**London Weather Forecast Report**

**Module: Programming for machine learning**

**Cohort: BCNS/24A/FT1**

**KISSOON Dhruv (2403\_27574),**

**HONG-LIN Severine (2403\_27569)**

**GHOORUN Vriddhi (2403\_27564),**

**HARRAH Bhevisha (2403\_27568)**

**GOPEE Shahista (2403\_27566),**

**KHODABOCUS Rehaan (2403\_27573)**

Abstract

# Introduction

Weather forecasting has traditionally been done by physical models of the atmosphere, which are unstable to perturbations, and thus leading to an inaccuracy in a long time run. Machine learning techniques are more robust to perturbations and can therefore make better predictions on weather conditions.

In this report, a machine learning approach will be implemented to showcase how the company, Climate Insights Ltd has used the machine learning algorithm based on pre-stored dataset to make weather – related predictions by analysing and providing a better insight for local businesses in London, especially those who are dependent on weather conditions. How the machine learning algorithm will identify the missing features and analyse the dimensionality of the dataset provided, through the exploratory data analysis to preprocess the data.

# Analyzing datasets and identifying any missing features or additional ones

**Existing weather features:**

Light rain, heavy rain, snow, wind, precipitations, fog

**Additional Features**

1. **Visibility**

* Visibility can be predicted from foggy weather conditions. For example, when fog level is too high, the visibility will be at its highest. From the fog column, visibility can be useful in explaining atmospheric conditions and how fog, visibility, humidity and precipitation are correlated and may lead to rain.

1. **Humidity**

* Humidity can help explain certain weather phenomena. For example, high humidity levels are often accompanied precipitation. Even though the dataset has a precipitation column, humidity could provide more insight into the atmospheric conditions leading to rain or fog. Humidity, temperature, and wind are all interrelated. For instance, the combination of high humidity and high temperature can make the weather feel hotter due to the heat index, and variations in humidity levels can influence how temperature changes are experienced.

1. **Wind direction**

* From the wind column dataset, wind direction may be predicted to have a better insight into how wind can have an impact on the direction of rain.

**For DATE Column**

* Extract days, months and years
* Extract by day-of-week
* Deduce SEASON\*

**Calculations/Predictions**

* ***Count occurrences of different weather conditions => see which type is more common***
* ***Compare weather across different months => calculate mean***
* Average daily wind
* To verify the datatypes used for manipulations
* ***Graph plot to analyze the data for any gaps or superfluous data***

# Determination of stability by analysing the dataset’s dimensionality

The dataset is a low-dimensional dataset as the dataset has only 6 columns of data. It is easier to interpret relationships between a small set of variables. With fewer features, there will be a lower risk of overfitting, especially if the sample size is large relative to the number of features, leading to the creation of simpler models. The current data set available captures only the basic weather information. However, if the study requires a deeper analysis such as predicting subtle weather patterns or capturing other climatic effects, additional variables like humidity or wind direction might be required as mentioned above. The date column can be broken down to create extra temporal features (day, month and year). While this increases dimensionality, it can be beneficial if any additional analysis is important for the study.

If the company focuses only on general weather trends or simple analysis, the six columns’ data set may be entirely adequate, especially when temporal features from the date have been extracted, thus making it straightforward for exploration analysis and initial modeling. Its suitability depends entirely on the study’s goals. For basic weather analysis, the columns are sufficient. However, if the study demands a more detailed understanding of other weather conditions, additional relevant variables may be required to have a better interpretation of the data set.

Chosen Machine Learning Model: Classification

The goal is to determine whether a Classification or Regression Model is more suitable for predicting weather conditions based on the given dataset: London\_Weather.csv.

Why is Regression not used?

Regression is a kind of predictive modeling where the variable to be estimated is of continuous nature. This helps to predict a range of values based on previously taken measurement. For example: price or temperature, their values never stay the same, they continue to change, hence the term “continuous”.

The output type for Regression is Continuous, that is; numbers that may be decimal or integer.

Why we could have chosen Regression ?

