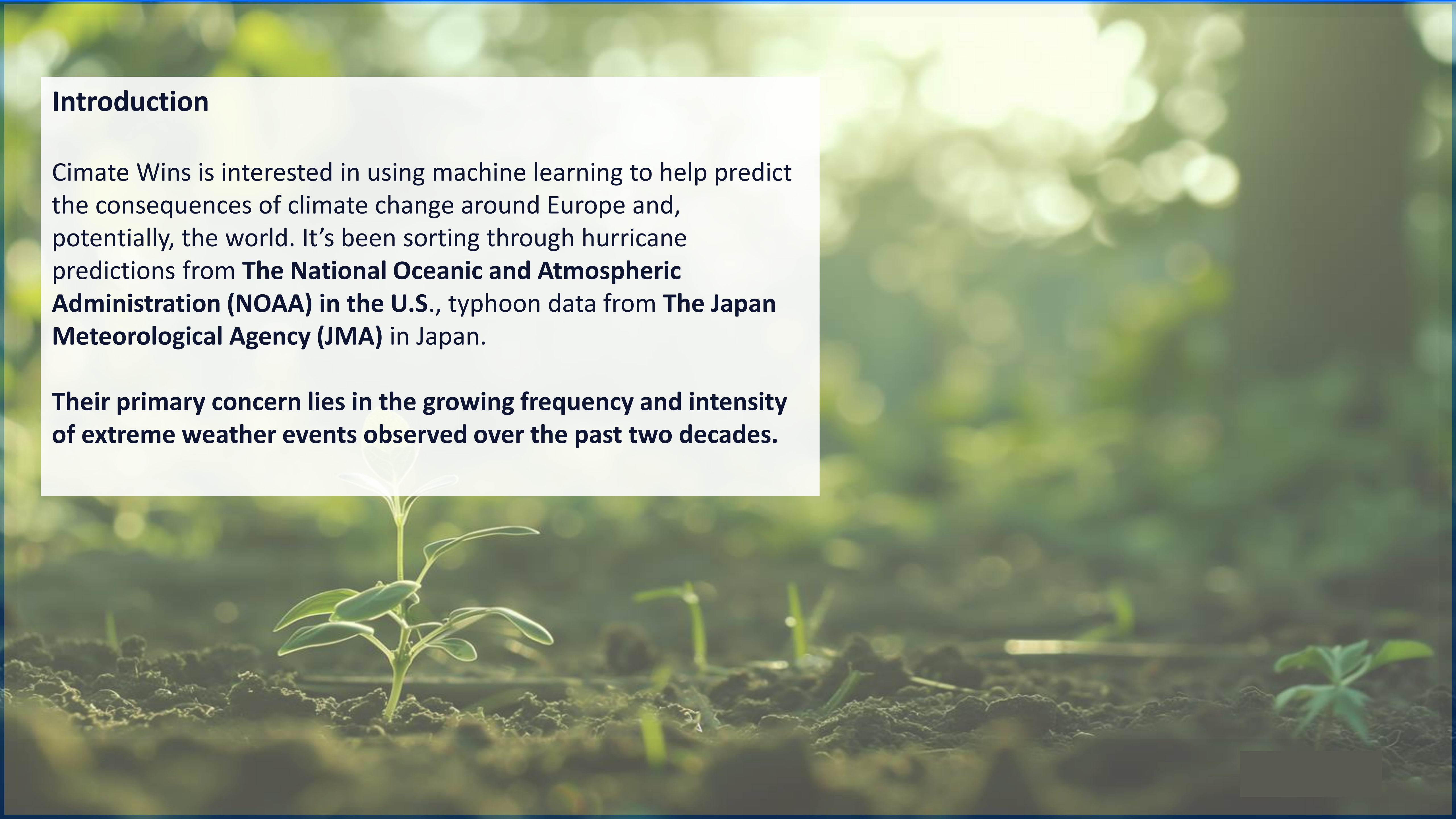
Weather Conditions and Climate Change

ClimateWins

Stephanie Ugwuanya

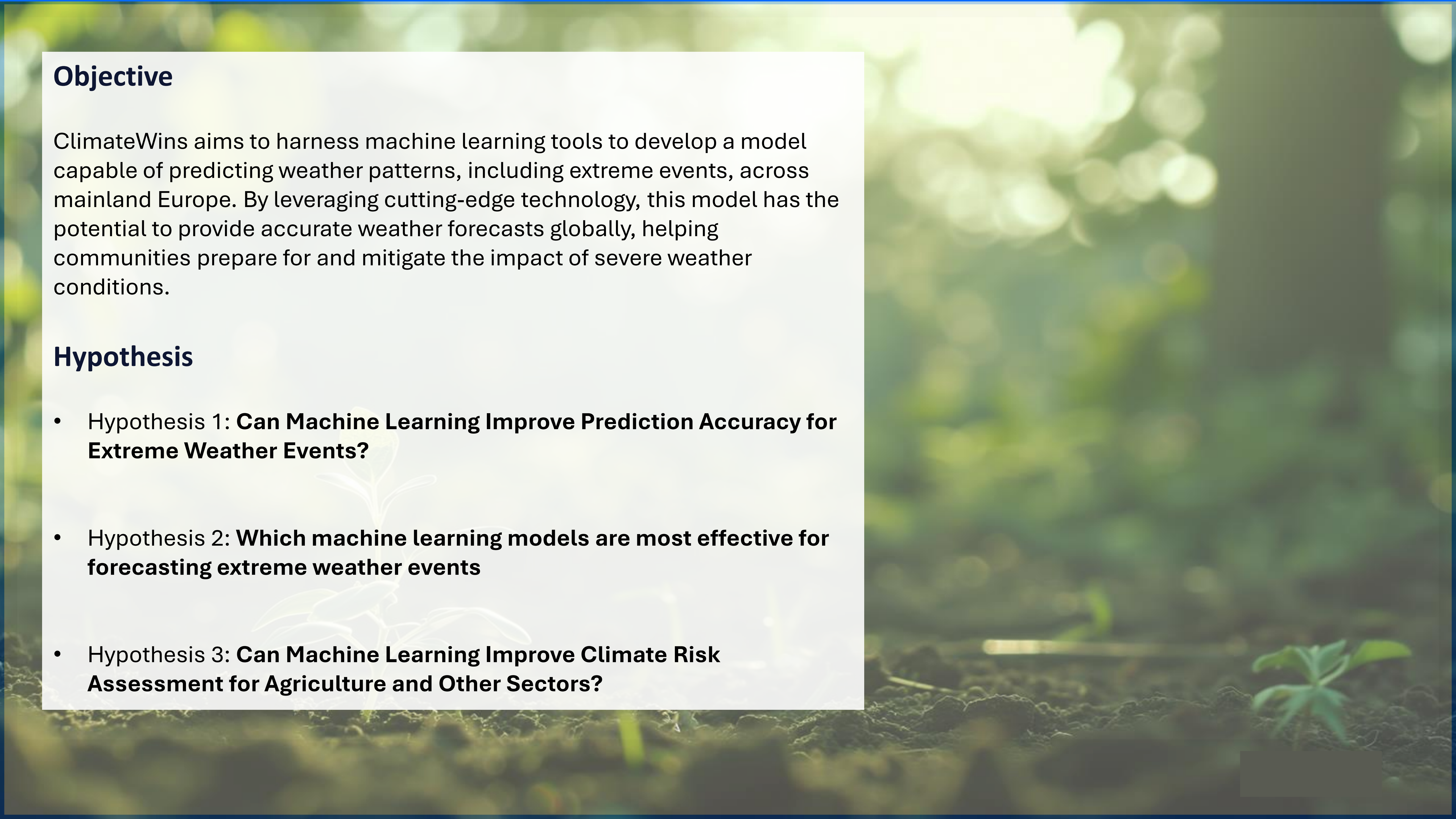
December 2024



Introduction

Cimate Wins is interested in using machine learning to help predict the consequences of climate change around Europe and, potentially, the world. It’s been sorting through hurricane predictions from **The National Oceanic and Atmospheric Administration (NOAA) in the U.S**., typhoon data from **The Japan Meteorological Agency (JMA)** in Japan.

