Machine Learning – Problem Framing Case Studies

# Machine Learning Problem Framing:

The objective of this workshop is to work through several conceptual Machine Learning problems, identify good ML problems, set up ML problems and frame the ML problems. For each of the case studies we should aim to identify the following characteristics/answer the following questions.

**Characteristics of Good ML Problems and Approach:**

1. The problem has a clear use case.
2. Know the problem before focusing on the data.
3. There is adequate data available for the problem
4. The features in the data have predictive power
5. The objective is to make decisions not predictions

**ML Problem Set Up**

1. Clearly and simply state what you would like machine learning model to do.
2. What is your ideal outcome?
3. Success and failure metrics.
4. What output would you like the ML model to produce?
5. Could you solve your problem without ML?

**ML Problem Framing**

1. Articulate your problem.
2. Start simple.
3. Identify Your Data Sources.
4. Design your data for the model.
5. Determine easily obtained inputs.
6. Ability to Learn.
7. Think About Potential Bias.

# Problem 1 – Sales and Attrition

Customer retention is one of the primary growth pillars for products with a subscription-based [business model](https://www.altexsoft.com/blog/business/software-business-models-examples-revenue-streams-and-characteristics-for-products-services-and-platforms/). Competition is tough in the SaaS (Software as a Service, think data centers, email, Amazon Web Services, Azure Cloud) market where customers are free to choose from plenty of providers even within one product category. Several bad experiences – or even one – and a customer may quit. And if droves of unsatisfied customers churn at a clip, both material losses and damage to reputation would be enormous.

**Customer churn (or customer attrition)**is a tendency of customers to abandon a brand and stop being a paying client of a particular business. The percentage of customers that discontinue using a company’s products or services during a particular time period is called a customer churn (attrition) rate. One of the ways to calculate a churn rate is to divide the number of customers lost during a given time interval by the number of acquired customers, and then multiply that number by 100 percent. For example, if you got 150 customers and lost three last month, then your monthly churn rate is 2 percent.

Churn rate is a health indicator for businesses whose customers are subscribers and paying for services on a recurring basis, notes head of data analytics department at ScienceSoft [Alex Bekker](https://twitter.com/alexlbekker), “Customers [of subscription-driven businesses] opt for a product or a service for a particular period, which can be rather short – say, a month. Thus, a customer stays open for more interesting or advantageous offers. Plus, each time their current commitment ends, customers have a chance to reconsider and choose not to continue with the company. Of course, some natural churn is inevitable, and the figure differs from industry to industry. But having a higher churn figure than that is a definite sign that a business is doing something wrong.”

There are many things brands may do wrong, from complicated onboarding when customers aren’t given easy-to-understand information about product usage and its capabilities to poor communication, e.g. the lack of feedback or delayed answers to queries. Another situation: Longtime clients may feel unappreciated because they don’t get as many bonuses as the new ones.

In general, it’s the overall customer experience that defines brand perception and influences how customers recognize value for money of products or services they use.

The reality is that even loyal customers won’t tolerate a brand if they’ve had one or several issues with it. For instance, 59 percent of US respondents to the [survey by PricewaterhouseCoopers](https://www.pwc.com/us/en/advisory-services/publications/consumer-intelligence-series/pwc-consumer-intelligence-series-customer-experience.pdf#page=8) (PwC) noted that they will say goodbye to a brand after several bad experiences, and 17 percent of them after just one bad experience.

You are in charge of retention for a large SaaS service and are tasked with increasing customer retention and decreasing churn. Your company has been gathering customer data for a while, and wishes to use it to identify behavior patterns of potential churners, segment these at-risk customers, and take appropriate actions to gain back their trust.

Taken from (<https://www.kdnuggets.com/2019/05/churn-prediction-machine-learning.html>)

## Data:

***Samples:***

100,000 Repeat Customer Monthly Profiles

10,000 Churned Customer Monthly Profiles

***Variables:***

* **customer demographic features** that contain basic information about a customer (e.g., age, education level, location, income)
* **user behavior features** describing how a person uses a service or product (e.g., lifecycle stage, number of times they log in into their accounts, active session length, time of the day when a product is used actively, features or modules used, actions, monetary value)
* **support features** that characterize interactions with customer support (e.g., queries sent, number of interactions, history of customer satisfaction scores)
* **contextual features**representing other contextual information about a customer.