As Regression is a type of Supervised Learning that deals with predicting continuous values or numeric outcomes, it would have been enough to use it for the dataset that was given as it contains numeric continuous values such as Maximum Temperature, Minimum Temperature, Wind, and Precipitation with are all continuous values that changes over time.

If we used Regression and we have the output, it would look like:

* Predicting Temperature
  + Output will be Numerical: 23.5°C, instead of “Sun”
* Predicting Precipitation Amount
  + Output will be Decimal: 8.2mm instead of “Heavy Rain”
* Predicting Wind Speed:
  + Output will be Numerical: 15.7 km/h.

Business would want to know the Weather instead of the actual temperature.

Example: It will be Sunny it better instead of saying its 23.5°.

Classification

Classification is a type of Supervised Learning where the model learns in an environment with labelled data. Classification is used for predicting a discrete output, for example in this case: Sunny, Rainy, Snow.

Classification predictive modeling is the task of approximating a mapping function, f from input variables, X to discrete output variables, y.

The output variables are often called labels or categories, The mapping function predicts the class or category for a given observation.

Classification models are used when the outcome variable is categorical or binary, meaning it has a limited number of options.

Types of Classification:

* Binary Classification which can have only two possible classes; Example: Spam or Not Spam.
* Multi-class Classification where more than two possible classes. For Example: Identifying a picture which can be cats, dogs, or birds. In our case, it can be Sunny, Rainy, Snow, Heavy Rain or different outputs.

Why have we chosen Classification.

As the output target variable is categorical (Sunny, Rain, Snow, etc....), having discrete classes, it has to classify instances into pre-determined classes.

The data set used in this study comes from “London\_Weather.csv” which was provided. The data in the resolution of one day spans the period from 1st January 2020 to 31st December 2023. A total of 1463 attributes are present in the raw dataset.

With Classification, we have Evaluation Metrics.

* Accuracy.
* Precision.
* Recall.
* F1-Score.
* MSE.
* MAE (not sure need to recheck).

1. Algorithm Suitability

* With Multi-Class Classification, the dataset’s target has multiple classes making algorithms like Decision Trees, Random Forest ideal for capturing non-linear relationships between features, (e.g., Temperature, Wind) and weather types.

1. Limitations of Regression

* Regression Models, e.g., Linear Regression, predicts numerical values, which are irrelevant here since the target is categorical.
* Converting the categorical “weather” variable into numerical labels, e.g., “0” for “Sun” and “1” for “Wind”, would force regression models to treat these classes as **ordered values**, leading to inaccurate interpretations.
  + Ordered Values here mean that regression models would inherently assume that such relationship exists between labels, that is the values are not identifiers but carry a mathematical relationship. Example: 0<1<2. This can lead to meaningless interpretations if the categories are not inherently ordered.
  + If we had:
    - Sun = 0
    - Rain = 1
    - Snow = 2
    - If we had a prediction of 1.5, then it would imply something meaningless like “Half Rain Half Snow” which is not ideal.

Choosing the Best Classification Model for Weather Prediction

When selecting a machine learning model for weather classification, it’s essential to consider accuracy, interpretability, computational cost, and how well the model handles complex patterns. Below, we analyze three popular classification models: Random Forest, Gradient Boosting Machines (GBM), and Support Vector Machines (SVM) that are best suited for our dataset.

1. Random Forest Classifier

Random Forest is an ensemble learning method that creates multiple decision trees during training. Each tree votes on the final classification, and the majority vote determines the prediction.

Why Use It for Weather Prediction?

* It handles missing data well.
* Works effectively with both numerical and categorical weather features.
* Reduces overfitting because it combines multiple trees.
* Provides feature importance, helping us understand which weather factors impact predictions the most.

Limitations:

* Slower than simpler models because it builds multiple trees.
* Less interpretable than a single decision tree.

1. Gradient Boosting Machines (GBM)

GBM is also an ensemble learning method, but instead of training multiple trees independently (like Random Forest), it builds trees sequentially, each correcting the errors of the previous tree. This leads to better accuracy but requires more tuning.

Why Use It for Weather Prediction?

* Often more accurate than Random Forest due to sequential learning.
* Can detect complex patterns in the data, making it effective for weather classification.
* Reduces both bias and variance, leading to better generalization.

