ChurnSim: A Customer Churn Behavioral Simulation System For Machine Learning Education and Experimentation

Carl Gold*

*Independent Researcher, Berkeley, CA

August 9, 2023

Abstract

This report presents ChurnSim, a configurable simulation system that generates customer churn data for use in Machine Learning experiments and education. Customer churn behavior is of interest in a wide variety of business contexts and churn prediction with Machine Learning is now commonplace. However, raw data sources for studying customer churn are proprietary and not widely available. ChurnSim allows the generation of raw data that realistically matches a wide variety of customer behaviors related to the churn and renewal of subscription products. The data generated can be used for the study of Machine Learning churn prediction methods and for training students in applied Machine Learning methods and feature engineering.

Keywords: Simulation, Machine Learning, Churn, Customer, Subscription,

1 Introduction

Originally developed by the Telecom industry, in the past decade customer churn prediction has become a staple machine learning and data science problem. Many papers, code and products are available to predict customer churn (for examples see e.g. [1, 4, 6, 9] .) However, data sources pertaining to customer churn or normally proprietary: Even if the data is stripped of PII it remains highly strategic and businesses are loath to give public access. As a result there are limited options available for a student to study a churn problem, and also limited options for researchers analyzing churn prediction methods. The ChurnSim system fills this gap by providing a simulation of customer churn that is realistic according to a variety of measures.

A simple version of the ChurnSim model was first introduced in [4] with little explanation. The purpose of this report is to elaborate on the model methods and introduces extensions added after the publication of [4]. All code described here is available from [5]. This report provides a demonstration of and a companion to the code for students and researchers interested in using the simulation.

1.1 Characteristics of Customer Churn

Real case studies of churn like those illustrated in [4] have demonstrated several distinctive properties common to most customer churn scenarios. Some important examples include:

- For products with multiple price points, users of the most expensive version may churn at a lower rate than users of less expensive version. This is found in both individual and multi-user products.
- As a result of the preceding point, churn measured by monetary value (monthly recurring revenue, or MRR, churn) is generally lower than churn measured by the count of customers.
- For products with upgrades and add-on products, Net (Dollar) Retention may be greater than 100%.
- Measurements of actions taken using Software as a Service (SaaS) products show a long tail (heavy right skew) - the "power users" far exceed the typical users, for both consumer and enterprise software products.

- The amounts of different actions taken by different users on the product are highly correlated - “power users” do more of everything.
- Simple counts of actions taken using the product are often highly predictive of churn, but any single type of action has a diminishing impact on churn the more that a customer takes the action.
- Even actions which are disengaging (churn inducing) appear to be associated with reduced churn in a single variable analysis, due to the aforementioned “power user” phenomena. But the disengaging effect of an action can be demonstrated by studying ratios of actions.

The ChurnSim model has been specifically designed to match these and other characteristics of real churn use cases.

1.2 Core ChurnSim Model

CuhrnSim simulates customer accounts as the users interact with a subscription product, and saves data in a format that mimics real churn use cases. An overview of the model is shown in Figure 1. These core components were used in the simulated data presented in [4] and are described in section 3.2.

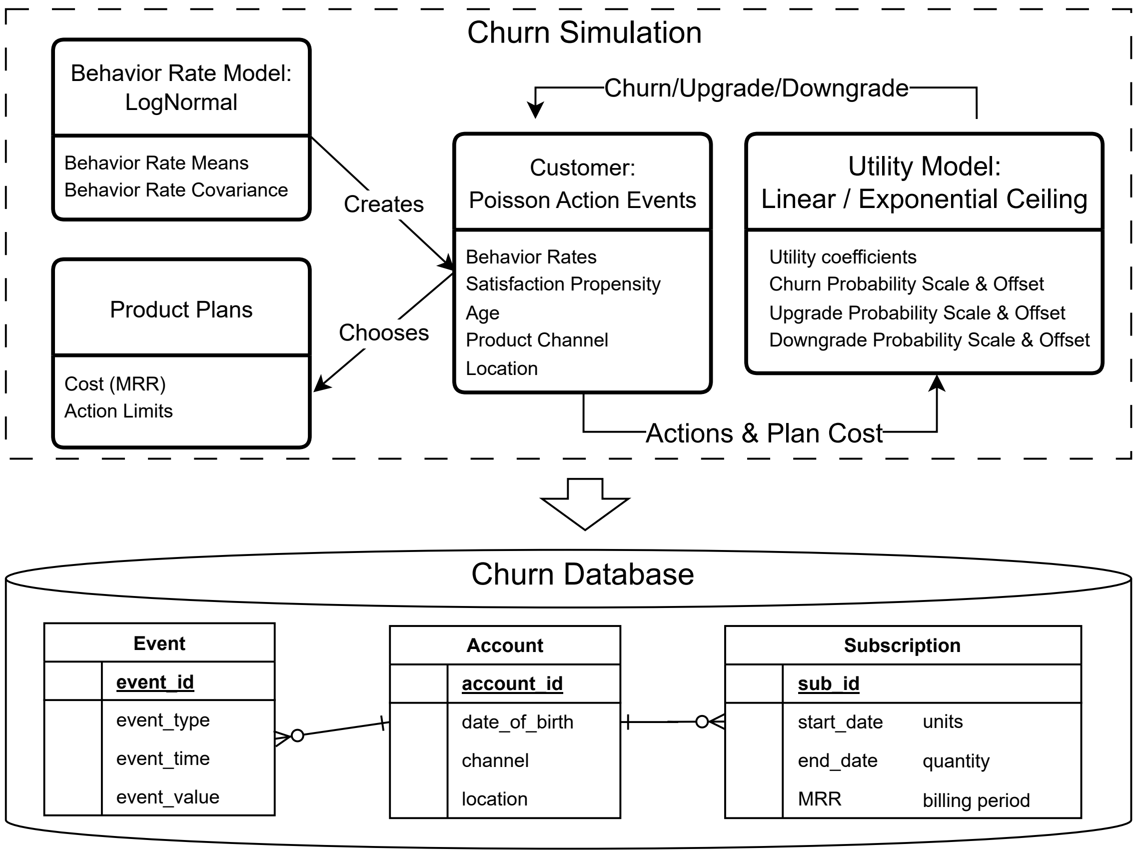

Figure 1: Overview of the ChurnSim Model

Figure 1 illustrates the core ChurnSim model components:

1. Behavior Rate Model - This model determines the rates at which a customer uses different product features with a log-normal model to select correlated behavior rates for each simulated customer. With a LogNormal model behavior rates show a long tail, as described in section 3.2.1.
2. Customer - After behavioral rates are chosen, the customer generates actions according to a Poisson model as described in section 3.2.1. The customer also has secondary attributes, like age and channel. These customer attribures affect the simulation as described in section 3.4.

3. Utility Model - This model calculates a hidden "utility" value for each customer. This is utility in the economic sense of the satisfaction or dissatisfaction that results from the customers' use of the product. The utility model consists of multiplyng the customer activity counts by coefficients that model each activity's benefit or harm. The result is then transformed via an exponential ceiling function which models hedonic adaptation, as described in section 3.2.2. A customer's utility determines the probability that the customer churns after each month of simulated activity based on a sigmoidal function as described in 3.2.3. The customer may also upgrade or downgrade their product choice as described in section 3.5.4.

Time is discretized into simulated months of account activity and the simulated customers may choose to churn or continue their use every month. The core loop by which each customer's behavior is simulated until they churn, and a population of customers is grown over a multi-month simulation is described in section 3.3.

Figure 1 also illustrates the data generated by the simulation. The data generated by the simulation is written to a SQL database and is intended to mimic the data available to practitioners in actual customer churn scenarios, albeit in somewhat simplified format. The data generated by ChurnSim consists of:

1. Account records with a few descriptive fields.
2. Co-terminal subscription records, in which the start date of new subscriptions coincides with the end date of the previous subscriptions. Customers have multiple subscription records until they churn. As described in [4] this is one of the standard ways to track subscriptions and in fact is the best practice. Subscription records include other details like the cost in terms of monthly recurring revenue (MRR) and if there is a specific quantity of a budgeted feature that the subscription entitles the user to access.
3. Event records log the actions that customers take on the system. These are point in time events of different types that may also have a real value associated with them, such as a currency amount or a duration. Such events are feature engineered before use in machine learning models - feature engineering is not part of ChurnSim and is left to practitioners. (For demonstration of appropriate methods see [4] chapters 3, 4 and 7.)

1.3 ChurnSim Model Extensions

The simulations also included additional features designed to demonstrate specific areas of churn related data science and analysis that were used in [4]. These additional model components are described in section 3.4:

1. Weekday variation - Customer behavior intensity follows weekly cycles of intensity.
2. Product Channels - Defines different populations of users that use the product with different rates.
3. Customer satisfaction propensity - Customers may be easier or harder to satisfy, making them less predictable.
4. Customer Demographics - Defines customer age and location characteristics that may interact with their behavior.

