PREDICTING CLIMATE CHANGES WITH MACHINE LEARNING.

By: Jeff Liv
Date: 08/26/2024

CONTENTS

PROJECT OVERVIEW

ClimateWins has interest in Machine Learning and the space it can fill with Weather Prediction; What algorithms can be used to leverage Machine Learning to help identify and analyze past trends, as well as predict future weather conditions.

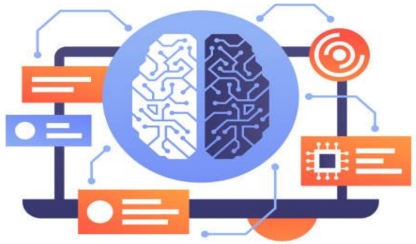
ETHICAL CONCERNS

Using AI even in the Weather Predicting space does pose some ethical concerns. Some of these concerns are:

- Data and Algorithmic Biases – If historical weather data is biased or incomplete, predictions may perpetuate biases based on the data, which can result to inaccurate predictions.
- Model Transparency and Accountability – Many deep learning models are considered ‘Black Boxes’, making it difficult to understand how decisions are made.

- Environmental Impact – Training large machine learning models can be complex. This may require significant computational power.
- Global Disparities – Not all regions have equal access to technological infrastructure needed for accurate machine learning.
- Model Overfitting and Ethical Use of Predictions – Machine Learning models trained on current climate data may not adapt to climate scenarios leading to poor predictions. Predictions have a moral responsibility to be used to protect lives and property.

MACHINE LEARNING ALGORITHMS



- Random Forest

An ensemble of decision trees that combines to produce a more accurate model than an individual tree.

The ensemble introduces randomness in both data and features at each split. It combines the predictions of multiple trees.

- Convolutional Neural Networks and Recurrent Neural Networks

Neural networks that specialize in handling different types of data and tasks.

CNNs are best for tasks involving images or spatial data, where understanding local patterns and features are crucial.

RNNs are suited for tasks involving sequential data, where understanding the order and context of the data points is key.

- Generative Adversarial Networks (GANs)

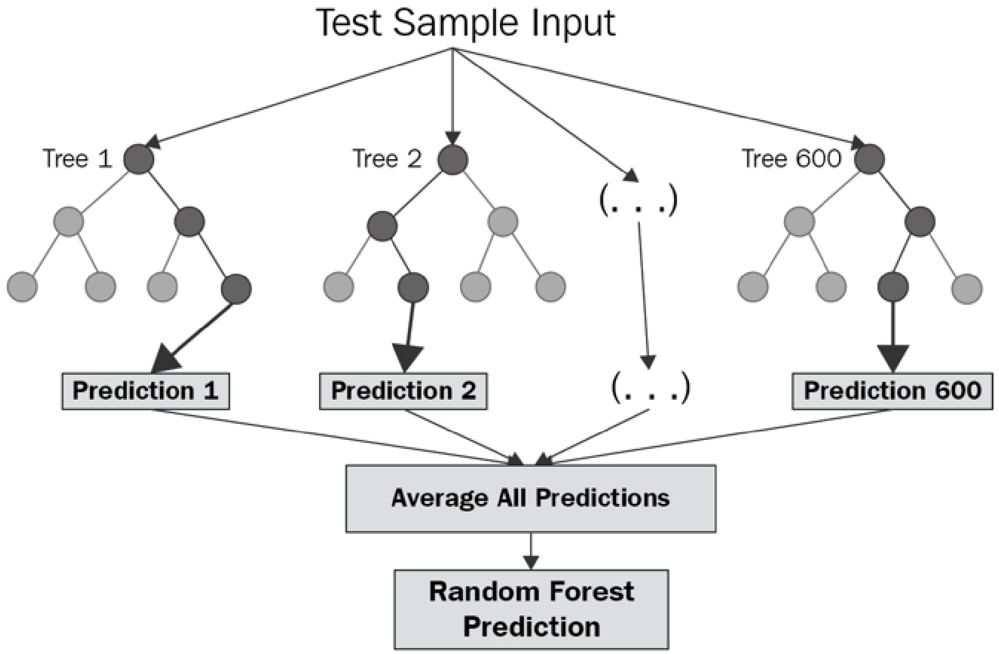
A class of neural network frameworks designed to generate new and synthetic data.

Widely used in tasks to generate images, videos, and even text by pitting a Generator and Discriminator network to produce highly realistic data.

