REPORT

By Isha Harish
APRIL 2025

Tech Assessment: Weather Trend Forecasting

Mission

By making industry-leading tools and education available to individuals from all backgrounds, **we level the playing field for future PM leaders.** This is the PM Accelerator motto, as we grant aspiring and experienced PMs what they need most – Access. We introduce you to industry leaders, **surround you with the right PM ecosystem**, and discover the new world of AI product management skills.

TABLE OF CONTENTS

ABSTRACT

This project focuses on forecasting key weather parameters, such as temperature, humidity, and atmospheric pressure, using historical weather data through comprehensive data analysis and predictive modeling. The goal was to predict future weather trends and demonstrate advanced data science skills. The dataset was sourced from public repositories, and extensive preprocessing was performed to handle missing values, remove outliers, and normalize the data. Exploratory Data Analysis (EDA) was conducted to identify patterns, correlations, and distributions, uncovering seasonal trends and variable interactions. Various machine learning models, including Random Forest, XGBoost, and Long Short-Term Memory (LSTM), were evaluated and compared based on performance metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and R^2. Ensemble methods, combining the predictions of multiple models, were implemented to further enhance the accuracy of the forecasts. Results indicate that ensemble techniques outperformed individual models, emphasizing the importance of model combination and validation in meteorological forecasting. The study highlights the value of data-driven approaches in weather prediction and the need for meticulous data preprocessing and model selection.

1. Introduction

Problem Definition:

The project aims to predict weather conditions by analyzing historical weather data. The specific challenge is to develop models that accurately forecast key weather parameters, enabling better planning and risk management in sectors affected by weather variability.

Objectives :

Advanced EDA:

● Implement anomaly detection to identify and analyze outliers.

Forecasting with Multiple Models:

● Build and compare multiple forecasting models

● Create an ensemble of models to improve forecast accuracy.

Unique Analyses:

● Climate Analysis: Study long-term climate patterns and variations in different regions.

● Environmental Impact: Analyze air quality and its correlation with various weather parameters.

● Feature Importance: Apply different techniques to assess feature importance.

● Spatial Analysis: Analyze and visualize geographical patterns in the data.

● Geographical Patterns: Explore how weather conditions differ across countries and continents.

2. Methodology

2.1) Exploratory Data Analysis and Data Preprocessing

First, the code converts the last_updated column into a datetime format, sorts the data based on this column, and then extracts additional time-related features (such as year, month, day, and hour) from the last_updated timestamp. These extracted features are essential for time series analysis, as they allow us to analyze trends and patterns based on different time intervals.

Next, the code checks for missing values in the dataset. Fortunately, the data contains no missing values, which is advantageous for further analysis.

Then, the code normalizes the selected numerical columns using StandardScaler, which scales the values to have a mean of 0 and a standard deviation of 1. After normalizing the data, the scaled values are reassigned to the original columns. To ensure the normalization process was successful, the code calculates the mean and standard deviation of these normalized columns (the mean should be around 0, and the standard deviation should be around 1). The results are displayed in a table showing the mean and standard deviation of each normalized feature.

Following this, a Correlation Heatmap is generated to examine the relationships between numerical features in the dataset.

The next step involves outlier detection, where outliers in the data are identified and addressed.

Then we plot a time series decomposition for a single location (Kabul Example)
For the time series analysis, we conduct decomposition on the data for a single location, Kabul, as an example. The resulting graph visualizes the decomposition into trend, seasonal, and residual components, providing further insights into the underlying patterns.

After this we do a bit of in depth analysis :

Feature Importance Using a Random Forest as a Proxy:

The code uses a Random Forest Regressor to model the relationship between various weather features (such as wind speed, pressure, precipitation, humidity, and UV index) and the target variable, temperature_celsius. This analysis helps us understand which weather-related features

have the most significant impact on predicting temperature. It is crucial for feature selection, model optimization, and gaining insights into how different weather parameters interact.