Limitations:

* Computationally expensive training takes longer, especially with large datasets.
* Requires hyperparameter tuning (e.g., learning rate, number of trees).
* More sensitive to outliers than Random Forest.

1. Support Vector Machines (SVM)

SVM is a supervised learning algorithm that finds the optimal decision boundary (or hyperplane) that best separates different weather categories. It works well when there’s a clear separation between classes.

Why Use It for Weather Prediction?

* Works well for small datasets where clear decision boundaries exist.
* Effective when there are a few irrelevant features in the dataset.
* Can use different kernels (linear, polynomial, RBF) to fit different types of data.

Limitations:

* Slow for large datasets (ours has ~1,462 rows).
* Requires careful feature scaling (normalization).
* Difficult to tune hyperparameters like kernel type, C, and gamma.

Which Model Is Best for This Weather Dataset?

Considering that our dataset has weather-related numerical data, we need a model that:

* Handles missing values well
* Works with mixed data types (temperature, precipitation, wind speed, etc.)
* Provides good accuracy and interpretability

Final Recommendation:

* If speed and interpretability matter → Use Random Forest.
* If accuracy is the priority and computational cost is not an issue → Use GBM.
* If dataset is small and decision boundaries are clear → Use SVM (but not ideal here).

The London Weather Analysis Assignment is designed to predict weather types based on pre-recorded weather data values. This assignment utilizes a machine learning model to classify the weather into its correct categories.

Features

* Input Parameters: The code uses a variety of weather-related data such as Date, Precipitation, Maximum Temperature, Minimum Temperature and Wind Speed.
* Prediction: Based on the input features, the machine learning model classifies the weather into its own category, for instance: Sunny, Heavy Rain, Snow, Light Rain, Fog.
* Data Exploration: The dataset includes outliers and a range of values.

Technologies Used

* Backend: Python.
* Data Handling: pandas, NumPy.
* Data Visualization: seaborn, matplotlib.

Dataset

This assignment is provided with the weather data set that will be fed into the machine learning algorithm. The dataset includes:

* date: The date of when the data was recorded.
* precipitation: Decimal values representing precipitation.
* temp\_max: Decimal values presenting the maximum temperature.
* temp\_min: Decimal values presenting the minimum temperature.
* wind: Decimal values representing wind speed.
* weather: Categorical target variable for classification.

Best Classification Models for this case.

1. Random Forest Implementation
   1. Handles non-linear relationships and feature interactions (e.g., how wind and temperature jointly affect snow likelihood)
2. K-Nearest Neighbors (KNN)
   1. Classifies based on the majority class of the nearest points.
3. Support Vector Machine Implementation
   1. Suitable for high-dimensional data after encoding categorical features.
4. Random Forest

Random Forest (RF) is a commonly utilized supervised learning algorithm used for classification as well as regression tasks. It is a collective learning technique that merges various decision trees to create a strong and precise model. A decision tree is a forecasting model that uses a tree-shaped diagram to outline different potential results and the choices that result them, see Figure 2. In this structure, every internal node represents a choice, whereas each leaf node indicates a predicted result.

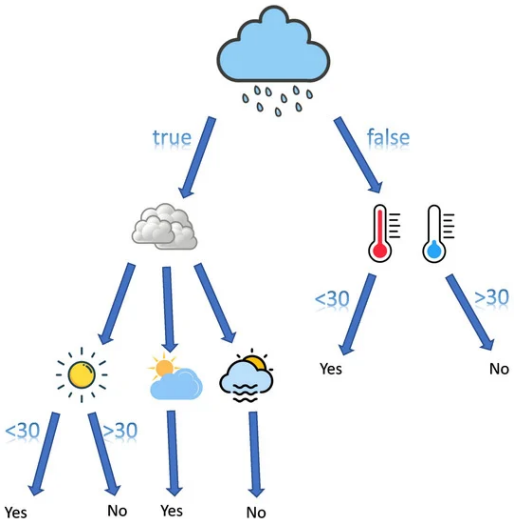


Figure 2

RF models provide multiple benefits, such as the ability to manage high-dimensional data, non-linear correlations, and absent values.

