Report 1: Data Science (Part 1)

1. Introduction

This project aims to predict whether it will rain or not based on historical weather data. Farmers rely on accurate weather forecasts to plan irrigation, planting, and harvesting. However, traditional weather forecasts are not always reliable for hyper-local conditions. To address this, we built a machine learning model using historical weather data to predict rain.

---

2. Data Preprocessing

The dataset contains daily weather observations for 300 days, including:

- **Average Temperature (°C)**
- **Humidity (%)**
- **Average Wind Speed (km/h)**
- **Rain or Not (1 = Rain, 0 = No Rain)**
- **Date**

Steps Taken:

1. **Handled Missing Values**:
    - Filled missing values with the *mean* of each column.
2. **Corrected Incorrect Entries**:
    - Replaced negative values (e.g., negative humidity) with the column mean.
3. **Feature Engineering**:
    - Extracted useful features from the date column, such as *day_of_week* and *month*.
4. **Scaling**:
    - Scaled numerical features (e.g., temperature, humidity) to ensure all features are on the same scale.

---

3. Exploratory Data Analysis (EDA)

We analyzed the dataset to understand the relationships between features and the target variable (*rain_or_not*).

Key Insights:

1. **Temperature and Rain**:

    - Lower average temperatures are more likely to result in rain.
2. **Humidity and Rain**:
    - Higher humidity levels are strongly correlated with rain.
3. **Wind Speed and Rain**:
    - Moderate wind speeds are more likely to result in rain compared to very low or very high wind speeds.

Visualizations:

- **Histograms**: Showed the distribution of temperature, humidity, and wind speed.
- **Box Plots**: Compared humidity and wind speed for rainy and non-rainy days.
- **Heatmap**: Highlighted correlations between features.

---

4. Model Training and Evaluation

We trained and evaluated multiple machine learning models to predict rain.

Models Used:

1. **Logistic Regression**:
    - Simple and interpretable model.
2. **Random Forest**:
    - Handles non-linear relationships and interactions between features.
3. **Gradient Boosting**:
    - Powerful model for complex datasets.

Evaluation Metrics:

- **Accuracy**: Percentage of correct predictions.
- **Precision**: Percentage of correctly predicted rainy days.
- **Recall**: Percentage of actual rainy days correctly predicted.
- **F1-Score**: Balance between precision and recall.
- **ROC-AUC**: Ability to distinguish between rainy and non-rainy days.

Results:

- **Random Forest** performed the best with an accuracy of **65%** and an ROC-AUC score of **0.6**.

---

5. Model Optimization

To improve the model's performance, we:

1. **Tuned Hyperparameters**:
    - Used GridSearchCV to find the best parameters for the Random Forest model.
2. **Feature Engineering**:
    - Added interaction terms (e.g., temperature × humidity) to capture complex relationships.

---

6. Final Output

We used the best model to predict the probability of rain for the next 21 days. The predictions were saved in a CSV file (future_predictions.csv).

---

7. Conclusion

The Random Forest model achieved strong performance in predicting rain. By using this model, farmers can make better decisions about irrigation, planting, and harvesting. Future work could include incorporating real-time data from IoT sensors to improve predictions further.