Their primary concern lies in the growing frequency and intensity of extreme weather events observed over the past two decades.



Objective

ClimateWins aims to harness machine learning tools to develop a model capable of predicting weather patterns, including extreme events, across mainland Europe. By leveraging cutting-edge technology, this model has the potential to provide accurate weather forecasts globally, helping communities prepare for and mitigate the impact of severe weather conditions.

Hypothesis

- Hypothesis 1: **Can Machine Learning Improve Prediction Accuracy for Extreme Weather Events?**
- Hypothesis 2: **Which machine learning models are most effective for forecasting extreme weather events**
- Hypothesis 3: **Can Machine Learning Improve Climate Risk Assessment for Agriculture and Other Sectors?**

Data Set

Dataset based on weather observations from 18 different weather stations across Europe.

Contains data ranging from the late 1800s to 2022.

Recordings exist for almost every day with values such as temperature, wind speed, snow, global radiation etc.

This data is collected by the European Climate Assessment & Data Set project.

Dataset Link

Data Set Bias

Temporal Bias

•**Problem:** The data spans over 100 years (late 1800s to 2022). Changes in measurement tools, recording methods, and even changes in climate over time could lead to inconsistencies.

•**Impact:** Data recorded in the 1800s might be less accurate than data from modern weather stations, leading to biases in prediction models.

Seasonal Bias

•**Problem:** Weather data from the different stations may be more robust for certain seasons or years. For instance, certain stations might be more active or well-monitored in summer months when temperature fluctuations are more pronounced, while winter data might be sparse.

•**Impact:** Weather patterns and events are seasonal, and if certain seasons are underrepresented or overrepresented in the data, the model may not generalise well across different times of the year.

Human Bias in Data Recording

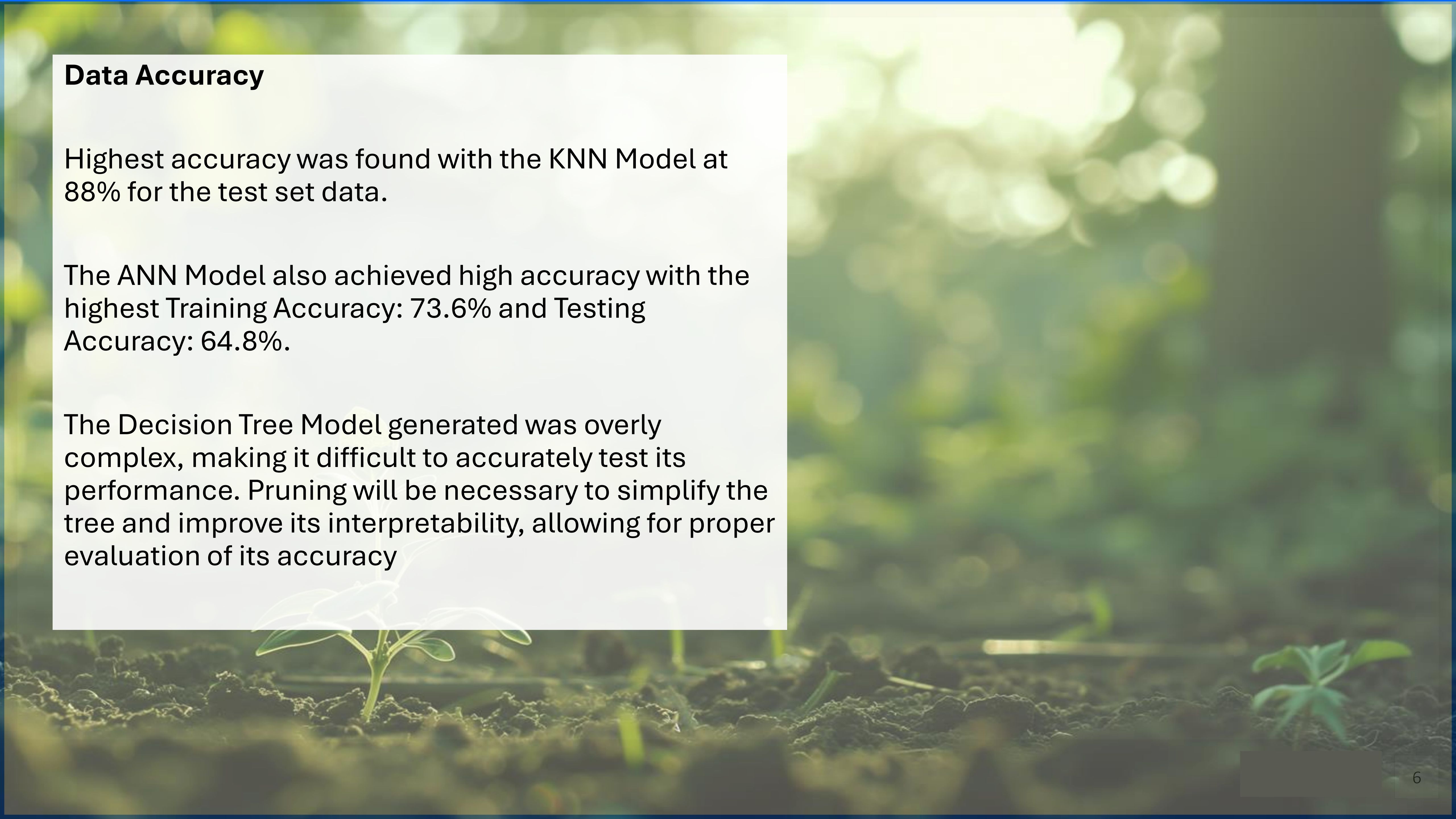
•**Problem:** Human error in data entry or observation can introduce bias. For example, stations in some areas may have more rigorous protocols, while others may have less detailed or incomplete records.

•**Impact:** Missing data or inaccurately recorded measurements can lead to gaps or distortions in the dataset.

Bias from Underreported or Missing Data

•**Problem:** Some stations might have data gaps or fewer measurements during specific years or months due to political, economic, or logistical challenges (e.g., war, financial constraints, or weather station malfunction).

•**Impact:** Gaps in data can lead to incomplete predictions and overemphasise more well-reported regions.



Data Accuracy

Highest accuracy was found with the KNN Model at 88% for the test set data.

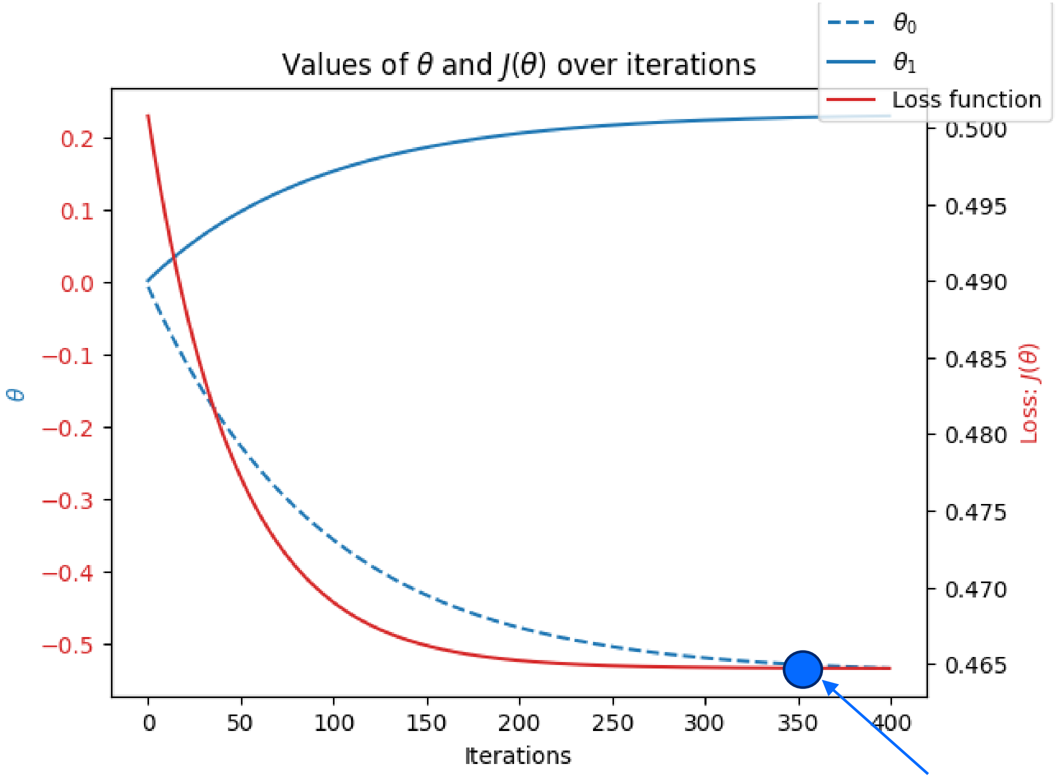
The ANN Model also achieved high accuracy with the highest Training Accuracy: 73.6% and Testing Accuracy: 64.8%.

