dataiku

HOW TO: ADDRESS CHURN WITH PREDICTIVE ANALYTICS

Understand, Predict, and Minimize Customer Loss

GUIDEBOOK

www.dataiku.com

SUMMARY

ABOUT THE GUIDEBOOK

This guide is intended to provide a high-level overview of the process required to predict churn (specifically non-subscription churn) for businesses in any industry. By the end, readers should have an understanding of the steps and work required to build a scalable solution for addressing customer churn and should feel ready to get started on their own churn prediction project.

In addition to covering the basic requirements (from formulating a definition and action plan for churn to deploying predictive models into production), this guide will delve briefly into some more advanced concepts for those with a more technical understanding. The advanced portions are also intended for team leaders tasked with a churn prediction project who may want to pass along more specific direction and best practices to their team.

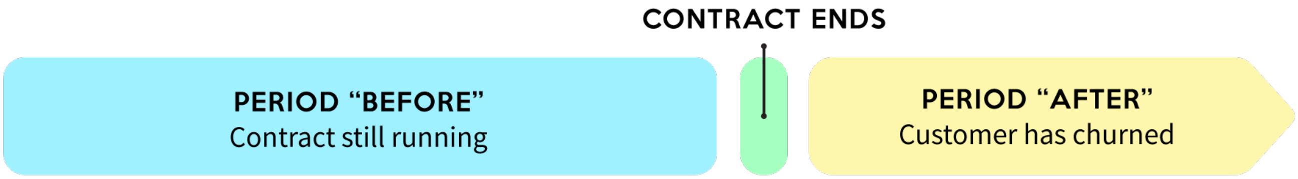
What and Why?

It's simple: churn (or attrition) is when customers leave, and companies in nearly every industry have to address it because it has the power to plateau the growth of any businesses even if that business is gaining customers quickly. The most successful companies address it by building predictive models that accurately predict churn; then they take action by building targeted marketing campaigns around preventing it or by making product changes that combat churn.

SUBSCRIPTION CHURN AND NON-SUBSCRIPTION CHURN

There are two basic types of churn: **subscription churn and non-subscription churn**

Subscription churn happens in businesses where users or customers are on contract for a set period of time (monthly, annually, etc. - think cable, network, or phone providers), and customers choose not to come back after that contract is up. It is easy to define, predict, and prevent since there's a clear, defined window with risk of churn where marketing activities can be focused.

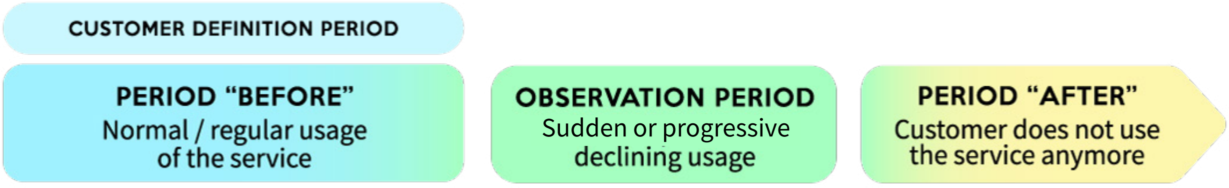


Non-subscription churn happens when users or customers can end their relationship with your business at any time - they come and go at will. A customer may gradually over time reduce their purchase frequency, or they may all of a sudden never buy again. This guide will focus on the process for preventing non-subscription churn because:

1. Non-subscription churn is a prime candidate for prediction since there is no set renewal time and because it's not clear when this audience will need or be receptive to marketing materials aimed at preventing churn.

2. Non-subsription churn requires collaboration among several teams to predict accurately - the business side generally defines churn (lack of action after weeks, months, or years), and then it's a back-and-forth, iterative process with data teams to arrive at the right model.

Note that some industries might deal with a combination of both types of churn, for example, banks where basic services are non-subscription, but credit cards with annual fees might be subscription-based.



How?

Tackling churn by successfully predicting those that will churn is as easy as following **the seven fundamental steps to complete a data project[1]**. Some particular nuances and details for churn prediction:

1 Understand the Business

How will your specific business define churn? This step is crucial - defining a churn period that is too long risks creating predictive models with artificially low churn rates, not capturing enough people and defeating the purpose of predictive modeling. **But defining a churn period that is too short makes it difficult for marketing teams to evaluate churn prevention campaigns** because they ultimately can't distinguish between organic actions (users or customers who would have come back anyway without intervention) and effective campaigns.

It's also a good idea **to do basic analysis upfront (unsupervised/clustering)** to decide which users should even be considered in the churn analysis. For example, if someone used the product or service only one time, are they considered a churner after that? Or is there some minimum threshold after which a user should be considered and included in churn analysis?

