**Project Title: Delhi Weather Forecast Project**

**Objective:**

The main objective of this project is to predict future weather conditions in Delhi, focusing on mean temperature, humidity, wind speed, and pressure using time series forecasting models like RNN, LSTM, and GRU. These models will capture the temporal dependencies in the data to make accurate predictions.

**Dataset Overview:**

The dataset used is **Daily Delhi Climate Data**, consisting of the following variables:

1. **date**: The date of the weather observation.
2. **meantemp**: The average temperature on a given day.
3. **humidity**: The humidity percentage.
4. **wind\_speed**: The speed of the wind (possibly in km/h or m/s).
5. **meanpressure**: The average atmospheric pressure (in hPa).

There are **1,462 observations** in the dataset spanning from **2013-01-01 to 2017-01-01**.

**Analysis Process:**

**1. Exploratory Data Analysis (EDA):**

* **Loading the Data**: The dataset is loaded using Pandas, and a preliminary inspection is conducted to check for missing values, data types, and column names.
* **Statistical Overview**: Basic statistical analysis, such as the mean, standard deviation, and ranges of key variables, was likely performed to get an understanding of the dataset.

**2. Data Preprocessing:**

* **Handling Missing Values**: No missing values were found in the dataset, which means that the data is complete and ready for modeling.
* **Normalization**: Time series models, particularly neural networks, benefit from normalized data. Variables like temperature, humidity, and wind speed might have been normalized or scaled to improve model performance.

**3. Feature Engineering:**

* **Lagged Features**: In time series modeling, it's common to create lagged features. These features capture the previous day's temperature, humidity, etc., to predict future values.
* **Sliding Window Approach**: A sliding window technique might be applied to frame the time series problem, where a sequence of past days (e.g., the past 5 days of weather) is used to predict the next day.

**4. Model Building:**

This project likely applies three types of neural network architectures for time series forecasting:

**a. RNN (Recurrent Neural Network):**

* RNN is a classic model for sequential data. It processes sequences by passing the output from one time step as input to the next.

**b. LSTM (Long Short-Term Memory):**

* LSTM is a more advanced RNN variant that can capture long-term dependencies in time series data. It's particularly useful when there's a need to remember weather patterns that span over longer periods.

**c. GRU (Gated Recurrent Unit):**

* GRU is another variant of RNN, similar to LSTM but with fewer parameters. GRU might perform well on datasets where computational efficiency is a concern.

**5. Model Evaluation:**

* **Train-Test Split**: The dataset was likely split into training and testing sets, with the training set used to build the model and the testing set for evaluating its performance.
* **Error Metrics**: Common metrics such as **Mean Absolute Error (MAE)**, **Mean Squared Error (MSE)**, and **Root Mean Squared Error (RMSE)** are used to evaluate the accuracy of the models.
* **Cross-Validation**: Cross-validation might be applied to ensure that the model generalizes well to unseen data.

**6. Model Comparison:**

The performance of the RNN, LSTM, and GRU models are likely compared in terms of accuracy and computational efficiency. Visualization techniques such as **loss curves** and **prediction vs actual plots** are used to evaluate the models.