CODE NAME: 'READER'

What if we could take our recorded weather history and have an algorithm that can determine what would be pleasant and unpleasant weather?

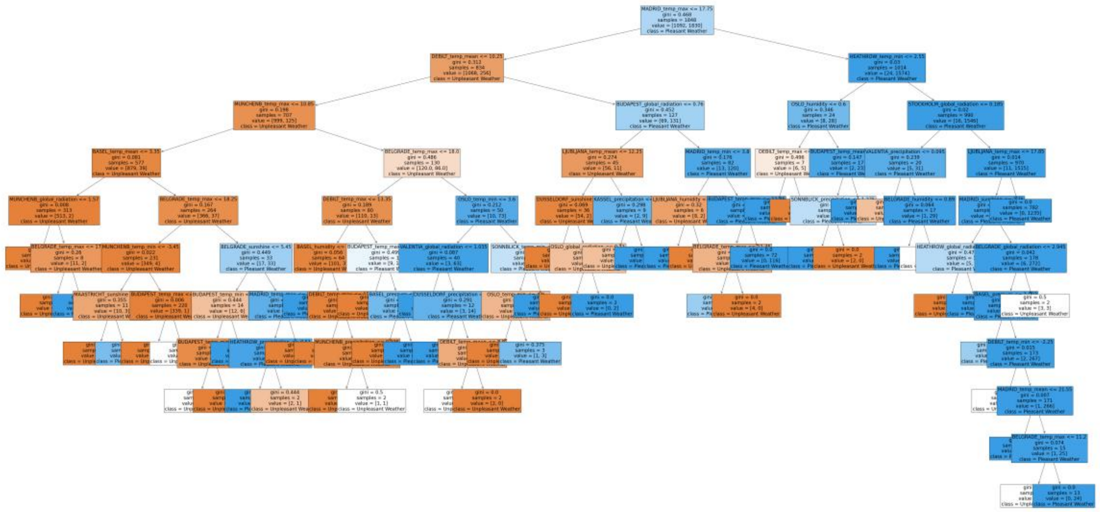
The Random Forest algorithm would 'Read' our past historical data to form classifications of pleasant and unpleasant weather.



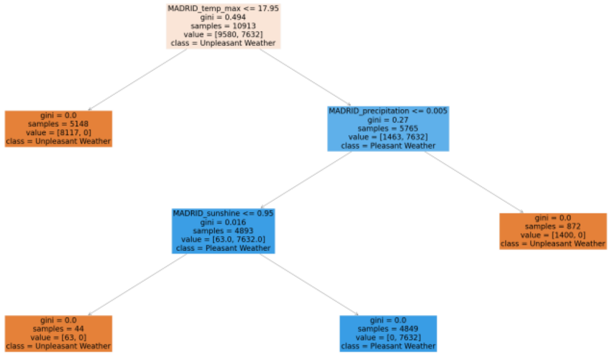
CODE NAME: 'READER' IN PRACTICE

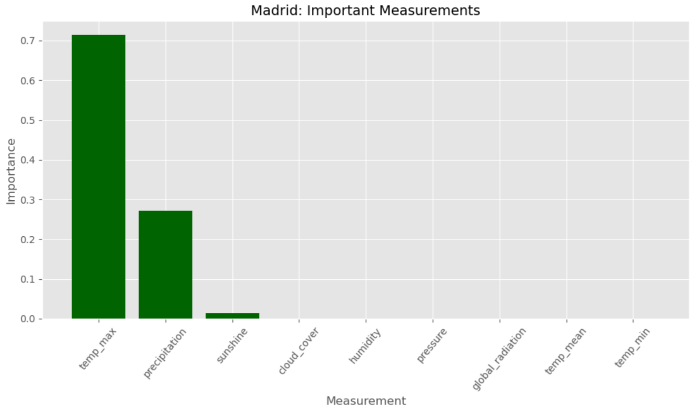
Here is what our random forests would look like in our case.

To the right, we see our major forest that envelops all weather stations to predict Pleasant and Unpleasant weather.



