AI-Powered Rainfall Prediction for High-Impact Decision Making

GROUP 2 CAPSTONE PROJECT

1 Business Understanding & Problem Statement

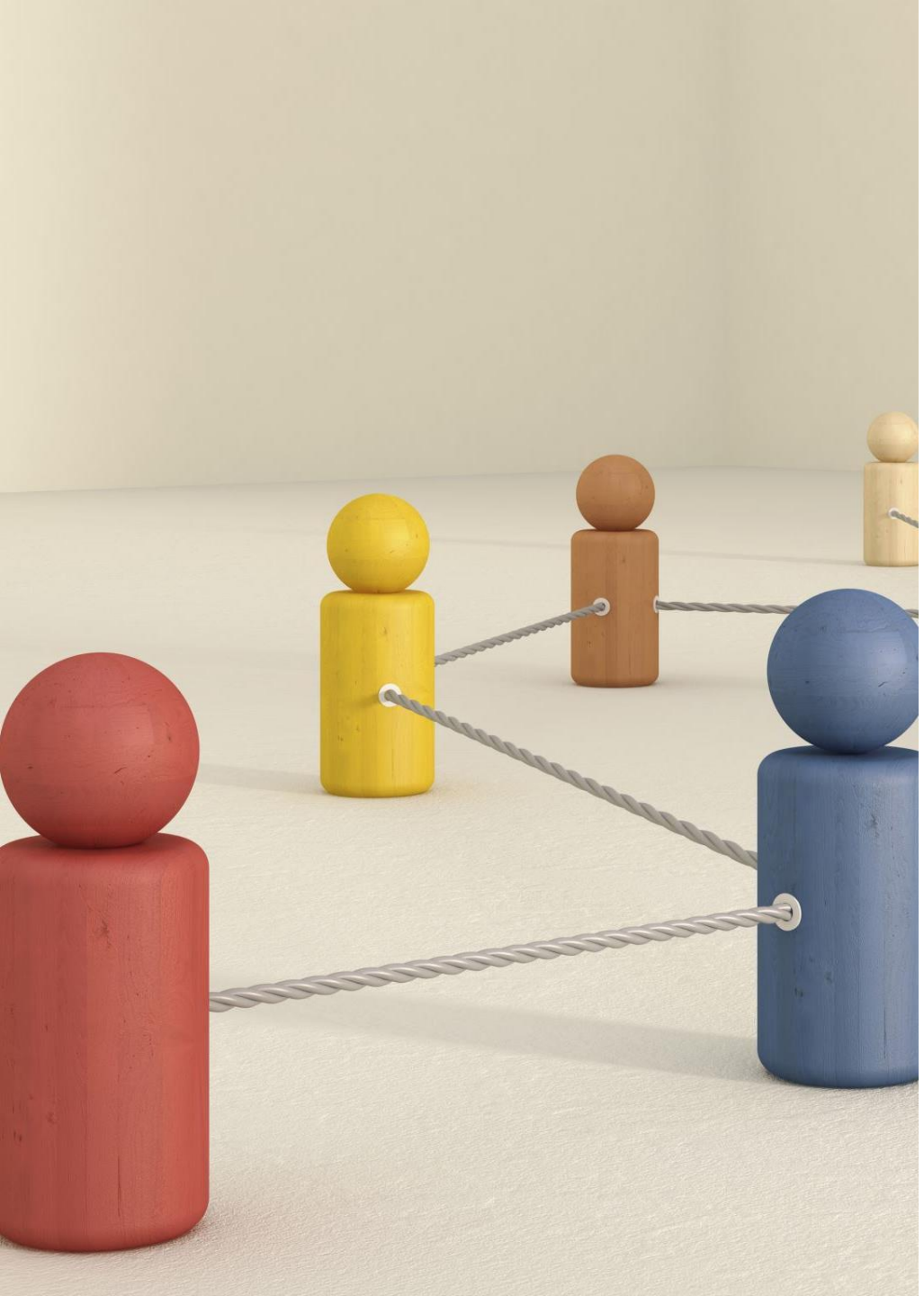
🌎 Context & Motivation

•Accurate rainfall prediction is essential in **agriculture, disaster preparedness, and urban planning**. A missed forecast can lead to **crop failures, infrastructure damage, and economic disruptions**. Traditional weather prediction models often rely on **rule-based approaches**, which fail to capture the **complex, non-linear relationships** between meteorological variables.

•This project takes an **AI-driven approach**, leveraging **advanced machine learning** techniques to develop a high-accuracy **binary classification model** that predicts **rainfall occurrence** with precision.

💪 Why This Matters

- **Farmers & Agribusiness**: Optimize irrigation schedules and reduce crop loss risk.
- **Disaster Management**: Enhance flood forecasting and emergency preparedness.
- **Urban Infrastructure**: Assist city planners in drainage and water resource management.



Project Challenge & Competitive Edge

- This project seeks to address these challenges by adopting a modern, AI-driven approach to rainfall prediction.
- Using advanced machine learning techniques, we aim to develop a high-accuracy binary classification model that can predict rainfall occurrence with unprecedented precision, ultimately improving decision-making across multiple critical sectors.

2 Project Objectives & Key Performance Indicators (KPIs)

🎯 Primary Objective

- Develop a **high-accuracy machine learning model** to predict **rainfall occurrence** (Binary Classification: **Rain = 1, No Rain = 0**).

Secondary Objectives

Investigate underlying weather patterns that influence rainfall

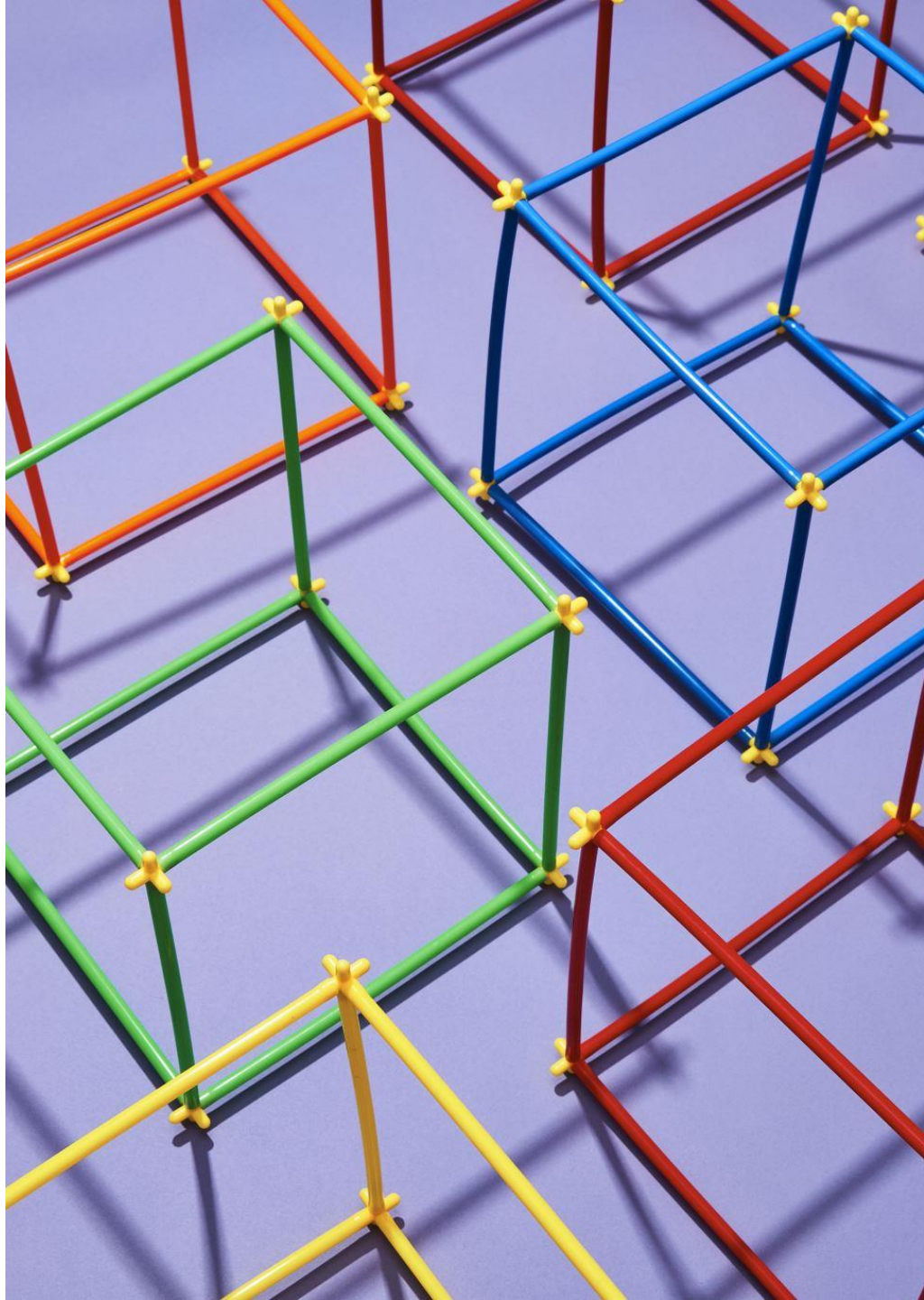
Implement and test various machine learning algorithms (e.g., Logistic Regression, Decision Trees, Random Forest, XGBoost, etc.).

