Weather and Climate Predictions using Machine Learning Techniques

Ayush Sharma

*Abstract: The aim of this project was to successfully use machine learning techniques to predict weather and climate. Various python libraries were used to analyse and clean the data. Tensorflow was used to create an efficient machine learning model for future predictions.*

# Introduction

Weather and climate affect our daily lives in many ways. Bad weather can result in transport delays, accidents and also affect us emotionally. On the other hand, climate affects us economically and emotionally and has a larger impact than weather. Climate variations like monsoon seasons can affect the crop production, summers can cause record high temperatures which affects supply of water in various countries and so on. Predicting both and knowing how they will change can help us prepare for any trouble.

Climate and weather are same and different at the same time. Both of them are a measure of the atmosphere around us. But the difference between them is of the range of time period.

Weather is measured on daily basis most of the times and tells you about the variation in atmosphere of a day. Weather is commonly stated in terms of temperature, humidity, precipitation, cloudiness, brightness, visibility, wind, and atmospheric pressure [1]. Weather varies a lot, day-to-day or even every hour. Predicting it can help us prepare for any calamity that can happen in week or a month. Since weather also includes factors like sunshine, rain, cloudiness, wind speeds it can help us prepare for events like a thunderstorm, or a tornado and such dangerous events that can hugely impact people's lives.

Climate on the other hand, is the average of weather. Climate is described over a longer period of time than weather. Climate shows the pattern of change in weather and is normally described over years. Climate is generally stated in precipitation, hail storms, snow and other phenomena that occur over a long period of time [1]. Predicting climate can help us understand the major changes that are occurring around the world. For example, global warming can be predicted by climate to see how high the temperatures rise in the summer. These high temperatures lead to some negative changes like poor crop yield, shortage of water, rise in sea level and many other.

# Method

The data that was used in this project was from Global Historical Climatology Network. It is an integrated database of daily climate information from the stations across the globe [2]. It covers 180 countries and territories and almost 100000 stations in total. The data is given as values measured by each station. The stations are categorised by their coordinates and the countries they are located at. Each station contains numerical values of the date, maximum temperature on that date and minimum temperature on that day. They also contain the data of precipitation recorded and snowfall recorded. Most station also contain data of average temperature on the specific date.

The data for each station varies, as some station have not started recording data until 1990s while some have data as old as 1900s. They also contain a lot of NaN (Not a number) values as climate can vary on location. For example some countries might not record snow or rainfall which leads to NaN values.

The method used first analyses the data from the station chosen. Since a station with least number of NaN values will have the smallest gap between the data. After analysing, the data was cleaned using pandas library. Cleaning of data means to remove or interpolate the NaN values which causes gaps in the data. This will make our graphs smoother and will also make the machine learning model more precise, because if we have a lot of NaN values the model will get confused and give an output which will be like the gaps because of these values.

After getting a clean dataset we then used it to train our model. The model was trained separately on each variable (maximum temperature, minimum temperature, etc). The accuracy of the model was judged by the loss and by comparing the predicted values with real-time values. Since all the GHCN dataset stop around 2020, we have 3 years of real-time data to compare it with the predictions made by the model.

The first aim was to understand what the model can do, so daily predictions was also conducted to understand if the model was predicting a new value or just copying the values of previous day as an output.

The second objective was completed by predicting the climate a year in advance. All the GHCN daily data was converted to weekly and then an offset was set which corresponded to a year, 52 weeks, to get the predictions.

# Results

# Data Analysis

For this project the city chosen was Guwahati, Assam, India. Cities in India generally do not experience snowfall until you are at north or north-east of the country. Since Guwahati is in east India, we do not expect snowfall. So we only analysed and predicted the three factors: maximum temperature, minimum temperature and precipitation.

The dataset was directly imported to the code by using pandas and the url: 'https://www.hep.ucl.ac.uk/undergrad/0056/other/projects/ghcnd/ghcnd\_gsn/ IN003041800.dly’. Few functions were used to split the data into columns and then it was displayed in graphs to understand the data more easily.

For maximum temperature (TMAX), as you can see in the graph 1.0 the data starts around 1986 and goes on until 2020. But it is quite visible that data from 2016 to 2020 is missing a lot of values which can affect the prediction of our model.

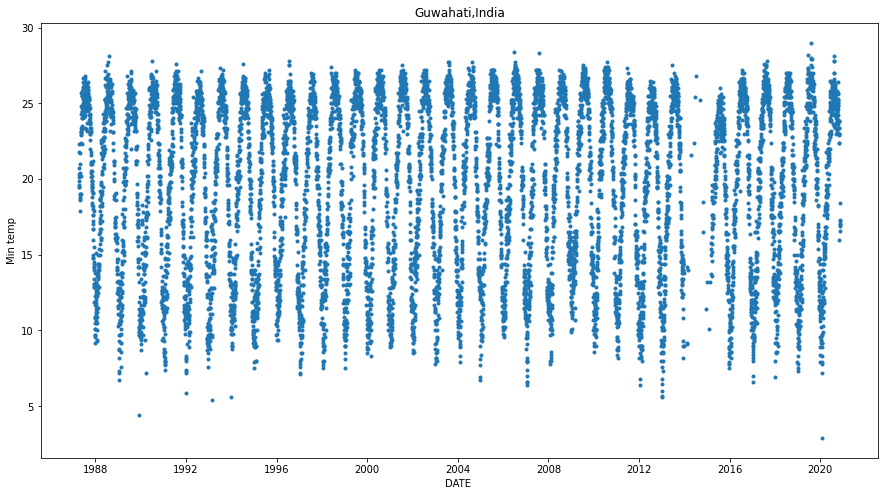
Chart, scatter chart

Description automatically generated with medium confidence

Graph 1.0: This the scatter plot of maximum temperature (°C) against the date in years.