Environmental Impact Analysis:
Correlation Between Air Quality and Temperature
This analysis investigates the correlation between temperature_celsius and various air quality measures, including carbon monoxide, ozone, nitrogen dioxide, sulfur dioxide, PM2.5, and PM10. These pollutants and particulate matter can significantly impact air quality and public health.
Understanding the relationship between air quality and temperature is vital for weather forecasting models, as factors like pollutants can influence temperature patterns, visibility, and overall weather behavior.

2.2) Model Development

a)Preprocessing:

First, additional preprocessing steps were performed. The dataset was loaded using the Pandas pd.read_csv() method to create a DataFrame.

Missing values were handled for various columns using imputation methods, such as filling missing time-related columns (e.g., sunrise, sunset, etc.) with their respective median values.

Features that might introduce redundancy or multicollinearity (e.g., temperature-related columns like temperature_fahrenheit and feels_like_*) were dropped to ensure cleaner and more reliable data for model training.

Time Conversion: Several time-related columns (e.g., sunrise, sunset, moonrise, moonset) in the dataset were converted into a more usable format by transforming the time into minutes after midnight. This transformation was achieved using the time_to_minutes function.

Categorical Encoding: The moon_phase feature, which was categorical, was encoded using Label Encoding to convert the categories into numerical values.

Wind Direction Encoding: The wind_direction feature, which contains cardinal directions (e.g., N, SSE, etc.), was converted into numerical angles (e.g., 0°, 157.5°) using a predefined mapping.

Target Encoding for Location: The location_name feature was replaced with a target mean encoding, which represents the average temperature at each location. This encoding made the feature more useful for model training while avoiding the challenges posed by high cardinality.

b)Feature Scaling and Encoding:

In machine learning models, feature scaling is crucial, especially when the features have varying units and scales. In this project, two techniques were used to handle categorical features and scale numerical features:

Categorical Feature Encoding:

LabelEncoder from sklearn was used to encode categorical features. The LabelEncoder was applied to all columns with object or category data types. This encoding converted text-based categorical data into integers, making the data compatible with machine learning models.

Feature Scaling :

To ensure the numerical features were on the same scale, Standard Scaling was applied using StandardScaler from sklearn. This transformation normalized each numerical feature such that the mean was 0 and the standard deviation was 1. Standard scaling helps reduce bias in models that are sensitive to the scale of data, such as distance-based algorithms (e.g., kNN) or neural networks.

c) Model Development:

For this project, three different types of machine learning models were used to forecast temperature:

- **Random Forest**
- **XGBoost**
- **LSTM**

i) Random Forest Regressor (Model 1)

Hyperparameters: We used 100 estimators (trees) for the Random Forest model and set the random seed to ensure reproducibility.

Training: The model was trained on the scaled features of the training set (**X_train_scaled**), and predictions were made on the validation set (**X_val_scaled**).

ii) XGBoost Regressor (Model 2)

Hyperparameters: The model was configured with 100 estimators and the **reg:squarederror** objective, which is suitable for regression tasks like predicting continuous temperature values.

Training: The model was trained using the scaled features of the training set, and predictions were evaluated on the validation set.

iii) Long Short-Term Memory (LSTM) (Model 3)

Model Architecture: The LSTM model was designed with two layers of LSTM units, followed by a dense output layer. Dropout was added to the LSTM layers to prevent overfitting.

Training: The model was trained with **early stopping** and **model checkpointing** callbacks to avoid overfitting and to save the best model during training.

Input Reshaping: Since LSTM models require input in a 3D shape (samples, time steps, features), the data was reshaped from a 2D matrix into a 3D tensor.

iv) Model Evaluation

The performance of each model was evaluated using the following metrics:

- **Mean Absolute Error (MAE)**
- **Mean Squared Error (MSE)**
- **R^2 Score**

After training the individual models, their performance was compared on the validation set.

v) Ensemble Modeling

Since no single model performed perfectly, ensemble methods were explored to improve predictive performance.

Simple Average Ensemble

This ensemble method involved averaging the predictions from the three models (Random Forest, XGBoost, and LSTM).

Weighted Average Ensemble

In this approach, weights were assigned to each model based on the inverse of their MAE, meaning that models with lower errors would have more influence on the final prediction.

