## $CO_2$ Emissions Prediction using Machine Learning

This project focuses on predicting $CO_2$ emissions using machine learning models. The goal is to analyze historical emissions data across various countries and use robust modeling techniques to forecast future emission trends. The analysis provides insights that can support climate policy and environmental planning.

### 📁 Dataset

- Source: [Kaggle - $CO_2$ Emissions by Country](https://www.kaggle.com/)
- Fields include:
  - `Country`
  - `Year`
  - `Emissions`
  - Additional: `Data source`, `Sector`, `Gas`, `Unit`

### 🔧 Data Preprocessing

- **Reshaping**: Transformed from wide format to long format using `pandas.melt()`
- **Type Conversion**: `Year` to `int`, `Emissions` to `float`
- **Missing Values**: Forward-filled within each `Country` and `Sector` group, followed by `.dropna()`
- **Normalization**: Applied `MinMaxScaler` on emissions
- **Log Transformation**: Used `np.log1p()` for skew correction
- **Train-Test Split**: 80% training, 20% testing

### 📊 Exploratory Data Analysis (EDA)

Visualizations (in code) include:
- Time series plots by country/sector
- Distribution of emissions (histograms, box plots)
- Correlation analysis

### 🤖 Models Used

1. **Linear Regression**
   - One-Hot Encoding for `Country`
   - Scaled `Year`
   - Used `Pipeline` and `ColumnTransformer`

2. **Random Forest Regressor**
   - Label Encoding for `Country`
   - `Year` not scaled
   - Captures non-linear relationships

3. **XGBoost Regressor**
   - Target Encoding for `Country`
   - `Year` passed as-is
   - Gradient boosting for enhanced performance

## 📈 Evaluation Metrics

- **$R^2$ Score**
- **Mean Absolute Error (MAE)**
- **Root Mean Squared Error (RMSE)**

Each model was evaluated on both training and test datasets using these metrics to ensure robust performance and generalization.

## ✅ Conclusion

- Successfully implemented and evaluated multiple models for predicting $CO_2$ emissions.
- The project demonstrates effective data preprocessing, exploratory analysis, and model deployment strategies.
- Predictive models like Random Forest and XGBoost outperformed simpler models, showing the value of ensemble techniques.

## 🧰 Technologies Used

- Python
- Pandas, NumPy
- Scikit-learn
- XGBoost
- Matplotlib / Seaborn (for EDA)