The Decision Tree Model generated was overly complex, making it difficult to accurately test its performance. Pruning will be necessary to simplify the tree and improve its interpretability, allowing for proper evaluation of its accuracy

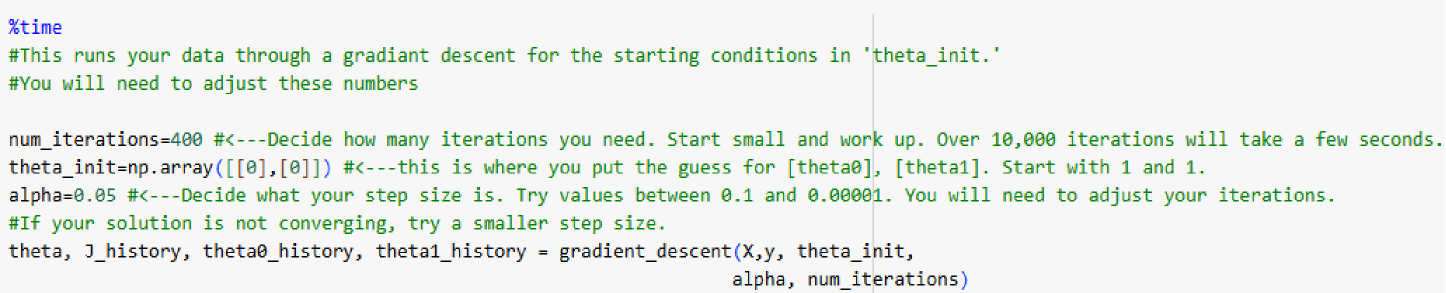
Data Optimisation

- The data set was optimised using Gradient Descent.
- Gradient Descent is a straightforward method for finding a local minimum (or valley), and it can be applied to both linear and nonlinear scenarios.
- In this case, we utilised Gradient Descent to minimise the error by iterating through several steps, adjusting both the number of iterations and the step size (alpha), which varied depending on the situation.
- By fine-tuning the values of theta0 and theta1, along with the iterations and alpha, we were able to achieve a result close to zero.

Example - **Gradient Descent Graph**



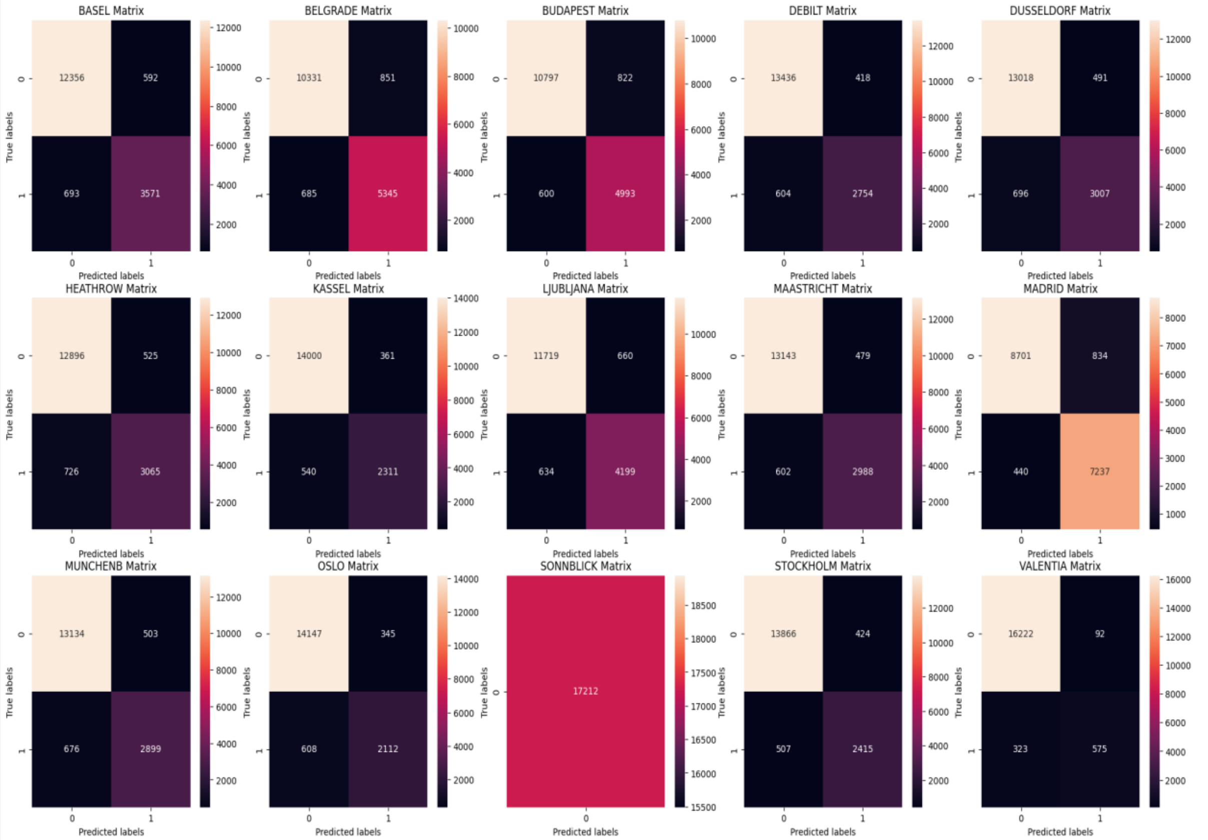
Local minimum



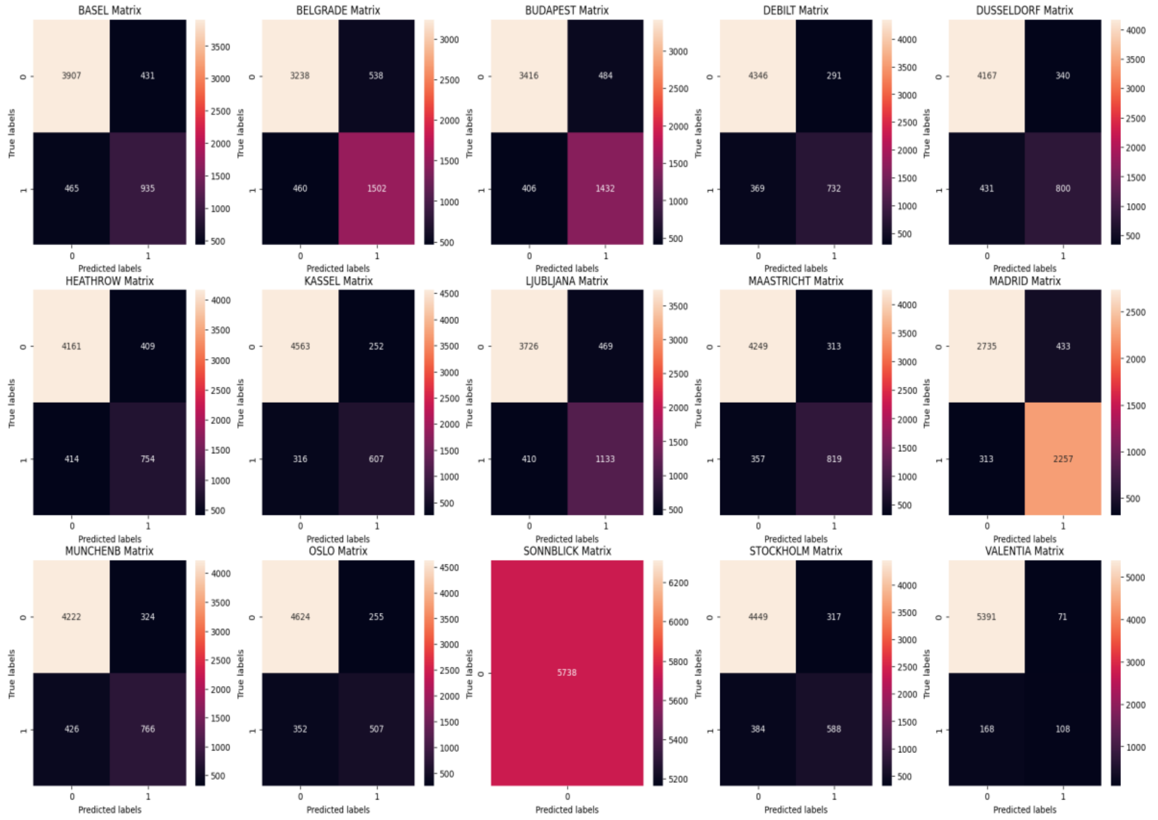
```
%time
#This runs your data through a gradiant descent for the starting conditions in 'theta_init.'
#You will need to adjust these numbers

num_iterations=400 #<---Decide how many iterations you need. Start small and work up. Over 10,000 iterations will take a few seconds.
theta_init=np.array([[0],[0]]) #<---this is where you put the guess for [theta0], [theta1]. Start with 1 and 1.
alpha=0.05 #<---Decide what your step size is. Try values between 0.1 and 0.00001. You will need to adjust your iterations.
#If your solution is not converging, try a smaller step size.
theta, J_history, theta0_history, theta1_history = gradient_descent(X,y, theta_init,
                                                                     alpha, num_iterations)
```