**Characteristics of Good ML Problems and Approach:**

1. The problem has a clear use case:

The company wants to reduce the number of churners my identifying them early and stopping them before they do- improve retention rate/number of customers

Blanket approach- you could send all customers deals/discounts or lower price for all customers but that’s not a great business approach

1. Know the problem before focusing on the data:

Reasons for leaving

Is there a specific demographic (age, education etc.) that has less tolerance to poor customer service?

Does usage decrease/increase as they get close to the “churn”?

Is there a minimum level of customer satisfaction that people will withstand?

1. There is adequate data available for the problem:

Yes, it’s a good number of samples. Slightly unbalanced but probably just describes company

1. The features in the data have predictive power:

Yes, these variables describe the activity

1. The objective is to make decisions not predictions:

Improve services that made people leave

Give people deals that look like they’re about to leave

**ML Problem Set Up**

1. Clearly and simply state what you would like machine learning model to do:

Based off the demographics/usage/satisfaction, flag the customers whose behaviour indicates they may leave in the next x days

Identify important factors

1. What is your ideal outcome?

Decrease churn rate, identify churners

1. Success and failure metrics:

Difficult to measure success because if a potential “churner” does not leave, you can’t tell if they were going to in hindsight

Overall churn rate goes down, measure regularly (monthly?)

1. What output would you like the ML model to produce?

Binary- will or won’t leave in the next month. Give a probability they will leave

1. Could you solve your problem without ML?

Yes, ask each individual customer

**ML Problem Framing**

1. Articulate your problem:

Binary classification- will they leave yes or no?

1. Start simple:

Use a small subset of variables (e.g. just demographic) and see if there were patterns in that amongst churned customers.

1. Identify Your Data Sources:

Churned and repeat customer data

Good quality as collected by company. Regular metrics of website usage so little missing/inconsistency

1. Design your data for the model:

Entry per customer per month

1. Determine easily obtained inputs:
2. Ability to Learn:

External factors

1. Think About Potential Bias:

Outside factors, try to apply it to different product

# Problem 2 – Predicting Diabetes Using Machine Learning

Diabetes is a common chronic disease and poses a great threat to human health. The characteristic of diabetes is that the blood glucose is higher than the normal level, which is caused by defective insulin secretion or its impaired biological effects, or both ([Lonappan et al., 2007](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6232260/" \l "B29)). Diabetes can lead to chronic damage and dysfunction of various tissues, especially eyes, kidneys, heart, blood vessels and nerves ([Krasteva et al., 2011](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6232260/" \l "B22)). Diabetes can be divided into two categories, type 1 diabetes (T1D) and type 2 diabetes (T2D). Patients with type 1 diabetes are normally younger, mostly less than 30 years old. The typical clinical symptoms are increased thirst and frequent urination, high blood glucose levels ([Iancu et al., 2008](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6232260/#B12)). This type of diabetes cannot be cured effectively with oral medications alone and the patients are required insulin therapy. Type 2 diabetes occurs more commonly in middle-aged and elderly people, which is often associated with the occurrence of obesity, hypertension, dyslipidemia, arteriosclerosis, and other diseases ([Robertson et al., 2011](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6232260/#B40)).

With the development of living standards, diabetes is increasingly common in people’s daily life. Therefore, how to quickly and accurately diagnose and analyze diabetes is a topic worthy studying. In medicine, the diagnosis of diabetes is according to fasting blood glucose, glucose tolerance, and random blood glucose levels ([Iancu et al., 2008](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6232260/#B12); [Cox and Edelman, 2009](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6232260/#B6); [American Diabetes Association, 2012](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6232260/#B2)). The earlier diagnosis is obtained, the much easier we can control it. Machine learning can help people make a preliminary judgment about diabetes mellitus according to their daily physical examination data, and it can serve as a reference for doctors ([Lee and Kim, 2016](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6232260/#B23); [Alghamdi et al., 2017](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6232260/#B1); [Kavakiotis et al., 2017](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6232260/#B18)). For machine learning method, how to select the valid features and the correct classifier are the most important problems.

Taken from: Quan Zou, Kaiyang Qu, Yamei Luo, Dehui Yin, Ying Ju, and Hua Tang: Predicting Diabetes Mellitus With Machine Learning Techniques (Frontiers in Genetics, 2018, 9: 515)

## **Available Data:**

***Samples:***

151,598 Diabetic Samples

69,082 Healthy Samples

***Variables:***

* Age
* Pulse rate
* Breathe
* Left systolic pressure (LSP)
* Right systolic pressure (RSP)
* Left diastolic pressure (LDP)
* Right diastolic pressure (RDP)
* Height
* Weight
* Physique index
* Fasting glucose
* Low density lipoprotein (LDL)
* High density lipoprotein (HDL)

## Frame Problem:

**Characteristics of Good ML Problems and Approach:**

1. The problem has a clear use case:

Predict whether someone is at risk of diabetes and either correct lifestyle (type 2) or medicate (type 1) to try help prevent/manage

1. Know the problem before focusing on the data:

How do you normally diagnose?

