High Level Design

Bank Marketing Analytics

Written By	**Aditya Gautam**
Document Version	**1.0**
Last Revised Date	**12-April-2023**

Document Version Control

Date Issued	Version	Description	Author
12-April-2023	1.0	First Version of Complete HLD	Aditya Gautam

Contents

1 Introduction

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

(1) Present all of the design aspects and define them in detail
(2) Describe the user interface being implemented
(3) Describe the hardware and software interfaces
(4) Describe the performance requirements
(5) Include design features and the architecture of the project
(6) List and describe the non-functional attributes like:
    (a) Security
    (b) Reliability
    (c) Maintainability
    (d) Portability
    (e) Reusability
    (f) Application compatibility
    (g) Resource utilization
    (h) Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

2 General Description

2.1 Product Perspective & Problem Statement

The data is related to direct marketing campaigns (phone calls) of a Portuguese banking institution. The classification goal is to predict if the client will subscribe to a term deposit.
The data is related to direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required, in order to access if the product (bank term deposit) would be subscribed or not.
This dataset contains 4 files.:
1) bank-additional-full.csv with all examples (41188) and 20 inputs, ordered by date (from May 2008 to November 2010)
2) bank-additional.csv with 10% of the examples (4119), randomly selected from 1), and 20 inputs.
3) bank-full.csv with all examples and 17 inputs, ordered by date (older version of this dataset with fewer inputs).
4) bank.csv with 10% of the examples and 17 inputs, randomly selected from 3 (older version of this dataset with fewer inputs).
The smallest datasets are provided to test more computationally demanding machine learning algorithms
Find key metrics and factors and show the meaningful relationships between attributes.

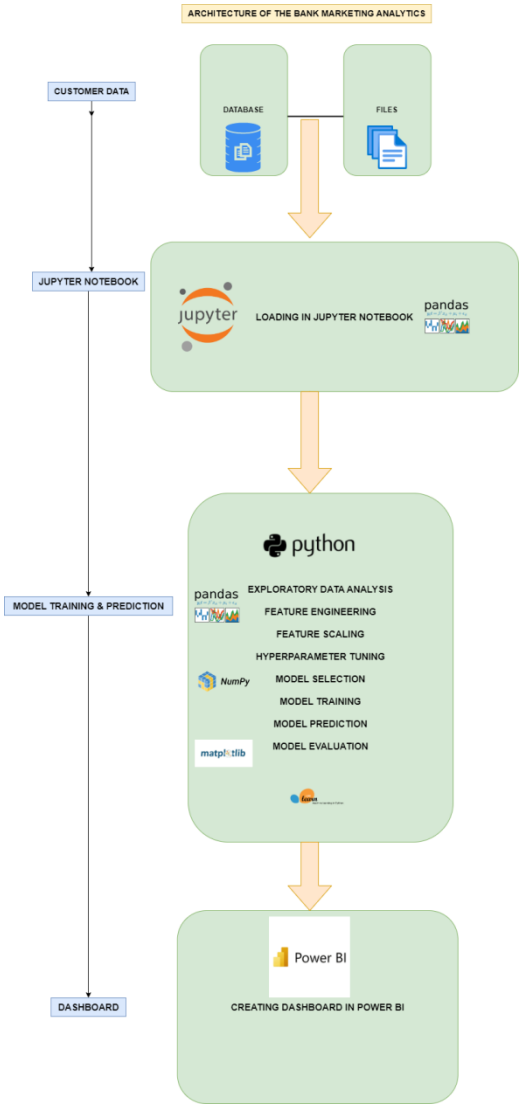
2.2 Tools Used

Jupyter Notebook, Python Libraries such as Pandas, Numpy, Matplotlib, seaborn, Sklearn and

Business Intelligence such as Power Bi.