Recently the ChurnSim model has been updated to include additional components that can be used to simulate more complex product situations. This allows demonstration and analysis of more complex product scenarios than the simulation of [4]. These advanced features are described in section 3.5:

1. Multi-User Products - Determines the the number of users, their usage and utility for customer accounts with multiple users.
2. Valued events - Such as monetary transactions or activities with durations like streaming media.
3. Product Plans including plan limits, cost and billing periods - Allows different levels of product subscription with different prices (costs), limits on the users and usage.
4. Add on products - Additional optional product components with their own prices and action allowances
5. Upgrade & Downgrade Model - Simulates if a customer account switches product level or add-ons.
6. Billing Periods - Customers may sign up for multi-month subscriptions, and change their billing period over time.
7. Discounts - Customers may pay varied amounts less than the list price for the product.

1.4 Related Work

ChurnSim is novel at this time in that there are no other simulation environment specifically designed to match customer churn behaviors as described in section 1.1. But other simulation systems have been developed in other cases where real data sources tend to be propriety. One example of such a simulation is [7], which creates simulated data sets based on sequential user interaction scenarios for reinforcement learning and recommender systems. Simulation environments are also included in some reinforcement learning platforms such as [2]. A distinctive aspect of ChurnSim is that it produces raw data in a format as would be found in real churn analysis use case. A data set suitable for machine learning churn prediction algorithms must be derived through feature engineering and outcome measurement as described in [4].

2 Example Simulated Case Study Results

This section illustrates simulated case study results created with ChurnSim. The results highlight some key areas in which the simulation system can realistically reproduce key aspects of subscription product churn and demonstrates ways that the simulation system can help to analyze machine learning modeling questions.

2.1 Simulation CRM Product

The simulation presented here is for a SaaS Customer Relationship Management (CRM) system. In the simulation, approximately 2,000 multi-user customers are simulated on a product with multiple plan levels and add-on products over a 24 months period. Table 1 shows an example of the type of product plans supported by the simulation: There are five plan levels with maximum user allowances ranging from 5-100 users and prices ranging from $90/month to $2000/month. Each plan has three options for the billing period: Monthly, Bi-Annual and Annual. Longer term plans offer modest discounts.

Table 2 lists the behaviors available to simulated customers in the CRM simulation. There are 26 types of actions which represent common events in CRM systems: adding contacts and leads, winning and losing opportunities (meaning sales), etc. In addition, the behaviors for winning and losing opportunities have a (monetary) value assigned to each event. Each behavior in the simulation is configured with a mean rate, a maximum rate and an impact on utility per event (or per value in the case of valued events.) The simulation also configures a correlation matrix between the customer rates for related events. Compete details of the behavioral simulation are described in the Methods section 3.

Plan Level	Max users	MRR Monthly	MRR Bi-Annual	MRR Annual
Starter	5	100	95	90
Basic	10	200	190	180
Standard	25	500	475	450
Advanced	50	1000	950	900
Premier	100	2000	1900	1800

Table 1: Plans for the SaaS CRM product simulation

2.2 Analytic Results

The following sections describe analytic results that realistically match known characteristics of subscription churn. By matching such phenomena in the simulation, students and researchers can have confidence that training and machine learning experiments based on the data have a high degree of realism.

2.2.1 Churn Rates

Figure 2 shows month by month churn rates generated by the CRM simulation. Three types of churn measurements are shown:

1. Standard account (logo) based churn - measures the percentage of customers from the start of each month that churn during the month. See [4] chapter 2, section 2.4 for details.
2. MRR churn - measures the percentage of revenue lost to both outright customer churn and downgrades. See [4] chapter 2, section 2.6 for details.

Behavior	Mean	Maximum	Utility	Behavior	Mean	Maximum	Utility
add_competitor	3	50	-5	edit_search	2	30	-1
add_contact	3	20	20	email_lead	5	50	1
add_lead	5	50	10	email_list	2	10	2
advance_stage	2	30	2	lose_opportunity: count	2	20	NA
call_lead	5	50	1	lose_opportunity: value	1000	5000	-0.07
cancel_meeting	2	10	-2	meeting	3	90	20
create_list	2	5	-1	MRR	NA	NA	-0.1
create_opportunity	9	90	1	quote	3	30	2
create_search	2	30	-1	schedule_meeting	4	60	1
delete_list	1	5	1	search	5	50	10
delete_search	1	10	-1	unsub_lead	3	20	-1
disqualify_opportunity	3	50	-5	users	10	100	0
edit_contact	3	50	-1	win_opportunity: count	4	40	NA
edit_lead	3	25	-1	win_opportunity: value	1000	5000	0.4
edit_meeting	4	50	-1				

Table 2: Behaviors for the SaaS CRM product simulation

3. Net retention - measures the amount by which revenue from the customers subscribed at the start of the month has grown or decreased by the end of the month. See [4] chapter 2, section 2.3 for details.

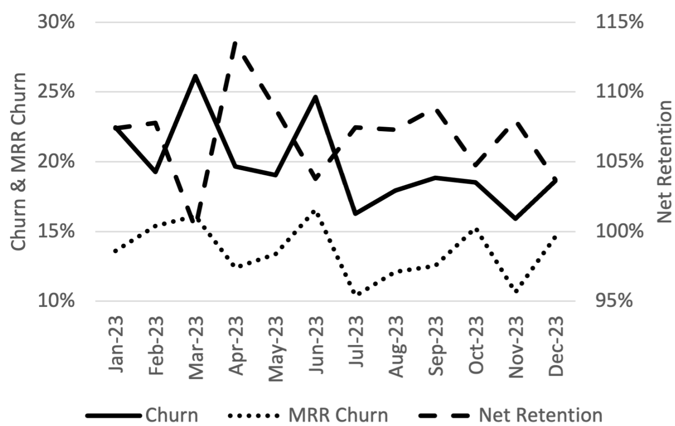

Figure 2: Annual Churn Rate, MRR Churn Rate and Net Retention for the CRM Simulation: MRR churn is less than Account churn and Net Retention is greater than 100%.

Figure 2 demonstrates the simulation producing typical results for real (successful) SaaS companies:

- MRR churn is less than account churn. This arises from the fact that customers with many users who pay more are less likely to churn than customers with fewer users paying less.
- Net retention is greater than 100%. This is typical of successful SaaS companies that generate more new revenue in upgrades than they lose from churns and downgrades. (Note the right-side axis for Net Retention in Figure 2.)
- Churn rates vary from month to month, which is typical of enterprise SaaS products with a relatively low number of users.

In the simulation variability is purely due to the natural variation when observing a small population with a relatively low event rate. For churn in the real economy variability in churn rates also arises from changes in the market environment and seasonality.

2.2.2 Long Tail Behavioral Distributions

One well known characteristic of customer behavior on software platforms is that behavioral counts are long tailed, or right skewed [4]. A simulated reproduction of this phenomena is illustrated in Figure 3 which shows a histogram of measurements of the number of won opportunities in the prior month for the CRM simulation. The distribution is strongly skewed with a mean value of 75, a maximum of around 3,200 and a skew of 6.2. The count decays approximately linearly on a log-log scale. The ChurnSim model reproduces this behavior with a Log-Normal behavioral rate model, described in Section 3.2.1.

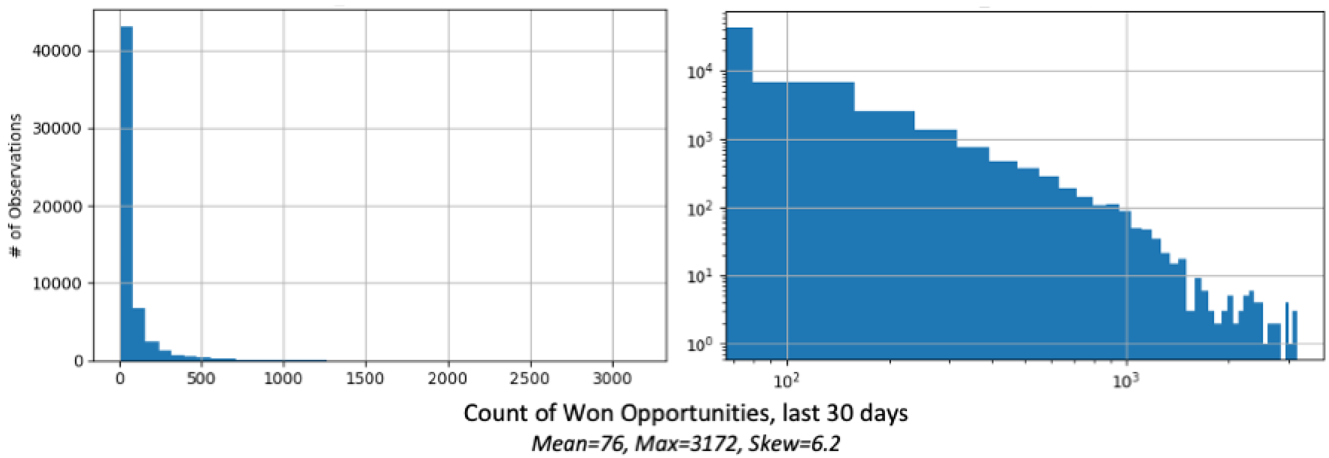

Figure 3: Distribution of Number of Won Opportunities. Left: Natural Scale. Right: Log-log scale.