Showcase a comprehensive, end-to-end AI-driven workflow that can be adopted for real-world weather forecasting applications

3 Data Understanding & Competitive Dataset Analysis

📚 Dataset Source & Overview

- This project utilizes **Kaggle's Playground Series - S5E3 competition dataset**, consisting of **historical meteorological data** for predictive modeling.



Dataset Breakdown

- **Train Dataset (train.csv)**: 2,190 samples with 13 features.
- **Test Dataset (test.csv)**: 730 samples with 12 features (excludes rainfall target variable).
- **Submission File (sample_submission.csv)**: Kaggle's submission format for predicted outputs.

📊 Feature Engineering Considerations

Feature	Description & Significance
Day	Sequential identifier (potential time-series dependencies).
Pressure	Atmospheric pressure, influencing rainfall patterns.
Max Temp	Maximum recorded temperature, a potential indicator of precipitation likelihood.
Temperature	Average recorded temperature, linked to evaporation and condensation cycles.
Min Temp	Minimum temperature, useful for analyzing dew point variations.
Dew Point	Key metric for moisture content in the air.
Humidity	Relative humidity (%), highly correlated with rainfall probability.
Cloud	Cloud cover percentage (%), a strong predictor for precipitation.
Sunshine	Total hours of sunshine, inversely affecting rainfall chances.
Wind Direction	Wind direction, impacting weather system movements.
Wind Speed	Wind speed, affecting cloud formation and storm intensity.
Rainfall	**Target Variable** (1 = Rain, 0 = No Rain).

🔬 Initial Observations & Challenges

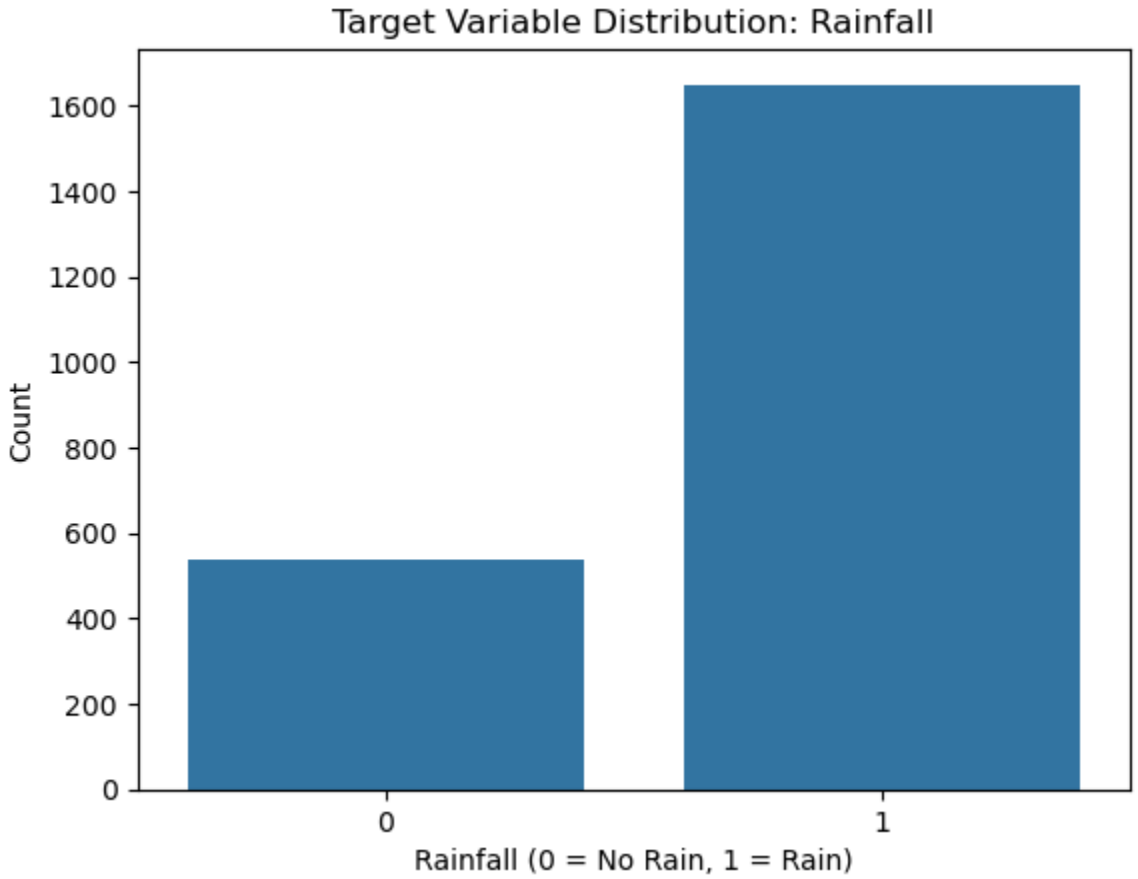
- **All features are numerical**, simplifying preprocessing.
- **Potential Class Imbalance**: Requires resampling techniques (e.g., SMOTE, undersampling).
- **Feature Correlation Analysis**: High correlation expected among ***humidity***, ***dewpoint***, and ***cloud***.
- **Outlier Detection**: Potential extreme values in ***pressure*** and ***windspeed***.
- **Missing Values**: 1 missing value in ***winddirection***, to be imputed.

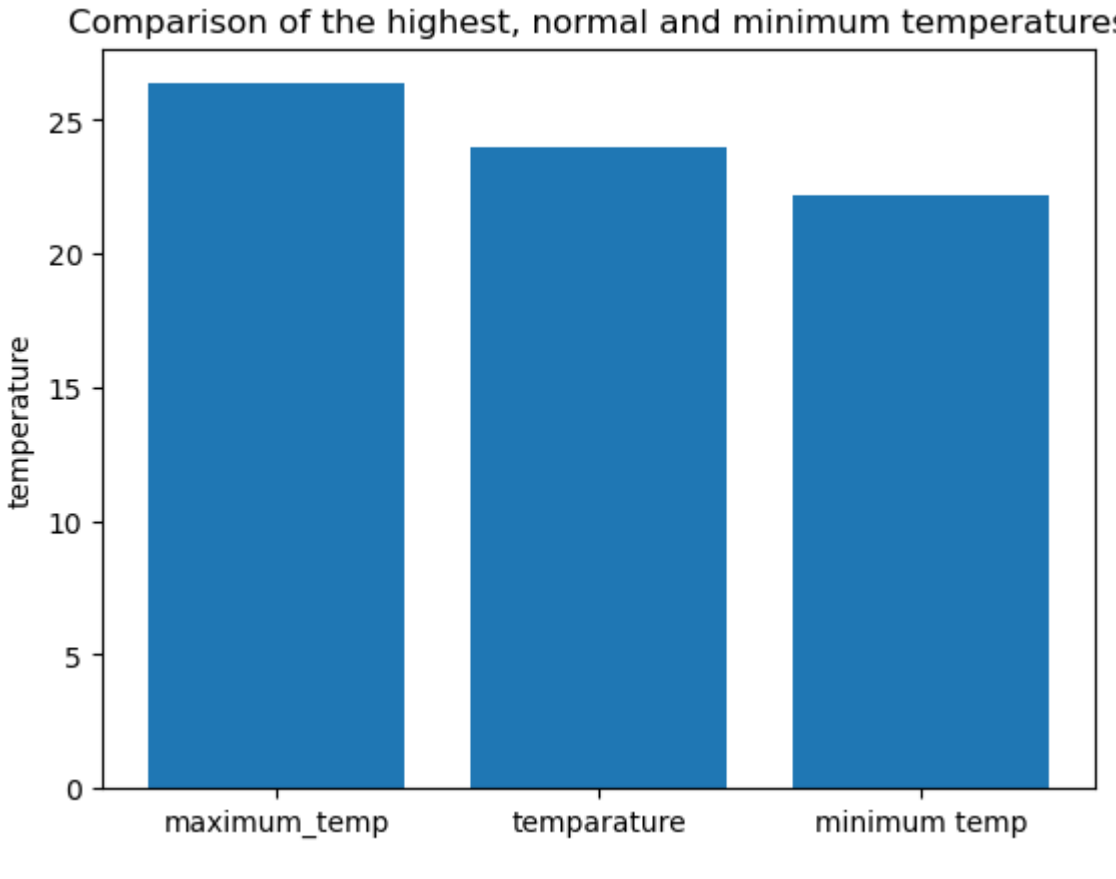
🚀 Next Steps & Strategic Roadmap

🔎 Step 1: Exploratory Data Analysis (EDA)

Visualize distributions, relationships, and correlations.

Identify missing values, feature importance, and outliers.


