Submitted by Pratik Bhavsar

Problem 1

Presto SQL query for retention table

with transactions\_info\_per\_user as (

select user\_id, min(transaction\_date) as first\_transaction,

date\_trunc('month', min(transaction\_date)) as activation\_cohort

from transactions

group by user\_id

),

users\_per\_activation\_cohort as (

select activation\_cohort, count(\*) as no\_of\_users

from transactions\_info\_per\_user

group by activation\_cohort

),

months\_after\_activationivation\_per\_purchase as (

select distinct trans.user\_id, tiu.activation\_cohort, date\_diff('month', tiu.first\_transaction, trans.transaction\_date) AS months\_after\_activation

from transactions as trans

left join transactions\_info\_per\_user as tiu

on trans.user\_id = tiu.user\_id

),

activation\_cohort\_user\_purchases as (

select activation\_cohort, months\_after\_activation, count(\*) as user\_purchases\_per\_month\_per\_cohort\_after\_activation

from months\_after\_activationivation\_per\_purchase

group by activation\_cohort, months\_after\_activation

),

retention\_data\_before\_pivot as (

select activation\_cohort\_user\_purchases.activation\_cohort, months\_after\_activation,

users\_per\_activation\_cohort.no\_of\_users,

cast((user\_purchases\_per\_month\_per\_cohort\_after\_activation\*100.0/users\_per\_activation\_cohort.no\_of\_users) as integer) as pcent

from activation\_cohort\_user\_purchases

join users\_per\_activation\_cohort

on users\_per\_activation\_cohort.activation\_cohort = activation\_cohort\_user\_purchases.activation\_cohort

)

SELECT activation\_cohort,

no\_of\_users,

month\_pct\_map[1] as month\_1,

month\_pct\_map[2] as month\_2,

month\_pct\_map[3] as month\_3,

month\_pct\_map[4] as month\_4,

month\_pct\_map[5] as month\_5,

month\_pct\_map[6] as month\_6,

month\_pct\_map[7] as month\_7,

month\_pct\_map[8] as month\_8,

month\_pct\_map[9] as month\_9,

month\_pct\_map[10] as month\_10,

month\_pct\_map[11] as month\_11,

month\_pct\_map[12] as month\_12

FROM (

SELECT activation\_cohort, no\_of\_users, map\_agg(months\_after\_activation, pcent) month\_pct\_map

FROM retention\_data\_before\_pivot

GROUP by activation\_cohort, no\_of\_users

ORDER by activation\_cohort ASC

) t

Problem 2

Modeling

[Google Colab Notebook Link](https://colab.research.google.com/drive/1XoZ8FSAvXKY_vD2vwQv-0z4gmyQ4uJ0i?usp=sharing)

**a) Perform exploratory analysis and extract insights from the dataset.**

These variables were dropped

**total day charge** is highly correlated with total day minutes (ρ = 1)

**total eve charge** is highly correlated with total eve minutes (ρ = 1)

**total night charge** is highly correlated with total night minutes (ρ = 1)

**total intl charge** is highly correlated with total intl minutes

Other insights

**total day minutes** - churned customers are talking more in day time

**customer service calls** - churned customers are making more customer support calls

**total eve minutes** - there is a slightly higher evening usage in churned customers

**international plan** - high churn is seen in those who have taken the plan

**voice mail plan** - less churn is seen in those who have taken the plan

**state,account length, area code** - no difference in distribution of users

**b) Split the dataset into train/test sets and explain your reasoning.**

No churn - 2850

Churn - 483

Imbalance - 5.9:1

Because the data is imbalanced, the train test split is done in a stratified way to maintain same distribution in train and test as shown below.

Test size is kept 20 to have enough test samples.

train\_X, test\_X, train\_Y, test\_Y = train\_test\_split(df[cols], df[target\_col], test\_size=0.20, stratify=df['churn'], random\_state = random\_state)

**c) Build a predictive model to predict which customers are going to churn**

**and discuss the reason why you choose a particular algorithm.**

After trying out different models manually and with H2O, this is the recommendation. I haven't done extensive hypertuning and Cross Validation on manual models but it can be added for real scenario.

H2O does hypertuning and CV and so the results are robust.