2.2.3 Hedonic Adaptation To Engaging Events

Figure 4 illustrates the dependence of the churn rate on the value of opportunities closed by simulated accounts using the CRM system. In CRM parlance an "opportunity" is a potential sale to a customer, and closing the opportunity means completing the transaction. This is an example of an event with a dollar value associated with it, in which the event is engaging for the customer - in the underlying churn model, the event has positive utility associated with it (see Table 2). Figure 4 also illustrates hedonic adaptation in the model: Churn is substantially higher for accounts that close less than approximately \$100K in sales per month. But beyond around \$250K in sales per month, additional opportunity value has little impact on the churn rate. For details of how such analyses are performed, see [4] Chapter 5. The model that produces hedonic adaptation in the simulations is described in Section 3.2.2.

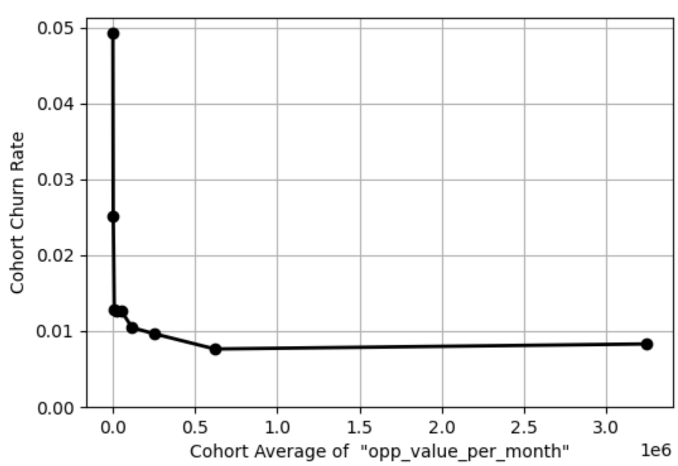

Figure 4: Churn Rate vs. Opportunity Value Closed per Month

2.2.4 Disengaging Events and Churn

Figure 5 illustrates the dependence of the churn rate on the number of opportunities lost (meaning potential sales that didn't close) by simulated accounts using the CRM system. This is an example of a disengaging event - one for which customer derives negative utility or dissatisfaction with the product when it occurs (see Table 2). However, high levels of the disengaging event are still associated with lower churn rates. The reason this is commonly observed in real SaaS product usage and churn is that the best customer accounts, those with more users in enterprise SaaS or higher personal usage for a consumer SaaS product, have more events overall due to correlation between different types of usage. So the best accounts tend to have more disengaging events along with their engaging events. Such "whale" (enterprise) or "power user" (individual) accounts churn less because the engagement outweighs the disengagement. As a result, in single variable analysis, like the behavioral metric cohort in Figure 5, the disengaging event may appear to be associated with lower churn.

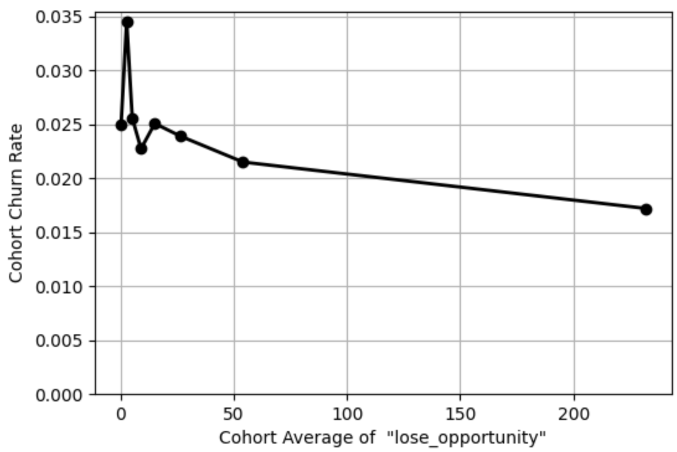

Figure 5: Churn Rate vs. Number of Opportunities Lost Per Month

Figure 6 illustrates an analytic technique that helps to reveal disengaging in a single variable analysis: Rather than analyzing the event in isolation, analyze the rate (or ratio) of such events to a related event. In figure 6 the measurement is made on the percentage of opportunities that are lost to competition out of the total. Such a measurement controls for the effect of the overall account activity level and reveals the disengaging event because a higher *proportion* of the event is associated with churn even as higher count of the event is associated with retention. In this case loss rates below around 30% all have relatively low churn but the churn rate rises steeply for accounts with loss rates above around 50%. For details on how to perform such an analysis see [4], Chapter 7.

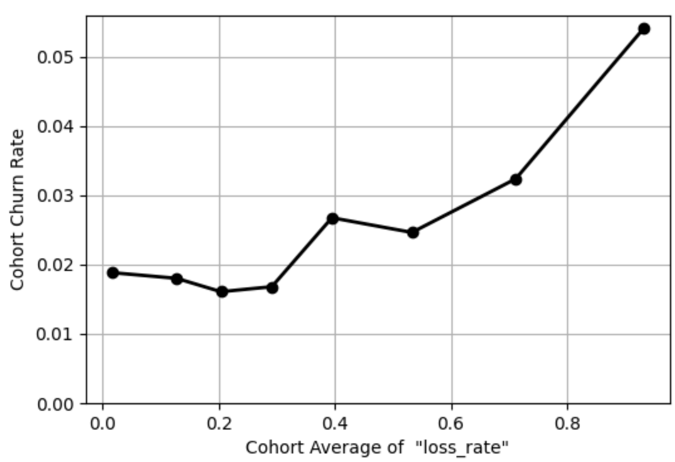

Figure 6: Churn Rate vs. Percent of Opportunities Lost Per Month

2.2.5 Churn and MRR

Figure 7 illustrates the dependence of churn on the MRR paid by the customer in the simulation. MRR is an example of a disengaging event: It has negative utility in the simulation and tends to cause churn (see Table 2). However, accounts with more users also have more positive events such as closing opportunities. As a result, churn is generally lower at accounts which pay more. The CRM simulation exhibits this pattern and the result is illustrated in figure 7.

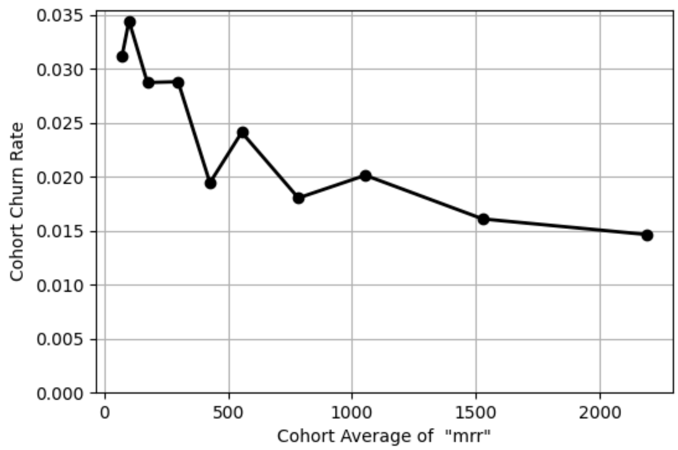

Figure 7: Churn Rate vs. MRR Paid per Month

Although high MRR is associated with low churn, when MRR is viewed on a relative basis compared to the benefits a customer receives it is apparent that paying high MRR is actually disengaging. This is the same type of analysis as shown in section 2.2.4. Figure 8 shows the churn rate in comparison to MRR divided by the monthly opportunities closed (the same metric as in Figure 4.) For most customers the rate is below 0.01 and has no impact on churn, but for those customers where the rate is above 0.05 the impact on churn is pronounced. For details and comparable figures from a real case study see [4] Chapter 7, Figures 7.2 and 7.3.