Target Variable Distribution: Rainfall
Count
0
200
400
600
800
1000
1200
1400
1600
0
1
Rainfall (0 = No Rain, 1 = Rain)


Comparison of the highest, normal and minimum temperatures
temperature
0
5
10
15
20
25
maximum_temp
temparature
minimum temp

Univariate Analysis

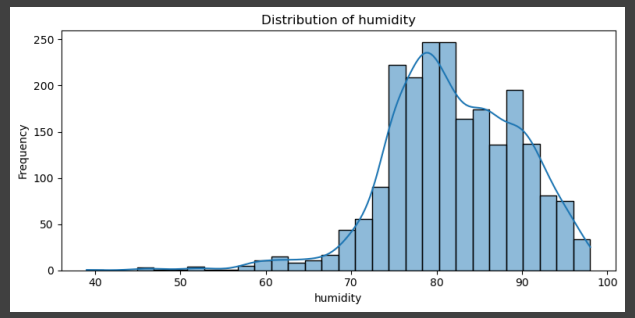
📌 What Happened?

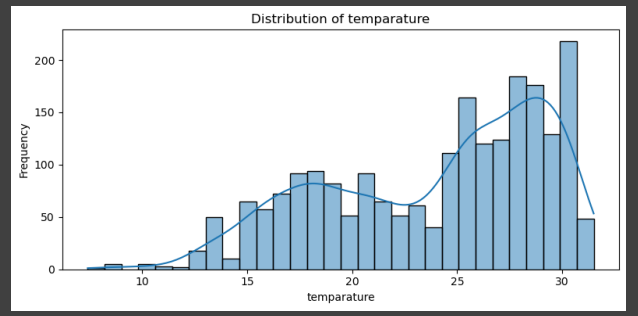
Explored the distribution of individual meteorological features using histograms and boxplots. Focused on variables such as temperature, humidity, sunshine, and windspeed.

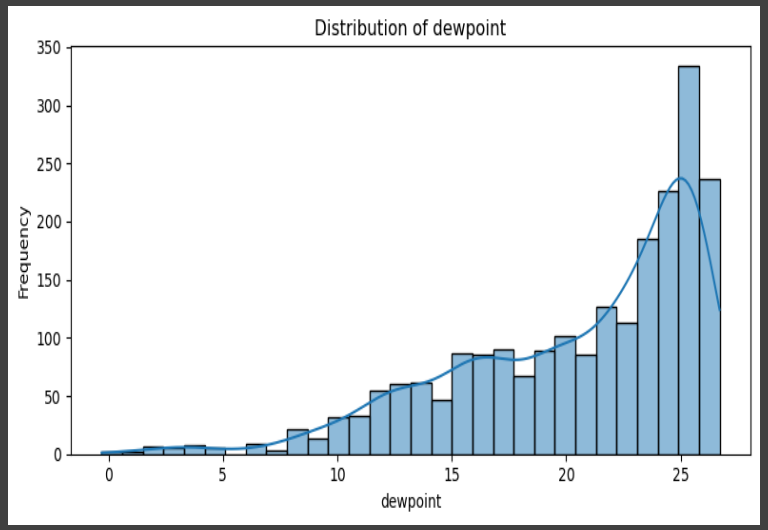
📊 Key Output:

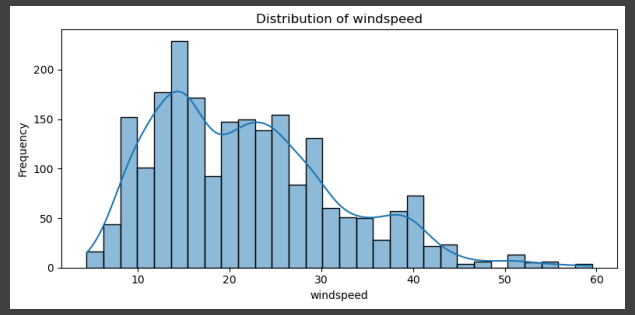
- Identified feature skewness, normality, and potential outliers.
- Boxplots highlighted outliers in windspeed and temperature.
- 💡 Why It Matters?

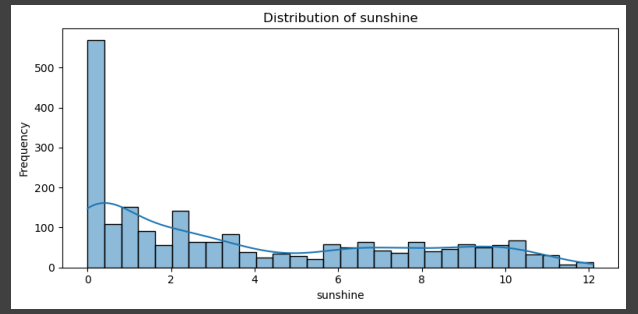
  Understanding single-variable distributions provides insight into data quality, variability, and shapes feature engineering choices.

