# Data Science Project

## Research Question

By analysing historical weather data from the past three years, this project aims to discover if it is possible to identify the optimal days for booking holiday leave based on favourable weather conditions for Manchester in April 2025.

## Executive Summary

Weather is notoriously difficult to forecast long term. The model needs improving and updating to fulfil the research question.

## Project Background

Human Resources require all holiday entitlement must be used before 01st May 2025. The aim of this project is to identify which days in April 2025 would have the fairest weather conditions for cycling. The method for identifying the fairest weather is to develop temperature and rainfall time series forecasts for April 2025. Using a combination of these forecasts to identify the optimal days based on favourable weather conditions for booking holiday leave to cycle.

## Data Collection

Historical weather data provided by https://www.visualcrossing.com/weather-data/ exported as a .csv file for the Manchester area.

### Variables

The data contains temperature in degrees Celsius for min, max, average and feelslike daily. I have selected feels like as the temperature variable as a cyclist will be exposed to the elements. Precipitation is in millimetres daily. To ascertain the optimal days the proposal is to engineer a variable that combines temperature and rainfall. Based on Exploratory Data Analysis EDA temperature and rain will be categorised and ranked best to worst. Rain will be given a heavier negative weighting than temperature as wet conditions less favourable than cooler temperatures.

## Tools used

Python is used for this project on the Google Colab environment. Python is a programming language with libraries of pre-written code. The pre-written code provide easy to use functionalities without coding everything from scratch.

## Exploratory Data Analysis EDA

Figure 1 shows dataframe is creation, Initial transformations are complete unwanted columns are removed and columns names updated. The index (date) and the frequency of the data (daily) are set to prepare the data for a time series analysis.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1 – Data preparation

An initial plot of the data in figure 2 showed missing data (March-June 2022). Rather than infer the values using a mean calculation the data was re-run to locate the missing values. Further in attempt to get a more accurate forecast the history of the data was expanded to start at 01/01/2018.

A graph showing the temperature of a person

AI-generated content may be incorrect.

Figure 2 – missing data 2022-23 (similar for rainfall)

Checks are run for missing values and to check data types: variables are float 64 (floating decimal integers).

The Data is plotted, figure 3 as expected temperature shows a definite annual seasonal pattern reflecting winter and summer. There is no discernible trend. Figure 4 shows a normal temperature distribution grouped around the mean value with fewer hotter and colder days at the extremes.  
  
A graph showing the temperature

AI-generated content may be incorrect.

Figure 3 – Temperature plot

A graph of a temperature distribution

AI-generated content may be incorrect.

Figure 4 – Temperature Distribution

Figure 5 shows the Rainfall plot visual inspection does not show and trend or pattern in the data available. Figure 6 the distribution shows a right skew most days with no rainfall tailing off to extreme rainfall.

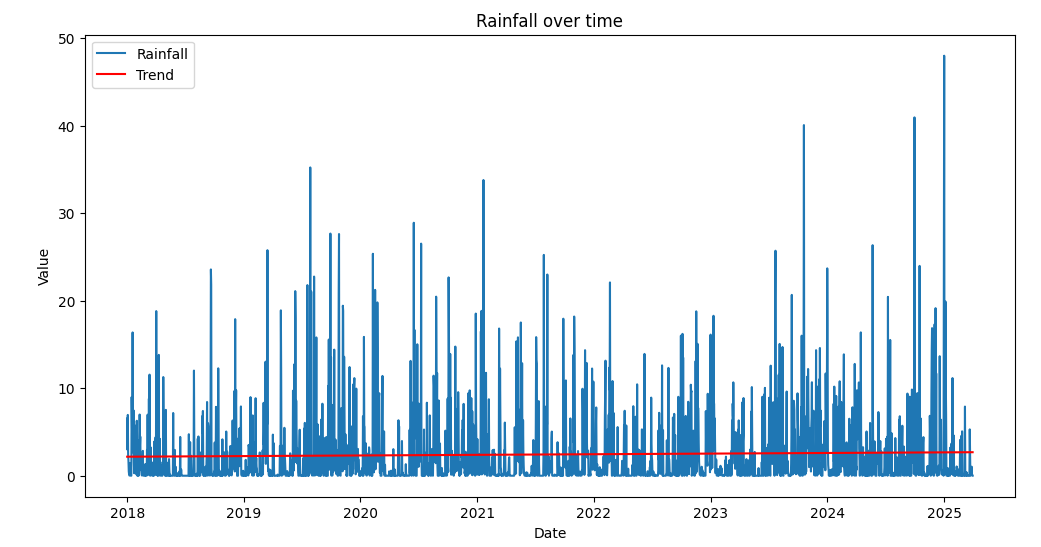


Figure 5 – Rainfall plot

A graph of a number of rainfall

AI-generated content may be incorrect.

Figure 6 – Rainfall distribution

Summary data statistics are printed there are 2467 lines of Data, the mean temperature is 8.8 degrees Celsius, mean rainfall is 2.4 mm. The quartiles are printed and can be used as the classifiers for hot/cold & dry/wet days.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 7 – Summary statistics

# Model

The model selected to make predictions is ARIMA:  
*“ARIMA, which is one of the most popular (if not the most popular) time series forecasting techniques. ARIMA is popular because it effectively models time series data by capturing both the autoregressive (AR) and moving average (MA) components, while also addressing non-stationarity through differencing (I). This combination makes ARIMA models especially flexible, which is why they are used across very different industries, like finance and weather prediction.”* (datacamp, 2025)

The model is made up of three parts

Auto regressive (AR) – The model checks how lags in the data correlate with themselves to build a trend forecast.

