Bank Marketing Workflow

- The project revolves around leveraging machine learning techniques to optimize marketing strategies for a banking institution, drawing insights from the publicly available dataset from the UCI Machine Learning Repository.
- The dataset, sourced from a Portuguese banking institution, contains information regarding direct marketing campaigns, including attributes such as client demographics, economic indicators, and campaign outcomes.
- By analyzing this dataset, the aim is to develop predictive models that can assist in targeting potential customers by identifying and excluding non-potential customer from the campaign

Following is one of the industry approaches with which we can implement the model in production:

Azure Databricks Workflow

Data Ingestion and Cleaning

- For the 101_data_cleaning script:
- Read the CSV files from an Azure Data Lake Storage (ADLS) container mapped as a mount in Azure Databricks.
- The output of 101_data_cleaning is properly formatted and saved for input into the subsequent feature engineering step.
- Reference- Documentation

Job Cluster Setup

- Configure a job cluster with appropriate settings for running the notebook. Available options for job clusters include:
    - Cluster Type: Standard, High Concurrency, or GPU.
    - Node Types: Choose from Standard_DS, Memory_Optimized, Compute_Optimized, etc., depending on workload requirements.
    - Autoscaling: Enable autoscaling for dynamic resource allocation based on workload demands.
    - Spark Version: Select the appropriate version of Apache Spark compatible with your notebook and dependencies.
    - Python Version: Choose the required Python version for executing the notebook.
    - Driver & Worker Node Configuration: Adjust the number and size of driver and worker nodes based on workload characteristics.
- Libraries/Dependencies: Install necessary libraries and dependencies for the notebook execution. Specify these requirements in the job cluster setup,
- Reference- Documentation

Git Integration

- Tag the workflows to the Git repository HTTPS link for accessing the setup.py, provided for installing wheel containing the required dependent libnraries into the job compute
- Reference- Documentation

Feature Engineering

- The next step in the workflow is 201_feature_engineering script to perform feature engineering tasks.
- This script will receive the cleaned data from 101_data_cleaning as input.
- Convert columns and categories within the columns to the required data types suitable for downstream analysis and model training.
- Sample the data from this step for monitoring to ensure data quality for the model.
- Follow the same steps as done for setup of 101_data_cleaning.

Model Deployment and Evaluation

- Import the CatBoost model from the pickle file saved in 302_model_training.
- Apply the model to the data saved from previous steps for predictions.
- Save the data to the downstream process and sample the data for performance metric checks.

Data Sampling and Performance Metrics

- Sample the data using 302_model_interpreation and do performance evaluation.
- Run various performance metrics tests on the sampled data to assess the model's performance accurately as mentioned in the notebook
- Metrics include accuracy, precision, recall, F1 score, ROC curves, etc.
- Present the metrics to MLOps engineer whenever the scheduled workflow runs.