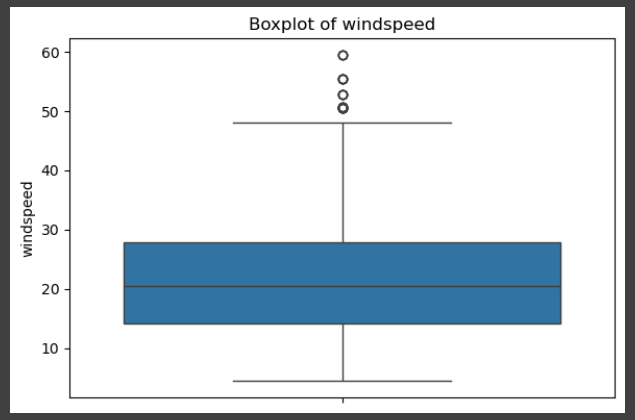

Distribution of humidity
Frequency
humidity

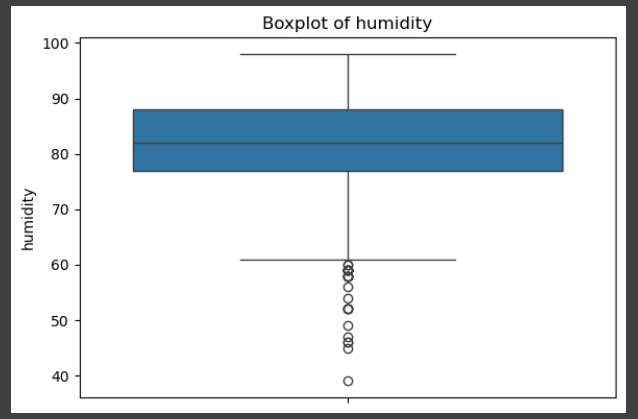

Distribution of temparature
Frequency
temparature

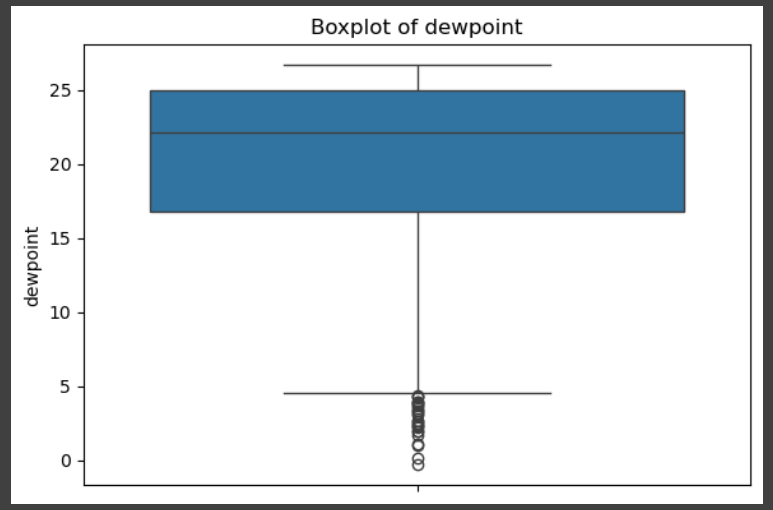

Distribution of dewpoint
Frequency
dewpoint

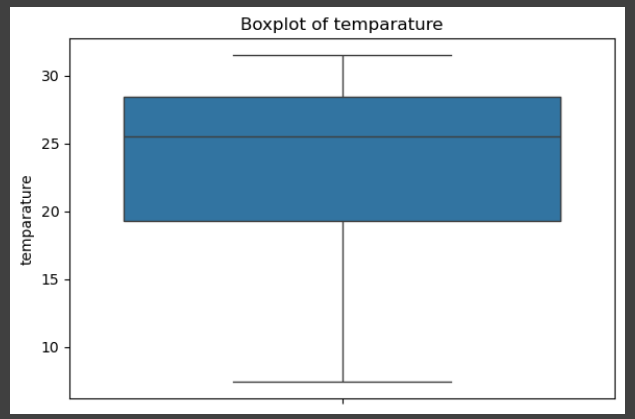

Distribution of windspeed
Frequency
windspeed

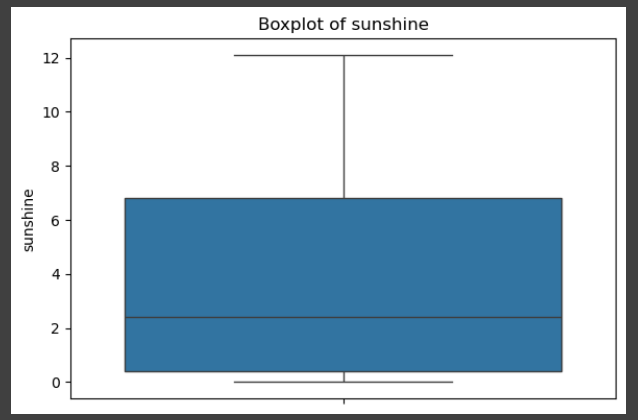

Distribution of sunshine
Frequency
sunshine


Boxplot of windspeed
windspeed
60
50
40
30
20
10


Boxplot of humidity
humidity
100
90
80
70
60
50
40


Boxplot of dewpoint
dewpoint
25
20
15
10
5
0


Boxplot of temparature
temparature
30
25
20
15
10

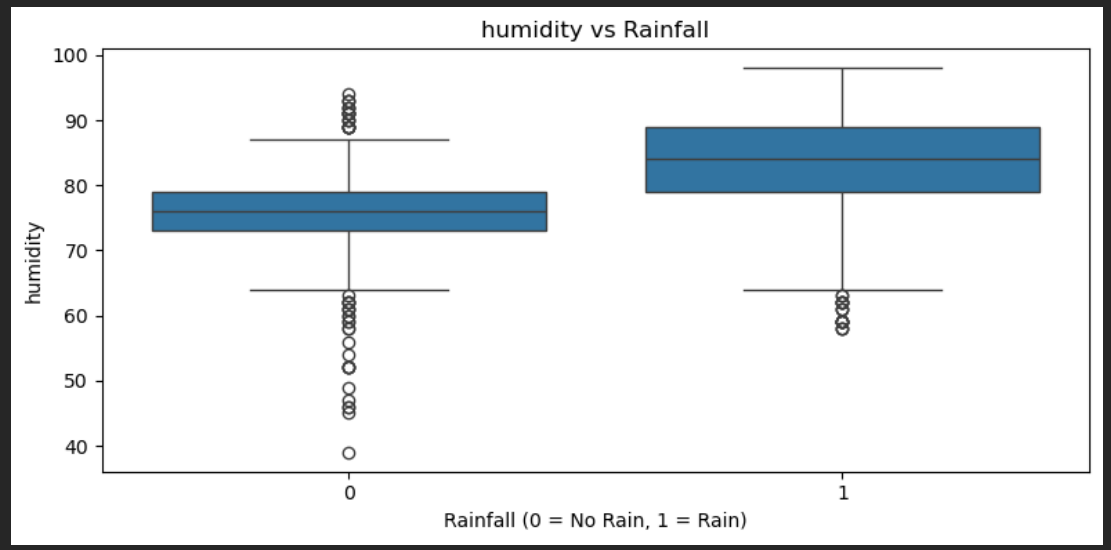

Boxplot of sunshine
sunshine
12
10
8
6
4
2
0

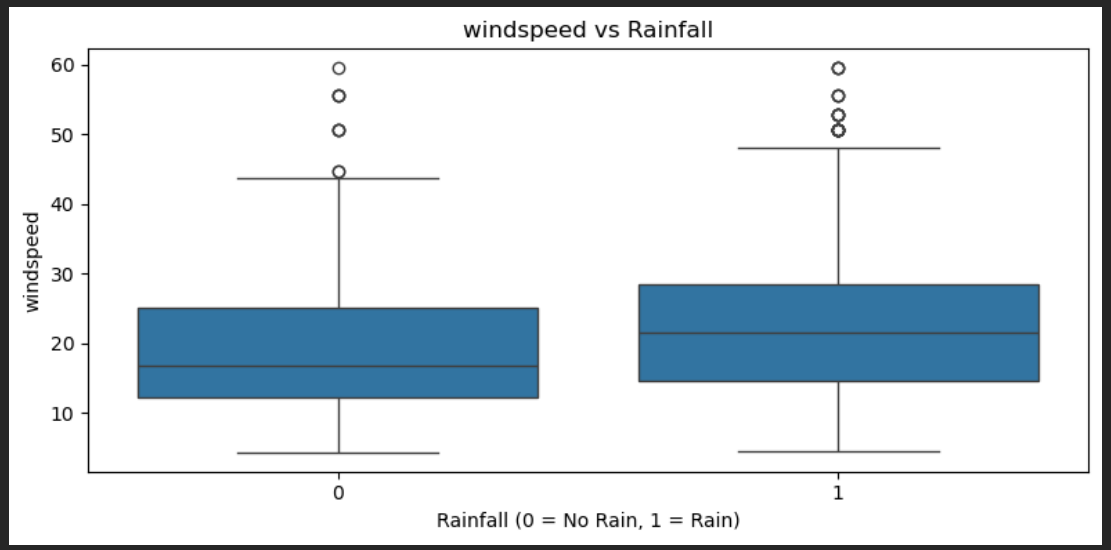
Bivariate Analysis

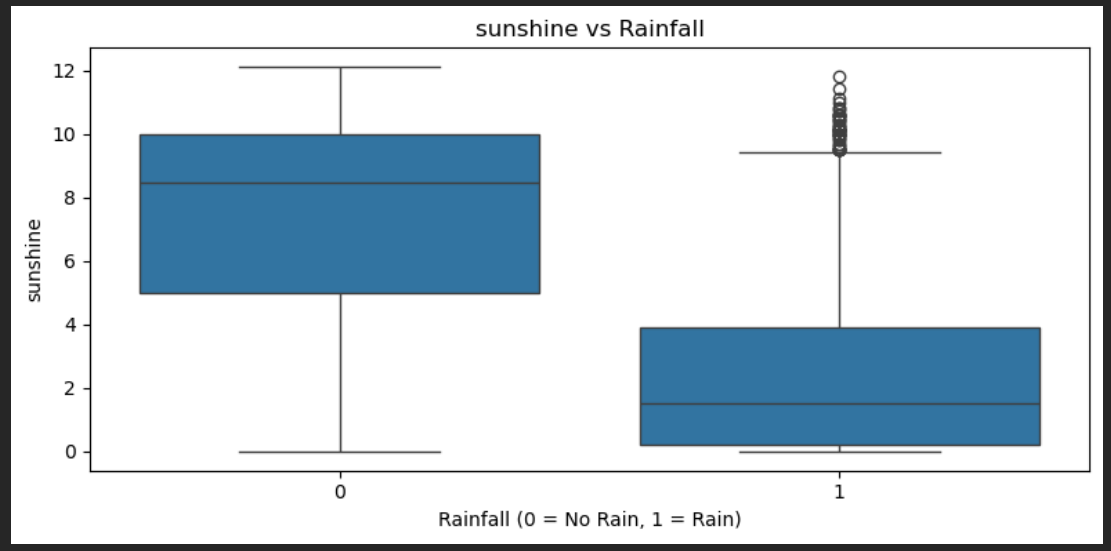
- 📌 What Happened?
  Analyzed relationships between predictor variables and the target variable (*rainfall*)

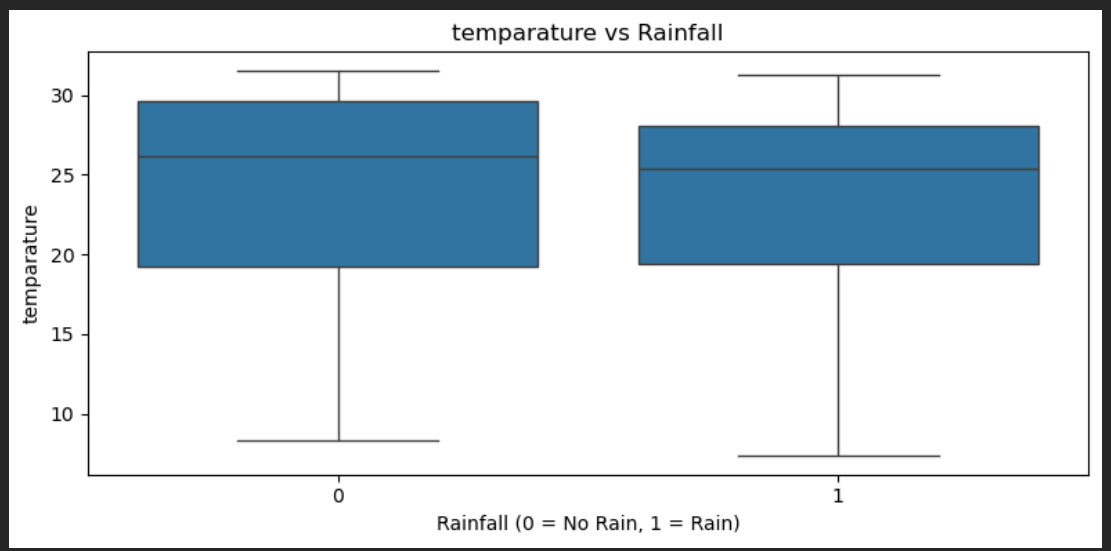
📊 Key Output:

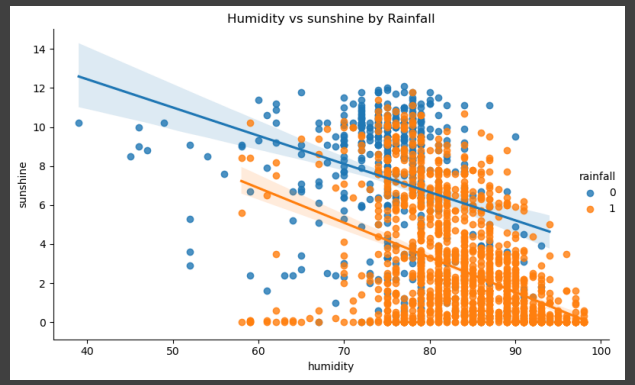
- Humidity, cloud cover, and sunshine showed distinct patterns for rainy vs. non-rainy days.
- Windspeed and temperature varied subtly with rainfall presence.
- 💡 Why It Matters?
  Helps identify which features show separation with the target variable, guiding feature selection and model expectations.