1. KNNs

KNN is a widely used supervised learning algorithm. It works by locating the k nearest training samples to a specific test sample and categorizing the test sample into the most frequent class withing its k nearest neighbors, see Figure 3.

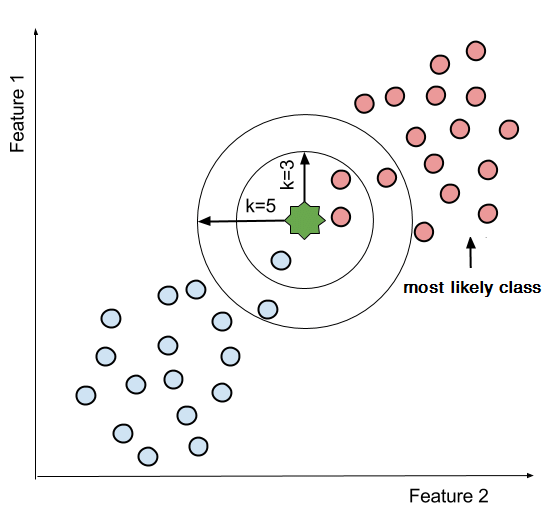


Figure 3

A typical example of a KNN classification for two class problems (e.g., Pink and Blue Circles) when the k parameter is set to 3 and 5. The green star represents a unseen sample point.

Nonetheless, KNN has certain drawbacks, such as being sensitive to outliers and facing challenges when mapping high-dimensional data.

1. SVMs

An SVM is a classification tool aimed at finding the best hyperplane that increases the margin between two categories of data points, as illustrated in Figure 1 below.

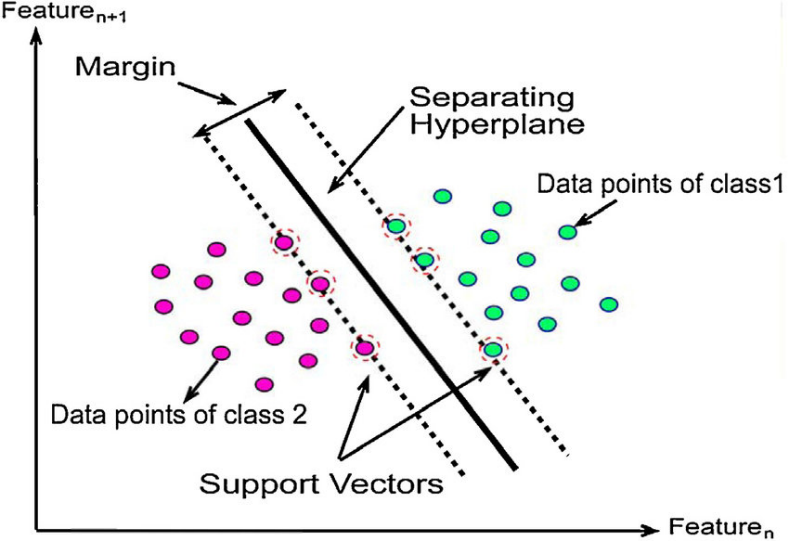


Figure 1

The margin refers to the gap between the hyperplane and the closest data point from both classes. SVMs aim to increase this margin while reducing classification mistakes. SVMs can be divided into linear and non-linear classifications.

Advantages & Disadvantages

Random Forest

Advantages:

* Robust and Accurate
* Handles Missing Data

Disadvantages:

* Less Interpretable
* Needs Tuning

KNN

Advantages:

* Simple and adaptable
* No training phase

Disadvantages:

* Slow on large datasets
* Sensitive to noise

SVMs

Advantages:

* Works well in high dimensions
* Avoid overfitting

Disadvantages:

* Needs careful tuning
* Slow on large data sets

Implementation of Classification Models

Step 1: Data Preprocessing

Before training any model, we must:

* Handle missing values (e.g., precipitation = 0)
* Split the data into training and testing sets.

Step 2: Train and Evaluate Each Model

* Split data into training and testing sets.
* Use cross-validation to optimize hyperparameters.

