Predicting Weather with ClimateWins

Susan Wang | October 23, 2025

Overview

Objective:

ClimateWins, a European nonprofit organization, is interested in using machine learning to help predict the consequences of climate change around Europe and, potentially, the world.

Context:

As a data analyst for ClimateWins, I will assess the tools available to categorize and predict the weather in mainland Europe.

ClimateWins believes that even weather extremes could be predicted and planned for using advanced tools such as machine learning. With data from the past century, it hopes to create a model for what the future will hold.

HYPOTHESES

Machine-learning algorithms can be applied to the data to predict weather conditions.

Variations on geographical locations of weather stations affect the accuracy of predictions.

Machine learning can predict extreme weather events in the future.

Data Source and Biases

DATA SET:

- From 18 weather stations across Europe
- Ranging from late 1800's to 2022
- Values such as temperature, wind speed, snow, precipitation, global radiation and more
- Collected by the European Climate Assessment & Data Set project

BIASES and other Considerations:

- There may be a **regional bias** towards larger cities where weather stations tend to be located
- **Human biases** may affect how machine learning is trained to detect extreme weather events
- Weather features from various stations are **not uniform**
- Type of **geographical region** may skew variables, such as snow fall or temperature

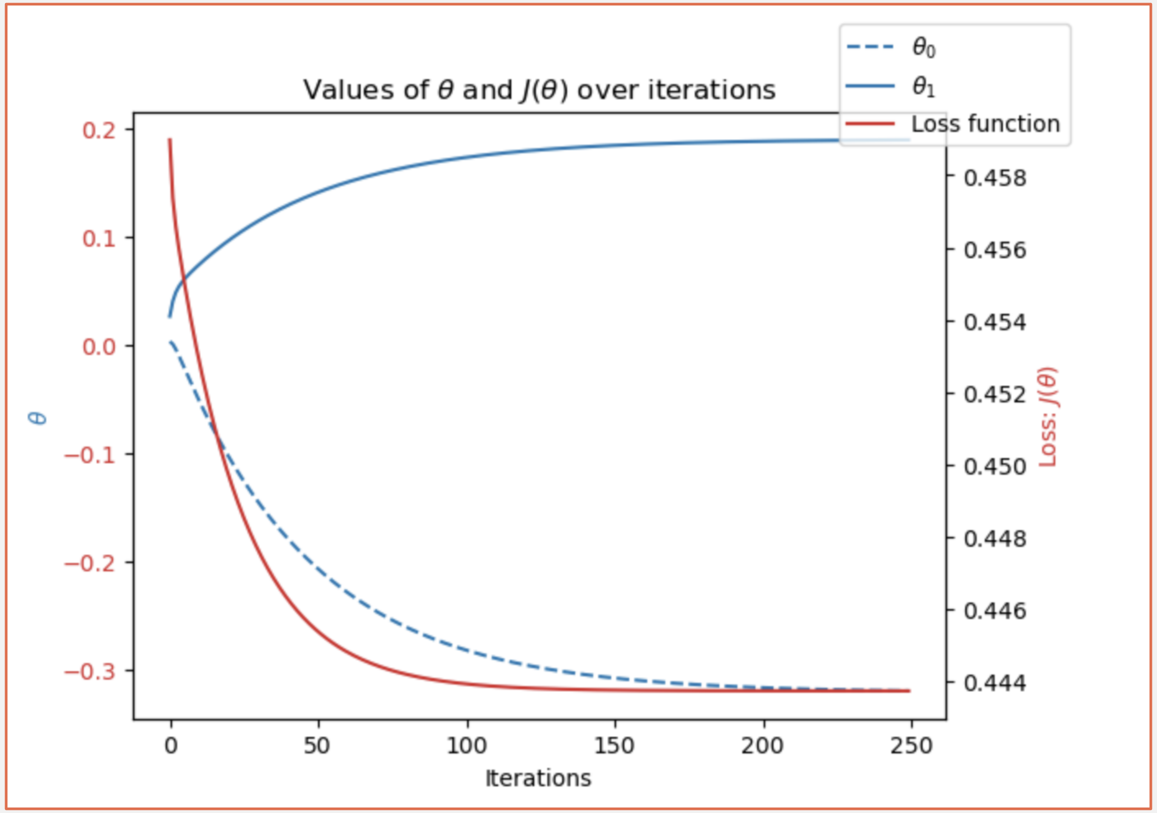
Optimization

To better understand the structure of our data and optimize it for machine learning, we can run an optimization algorithm such as **gradient descent** on a feature of the data.

Here we are fitting a regression model by running gradient descent on daily mean temperatures of various stations for a chosen year.

Gradient Descent on Kassel Station, 1960:

- ○ Iteration: 200, Step size: 0.1
- ○ Successfully converges
- ○ Minimum achievable loss at 0.43



Kassel 1960 Loss Function

Preparation for Supervised Learning

Provide the answers for training

A separate data set provides answers on whether a particular day in each weather station is pleasant or not. The model will train on predicting these answers.

