Weather Prediction

Project Report submitted in partial

fulfillment for the award of the

degree of

BACHELOR OF TECHNOLOGY

Submitted by

KAVYA SINGH

(18SCSE1010609/18021011834)

DEEPANSHU SINGH

(18SCSE1010538/18021011765)

IN

COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Under the Supervision of

NITIN MISHRA

PROFESSOR

SCHOOL OF COMPUTING AND SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this project report **"WEATHER PREDICTION "**

Is the bonafide work of **KAVYA SINGH(18021011834) DEEPANSHU SINGH(18021011765) ."** who carried out the project work under my supervision.

SIGNATURE OF DEAN

MANISH SABHARWAL

Dean of School

COMPUTER SCIENCE AND ENGINEERING

SIGNATURE OF SUPERVISOR

NITIN MISHRA

SUPERVISOR

PROFESSOR

COMPUTER SCIENCE AND ENGINEERING

Statement of Project Report Preparation

1. Thesis title: WEATHER PREDICTION
2. Degree for which the report is submitted: ONGOING
3. Project Supervisor was referred to for preparing the report.
4. Specifications regarding thesis format have been closely followed.
5. The contents of the thesis have been organized based on the guidelines.
6. The report has been prepared without resorting to plagiarism.
7. All sources used have been cited appropriately.
8. The report has not been submitted elsewhere for a degree.

(Signature of the student)

DEEPANSHU SINGH

KAVYA SINGH

ACKNOWLEDGEMENT

We would like to express my special thanks of gratitude to my Mentor (**Dr. Nitin Mishra)** who gave me the golden opportunity to do this wonderful project on the topic (**Nature Analysis-Weather Prediction**), which also helped me in doing a lot of Research and we came to know about so many new things we am really thankful to them.

Secondly we would also like to thank my parents and friends who helped us a lot in finalizing this project within the limited time frame.

Name :Kavya Singh(18021011834)

Deepanshu Singh(18021011765)

ABSTRACT

Due to various reasons we had hampered the growth of the nature and affected it badly.The working of nature is very unique.

One doesn’t know what would happen and when would happen.We have to learn to respect the nature.Therefore,in practice to understand the nature we had tried to implement the project of temperature prediction.

With the help of this project one can understand the conditions going to be prevailed in future and be prepared for the future.

We had tried to implement our best at our own level.

TABLE OF CONTENT

List of Abbreviations

Diff -----different

HTML--- Hypertext Markup language

AI ---- Artificial Intelligence

URL ----- Uniform Resource Locater

RAM Random Access Memory

Chapter-1

Introduction

1.1 Purpose:

The basic purpose of this project is have the reading from the outer environment to predict which would help us to lead to particular observation regarding the environment.

After the temperature is being predicted we can use the result in more precise form to use the result to predicts various different result using higher intelligence tools and also can create weather reports,dashboards,compare the previous weather records,provide the reason for the change in the environment.

The basic aim of our project is to make realize others that our life and our habits all things are affected by nature.Nature is the very critical gift given by god to understand.Sometimes it loves and affection we can see in the form of rain, snow and sometimes the angry side of nature we see in the form of any natural disaster.

The main objective is to make the other realize that we should understand the nature and work according to him so that in future we may not observe any kind of disaster which would take the lives of millions of people.

1.2 Problem statement:

In our previous phases the accuracy of our machine learning model was not such accurate to be used further.So after that we created a model whose accuracy is approx to the actual values.

Our objective was to have a model with a such accuracy and less error rate.With this we were also focused to display our results after calculation on a web framework.

As we were using flask and there is no storage system to store the results so this is main challenge now for us to store the results and present it in the form of a dashboard report and active report to make our main aim successful.

But the main problem towards us is that we are not able to provide connectivity of a database to flask application which might take some time.

But our Main aim of generating the results on a web framework has been achieved with a good success.

1.3 Motivation and Perspective:

The motivation behind our project is the nature only which had motivated us to bring out this concept to implement.

We thought of and project/idea where we can have an alert messaging system for that areas where our model could predict the upcoming of any natural disaster.

The motivation to make the people safe from any natural disaster heading towards them had made us to think for this kind of idea.

Here we were focusssed mainly on temperature as temperature is to predict any kind of environment around us depending upon the location.

After the temperature is being predicted we can have various fields that can be

measured with the help of temperature like vegetation growth over an area,the cultivation activity,soil erosion ,air pollution activities,a particular disease growing activity.

1.4 Applications:

Temperature prediction can have various applications at the higher as well as small levels.

Tourist:

1. This would help us to guide the tourist to know what the weather is, how it would be in further time so that they can plan their journey according to that.

Safety:

1. This would also help us to get safety of people on roads when there is heavy rainfall, it would guide them how we can drive the car so that the life of people could be saved.

2. Weather predictions are important for the road safety specially on hilly areas, so in case of heavy rainfall or snowfall the weather predictions could be very helpful in travelling plans and making plans too.

Vegetation analysis and cultivation:

1. With the help of this one can know what type of vegetation is being found on a particular area and what types of crops can be grown in order to get profit.

Disease prediction and analysis:

1. With the help of this project we are able to predict what kind of disease in what season upto which extent.

Pollution activity:

1. we can keep a check on pollution with the help of this project.

Natural disaster warning system:

1. We can build a system to warn any area over coming of a natural disaster like flood,droughtetc.

1.5 Objectives:

Phase-1:

- the main aim to build a model to just understand the concept of working of a model.
- Here we tried to understand the requirements need to build a model.
- We tried to implement the Logistics regressions to just check check the accuracy without making changes in dataset.
- We had tried to seen the working of the environment while building the model.
- We implemented the training and testing of the datasets.

PHASE-2:

- Our main objective was to understand the relations between the attributes to predict anything from a dataset.
- We tried to corelate ecah attribute with one another to under the importance of each variable or attribute.
- We had tried to implement the Visulaization ofheatmaps to understand the corealtion graphically.
- We implemented the naive bayes algorithm to increase the efficiency of our Model.
- After using regression technique we tried to use the classification technique to increasethe efficiency of the Model.