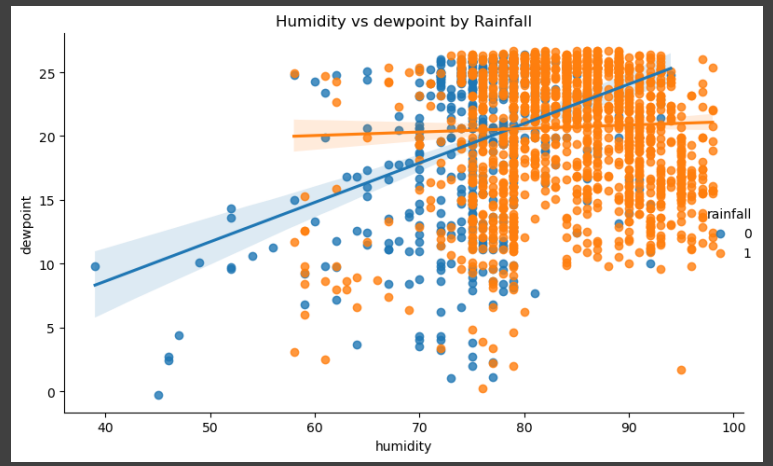

humidity vs Rainfall
humidity
Rainfall (0 = No Rain, 1 = Rain)

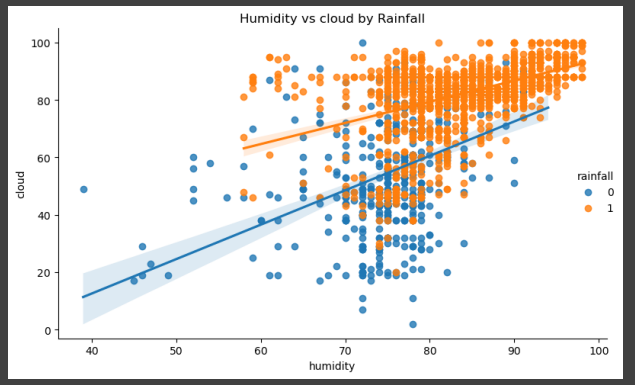

windspeed vs Rainfall
windspeed
Rainfall (0 = No Rain, 1 = Rain)

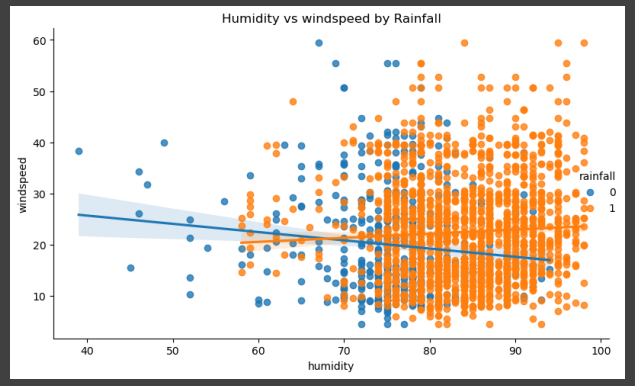

sunshine vs Rainfall
sunshine
Rainfall (0 = No Rain, 1 = Rain)

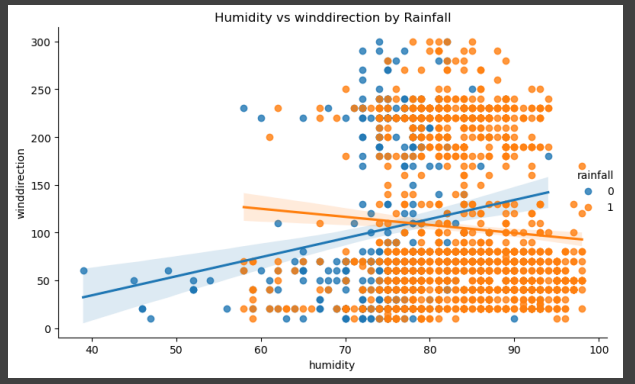

temparature vs Rainfall
temparature
Rainfall (0 = No Rain, 1 = Rain)


Humidity vs sunshine by Rainfall
sunshine
humidity
rainfall
0
1


Humidity vs dewpoint by Rainfall
dewpoint
humidity
rainfall
0
1


Humidity vs cloud by Rainfall
cloud
humidity
rainfall
0
1


Humidity vs windspeed by Rainfall
windspeed
humidity
rainfall
0
1


Humidity vs winddirection by Rainfall
winddirection
humidity
rainfall
0
1

Multivariate Analysis

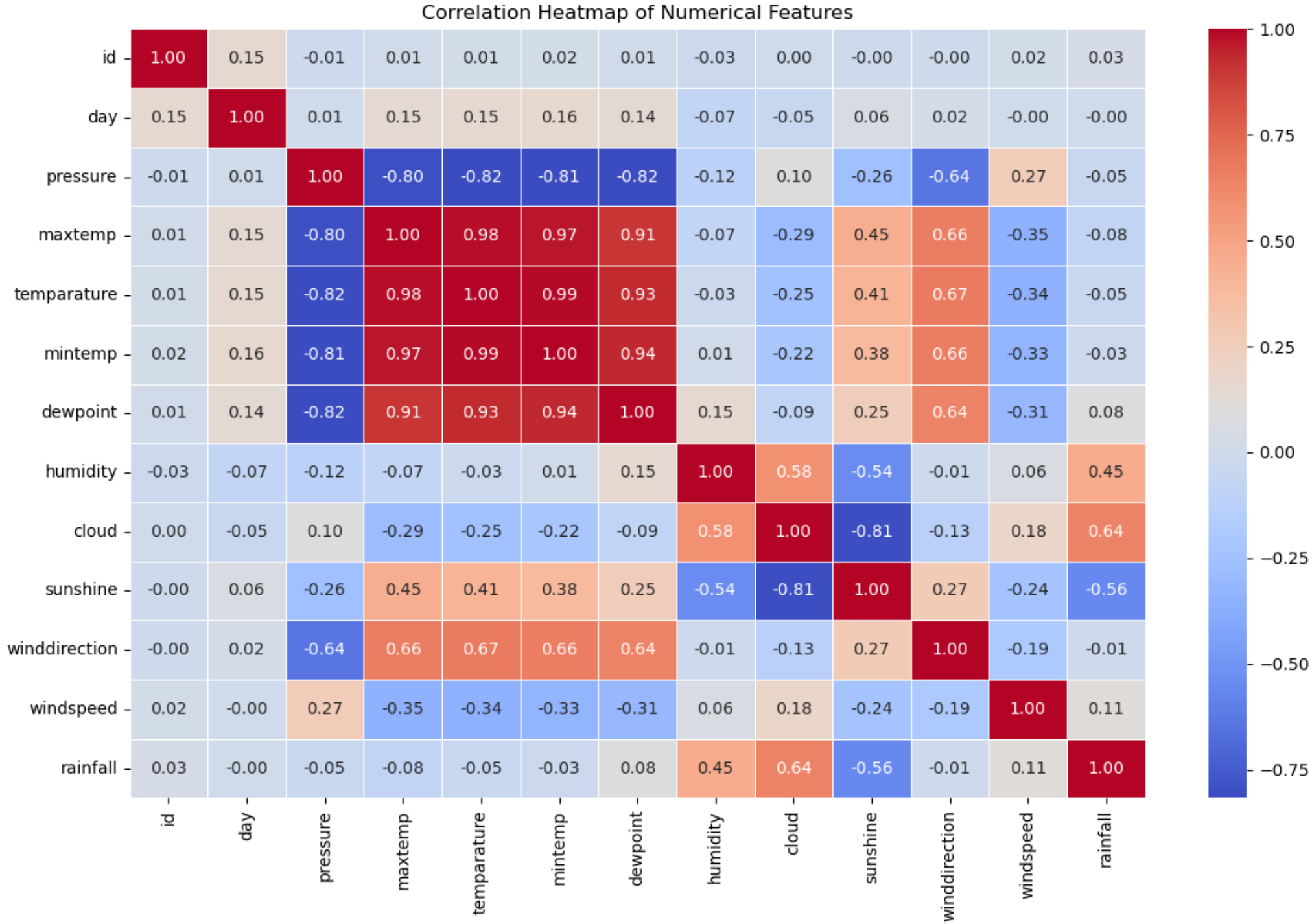
📌 What Happened?

