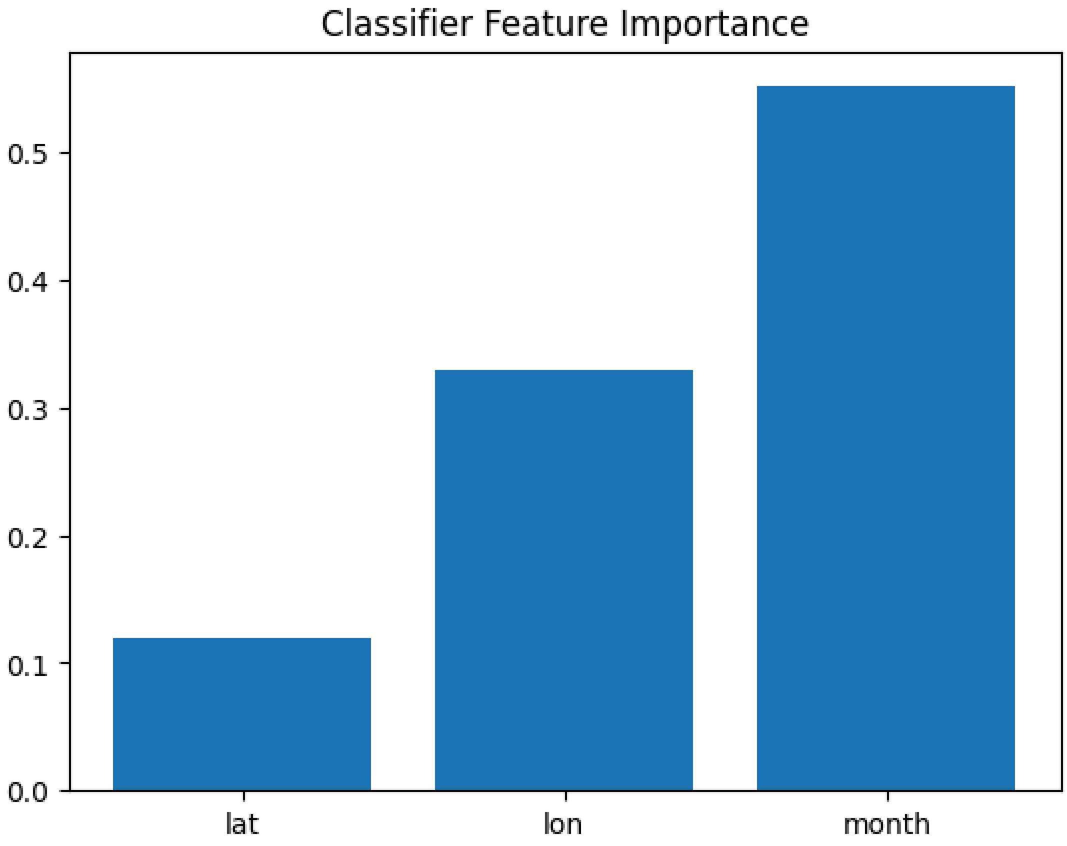
Temperature predictor uses a random forest regressor model trained on longitude, latitude, year, month, day, and day of week. The data was sourced from https://github.com/neetika6/Machine-Learning-Model-for-Weather-Forecasting/tree/main ,where the content of the weather from kanpur.csv were parsed to include only historical data involving temperature from 2014-2021. Since the data was divided in two csv file was first combined the data into a singular data frame, afterwards we preprocessed the data in order to split the columns within the data frame, convert the data types and handle any missing values. (Quick note the temperature data type changed to Fahrenheit though a quick edit to other temperature type would alter temp output) Features such as year, month, day, and day of week and target which is temperature are selected and split into two sets of data for training and testing. The model provides us Mean absolute error, Mean squared error, R-squared(Goodness of fit) to determine the accuracy of the temperature predicted. Given user input for any location in the United States and date the model will output a temperature with an average deviation being 2.29(MAE), the MSE also providing a difference value, and R-squared giving a metric indication of the proportion of variance ranging from 0-1, (R-Squared = 0.921).

The rainfall model used a random forest classification model. It was trained on longitude, latitude, and months of the year. The model was trained on data pulled from earthaccess. Specifically, the data was pulled from`"M2SDNXSLV".` In order to specifically get more data, we looped over 10 latitudes and longitudes in a min max region. We set our bounding box to the United States. The reason why this is important is because for certain regions such as the tropics or very dry areas, it may not respond correctly. Below is the classifier feature importance. It shows that while the model was trained on various different aspects, the domininating feature that trained the model was the month with the next feature being the longitude of the area. The model had achieved a 63 percent accuracy. In order to increase the accuracy, we could have increased the bounding box, but the model did not load and at this current time, we had beliefs that if the model did not load, then it would take too long to retrieve any data. The model has 2 main features: give a predictability score, classify the score as rainy or not rainy. Testing various regions around the United States had yielded positive results as areas such as the deserts in Arizona gave a low score of rain and a non rainy rating.

Initially the plan for the temperature predictor model was to give a range of hot, moderate and cold temperatures for the user to make their judgement on, given these classification for the predictions we opted to use a random forest classifier model to determine the classification of the day's temperature via likelihood. However given the scope of temperature we were working with we kept getting a moderate output for all days and locations. The only real reason we wanted to classify the temperature was to output a below freezing string that would alter the kind of precipitation that would fall i.e. either snow or rain. However, given that freezing temperature is below 32 degrees Fahrenheit that could easily be adjusted. So we switched over to a regression model to provide a numeric prediction averaged from the models decisions.


Classifier Feature Importance
0.0
0.1
0.2
0.3
0.4
0.5
lat
lon
month