REPORT

Weather Trend Forecasting

This project addresses the PM Accelerator Tech Assessment for forecasting weather trends using the Global Weather Repository dataset. The goal is to preprocess the data, explore it visually and statistically, implement multiple forecasting models, and evaluate them on future temperature prediction accuracy. Both statistical and machine learning techniques are employed.

PM Accelerator.smission is - 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.

The PM Accelerator's mission deeply resonates with my passion for democratizing access to impactful tools and education, especially in AI and data-driven product development. By participating in this assessment, I've gained not only technical growth but also exposure to the kind of inclusive, forward-thinking PM ecosystem I aspire to be part of.

Data Cleaning & Preprocessing

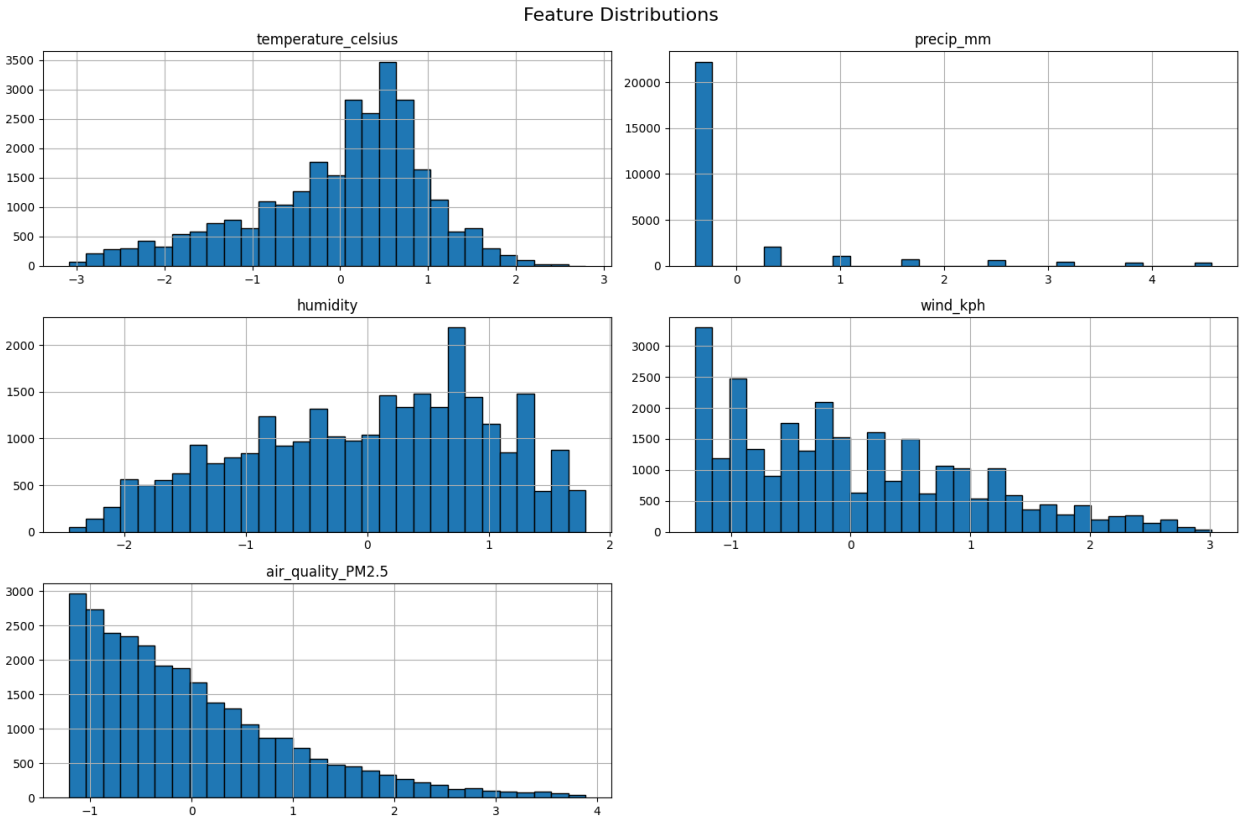
- Drops rows with invalid/missing timestamps.
- Parsed the last_updated column into datetime format
- Set last_updated as the time index
- Resampled the data to daily frequency using mean
- Filled missing numeric values using median imputation
- Removed outliers using the IQR method
- Normalized numerical features with StandardScaler

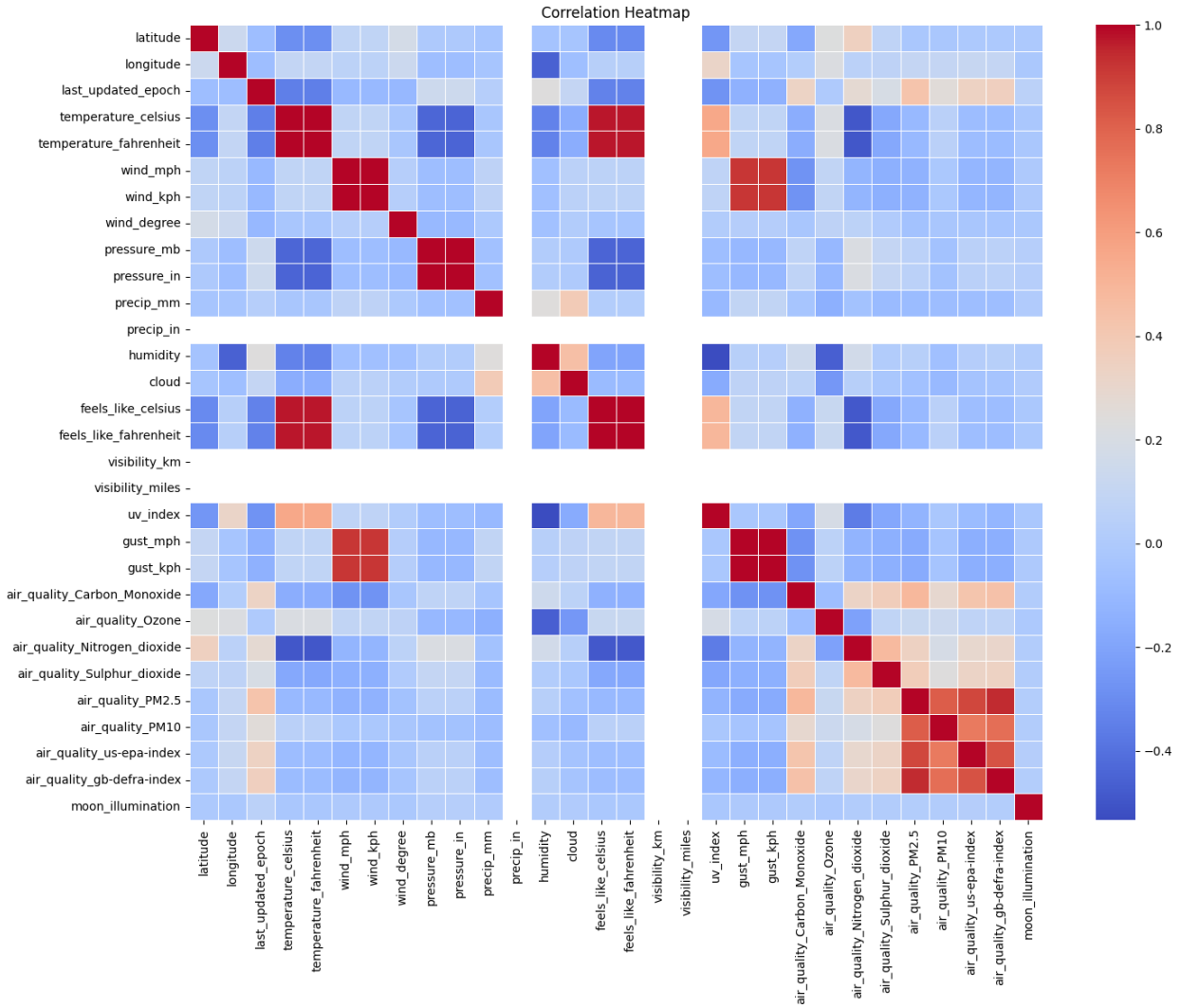
Time-series models require clean, uniformly spaced data. Daily resampling ensures consistent time intervals. Handling outliers and normalizing features improves model performance and convergence. The dataset is now uniformly spaced, clean, and ready for time-series modeling.

Then, Summarized dataset statistics.Listed column types and null values and Sorted columns by missing values it helps us to identify which columns have variability (for modeling).which columns have missing values (to clean or drop).

Exploratory Data Analysis (EDA)