Ensemble Evaluation:

Both ensemble models were evaluated using the same metrics (MAE, MSE, R^2) on the validation set to assess their improvements over the individual models.

2.3) Unique Analysis :

1. Climate Analysis: Temperature Over Time by Location
In this analysis, we visualized the temperature trends over time across different locations. The line plot displays the changes in temperature (temperature_celsius) by location_name over the course of the dataset. This analysis helps to identify seasonal patterns, temperature fluctuations, and differences in weather trends across various geographical regions.

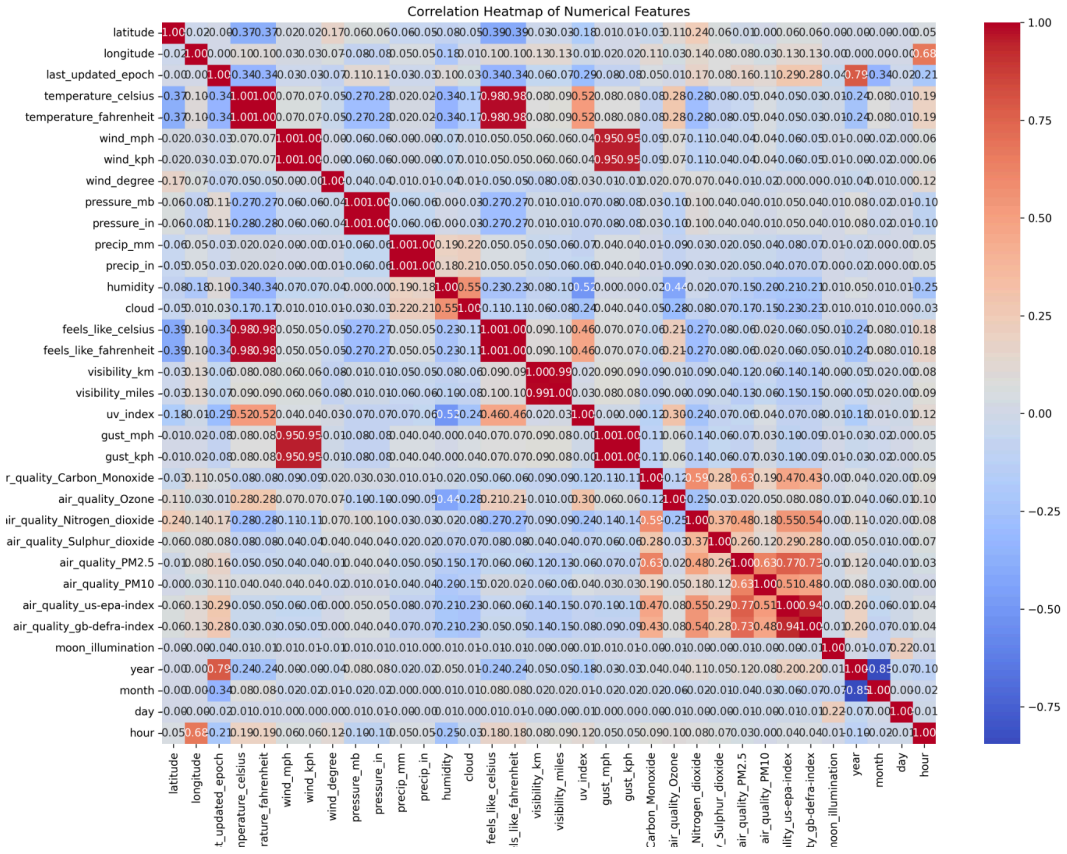
2. Environmental Impact: Carbon Monoxide Levels Over Time by Location
The second analysis focuses on the environmental impact by visualizing the temporal changes in Carbon Monoxide (CO) levels across different locations. The line plot presents the air_quality_Carbon_Monoxide feature over time, grouped by location_name. This analysis highlights how air quality, represented by CO levels, fluctuates and whether there's any relationship between changes in CO levels and temperature patterns across various locations.