K-Nearest Neighbour Model

Training Set (Confusion Matrix) – **Accuracy : 93.91%**

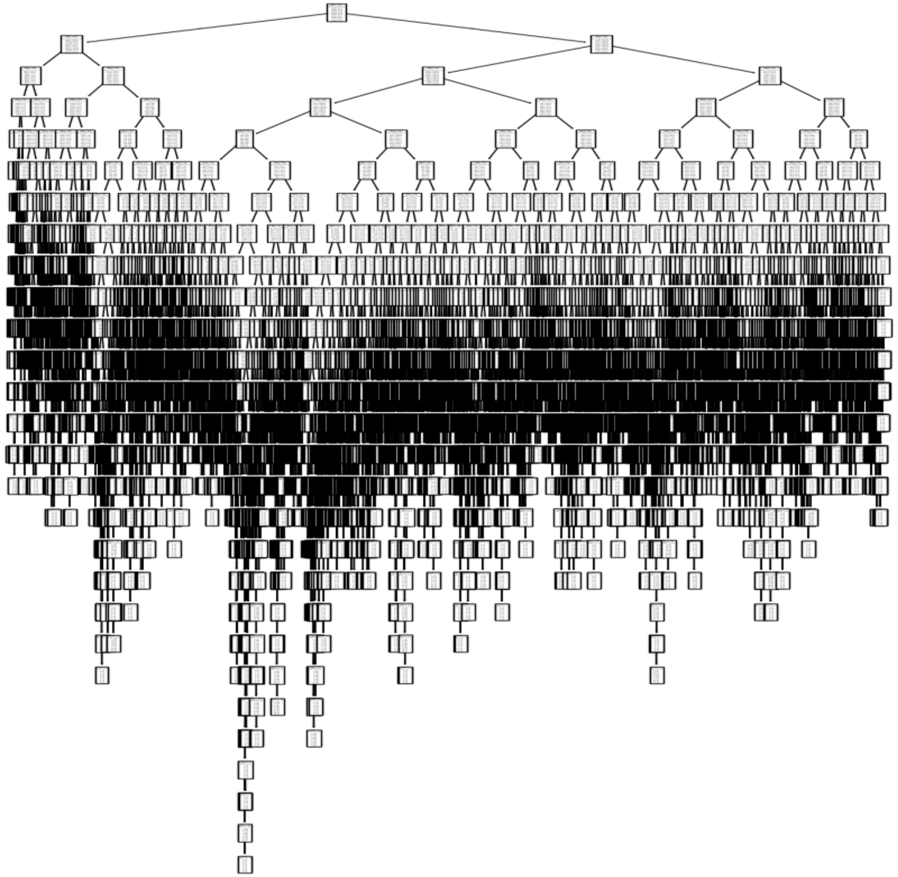


Test Set (Confusion Matrix) – **Accuracy : 88.34%**



The **K-Nearest Neighbor (KNN)** algorithm is a simple machine learning model that classifies data points based on their closest neighbours. When a new point needs to be classified, KNN looks at the "K" nearest points in the training data and assigns the new point to the most common class among those neighbours.

Decision Tree Model



A Decision Tree is a machine learning model that makes predictions by splitting data into branches based on certain conditions, much like a flowchart. At each "node" in the tree, the model asks a question (e.g., is temperature above 30°C?), and the data is split accordingly. This process continues until a final decision is made at a "leaf" node, representing the output prediction.

Test Data Accuracy

City	Accuracy (%)
BASEL	94.1%
BELGRADE	95.5%
BUDAPEST	95.8%
DEBILT	92.8%
DUSSELDORF	98.7%
HEATHROW	90.3%
KASSEL	94.4%
LJUBLJANA	98.2%
MAASTRICHT	93.6%
MADRID	93.9%
MUNCHENB	94.7%
OSLO	95.4%
SONNBLICK	100%
STOCKHOLM	92.4%
VALENTIA	93.7%

The test data shows high accuracy, but the decision tree is overly complex and requires pruning for practical use.

Issues with Complexity:

•Reduced clarity which is crucial for applications like extreme weather prediction where clear decision-making is essential.

•Difficult to communicate results to stakeholders.

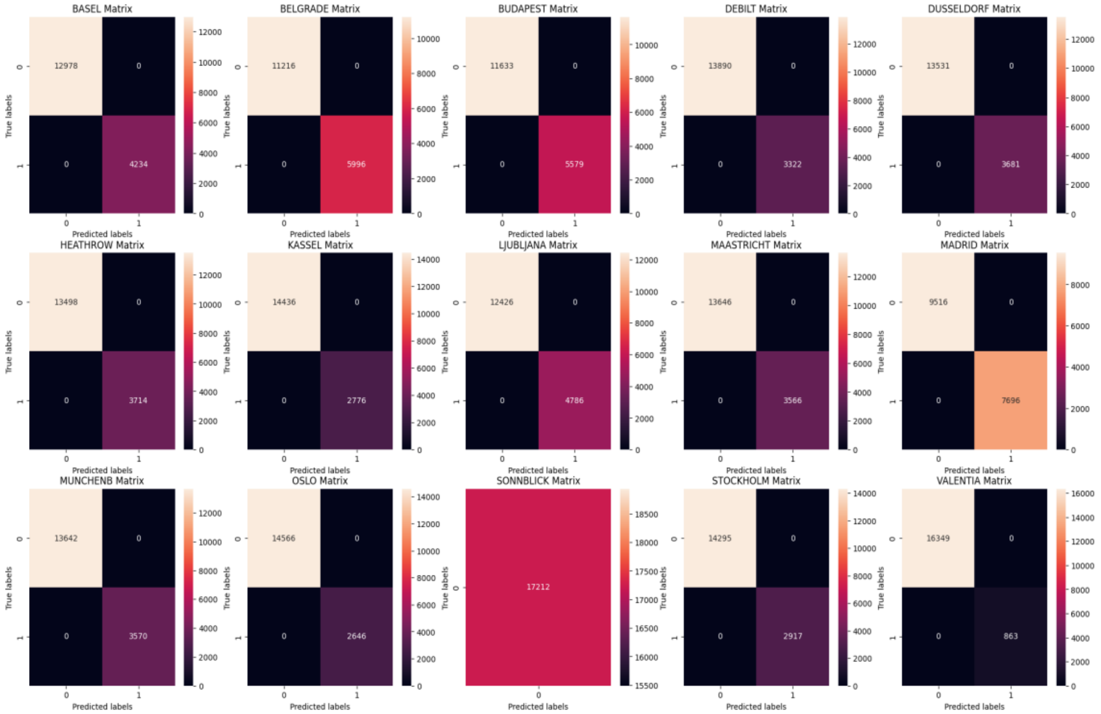
Prediction Time Concerns:

A large tree with excessive splits can slow prediction times, making it inefficient for real-time applications, such as predicting extreme weather events where rapid and accurate responses are critical.