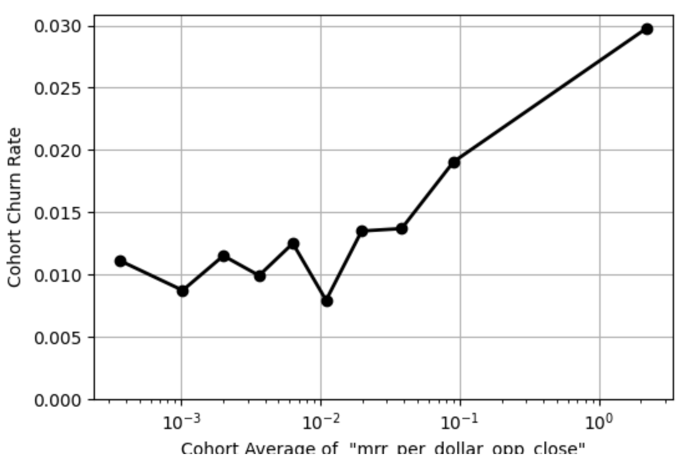

Figure 8: Churn Rate vs. MRR per Dollar of Opportunities Closed per MOnth

2.2.6 Churn vs. Billing Period

Figure 9 shows the churn rate by billing period in the CRM simulation with two points of view: From a monthly churn rate perspective (left axis), churn is far lower for longer billing periods. But when viewed from the point of view of the proportion of customers that churn upon their renewal date the higher billing period plans have a much higher churn rate. The low monthly churn rate for long term plans is due to the fact that even when custmers want to churn, they typically wait until the end of their term. This also explains the high churn rate when the renewal date arrives: Customers on long term plans have a long time in which they may have decided to churn and so on the renewal date churn is much more likely. These results are typical of real products sold with multiple billing periods. Financially, the point of view of monthly churn rates is what is most important and is why organizations prefer to sell longer billing periods. For detailed discussion see [4] chapter 5, Section 5.1.5.

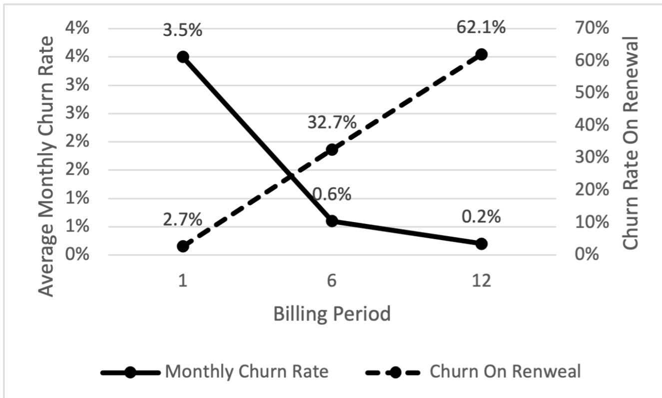

Figure 9: Churn Rate vs. Subscription Billing Period
. Left Axis: Average Monthly Churn Rate; Right Axis: Churn Rate Upon Renewal

2.3 Churn Prediction with Machine Learning

The following sections present examples of machine learning experiments that are made possible by ChurnSim. Multiple instantiations of customer populations can be produced, following the same underlying dynamics or with variation on individual parameters. This allows students, practitioners and researchers to investigate aspects of churn modeling that are not addressable in the real world.

2.3.1 Interpreting Churn Models with the SHAP method

At this time, gradient boosting algorithms like XGBoost [3] are state of the art methods to predict customer churn with Machine Learning [9, 4]. Tree based boosting methods like XGBoost are interpretable through the use of the Shapely Additive Explanation (SHAP) method [8]. However, researchers in the field have observed that the precise influence of less important features can vary when models are refit over time or with different parameters. Figure 10 illustrates a more meta variant of this problem: The simulation is performed 3 different times, producing different populations following precisely the same model. Churn is predicted using an XGBoost model and a variety of aggregate features measuring product actions by the customers plus ratios of the same. (The feature engineering approach is described in [4] chapter 3 and chapter 7.) Due to natural variability in the population the ranking of the significant features after the top 3 are highl variable.

The situation is analyzed in figure 11 which measures the mean absolute SHAP values and the correlation of the SHAP values with the features over multiple generated populations and models. After the two most significant features, the standard deviation of the average absolute SHAP values are high relative to the average absolute SHAP values. So in different populations the features will show different ranking in the SHAP summary plot despite the same underlying dynamics. This finding should inject a note of caution for those trying to use these methods to assign significance to customer actions based on boosting model results. Fortunately, the correlation of SHAP values with the features, which describes whether or not the feature consistently influences the prediction in the same way, are relatively more consistent across the different populations and models.

2.3.2 Model Evaluation and Acausal Churn

Another issue important to machine learning practitioners is whether a model's performance is good or bad relative to similar use cases, and how much the performance of their model depends on the modeling

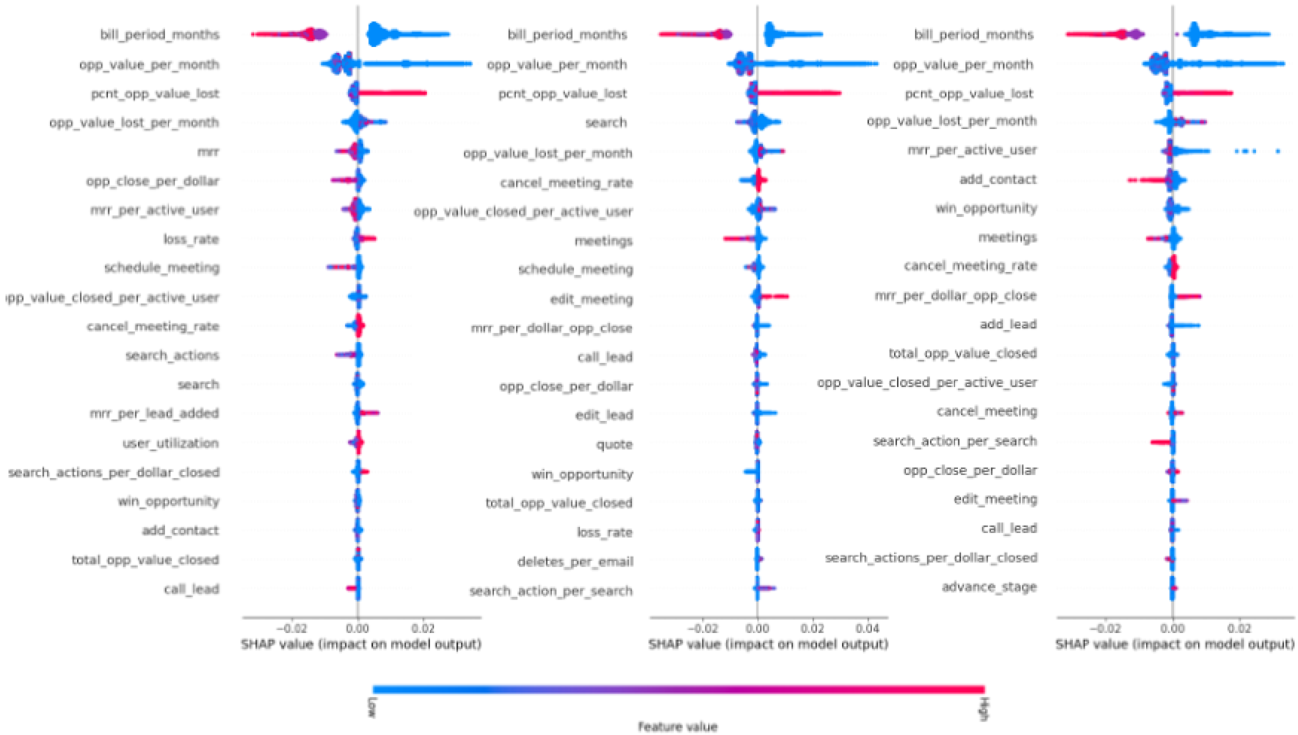

Figure 10: Comparison of SHAP Summary Plots on 3 Identically Parameterized Simulations

decisions versus the inherent difficulty of the problem. [4] provided some guidelines for model evaluation using Area Under the Receiver Operator Curve (AUC) for churn use cases, suggesting that AUC of churn prediction models should generally lie in a range from 0.6 to 0.8: AUC below 0.6 suggests inadequate or erroneous data. AUC greater than 0.8 suggests data leakage in the modeling setup because user behavior is not that predictable even with good data and state of the art modeling techniques.