Additionally, before moving on to any other steps, **it's essential to decide first what the churn predictions will be used for.** The marketing and product teams should be fully looped in and have a concrete plan for using predictions to prevent churn. **Otherwise, there is a risk of wasting time and resources modeling churn predictions that go unused**. Predictions can be used for short-term solutions like marketing campaigns **to re-engage likely churners** (more on this later), or they can help uncover potential deeper drivers of churn that can be addressed long term. For example, maybe there is an issue with the product that is blocking customers' ability to come back easily or there are in-product improvements (or potential new features) to be made to prevent attrition.

2 Get Your Data

The minimum data required to predict churn is simply some form of customer identification and a date/time of that customer's last interaction. This data, though not incredibly detailed, would allow you to build models to predict churn at a basic level.

However, the reality is that **adding additional data on top of this minimum data set is recommended and highly encouraged.** The more data included, the better the churn predictions will be, so if available, also include things in the dataset like static demographic information about users, details on specific types of user actions, etc. The more sources, the better.

ADVANCED/EXPERT: We are typically given two datasets to start with:

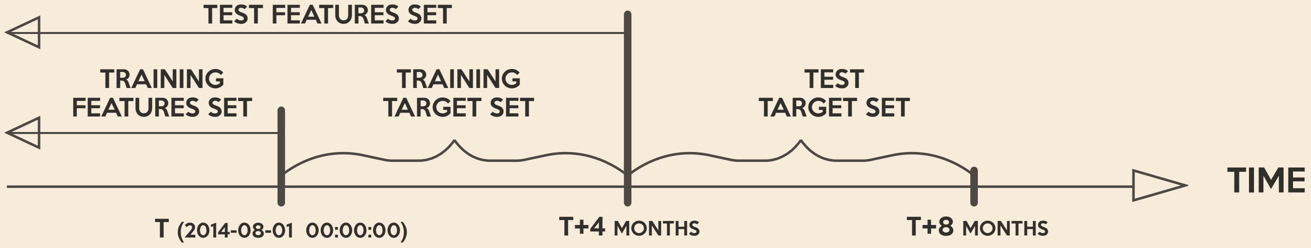
- a log/transaction dataset `events` , it is the events on your website, what page users see, what product they like or purchase, with schema `user_id | event_timestamp | event_type | product_id.`
- a products dataset `product`, is a lookup table containing the product_id and information about its category and price.

This is a very generic setup, and it is usually the bare minimum to start a churn modeling project.

3 Explore and Prepare Data

Remember that **this step of the process can account for up to 80% of the total time spent on the project,** so don't be discouraged as you get your data into a useable format. Take time to ensure you understand what all the different variables in your data mean **before moving on to cleaning up different spellings or possibly missing data to ensure everything is homogeneous.** Thoroughly exploring and cleaning will save time in subsequent steps, particularly when it comes time for prediction.

ADVANCED/EXPERT: Firstly, we need to define what is a churner. Here, we will consider someone as a churner if they haven't taken any action on the website for four months.
The target churner variable is built by looking ahead over a four month period from a reference date and flagging the customers if they purchased a product in this time frame. We will also restrict our dataset to customers who made at least one purchase in the four months previous to the reference date for a total amount greater than 30. That way, we only look at the high-value customers. One common pitfall in churn modelling is an improper evaluation scheme. Here is how we can properly evaluate our model with a time-based train and test split.



We can now create our target (churner or not) in SQL (Postgres):

We create a table `events_complete` as follows:

```
SELECT *
FROM events e
LEFT JOIN products p
ON e.product_id = p.product_id
```

We restrict the training set to customers that spend more than $30 by creating the table `train_active_clients:`

```
SELECT
  *
FROM (
  SELECT
    user_id,
    SUM(price::numeric) AS total_bought
  FROM
    events_complete events
  WHERE
        event_timestamp <  TIMESTAMP '2014-08-01 00:00:00'
    AND event_timestamp >= TIMESTAMP '2014-08-01 00:00:00' - INTERVAL '4 months'
    AND event_type = 'buy_order'
  GROUP BY user_id
) customers

WHERE
  total_bought >= 30
```

We are now ready to define our churner target for the training set by creating the table `train` :

```
SELECT
  customer.user_id,
  CASE
    WHEN loyal.user_id IS NULL THEN 1
    ELSE 0
  END as target
FROM
  train_active_clients customer
  LEFT JOIN (
    -- Users that actually bought something in the following 4 months
    SELECT distinct user_id
    FROM events_complete
    WHERE event_timestamp >= TIMESTAMP '2014-08-01 00:00:00'
      AND event_timestamp <  TIMESTAMP '2014-08-01 00:00:00' + INTERVAL '4 months'
      AND event_type = 'buy_order'
    ) loyal ON customer.user_id = loyal.user_id
```

4 Enrich Data

If you're working with a more advanced data set than simply customer identification and date/time of last interaction (which is, as mentioned, highly recommended for better prediction), **this is the time to enrich that data and join it to get down to the essentials.** For example, if you have one data set with customer identification and date/time of last interaction and another with customer identification and demographic information, you'll want to join these into one set of data.