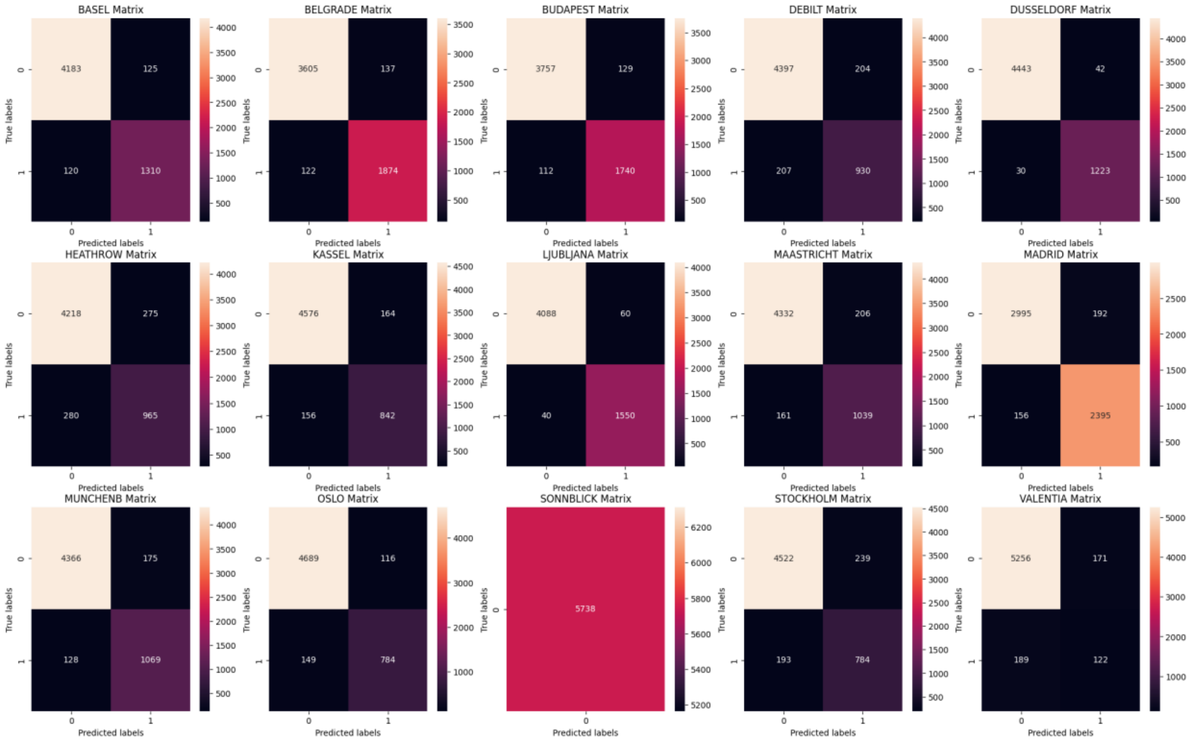
Pruning the tree would improve efficiency, interpretability, and usability.

Decision Tree

Training Set (Confusion Matrix)



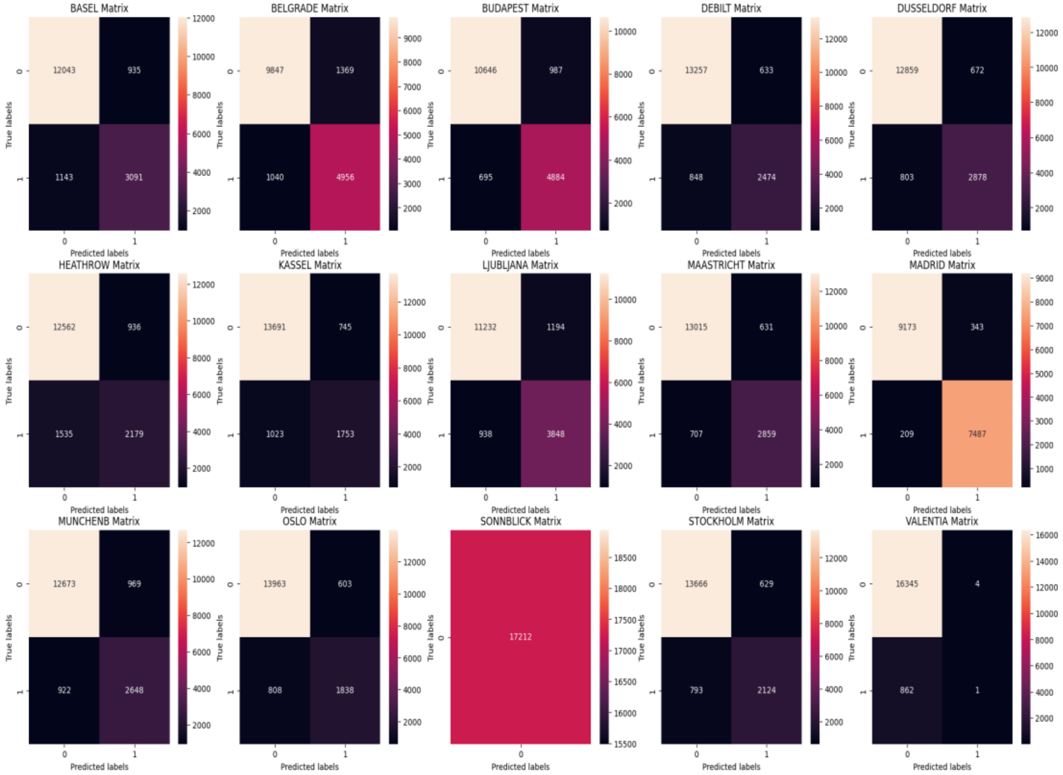
Test Set (Confusion Matrix)



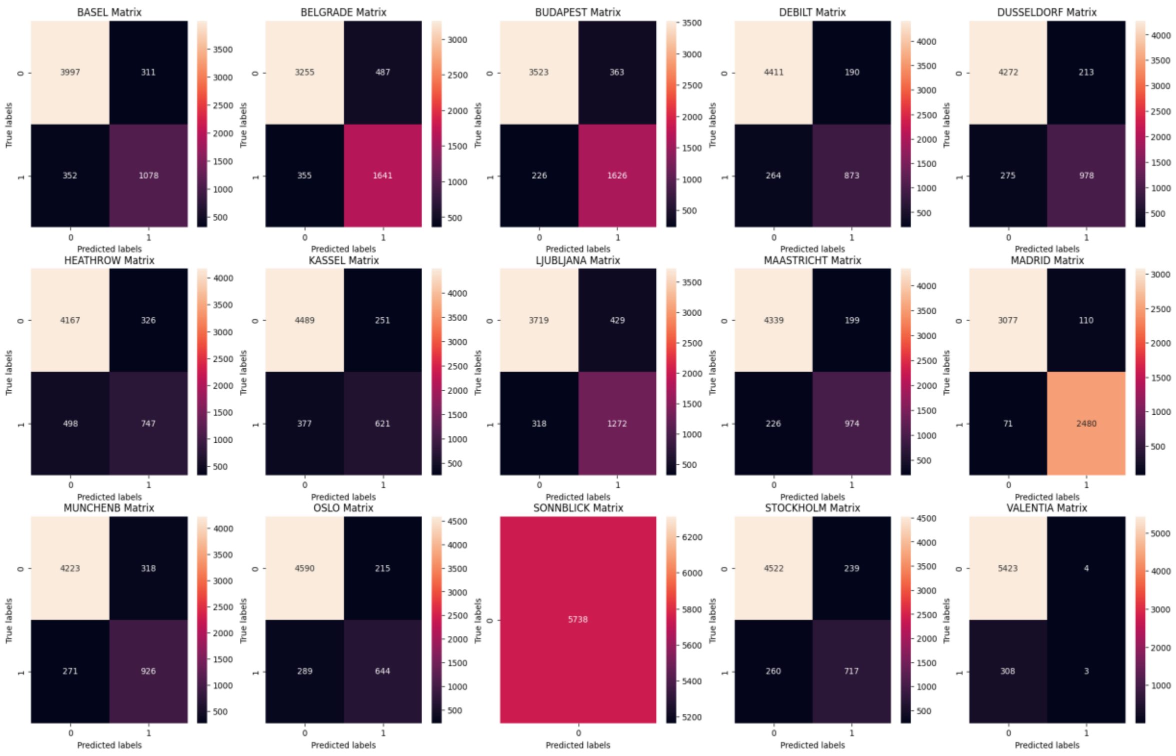
ANN Model (Scenario 1)