Predict type of diabetes or none, predict time until symptoms, likelihood

1. There is adequate data available for the problem:

Where was data from? How did they pick the sample? Is a representative?

Unbalanced

1. The features in the data have predictive power:
2. The objective is to make decisions not predictions:

**ML Problem Set Up**

1. Clearly and simply state what you would like machine learning model to do:

Predict likelihood of diabetes in patients

1. What is your ideal outcome?

Give a probability of whether a patient has type 1 or 2 based on health data

1. Success and failure metrics:

Don’t figure out failures, if we say someone doesn’t have diabetes and they do then they disappear for months and come back when it’s worse

Want to reduce false negatives

1. What output would you like the ML model to produce?

Risk/probability of people with diabetes

1. Could you solve your problem without ML?

**ML Problem Framing**

1. Articulate your problem:

Binaray – yes or no

Multi class classification – type 1, type 2

Clustering – at risk or not

1. Start simple:

Glucose and pulse rate- put strong predictors in early

1. Identify Your Data Sources:

Health check up data

1. Design your data for the model:

Row per person per day, response = yes/no

1. Determine easily obtained inputs:

Glucose and pulse rate- put strong predictors in early

1. Ability to Learn:

External factors, not definitive yes or no

1. Think About Potential Bias:

Where did you get people from? Are the samples representative of the bigger population? E.g. location

# Problem 3 – Scouting Pitchers in baseball

You are a scout for a college baseball team. You have been tasked with identifying the next generation of pitchers whom the college wants to recruit. A key concern for the team is identifying players who have major league potential with the aim of improving the visibility of the college team and improving the profile of the program. You have gathered data on many youth players and players who have passed through the college system over the years. In addition you have been able to track several players from the youth system through to the major leagues.

In order to avoid manually categorizing future you players you wish to apply machine learning to the problem to improve the accuracy of your scouting and reduce the man hours necessary to spend individually reviewing players.

## Data:

**Samples:**

Youth samples: 10,000

Minor League Samples: 1,000

Major League Samples: 100

***Variables:***

* WAR (wins above replacement),
* *VORP* (Value over replacement pitcher)
* FIP (fielding independent pitching)
* WHIP (walks plus hits per inning pitched),
* wOBA (weighted on-base average)
* O.P.S. (on-base percentage plus slugging)
* Kinatrax Pitching metrics: Leading Leg Flexion, Throwing Forearm Pronation, and Internal/External Shoulder Rotation.
* Baseball pitching metrics (In index)

**Characteristics of Good ML Problems and Approach:**

1. The problem has a clear use case.
2. Know the problem before focusing on the data
3. There is adequate data available for the problem:
4. The features in the data have predictive power:
5. The objective is to make decisions not predictions:

.

**ML Problem Set Up**

1. Clearly and simply state what you would like machine learning model to do:
2. What is your ideal outcome?
3. Success and failure metrics:
4. What output would you like the ML model to produce?
5. Could you solve your problem without ML?

**ML Problem Framing**

1. Articulate your problem:

1. Start simple:
2. Identify Your Data Sources:
3. Design your data for the model:
4. Determine easily obtained inputs:
5. Ability to Learn:
6. Think About Potential Bias:

**Pitcher characteristics over time:**

|  |  |  |
| --- | --- | --- |
| **Years** | **Height** | **Weight** |
| 1960 – 1962 | 5’7” | 167 |
| 1971 - 1974 | 5’8” | 173.5 |
| 1976 - 1980 | 5’8” | 171.5 |
| 1988 – 1994 | 5’8” | 177 |
| 1999 – 2002 | 5’8” | 186 |
| 2013 | 6’1 | 207 |

# Problem 4 – Identify complex patients in a healthcare setting

By 2020, over 81 million persons in the United States will have 2 or more chronic conditions.1 Multimorbidity results in adverse health outcomes and higher healthcare costs, and challenges current models of care delivery.2,3 Care management has the potential to improve health outcomes for persons with multimorbidities. However, most disease and care management strategies have been developed to improve specific health outcomes for populations defined by single diseases or specific circumstances (such as hospital discharge).4-11 There is a need for strategies that can identify sub-populations with multiple, interacting diseases, in order to provide them with appropriate and relevant care management support.