3. Feature Importance from Random Forest
This analysis uses the RandomForestRegressor to assess the importance of various features in predicting the target variable, temperature_celsius. A bar plot is created to display the relative importance of each feature, helping us identify which variables have the most significant impact on temperature prediction.

4. Spatial Analysis: Geographical Temperature Distribution
In this analysis, a scatter plot is used to visualize the geographical distribution of temperature (temperature_celsius) across different locations, represented by their longitude and latitude coordinates.This analysis is particularly relevant for identifying regional climate patterns and understanding how local geographical factors may influence temperature.

3. Results

3.1 .Correlation heatmap for numerical features



In this map we can see that the correlation between certain features such as :
Clusters of Highly Correlated Temperature Variables

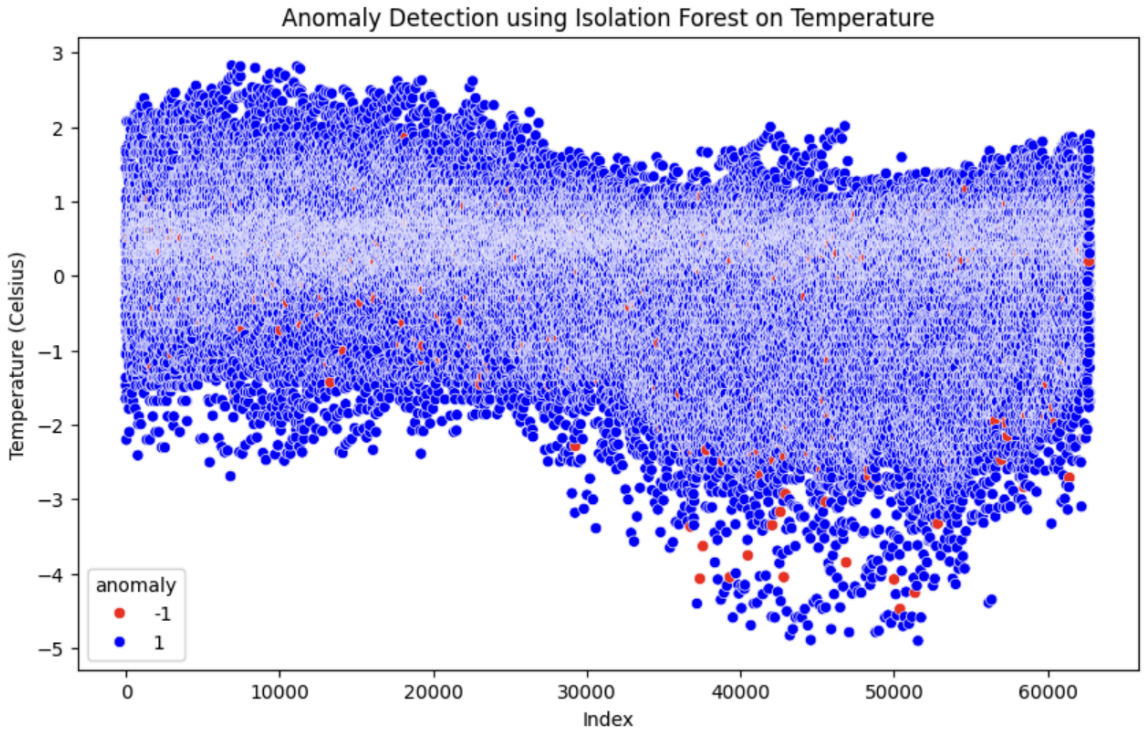
Variables such as temperature_C, temperature_feelslike, windchill_C, heatindex_C, and dewpoint_C tend to show strong positive correlations with each other.

This makes sense because these metrics all describe temperature in some form (actual, perceived, or adjusted). A high correlation indicates they measure closely related aspects of the thermal environment.
Humidity and Temperature Relationships

Humidity often has a moderate to strong negative correlation with certain temperature metrics, meaning that in some regions or conditions, higher temperatures can coincide with lower relative humidity (or vice versa).