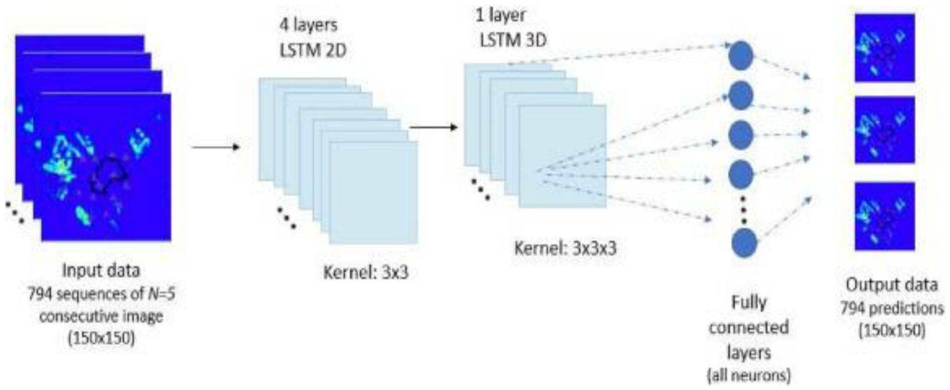
Below we have what it would look like for a singular weather station, and to the right of it the top three measurements that weighted the result.





CODE NAME: 'DECODER'

Now what if we could take our past historical data and find any patterns that were hidden previously, either based in time or spatial regions?



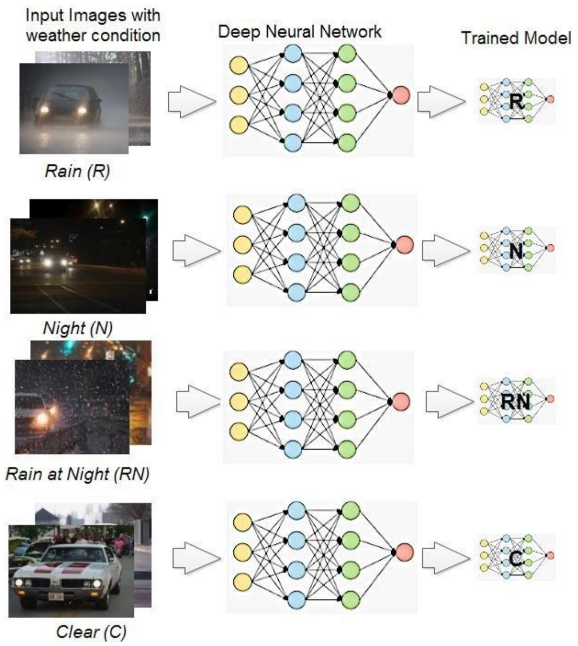
An example of RNN to the left.

The CNN and RNN algorithms 'Decode' our data; acting to finds patterns and make accurate predictions.

In practice, using the RNN algorithm would look at the last decade of weather for a station. It would take that decade and then use the changes of temperature to predict the weather for the future.

The CNN would then take spatial dependencies and patterns to find and predict values such as temperature gradients.

An example of CNN to the right.



CODE NAME: 'CRYSTAL BALL'

Finally, what would it be like if we could take our historical data and create realistic values, like images of what future weather could look like?

As you can guess, the last algorithm, would act as a 'Crystal Ball'. The most detailed, and rightly code named, as it could provide insight into what weather may look like.

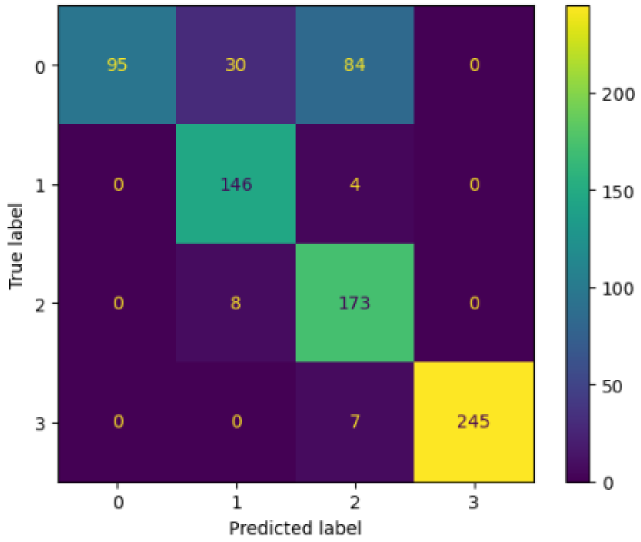
When implemented, it would take historical data of images and produce new images with the desired parameters. The example to the right shows an image along with the base parameter and desired parameter to produce an image of what it may look like.

(a) Sunny → Cloudy (b) Cloudy → Sunny (c) Cloudy → Snowy

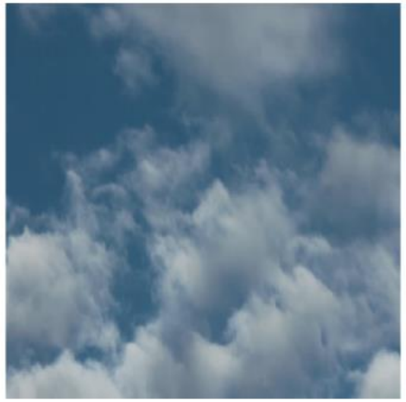
CODE NAME: 'CRYSTAL BALL' IN PRACTICE

Here is an example of what the trained GAN would look like. Here we have images in which our algorithm made predictions, as you can see, although one prediction of Cloudy is correct, the second is not. In this case we would need to train our GAN with more images.

Below is a confusion matrice of the images predicted. From the below, we can see it struggled to identify '0' which would have been 'Cloudy' Images.



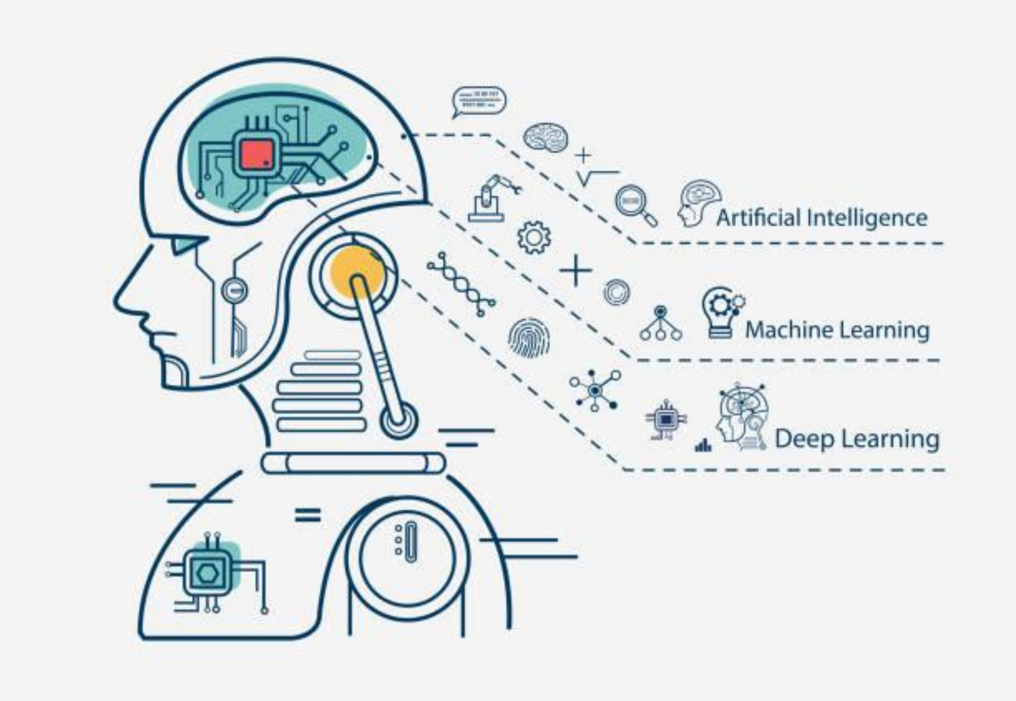
Correct Prediction - class: Cloudy - predicted: Cloudy[0.37554193 0.14413515 0.3283604 0.1519625]



Incorrect Prediction - class: Cloudy - predicted: Shine[0.11950131 0.180419   0.6746037 0.025476]

CONCLUSIONS & RECOMMENDATIONS

While all algorithms are viable options for what ClimateWins would like to achieve; it is recommended that ClimateWins leverages multiple algorithms to achieve the most ideal results.



FAQS & OTHER QUESTIONS

- Why all three?
  - While each algorithm individually does well, by using all three algorithms we cover more bases. This can result in the best predictions.
- If we were to select just one, which would be the most efficient?
  - That would depend on ClimateWins end goal;
    - For simple Pleasant/Unpleasant classifications, the Random Forest.
    - For spatial and weather pattern predictions our CNN and RNN models.
    - For real time predictions with images of what weather may look like, the GAN.

THANK YOU

Jeff Liv
801-859-8191
jchadliv@gmail.com