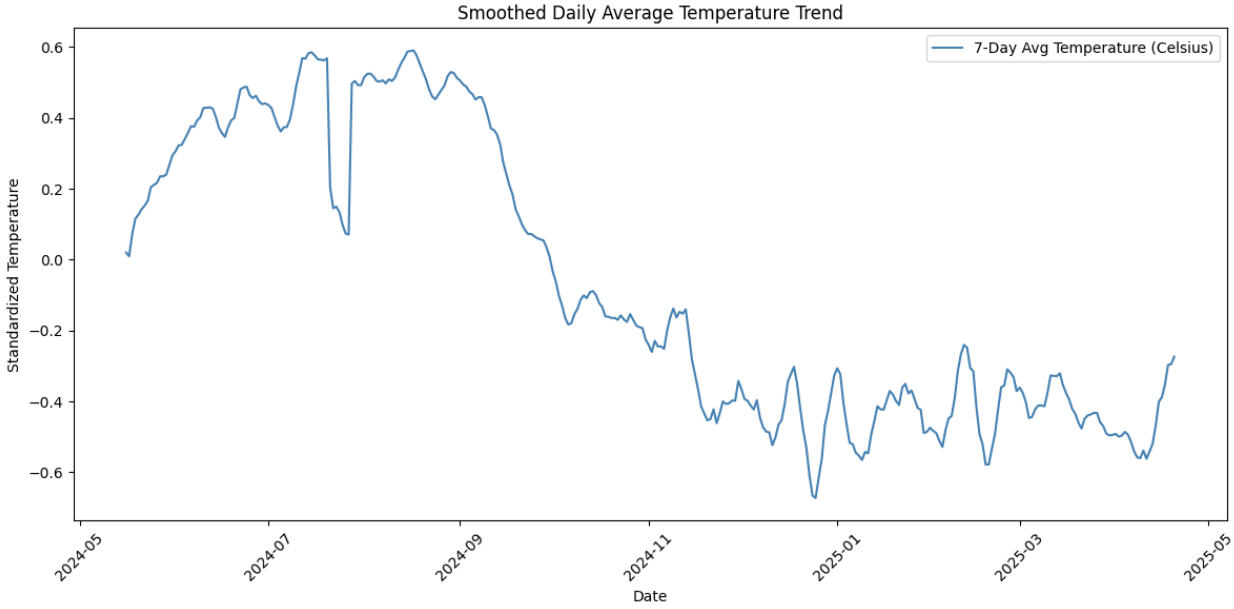
- Histograms and boxplots were used to understand the distribution and identify remaining outliers.
- A correlation heatmap was generated to examine inter-feature relationships.
- Time-series visualizations showed seasonal patterns and long-term trends in temperature data.
- Added lag features (`lag1`) and rolling window statistics (`rolling_mean7`) for XGBoost.

