Hello,

The goal of this exercise is to help us understand your approach towards and experience with building ML solutions. There are several problems in the FinTech domain and we've chosen churn prediction as a case-study for you. You will find all the necessary information below. In case of any doubts, please write back to the concerned HR/POC immediately.

Problem statement:

Customer churn (or customer attrition) is a tendency of customers to abandon a brand and stop being a paying client of a particular business. Churn analytics provides valuable capabilities to predict customer churn and also define the underlying reasons that drive it.

Your focus in this exercise should be on the following:

1. Data - Implement a framework that enables us to extract meaningful information from the data for our current use-case, and can be extended to other scenarios within the domain.
2. Model - Showcase a systematic approach towards model selection and validation keeping in mind the logistics associated with model maintenance and deployment.
3. System design - Layout a reasonable architecture that accounts for the ingestion, processing and storage of the data, followed by a standard pipeline that isolates and employs a Model Development Life Cycle, and finally explains an inference system (in production) that also connects with monitoring and data/model management services.

Your pipeline should be well thought, scalable and reflect your knowledge of real production setup. Take this opportunity to show your skills.

We don't expect you to have the time to code everything that you could think of, but your solution should be sufficiently complete and documented for us to evaluate you on coding standard, ML, documentation, system design, etc.

Timeline:

We expect you to do your best and submit a solution within a week. Please write to us in case you need more time.

Deliverables:

You are asked to send the following deliverables in a zip file:

1. A report (PDF) detailing:
    a. Description of the pipeline and design choices
    b. Performance evaluation of the model
    c. Discussion of how to scale up the pipeline to process tens of billions data points

    d. Discussion of future work
2. The source code used to create the pipeline

Tasks/Activities list:

Your code should contain the following activities/Analysis:

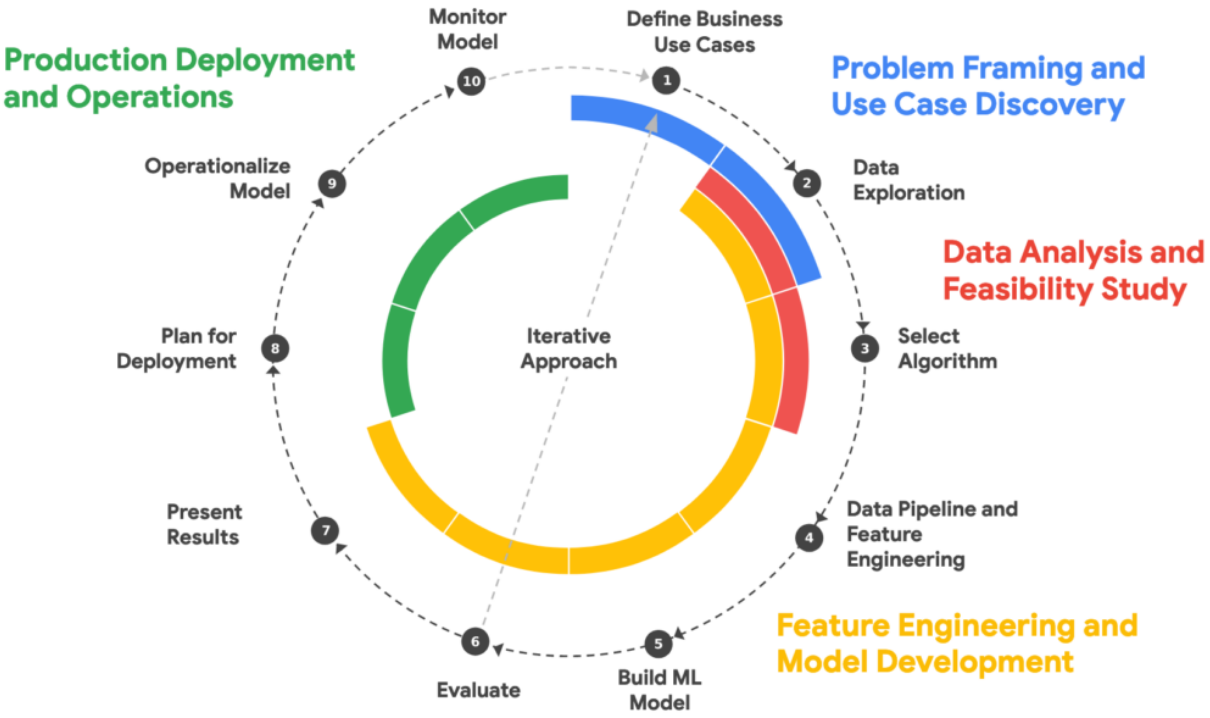
1. Collect the data from https://www.kaggle.com/blastchar/telco-customer-churn. This data does not include any time-related field so you must add a new feature of time interval (say timestamps dated June-Aug 2021 at a reasonable frequency)
2. State a hypothesis using the features included in the data and connect them to their potential influence on customer churn
3. Standard Exploratory Data Analysis (EDA) - Show the Covariance and Correlation Matrix, Data quality check, treat the missing values, outliers etc.
4. Feature Engineering and feature reduction (use a max of 6 raw/derived features)
5. Train/Test Split - Apply a sampling distribution to find best split
6. Decide on the Model Evaluation Metrics and explain how they can be connected to business KPIs
7. Model Selection, Training, Prediction and Assessment
8. Hyperparameter Tuning/Model Improvement
9. Model Validation Statistics
10. Model deployment plan and architecture design (a working solution, if possible)

Success Metrics:

Below are the metrics for a successful submission of this case study.

1. Classification labels/Targets Variables:
   Churn — Whether the customer churned or not (Yes or No)
   Tenure — Number of months the customer has stayed with the company
   Reasons - Reasons behind the customer churn (Multiple Reasons possible).
   NA if Churn = No
2. Hypothesis Building should be concise
3. Time based features are a must. Please add time-based feature during training and testing the model
4. Deploy max 6 most important features after applying most suitable feature reduction techniques
5. Please use AutoML if possible otherwise define/create minimum two ML models for comparison
6. Accuracy of the model on the test data set should be > 70%
7. Add at least one method for Hyperparameter tuning.
8. Add at least one method for model validation other than Train/Test Split

9. Model deployment plan should highlight the basic components for model serving/roll-out, monitoring, logging and iterating/updation. Please use the below image as a reference



Bonus points:

1. You can package your solution in a zip file included with a README that explains the installation and execution of the end-to-end pipeline.
2. Highlight how will you utilize CI/CD solutions and implement CI in this exercise using Github Actions
3. You can document/showcase an airflow/kubernetes based deployment
4. For post model-serving stages, you can showcase strategic roll-out and A/B testing plans for the current use-case. (Refer Seldon)
5. You can showcase your documentation skills explaining how it helps our organization.