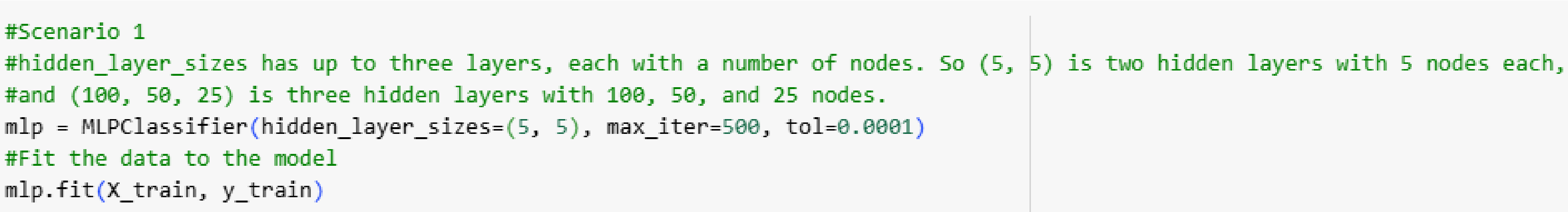
An Artificial Neural Network (ANN) is a computer model that works like the human brain. It has layers of connected "neurons" that process information. Each neuron makes simple calculations based on input data. The network learns by adjusting these connections to improve its answers.

Training Set (Confusion Matrix) -   Training Set **Accuracy - 49%**



Test Set (Confusion Matrix) -   Test Set **Accuracy - 49%**

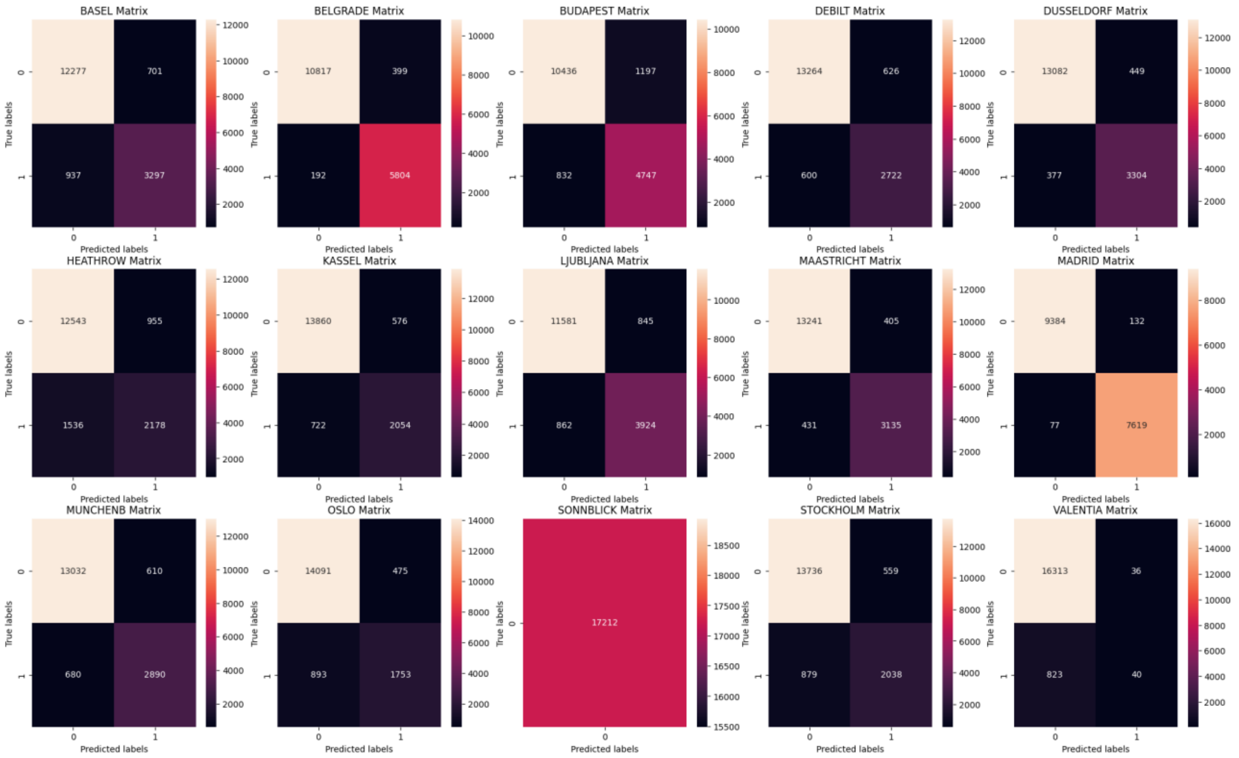




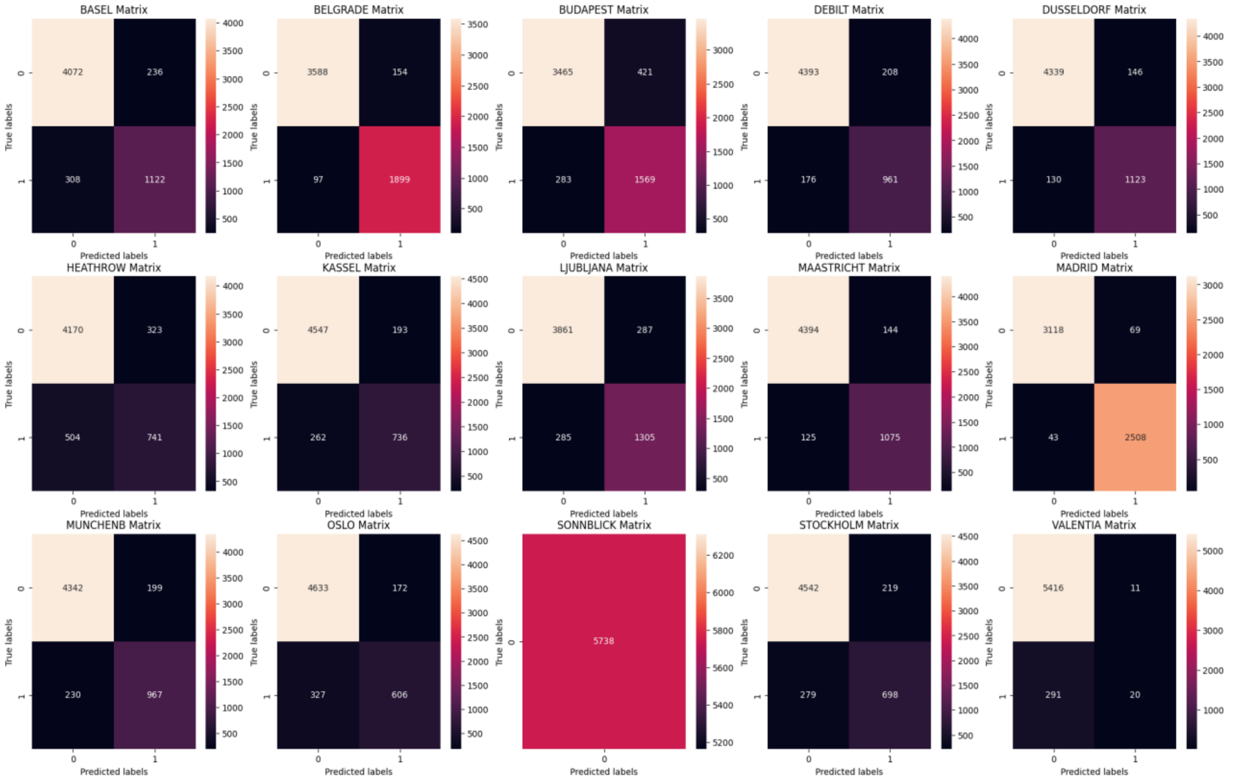
```
#Scenario 1
#hidden_layer_sizes has up to three layers, each with a number of nodes. So (5, 5) is two hidden layers with 5 nodes each,
#and (100, 50, 25) is three hidden layers with 100, 50, and 25 nodes.
mlp = MLPClassifier(hidden_layer_sizes=(5, 5), max_iter=500, tol=0.0001)
#Fit the data to the model
mlp.fit(X_train, y_train)
```