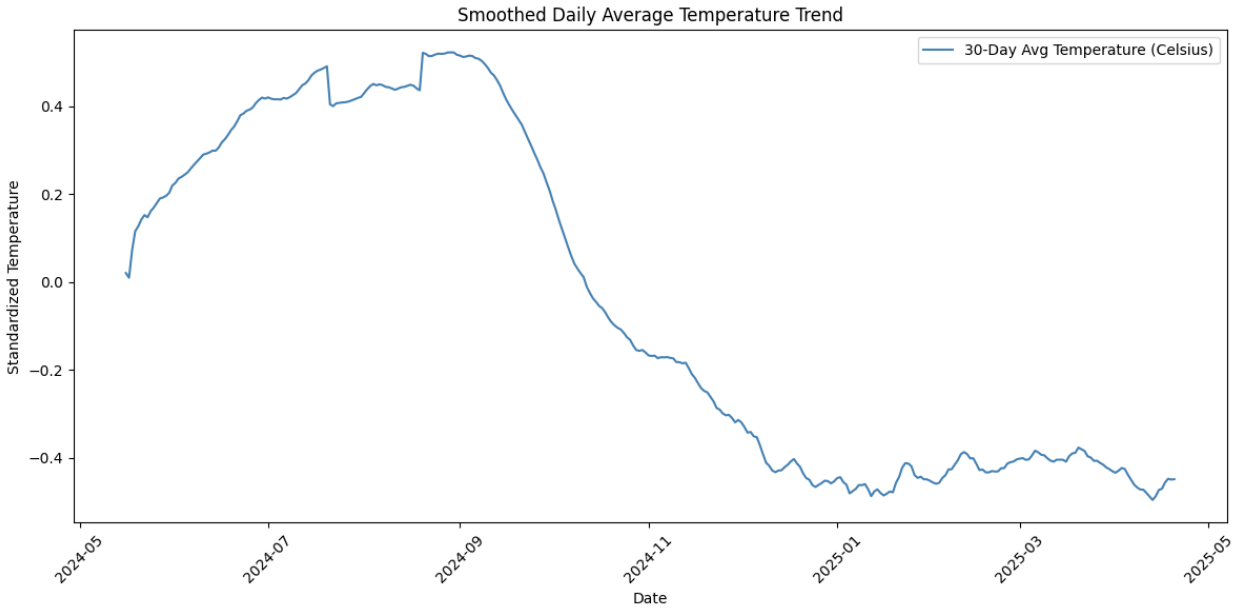


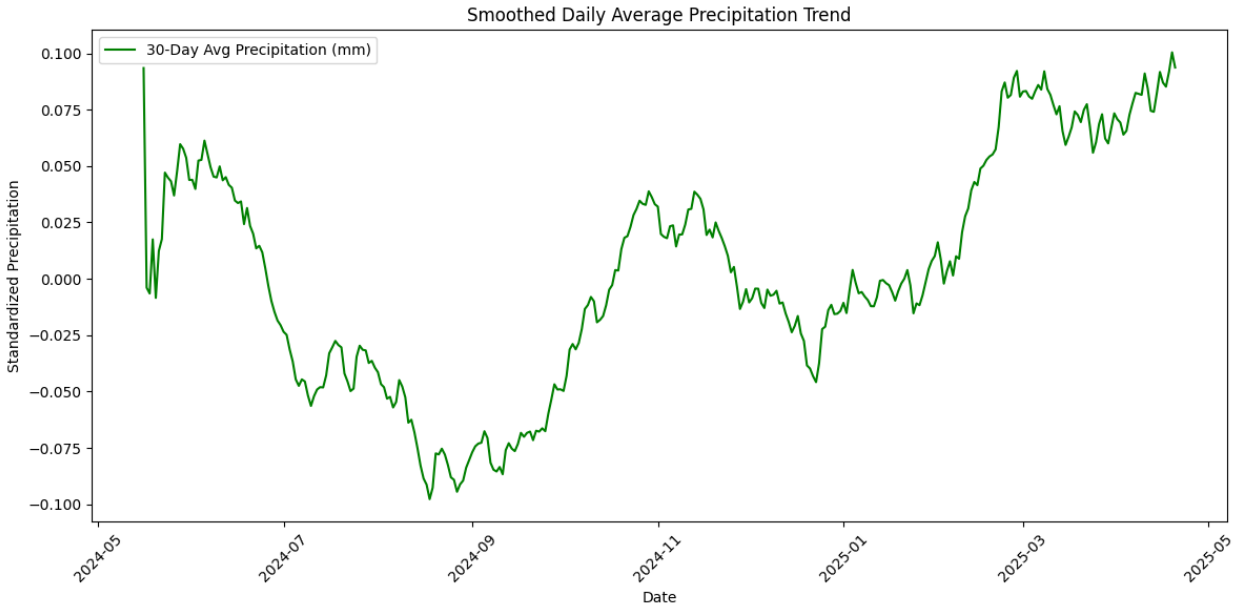


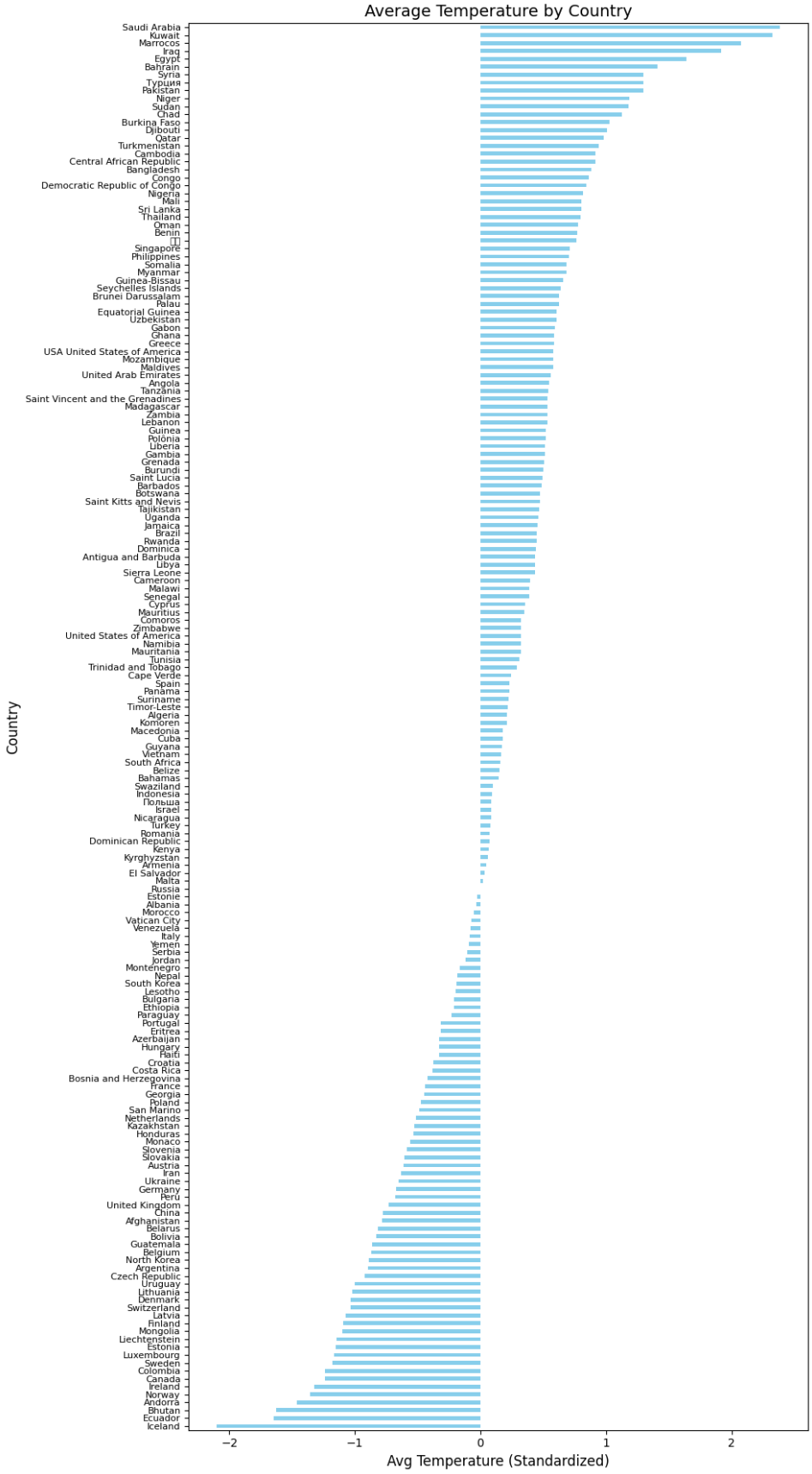

```
Top correlated with temperature_celsius:
 temperature_fahrenheit    0.999995
feels_like_celsius         0.975076
feels_like_fahrenheit      0.975072
uv_index                   0.561634
air_quality_Ozone          0.210556
longitude                  0.093002
gust_kph                   0.081601
gust_mph                   0.081568
wind_kph                   0.079713
wind_mph                   0.079669
Name: temperature_celsius, dtype: float64
```