Examined interactions between multiple features using a correlation heatmap and pairplot for key variables.

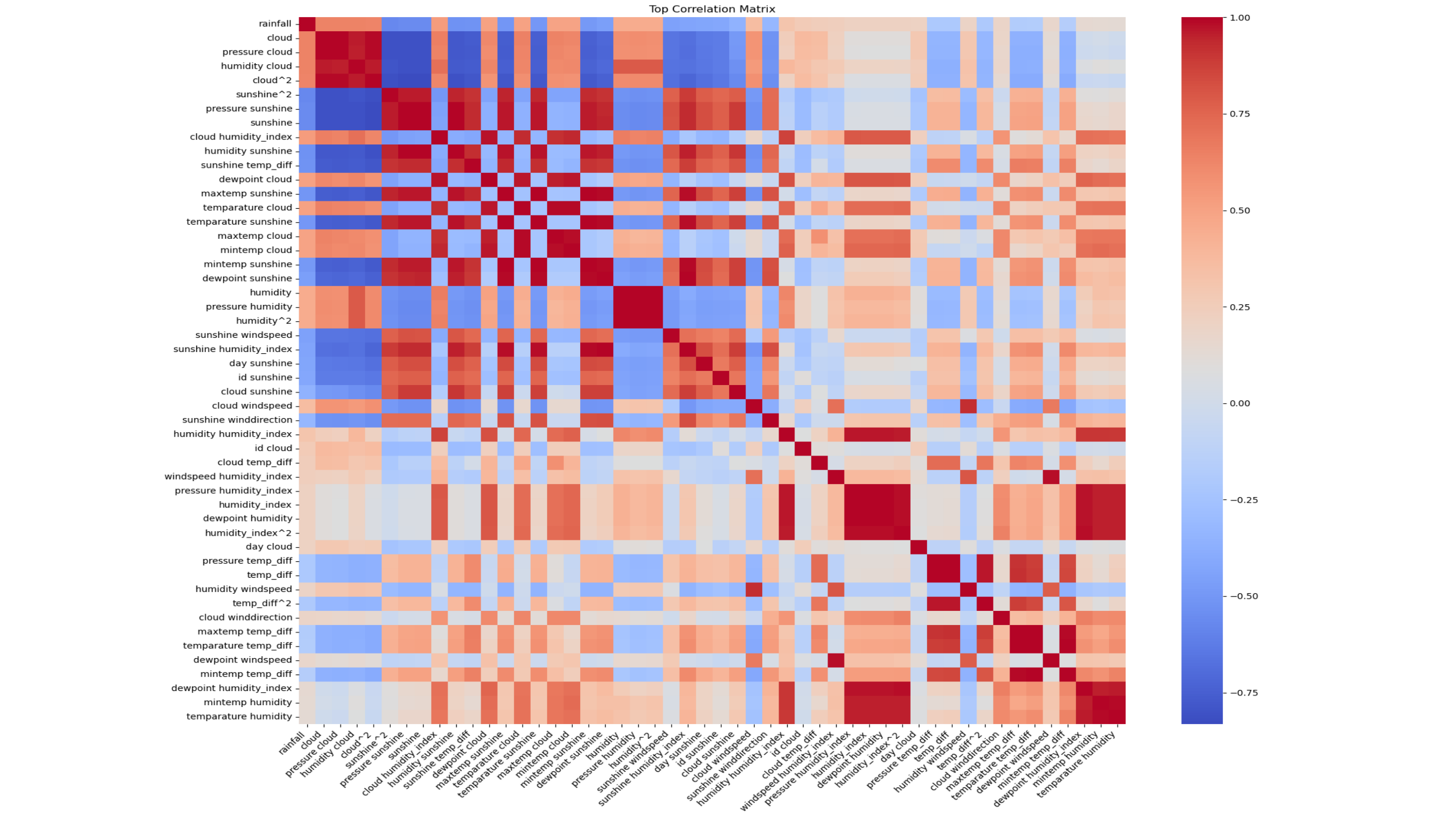
📊 Key Output:

- Identified highly correlated variables (e.g., maxtemp and temperature).
- Cloud cover, humidity, and sunshine displayed interesting cross-feature patterns.
- 💡 Why It Matters?

  Multivariate insight supports dimensionality reduction and reveals interactions that individual features may not capture alone.


Correlation Heatmap of Numerical Features
id 1.00 0.15 -0.01 0.01 0.01 0.02 0.01 -0.03 0.00 -0.00 -0.00 0.02 0.03
day 0.15 1.00 0.01 0.15 0.15 0.16 0.14 -0.07 -0.05 0.06 0.02 -0.00 -0.00
pressure -0.01 0.01 1.00 -0.80 -0.82 -0.81 -0.82 -0.12 0.10 -0.26 -0.64 0.27 -0.05
maxtemp 0.01 0.15 -0.80 1.00 0.98 0.97 0.91 -0.07 -0.29 0.45 0.66 -0.35 -0.08
temparature 0.01 0.15 -0.82 0.98 1.00 0.99 0.93 -0.03 -0.25 0.41 0.67 -0.34 -0.05
mintemp 0.02 0.16 -0.81 0.97 0.99 1.00 0.94 0.01 -0.22 0.38 0.66 -0.33 -0.03
dewpoint 0.01 0.14 -0.82 0.91 0.93 0.94 1.00 0.15 -0.09 0.25 0.64 -0.31 0.08
humidity -0.03 -0.07 -0.12 -0.07 -0.03 0.01 0.15 1.00 0.58 -0.54 -0.01 0.06 0.45
cloud 0.00 -0.05 0.10 -0.29 -0.25 -0.22 -0.09 0.58 1.00 -0.81 -0.13 0.18 0.64
sunshine -0.00 0.06 -0.26 0.45 0.41 0.38 0.25 -0.54 -0.81 1.00 0.27 -0.24 -0.56
winddirection -0.00 0.02 -0.64 0.66 0.67 0.66 0.64 -0.01 -0.13 0.27 1.00 -0.19 -0.01
windspeed 0.02 -0.00 0.27 -0.35 -0.34 -0.33 -0.31 0.06 0.18 -0.24 -0.19 1.00 0.11
rainfall 0.03 -0.00 -0.05 -0.08 -0.05 -0.03 0.08 0.45 0.64 -0.56 -0.01 0.11 1.00
id
day
pressure
maxtemp
temparature
mintemp
dewpoint
humidity
cloud
sunshine
winddirection
windspeed
rainfall
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

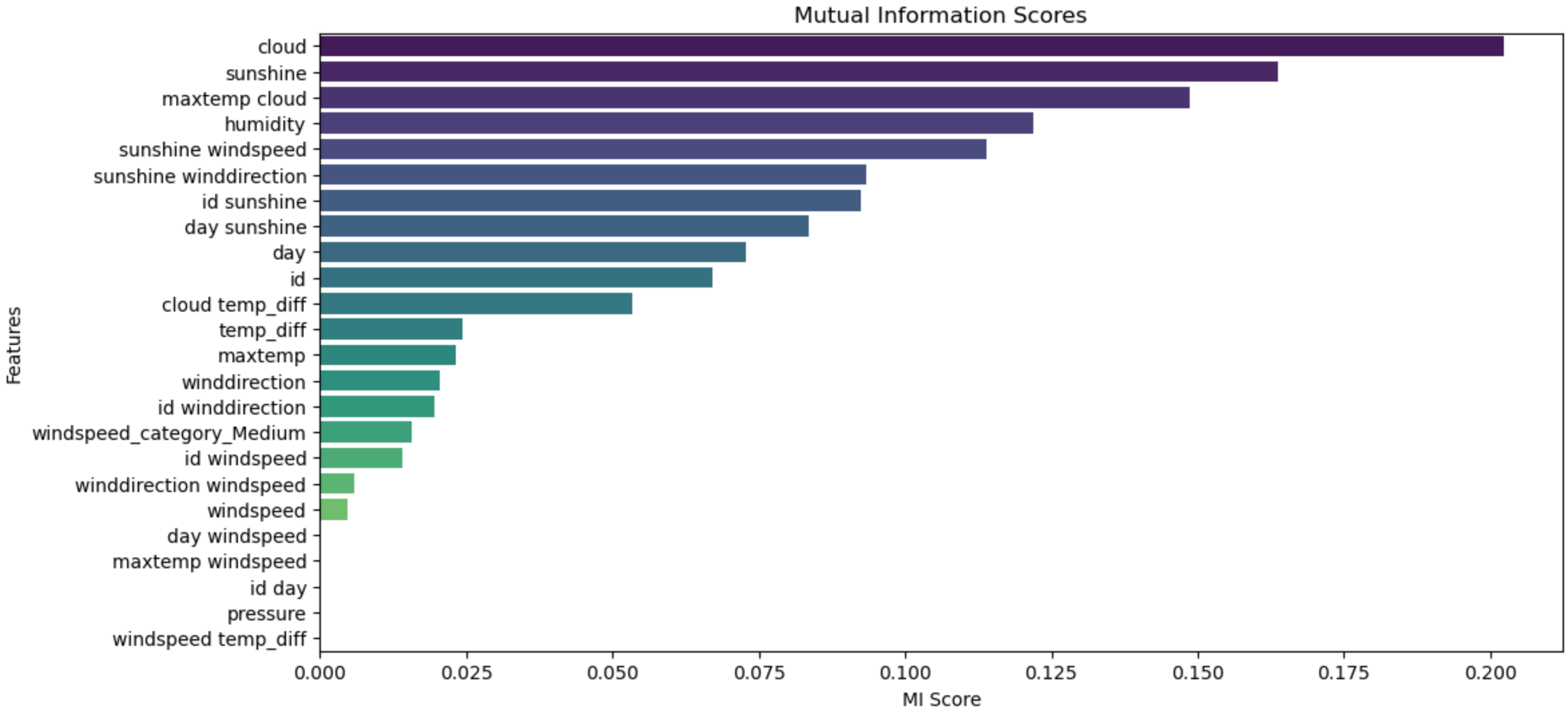

Pairwise Relationships by Rainfall
humidity
cloud
sunshine
temparature
windspeed
rainfall
0
1