An interesting question that can be addressed via simualtion is how much the model performance is influenced by “acausal” churn, which in this context means churn due to all causes that are unrelated to the measured use of the product (about which it is assumed we have complete information.) Acausal churn is included in the ChurnSim model as detailed in section 3.5.8. Figure 12 shows how the cross-validated AUC depends on the degree of acausal churn for the CRM simulation. Churn is predicted using an XGBoost model and a variety of aggregate features measuring product actions by the customers plus ratios of the same. (The feature engineering approach is described in [4] chapter 3 and chapter 7.) Without any acausal churn at all the AUC is 0.9 which would be considered a an implausible result in a real churn case study. It is worth noting that with no acausal churn the underlying churn rate is 1% per month. When acausal churn is much higher at 5%, the model AUC decreases to 0.73 which is more consistent with typical churn scenarios. Overall churn at that rate of acausal churn is 4.3%. Note that the monthly churn rate may be lower than the acausal churn rate because customers are still bound by multi-month contracts as described in section 3.5.6.

3 Simulation Methods

3.1 Notation

The following sections details the workings of the ChurnSim model components. As ChurnSim is a random simulation, much of the method concerns drawing variables from a variety of random distributions. The following notation is used to indicate drawing random variables from the indicated distribution:

- N : the Normal Distribution
- P : the Poisson Distribution

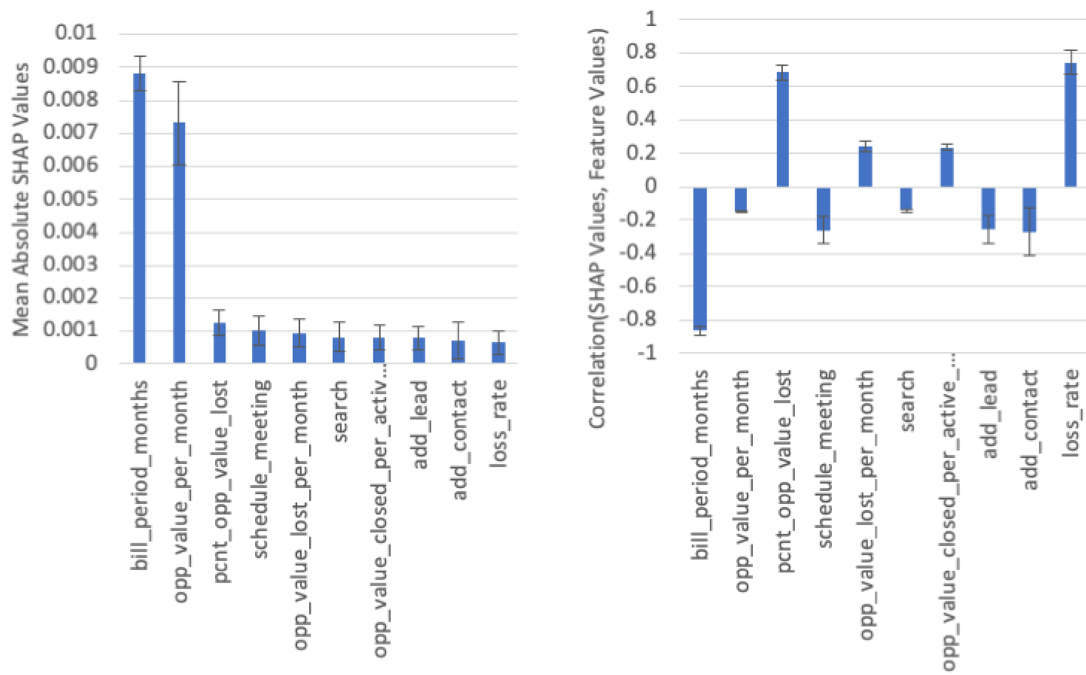

Figure 11: Comparison of SHAP Values, Mean and Standard Deviation: Mean Absolute SHAP Values, Left; Correlation of SHAP values with Fetures.

- U : the Uniform Distribution
- $\sim$: binary operator indicating that the LHS is drawn from probability distribution given on the RHS. For example, $x \sim N(0,1)$ states that x is drawn from a standard normal distribution (0 mean, 1 standard deviation).

3.2 Core ChurnSim Model

The following sections describe the core churn model that was used when creating the simulated case study in [4] and the results presented in section 2.

3.2.1 Behavioral Model

The ChurnSim behavioral model assumes each customer has average monthly rates for each of a set of actions that can be taken when using the product. The rates for each customer account are determined from a log-normal model: Rates are drawn from a multivariate Normal distribution, but the actual rate is given by a base value exponentiated to the power drawn from the Normal distribution. The log-normal model accounts for the common observation that real customer behavior intensities follow a "long tail": The most active customers use the product at a rate that is far above the mean, so the distributions are right skewed. The Log-normal approach also guarantees positive rates.

Each ChurnSim model defines a vector of mean behavior rates $\bar{\mu}$ and positive definite behavioral rate covariance matrix Σ (for interpretability the configuration accepts a correlation matrix in the configuration). Given those, an individual customer has a vector of average action counts per month $\bar{\omega}$ given by:

$$\bar{\omega} \sim \bar{a}^{N(\bar{\mu},\Sigma)} \tag{1}$$

Where N represents the joint multivariate Normal distribution and the base of the logarithm a is chosen to determine the degree of "extreme" customer behavior. (In the current version of ChurnSim the base a is the same for all actions so the notation $\bar{a}$ indicates $a \times \bar{1}$.)

A simulated customer is created with a vector $\bar{\omega}$ of average monthly action rates, and the rates are never changed for that customer. For simulation, the action rates converted to daily rates by dividing

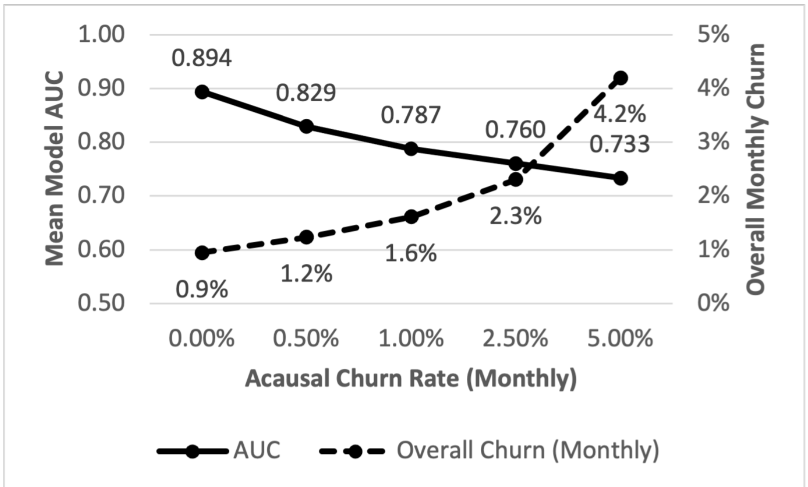

Figure 12: Model AUC vs. Acausal Churn Rate

by 30 ($\hat{\omega} = \bar{\omega}/30$). The number of each action that a customer takes in a given day $\bar{\alpha}$ is drawn from a Poisson (P) distribution with the daily customer rates as the mean:

$$\bar{\alpha} \sim P(\hat{\omega}) \tag{2}$$

where $\hat{\omega}$ are the daily action rates.

3.2.2 Utility Model

The ChurnSim utility model determines how much satisfaction or dissatisfaction a customer derives from the actions that they take using the product. The utility model simulates hedonistic adaptation by customers using a bounded exponential growth model based on the number of actions that a customer takes. In this context hedonistic adaptation refers to the observation that after a positive or negative experience using a product, further instances of the same experience have less and less positive or negative affect on the customer.

The utility model with hedonistic adaptation resulting from a single type of customer action has the form:

$$\upsilon_i = \mu_i u_i (1 - e^{-c\alpha_i/\mu_i}) \tag{3}$$

In equation 3 :

- u_i is an action specific utility coefficient which may be positive for an engaging (satisfying) action or negative for an action that causes disengagement (dissatisfaction).
- μ_i is the average number of customer actions per month from equation 1.
- α_i is the number of actions taken by the customer in the past month from equation 2.
- c is a scaling constant that sets the rate of hedonistic adaptation for all behaviors in the simulation.

In equation 3 the utility for a given number of actions approaches the utility coefficient u_i for that action times the mean number of actions μ_i, which sets the limit on the amount of utility which that action can provide to any customer. The fraction of the maximum utility the customer receives "decays" (upwards) to the limit with the ratio of the number of actions α_i to the average number of actions μ_i.

The total utility which a customer receives in a given month is given by the sum over all actions of equation 3:

$$\upsilon = \sum_i \mu_i u_i (1 - e^{-c\alpha_i/\mu_i}) \tag{4}$$

3.2.3 Churn Model

The churn probability for a customer at the end of one month is given by a standard sigmoidal function of the customer utility in that month:

$$P_{churn} = 1.0 - \frac{1}{1 + e^{-\upsilon \xi_{churn} + \Delta_{churn}}} \tag{5}$$

where υ is the total utility from equation 4, ξ_{churn} ($\xi_{churn} > 0$) is a scaling constant and Δ is an offset. Using the constants any given set of events and utility parameters can be scaled to a desired churn rate suitable for the simulation.