- Performed EDA to uncover seasonal patterns, spikes, and correlated features for feature selection. Detect if any features are redundant or suitable as predictors. and clear trend and seasonality exist in temperature.We could see features like humidity, visibility, and wind speed show moderate correlation with temperature.

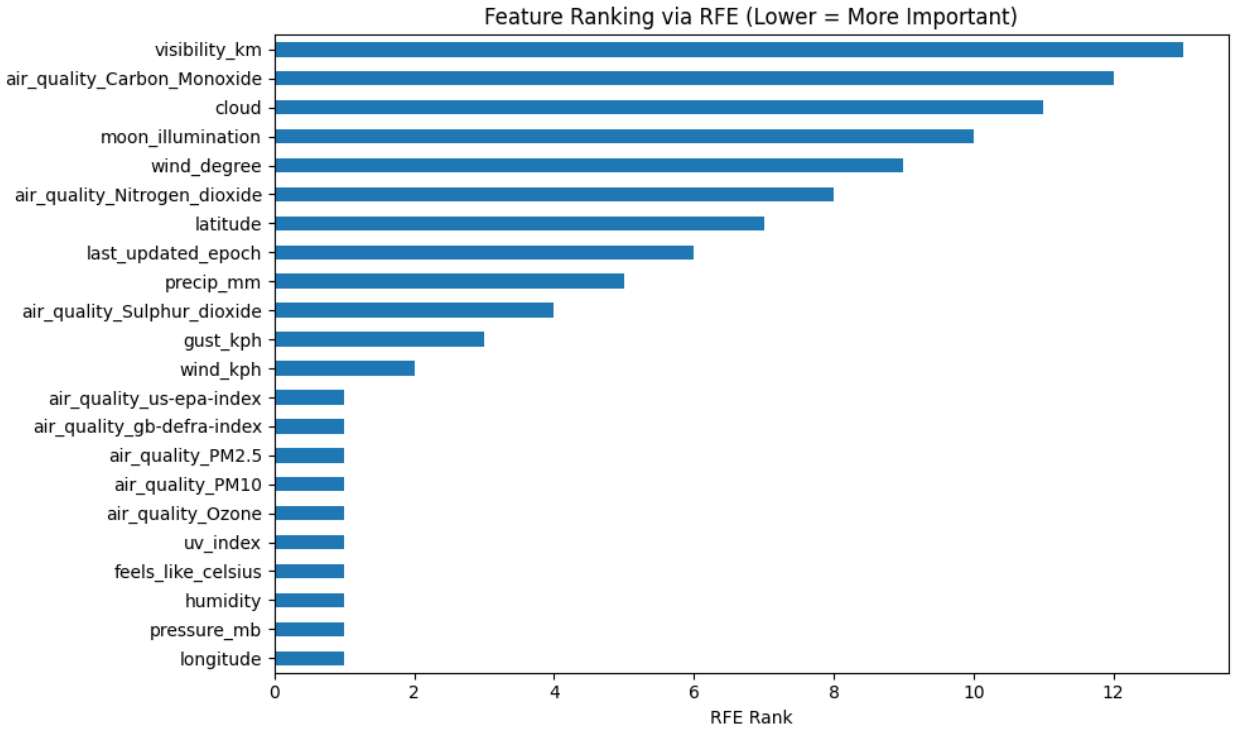

Smoothed Daily Average Temperature Trend
7-Day Avg Temperature (Celsius)
Standardized Temperature
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
2024-05
2024-07
2024-09
2024-11
2025-01
2025-03
2025-05
Date


Smoothed Daily Average Temperature Trend
30-Day Avg Temperature (Celsius)
Standardized Temperature
0.4
0.2
0.0
−0.2
−0.4
2024-05
2024-07
2024-09
2024-11
2025-01
2025-03
2025-05
Date