3 Design Details

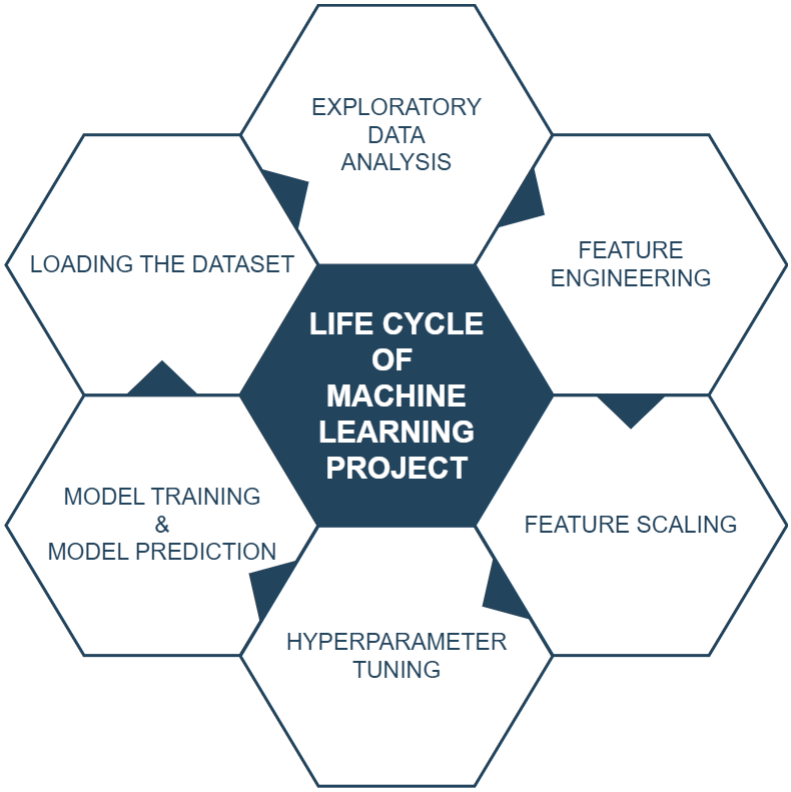
3.1 Architecture of the Bank Marketing Analytics



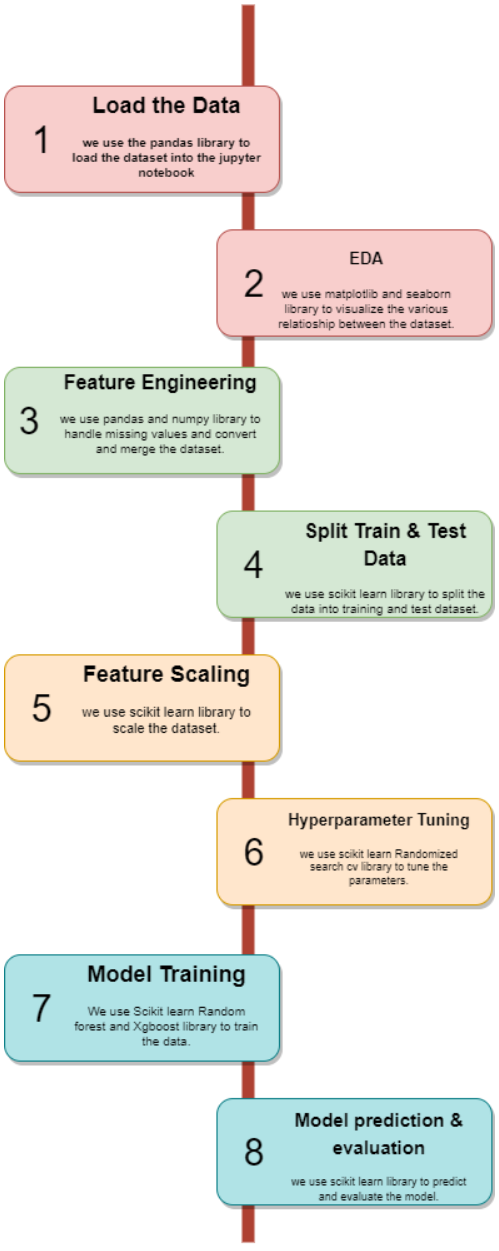
The detailed architecture of the Bank Marketing Analytics has been discussed in the above architecture diagram which gives a overview of the step by step process of the project which gives an idea about flow of the data from original sources to database, then exporting the data from database to importing the data into jupyter notebook by using

pandas library for data cleaning process, then for visualize the data, visualization library such Matplotlib and seaborn is used for the purpose and pandas library is used for Feature engineering. Then scikit learn library is used for feature selection , model training, hyperparameter tuning and model evaluation of the data. And finally, deploying the trained data into Power Bi for creating an interactive dashboard.