Scale the data

Weather variables have differing range of values. Scaling the data prevents the machine learning from attributing more weight to higher values.

Split the data

The data is split 70% for training the model, and 30% for testing the accuracy of the trained model.

Apply algorithms

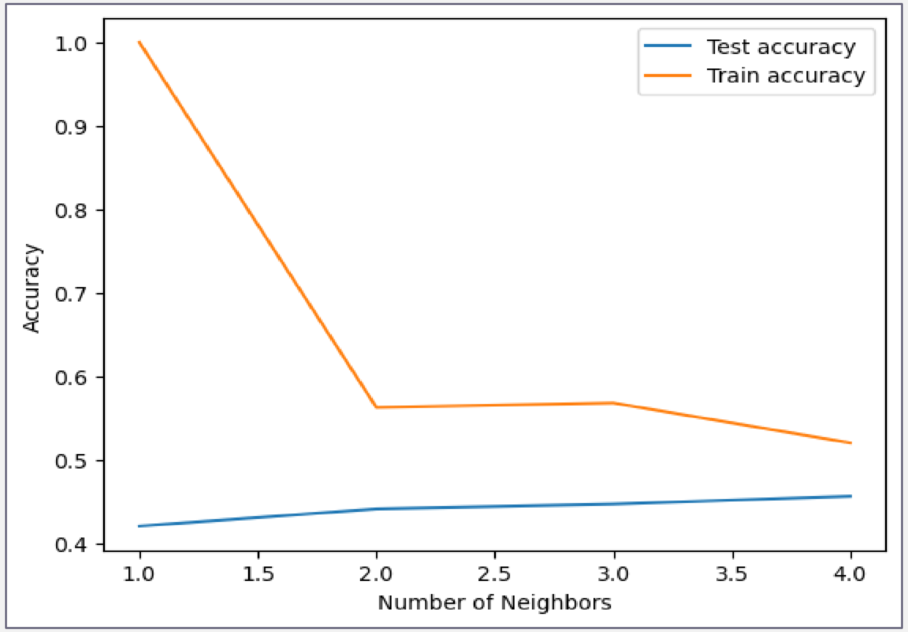
In this project, we will apply the K-Nearest Neighbor (KNN) model, the Decision Tree model, and the Artificial Neural Networks (ANN) model.

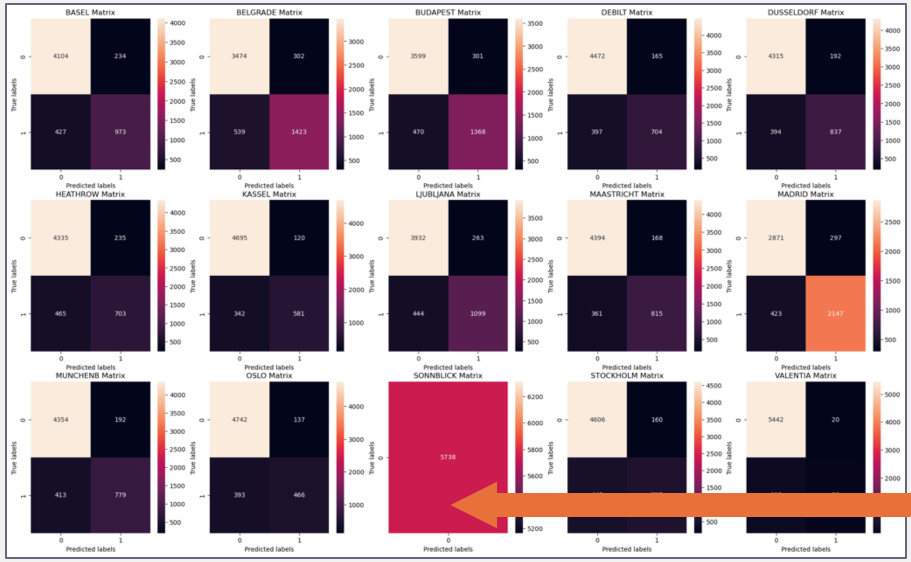
K-Nearest Neighbor (KNN)

Parameters: k-range = 1 – 4

Testing for Nr of Neighbors:

- Model is most accurate for the training set at 1 number of neighbors.
- Model works poorly for the test set, predicting the values at less than 50%.





Confusion Matrix

- The confusion matrix shows us that the KNN algorithm predicts with accuracy ranging from **85% to 100%,** with an average of 90%.
- Sonnblick station has 100% accuracy, meaning the model is **overfitting** to Sonnblick.

