<https://www.kaggle.com/competitions/telecom-churn-case-study-hackathon-c58/overview>

**Overview**

**Description**

**Problem Statement**

In the telecom industry, customers are able to choose from multiple service providers and actively switch from one operator to another. In this highly competitive market, the telecommunications industry experiences an average of 15-25% annual churn rate. Given the fact that it costs 5-10 times more to acquire a new customer than to retain an existing one, customer retention has now become even more important than customer acquisition.

For many incumbent operators, retaining high profitable customers is the number one business goal. To reduce customer churn, telecom companies need to **predict which customers are at high risk of churn**. In this project, you will analyze customer-level data of a leading telecom firm, build predictive models to identify customers at high risk of churn.

In this competition, your goal is to build a machine learning model that is able to predict churning customers based on the features provided for their usage.

**Evaluation**

**Goal**

It is your job to predict if a customer will churn, given the ~170 columns containing customer behavior, usage patterns, payment patterns, and other features that might be relevant. Your target variable is "churn\_probability"

**Metric**

Submissions are evaluated on [Classification Accuracy](https://scikit-learn.org/stable/modules/model_evaluation.html#accuracy-score) between the value of the predicted value and the actual value of churn for each of the customers.



The public leaderboard is going to rank your submission against other users while the competition is active, however, once the competition is ended, the final ranks will be calculated on the private leaderboard.

**Submission file format**

The file should contain a header and have the following format (CSV):

id,churn\_probability

70005,0.0

70006,1.0

70007,0.0

etc.

A sample file can be found attached in the Data section of this competition.

**FAQs**

**How do I contact support?**

For this competition, any questions, concerns, or technical difficulties will be addressed at the Discussion forum on UpGrad Learn platform. Any discussion posted regarding support on this competition page will have to be redirected to the same portal.

**What’s the difference between a private and public leaderboard?**

The Kaggle leaderboard has a public and private component to prevent participants from “overfitting” to the leaderboard. If your model is “overfit” to a dataset then it is not generalizable outside of the dataset you trained it on. This means that your model would have low accuracy on another sample of data taken from a similar dataset.

**Public Leaderboard:** For all participants, the same 70% of predictions from the test set are assigned to the public leaderboard. The score you see on the public leaderboard reflects your model’s accuracy on this portion of the test set.

**Private Leaderboard:** The other 30% of predictions from the test set are assigned to the private leaderboard. The private leaderboard is not visible to participants until the competition has concluded. At the end of a competition, we will reveal the private leaderboard so you can see your score on the other 30% of the test data. The scores on the private leaderboard are used to determine the competition winners.

**What are kernels?**

Kaggle Kernels is a cloud computational environment that enables reproducible and collaborative analysis. Kernels support scripts in R and Python, Jupyter Notebooks, and RMarkdown reports. Go to the Kernels tab to view all of the publicly shared code on this competition.

**Data**

**Dataset Description**

**File descriptions:**

* **train.csv**: Contains 172 columns. The primary key that represents each customer is id. The target variable that you need to predict is churn\_probability which contains a value of 0 or 1. This data is what you are going to use for EDA, cleaning, feature engineering, model building, model evaluation, model selection, and finally model training.
* **test.csv**: Contains 171 columns, doesn't contain the target variable churn\_probability. You will apply all the necessary preprocessing steps to get this data into the right format and then use the model trained using the *train.csv* file to make predictions with this. This is unseen data! Only the competition hosts know the actual values of the target feature for this data and therefore your submissions will be evaluated on how well your model does with this dataset.
* **sample.csv**: This contains the format in which you need to submit the solutions to Kaggle. The id column in this dataset exactly the same as the id column in *test.csv*. You will make your predictions on the *test.csv* data and store them in a submission file that has the same format as this file. Check the **Overview>Evaluation** tab for mode details.
* **data\_dictionary.csv**: This contains the definitions for the various acronyms that you will need to understand each variable. For example, the variable total\_og\_mou\_7, contains the acronyms total, og, mou, and 7, which can be translated as the total outgoing minutes of voice calls made by the user in month of July.

**Data Definitions**

The definitions are also listed down below:

* CIRCLE\_ID : Telecom circle area to which the customer belongs to
* LOC : Local calls - within same telecom circle
* STD : STD calls - outside the calling circle
* IC : Incoming calls
* OG : Outgoing calls
* T2T : Operator T to T, i.e. within same operator (mobile to mobile)
* T2M : Operator T to other operator mobile
* T2O : Operator T to other operator fixed line
* T2F : Operator T to fixed lines of T
* T2C : Operator T to it’s own call center
* ARPU : Average revenue per user
* MOU : Minutes of usage - voice calls
* AON : Age on network - number of days the customer is using the operator T network
* ONNET : All kind of calls within the same operator network
* OFFNET : All kind of calls outside the operator T network
* ROAM : Indicates that customer is in roaming zone during the call
* SPL : Special calls
* ISD : ISD calls
* RECH : Recharge
* NUM : Number
* AMT : Amount in local currency
* MAX : Maximum
* DATA : Mobile internet
* 3G : 3G network
* AV : Average
* VOL : Mobile internet usage volume (in MB)
* 2G : 2G network
* PCK : Prepaid service schemes called - PACKS
* NIGHT : Scheme to use during specific night hours only
* MONTHLY : Service schemes with validity equivalent to a month
* SACHET : Service schemes with validity smaller than a month
* \*.6 : KPI for the month of June
* \*.7 : KPI for the month of July
* \*.8 : KPI for the month of August
* FB\_USER : Service scheme to avail services of Facebook and similar social networking sites
* VBC : Volume based cost - when no specific scheme is not purchased and paid as per usage

Competition Overview in Upgrad

This segment will contain details about the competition, the overall problem statement and the suggested steps to solve the given problem statement.

Competition Details

* [Competition Link](https://www.kaggle.com/competitions/telecom-churn-case-study-hackathon-c58/overview)(Note - This link is only for your batch to participate in the competition and you're not authorised to share it with anybody else).
* **Please note that you need to submit only from one account on Kaggle and the team name should be: Name\_of\_member1\_Name\_of\_member2**
* **Deadlines**
  + There are two submissions you have to do. One submission is through the platform and another submission is through the Kaggle platform. Deadlines for both are the same.
* Check the different tabs for details about the competition and details on the problem statement. Make sure you go through them in detail to understand the different aspects of this competition. Here's a brief about each of them:
  + **Overview** - Contains the description of the problem statement, the evaluation metric used in the competition and some FAQs about the leaderboard. In this competition, your goal is to build a machine learning model that is able to predict churning customers for a telecom operator based on the features provided for their usage.
  + **Data** - Contains the datasets, data dictionary and the sample submission file to be used in this competition.  The data dictionary contains the meanings of some of the commonly used abbreviations in the dataset. Some frequent ones are loc (local), IC (incoming), OG (outgoing), T2T (telecom operator to telecom operator), T2O (telecom operator to another operator), RECH (recharge), etc. The attributes containing 6, 7, and 8 as suffixes imply that those correspond to the months 6, 7, and 8, respectively.
  + **Code**- This tab contains the Starter Notebook that contains sample steps that you can follow in this case study. Use this notebook file as a reference to build your final solution.
  + **Discussion**- This tab is for initiating and participating in discussions. For this competition, we won't be using the discussion tab on Kaggle. Rather you should be posting your queries on the upGrad's discussion forum only.
  + **Leaderboard** - Once you start making submissions, you'll be ranked based on the performance of the evaluation metric. You can keep submitting better predictions(max 10 submissions per day) until the end of the competition. Note that the ranking that's visible to you is on the Public Leaderboard which is final.
  + **Rules**- Please go through this tab to understand the rules for this competition.
  + **Teams** - This tab is for you to create a team name that'll appear on the leaderboard. ***Note:****that this is a group competition, only one person in the team should register(according to the team name mentioned earlier).*
  + **My Submissions** - This will show you the submissions that you have uploaded till then.
  + **Submit Predictions** - This will allow you to start submitting new predictions.

Objectives

The main goal of the case study is to build ML models to predict churn. The predictive model that you’re going to build will the following purposes:

1. It will be used to predict whether a high-value customer will churn or not, in near future (i.e. churn phase). By knowing this, the company can take action steps such as providing special plans, discounts on recharge etc.
2. It will be used to identify important variables that are strong predictors of churn. These variables may also indicate why customers choose to switch to other networks.
3. Even though overall accuracy will be your primary evaluation metric, you should also mention other metrics like precision, recall, etc. for the different models that can be used for evaluation purposes based on different business objectives. For example, in this problem statement, one business goal can be to build an ML model that identifies customers who'll definitely churn with more accuracy as compared to the ones who'll not churn. Make sure you mention which metric can be used in such scenarios.
4. Recommend strategies to manage customer churn based on your observations.

Note that it's highly likely that you'll need to build multiple models to fulfil the objectives mentioned in Points 1 and 2.  Since here, you have a large number of attributes, and thus you should try using a dimensionality reduction technique such as PCA and then build a predictive model. After PCA, you can use any classification model.

The above model will only be able to achieve one of the two goals - to predict customers who will churn. You can’t use the above model to identify the important features for churn. That’s because PCA usually creates components that are not easy to interpret.

Therefore, build another model with the main objective of identifying important predictor attributes which help the business understand indicators of churn. A good choice to identify important variables is a **logistic regression** model or a model from the **tree family**.