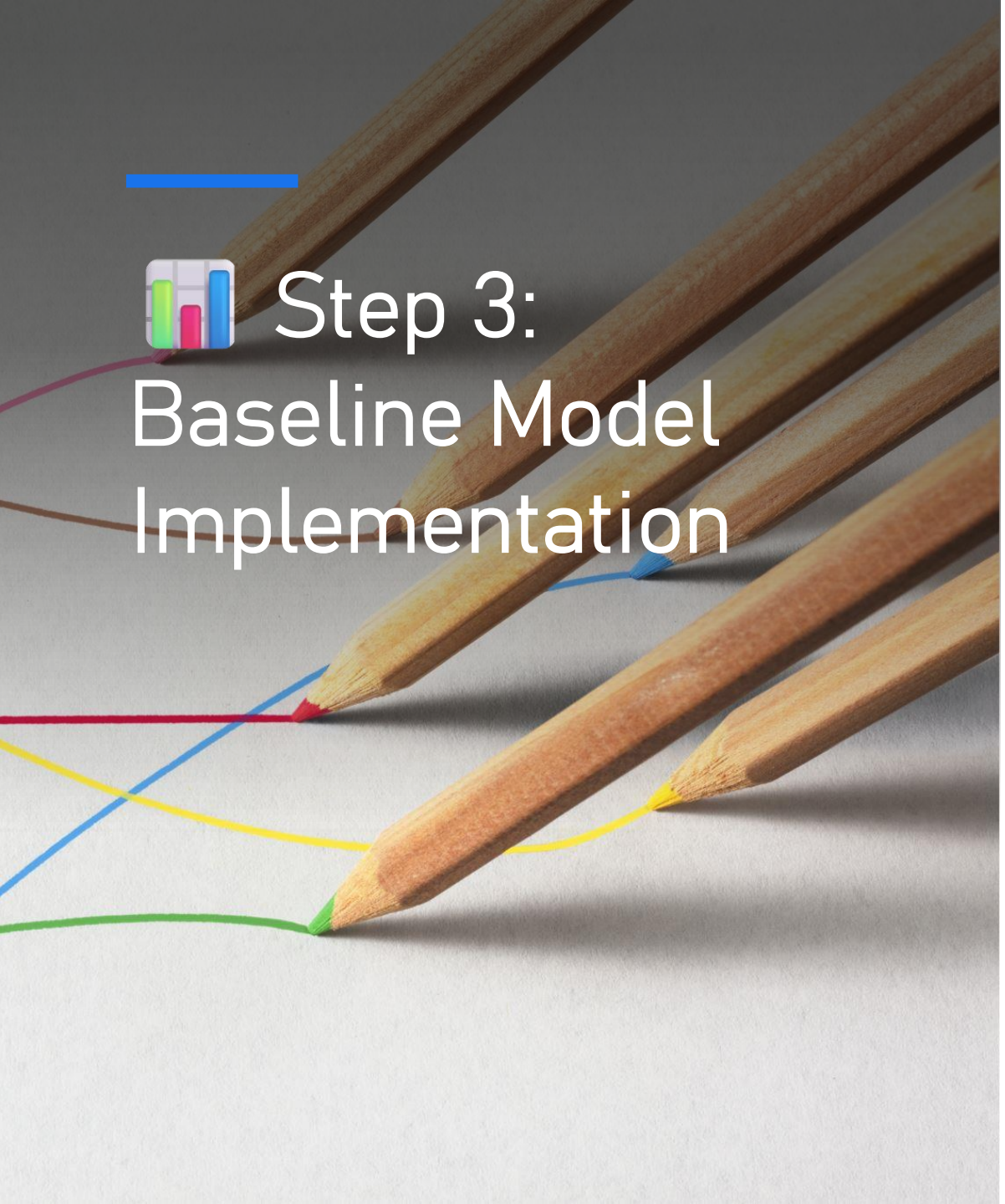

Top Correlation Matrix
rainfall
cloud
pressure cloud
humidity cloud
cloud^2
sunshine^2
pressure sunshine
sunshine
cloud humidity_index
humidity sunshine
sunshine temp_diff
dewpoint cloud
maxtemp sunshine
temparature cloud
temparature sunshine
maxtemp cloud
mintemp cloud
mintemp sunshine
dewpoint sunshine
humidity
pressure humidity
humidity^2
sunshine windspeed
sunshine humidity_index
day sunshine
id sunshine
cloud sunshine
cloud windspeed
sunshine winddirection
humidity humidity_index
id cloud
cloud temp_diff
windspeed humidity_index
pressure humidity_index
humidity_index
dewpoint humidity
humidity_index^2
day cloud
pressure temp_diff
temp_diff
humidity windspeed
temp_diff^2
cloud winddirection
maxtemp temp_diff
temparature temp_diff
dewpoint windspeed
mintemp temp_diff
dewpoint humidity_index
mintemp humidity
temparature humidity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

Step 2: Feature Engineering & Data Preprocessing

Create derived features (e.g., **humidity-temperature index, pressure deltas**).

Normalize & scale features for improved model performance.


Mutual Information Scores
cloud
sunshine
maxtemp cloud
humidity
sunshine windspeed
sunshine winddirection
id sunshine
day sunshine
day
id
cloud temp_diff
temp_diff
maxtemp
winddirection
id winddirection
windspeed_category_Medium
id windspeed
winddirection windspeed
windspeed
day windspeed
maxtemp windspeed
id day
pressure
windspeed temp_diff
Features
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
0.200
MI Score



📊 Step 3: Baseline Model Implementation

- Train **Logistic Regression, Decision Trees, and Random Forest** as benchmarks.

RANDOM FOREST

- We will be using our random forest model as the baseline model

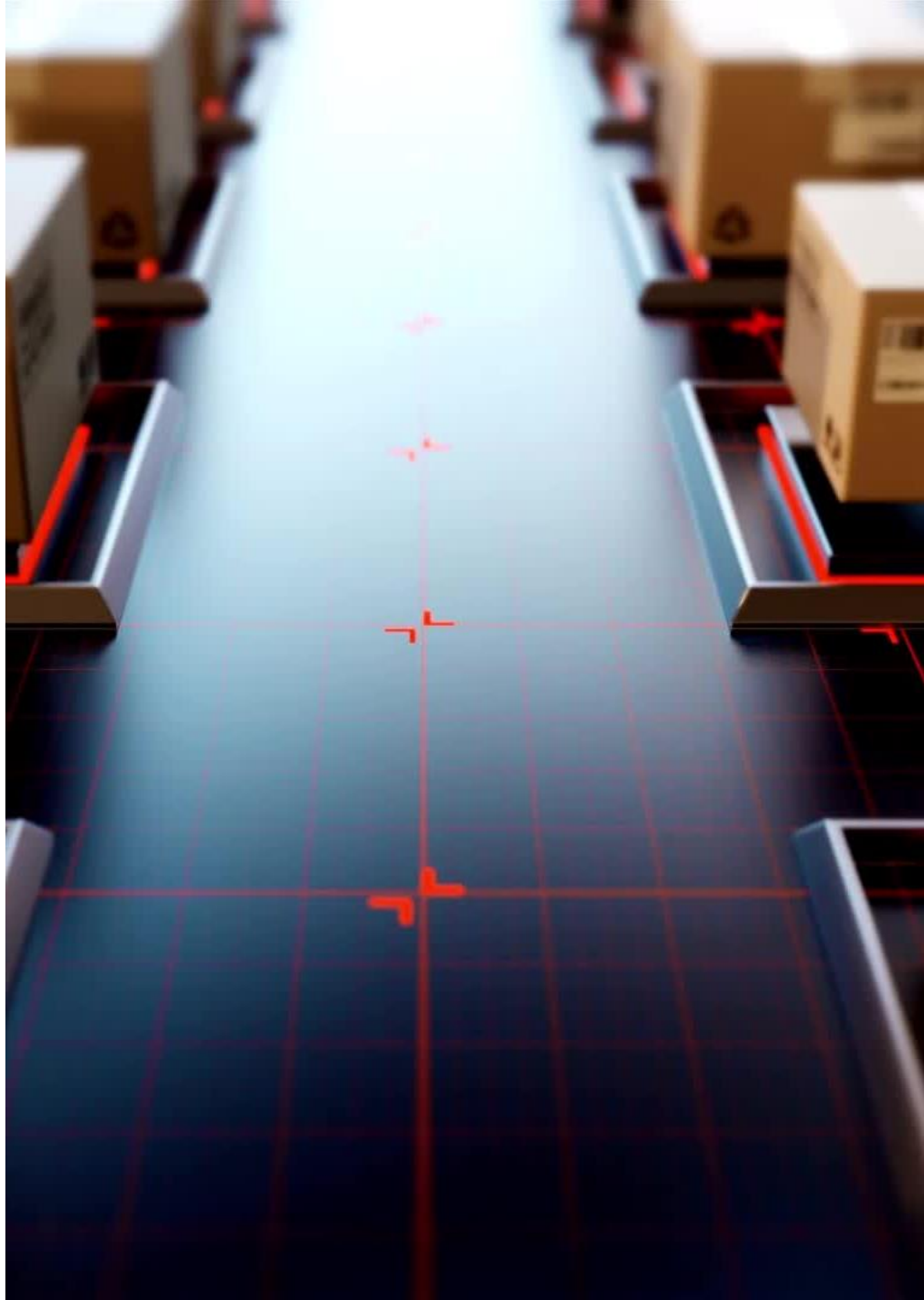
```
🔍 Evaluating Random Forest...
📊 Accuracy: 0.8607
📈 ROC AUC Score: 0.8583
📋 Classification Report:
              precision    recall  f1-score   support

           0       0.82      0.63      0.71       119
           1       0.87      0.95      0.91       319

    accuracy                           0.86       438
   macro avg       0.84      0.79      0.81       438
weighted avg       0.86      0.86      0.85       438
```