3.3 Simulation Algorithm

Combining the different parts of the model, the logic of simulating a single customer until they churn is shown in Algorithm 1 and the algorithm for a complete simulation is shown in Algorithm 2: The simulation consists of creating a fixed number of customers in a start month (T_1) and simulating each until they churn or the maximum time (T_2) is reached. At creation each customer has its action rates set, and in each month the customer's number of action is randomly determined according to the rates. Churn at the end of each month is randomly determined according to the utility the customer's actions generate. In subsequent months new customers are added according to a growth rate. For realism of the stored simulated data customers are randomly assigned to to different times of day.

Algorithm 1 Simulate Customer from T_1 until Churn or T_2

Draw customer behavior mean values $\bar{\omega}$ according to eq. 1
Set daily action rates $\hat{\omega} = \bar{\omega}/30$
Month $T \leftarrow T_1$
while true **do**
    $\bar{\alpha} \leftarrow 0$
    for $t \in T$ **do**
        Pick the number of actions $\bar{\alpha}_t$ for day t according to eq. 2 with $\hat{\omega}$
        $\bar{\alpha} \leftarrow \bar{\alpha} + \bar{\alpha}_t$
    end for
    Calculate the customer's utility υ for the month using $\bar{\alpha}$ according to eq. 4
    Determine if the customer *churn*'s using υ with equation 5
    if *churn* **or** $T >= T_2$ **then** break
    else $T \leftarrow T + 1$ month
    end if
end while

Algorithm 2 Simulate N Initial Customers with growth rate γ from $T_1 \rightarrow T_2$

$t \rightarrow T_1$
Simulate N Customers $t \rightarrow T_2$
while $t <= T_2$ **do**
    $T \leftarrow T + 1$ month
    $n \leftarrow N[(1+\gamma)^{(t-T_2)} - 1]$
    Simulate n new Customers from $t \rightarrow T_2$
end while

3.4 Additional ChurnSim Model Components

The model components described in this section were used in [4]. These add greater realism in certain aspects of customer churn analysis but they do not significantly change the simulation dynamics.

3.4.1 Day of Week Behavioral Fluctuation

The ChurnSim simulation can include a coefficient to either increase or decrease behaviors on weekdays versus weekends. This allows the model to realistically simulate the daily ebb and flow of customer behavior on different types of products: Consumer products tend to see higher behavior rates on the weekend, while business products see more activity on weekdays. The simulation allows definition of two constants $\psi_{weekday}$, $\psi_{weekend} \in [-1, 1]$. For any given date in the simulation, a day-of-week scaling coefficient Ψ_t is drawn from a uniform distribution U with a range defined by:

$$\begin{aligned} \Psi_t &\sim U(1 - \psi * 0.1, 1 + \psi) \Leftarrow \psi > 0 \\ \Psi_t &\sim U(1 + \psi, 1 - \psi * 0.1) \Leftarrow \psi < 0 \end{aligned} \tag{6}$$

The -0.1ψ terms in equation 7 allows a small chance that any given day will go against the trend. The scaling coefficient Ψ_t for a specific date is drawn once and then all customer's actions are influenced by it on that date. Equation 2 for drawing the number of actions $\bar{\alpha}_t$ is modified into:

$$\bar{\alpha}_t \sim P(\Phi_t \hat{\omega}) \tag{7}$$

In this way every customer's actions are still random, but collectively they will be influenced by the day of week leading to realistic looking weekly fluctuations.

3.4.2 Product Channels

Subscription products may be accessed via multiple channels, for example on the web or via different types of mobile devices. ChurnSim allows simulated customers using the product to be on different channels and to have channel dependent behavior patterns. Each product channel may have its own version of the mean behavior vector $\bar{\mu}$ and covariance matrix Σ from equation 1. Each new customer has a channel selected randomly with the percentage of the population to come from each channel set by configuration. On creation, the customer draws its behavioral mean vector $\bar{\omega}$ from the distribution for their channel. After that, the simulation of each customer is identical - they have the same utility and churn equations, etc.

3.4.3 Customer Satisfiability Coefficient

To increase the unpredictability of customer behavior in the simulation, every customer has a random satisfiability coefficient. The coefficient is greater than zero and is drawn from a range centered around one. The satisfiability coefficient multiplies the utility the customer has received from their behavior in equation 4, whenever the customers total utility is greater than zero. If the customer utility is less than zero the utility is divided by satisfaction propensity making it less negative. If the satisfaction propensity for a customer is ζ then equation 4 becomes

$$\begin{aligned} \upsilon' &= \sum_i \mu_i u_i (1 - e^{-c\alpha_i/\mu_i}) \\ \upsilon &= \zeta \upsilon' \Leftarrow \upsilon' > 0 \\ \upsilon &= \upsilon'/\zeta \Leftarrow \upsilon' < 0 \end{aligned} \tag{8}$$

Each simulated customer's satisfaction propensity is picked randomly on a uniform exponential scale. The configurable parameters are the satisfaction propensity exponent base β_ζ and a scale is κ_ζ; the satisfaction propensity is drawn according to:

$$\zeta \sim \beta_\zeta^{U(-\kappa_\zeta, \kappa_\zeta)} \tag{9}$$

where U in equation 9 is a uniform distribution.

3.4.4 Customer Age

To allow simulation and analysis of customer demographic traits the simulation includes the age of each customer (for business products this is interpreted as the age of a firm.) The age of the customer is drawn from a uniform distribution ranging from a minimum age to a maximum. If the minimum age is χ_{min} and the maximum age is χ_{max} then age is drawn from $U(\chi_{min}, \chi_{max})$ where U is a uniform distribution.

Age affects the simulation by adding a bias to the otherwise uniform satisfaction propensity described in section 3.4.3. An additional parameter τ determines the impact of age on customer satisfaction by modifying equation 9 with a bias added to the random component of the satisfiability coefficient. The bias is linearly proportional to the customer's age divided by the allowed range. If the customer's age is a then equation 9 becomes:

$$\begin{aligned} \Delta &= \tau \frac{a - \chi_{min}}{\chi_{max} - \chi_{min}} \\ \zeta &\sim \beta_{\zeta}^{U(-\kappa_{\zeta}, \kappa_{\zeta}) + \Delta} \end{aligned} \tag{10}$$

The age coefficient τ would normally be set to a small value relative to the satisfiability scaling coefficient κ_{ζ}. With such settings age makes a small but noticeable different in the simulated customers' churn probabilities.

3.4.5 Customer Location

The simulation also gives each customer a randomly assigned location drawn according to a specified distribution. However, location makes no difference in the simulation - it is included for educational purposes so students can experience how some categories can be misleading. For locations where a large number of customers reside, the measured churn rate will tend to the mean. For locations with few customers, the average churn rate may diverge significantly from them mean due to random variation; careful analysis should reveal that such measurements lack statistical significance.

In contrast, the product channel described in section 3.4.2 is a categorical variable that is really associated with true differences in the simulated customer behavior. This is not meant to imply that channel is always associated with customer behavioral differences and location is not - the distinction is for educational purposes so that the simulation includes one categorical variable that really does make a difference and one that does not.

3.5 Advanced ChurnSim Features

This section describe additional ChurnSim model features that can be used to make more realistic and complex simulations. These features were added to the simulation after the publication of [4].

3.5.1 Multi-User Accounts

Customer accounts can be defined to be multi-user. For multi-user accounts, the average number of users per month is selected according to equation 1 like any other behavior: The number of account users has a mean in $\bar{\mu}$, and the number of users at an account may also be correlated with other behaviors via the covariance matrix Σ. For multi-user accounts, *the other action rates in $\bar{\mu}$ are interpreted as rates per user.* In the simulation for multi-user accounts, on each day of simulation the number of users is selected before the number of other actions is selected. However, he number of users is not divided by 30 to make a daily rate; it is already considered to be an average per day. After the number of users is selected for a given day, the number of other actions taken on the day are multiplied by the number of users. Instead of the single user action count equation (2) the expected customer rate is multiplied by the day's user count:

$$\begin{aligned} u &\sim P(\omega_u) \\ \bar{\alpha_t} &\sim uP(\hat{\omega}) \end{aligned} \tag{11}$$

where ω_u is the expected number of users, $\hat{\omega}$ is the customer's daily action rate vector not including the user rate, and P indicates a draw from the Poisson distribution (as in equation 2). If the multi-user option is used in combination with day of week scaling (section 3.4.1) then the day-of-week scaling multiplier affects both the number of users and the actions per user according to equation 7.