ANN Model (Scenario 2) -

Training Set (Confusion Matrix)

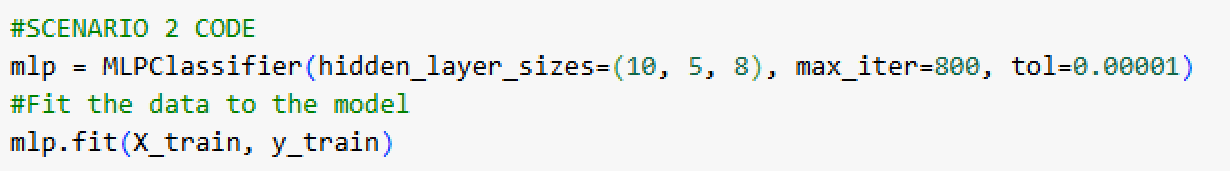


Test Set (Confusion Matrix)



Training Set Accuracy: 53.7%

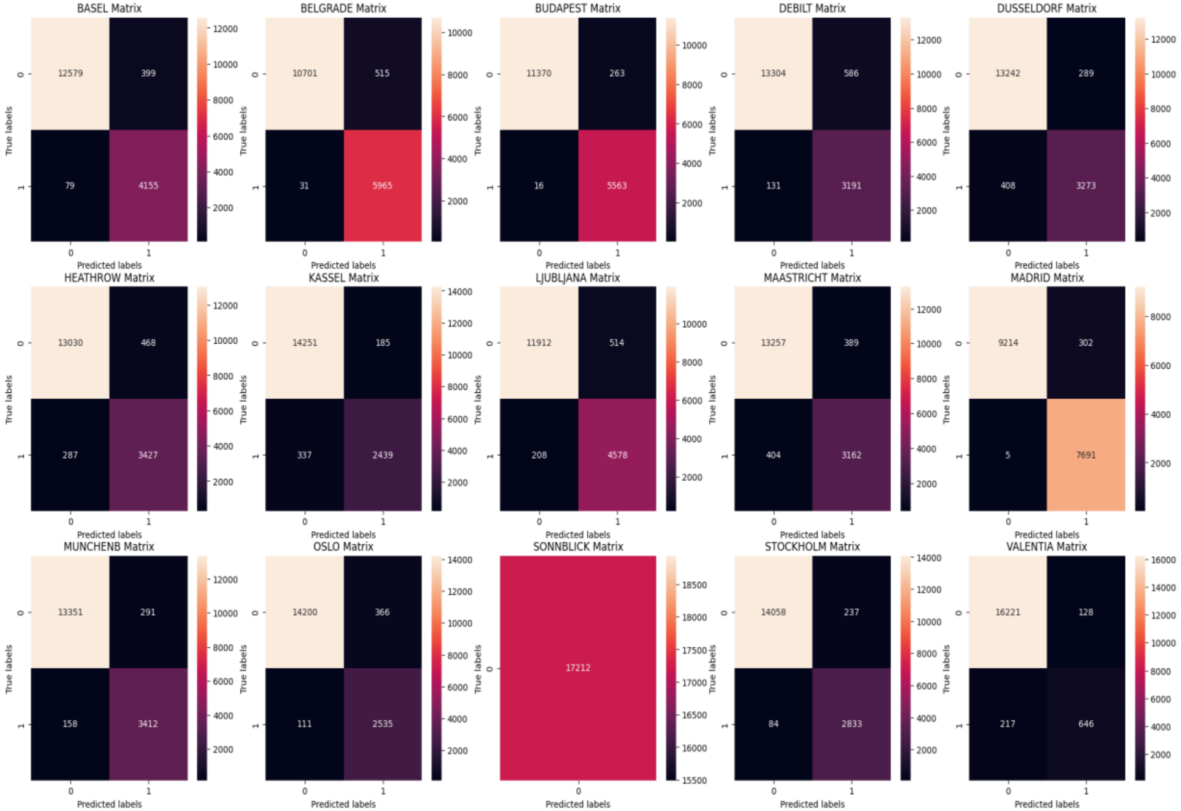
Test Set Accuracy: 53.2%



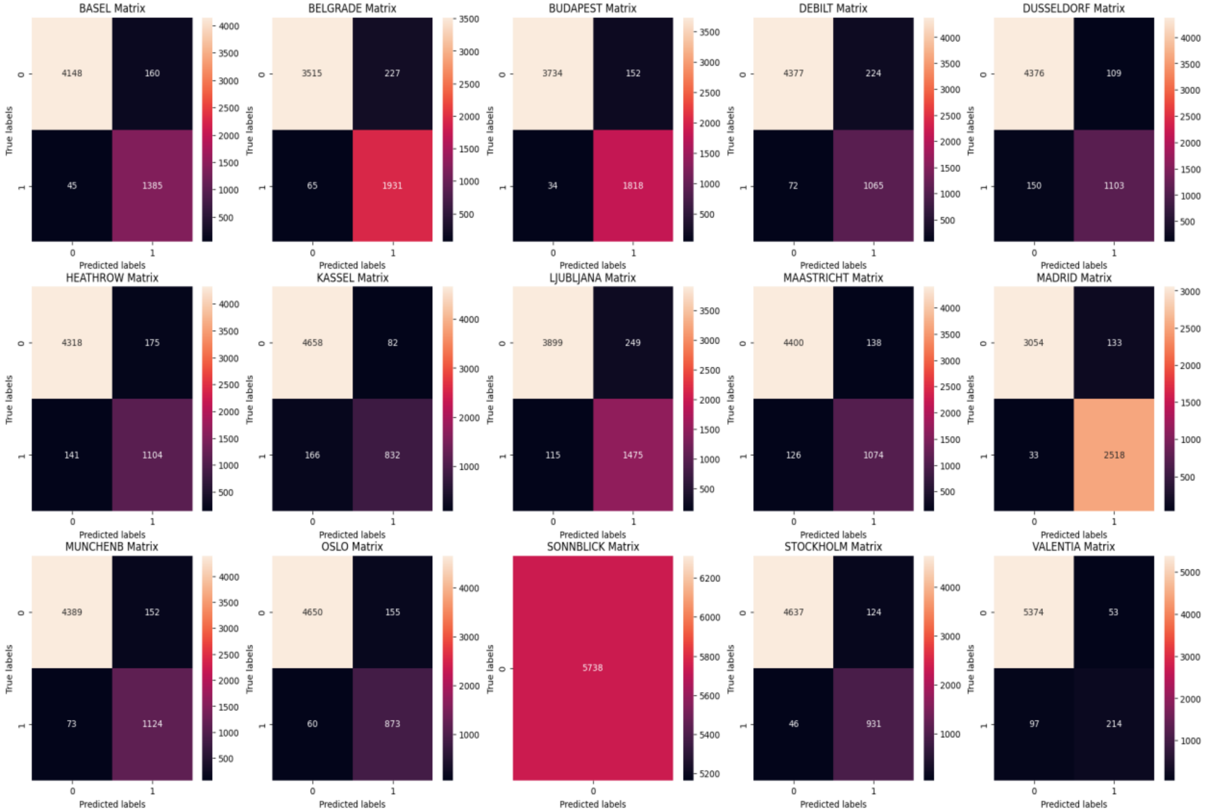
```
#SCENARIO 2 CODE
mlp = MLPClassifier(hidden_layer_sizes=(10, 5, 8), max_iter=800, tol=0.00001)
#Fit the data to the model
mlp.fit(X_train, y_train)
```