1.**Accuracy**: The model achieved an accuracy of **86.07%**, meaning it correctly predicted about 86% of instances.
2.**ROC AUC Score**: A score of **0.8583** indicates good performance in distinguishing between the two classes.
4.**Overall Metrics**:
- Macro Average: Precision **0.84**, Recall **0.79**, F1-Score **0.81**
- Weighted Average: Precision **0.86**, Recall **0.86**, F1-Score **0.85**

In summary, the model performs well, particularly on class 1, with strong precision and recall metrics, while class 0 shows lower recall.

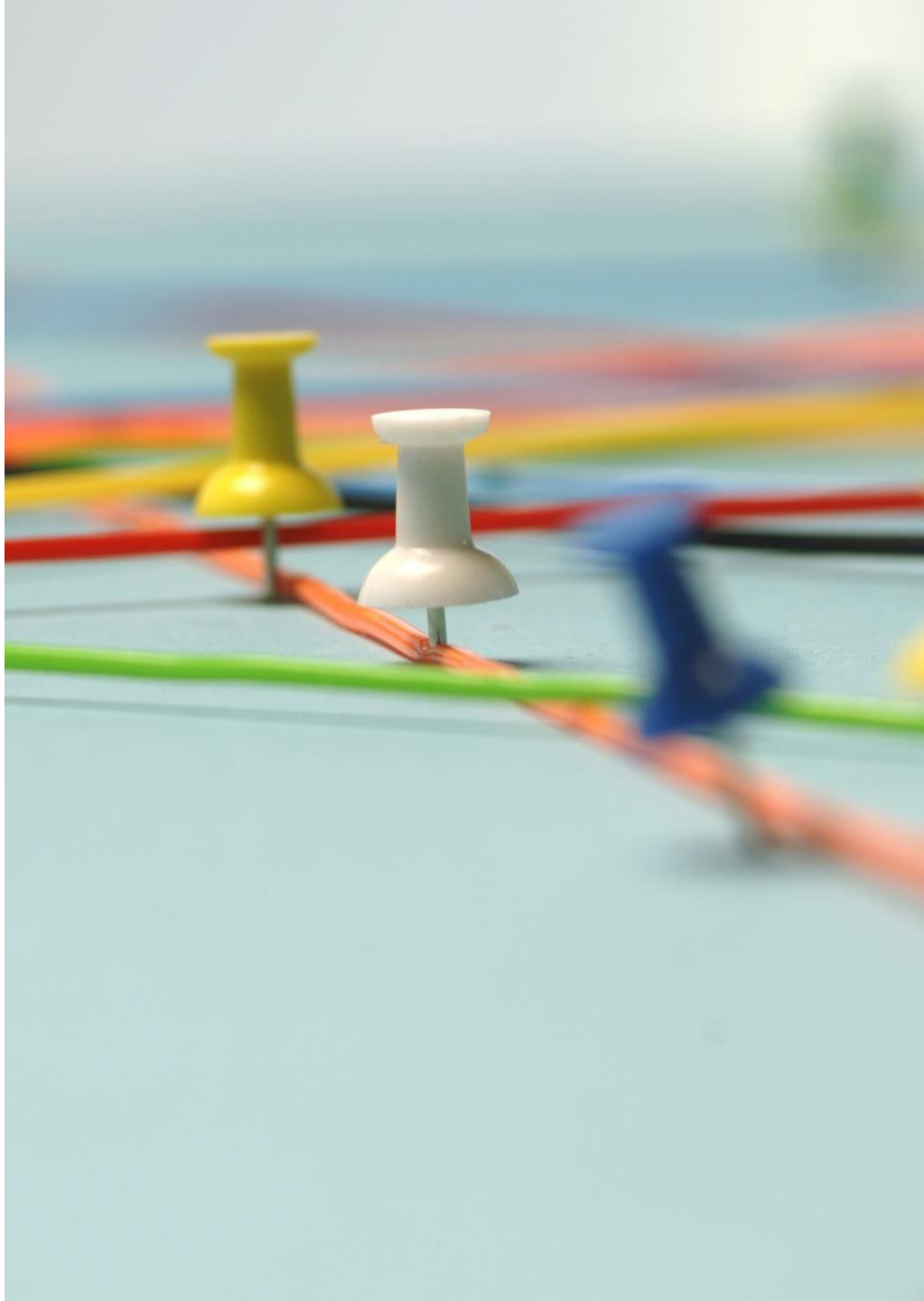
👩‍💻 Step 4: Advanced Model Development & Hyperparameter Tuning

- Implement **XGBoost and LightGBM**
- Optimize using **GridSearchCV and Bayesian Optimization**



📊 Step 5: Model Evaluation & Leaderboard Strategy

- Use **AUC-ROC, Precision-Recall, and Cross-Validation** to fine-tune accuracy.
- Deploy **Stacking, Blending, and Ensemble Learning** for leaderboard performance.



STACKING ENSEMBLE MODEL

- We will use this model as it is the best performing model

```
🔍 Evaluating Stacking Ensemble...
📊 Accuracy: 0.8516
📈 ROC AUC Score: 0.8623
📋 Classification Report:
              precision    recall  f1-score   support

           0       0.80      0.61      0.69       119
           1       0.86      0.94      0.90       319

    accuracy                           0.85       438
   macro avg       0.83      0.77      0.80       438
weighted avg       0.85      0.85      0.84       438
```

1. **Accuracy**: The model achieved an accuracy of **85.16%**, indicating it correctly predicted about 85% of the instances.
2. **ROC AUC Score**: A score of **0.8623** suggests the model has a good ability to distinguish between the two classes.
3. **Overall Metrics**:
    1. **Macro Average**: Precision **0.83**, Recall **0.77**, F1-Score **0.80**
    2. **Weighted Average**: Precision **0.85**, Recall **0.85**, F1-Score **0.84**

- In summary, the Stacking Ensemble model performs well, particularly on class 1, with high precision and recall, while class 0 shows lower recall. Overall, the model demonstrates solid performance metrics.

📝 Step 6: Reproducibility & Documentation

- **Environment Setup**: Create *requirements.txt* for dependencies.
- **Code Modularity**: Structure notebooks for clarity.
- **README Optimization**: Clearly document project workflow.
- **GitHub Repository Compliance**: Ensure README includes **elevator pitch, dataset details, implementation steps, and model performance**.

🎮 Step 7: Final Submission & Academic Presentation

Optimize final model selection and prepare Kaggle submissions.

Document findings in **Jupyter notebooks** & **GitHub README** for industry-grade presentation.

Prepare for **capstone defense** with clear justifications for model choices.

🏆 Conclusion: The Road to Kaggle & Academic Excellence

- This project represents a **cutting-edge application of AI in meteorology**, bridging academia and industry by showcasing **practical, high-impact machine learning workflows**. Through rigorous **data exploration, feature engineering, model optimization, and leaderboard analysis**, we aim to achieve a **Top 10 Kaggle ranking** while contributing **meaningful insights to real-world weather forecasting applications**.

Thank you!