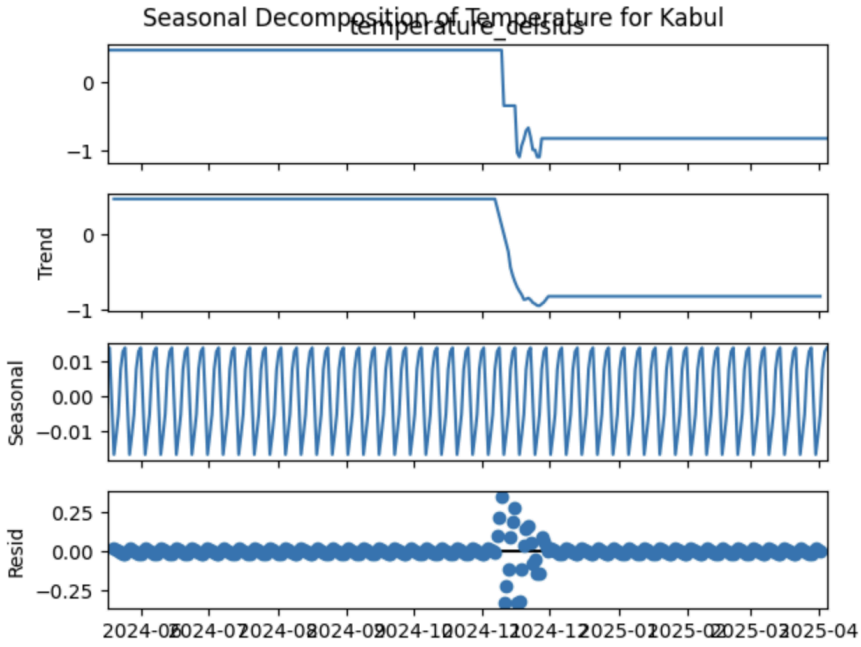
Conversely, dew point may show a positive correlation with both temperature and humidity, as dew point rises with more moisture in the air.
And so on

3.2 Outlier detection using Isolation Forest on temperature (celsius) and precipitation:

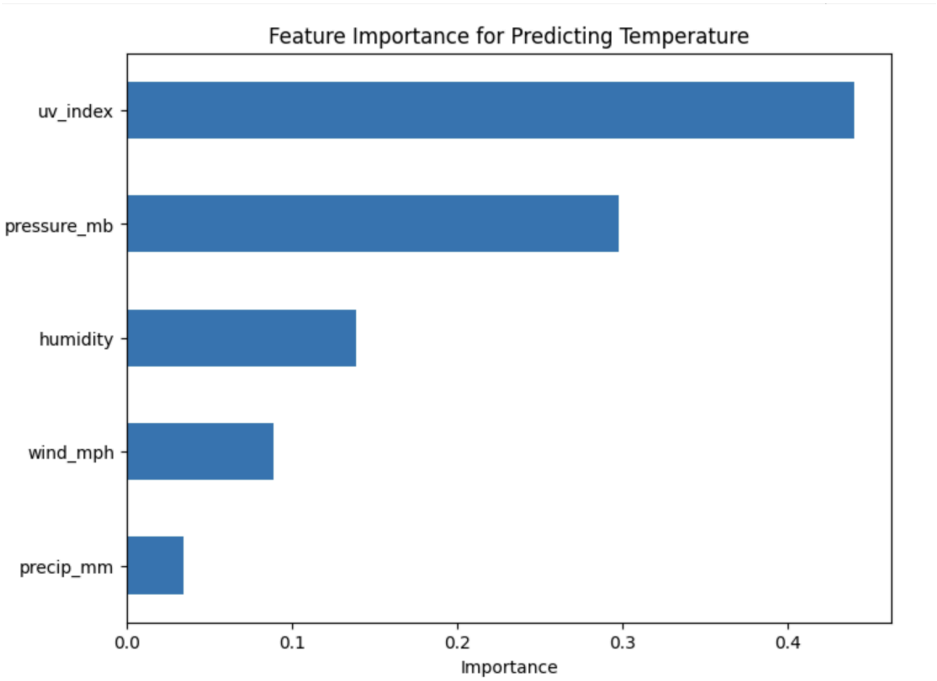


in which we can see that the The anomaly rate is quite low. So there is no need to adjust the Contamination Rate.