PHASE-3:

- This was our final phase where we tried to sum up the total work and tried to encapsulate it.
- We here tried to first of all make a model with more accuracy rate.
- We here tried to build a web page to display our results with a very flexible framework.
- As we were running out of time we tried to sum up the work as fast as we can and tried to use a flexible and easy framework to use.
- We made connectivity of a model with the web application to showcase our results.

CHAPTER-2

SUMMARY OF ALL PHASES

2.1 PHASE-1

2.1.1 AIM AND OBJECTIVE:

- The main aim to build a model to just understand the concept of working of a model.
- Here we tried to understand the requirements need to build a model.
- We tried to implement the Logistics regressions to just check check the accuracywithout making changes in dataset.
- We had tried to seen the working of the environment while building the model.
- We implemented the training and testing of the datasets.

2.1.2 Problem statement:

We were given a datset to predict the temperature,by building a machine learning model whose accuracy may stood upto the mark and can be very efficient in its own work.

2.1.3 Uses:

1-This project may help Us Analyzing The vegetation of a particular area.

2-This project Would help to Analyze the weather conditions change over

an Area. 3-This would help us to predict the Natural Calamities.

4-This project would also play a major role in Disaster Management.

2.1.4 OVERALL DESCRIPTION

The System has been mainly categorized into 4 groups based on our requirements:

1- Loading stage

2- Manipulations and Transformations

3- Building a Model

4- Creating reports and charts(Statistical Analysis)

Loading Stage

In this our data would be being collected from different data sources to build our model.The data we here used id mainly taken from kaggle but later on on increasing the platform the data would be collected from the sensors such as Rasberry pie etc.

Here the data first be loaded into data warehouse and then after ETL process the would loaded into our servers for the further uses.

Manipulations and Transformations

Still the data is being loaded into the warehouse but now also we need to rectify it according to our needs so we would apply some sort of algorithms to manipulate and transform the data according to our needs.

This would helps us to get the perfect data which we need for the further predictions and building a model.

Building a Model

Building a model means to train a machine on some sort of algorithms to predict the output and have good efficiency with the output data that is being given in the dataset.

The outputs of prediction and feature engineering are a set of label times, historical examples of what we want to predict, and features, predictor variables used to train a model to predict the label.

The process of modeling means training a machine learning algorithm to predict the labels from the features, tuning it for the business need, and validating it on holdout data.

Creating reports and charts(Statistical Analysis)

When we had created the model we will be getting the historical data and the current data,therefore we need some sort of visualizations tools to visualize the the data so that we can measure the accuracy and efficiency of the model.
This would also helps other people to understand our model ,to analyze the results and also would help them to interpret the the results from the data very easily.
There are various kinds of visualizations in python available which we can use the interpret our results.

SOME WORKING OUTPUTS OF PHASE-1:

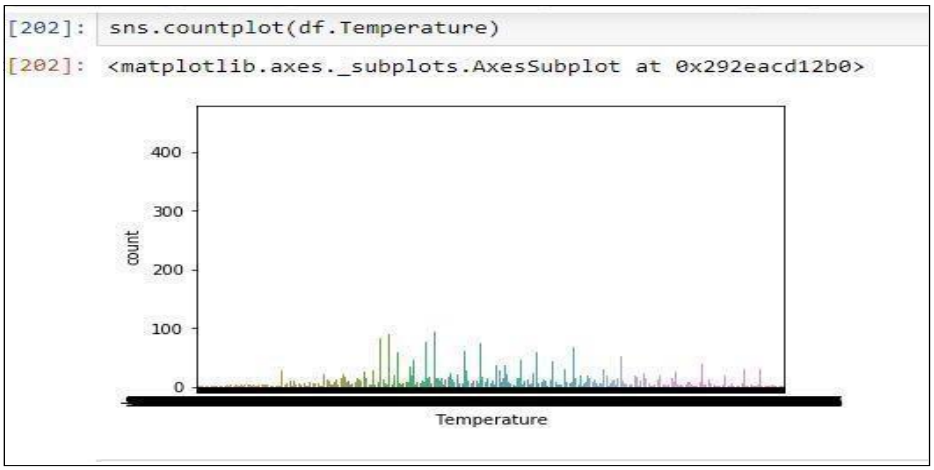

Fig-1

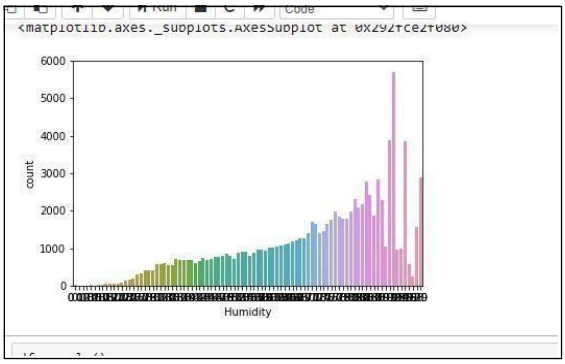

Fig-2

```
In [213]: model=LogisticRegression()

In [214]: model.fit(x_train,y_train)
          prediction=model.predict(x_train)
          score=accuracy_score(y_train,prediction)
          print(score)

          F:\hello\lib\site-packages\sklearn\linear_model\logistic.py:433: FutureWarning: Default
          22. Specify a solver to silence this warning.
            FutureWarning)

          --------------------------------------------------------------------------
```

Fig-3

2.2 PHASE-2

2.2.1 Overview:

Weather Prediction is one of the major aspect that every country wants to achieve.They want to know each and everything about weather which might occur in future.

It is not necessary that in race of being first developed countries they want to achieve this aspect but they might do it for the welfare of the people.

For example:

They could do it for the management of natural disasters,they might do it for promoting the tourism in their country,they might want to increase the vegetation of the particular area so that,the area could be grown with good vegetation to reduce the conditions like drought,soil erosion.

2.2.2 Objective:

- Our main objective was to understand the relations between the attributes to predict anything from a dataset.
- We tried to corelate ecah attribute with one another to under the importance of each variable or attribute.
- We had tried to implement the Visualization ofheatmaps to understand the corealtion graphically.
- We implemented the naive bayes algorithm to increase the efficiency of our Model.

