purchase	three_weeks_sales	three_weeks_disc	three_weeks_freq	five_weeks_sales	five_weeks_disc	five_weeks_freq
three_weeks_sales	0.65						
three_weeks_disc	0.52	0.8					
three_weeks_freq	0.25	0.52	0.47				
five_weeks_sales	0.62	0.95	0.76	0.51			
five_weeks_disc	0.51	0.77	0.94	0.48	0.81		
five_weeks_freq	0.25	0.5	0.46	0.97	0.54	0.5	
weeklag	-0.06	-0.21	-0.18	-0.21	-0.21	-0.19	-0.21

High correlation between variables corresponding to three and five weeks data suggested to take only one set of those predictors for prediction purposes

We developed four commonly used churn models before finalizing on one. The below flowchart explains how we tested out the approaches to model churn.

Modeling flowchart

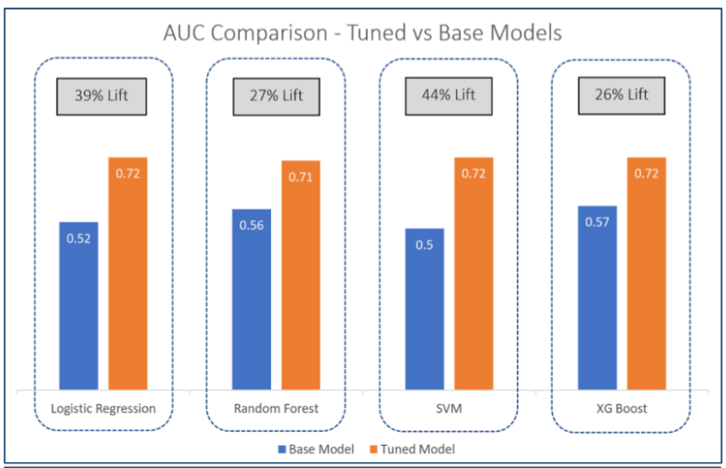


Each of the 4 models had the following variables:

Independent variables with their description:
total_sales – sales amount for that week
overall_weekly_purchase – total quantities purchased that week
overall_weekly_disc– total discount availed that week
row_id – “n-th” order of a household for a store
five_weeks_sales – total amount spent in a store in past 5 weeks
five_weeks_disc – total discount in a store in past 5 weeks
five_weeks_freq – avg. ordering frequency in a store in past 5 weeks
weeklag – number of weeks before the current week when the last purchase was made in a store
Dependent: if_churn

We used 10-fold cross validation to test the 4 models out. The lift for SVM and Logistic Regression as compared to the models with default parameters is the highest at 44% and 39% respectively.

A/f Fine Tuning Model Parameters



Model parameters are fine-tuned to optimize the AUC-values, using 10-fold cross validation. We see that the lift is highest for **SVM and Logistic Regression** models.

Logistic regression showed comparable results with XG-Boost in terms of accuracy, AUC and recall, hence was chosen because of coefficient- interpretability and lesser run-time. The table in the next page show the results of all the fine-tuned models.

Final results

Model	Accuracy	Precision	Recall	F-score	AUC
Logistic Regression	0.67	0.32	0.8	0.46	0.72
Random Forest	0.73	0.35	0.68	0.46	0.71
SVM	0.66	0.31	0.8	0.45	0.72
XG-Boost	0.72	0.34	0.73	0.46	0.72

Final model chosen is **Logistic Regression** because of coefficient interpretability and lesser run-time as compared to other models

Once we figured out customers that are likely to churn, we needed to come up with a strategy to prevent them from leaving.

Product Recommendation

Here we follow a two-staged approach.

Step 1: Once we have a list of customers that are likely to churn from a store we find their recent purchase data. After eliminating low frequency, high value products (example Television, Vacuum cleaner, etc.) using a filter on quantity, we find the top *n* products per customer by sales value (by customer wallet share). This is the list of products that are most likely to bring the customer back to the store if there are discounts offered on them. This is based on the premise that a rational buyer will respond to a monetary incentive on the products that are important (in terms of wallet share) to the customer.

Step 2: Once such important products are identified in Step 1 we can augment this list with products that were purchased along with them. These products are found using an item-to-item collaborative filtering approach (roughly like [1]). We vectorize the items across the customers and calculate the cosine distance between all such pairs of vectorized items. The pair of items with higher cosine similarity, i.e. lower cosine distance are more similar and we use this information to make product recommendations.