While certain data can enrich your analysis, be aware that group/pivot operations could generate hundreds or thousands of features if done blindly. So ensure the data you're using has ultimate value for your defined churn goals (see step 6 before adding too many features), and get ready to build some very large data sets!

ADVANCED/EXPERT: We enrich our dataset by creating features for each user. Some examples of features you might create are action-based or time-based. For example:

- Number of actions in the last five,10,30, etc., days
- Average length of those actions
- Average and max "sleep" time between two actions
- Distribution of hourly activity
- Distribution of day of week activity

Here is a first SQL script to generate some features (number of product purchased per category, total amount spent, time since last purchase,...) for the train set:

```
SELECT train.*
    , nb_products_seen
    , nb_distinct_product
    , nb_dist_0 , nb_dist_1, nb_dist_2
    , amount_bought , nb_product_bought
    , active_time
FROM train
LEFT JOIN (
    -- generate features based on past data
    SELECT user_id
        , COUNT(product_id) AS nb_products_seen
        , COUNT(distinct product_id) AS nb_distinct_product
        , COUNT(distinct category_id_0) AS nb_dist_0
        , COUNT(distinct category_id_1) AS nb_dist_1
        , COUNT(distinct category_id_2) AS nb_dist_2
        , SUM(price::numeric * (event_type = 'buy_order')::int ) AS amount_bought
        , SUM((event_type = 'buy_order')::int ) AS nb_product_bought
        , EXTRACT(
           EPOCH FROM (
            TIMESTAMP '2014-08-01 00:00:00' - MIN(event_timestamp)
           )
          )/(3600*24)
            AS active_time
    FROM events_complete
    WHERE event_timestamp < TIMESTAMP '2014-08-01 00:00:00'
    GROUP BY user_id
    ) features ON train.user_id = features.user_id
```

Like earlier, you can replicate this feature creation for the test set by just changing the timeframe.

5 Get Predictive

When building a predictive model, one has to be careful that it will actually learn what you want. For instance, one of the common pitfalls for a churn modelling project is to train your model on both past and future events. **To avoid this common mistake, you need to put yourself in the position you'll be in when your model will be deployed into production:** What data will be available to you? When would you like your prediction to be: for next week, next month?

An important part of the predictive process is the interaction and iteration between predictive modeling and feature engineering. In step 4, you enriched your data and generated features. **Now it's time to see if the features you've added are actually valuable to your model.** Try keeping the feature set relatively small at first and then run your model(s) to evaluate performance. Little by little, continue to add features and evaluate their effect on the accuracy of the model.

When in this design stage testing features and iterating, note that it's not necessary to run complex models at this point in time. **Instead, focus on optimizing for the right features first and running simple models.** Later, once you know you have the best features, you can optimize and find the best model. This will save time and resources in the long run.

Regarding finding the best model, another critical step in this process is choosing how you evaluate which model is best. **You want to choose an evaluation metric that fits with the business need.** For example, maybe the goal is to identify everyone who has higher than N likelihood of churning, and the marketing or product teams will address all of those individuals. But if the marketing team only has the budget to address a small portion of those individuals, evaluating lifetime value to prioritize the most valuable churners would be beneficial.

If you're a beginner when it comes to machine learning and algorithms, **you can use a tool like Dataiku Data Science Studio (DSS) to run open source algorithms to predict churn in a clickable interface without having to write any code.**

ADVANCED/EXPERT: We should now be ready to start spinning our favorite machine learning models. But before that, we need to consider what data to train our model on. If volume is not an issue, we can train on all the available month partitions. Otherwise, here is a little trick: take the last available partition.

For example, in our previous example,if the month is August, the last available data is February since you need `Y = 6` months to know if someone is a churner or not. But you can also take the data of the current month from the previous year to add some seasonality in the model. So in this case, you would take data from February 2016 and August 2015 together as your train set.

Last but not least as our train dataset contains a row for each pair (User, month) and so we have a temporal dependence between two rows with same user. Hence we choose to do a time-based split for the train/test.