Suggested Steps

In the competition link, check the Code tab for the Starter Notebook that you can use as a reference for this entire case study. Some of the steps that you can use are as follows:

* **Data Understanding, Preparation, and Pre-Processing :**
  + Data understanding, identification of potentially useful and non-useful attributes and variable importance and impact estimation
  + Data preparation, performing data cleaning, missing values imputation, outlier removal, and column level standardization (for e.g., date, etc.) into one format
* **Exploratory Data Analysis :**
  + Performing basic preliminary data analysis including finding the correlation between variables and scatter plots to identify relationships between variables
  + Performing advanced data analysis, including plotting relevant heatmaps, histograms, and basic clustering to find patterns in the data
* **Feature Engineering and Variable Transformation :**
  + Feature engineering and performing one or more methods on attributes that can lead to the creation of a new potentially useful variable; for e.g., day from the date
  + Variable transformation and applying categorical variable transformations to turn into numerical data and numerical variable transformations to scale data
* **Model Selection, Model Building, and  Prediction :**
  + Identifying the type of problem and making a list of decisive models from all available choices
  + Choosing a training mechanism; for e.g., cross-validation, etc., and tuning hyperparameters of each model
  + Testing each model on the respective model evaluation metric
  + Choosing the best model based on the fit of the data set and output variable
  + Using ensemble options to improve the efficacy based on the evaluation metric stated in the problem

**Submission and Evaluation Guidelines**

This assignment will have two separate submissions as mentioned below. Please go through the following guidelines in detail to understand the requirements of both

Internal Submission

* **Weightage** - 80%
* For the internal submission, you need to complete the steps mentioned in the Objectives section in the previous segment. You need to submit the following files
  + **Solution file** - This Jupyter notebook will contain the overall solution that you have created for the given problem statement. It will contain all the steps like data pre-processing, EDA, model building and evaluation and final recommendations as detailed out in the previous segment.
  + **Submission.csv file** - This file will contain the churn predictions for observations in the test.csv  file made by the final model that you've chosen in the Solution File. The format for this submission file will be the same as the one which you'll be uploading on the Kaggle platform.
* You're allowed to submit this only once.

Kaggle Submission

* **Weightage** - 20%
* For Kaggle Submission, you will make your predictions on the *test.csv* data and store them in a CSV file that has the same format as the sample submission file present on the Kaggle platform.
* You can be creative and use additional techniques like Class Imbalance handling, Boosting, etc. on top of the model that you built in your internal submission to improve your accuracy.
* You can make up to 10 submissions per day till the end of the competition deadline.

**[Important]**

**Initial Submission**- It is highly recommended to make the initial submission within 4 days of the competition launch. This is to ensure your team name is registered and remains unique for the rest of the competition duration. The steps for making the Kaggle submissions (including the Initial submission are mentioned in this document.

Evaluation and Penalties  
This is a group case study.  For submissions obtained within one week after the deadline, there will be a 30% penalty. Submissions beyond one week after the deadline will not be accepted.

**Evaluation Rubrics**

There will be two parts to the solution. For each of them the evaluation rubrics are as follows:

Kaggle Submission

**Weightage - 20%**

Based on your Kaggle Submission, you'll be given an evaluation score and a final rank on the Public leaderboard. This will be used to compute your Kaggle Submission Score, which will be calculated out of 20% of your overall case study marks.

**Note: Make sure your accuracy is greater than the sample submission that is present in the leaderboard**

Internal Submission

**Weightage - 80%**

Your internal submission which will carry 80% is going to be evaluated based on the following rubrics.

|  |  |  |
| --- | --- | --- |
| **Stage** | **Meets expectations** | **Does not meet expectations** |
| Data understanding, preparation & pre-processing (10 %) | Has used imputation techniques to treat missing values    Has identified potentially useful and non-useful attributes using feature importance and impact estimation. | Has not used any imputation techniques to treat missing values  Has not used identified potentially useful attributes using feature importances |
| EDA and Feature Engineering (30 %) | Has Performed correlation analysis between variables and has used plots to identify relationships between variables    Has Performed advanced data analysis, including plotting heatmaps and histograms    Has Performed feature engineering i.e. one or more methods on attributes that lead to the creation of a new potentially useful variable e.g. day from the date    Has applied variable transformation which are methods to turn categorical variable into numerical data and scale transformations to numerical data | Has not performed correlation analysis    Has not performed advanced data analysis Has not performed feature engineering    Has not applied variable transformation |
| Model selection, model building, evaluation & prediction (35 %) | Has tried more than one model for training and evaluation purposes on respective metric    Has used cross-validation and hyperparameter tuning for each model        Choosing the best model based on the fit of the dataset and output variable. The relevant evaluation metrics have been calculated and interpreted correctly as per the business context | Has not tried more than one model for the case study      Has not used cross-validation and hyperparameter tuning for each model      An incorrect model has been chosen at the end with wrong and incomplete inferences about the evaluation metrics. |
| Code readability and conciseness (5%) | The code is well commented and the text is written in detail to explain the thought process.    Efficient, concise code is written. | The code is not commented well / text is not written in detail.    Inefficient/verbose code is written. |