MACHINE LEARNING MODEL DEPLOYMENT WITH IBM CLOUD WATSON STUDIO

**PHASE-3 Development Part-1**

**Submitted By:**

**Yemireddy Ramya**

**NM ID:au723921104056**

**EMAIL:ramyavemireddy732@gmail.com**

**project : MACHINE LEARNING MODEL DEPLOYMENT WITH IBM CLOUD WATSON STUDIO**

**Topic :**

**Phase-3 Devlopment part-1**

**Phase-3 Development Part-1**

**INTRODUCTION :**

Create an IBM Cloud Account: If you don't have an IBM Cloud account, you'll need to sign up for one. Access Watson Studio: Log in to your IBM Cloud account and navigate to Watson Studio. Create a New Project: Inside Watson Studio, create a new project. This is where you'll organize your work.

Add a Data Asset:

Upload or connect to the dataset you want to use for training your machine learning model. Open IBM Watson Studio's Modeler: Watson Studio typically provides a drag-and-drop interface for building models. Look for a tool or service within Watson Studio that allows you to create a model visually.Define Target Variable and Features:Specify the target variable (what you want to predict) and the features (input variables) in your dataset.

Select Algorithm: Choose a machine learning algorithm that suits your problem. Watson Studio often provides a variety of algorithms to choose from.Train the Model: Use your dataset to train the machine learning model. This involves feeding the algorithm with labeled

data and letting it learn the patterns.

Evaluate the Model: Assess the performance of your model using evaluation metrics. Watson Studio typically provides tools for this purpose.Deploy the Model: If you're satisfied with the model's performance, you can deploy it to make predictions on new data.

Monitor and Update: Continuously monitor your model's performance. If necessary, update it with new data to ensure it stays accurate over time.

Predictive Use Case:

Customer Churn Prediction

Definition:

Customer churn prediction is a predictive use case where machine learning models are applied to analyze customer data and predict the likelihood of a customer discontinuing their relationship with a product or service.

Components:

Data Collection: Gather historical data on customer interactions, transactions, and behavior. Feature Selection: Identify relevant features or variables that could influence customer churn, such as customer satisfaction, usage patterns, or support interactions.

Labeling : Label historical data to indicate whether customers churned or not. This labeled data is used for training the predictive model.Model Training: Use machine learning algorithms to train a model on historical data. The model learns patterns and relationships between features and the likelihood of churn.

Prediction: Apply the trained model to new, unseen data to predict whether current customers are at risk of churning. Actionable Insights:Based on predictions, businesses can take proactive measures to retain at-risk customers. This might involve targeted marketing, personalized incentives, or improved customer support.

Benefits:

- Early identification of customers at risk, allowing for proactive retention efforts.

- Improved customer satisfaction by addressing issues before they lead to churn.

- Optimize resource allocation by focusing retention efforts on high-risk customers.

Explanation with Example :

A telecommunications company might use customer churn prediction to identify subscribers likely to switch to a competitor. By offering tailored promotions or addressing concerns, the company can reduce churn and enhance customer loyalty.

Predictive Use Case:

A predictive use case involves developing a model to forecast or estimate future outcomes based on historical data patterns. This could be applied to various fields such as finance, healthcare, marketing, or manufacturing. For example, predicting stock prices, diagnosing diseases, forecasting sales, or anticipating equipment failures are all predictive use cases.

Selecting a Relevant Dataset:

The choice of dataset depends on the specific predictive use case you're interested in. Here are a few examples:

1. Financial Predictions:

- Dataset: Historical stock prices, trading volumes, economic indicators.

2. Healthcare Predictions:

- Dataset: Patient records, medical history, diagnostic tests results.

3. Sales Forecasting:

- Dataset: Sales data over time, marketing expenses, customer demographics.

4. Equipment Failure Prediction:

- Dataset: Maintenance logs, sensor data from equipment, historical failure records.

5. Weather Predictions:

- Dataset: Historical weather data, satellite imagery, atmospheric conditions.

Ensure that the dataset is relevant to your use case, has enough data for meaningful predictions, and is representative of the scenarios you want the model to handle. Additionally, consider the quality of the data and whether any preprocessing is required to make it suitable for predictive modeling.

1. Import the Dataset:

- In your Watson Studio project, navigate to the assets section.

- Upload or import your dataset into the project.

2. Explore the Dataset:

- Use Watson Studio tools to explore and understand the characteristics of your dataset. This includes checking for missing values, data types, and statistical summaries.

3. Preprocess the Data:

- Handle missing values: Impute or remove missing values based on the nature of your data.

- Encode categorical variables: Convert categorical variables into numerical format using techniques like one-hot encoding.

- Normalize or standardize numerical features if needed.

4. Select Features:

- Use Watson Studio's tools to select relevant features for your model. This might involve statistical analysis or feature importance techniques.

5. Split the Dataset:

- Divide your dataset into training and testing sets. This helps you assess the model's performance on unseen data.

6. Choose and Train the Model:

- Select a machine learning algorithm available in Watson Studio.

- Train the model using the training dataset.

Evaluate Model Performance:

Use Watson Studio tools to evaluate the model's performance on the testing dataset. Common metrics include accuracy, precision, recall, and F1 score.