3.2 **Life Cycle of Machine Learning Project**



3.3 **Detailed Architecture of Bank Marketing Analytics**



3.4 **Function Design**

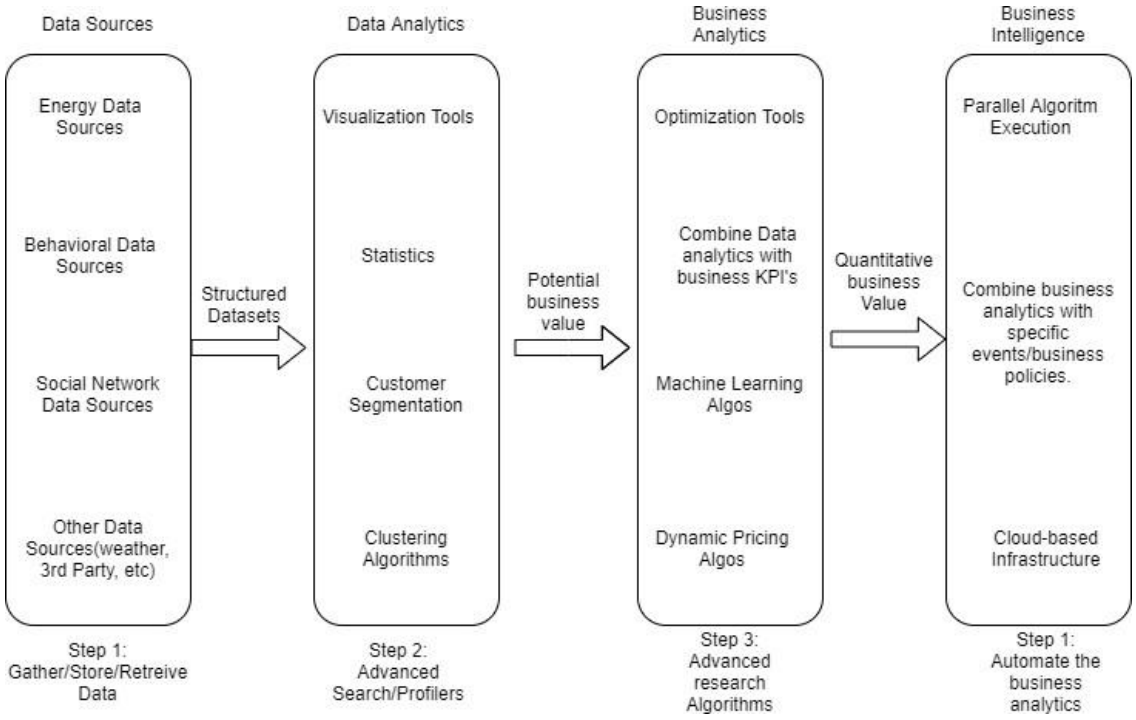
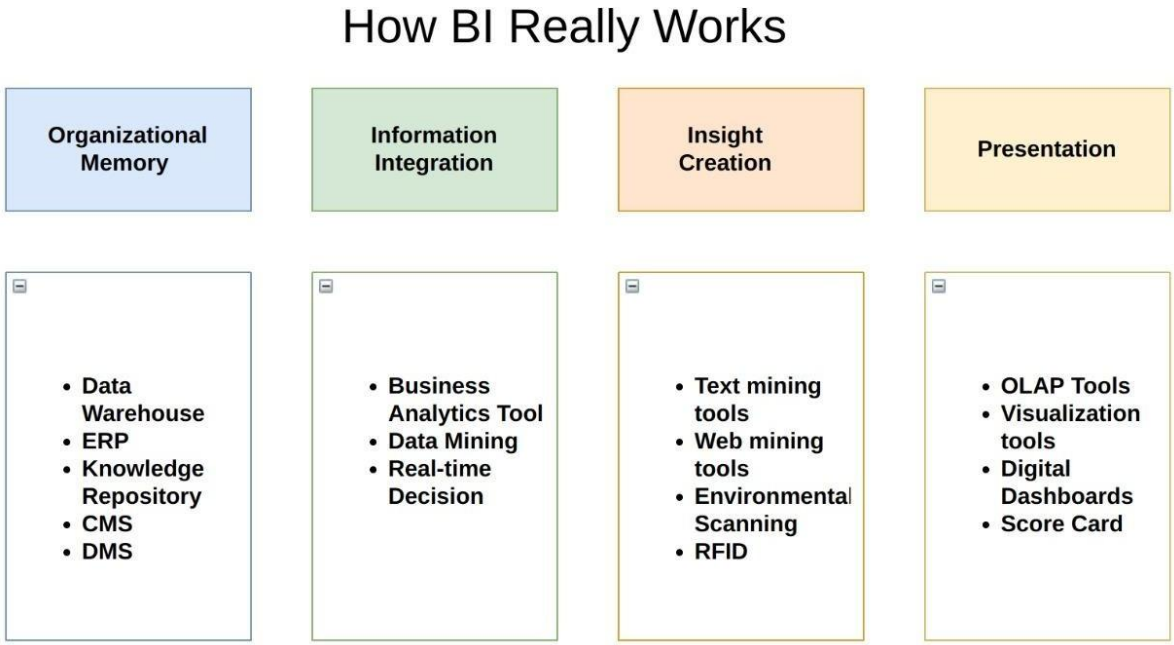