2.2.3 Problem Statement:

After the model was built with the logistics regressions the accuracy could not achieve the value of 50% so,we need to understand the dataset,therefore we tried to analyze the attributes in the dataset to make our confusion more clearer.

We tried to clean data here and tried to improve the quality of the data.We here used the visualizations techniques to understand the corelation of the attributes within each other.

We also tried to use classification technique to improve the accuracy of the model.

2.2.4 OVERALL DESCRIPTION:

About Heatmaps:

A heatmap is a visualization tool that helps us to analyze the data and its attributes through different color variations in it.

A heat map basically helps us to know which areas need to be paid attention and which to be not so that the user must focus the most important part of the dataset.

With the help of different color variations in heatmaps we can visualize which part of the data is necessary for us to consider and which not.

A heatmap uses variation of color to show the intensity of importance the data present in our dataset.

It also is useful to find the correlation between 2 attributes of a dataset.It tells how that 2 attributes are connected to each other ,it tells what are the factors which bind the both attributes.

NAIVE BAYESALGORITHM:

It is the type of classification technique in which we use labeled data to predict out the probability of the output using the Most Famous Algorithm of mathematics that is Conditional Probability and Bayes Theorem.

Naive Bayes model is very easy to build and mostly used when we have usually large dataset.

It mostly uses the categorical data to predict the output.

Bayes theorem helps us to calculate posterior probability P(c|x) from P(c), P(x) and P(x|c).

The process of modeling means training a machine learning algorithm to predict the labels from the features, tuning it for the business need, and validating it on holdout data.

SOME WORKING OUTPUTS OF PHASE-2:

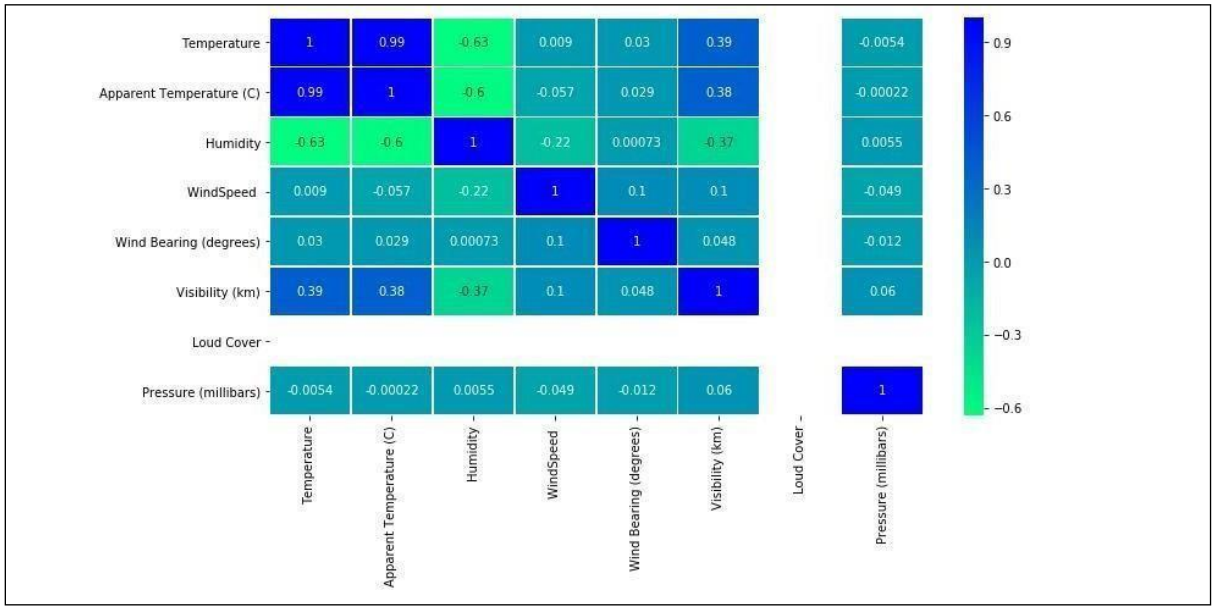

Fig-4

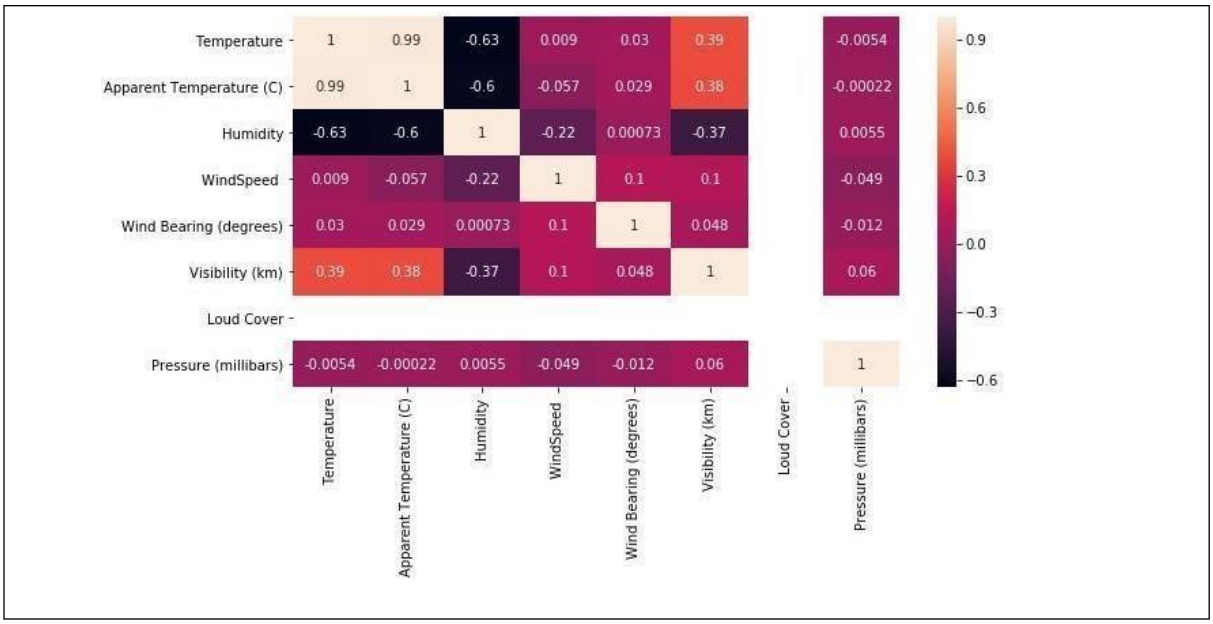
Temperature
Apparent Temperature (C)
Humidity
WindSpeed
Wind Bearing (degrees)
Visibility (km)
Loud Cover
Pressure (millibars)
1 0.99 -0.63 0.009 0.03 0.39 -0.0054
0.99 1 -0.6 -0.057 0.029 0.38 -0.00022
-0.63 -0.6 1 -0.22 0.00073 -0.37 0.0055
0.009 -0.057 -0.22 1 0.1 0.1 -0.049
0.03 0.029 0.00073 0.1 1 0.048 -0.012
0.39 0.38 -0.37 0.1 0.048 1 0.06
-0.0054 -0.00022 0.0055 -0.049 -0.012 0.06 1
0.9
0.6
0.3
0.0
−0.3
−0.6

