The Future of Harveston

By Caramel Labs - Data_Crunch_092
General Sir John Kotelawela Defence University

1. Problem Understanding & Dataset Analysis

The primary objective of this forecasting task is to predict key Spatio-temporal variables which includes Average Temperature, Radiation, Rain Amount, Wind Speed, and Wind Direction. While the rest of the variables act as exogenous variables influencing the target variables. The expected outcome is an accurate forecasting model that can provide insights into weather trends and support decision-making processes in agriculture, disaster management, and climate research.

Key Findings from Data Analysis

Using exploratory data analysis (EDA), we identified key trends and patterns in the dataset. The following techniques were applied:

- **Descriptive Statistics**: Summary statistics revealed mean, median, and standard deviation values for each variable, helping us understand the data distribution.
- **Correlation Analysis**: Pearson correlation coefficients indicated strong relationships among temperature-related variables and moderate correlations with radiation and wind speed.
- **Seasonal Decomposition**: Prophet's decomposition highlighted clear seasonal patterns in temperature and radiation, reinforcing the importance of capturing seasonality in our models.
- **Missing Value Handling**: The dataset was checked to ensure no missing values are present.
- **Outlier Detection**: Box plots and z-score analysis helped detect and remove anomalies, particularly in wind speed readings.
- **Scaling & Transformation**: Normalization was used for neural networks to improve convergence, while differencing was applied to ensure stationarity in ARIMA models.
- **Date Feature Engineering**: A new date column (ds) was created by combining year, month, and day columns, allowing models like Prophet to leverage time-based trends effectively.
- **Temperature Unit Correction**: Temperatures exceeding 100 were converted from Kelvin to Celsius, ensuring consistency in measurements.
- **Geographic Analysis**: A scatter plot of latitude and longitude helped visualize geographic distribution of weather stations.
- **Kingdom Distribution**: A histogram of the kingdom column was used to identify any class imbalances.
- **Feature Correlation**: A heatmap of numerical features highlighted relationships between variables, aiding in identifying feature influences.

2. Feature Engineering & Data Preparation

In time series prediction, especially with Prophet, selecting the right features significantly enhances the model's ability to detect trends, seasonality, and external influences. Below, we detail the key preprocessing steps and justifications for feature selection, focusing on their impact on Prophet's predictive performance.

Data Preprocessing

- **Handling Dates:**
  - The dataset initially had separate columns for `Year`, `Month`, and `Day`.
  - These were combined into a single `ds` (datetime) column for Prophet compatibility.
  - Invalid dates (e.g., February 30) were removed.

- **Temperature Conversion:**
  - The dataset contained temperatures in both **Kelvin and Celsius**.
  - All temperatures >100 (indicating Kelvin) were converted to Celsius using:

$$\textbf{Celsius} = \textbf{Kelvin} - 273.15$$

Feature Creation

- **Lags & Rolling Statistics:**
  - Prophet inherently handles trend and seasonality, so manual lag features were **not explicitly created**.
  - However, external regressors (like `Rain_Amount`, `Wind_Speed`) were incorporated to improve forecasting.

- **Transformations for Stationarity:**
  - Prophet automatically handles **non-stationary data** by decomposing trends and seasonality.
  - No additional differencing or log transformations were applied.
- **Normalization:**
  - Prophet does **not require scaling**, as it is based on an additive model.

3. Model Selection & Justification

Why Prophet?

- **Handles missing data and outliers** gracefully.
- **Automatic seasonality detection** (daily, weekly, yearly).
- Supports **external regressors**.
- Provides **uncertainty intervals** for forecasts.

Alternative Models Considered

- **ARIMA:**
  - Requires manual differencing for stationarity.
  - Struggles with multiple seasonalities.

- **LSTM (Deep Learning):**
  - Requires extensive tuning and more computation.
  - Less interpretable than Prophet.

- **XGBoost:**
  - Needs feature engineering for time dependencies.
  - Less suited for pure time series forecasting.

Hyperparameter Optimization

- **Default settings** were used initially.
- **Key parameters tuned**
  - `yearly_seasonality=True`
  - `weekly_seasonality=True`
  - `daily_seasonality=False`
  - `seasonality_mode='multiplicative'`
  - `growth="flat"`

Validation Approach

- **Time-based cross-validation** (rolling window) was used.
- The dataset was split into **training (past years)** and **testing (most recent data)**.

4. Performance Evaluation & Error Analysis

Evaluation Metrics

- **SMAPE (Symmetric Mean Absolute Percentage Error):** Used as it was the metric in the Kaggle leaderboard.

Model Comparison

Model	Score
Prophet	36
ARIMA	40
XGBoost	40
LSTM	42
Encoder-Decoder LSTM	49

Prophet performed best due to its ability to capture multiple seasonalities and trends effectively.

Residual Analysis

- **Autocorrelation:** Residuals showed **no significant autocorrelation**, indicating Prophet captured temporal patterns well.
- **Normality:** Residuals were **approximately normally distributed**.
- **Heteroscedasticity:** No strong evidence of changing variance over time.

Limitations

- **Struggles with abrupt changes** (e.g., sudden weather shifts).
- **External regressors must be forecasted** if used in future predictions.

5. Interpretability & Business Insights

Applications

- **Agriculture:** Predict optimal planting and harvesting times, improving irrigation and pest control decisions. Prophet's seasonality modeling aids long-term agricultural planning.

- **Energy Management:** Forecast temperature trends to optimize HVAC efficiency and power grid stability. Prophet's external regressor support enhances renewable energy integration.

- **Disaster Preparedness:** Predict extreme weather events, enabling early warnings and infrastructure planning. Prophet's trend detection helps identify high-risk periods.

Improvements

- **Incorporate More Variables:** Adding humidity, pressure, and satellite data improves accuracy, capturing complex weather dynamics.

- **Hybrid Models:** Combining Prophet with XGBoost enhances non-linear pattern detection, improving extreme weather forecasting.
- **Real-Time Updates:** Integrating live data streams and rolling forecasts ensures adaptability to sudden weather changes.

6. Innovation & Technical Depth

This approach enhances time series forecasting through advanced modeling and robust preprocessing:

1. **Optimized Prophet Model** – Utilizes **multiplicative seasonality**, **flat growth**, and **seasonality prior scaling** for better trend and pattern capture.

2. **Intelligent Outlier Handling** – Implements **IQR-based detection**, **rolling median smoothing**, **capped percentile replacement**, and **adaptive linear interpolation** to correct anomalies while preserving temporal consistency.

3. **Hierarchical Prediction Strategy** – Forecasting is structured with an **outer loop per region (kingdom)** and an **inner loop per variable**, ensuring tailored predictions and dynamic test period alignment.

4. **Scalable Multi-Model Strategy** – Prophet, designed for single-variable forecasting, was adapted to run **150 independent models**, iterating over each **kingdom and variable** separately.

These innovations refine forecasting accuracy and reliability while optimizing efficiency.

7. Conclusion

Best Model: Prophet, due to its accuracy, interpretability, and ease of use.

Challenges: Handling sudden weather anomalies and missing external regressor forecasts.

Future Work:

- Test hybrid models (Prophet + LSTM).
- Deploy model in a real-time forecasting system.

Final Recommendation

Prophet is the best choice for this forecasting task due to its **robustness, interpretability, and performance**. Further improvements could involve **hybrid modeling** and **real-time data integration**.