3.5.2 Action Values

A ChurnSim simulation can also simulate actions that have a numeric value associated with them. For example the value may be the monetary value of a transaction or the length of time that a streaming media was played. Action value distributions are specified in the same covariance matrix Σ used for the action counts and number of users. When a customer is created, the mean values $\bar{\omega}$ are sampled jointly according to equation 1 and this includes the expected value of events. The following procedure is used for simulating events with values: First, the number of actions taken by customers for a given day is drawn from equation 2 based on the mean rate $\bar{\omega}$ as usual. Then for each event the value ν associated with the event is drawn according to a log-normal distribution :

$$\nu_{event} \sim e^{N(\log(\omega_{event},1))} \tag{12}$$

where ω_{event} is the mean customer *value* for such events, and N is the normal distribution: Draw a normal random variable having a mean given by the log of the expected event value and unit standard deviation, and take the base of the natural logarithm e to that power. With this formulation the expected value of the event is the parameter ω_{event} but it follows a natural looking, strictly positive distribution that requires only a single parameter to define. (An enhanced simulation could allow separately controlling the standard deviation of each event value.)

3.5.3 Product Plans, MRR and Limits

A ChurnSim simulation can include a list of available plans that have different prices and limits on monthly users and/or actions. Prices are expressed in Monthly Recurring Revenue or MRR. If a simulation contains plans with different prices, it is required that the model utility coefficients $\bar{\mu}$ described in section 3.2.2 has a negative coefficient μ_{MRR}. The MRR utility coeficient is applied every month to the customer's MRR and added to the utility function, equation 4 which becomes:

$$\upsilon = \sum_i \mu_i u_i (1 - e^{-c\alpha_i/\mu_i}) - \mu_{MRR} MRR \tag{13}$$

Note that the MRR term does not include a hedonic adaptation mechanism like other behaviors (see section 3.2.2 and equation 3).

The plan limits constrain the customer's number of users and/or action counts given by equation 2. For a single action i with monthly rate ω_i and a plan limit λ_i the count of monthly actions becomes:

$$\alpha_i \sim \min(\sum_t P(\omega_i), \lambda_i) \tag{14}$$

In equation 14 the sum over t represents the daily generation of events described in section 3.2.1, and in the simulation the limit is checked each day and further actions are dropped after the limit is reached.

A customer's initial plan is picked uniformly at random from among those plans where the customer's behavioral rate(s) is(are) at least 1/3 of the plan limit(s) and less than 3× the plan limit(s). This mechanism ensures that a customer will not start out on a plan with a limit that is orders of magnitude more or less than the customer's own typical behavior rate. However, there is some randomness and some customers may still start out on moderately inappropriate plans. Customers in the simulation may change plans as described in the next section.

3.5.4 Upgrade & Downgrade

If a simulated product includes multiple plans then it is possible for customers to upgrade or downgrade their plan. Upgrade and downgrade are simulated with the same dynamics as churn: The probability of upgrading or downgrading is based on the customer utility using a sigmoidal function:

$$P_{upgrade} = \frac{1}{1 + e^{-\xi_{up}\upsilon + \Delta_{up}}} \tag{15}$$

$$P_{downgrade} = 1 - \frac{1}{1 + e^{-\xi_{down}\upsilon + \Delta_{down}}} \tag{16}$$

In equations 15 and 16 υ is the utility from equation 4, the ξ are scaling constants ($\xi_{up/down} > 0$) and Δ's are offsets. The equations are structured so that upgrade probability increases with increasing customer utility and the downgrade probability decreases with increasing utility.

At the end of each month in the simulation where the customer does not churn, the customer is first given the chance to upgrade. For a customer to upgrade to a plan with a higher rate limit it is also required that the customer's average rate for the behavior be at least 50% of the new plan limit. If the customer does not upgrade, another random draw determines if they downgrade. If a customer downgrades then they will switch to the next lower plan (in terms of price and limits) than their current plan.

3.5.5 Add on Products

Subscription products can also include "add-ons" which are additional product components sold separately. Every add-on in the simulation includes a price and a limit on one or more actions. The action limit of the add-on is generally higher than a pre-existing limit in the base plan subscription: If a customer takes an add-on they receive a higher limit on their action (equation 14) and pay an additional cost and loss of utility (equation 13.).

Add-ons are chosen and discarded as part of the upgrade/downgrade logic: If the customer does not upgrade or downgrade their plan, a second draw is taken with the same probability given by the upgrade equation 15 to determine if the customer chooses a new add-on. To choose an add-on, a customer must have a behavior rate that is within 50% of the limit in the plan they are buying. This is to prevent the customer from upgrading to something with a limit that drastically exceeds their own behavior (the add-on can still exceed their usual behavior by as much as $2\times$, so customers can be somewhat irrational.)

If the customer has not received an upgrade or downgrade on their base plan, or added a new add-on, they may cancel an existing add-on product. This occurs according to a separate draw with the same probability as a downgrade (equation 16).

3.5.6 Billing Periods

Plans may include different billing frequencies, measured in months. Typically these would include monthly and annual plans and a plan with a longer (less frequent) billing period would sell at a lower price (this is specified for each simulation along with the plans.) It is assumed that the subscriptions are paid in advance at the start of each billing period and the same billing period applies to both the base product subscription and any add-on products. Multi-month billing periods affect the simulation of customer churn through the following logic:

1. The simulation described in section 3.3 proceeds as usually and at the end of every month the customer produces a churn intention *indicator* in the usual way (according to the monthly utility and equation 5.) However, for multi-month billing periods this is only an indicator of intent to churn and does not mean an immediate churn.
2. If a customer has had any positive churn result in any month of their current subscription then they churn when their subscription completes. There is no mechanism in the simulation to undo an intention to churn.
3. If a customer has a billing period between 2 and 6 months (inclusive) then they will churn immediately (before the end of their current subscription term) if they receive a second positive churn indicator result within one subscription term
4. If a customer has a billing period of 7 or more months then they will churn immediately (before the end of their current subscription term) if they receive three positive churn results within one subscription term.

By following this logic a longer billing period will increase the length of time a customer goes without churning. However, significantly unhappy customers (very low utility) will be able to churn before the entire subscription time is up. This is analogous to a customer being so unhappy that they call and complain to receive money back before their current subscription term completes.

At instantiation every customer has a maximum billing period that they will accept, which is chosen uniformly at random from the available billing periods. The customer's initial plan is chosen randomly from among those plans that are not above their maximum billing period (and they are otherwise eligible as described in section 3.5.3).

During simulation a customer may change their billing period to a longer billing period as part of the logic for upgrades and add-ons (sections 3.5.4): If the customer has been determined to take an upgrade (via a random draw with probability given by equation 15) but there is no suitable higher limit plan for them then they will instead lengthen their billing period. The billing period can increase to any available higher billing period that is not above the customer's maximum billing period. Like an upgrade, extending the plan billing period is an action that tends to be taken by customers with high utility from use of the product. Similarly, if a customer is determined to downgrade (equation 16) and there is no lower level product plan to downgrade to then the customer will reduce their billing period to an available lower billing period chosen randomly. Reducing the billing period increases the probability of future churn by eliminating the "waiting period" introduced by the billing period.

3.5.7 Discounts

Subscription products often have discounts on the price. ChurnSim can include discounts which are randomly assigned with a fixed probability when the customer plan is selected (either on customer creation or upon an upgrade or downgrade.) If a discount is determined to apply the % amount of the discount is chosen randomly in a fixed range $[\delta_{min}, \delta_{max}]$ in fixed steps of δ_{min}. For example, from 5% to 50% in 5% steps. The discount reduces the customers base plan MRR by the indicated discount amount:

$$MRR_{discount} = (1 - \delta)MRR \tag{17}$$

Discounts can also be set to have an enhanced affect on satisfaction. This represents the psychological satisfaction many people feel from knowing they are getting a good deal. Enhanced discount satisfaction modifies the utility from MRR by an extra term proportional to the discount amount. The utility equation with MRR (eq. 13) becomes:

$$\upsilon = \sum_i \mu_i u_i (1 - e^{-c\alpha_i/\mu_i}) - \mu_{MRR} MRR + \delta MRR \mu_{MRR} \zeta_\delta \tag{18}$$

where μ_{MRR} is the utility coefficient for MRR, δ is the discount percent (δMRR is the discount amount) and ζ_δ is the coefficient for discount satisfaction.

3.5.8 Acausal Churn

Sometimes even the very best customers churn for no apparent reason. This phenomena can be include in a ChurnSim model by defining an "acausal" churn probability. A high rate of acausal churn makes churn more difficult to predict using a model.