If you don't have enough data to go back far enough for a time-based split, for churn, it's not recommended to use a random split - this is a common mistake that can risk overfitting the model and will not allow churn predictions to be generalized to the future (which is the primary goal of your project!). Another factor to watch for with churn prediction is making sure that the distribution of the inputs used as predictors don't change between training and production stages (called covariate shift). For example, if you're using total purchases by a customer, the input should be purchases over a specified time period instead of total all-time purchases.

6 Visualize

Now that you have explored and know your data by digging in, cleaning, and enriching it, it's time to visualize. **Visualization is an important step in the process because it allows a way for end-users - in the case of churn, this is the marketing team and/or the product team - to consume the data quickly and easily.**

Ensure you are aligned with your end-users here and give them visualizations in a format that is actually helpful for them. Some helpful visualizations for marketing and product teams with regard to churn might be:

- The evolution of churn over time and targeted churners
- Which product features have an impact on churn
- Descriptive statistics of those key features for easy reference or visual simulations illustrating how changing features would impact churn probability
- Additional insights about the chosen churn model

Also, you may consider exploring visualizations not as a product for end-users **but as a way to uncover additional insights and trends you may want to explore or explain with predictive modeling.** For example, maybe by creating a churn visualization on a map, you find that certain geographies churn at a higher rate than others, and you would like to explain why.

7 Iterate and Deploy

This is where the interplay between data science and business is strongest - work together to determine if the model is actually effective. In particular, **ensure models are sufficiently generic, which means using training, validation, and testing sets that are not specific to a certain time period or to a certain type of customer.** For example, you would not want to train or test based on a data set from a time period where there was perhaps a pricing change or some other factor that caused churn rates to be different than usual.

The most important part of a churn project is deploying a churn solution into production. Looking at churn one time and evaluating models but not taking any real action to set up a continuous churn prevention strategy doesn't do much good!

ADVANCED/EXPERT: Consider the underlying infrastructure when it comes to churn prediction in production. You will need enough computing resources to train the chosen models and dedicated servers if you plan to deploy a model in production via REST APIs. You might choose to score churn realtime if, for example, potential churners visiting your website will be shown a special offer or prices.

END NOTES

[1] http://blog.dataiku.com/2016/07/06/fundamental-steps-data-project-success

[2] http://www.slideshare.net/PierreGutierrez2/beyond-churn-prediction-an-introduction-to-uplift-modeling?ref=https://www.linkedin.com/

What are the pitfalls?

DEVELOP THE RIGHT TEAM.

The team tasked with predicting and addressing churn should be collaborative and cross-functional, including data specialists, marketing, business, etc. Involving just one of these profiles won't produce effective churn prediction results. Note that predicting churn doesn't require any specialized machine learning skill sets unlike other types of advanced learning like recommendation engines, which may require team members with specific backgrounds or profiles. Testing and building these types of models is accessible and could be executed by team members who are proficient in SQL.

TARGET THE RIGHT AUDIENCE.

When predicting churn, the tendency is to target the people who are the most likely to churn. However, the real users to affect should be those who are likely to churn AND those who are most likely to positively change their behavior (i.e., not churn) when they are targeted with an action.

MAKE CHURN PREVENTION MORE THAN JUST A ONE-TIME PROJECT.

As with any data project, one of the most important steps (but also one of the most common places where teams go wrong) is making sure that there's a way to incorporate it into the business in a scalable way. In other words, in order to get value out of churn prevention efforts, it needs to be put into production and made part of regular processes.

What next?

Once you have a good churn prediction model in place, the job is only half complete. The final (and perhaps most important) step is to take actions based on predictions. But where to begin? What's the best way to tackle potentially large swaths of potential churners? Many businesses make the mistake of taking those who scored the highest (i.e., are most likely to churn) and targeting them.

Decide which of the likely churners to target: Realistically, not every single customer who churns will come back. To save time and resources, effective teams go one step further and use uplift modeling and client clustering to drive return on investment (ROI) for churn marketing campaigns. In other words, you should only spend time and resources targeting those churners who will respond positively to your campaign.

Decide how to reach likely churners: Short term, marketing campaigns (particularly those offering special deals or discounts) are the most effective means of re-engaging predicted churners. You'll need to decide exactly how you will reach these customers - emailing? On-site promotions? Some other way?

