**Mindset**

1. Evaluation will be kept lenient, so make sure you attempt this case study.
2. It is understandable that you might struggle with getting started on this. Just brainstorm, discuss with peers, or get help from TAs.
3. There is no right or wrong answer. We have to become comfortable with dealing with uncertainty in business. This is exactly the skill we want to develop.

**Problem Statement**

Recruiting and retaining drivers is seen by industry watchers as a tough battle for Ola. Churn among drivers is high and it’s very easy for drivers to stop working for the service on the fly or jump to Uber depending on the rates.

As the companies get bigger, the high churn could become a bigger problem. To find new drivers, Ola is casting a wide net, including people who don’t have cars for jobs. But this acquisition is really costly. Losing drivers frequently impacts the morale of the organization and acquiring new drivers is more expensive than retaining existing ones.

You are working as a data scientist with the Analytics Department of Ola, focused on driver team attrition. You are provided with the monthly information for a segment of drivers for 2019 and 2020 and tasked to predict whether a driver will be leaving the company or not based on their attributes like

* Demographics (city, age, gender etc.)
* Tenure information (joining date, Last Date)
* Historical data regarding the performance of the driver (Quarterly rating, Monthly business acquired, grade, Income)

**Dataset:**

Dataset Link: [ola\_driver.csv](https://d2beiqkhq929f0.cloudfront.net/public_assets/assets/000/002/492/original/ola_driver_scaler.csv)

**Column Profiling:**

1. MMMM-YY : Reporting Date (Monthly)
2. Driver\_ID : Unique id for drivers
3. Age : Age of the driver
4. Gender : Gender of the driver – Male : 0, Female: 1
5. City : City Code of the driver
6. Education\_Level : Education level – 0 for 10+ ,1 for 12+ ,2 for graduate
7. Income : Monthly average Income of the driver
8. Date Of Joining : Joining date for the driver
9. LastWorkingDate : Last date of working for the driver
10. Joining Designation : Designation of the driver at the time of joining
11. Grade : Grade of the driver at the time of reporting
12. Total Business Value : The total business value acquired by the driver in a month (negative business indicates cancellation/refund or car EMI adjustments)
13. Quarterly Rating : Quarterly rating of the driver: 1,2,3,4,5 (higher is better)

**Concepts Tested:**

* Ensemble Learning- Bagging
* Ensemble Learning- Boosting
* KNN Imputation of Missing Values
* Working with an imbalanced dataset

**What "good" looks like:**

* Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset.
* Convert date-like features to their respective data type
* Check for missing values and Prepare data for KNN Imputation
  + You may consider only numerical features for this purpose
* Aggregate data in order to remove multiple occurrences of same driver data (We did something similar in Delhivery business Case)
  + You can start from storing unique Driver IDs in an empty dataframe and then bring all the features at same level (Groupby Driver ID)
* Feature Engineering Steps:
  + Create a column which tells whether the quarterly rating has increased for that driver - for those whose quarterly rating has increased we assign the value 1
  + Target variable creation: Create a column called target which tells whether the driver has left the company- driver whose last working day is present will have the value 1
  + Create a column which tells whether the monthly income has increased for that driver - for those whose monthly income has increased we assign the value 1
* Statistical summary of the derived dataset
* Check correlation among independent variables and how they interact with each other
* One hot encoding of the categorical variable
* Class Imbalance Treatment
* Standardization of training data
* Using Ensemble learning - Bagging, Boosting methods with some hyper-parameter tuning
* Results Evaluation:
  + Classification Report
  + ROC AUC curve
* Provide actionable Insights & Recommendations

**Evaluation Criteria (100 Points):**

1. Define Problem Statement and perform Exploratory Data Analysis **(10 points)**  
   * Definition of problem (as per given problem statement with additional views)
   * Observations on shape of data, data types of all the attributes, conversion of categorical attributes to 'category' (If required), missing value detection, statistical summary.
   * Univariate Analysis (distribution plots of all the continuous variable(s) barplots/countplots of all the categorical variables)
   * Bivariate Analysis (Relationships between important variables)
   * Illustrate the insights based on EDA
     + Comments on range of attributes, outliers of various attributes
     + Comments on the distribution of the variables and relationship between them
     + Comments for each univariate and bivariate plots
2. Data Preprocessing **(50 Points)**
   * KNN Imputation
   * Feature Engineering
   * Class Imbalance treatment
   * Standardization
   * Encoding
3. Model building **(20 Points)**
   * 1 Ensemble - Bagging Algorithm
   * 1 Ensemble - Boosting Algorithm
4. Results Evaluation **(10 Points)**
   * ROC AUC Curve & comments
   * Classification Report (Confusion Matrix etc)
5. Actionable Insights & Recommendations **(10 Points)**

**Submission Process:**

* Type your insights and recommendations in the text editor.
* Convert your jupyter notebook into PDF (Save as PDF using Chrome browser’s Print command), upload it in your Google Drive (set the permission to allow public access), and paste that link in the text editor.
* Optionally, you may add images/graphs in the text editor by taking screenshots or saving matplotlib graphs using plt.savefig(...).
* After submitting, you will not be allowed to edit your submission.