When an acausal churn probability is set for a simulation every customer may churn every month with the acausal churn probability (or have an intention to churn, in the case of multi-month billing periods described in section 3.5.6). If the customer does not churn acausally, then the normal logic for churn takes place: The intention to churn occurs with the probability given by equation 5 in section 3.2.3. If a customer is on a plan with a billing period greater than one month, then an acausal intention to churn functions identically to a utility based intention to churn, as described in section 3.5.6.

4 Discussion

This report provides a full description of the customer churn simulation used in [4]. Also, the report describes new advances in churn simulation techniques. Together these components allow simulations that can match a wide variety real customer behaviors.

Multiple ways in which a simulation of customer churn can realistically reproduce customer behavior were demonstrated. Realistic customer churn behaviors include individual behaviors having a long tailed distribution of customer activity rates, churn measurements showing hedonic adaptation to high rates of engaging behaviors, and correlation between engaging and disengaging product interactions such that disengaging events are revealed by ratio measurements. Monthly recurring revenue (MRR) in particular is correlated with low churn rates through a single variable analysis, but higher churn when viewed in relation to truly engaging behaviors. In terms of churn rates a realistic simulation can reproduce the scenario where enterprise products have MRR churn that is less than account based churn, net retention is greater than 100% and long billing periods have a low monthly churn rate but a high rate of churn upon renewal.

The reproduction of realistic elements of customer churn dynamics give students and researchers confidence that a churn simulation is meaningful enough to understand true churn behavior. With such a realistic simulation it is possible to analyze aspects of machine learning methods when applied to customer churn. Examples were shown analyzing the variability of feature significance and also the variability of model prediction accuracy and how it relates to the rate of "acausal" (unexplainable) churn.

One key challenge to address in future work is the complexity of setting model parameters to match a particular case study. At present the only method available is manual experimentation in the large parameter space. Automated parameter tuning is difficult due to the large number of parameters and the wide range of realistic churn characteristics one typically wishes to match in a single simulation. Automated tuning of simulated churn parameters is an attractive area for future research.

References

[1] Burez, J., & Van den Poel, D. *Handling class imbalance in customer churn prediction.*, Expert Systems with Applications, 2009, 36(3), 4626-4636.

[2] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang & Wojciech Zaremba. *OpenAI gym.* arXiv preprint:1606.01540, 2016 arxiv.org

[3] Chen, T., & Guestrin, C. *Xgboost: A scalable tree boosting system.* In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, August, pp. 785-794

[4] Gold 2020, *Fighting Churn With Data*, Manning Publications, Long Island, NY 2020

[5] Gold 2019, *fight-churn* [Online] Fighting Churn With Data Python and SQL Source Code. Available: https://github.com/carl24k/fight-churn, 2019

[6] Huang, B., Kechadi, M. T., & Buckley, B. *Customer churn prediction in telecommunications. Expert Systems with Applications, 2012, 39(1), 1414-1425.*

[7] Eugene Ie, Chih-Wei Hsu, Martin Mladenov, Vihan Jain, Sanmit Narvekar, Jing Wang, Rui & Craig Boutilier *RecSim: A Configurable Simulation Platform for Recommender Systems* arXiv preprint:1909.04847, 2019 arxiv.org

[8] Scott Lundberg & Su-In Lee, *A Unified Approach to Interpreting Model Predictions* Advances in neural information processing systems 30 (2017).

[9] Vafeiadis, T., Diamantaras, K. I., Sarigiannidis, G., & Chatzisavvas, K. C. . *A comparison of machine learning techniques for customer churn prediction. Simulation Modelling Practice and Theory, 2015, 55, 1-9.*

Appendix A ChurnSim Configuration

The following section summarizes the details of parameterizing a ChurnSim model. The complete parameterization of a model consists of the following:

1. A YAML file containing parameters.
2. One or matrices of behavioral model parameters in a CSV file.
3. One CSV file containing a table of the product plans; and optionally a second CSV file summarizing the add-on products.

All of the options and parameters are summarized in Table 3. Examples of all of the configurations can be found in [5] in the folder `fightchurn/churnsim/conf`.

A.1 Parameter YAML Files

Parameters are all stored a single YAML file where the file name is a name for the model it describes, `<model>.yaml`. The YAML file includes all scalar parameters as well as vectors describing the utility coefficients, the population channels, and the population location options. There is a `default.yaml` file containing defaults for the scalar parameters - utility coefficients, channels and locations must be provided in the model specific model YAML file.

A.2 Behavior CSV Files

Each behavioral CSV files provide the mean vector and covariance matrix for the behaviors (section 3.2.1) of the population defined by one channel (section 3.4.2.) The name of the behavioral file must be `<model>_<channel>.csv`. The rows in the file should all start with a behavior name, in the same order as the utility coefficient list in the `<model>.yaml` file. The first column in the file lists the mean values; there is an option to provide maximum values for each behavior rate in the second column; the remainder of the file is a covariance matrix for the behavior rates. The covariance matrix can be (and usually is) specified as a correlation matrix - the appropriate covariance will be created from the mean vector and correlations at simulation runtime.

A.3 Plan and Add-On CSV Files

The file <model>_plans.csv lists the product plans (section 3.5.3), one row per plan with the following columns:

1. Plan name ("plan")
2. MRR per month ("MRR")
3. Plan billing period in months ("bill_period")
4. Additional columns are named for behaviors and provide the limits associated with the plan.

An optional file of add-on products (section 3.5.5) may be included, named <model>_addons.csv. The format of the add on file is the same as the plan file, but without the billing period (add-ons always follow the billing period of the base plan.)

Description	Report Variable	File/Parameter	Section
Behavior Rate Means	$\bar{\mu}$	<model>_<channel>.csv	3.2.1, 3.4.2
Behavior Rate Covariance	Σ	<model>_<channel>.csv	3.2.1, 3.4.2
Behavior Rate Exponent Base	a	<model>.behave_exp_base (yaml)	3.2.1
Utility per Action	$\bar{u}$	<model>.utility (yaml)	3.2.2
Hedonistic Adaptation Scale	c	<model>.util_contrib_scale (yaml)	3.2.2
Churn Rate Scale	ξ_{churn}	<model>.churn.scale (yaml)	3.2.3
Churn Rate Offset	Δ_{churn}	<model>.churn.offset (yaml)	3.2.3
Initial Number of Customers	N	<model>.init_customers (yaml)	3.3
New Customer Growth Rate	γ	<model>.growth_rate (yaml)	3.3
Weekday Action Rate Scale	$\psi_{weekday}$	<model>.weekday_scale (yaml)	3.4.1
Weekend Action Rate Scale	$\psi_{weekend}$	<model>.weekend_scale (yaml)	3.4.1
Satisfiability Random Sacle	κ_{ζ}	<model>.satisfy_scale (yaml)	3.4.3
Satisfiability Exponent Base	β_{ζ}	<model>.satisfy_base (yaml)	3.4.3
Minimum Customer Age	χ_{min}	<model>.min_age (yaml)	3.4.4
Maximum Customer Age	χ_{max}	<model>.max_age (yaml)	3.4.4
Age satisfiability coefficient	τ	<model>.age_satisfy (yaml)	3.4.4
Customer Location Distribution	NA	<model>.country (yaml)	3.4.5
Acausal Churn Rate	NA	<model>.acausal_churn (yaml)	3.5.8
User Rate Mean & Covariance	μ_{user}, Σ	<model>_<channel>.csv key: user	3.5.1
Action Value Mean & Covariance	μ_{event}, Σ	<model>_<channel>.csv key: <action>_value	3.5.2
Utility of MRR	μ_{MRR}	<model>.utility.mrr (yaml)	3.5.3
Plan MRR	MRR	<model>_plans.csv	3.5.3
Plan Action Limits	λ_{action}	<model>_plans.csv	3.5.3
Plan Billing Periods	NA	<model>_plans.csv	3.5.6
Add-On MRR	NA	<model>_addons.csv	3.5.5
Add-On Action Limits	NA	<model>_addons.csv	3.5.5
Upgrade Rate Scale	ξ_{up}	<model>.upgrade.scale (yaml)	3.5.4
Upgrade Rate Offset	Δ_{up}	<model>.upgrade.offset (yaml)	3.5.4
Downgrade Rate Scale	ξ_{down}	<model>.downgrade.scale (yaml)	3.5.4
Downgrade Rate Offset	Δ_{down}	<model>.downgrade.offset (yaml)	3.5.4
Discount Probability	NA	<model>.discount_prob (yaml)	3.5.7
Minimum Discount	δ_{min}	<model>.min_discount (yaml)	3.5.7
Maximum Discount	δ_{max}	<model>.max_discount (yaml)	3.5.7
Discount Utility Scale	ζ_{δ}	<model>.discount_satisfy (yaml)	3.5.7

Table 3: ChurnSim Configurations