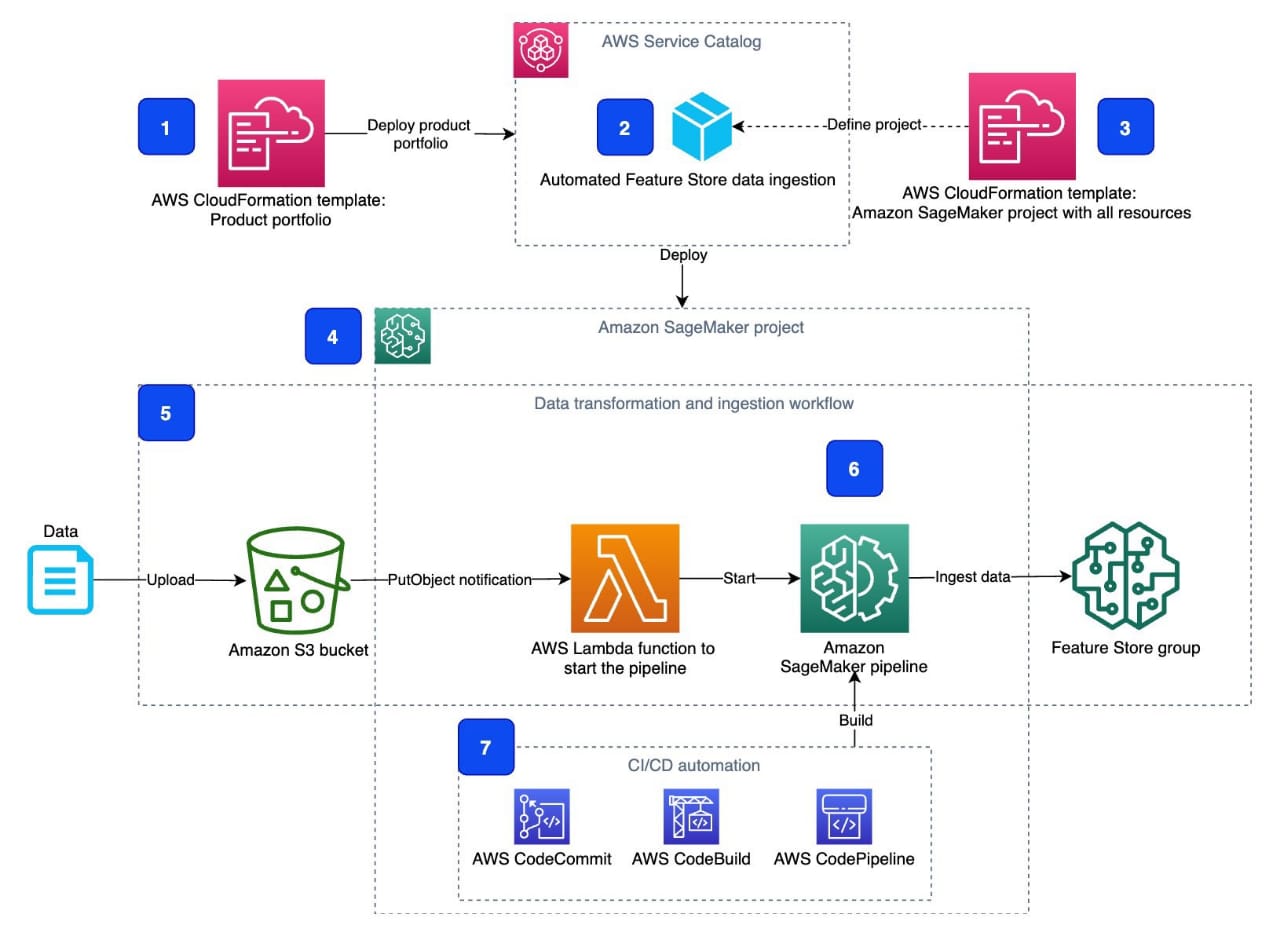
Iterate and Tune:

If the model performance is not satisfactory, iterate. You might need to tune hyperparameters or try a different algorithm.

Deploy the Model:

Once satisfied with the model, deploy it using Watson Studio to make predictions on new data.

Architecture Of Machine Learning Model:



Code Generation :

import necessary libraries

import pandas as pd

from sklearn.model\_selection import Train\_Test\_Split

from sklearn.ensemble import RandomForestClassifier

from sklearn.meterics import accuracy\_score, classification\_report

data = pd.read\_csv(“customer churn prediction.csv”)

x =data.drop(“churn”, axis =1)

y = data[“churn”]

x\_train,x\_test,y\_train,y\_tesst = train\_test\_split(x,y,test\_size=0.5,random\_state=50)

model = RandomForestCassifier(random\_state = 50)

model.fit(x, y)

y\_pred = model.predict(x\_test)

accuracy = accuracy\_score(y\_test,y\_pred)

print( F“accuracy:{accuracy:2f}”)

print(“classification report : \n”, classification\_report(y\_test,y\_pred))

CONCLUSION:

conlusion deploying a machine learning model with IBM Cloud Watson Studio offers a seamless integration of model development and deployment. The platform provides a collaborative environment for data scientists and developers, streamlining the process from data preparation to model deployment. With features like AutoAI and model monitoring, it enhances efficiency and ensures the continued performance of deployed models. Additionally, the cloud-based nature of Watson Studio allows for scalability and accessibility, making it a robust choice for organizations looking to leverage machine learning in their applications.