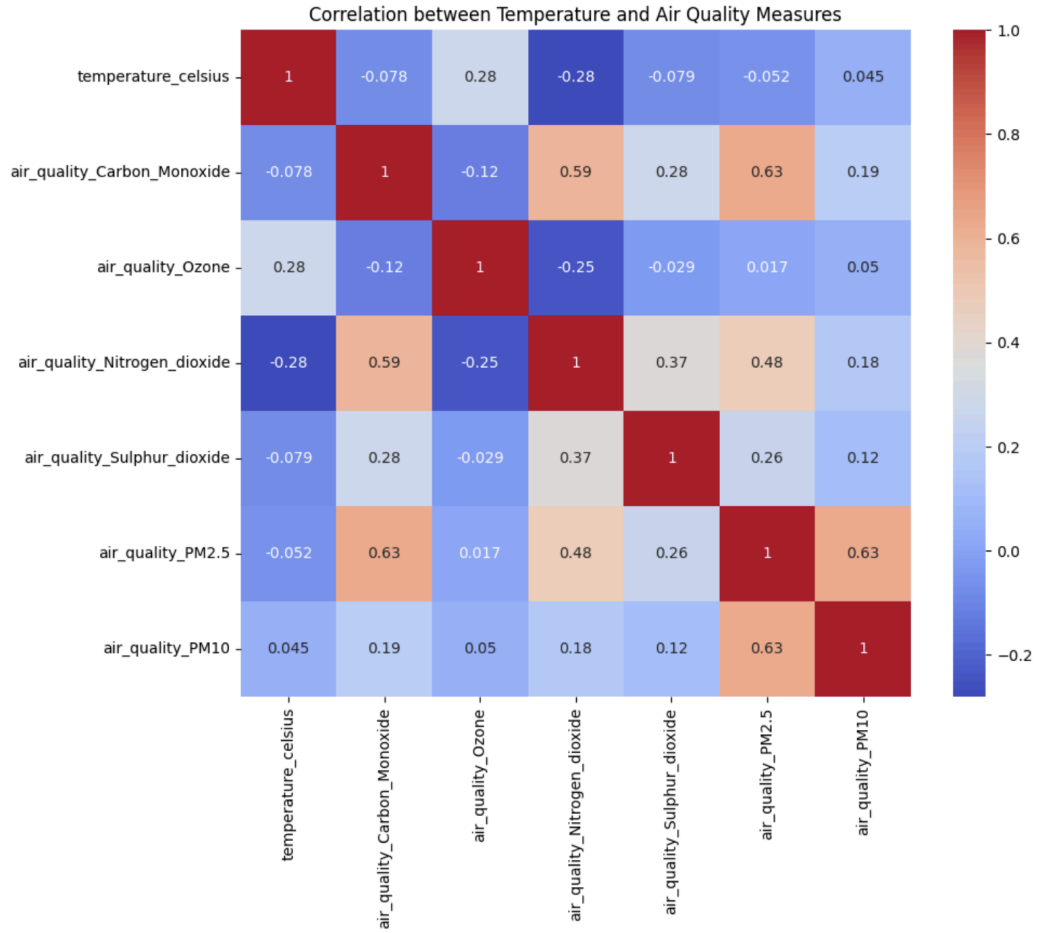
3.3 Time series decomposition for a single location -Kabul