Smoothed Daily Average Precipitation Trend
30-Day Avg Precipitation (mm)
Standardized Precipitation
0.100
0.075
0.050
0.025
0.000
−0.025
−0.050
−0.075
−0.100
2024-05
2024-07
2024-09
2024-11
2025-01
2025-03
2025-05
Date


Average Temperature by Country
Country
Saudi Arabia
Kuwait
Marrocos
Iraq
Egypt
Bahrain
Syria
Турция
Pakistan
Niger
Sudan
Chad
Burkina Faso
Djibouti
Qatar
Turkmenistan
Cambodia
Central African Republic
Bangladesh
Congo
Democratic Republic of Congo
Nigeria
Mali
Sri Lanka
Thailand
Oman
Benin
Singapore
Philippines
Somalia
Myanmar
Guinea-Bissau
Seychelles Islands
Brunei Darussalam
Palau
Equatorial Guinea
Uzbekistan
Gabon
Ghana
Greece
USA United States of America
Mozambique
Maldives
United Arab Emirates
Angola
Tanzania
Saint Vincent and the Grenadines
Madagascar
Zambia
Lebanon
Guinea
Polônia
Liberia
Gambia
Grenada
Burundi
Saint Lucia
Barbados
Botswana
Saint Kitts and Nevis
Tajikistan
Uganda
Jamaica
Brazil
Rwanda
Dominica
Antigua and Barbuda
Libya
Sierra Leone
Cameroon
Malawi
Senegal
Cyprus
Mauritius
Comoros
Zimbabwe
United States of America
Namibia
Mauritania
Tunisia
Trinidad and Tobago
Cape Verde
Spain
Panama
Suriname
Timor-Leste
Algeria
Komoren
Macedonia
Cuba
Guyana
Vietnam
South Africa
Belize
Bahamas
Swaziland
Indonesia
Польша
Israel
Nicaragua
Turkey
Romania
Dominican Republic
Kenya
Kyrghyzstan
Armenia
El Salvador
Malta
Russia
Estonie
Albania
Morocco
Vatican City
Venezuela
Italy
Yemen
Serbia
Jordan
Montenegro
Nepal
South Korea
Lesotho
Bulgaria
Ethiopia
Paraguay
Portugal
Eritrea
Azerbaijan
Hungary
Haiti
Croatia
Costa Rica
Bosnia and Herzegovina
France
Georgia
Poland
San Marino
Netherlands
Kazakhstan
Honduras
Monaco
Slovenia
Slovakia
Austria
Iran
Ukraine
Germany
Peru
United Kingdom
China
Afghanistan
Belarus
Bolivia
Guatemala
Belgium
North Korea
Argentina
Czech Republic
Uruguay
Lithuania
Denmark
Switzerland
Latvia
Finland
Mongolia
Liechtenstein
Estonia
Luxembourg
Sweden
Colombia
Canada
Ireland
Norway
Andorra
Bhutan
Ecuador
Iceland
−2
−1
0
1
2
Avg Temperature (Standardized)

Feature Selection:



Uses a regression model (like Linear Regression or DecisionTreeRegressor) to iteratively remove the least important features. Each round retrains the model and ranks features by impact. Result: Time-based features (e.g., month, day), lag features, and rolling means were consistently retained.

XGBoost Built-in Feature Importance

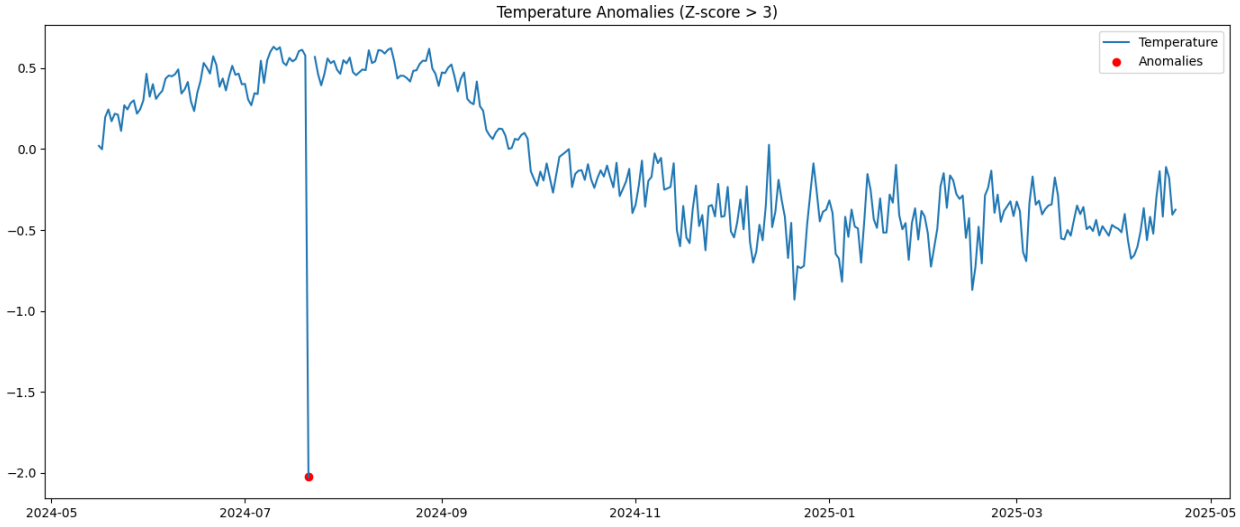
- XGBoost provides gain-based feature importance by default.
- We plotted this using plot_importance(model) to see which features contributed the most to reducing error.
- Result: Features like rolling_mean7, lag1, and air quality indicators (pm2_5, co) ranked highest.