Moving Average (MA) – Looks at the residual errors of previous data points and compares them to the observation. This can assist identifying trend and reduce the impact of random events.

Integrated (I) – The Box-Jenkins method for ARMA accurate forecasting states the data needs to be stationary. A technique to make the data stationary is differencing, this is the difference between current and previous observations. This part accounts for the differencing steps required to make the data stationary.

The time series is a simple model it uses previous values to make predictions about the future. It’s best used for short term forecasting.

### Stationarity – Augmented Dicky Fuller Test (ADF)

A stationary data series is one that does not have a trend over time. The ADF is a hypothesis test the null hypothesis is that the time series is not stationary. The ADF tests the significance level under the p-value 0.05. If the result is under 0.05 then the null hypothesis must be rejected confirming the data is stationary.

A screen shot of a computer code

AI-generated content may be incorrect.

Figure 8 – ADF test results.

The p-value of the tests is below the significance value confirming the series is stationary. However, as observed there is an obvious seasonal trend in figure 9.

A graph of blue squares and numbers

AI-generated content may be incorrect.

Figure 9 – seasonal box plot

To remove the seasonal trend seasonal differencing is applied to remove the seasonality. The difference is taken from the observation and the observation 365 days before, removing the yearly cycle. You can observe this feature without any seasonality in figure 9.

A graph showing a number of orange lines

AI-generated content may be incorrect.

Figure 10 – temperature seasonally differenced

In order to optimise an ARIMA model we need to determine the p,d,q values. P is the best lag for AR part of the forecast. It can be determined by an Autocorrelation Function (ACF) (figure 10) plotting the lags and how well they correlate with each other. Lags are the previous values in the series. D is the number of differencing steps required to make the data stationary. Q is the MA what is the best lag to go back to average over comparing the value to the first in the series. D can be determined by plotting the Partial Autocorrelation Function (PACF) (figure 10).

A group of graphs showing different types of data

AI-generated content may be incorrect.

Figure 11 – ACF PACF plots.

To interpret the plots where the bars (or lags) drop off is considered the optimal value, anything outside of the blue area is considered statistically significant. The values aren’t fixed and can be changed when training the model to find the optimal settings. For the first iteration of the rain forecast p=1,d=0,q=1. For temperature initially I will try p=5 as lags show over 60 (too much for the model), the others are in easier to infer d=0, q=1.  
The data is split into 80:20 train and test data sets continuous values are used for time series so we can compare the model against the test data.

## Results

The Rain and Temperature forecasts have both flat lined see figures 12 & 13. compared to the mean of the data 8 degrees Celsius and 2.43 mm of rain the forecast is April will be wetter and cooler than average for the year. The forecast results do not give sufficient information to categorise the days in April favourable to not as you can see in the visuals the forecast doesn’t change.

A graph of a graph showing a number of times

AI-generated content may be incorrect.

Figure 12 – Rain Forecast

A graph showing a graph showing a graph

AI-generated content may be incorrect.

Figure 13 – Temperature Forecast

Evaluating performance can be done with Root Mean Squared Error (RMSE) measuring Euclidian distance between the forecast and actual to give the error. The lower the RMSE the more accurate the forecast predicts actual. The scale is relative the RMSE 6.69 for Temperature indicates on average the prediction error is 6.69 degrees Celsius. The rain forecast RMSE is 5.75mm. These figures do not indicate an accurate model.

## Future Iterations.

The predictions do not reflect seasonality in the temperature forecast. One option would be to decompose the time series into trend, seasonality, and residual error then remove the seasonality and forecast that separately before adding it back into the final result.

SARIMA models or Seasonal ARIMA are models with additional p,d,q values for the seasonal component and an additional feature for the frequency of the seasonality in this instance 365 days. Using pmdarima library the auto\_arima function can calculates the optimal p,d,q values for use in the model. The additional p,d,q seasonal values allow the model to forecast seasonality accurately. This model would have been preferrable for the problem, however technical difficulties in the python environment prevented this extra work.

Machine learning functions in python can optimise and run a mass of different forecasting models. Assessing the performance of each to give you the optimum model.

## Reflection

Weather is notoriously unpredictable. The predictions of a time series model get weaker the further they go into the future. A more accurate model is required. This could then be supplemented by a short-term forecast nearer the time. Once an accurate rain and temperature model is generated further dimensions such as wind could be added.

# Appendices

Appendix 1 – python notebook



Appendix 2 – Data



# References

datacamp. (2025, January 2025). *ARIMA for Time Series Forecasting: A Complete Guide*. Retrieved 03 12, 2025, from datacamp: https://www.datacamp.com/tutorial/arima

# Bibliography

## Website used

[Time series Forecasting tutorial | DataCamp](https://www.datacamp.com/tutorial/tutorial-time-series-forecasting) (https://www.datacamp.com/tutorial/tutorial-time-series-forecasting)

[ARIMA for Time Series Forecasting: A Complete Guide | DataCamp](https://www.datacamp.com/tutorial/arima?dc_referrer=https%3A%2F%2Fwww.datacamp.com%2Ftutorial%2Farima) )https://www.datacamp.com/tutorial/arima?dc\_referrer=https%3A%2F%2Fwww.datacamp.com%2Ftutorial%2Farima)

[Practical Python for Time Series Analysis: A Real-World Guide](https://codezup.com/practical-python-for-time-series-analysis/) (https://codezup.com/practical-python-for-time-series-analysis/)