Figure 1: Functional Architecture of Business Intelligence



3.5 **Optimization**

(1) Handling the missing values

For optimizing the model, missing values can be replace with zero or with the average value.

(2) Handling the Categorical values

For optimizing the model, categorical values can be replace with dummy variables or can be mapped with the required numerical values.

(3) Handling the Multi-co-linearity between the variables

For optimizing the model, handling the multi-co-linearity between variables is a vital step Which is can be check by variance inflating factor and if VIF is above 4 or tolerance is below 0.25 indicates that multi-co-linearity might exist, and further investigation is required. When VIF is higher than 10 or tolerance is lower than 0.1 there is significant multi-co-linearity that needs to be corrected. Which can be handled by dropping some redundant variables.

(4) Selecting the Important feature

For optimizing the model, from Sklearn library import ExtraTreesClassifier , which helps to select the vital features for the model.

(5) Hyper parameter Tuning

For optimizing the model, from sklearn library import RandomizedSearchCV, which helps to fine tune the parameter of the model before training, which helps to attain an optimized model.

(6) Select the optimized model for training the dataset.

To attain the optimized model, model have to be evaluated using R2 score and RMSE, if the model is performing low in the R2 score and RMSE then change the model and select another model and follow this step until desired level of accuracy is attained.

Repeat the Following steps until desired level of accuracy is obtained.

4 **Key Performance Indicator**

Key Performance Indicator provides vital insights about the predicted model

4.1 **Insight of Bank Marketing Analytics**

(1) In Deposit Vs Age Plot, Age 32 is the Highest Depositor with 221 deposit, where Random Forest Model predicted as Age 31 is the Highest Depositor with 33 deposit and XG Boost Model predicted as Age 31 is the Highest Depositor with 39 deposit.

(2) In Deposit Vs Job Plot, Blue Collar job is the Highest Depositor with 708 deposit, where Random Forest Model predicted as Blue Collar job is the Highest Depositor with 192 deposit and XG Boost Model predicted as Blue Collar job is the Highest Depositor with 270 deposit.

(3) In Balance Vs Job Vs Deposit Plot, Management job has the Maximum Balance with 102127 Balance and Highest Depositor with 9458 deposit, where Random Forest Model predicted as Management job has the Maximum Balance with 102127 Balance and Highest Depositor with 1363 deposit and XG Boost Model predicted as Management job has the Maximum Balance with 102127 Balance and Highest Depositor with 1363 deposit.

(4) In Deposit Vs Month Plot, May Month has the Highest Depositor with 925 deposit, where Random Forest Model predicted as May Month has the Highest Depositor with 508 deposit and May Month has the Highest Depositor with 699 deposit.

(5) In Previous Marketing Campaign Plot, Unknown Category is the Highest Depositor with 3386 deposit, where Random Forest Model predicted as Unknown Category is the Highest Depositor with 586 deposit and Unknown Category is the Highest Depositor with 804 deposit.