Fig-5

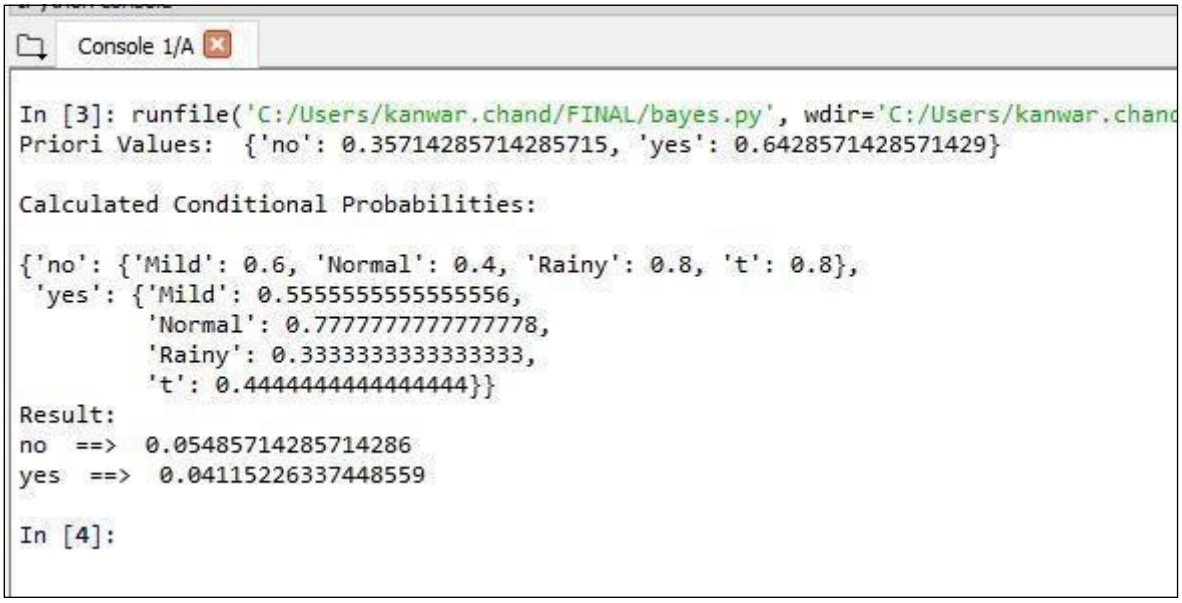
Console 1/A
In [3]: runfile('C:/Users/kanwar.chand/FINAL/bayes.py', wdir='C:/Users/kanwar.chand
Priori Values: {'no': 0.35714285714285715, 'yes': 0.6428571428571429}
Calculated Conditional Probabilities:
{'no': {'Mild': 0.6, 'Normal': 0.4, 'Rainy': 0.8, 't': 0.8},
'yes': {'Mild': 0.5555555555555556,
'Normal': 0.7777777777777778,
'Rainy': 0.3333333333333333,
't': 0.4444444444444444}}
Result:
no ==> 0.05485714285714286
yes ==> 0.04115226337448559
In [4]:

Fig-6

2.3 PHASE-3

2.3.1 Objective:

Here our main objective was to predict the temperature with the most accurate accuracyand display the conditions of the weather with respect to temperature on a web page by using web framework.

2.3.2 AIM:

- This was our final phase where we tried to sum up the total work and tried to encapsulate it.
- We here tried to first of all make a model with more accuracy rate.
- We here tried to build a web page to display our results with a very flexible framework.
- As we were running out of time we tried to sum up the work as fast as we can and tried to use a flexible and easy framework to use.
- We made connectivity of a model with the web application to showcase our results.

2.3.3 WORKING SUMMARY OF THE PROJECT:

1- First of all we had created the machine learning model to have an proper accuracy and less error rate.

2- After creating the model,we had dumped it into the pickle file for serializing for flask.

3- After the dumping the target was to where to use the flask application,so for that we created a web page that contains a form to accept the parameter and predict the results.

4- Now the web page was created then we created the flask file and then connected the flask file with the html web page by passing address of the page in form method.

5- Now we connected both and then for that we had to make a proper arrangement of the files and folder to run it.

6- Now using command prompt we used to RUN the application.

7- In command prompt when we ran the python file using **Python app.py**(flask file) then we prompted to use a url to reach to a apllication.

Working outputs of the PHASE-3:

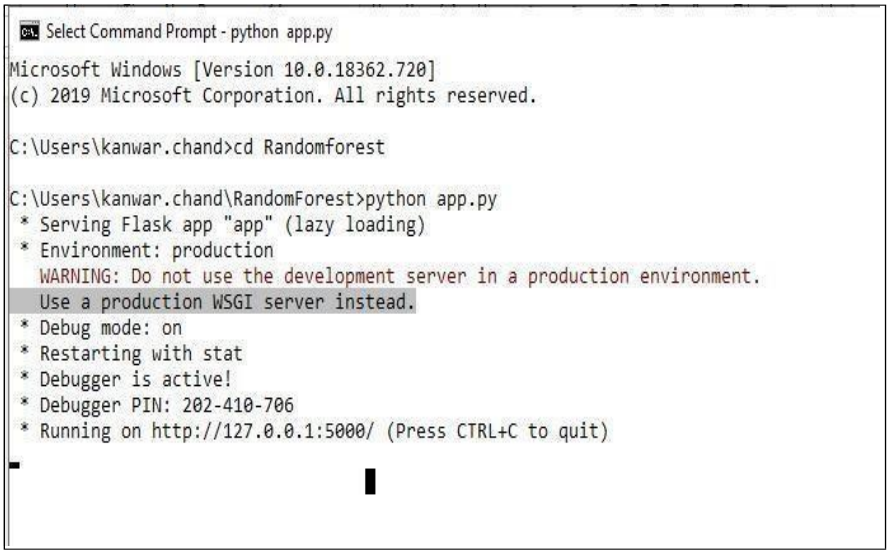

Fig-7

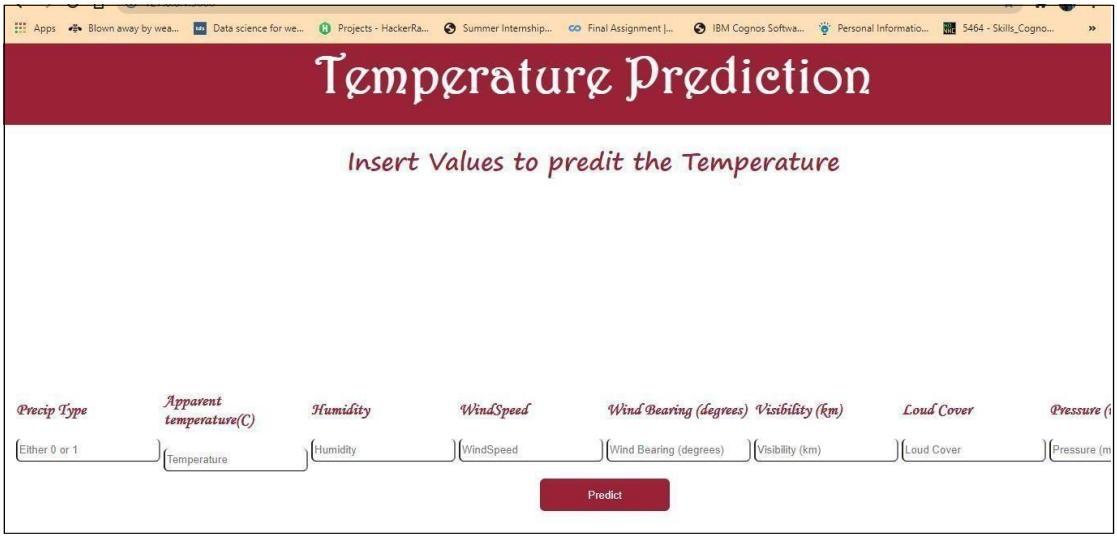

Fig-8

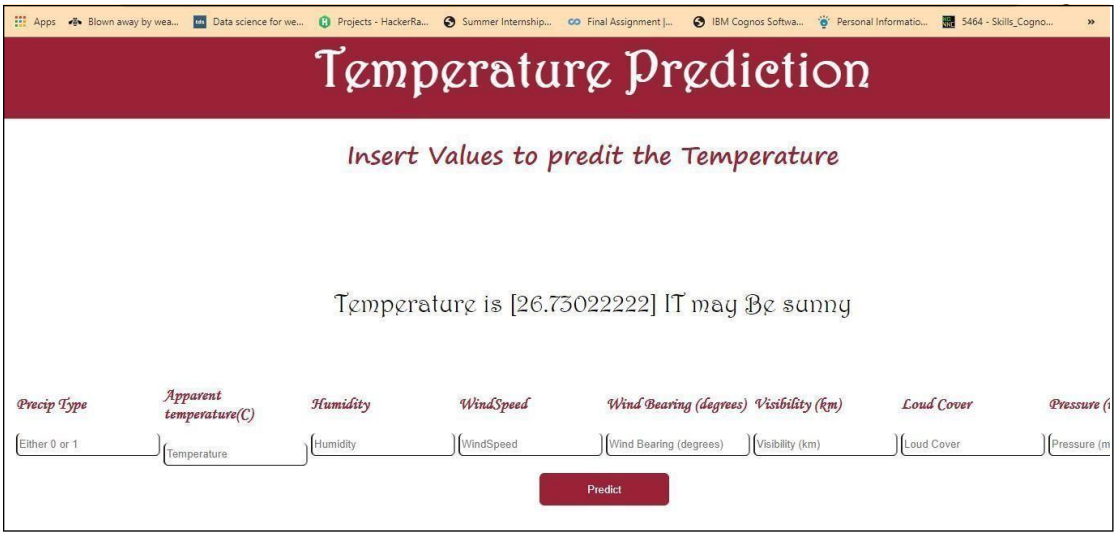
Apps
Blown away by wea...
Data science for we...
Projects - HackerRa...
Summer Internship...
Final Assignment |...
IBM Cognos Softwa...
Personal Informatio...
5464 - Skills_Cogno...
Temperature Prediction
Insert Values to predit the Temperature
Temperature is [26.73022222] IT may Be sunny
Precip Type
Apparent temperature(C)
Humidity
WindSpeed
Wind Bearing (degrees)
Visibility (km)
Loud Cover
Pressure (
Either 0 or 1
Temperature
Humidity
WindSpeed
Wind Bearing (degrees)
Visibility (km)
Loud Cover
Pressure (m
Predict

Fig-9

CHAPTER 3

Description of Phase 2

3.1 Objective:

Here our main objective was to predict the temperature with the most accurate accuracy

3.2 Our Findings of phase-II:

Our main findings in this phase was:

I- A visualiztion for understanding the relationships between Attributes.

II- Using classification techniques to build a model(Naive bayes).

3.3 Explainations :

3.4 Heatmaps

A **heat map** (or **heatmap**) is a data visualization technique that shows magnitude of a phenomenon as color in two dimensions. The variation in color may be by hue or intensity, giving obvious visual cues to the reader about how the phenomenon is clustered or varies over space. There are two fundamentally different categories of heat maps: the cluster heat map and the spatial heat map. In a cluster heat map, magnitudes are laid out into a matrix of fixed cell size whose rows and columns are discrete phenomena and categories, and the sorting of rows and columns is intentional and somewhat arbitrary, with the goal of suggesting clusters or portraying them as discovered via statistical analysis. The size of the cell is arbitrary but large enough to be clearly visible. By contrast, the position of a

magnitude in a spatial heat map is forced by the location of the magnitude in that space, and there is no notion of cells; the phenomenon is considered to varycontinuously.

Heat maps originated in 2D displays of the values in a data matrix. Larger values were represented by small dark gray or black squares (pixels) and smaller values by lighter squares. Loua (1873) used a shading matrix to visualize social statistics across the districts of Paris. Sneath (1957) displayed the results of a cluster analysis by permuting the rows and the columns of a matrix to place similar values near each other according to the clustering. Jacques Bertin used a similar representation to display data that conformed to a Guttman scale. The idea for joining cluster trees to the rows and columns of the data matrix originated with Robert Ling in 1973. Ling used overstruck printer characters to represent different shades of gray, one character-width per pixel. Leland Wilkinson developed the first computer program in 1994 (SYSTAT) to produce cluster heat maps with high-resolution color graphics. The Eisen et al. display shown in the figure is a replication of the earlier SYSTAT design.

Software designer Cormac Kinney trademarked the term 'heat map' in 1991 to describe a 2D display depicting financial market information. The company that acquired Kinney's invention in 2003 unintentionally allowed the trademark to lapse.

Advantages of Heatmaps:

1. Heatmaps give an *instant* overview of **key web performance parameters.**

2. Heatmaps provide a ***visual* approach** to understanding numeric values.

3. Heatmaps make it ***easier to learn*** from users to create smarter web design.

4. Heatmaps ***complement other tools* in your analyticaltoolkit.**

5. Even when heatmaps are **"confusing" or dispersed,** they point to interesting issuesand shouldn't be dismissed.

6. Heatmaps help you *understand* your visitors better and ultimately *give* ***them a better experience.***

7. Heatmaps help companies make informed choices that **improve the bottom line.**

Benefits of Heatmaps

Analytics tools like Google Analytics or Site Catalyst are great at providing metrics to show which pages users visit but they can lack detail when it comes to understanding how users engage with those pages. Heatmaps can give a more comprehensive overview of how users are really behaving.

Heatmaps are also a lot more visual than standard analytics reports, which can make them easier to analyse at a glance. This makes them more accessible, particularly to people who are not accustomed to analysing large amounts of data.

Good heatmapping tools, such as CrazyEgg and Clicktale, enable analysts to segment and filter the data. This means that it can be easy to see how different types of users are engaging with a particular page.

Considerations

Like most forms of web[iii] analytics, heatmaps need to have a large amount of data before they can be accurately analysed. Analysing heatmaps based on a small amount of data is similar to 'calling' A/B[iii] tests too early, based on too few visits or conversions. As heatmaps show trends, it is important to have enough information to ensure that any anomalies do not affect the overall heatmap picture.

When used incorrectly, heatmaps can be misleading. They can encourage analysts to make assumptions that may not be correct. It is important to remember that, as with other forms of quantitative data, heatmaps can tell you what has happened on a page, but cannot tell you why that happened.

It is also important to be aware of the limitations of heatmaps. Often, heatmaps will show clicks on a page (known as click maps) but this may only be a part of the story. For example, when looking at a heatmap of a form, it may show that users are clicking on the first field and that there are fewer clicks on the subsequent fields.

This could suggest that users are dropping out of the process after filling out the first field. What heatmaps do not show, however, is if users have used their keyboard to tab through form fields, rather than their mouse. To correctly analyse form usage, dedicated form analytics tools can be used. These will measure the time spent within each field, rather than just clicks.

Using Heatmaps:

The Case Against Heatmaps

Heatmaps can be very pretty, but this post from the Displayr blog explains why users can have trouble interpreting them because of human visual limitations.

A Heatmap of Actual Heat

On the other hand, heatmaps and similar graphics can be effective for showing trends over time. One example is this large chart of global average temperature from Climate Central.

Ineffective Graphics for a Laugh

Of course, there's also this approach, in which a completely illegible heatmap is presented simply because it "looks like a hipster's overshirt."

Effective Graphics Demonstrate Clear Conclusions

At the other extreme, heatmaps that are both useful and attractive appear in this Wall Street Journal feature from 2015. Each chart shows the number of cases of an infectious disease and the date that a vaccine was introduced. These charts are also interactive, in that you can hover over them for specific information.

USING HEATMAPS TO FIND CORELATION BETWEEN ATTRIBUTES

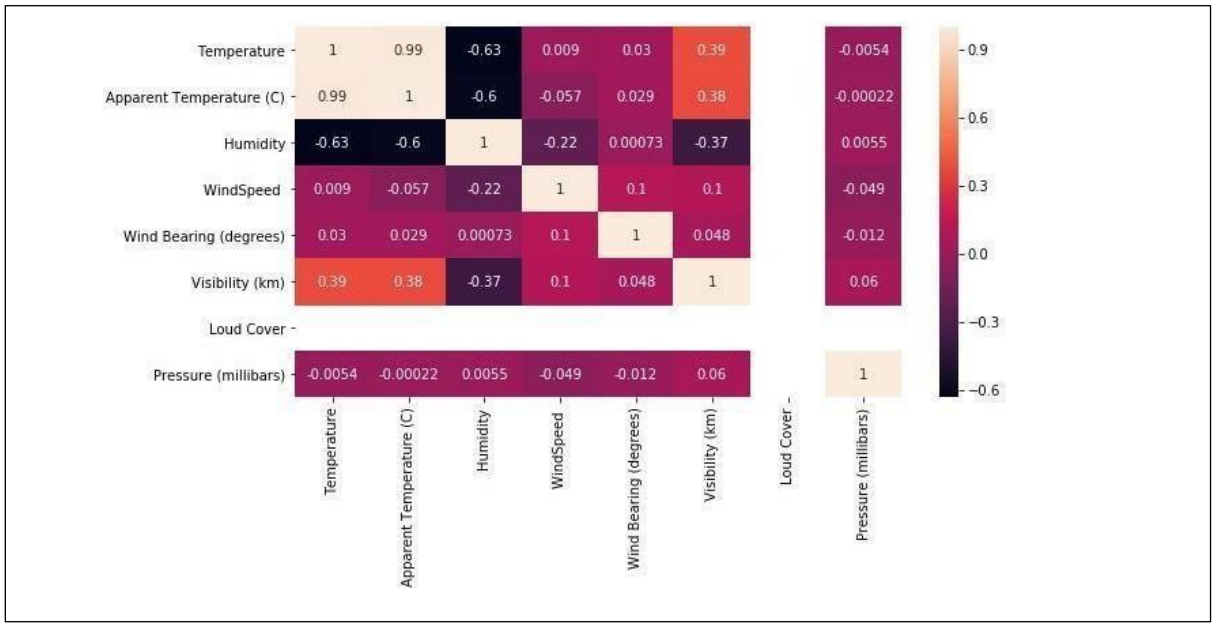

Fig-10

Heatmaps:

A heatmap is a visualization tool that helps us to analyze the data and its attributes through different color variations in it.

A heat map basically helps us to know which areas need to be paid attention and which to be not so that the user must focus the most important part of the dataset.

With the help of different color variations in heatmaps we can visualize which part of the data is necessary for us to consider and which not.

A heatmap uses variation of color to show the intensity of importance the data present in our dataset.

It also is useful to find the correlation between 2 attributes of a dataset.It tells how that 2 attributes are connected to each other ,it tells what are the factors which bind the both attributes.

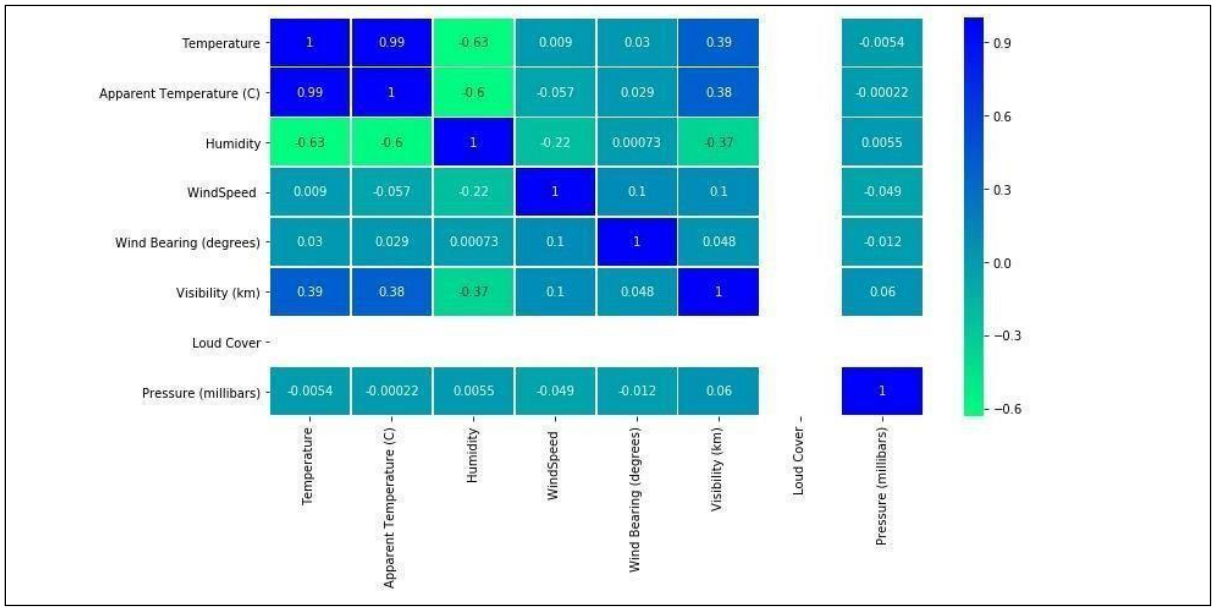

Fig-11

```
In [4]: import seaborn as sns
        import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt

        %matplotlib inline

In [5]: df=pd.read_csv('F:/New folder/weather-dataset/weatherHistory.csv')

In [6]: df.head()
```

Out[6]:

	Formatted Date	Summary	Precip Type	Temperature	Apparent Temperature (C)	Humidity	Wind Speed	Wind Bearing (degrees)	Visibility (km)	Loud Cover	Pressure (millibars)	Daily Summary
0	2006-04-01 00:00:00.000 +0200	Partly Cloudy	rain	9.472222	7.388889	0.89	14.1197	251	15.8263	0	1015.13	Partly cloudy throughout the day.
1	2006-04-01 01:00:00.000 +0200	Partly Cloudy	rain	9.355556	7.227778	0.86	14.2646	259	15.8263	0	1015.63	Partly cloudy throughout the day.
2	2006-04-01 02:00:00.000 +0200	Mostly Cloudy	rain	9.377778	9.377778	0.89	3.9284	204	14.9569	0	1015.94	Partly cloudy throughout the day.
3	2006-04-01 03:00:00.000 +0200	Partly Cloudy	rain	8.288889	5.944444	0.83	14.1036	269	15.8263	0	1016.41	Partly cloudy throughout the day.
4	2006-04-01 04:00:00.000 +0200	Mostly Cloudy	rain	8.755556	6.977778	0.83	11.0446	259	15.8263	0	1016.51	Partly cloudy throughout the day.

```
In [7]: df.dtypes
```