We can also see that average maximum temperature lies around 30°C which is valid as East India does not experience extreme high temperatures.

For minimum temperature the graph is much better and has more values than maximum temperature. We can see through graph 1.1 that the data is more visible and better to understand.



Graph 1.1: The scatterplot of Minimum temperature against date in years

The data of minimum temperature (TMIN) has a better pattern than TMAX as you can see that the data after 2016 is consistent and is in a sine graph pattern. It is also evident that there is a gap between the years 2013-2015 which will also affect the predictions of our model. Also, we can see the highest minimum temperature reached is near 30°C which agrees with the average maximum temperature. The average minimum temperature is around 18/20°C which is acceptable.

The last graph we understood was the precipitation graph before starting to clean the data. As seen in the graph 1.2 precipitation (PRCP) data has the least number of gaps. Note that NaN values in the dataset is not the same as 0. NaN values can not be plotted on the graph while 0 is plotted. There are also constant peaks in the graph suggesting that it rains heavily few months and then it decreases. We can also make an educated assumption that the rainfall has started to decrease after the 1990s in Assam as around that time PRCP peaked above 3000mm for the last time. This happened because of decrease in evapotranspiration and warming of the Indian Ocean [3].

Chart, histogram

Description automatically generated

Graph 1.2: Precipitation (x10mm) against date in years

# Machine Learning

Before making the model and testing it for predictions the data was cleaned. Using pandas library, the NaN values present in the data were removed making all the dataset continuous. After cleaning the data was split into two sets, training set and test set. It is as the names suggests the model will be trained using training set and the predictions will be calculated using the test set.

For this type of data, we will be using LSTM network. LSTM (long- short term memory) networks are designed to handle the RNN that has vanishing gradients. RNN (Recurrent neural network) are used to handle problems with a sequential data just like the weather data used. The main feature of RNNs is that they pass information from one step of the sequence to the next through a hidden layer that acts as a memory of the past inputs. This helps RNNs to learn patterns in the data that vary over time.

The model created for our first aim which was to predict the weather a day in future, consist of 2 LSTM layers and 3 dense layers with the last layer only having 1 neuron as we only want 1D output. The graphs of factor against the sample can be seen below for each of the factors. All the values were normalised against the maximum value for each of the variable. Normalisation helped with reducing the loss of the model.

As we can see the predictions of TMIN were most accurate and the model has been able to understand the pattern reliably. On the other hand, the predictions for TMAX are not as accurate and do not have as high peaks as the true values. The valid reason for such behaviour is the gap of 2016-2020 of NaN values that was removed.

For PRCP the peaks are way smaller than the true values this can happen because of the number of 0s present in each year. As this is daily data and rainfall in India is seasonal, the daily data contains a lot of 0 values for PRCP causing the model to average out the predictions accordingly.

The 1-day advance predictions for the factors were as follows:

TMAX: 24°C on 14th November 2020

TMIN: 15°C on 14th November 2020

PRCP: 113 mm on 14th November 2020

These predictions are not completely accurate as the temperature readings are ±4°C while there was no rain on that day.

Chart

Description automatically generated

Graph 2.0: Normalised TMAX predictions against the sample on the same graph as the true values for comparison

A picture containing text, different, several, flag

Description automatically generated

Graph 2.1: Normalised TMIN predictions against the sample on the same graph as the true values for comparison

Chart

Description automatically generated

Graph 2.2: Normalised PRCP predictions against the sample on the same graph as the true values for comparison

This suggests that the effect of the NaN values on the predictions were quite high giving them greater uncertainty.

For the second objective we used the same model and method but changed the time sequence to weekly instead of daily. As seen in the graphs below weekly TMIN has a better prediction compared to daily while the weekly PRCP has higher peaks but still seems to be inaccurate.

Chart, line chart

Description automatically generated

Graph 3.0: weekly normalised TMIN against sample

Chart, line chart

Description automatically generated

Graph 3.1: weekly normalised PRCP against sample

# Conclusion

The predictions of the model depend on a lot of factors. The graphs suggest that a few improvements can be made to get the predictions closer to the true values.

As we can see for variables which change daily like temperature, using daily data will yield higher accuracy while for factors like precipitation that affect the climate should be made over a longer period preferably monthly or yearly basis.

To improve the output first choose a better file to analyse with low numbers of empty values. Use a bigger model which might decrease the efficiency by taking longer time but will give a better output. Use more LSTM layers. Overall the results agree with true values and suggest that machine learning can be used for various other predictions which are time-dependent like weather and climate.

# References

|  |  |
| --- | --- |
| [1] | NASA, “Climate,” 2017. [Online]. Available: https://www.nasa.gov/mission\_pages/noaa-n/climate/climate\_weather.html. |
| [2] | NCEI. [Online]. Available: https://www.ncei.noaa.gov/products/land-based-station/global-historical-climatology-network-daily. |
| [3] | Mongbay. [Online]. Available: https://news.mongabay.com/2016/09/deforestation-reducing-monsoon-rainfall-in-india-new-study/#:~:text=Ghosh%20and%20his%20colleagues%20found,in%20north%20and%20northeast%20India.. |