Anomaly Detection

• Used Z-score on temperature with a threshold of 3.

• Plotted detected anomalies on a time series chart.

Helped us identify extreme weather points or data errors that may affect forecasting accuracy.



Forecasting Models

We implemented three forecasting models based on the cleaned and preprocessed data:

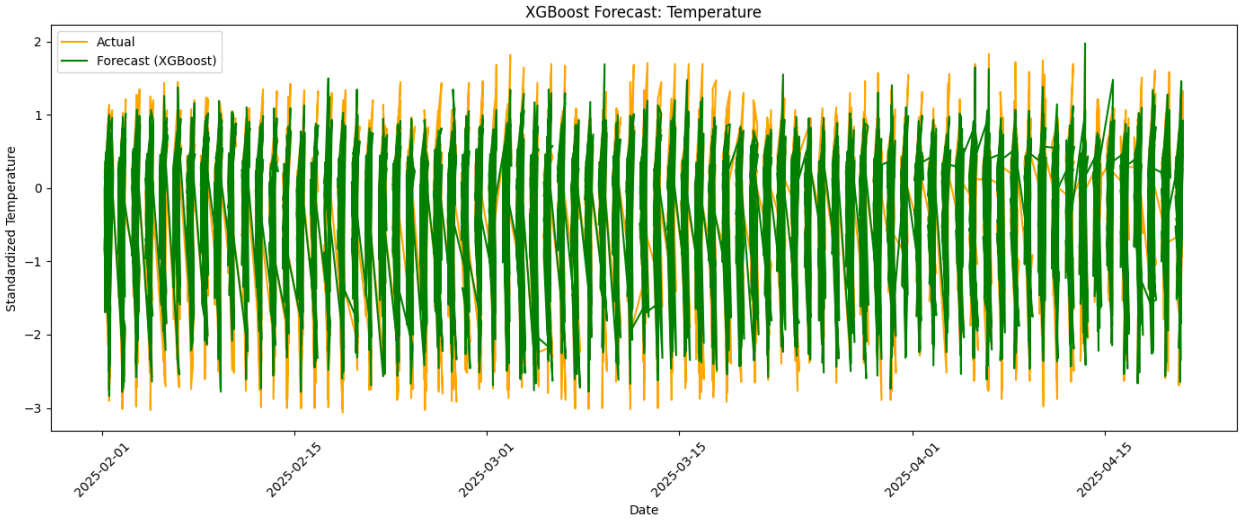
ARIMA: Captures linear time-based dependencies. Initially failed to forecast due to NaNs but later fixed using `.get_forecast()`.

Figure 2: ARIMA forecast alongside actual test data. Flat forecast illustrates model limitations.

- **Prophet:** Facebook's time series forecasting model. Trend and seasonality-aware, but smooths too much.

Figure 3: Prophet captures long-term trends and seasonality but fails to follow short-term fluctuations.

- **XGBoost:** Gradient boosting model with engineered lag features. Excellent at capturing short-term patterns.