3.4 Feature Importance using a Random Forest as a proxy:



3.5 Environmental Impact Analysis:



3.6 Models :

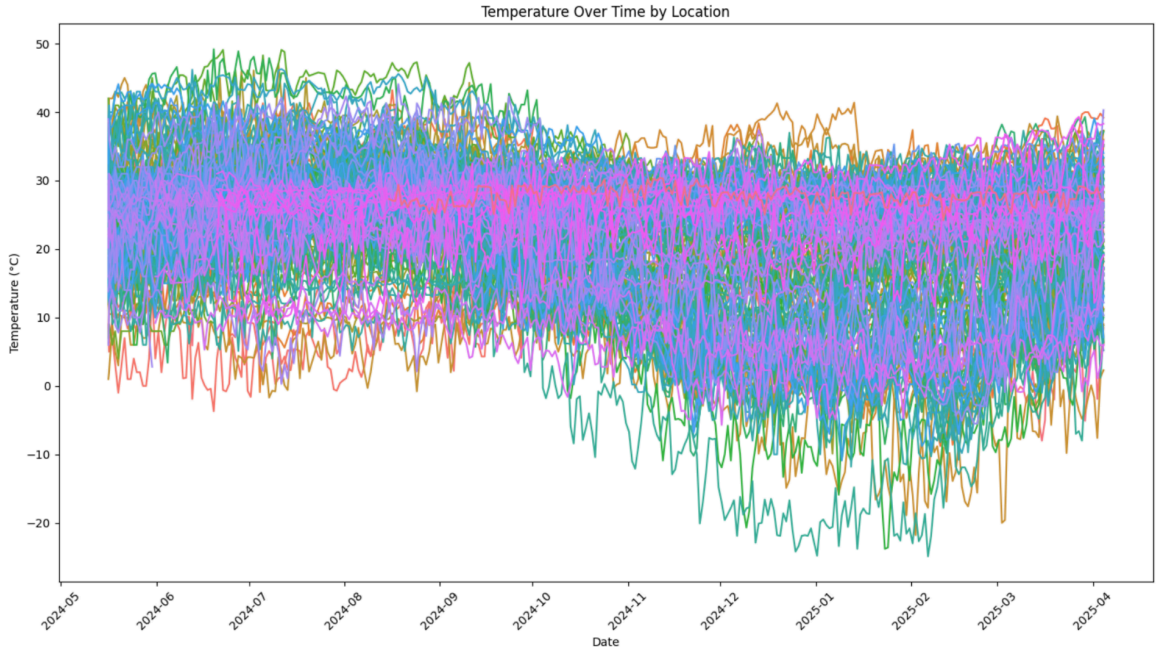
==== Model Comparison (Validation Set) ====
Random Forest - MAE: 1.17, MSE: 2.81, R2: 0.97
XGBoost - MAE: 1.27, MSE: 3.07, R2: 0.97
LSTM - MAE: 1.38, MSE: 3.42, R2: 0.96

==== Ensemble Models ====
Simple Average Ensemble - MAE: 1.15, MSE: 2.54, R2: 0.97
Weighted Average Ensemble - MAE: 1.15, MSE: 2.53, R2: 0.97