# Final model suggestion

**Option 1**

H2O's best model GBM

GBM\_grid\_\_1\_AutoML\_20200719\_075452\_model\_48

**AUC: 0.9143**

**Predict\_time\_per\_row\_ms: 0.036522 seconds**

**Option 2**

H2O's xgboost

3 times less latency compared to option 1 but slightly less AUC

XGBoost\_grid\_\_1\_AutoML\_20200719\_075452\_model\_11

**AUC: 0.9127**

**Predict\_time\_per\_row\_ms: 0.016924 seconds**

Note: For latency, see the last column in H2O leaderboard table in notebook

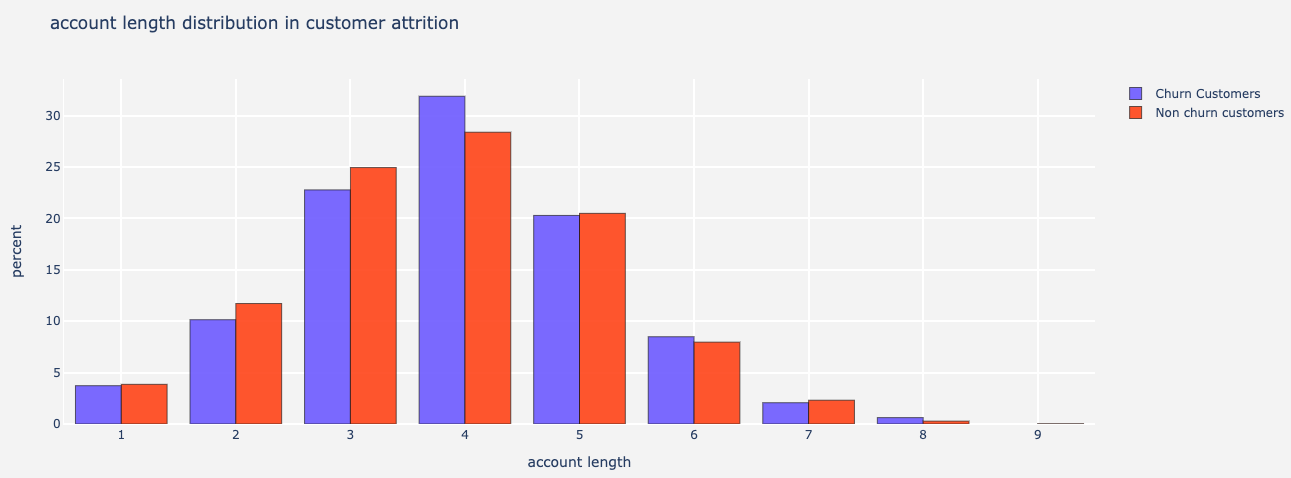
We can try more hypertuning and ensembling of models to achieve higher metric.

**d) Establish metrics to evaluate model performance.**

Since its an imbalanced dataset, we select AUC as the metric

**e) Discuss the potential issues with deploying the model into production.**

* We need to select an appropriate probability threshold so that we don’t end up targeting less confident predicted users and lose money in the process.
* As the data will keep changing we will need periodic retraining of models to capture new trends due to data drift.
* The model might predict churn but we will find it difficult to understand the reason of churning and may not be able to take the best action for retention
  + We can use libraries like SHAP by which we can get the interpretability of model inference and know the main features for prediction
* Model prediction throughput/latency can be a problem which can be solved by scalability. We can also take a simpler model but it can hurt the performance.
* The account length distribution is biased. Hence the model can be biased for very small and long account length and give us wrong predictions after deployment.



Account length distribution by months active

Problem 3

Experiment Design

**a) Establish the primary objective of the experiment and create metrics for**

**performance measurement.**

Ways to measure performance

* **Increase in user activity (day + evn + night calls + intl calls + vmail + etc)**
* More renewal in contracts
* Reduction in churn probability from the model for the user
* Spending increase

Sources of data

* Usage tracking systems
* Billing systems
* Customer care systems

**b) Create null hypothesis and alternative hypothesis and discuss**

**corresponding statistics.**

**Null hypothesis** - Giving offers will reduce churn

**Alternate hypothesis** - Giving offers will not reduce churn

**Statistics**

Current churn rate is 17% i.e retention of 83%.

Lets assume we want to increase retention by +5 to 88%.