Step 3: Compare Model Performance

After running all models, check the accuracy, precision, recall, and F1-score to determine the best one.

If GBM has the highest accuracy, but training time is a concern, Random Forest may be the best option.

Performance Metrics

The effectiveness of RF, KNN, SVMs can be measured using three performance metrics:

* Precision
* Recall
* F1-Score

These metrics are utilized to assess the predictive performance of the models.

Precision, determined by Equation (1), is an important performance measure for assessing the correctness of a model’s positive prediction. It specifically assesses the ratio of accurately recognized positive cases among all cases that are labeled as positive. In this measure, True Positives indicates the total of data points accurately recognized as positive by the model, whereas False Positives refers to the count of cases incorrectly labeled as positive when they are negative.

|  |  |
| --- | --- |
|  | (1) |

Recall assesses the model’s ability to correctly recognize positive instances, and it is determined using Equation (2). It assesses the ratio of true positive cases identified out of all the genuine positive instances in the dataset. Recall primarily emphasizes the model’s responsiveness to positive cases and its efficiency in reducing false negatives.

In Equation (2), False Negatives signify the quantity of instances that are truly positive yet are mistakenly labeled as negative by the model.

|  |  |
| --- | --- |
|  | (2) |

F1 Score, determined by Equation 3, merges precision and recall. A rise in the F1 Score, which varies from 0 to 1, signifies enhanced overall performance. It takes into account both precision and recall to attain equilibrium between two metrics.

|  |  |
| --- | --- |
|  | (3) |

# Conclusion

# References

<https://www.ecmwf.int/sites/default/files/elibrary/092024/81582-evaluation-of-ecmwf-forecasts_0.pdf>

<https://www.climatechange.ai/blog/2024-02-07-forecast-tutorials>

Tahafaisalkhan. (n.d.). *Tahafaisalkhan/weather-classifier: Weather type classification is a machine learning project that predicts weather types-rainy, sunny, cloudy, or snowy-using a synthetically generated dataset. this project focuses on classification algorithms, data preprocessing, and outlier detection, making it ideal for honing data science and machine learning skills.*

GitHub. <https://github.com/tahafaisalkhan/weather-classifier>

Aldous, K. (2023, May 2). *Comparing the Results: Classification vs. Regression Models in Machine Learning*. Team Acua.

<https://acua.qcri.org/blog/comparing-the-results-classification-vs-regression-models-in-machine-learning/>

Safia, M., Abbas, R., & Aslani, M. (2023). *Classification of weather conditions based on supervised learning for Swedish cities. Atmosphere, 14(7), 1174.* <https://doi.org/10.3390/atmos14071174>

*GeeksforGeeks. (2025, January 29). Classification vs Regression in Machine Learning.* GeeksforGeeks. <https://www.geeksforgeeks.org/ml-classification-vs-regression/>

Perera, J. (2022, January 6). *Applying linear regression on a weather dataset - Analytics Vidhya - medium.*

*Medium*. <https://medium.com/analytics-vidhya/applying-linear-regression-on-a-weather-data-set-e84901120f88>

Ak, A. (2023, October 9). *Regression vs. Classification: Understanding the Differences and Use Cases.*

*Medium*. <https://medium.com/@abelkuriakose/regression-vs-classification-understanding-the-differences-and-use-cases-6cd51c236112>

Dattaradon, G. (2024, February 26). *Advancing Weather Prediction : Comprehensive Classification Models using Python.*

*Medium*. <https://medium.com/@gigi.dattaradon/weather-prediction-with-python-a-comprehensive-analysis-of-classification-models-f7149d39995d>

<https://www.springboard.com/blog/data-science/regression-vs-classification>

<https://shannwang.github.io/Machine_Learning/supervised_learning_classifying_weather_conditions.html>

<https://machinelearningmastery.com/classification-versus-regression-in-machine-learning/>

<https://www.datacamp.com/blog/classification-machine-learning>

<https://www.researchgate.net/figure/A-typical-example-of-a-KNN-classification-for-a-two-class-problem-ie-the-pink-and_fig2_322358139>