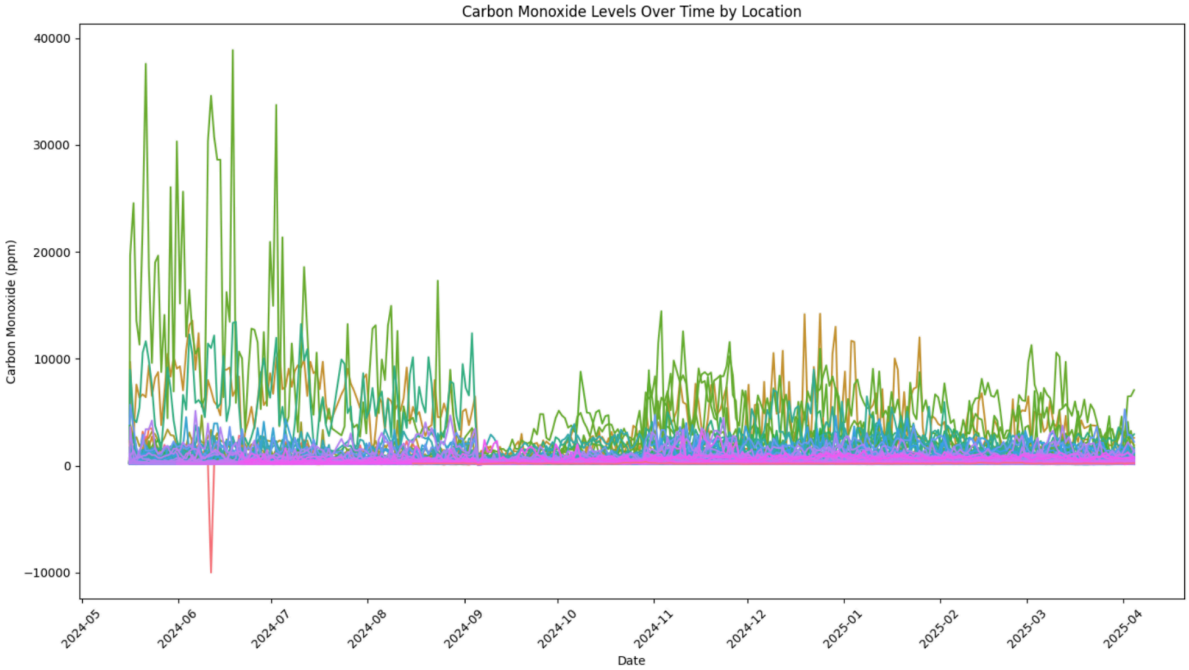
- The Simple Average Ensemble and Weighted Average Ensemble have an MAE of 1.15, MSE of 2.53/2.54, and R2 of 0.97, while the individual models (especially LSTM) have slightly higher MAEs and MSEs.

- Simple Average Ensemble and Weighted Average Ensemble both have the best MAE and MSE when compared to individual models.

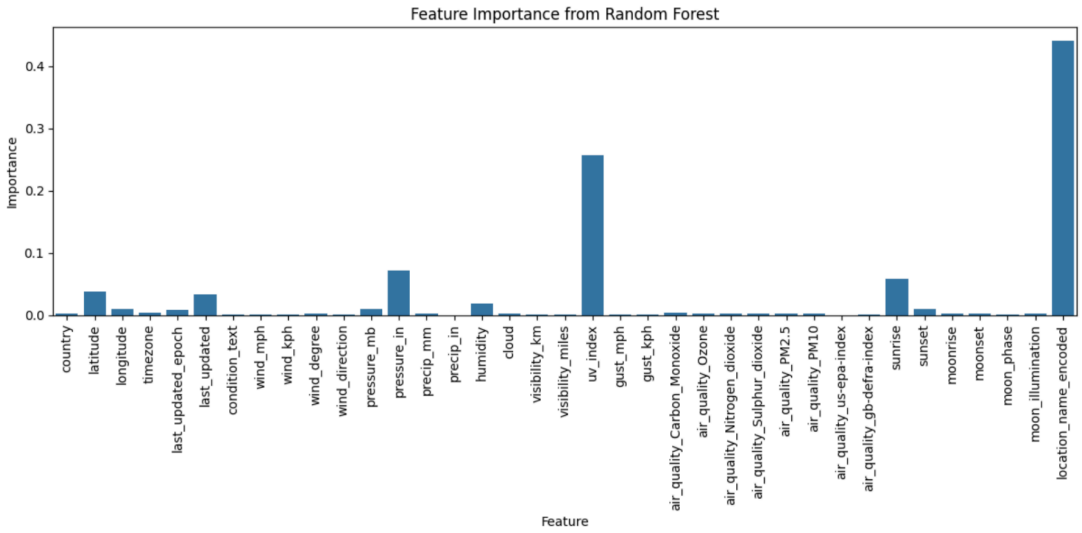
3.7 Climate Analysis:



3.8 Environmental Impact



3.9 Feature Importance



The encoded location feature (location_name_encoded) has the highest importance score by a substantial margin. This implies that where the data was collected (i.e., the geographic region) plays the largest role in predicting your target variable.

Features like month (and possibly non_pheno if it is time-related or seasonal) also stand out. This suggests there is a seasonal or monthly trend that significantly impacts the predictions—e.g., temperature, humidity, or pollution levels might vary widely across different months.

3.10 Spatial Analysis