**Dataset**

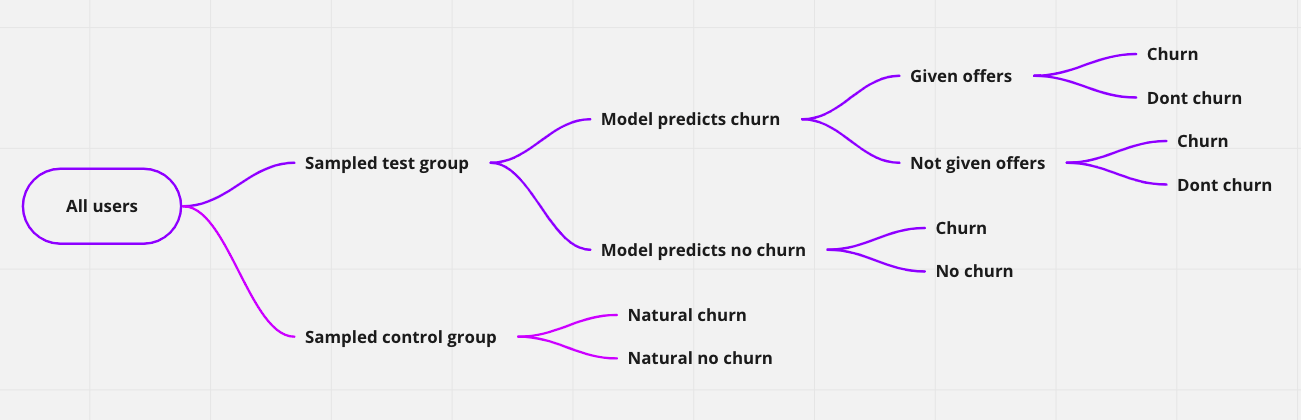
No churn - 2850

Churn - 483

Imbalance - 5.9:1

**c) Discuss how you setup the control and treatment group and overall**

**experiment workflow.**

****

Experiment design

To test the hypothesis we setup a sampled test group and a sampled control group.

By giving the offer, we are no longer sure if the user was going to churn or not. So we need an isolation mechanism.

Offer performance

We first predict the churn for the test group. To compare the model efficiency, we break the **predicted churnable users** into 2 cohorts - ***given offers*** and ***not given offers***. By comparing the real churn behaviour of this 2 we will know the **offer performance**.

Model performance

By comparing the test and control, we will know the **model performance**.

Business flow

For contacting the potential churn user, we break it down with different parameters

How to contact

* Email
* SMS
* Message in bills
* Personal call

When to contact

* 1 month before expiry of contract
* 2 month before expiry of contract
* 3 month before expiry of contract

Incentives to offer

* $50
* $100
* $150

How long should the contract renewal be

* 3 month
* 6 month
* 1 year
* 2 years

Total different way to contact

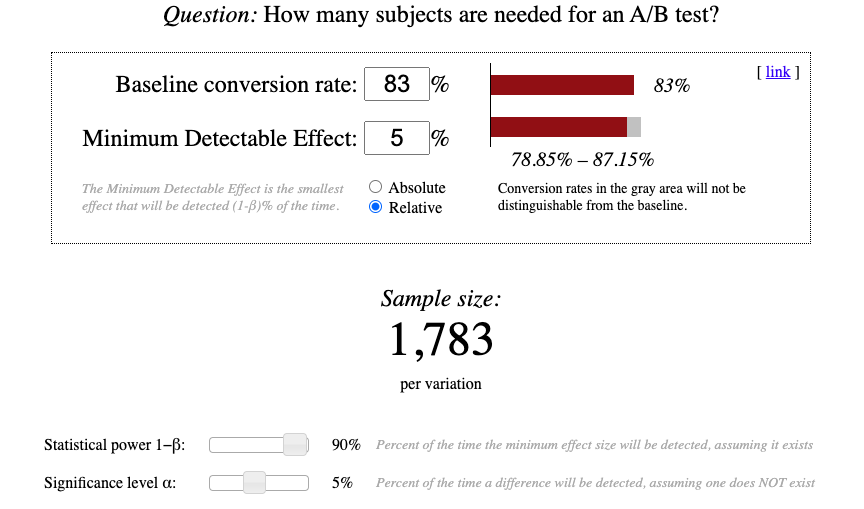
= 4 how \* 3 when \* 3 offers \* 4 contract renewal

= 144

Sample size estimation

Sample size required to increase retention of 83 by 5%

**Sample size per group**



<https://www.evanmiller.org/ab-testing/sample-size.html>

As per the above image, every group will contain 1783 randomly selected people showing a tendency to churn. They will be targeted with the selected group method.

Total test users

= 144\*1783

= 253186

Different contact actions will require different infrastructure and investment

* Email - IT infra
* SMS - IT infra
* Message in bills - IT infra
* Personal call - Customer support + IT infra

**d) Explain the risks of the experiment and how to mitigate the risks**

Risk 1

We have already mitigated the risk of getting incorrect value by breaking predicted churn cohort into given offers and not given offers as explained earlier.

Risk 2

If we give the same offers to all users hoping Customer lifetime value(CLV) > offer, we might end up losing more.

To circumvent this we can have another methodology to estimate CLV of users, segment them and then target them with small and big offers.

Risk 2

Contacting users with offers can make them irritated or aware of other providers. This can stimulate shopping behaviour which can lead to even higher churn.

We can reduce this by

* Contacting users way in advance of expiry of contract -> 6 months
  + They wont leave as they are already left with 6 more months in current contract.
  + If they are not happy about the offer, they might again become complacent by the time contract is over
* We can give better offers to make it easy for them to extend contract
  + High discounts to customers with high potential CLV