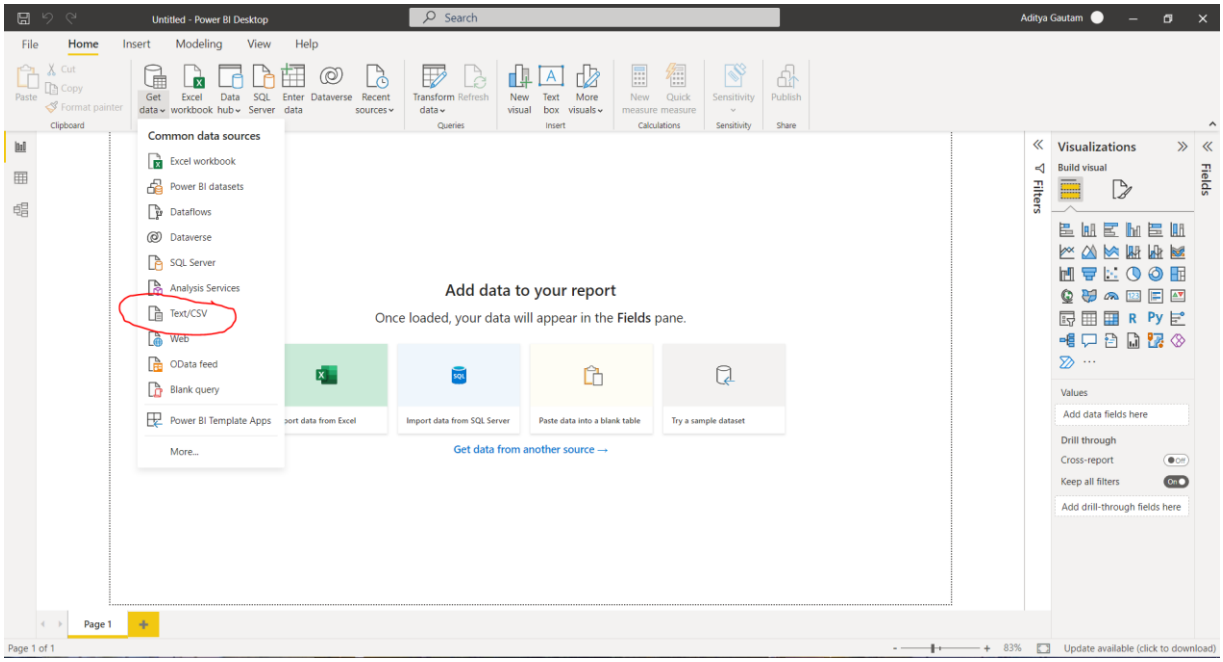
(6) In Deposit Vs Campaign Plot, Day 1 has the Highest Depositor with 2561 deposit, where Random Forest Model predicted as Day 1 has the Highest Depositor with 263 deposit and XG Boost Model predicted as Day 1 has the Highest Depositor with 332 deposit.

(7) After Analyzing the Whole Dataset, It is Predicted that Bank will lose 70%-80% of depositor in the future. So if the bank takes appropriate measures, it can save up to 70%-80% depositor in the future.

5. **Deployment**

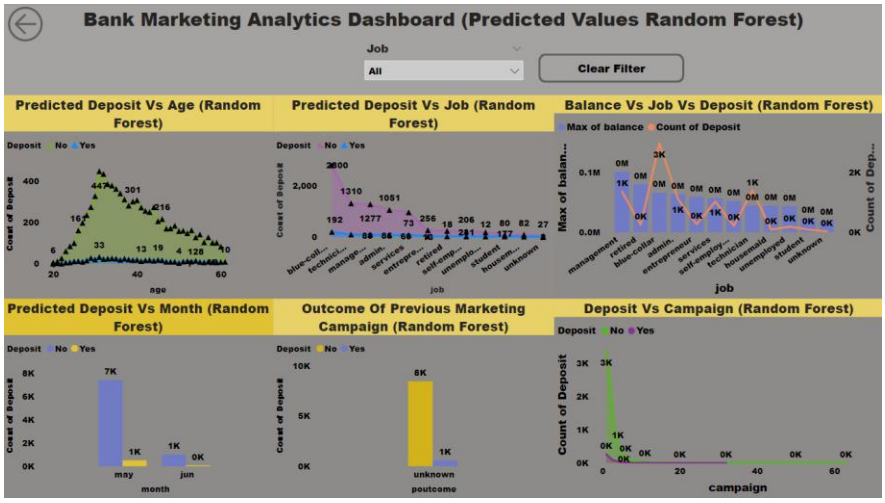
5.1 **Load the Dataset in Power BI**

Now, Load the Dataset into Power Bi for creating the dashboard.