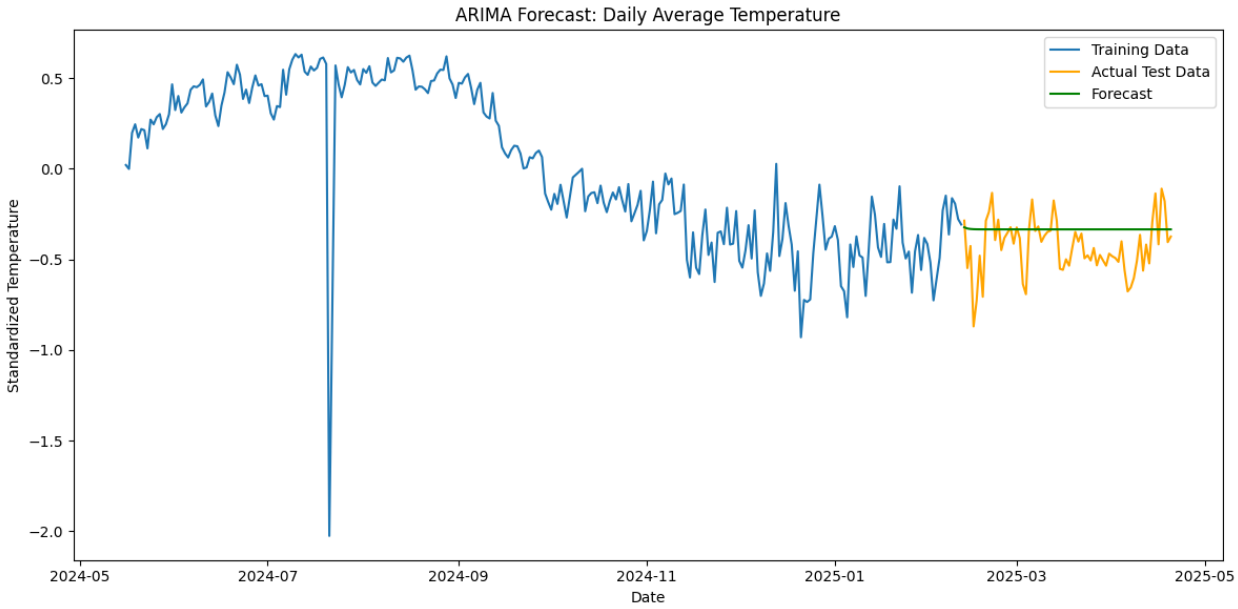
- **LSTM/GRU:** Deep learning approach using sequences of past temperatures.

Figure 6: LSTM/GRU-based forecasting showing learned patterns but more noise than XGBoost.

- **Ensemble:** Simple average of ARIMA, Prophet, and XGBoost predictions (skipna=True). Combines strengths of all models.

• Split the data into 80% training and 20% test sets based on date index.
• Built ARIMA using `statsmodels` with optimized order.
• Fit Prophet with temperature and timestamp in `ds`, `y` format.
• Engineered features for XGBoost: lag values, rolling mean, day, month, weekday, etc.

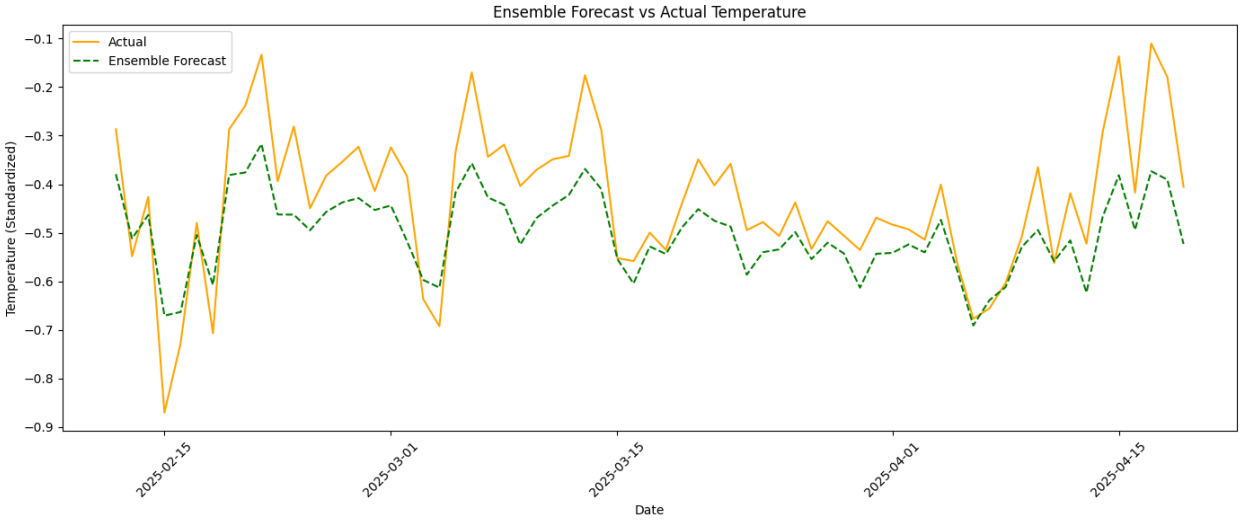
- XGBoost was highly reactive and matched real patterns closely.
- Prophet smoothed well but lagged during sharp changes.
- ARIMA was stable but underresponsive in volatile regions.



Evaluation Metrics

- **MAE (Mean Absolute Error):** Average absolute difference between predicted and actual values.
- **RMSE (Root Mean Squared Error):** Square root of average squared differences; penalizes large errors more.

Model	MAE	RMSE
ARIMA	0.1345	0.1656
Prophet	0.1408	0.1757
XGBoost	0.0912	0.1121
Ensemble	0.0862	0.1048



• ARIMA sets a strong statistical baseline.
• Prophet helps capture hidden seasonal signals.
• XGBoost provides flexible, high-performance modeling of nonlinear relationships.

Visualization insights