DATA SCIENCE with OESON LEARNING

Global Temperature Anomaly Prediction & Analysis Dashboard

Scenario:

As a Data Science Intern, you are tasked with developing an end-to-end machine learning pipeline that predicts global temperature anomalies based on historical data. The project will include building a predictive model using **Machine Learning (ML)**, deploying it with **Streamlit**, and visualizing the results using **Tableau/Power BI**. The objective is to create a comprehensive tool that allows users to analyze global temperature anomalies, make future predictions, and visually explore climate trends.

Dataset Description:

This dataset provides **monthly temperature anomalies** from 1880 to the present, with each row representing a different year. The temperature anomalies are compared to a long-term baseline (typically the 20th-century average) and are measured in **degrees Celsius**.

Columns in the Dataset:

1. **Year:** The year of observation.
2. **Month Columns (Jan-Dec):** The temperature anomaly for each month (from January to December).
3. **J-D:** The **annual mean temperature anomaly** (based on all 12 months of the year).
4. **D-N:** The **December to November** temperature anomaly.
5. **Seasonal Columns (DJF, MAM, JJA, SON):**
    - **DJF:** Winter temperature anomaly (December, January, February).
    - **MAM:** Spring temperature anomaly (March, April, May).
    - **JJA:** Summer temperature anomaly (June, July, August).
    - **SON:** Fall temperature anomaly (September, October, November).

****NOTE: Late Submission will have an effect on their scores, no extension will be given****

Tools and Technologies:

- **Python** (for data processing and machine learning)
    - **Pandas** and **NumPy** (data manipulation)
    - **Scikit-learn** (machine learning)
    - **Matplotlib**, **Seaborn** (visualization)
    - **Statsmodels** (statistical modeling)
- **Streamlit** (for deployment as an interactive web application)
- **Tableau/Power BI** (for advanced visualizations and dashboards)
- **PyCharm** (IDE for coding)

Project Phases:

Phase 1: Data Exploration and Preprocessing

1. **Data Loading:**
    - Load the global temperature anomaly dataset into a Pandas DataFrame.
    - Inspect the data for consistency, null values, and proper formats.
2. **Data Cleaning:**
    - Handle missing values in the dataset (e.g., forward fill, imputation).
    - Remove any rows with extreme outliers or duplicates if necessary.
    - Ensure all temperature anomaly columns are in numeric format.
3. **Feature Engineering:**
    - Create new columns for **yearly moving averages** (e.g., 5-year moving average) to capture long-term trends.
    - Extract time-based features, such as **year**, **month**, and **season**.
    - Encode seasonal information into categorical variables for machine learning models.
    - Normalize or scale the data to prepare it for training.

Phase 2: Exploratory Data Analysis (EDA)

1. **Visualizing Trends:**
    - Create **time series plots** to show the **global temperature anomaly trends** over time.
    - Compare temperature anomalies for different months and seasons using **line plots**.
    - Plot the **distribution of temperature anomalies** to understand how anomalies have evolved over the years.
    - Use a **correlation heatmap** to explore relationships between different months and seasonal temperature anomalies.
2. **Insights from EDA:**
    - Identify key trends such as the overall warming of the planet.
    - Determine if certain months or seasons have experienced more pronounced warming or cooling.
    - Look for any patterns in the temperature anomaly distribution (e.g., outliers, spikes, or dips).

3. **Advanced Visualizations (Tableau/Power BI):**
    - Import the processed dataset into **Tableau** or **Power BI** to create interactive visualizations.
    - Develop a dashboard that shows:
        - Yearly temperature anomaly trends.
        - Seasonal temperature anomaly comparisons.
        - Temperature anomaly predictions (once the model is built).

Example Visuals:

- Heatmap of monthly temperature anomalies.
- Interactive graphs for season-to-season temperature variations.
- Line plots comparing **actual** vs **predicted** temperature anomalies.

Phase 3: Machine Learning Model Development

1. **Model Selection:**
    - Choose appropriate machine learning algorithms for predicting temperature anomalies, such as:
        - **Linear Regression** (for a simple baseline model)
        - **Random Forest Regressor** (for capturing non-linear patterns)
        - **XGBoost** (for better performance on large datasets)
2. **Data Splitting:**
    - Split the data into **training** and **testing** sets (80% training, 20% testing).
    - Perform **cross-validation** to avoid overfitting and improve model generalization.
3. **Model Training:**
    - Train your machine learning model using the training data.
    - Evaluate the model performance on the test data using metrics like **Mean Absolute Error (MAE)**, **Mean Squared Error (MSE)**, and **R-squared**.
    - Fine-tune the model using **hyperparameter optimization** (e.g., grid search or random search).
4. **Model Evaluation:**
    - Compare model performance (Linear Regression, Random Forest, and XGBoost) to identify the best-performing model.
    - Visualize model performance using **residual plots** and **prediction vs. actual** plots.

Phase 4: Model Deployment with Streamlit

1. **Deploying the Model:**
    - Create a **Streamlit app** to deploy the trained model for making future temperature anomaly predictions.
    - Build a simple user interface where users can:
        - Input a year and retrieve the predicted temperature anomaly for that year.
        - View a plot comparing the predicted vs actual temperature anomalies over time.
2. **Interactive Features:**
    - Add interactive elements like **sliders** to select years, **dropdowns** for selecting months/seasons, and **date pickers**.
    - Include a section displaying insights based on the model predictions, such as:
        - How much the temperature anomaly is expected to change in the next few years.
        - The model's confidence in its predictions.

Phase 5: Final Reporting and Dashboard (Tableau/Power BI)

1. **Interactive Dashboard:**
    - Develop an interactive dashboard using **Tableau** or **Power BI** that provides an overview of the global temperature anomaly analysis and predictions.
    - Use the following components:
        - **Line charts** to show yearly trends.
        - **Heatmaps** for month-to-month and season-to-season comparisons.
        - **Forecast charts** that show predictions made by the model.
2. **Key Insights:**
    - Highlight trends such as significant warming, cooling, or seasonal shifts in temperature.
    - Include insights from the machine learning model, like predicted anomalies for the upcoming years.

Phase 6: Documentation and Final Presentation

1. **Code Documentation:**
    - Properly document the code with comments, explanations, and assumptions made throughout the process.

- Create a **README file** with clear instructions on how to run the code, use the Streamlit app, and interact with the dashboard.

2. **Final Report:**
   - Write a final report summarizing the entire project:
     - **Data Preprocessing** steps.
     - Key findings from **Exploratory Data Analysis**.
     - **Model Development** and evaluation results.
     - **Deployment** using Streamlit.
     - Key insights gained from the project and how they can be used to understand climate trends.
3. **Presentation:**
   - Prepare a **PowerPoint presentation** or **Jupyter Notebook** walkthrough to present your project to stakeholders.
   - Include visualizations, model evaluation results, and demonstrate how users can interact with the Streamlit app and Tableau/Power BI dashboard.

Deliverables:

1. **Python Code**
2. **Streamlit App**
3. **Dashboard (Tableau/Power BI): Links and Screenshot of Dashboard in presentations**
4. **Final Report and Presentation**
5. **LinkedIn Post (Separate File Added for Reference)**