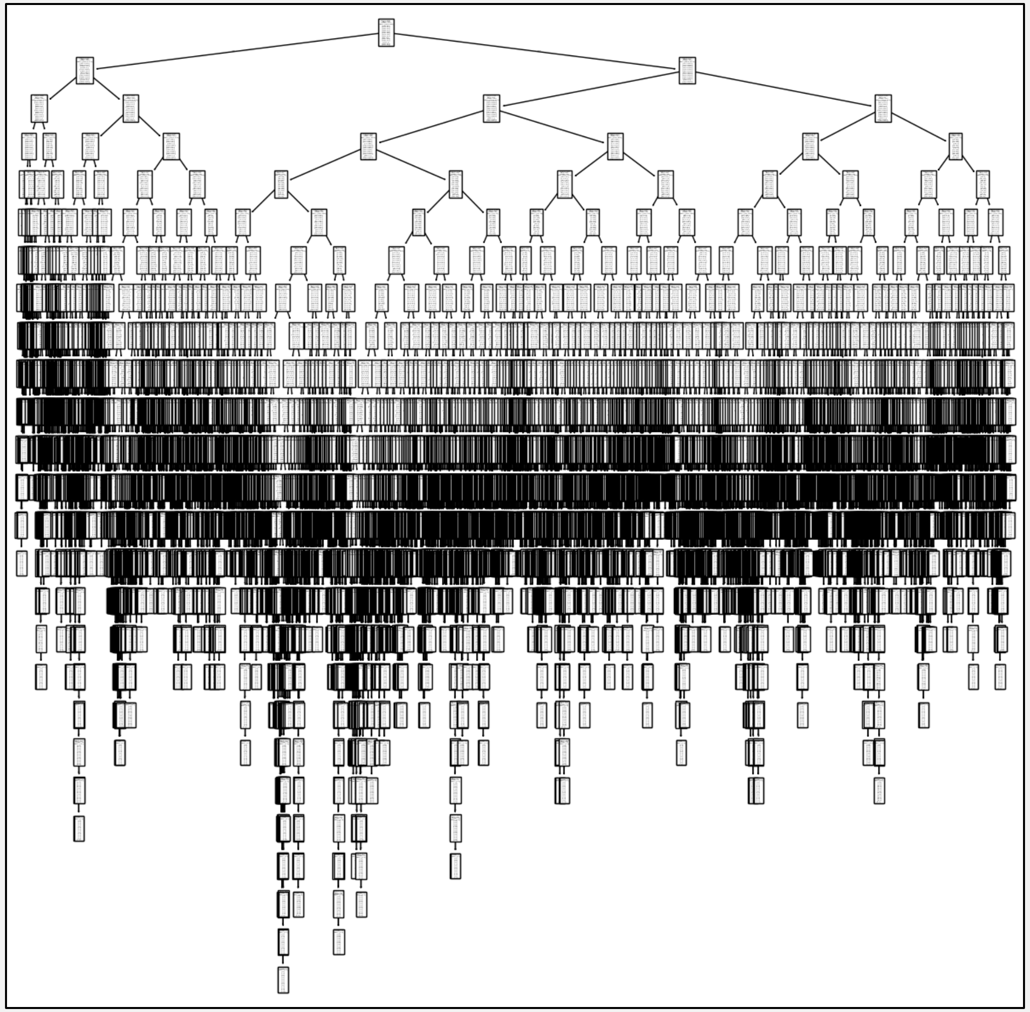
Decision Tree

Accuracy of the training and testing data:

Method	Train accuracy	Test accuracy
Cross validation	0.92	0.64
Confusion matrix	1.00	0.95

Overfitting

On the confusion matrix, this model predicts **100% of the training data.** This is a clear case of overfitting, as the model runs with much less accuracy on the test data. Pruning the decision tree may help with this issue.



Artificial Neural Networks (ANN)

Because of the hidden nodes, ANN involves experimenting with parameters to successfully converge and improve performance.

Here we have the results of three scenarios with varying parameters:

Parameters:

- Hidden layers = number of nodes in up to three hidden layers
- Maximum iterations = total number of times model will run
- Tolerance = lowest tolerance for loss improvement

Scenario	Hidden Layers	Max Iterations	Tolerance	Cross-validation Method		Confusion Matrix Method	
				Train Accuracy	*Test Accuracy*	*Train Accuracy*	*Test Accuracy*
1	(5, 5)	500	0.0001	0.49	0.50	0.91	0.91
2	(20, 10, 5)	1000	0.0001	0.56	0.55	0.94	0.93
3	(50, 20, 10)	1000	0.00001	0.71	0.62	0.97	0.95

Scenario 3, *with increased hidden layers, higher iterations, and lower tolerance* has produced the best accuracy yet. Further experimentation with parameters may yield even more accurate results.

Conclusion

Best Model for Predicting Weather:

Model	Test Accuracy (Confusion Matrix)
KNN	90%
Decision Tree	95%
ANN	95%

Why the ANN model works best in predicting current data:

- Most accurate predictions on both training and testing sets
- Less overfitting than the decision tree model
- Room for improvement and experimentation with the parameters

Next Steps:

1 **Improve the quality of the data set**
Observation of the distribution of different features shows that certain variables such as precipitation and snow depth have many outliers which may contribute to overall accuracy of the models.

2 **Address the next hypothesis**
Run further machine learning algorithms to evaluate occurrences of extreme weather events over the years and make predictions on future events.

Thank you!

Feel free to contact me if you have questions or comments.

Visit the project on Github: