Leonora Ala

Weather Predicting Model

INTRODUCTION

My project is to predict the weather temperature for any month of the year, in my case I chose July. With the basic weather app on my phone, I find that it is not the most accurate; when it says it will rain it does not, cloudy when it says sunny, the temperature is always changing, etc. My goal is to find what the average temperature of July this year will be based on 5 years’ worth of data. This is interesting for me to complete since in the future I could continue to develop this model in order to predict the temperature day by day. This project is a great first steppingstone to understanding how the machine learning agent-based modeling and simulations work. ABMS is important for predicting the weather as it learns to recognize and analyze patterns within the given data which it then will use to create its prediction. The agents’ environment is fundamentally observable and is the maximum, average, and minimum temperature of each day for the month of July dating from 2016-2020. The way the agents interact with this environment is by taking the data, creating multiple different test cases, averaging all of these predictions from the test cases and then arriving at a final prediction.

RELATED WORKS

In my project I have used a random forest regression model (RFRM). Other works that have used this model include predicting the poverty percentage in Bangladesh by combining RFRM with data from the demographic and health surveys, household wealth index, and Google satellite imagery (*Estimation of Poverty Using Random Forest Regression with Multi-Source Data: A Case Study in Bangladesh*, 2–19-01-20). Another work models predicting the cases of a patient having a cardiovascular disease by analyzing existing data to help research and prevention in this area of healthcare (M. (2020, November 19). MansiMeena/Cardiovascular-Disease-Prediction. GitHub. <https://github.com/MansiMeena/Cardiovascular-Disease-Prediction>). An example of using RFRM in the stock market is by using it to optimize the prices of online stores. It can adjust prices, compare the price to another store/website, and change the price automatically based on the time of day and demand of the consumer (*Estimation of Poverty Using Random Forest Regression with Multi-Source Data: A Case Study in Bangladesh*, 2–19-01-20).

As you can see, random forest regression mode is very important in all aspects of society, whether it be based in healthcare, stock market predictions, or determining the future poverty of a city.

RANDOM FOREST REGRESSION MODEL

Diagram

Description automatically generated

Here is a basic illustration of how a random forest regression model works. We start with the test sample input, which is a value we decide on how many test runs we want the model to run with the given data. Each separate test case will arrive at a prediction. The next step in this model to gather each separate prediction and to average them all out. This final averaged value is the random forest prediction.

Text

Description automatically generated

Here is my code for implementing the random forest regression model. To access this model I had to use the scikit-learn API which helped tremendously in using the model.

Text

Description automatically generated

This is the results when you run the code. You get the baseline error that must be beat, the training margin of error, along with the testing margin of error (to compare), and the median absolute error in terms of degrees. Using this data we see the testing/validation MAE is much smaller in comparison to the baseline, therefore the random forest regression model has worked successfully.

DATA USED

Initially I wanted to access a weather website’s API and make calls to their API to extract an abundance of data. However, all the websites I came across wanted me to pay, so I had to resort to manually inputting the data into an excel spreadsheet. As this took up time, I did not input a vast amount of data. My data included the maximum, average, and minimum temperature of each day for the month of July dating from 2016-2020. To code my project, I used python. While using python I used the special library scikit-learn in order to access their mean and median absolute error, their make\_pipeline, StandardScaler, RandomForestRegressor, and SelectKBest in order to code the actual regression model.

Text

Description automatically generated

Looking at this picture of my coding of the actual RFRM, commenting to each parameter is the description/ reasoning to why I had used that variable and that value.

SUMMARY

What I found in my observations is that the RFRM works with a 3.29 average improvement to the baseline error prediction. I also found that training had a greater improvement than the actual test cases. This makes sense since I chose to train 75% and test 25%, if we switched the values up there would be slight differences in the final result. Based on the results I would say my model is working very well. I. learned the basics of implementing machine learning, and I am able to now advance to the next steps in my future.