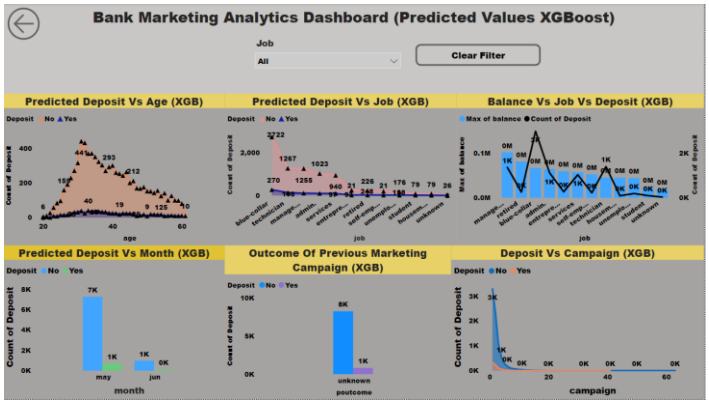


5.2 **Create the interactive dashboard**

Random Forest Classifier

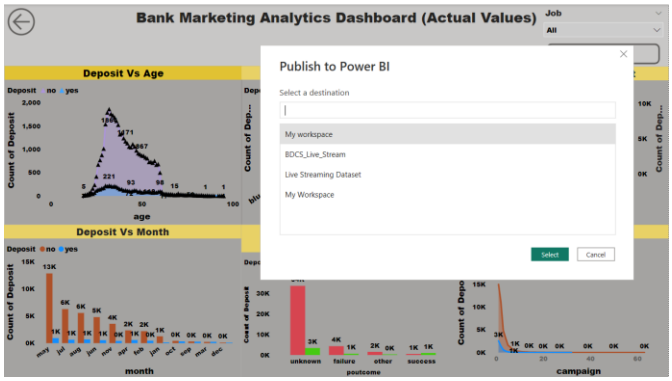


XGBoost Classifier

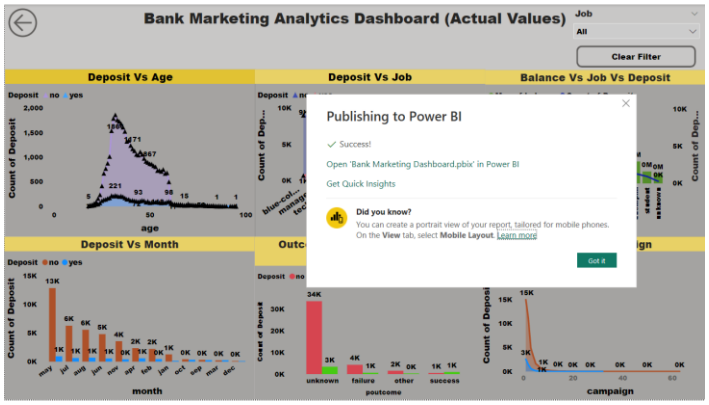


5.3 Publish to Power Bi account

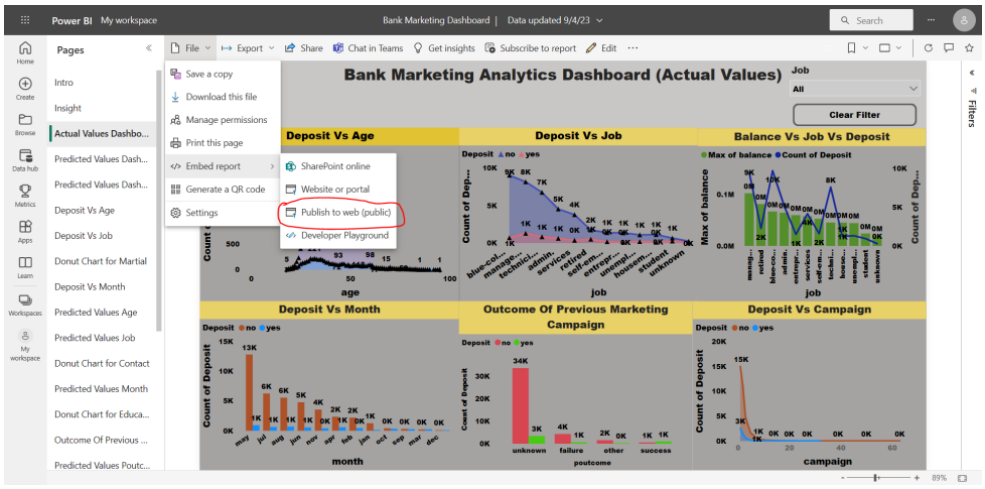
(1)



(2)



5.4 **Publish to Web**



5.5 **Share the Public Link to Client**