ANN Model (Scenario 3)

Training Set (Confusion Matrix)

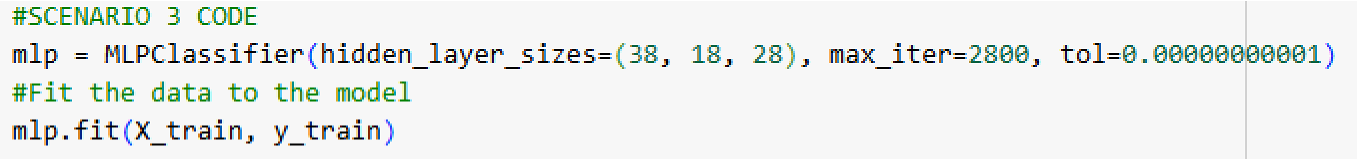


Test Set (Confusion Matrix)



Training Set Accuracy: 70.5%

Test Set Accuracy: 63.6%



```
#SCENARIO 3 CODE
mlp = MLPClassifier(hidden_layer_sizes=(38, 18, 28), max_iter=2800, tol=0.00000000001)
#Fit the data to the model
mlp.fit(X_train, y_train)
```

Hypothesis 1: **Can Machine Learning Improve Prediction Accuracy for Extreme Weather Events?**

Yes, The KNN model achieved the highest accuracy, reaching **93.9% on the training set and 88.3% on the test set**. Similarly, the Artificial Neural Network (ANN) model demonstrated respectable accuracy, with **70.5% on the training set and 63.6% on the test set**. However, it's important to note that the ANN model's performance was not consistent across all scenarios.

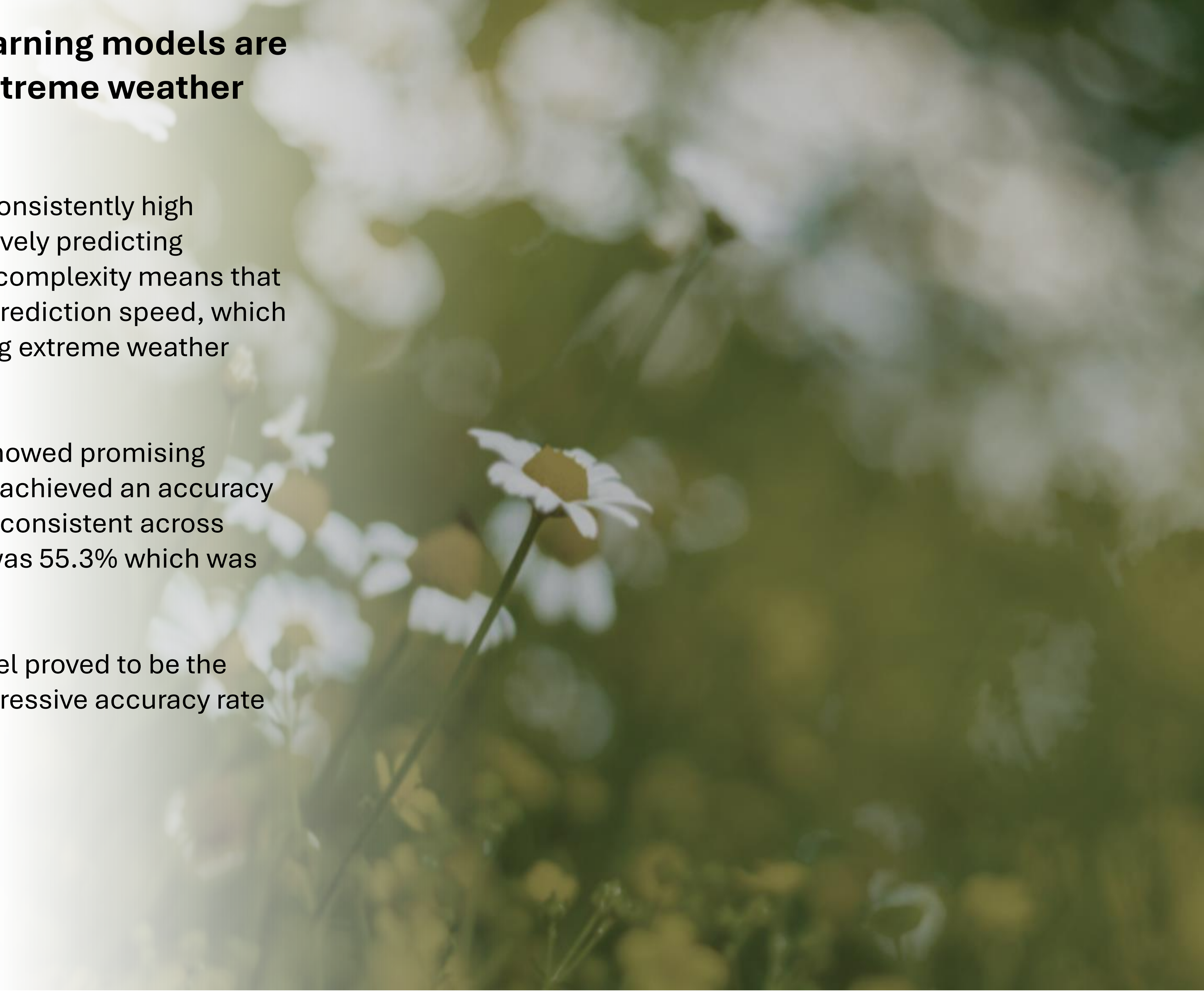
To further enhance overall performance, it would be beneficial to investigate specific locations with unusually high or low accuracy rates. For example, Sonnblick demonstrated a perfect 100% accuracy rate. While this might appear favorable, the consistency of this trend across all machine learning models suggests potential bias or issues within the dataset that warrant further analysis.

Hypothesis 2: **Which machine learning models are most effective for forecasting extreme weather events**

The Decision Tree model demonstrated consistently high accuracy across multiple locations, effectively predicting weather conditions. However, its inherent complexity means that pruning is needed to optise and enhance prediction speed, which is critical for responding to rapidly changing extreme weather scenarios.

The K-Nearest Neighbour (KNN) model showed promising results, particularly in Scenario 3, where it achieved an accuracy of 63.6%. However, its performance was inconsistent across different scenarios, and overall accuracy was 55.3% which was not as high as the ANN model.

The Artificial Neural Network (ANN) model proved to be the most reliable and robust, achieving an impressive accuracy rate of 88.34%.



Hypothesis 3: **Can Machine Learning Improve Climate Risk Assessment for Agriculture and Other Sectors?**

Further research is necessary to fully address this hypothesis. The high accuracy achieved with the KNN model is a promising starting point for its use in predicting weather conditions accurately. However, to confidently apply this model in risk assessments for industries such as agriculture, additional studies are needed. These studies should focus on identifying regions less vulnerable to extreme weather events and evaluating their economic and climatic stability.

The findings of this project serve as a strong foundation for future decision-making and demonstrate the potential of machine learning as a critical tool in forecasting and risk management. With continued development, these models could significantly impact how industries prepare for and adapt to changing weather conditions.

Conclusion

- The KNN Model achieved an average accuracy of 55%, indicating that it did not perform optimally in predicting weather conditions.
- In contrast, the ANN Model demonstrated significantly better performance, achieving the highest accuracy of 88.34%.
- The Decision Tree Model requires pruning to optimise its performance and allow for a more accurate assessment of its efficiency.

Next Steps

- We recommend continuing with the ANN model, as it proved to be the most successful in predicting weather conditions.
- This project provides a solid foundation for future decision-making and highlights the potential of machine learning as a valuable tool in weather forecasting and risk management. With further development, these models could play a pivotal role in helping industries prepare for and adapt to changing weather patterns.

Thank You

Stephanie Ugwuanya

LinkedIn GitHub

Stephanie-u@hotmail.co.uk

https://stephanie-u.wixsite.com/stephanieugwuanya