Investigations to identify these populations of complex patients have traditionally relied upon multivariable regression analyses to identify patient-level characteristics (such as demographics and diseases) that predict the outcome of interest (such as hospitalization).12-14 As compared with investigations that use multivariable regression analyses to identify individual disease predictors of specific outcomes, data mining techniques provide an opportunity to empirically identify groups of patients with similar patterns of multimorbidities.

Taken from: https://www.ajmc.com/view/ajmc\_11julaug\_newcomer\_e324to32

## Data Available:

***Samples:***

15,480 adult samples who had two or more of 17 common chronic medical conditions and were categorized in the top 20% of total cost of care for 2 consecutive years.

***Variables:***

Variables indicating presence of medical conditions in patients:

* Diabetes
* Chronic obstructive pulmonary disease (COPD),
* Chronic kidney disease,
* Stroke,
* Obesity,
* Dementia,
* Fall,
* Hip fracture,
* Chronic pain,
* Skin ulcer,
* Orthopedic surgery,
* Back surgery,
* Abdominal surgery,
* Gastrointestinal bleeding,
* Cancer (excluding non-melanoma skin cancer),
* Cardiac disease (which included coronary artery disease and congestive heart failure)

**Characteristics of Good ML Problems and Approach:**

1. The problem has a clear use case:
2. Know the problem before focusing on the data:

1. There is adequate data available for the problem:
2. The features in the data have predictive power:
3. The objective is to make decisions not predictions:

**ML Problem Set Up**

1. Clearly and simply state what you would like machine learning model to do:
2. What is your ideal outcome?
3. Success and failure metrics:
4. What output would you like the ML model to produce?
5. Could you solve your problem without ML?

**ML Problem Framing**

1. Articulate your problem:
2. Start simple:
3. Identify Your Data Sources:
4. Design your data for the model:
5. Determine easily obtained inputs:
6. Ability to Learn:

1. Think About Potential Bias:

# Baseball Pitching Metrics:

* BABIP - [Batting average on balls in play](https://www.baseball-reference.com/bullpen/Batting_average_on_balls_in_play) - batting average against a pitcher on batted balls ending a plate appearance, excluding home runs
* BB - [Base on balls](https://www.baseball-reference.com/bullpen/Base_on_balls) (also called a "walk") - times pitching four balls, allowing the batter-runner to advance to first base
* BB/9 - [Base on balls](https://www.baseball-reference.com/bullpen/Walks) times nine divided by innings pitched ([Bases on balls per 9 innings pitched](https://www.baseball-reference.com/bullpen/Bases_on_balls_per_9_innings_pitched))
* BF - [Total batters faced](https://www.baseball-reference.com/bullpen/Total_batters_faced) - opponent's total plate appearances
* BK - [Balk](https://www.baseball-reference.com/bullpen/Balk) - number of times pitcher commits an illegal pitching action or other illegal action while in contact with the pitching rubber, thus allowing baserunners to advance
* BS - [Blown save](https://www.baseball-reference.com/bullpen/Blown_save) - number of times entering the game in a [save](https://www.baseball-reference.com/bullpen/Save) situation, and being charged the run which ties the game.
* CERA - [Component ERA](https://www.baseball-reference.com/bpv/index.php?title=Component_ERA&action=edit&redlink=1) - an estimate of a pitcher's ERA based upon the individual components of his statistical line (K, H, 2B, 3B, HR, BB, HBP)
* CG - [Complete game](https://www.baseball-reference.com/bullpen/Complete_game) - number of games where player was the only pitcher for his team
* DICE - [Defense-Independent Component ERA](https://www.baseball-reference.com/bpv/index.php?title=Defense-Independent_Component_ERA&action=edit&redlink=1) - an estimate of a pitcher's ERA based upon the defense-independent components of his statistical line (K, HR, BB, HBP)
* ER - [Earned run](https://www.baseball-reference.com/bullpen/Earned_run) - number of runs that did not occur as a result of errors or passed balls
* ERA - [Earned run average](https://www.baseball-reference.com/bullpen/Earned_run_average) - earned runs times innings in a game (usually nine) divided by innings pitched
* G - [Games pitched](https://www.baseball-reference.com/bullpen/Games_pitched) (aka '[Appearances](https://www.baseball-reference.com/bullpen/Appearances)') - number of times a pitcher pitches in a season
* GF - [Games finished](https://www.baseball-reference.com/bullpen/Games_finished) - number of games pitched where player was the final pitcher for his team
* G/F - [Ground ball fly ball ratio](https://www.baseball-reference.com/bpv/index.php?title=Ground_ball_fly_ball_ratio&action=edit&redlink=1) - ground balls allowed divided by fly balls allowed
* GS - [Starts](https://www.baseball-reference.com/bullpen/Starts) - number of games pitched where player was the first pitcher for his team
* H/9 - [Hits per nine innings](https://www.baseball-reference.com/bullpen/Hits_per_nine_innings) - hits allowed times nine divided by innings pitched (also known as [H/9IP](https://www.baseball-reference.com/bullpen/H/9IP) - [Hits allowed per 9 innings pitched](https://www.baseball-reference.com/bullpen/Hits_allowed_per_9_innings_pitched))
* H - [Hits Allowed](https://www.baseball-reference.com/bullpen/Hits_Allowed) - total hits allowed
* HB - [Hit batsman](https://www.baseball-reference.com/bullpen/Hit_by_pitch) - times hit a batter with pitch, allowing runner to advance to first base
* HLD (or H) - [Hold](https://www.baseball-reference.com/bullpen/Hold) - number of games entered in a save situation, left in save situation, recorded at least one out, and not having surrendered the lead
* HR - [Home runs allowed](https://www.baseball-reference.com/bullpen/Home_runs_allowed) - total home runs allowed
* IBB - [Intentional base on balls](https://www.baseball-reference.com/bullpen/Intentional_base_on_balls) allowed
* IR - [Inherited runners](https://www.baseball-reference.com/bullpen/Inherited_runners) - number of runners on base when the pitcher enters the game
* IRA - [Inherited runs allowed](https://www.baseball-reference.com/bullpen/Inherited_runs_allowed) - number of inherited runners allowed to score
* IP - [Innings pitched](https://www.baseball-reference.com/bullpen/Innings_pitched) - number of outs recorded while pitching divided by three
* IP/GS - Average number of innings pitched per game
* K - [Strikeout](https://www.baseball-reference.com/bullpen/Strikeout) - number of batters who received strike three
* K/9 - [Strikeouts per nine innings](https://www.baseball-reference.com/bullpen/Strikeouts_per_nine_innings) - strikeouts times nine divided by innings pitched ([Strikeouts per 9 innings pitched](https://www.baseball-reference.com/bullpen/Strikeouts_per_9_innings_pitched))
* K/BB - [Strikeout-to-walk ratio](https://www.baseball-reference.com/bullpen/Strikeout-to-walk_ratio) - number of strikeouts divided by number of base on balls
* L - [Loss](https://www.baseball-reference.com/bullpen/Loss) - number of games where pitcher was pitching while the opposing team took the lead, never lost the lead, and went on to win
* OBA - [Opponents batting average](https://www.baseball-reference.com/bpv/index.php?title=Opponents_batting_average&action=edit&redlink=1) - hits allowed divided by at-bats faced
* PIT - Pitches thrown ([Pitch count](https://www.baseball-reference.com/bullpen/Pitch_count))
* RA - [Run average](https://www.baseball-reference.com/bpv/index.php?title=Run_average&action=edit&redlink=1) - number of runs allowed times nine divided by innings pitched
* RAA - [Runs Against Average](https://www.baseball-reference.com/bpv/index.php?title=Runs_Against_Average&action=edit&redlink=1) - a sabermetric statistic to predict win-percentage.
* SO - [Shutout](https://www.baseball-reference.com/bullpen/Shutout) - number of complete games pitched with no runs allowed
* SV - [Save](https://www.baseball-reference.com/bullpen/Save) - number of games where the pitcher enters a game led by the pitcher's team, finishes the game without surrendering the lead, is not the winning pitcher, and either (a) the lead was three runs or less when the pitcher entered the game; (b) the potential tying run was on base, at bat, or on deck; or (c) the pitcher pitched three or more innings
* W - [Win](https://www.baseball-reference.com/bullpen/Win) - number of games where pitcher was pitching while his team took the lead and went on to win (also related: **winning percentage**)
* WP - [Wild pitches](https://www.baseball-reference.com/bullpen/Wild_pitch) - charged when a pitch is too high, low, or wide of home plate for the catcher to field, thereby allowing one or more runners to advance or score