For example: if soft drink 1 is often found to be sold together with soft drink 2, it might interest the customer who purchases soft drink 1.

Here is an example of a product recommendation. The first product is the one from the customer purchase history. The other products are sorted based on their similarity to the first one.

	prod_names	product_id	similarity
0	SOFT DRINKS 20PK&24PK CAN CARB	948966	1.000000
1	SOFT DRINKS 6PK/4PK CAN CARB (	6704144	0.847998
2	SFT DRNK MLT-PK BTL CARB (EXCP	9803819	0.834019
3	SFT DRNK MLT-PK BTL CARB (EXCP	1132956	0.700076
4	TABLE SALT POPCORN SALTICE C	999318	0.657381

Evaluation of recommendations: To evaluate the quality of product recommendation, we found customers who take a break (more than or equal to two weeks) from the store. For such customers, we picked a duration of 10 weeks before and after the break. Of these purchases, we evaluate the top wallet share products and monitor the number of matches (say in the top five products).

Match_Level	Recommendation_Type	Customer_Matches	Return_Customers	Overall_Customers
1	History	32	60	73
1	History+CF	34	60	73
2	History	10	60	73
2	History+CF	14	60	73
3	History	3	60	73
3	History+CF	4	60	73

As can be seen above, at least one of the top five high sales value products match for 32 out of 60 customers who returned to shop at the store. Including the collaborative filtering-based recommendations (adds up to three collaborative filtering-based products per history-based products) 34 out of 60 returned customers had one of the top five high value purchases matching. Similarly, for a two-product match - a purely purchase history-based approach matched for 10 customers and history + collaborative filtering-based approach matched 14 customers.

History and history plus collaborative filtering-based products are very useful in predicting the high sales, value products of a customer. Thus, it is reasonable to expect discounts/offers on these products might persuade a customer to return to a store for more purchases.

Summary and Future scope

Our work was motivated by the problem of predicting and reducing churn as it has a direct impact on a retail chain's performance. Retaining customers is in general easier than acquiring new customers. If we could identify customers who have a high likelihood to churn, and could target them with attractive offers/coupons, we may be able to prevent losing them.

A two step approach;

Step 1: Identify customers with a high churn risk
Step 2: Target them with discounts/coupons on relevant products to prevent losing them

We studied the frequency distribution of the customer buying patterns and noticed that over 87% of customers make at least one purchase once every fortnight. We decided to make this as a criterion to identify churn because of the sharp drop in the frequency table after the second week. We tried different models and identified that logistic regression gave the best results in predicting churn based on parameters such as precision and recall.

In the next step, we try to make product recommendations for customers identified in the previous step. This works in two steps. In the first step, we look at the customer's transaction history and identify the highest sales value items before the customer churned. In the second step, we extend this list of products with products that were frequently purchased together with them. Such product-to-product relations are identified using a collaborative filtering approach.

Future Work

We can implement some more ideas for better churn prediction and product recommendation. We can try ensemble models, and oversampling methods like SMOTE to get better performance on recall. For better product recommendation, we can use matrix low rank approximation methods such as singular value decomposition or non-negative matrix factorization for collaborative filtering. We can also try to study the past recommendations and evaluate their performance to test our current recommendations and incorporate that into our models.

All the modelling and data manipulation for this project was done in Python and R.

References

1. Amazon Recommendations: https://www.cs.umd.edu/~samir/498/Amazon-Recommendations.pdf
2. Wouter Verbeke, Karel Dejaeger, David Martens, Joon Hur, Bart Baesens, New insights into churn prediction in the telecommunication sector: A profit driven data mining approach, European Journal of Operational Research, Volume 218, Issue 1, 2012, Pages 211-229, ISSN 0377-2217, https://doi.org/10.1016/j.ejor.2011.09.031. (http://www.sciencedirect.com/science/article/pii/S0377221711008599)
3. John Hadden, Ashutosh Tiwari, Rajkumar Roy, Dymitr Ruta, Computer assisted customer churn management: State-of-the-art and future trends, Computers & Operations Research, Volume 34, Issue 10, 2007, Pages 2902-2917, ISSN 0305-0548, https://doi.org/10.1016/j.cor.2005.11.007. (http://www.sciencedirect.com/science/article/pii/S0305054805003503)