**A blue and white flag with a logo

Description automatically generated**

**Frontier Tech** Leaders **Global Cohort Machine Learning Bootcamp #2**

**Title:**

Reducing Urban Poverty through

Economic Data Analysis

**Group9**

BY

Osamah SHARAF ALDEEN

Hadeel TARQI

**Machine Learning Project Documentation**

**Deployment**

**1. Overview**

The deployment phase involves making the machine learning model available for real-world use. This phase includes model serialization, model serving, API integration, implementing security measures, and setting up monitoring and logging to ensure the model performs reliably and securely in a production environment.

**2. Model Serialization**

Model serialization is the process of converting the trained machine learning model into a format that can be easily stored and loaded for making predictions. For this project, the model was serialized using the Joblib library, which efficiently handles large NumPy arrays and is well-suited for Scikit-Learn models.

**Serialization Code:**

import joblib

# Serialize the trained model to a file

joblib.dump(model, 'linear\_regression\_model.pkl')

joblib.dump(label\_encoders, 'label\_encoders.pkl')

joblib.dump(imputer, 'imputer.pkl')

# Load the model from the file (example usage)

linear\_regression\_model = joblib.load('linear\_regression\_model.pkl')

label\_encoders = joblib.load('label\_encoders.pkl')

imputer = joblib.load('imputer.pkl')

The model was saved in the 'linear\_regression\_model.pkl' file, ensuring that it can be quickly loaded for inference without the need for retraining.

**3. Model Serving**

The serialized model is served using a web framework, allowing it to handle prediction requests. Gradio was chosen for this project due to its simplicity and ease of creating interactive web interfaces.

**Model Serving Code with Gradio:**

import gradio as gr

import pandas as pd

import joblib

# Load the trained model (ensure the file exists in the same directory)

linear\_regression\_model = joblib.load('linear\_regression\_model.pkl')

# Load the label encoders (ensure the file exists in the same directory)

label\_encoders = joblib.load('label\_encoders.pkl')

# Manually extracted unique categories from your dataset

categories = {

'Region': ['East Asia & Pacific', 'Europe & Central Asia', 'Latin America & Caribbean', 'Middle East & North Africa', 'South Asia', 'Sub-Saharan Africa'],

'Latest household survey': ['2010', '2015', '2020'],

'Lending category': ['IDA', 'IBRD'],

'Currency Unit': ['Dollar', 'Euro', 'Yen'],

'2-alpha code': ['US', 'FR', 'JP'],

'System of National Accounts': ['SNA 1993', 'SNA 2008'],

'Other groups': ['HIPC', 'Euro area', 'Not HIPC'],

'WB-2 code': ['USA', 'FRA', 'JPN'],

'System of trade': ['Imports CIF, exports FOB', 'Imports and exports CIF'],

'Balance of Payments Manual in use': ['BPM5', 'BPM6'],

'External debt Reporting status': ['Reported', 'Not reported'],

'SNA price valuation': ['Market prices', 'Factor costs'],

}

# Function to make predictions

def predict\_national\_accounts\_reference\_year(\*inputs):

input\_data = pd.DataFrame([inputs], columns=[

'Region', 'PPP survey year', 'Latest household survey',

'Lending category', 'National accounts base year',

'Currency Unit', '2-alpha code', 'System of National Accounts',

'Latest population census', 'Other groups', 'WB-2 code',

'Long Name', 'Short Name', 'Country Code',

'National accounts reference year', 'Alternative conversion factor',

'System of trade', 'Balance of Payments Manual in use',

'External debt Reporting status', 'SNA price valuation'

])

# Encode categorical features

for col in label\_encoders:

input\_data[col] = label\_encoders[col].transform(input\_data[col])

# Make prediction

prediction = linear\_regression\_model.predict(input\_data)

return prediction[0]

# Create the Gradio interface

interface = gr.Interface(

fn=predict\_national\_accounts\_reference\_year,

inputs=[

gr.Dropdown(label="Region", choices=categories['Region']),

gr.Number(label="PPP survey year"),

gr.Dropdown(label="Latest household survey", choices=categories['Latest household survey']),

gr.Dropdown(label="Lending category", choices=categories['Lending category']),

gr.Number(label="National accounts base year"),

gr.Dropdown(label="Currency Unit", choices=categories['Currency Unit']),

gr.Dropdown(label="2-alpha code", choices=categories['2-alpha code']),

gr.Dropdown(label="System of National Accounts", choices=categories['System of National Accounts']),

gr.Number(label="Latest population census"),

gr.Dropdown(label="Other groups", choices=categories['Other groups']),

gr.Dropdown(label="WB-2 code", choices=categories['WB-2 code']),

gr.Textbox(label="Long Name"),

gr.Textbox(label="Short Name"),

gr.Textbox(label="Country Code"),

gr.Number(label="National accounts reference year"),

gr.Number(label="Alternative conversion factor"),

gr.Dropdown(label="System of trade", choices=categories['System of trade']),

gr.Dropdown(label="Balance of Payments Manual in use", choices=categories['Balance of Payments Manual in use']),

gr.Dropdown(label="External debt Reporting status", choices=categories['External debt Reporting status']),

gr.Dropdown(label="SNA price valuation", choices=categories['SNA price valuation']),

],

outputs=gr.Textbox(label="Predicted National Accounts Reference Year"),

title="National Accounts Reference Year Prediction",

description="Enter the socio-economic indicators to predict the National Accounts Reference Year of a country using the trained Linear Regression model."

)

# Launch the interface

interface.launch()

##### 4. Security Considerations

Security measures implemented during deployment include:

* **Authentication and Authorization:** Ensuring that only authorized users can access the API.
* **Encryption:** Using HTTPS to encrypt data in transit.
* **Data Privacy:** Ensuring that the data does not contain personally identifiable information (PII).

##### 5. Monitoring and Logging

Monitoring and logging are crucial for maintaining the performance and reliability of the deployed model. Metrics tracked include prediction latency, error rates, and resource usage. Alerts are set up to notify the team of any anomalies or performance issues.