ADVANCED/EXPERT: Uplift modeling looks to optimize the effects of marketing campaigns by predicting which customers are likely to take the desired action. Uplift models score customers into one of the following groups, of which only the Persuadables are targeted (along with a randomized control group) to maximize results and minimize wasted money and effort:

		IF TREATED	
		+	–
IF NOT TREATED	+	**Sure Things** Those that would have had a positive response either way and thus represent wasted marketing costs.	**Do Not Disturbs** Those who would have had a positive response but are then negatively impacted by marketing and thus should not be targeted.
	–	**Persuadables** The target group - those that would have had a negative response but are then positively persuaded by marketing.	**Lost Causes** Those that would have responded negatively with or without marketing and thus represent wasted marketing costs.

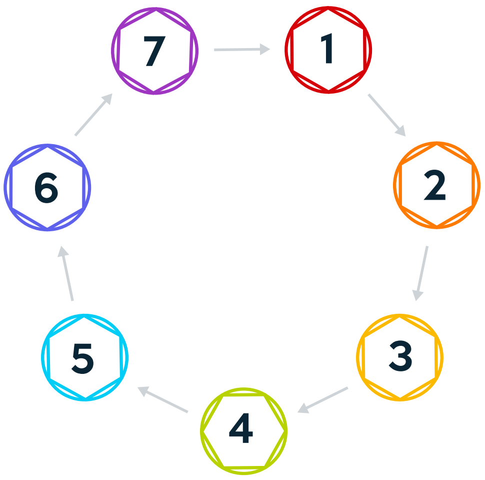
Read more in-depth[2] on uplift modeling, including more on machine learning for uplift and a closer look at different ways to evaluate uplift models.

INFOGRAPHIC

HOW TO: ADDRESS CHURN WITH PREDICTIVE ANALYTICS

Understand, Predict, and Minimize Customer Loss

It’s simple: **churn (or attrition) is when customers leave, and companies in nearly every industry have to address it because it has the power to plateau the growth of any businesses even if that business is gaining customers quickly.** The most successful companies address it by building predictive models that accurately predict churn; then they take action by building targeted marketing campaigns around preventing it or by making product changes that combat churn.



DEFINE

Define churn for your business:

Which customers should be included in churn modeling? At what point should they be considered churners?

Agree on what exactly churn predictions will be used for to avoid wasted effort.

IDENTIFY DATA

At minimum:

Customer ID + date/time of last interaction.

Include as many other relevant datasets as possible - in general, the more good data sources, the better.

CLEAN & ENRICH

Understand all variables.

Ensure clean, homogenous data.

PREDICT

Avoid the common churn modeling error of training your model on both past and future events.

Train only on data that will be available to you when predictive model is actually running.

Choose your evaluation method wisely; how you evaluate your model should correspond to your business need.

ITERATE

Determine the effectiveness of the model; is it sufficiently generic?

Ensure you've used training, validation, and testing sets that are not specific to a certain time or type of customer.

VISUALIZE

Communicate with product/marketing teams to build insightful visualizations.

Use visualizations to uncover additional insights to explore in the predictive phase.

DEPLOY

Ensure a continuous churn prevention strategy.

Avoid addressing churn as a one-time project.

TAKE ACTION

Use uplift modeling to reach churners most likely to react to engagement.

Build scalable marketing campaigns (short-term) and product improvements (long-term) to address churn.

data
iku

ABOUT DATAIKU

Dataiku is the advanced analytics leader and preferred software solution in helping organizations succeed in the world's rapidly evolving data-driven business ecosystem. Guided by the belief that true innovation comes from the effective combination of diversity of cultures, of mindsets, and of technologies, Dataiku's purpose is to enable all enterprises to imagine and deliver the data innovations of tomorrow.

ABOUT DATAIKU DSS (DATA SCIENCE STUDIO)

Dataiku DSS is a collaborative data science software platform that enables teams to explore, prototype, build, and deliver their own data products more efficiently. It is an open platform designed to accommodate rapidly evolving programming languages, big data storage and management technologies and machine learning techniques, and is conceived to accommodate the needs and preferences of both beginning analysts and expert data scientists. It also uniquely support:

Collaboration

Collaboration features make it easy to work as a team on ambitious data projects, to share knowledge amongst team members and to onboard new users much faster. You can add documentation, information or comments on all DSS objects.

Reproducibility

Every action in the system is versioned and logged through an integrated Git repository. Follow each action from the timeline in the interface, with easy rollback to previous versions.

Production Deployment

DSS lets you package a whole workflow as a single deployable and reproducible package. Automate your deployments as part of a larger production strategy. Run all your data scenarios using our REST API.

Governance and Security

DSS helps you create clearly defined projects and make sure your data is organized. And with fine grained access rights, your data is available only to the right persons.

Try Dataiku DSS for free by visiting www.dataiku.com/try