Fig-12

```
%matplotlib inline
df=pd.read_csv('F:/New folder/weather-dataset/weatherHistory.csv')
import matplotlib.pyplot as plt
plt.figure(figsize=(12,6))
sns.heatmap(df.corr(),annot=True,linewidth=0.5,cmap='winter_r')
```

Out[11]: <matplotlib.axes._subplots.AxesSubplot at 0x294fb58b4e0>

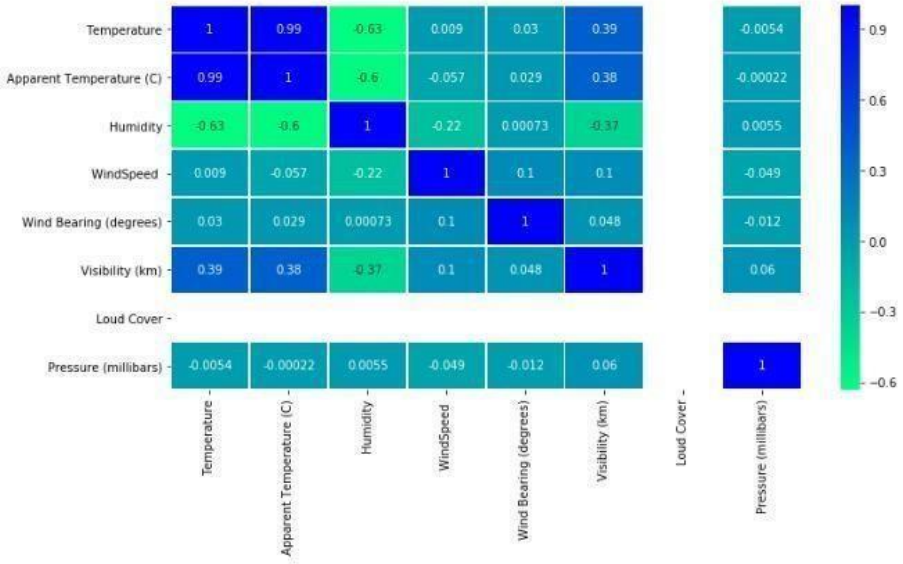

Fig-13.

```
In [12]: import seaborn as sns
import pandas as pd
import numpy as np
%matplotlib inline
df=pd.read_csv('F:/New folder/weather-dataset/weatherHistory.csv')
import matplotlib.pyplot as plt
plt.figure(figsize=(11,5))
sns.heatmap(df.corr(),annot=True,linewidth=0.5,cmap='RdBu')
```

Out[12]: <matplotlib.axes._subplots.AxesSubplot at 0x294f8ff9710>

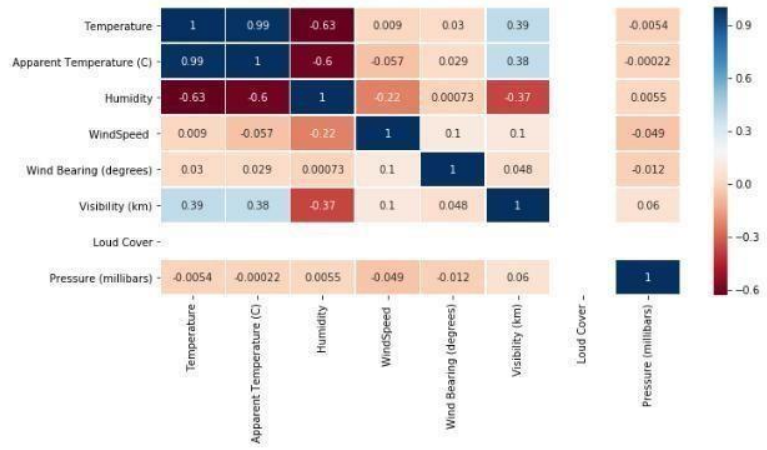

Fig-14

3.5 NAIVE BAYES ALGO IMPLEMENTATION:

NAIVE BAYES IMPLEMENTATIONS

It is the type of classification technique in which we use labeled data to predict out the probability of the output using the Most Famous Algorithm of mathematics that is Conditional Probability and Bayes Theorem.

Naive Bayes model is very easy to build and mostly used when we have usually large dataset.

It mostly uses the categorical data to predict the output.

Bayes theorem helps us to calculate posterior probability P(c|x) from P(c), P(x) and P(x|c).

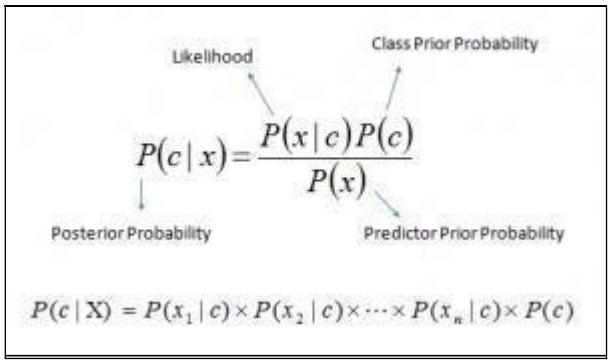

Fig-15

Probabilistic model

Abstractly, naïve Bayes is a **conditional probability** model: given a problem instance to be classified, represented by a vector representing some *n* features (independent variables), it assigns to this instance probabilities

for each of *K* possible outcomes or *classes* .

The problem with the above formulation is that if the number of features *n* is large or if a feature can take on a large number of values, then basing such a model on probability

tables is infeasible. We therefore reformulate the model to make it more tractable. Using Bayes' theorem, the conditional probability can be decomposed as

In plain English, using Bayesian probability terminology, the above equation can be written as

In practice, there is interest only in the numerator of that fraction, because the denominator does not depend on and the values of the features are given, so that the denominator is effectively constant. The numerator is equivalent to the joint probability model

$$p(C_k, x_1, \ldots, x_n)$$

which can be rewritten as follows, using the chain rule for repeated applications of the definition of conditional probability:

$$\begin{aligned} p(C_k, x_1, \ldots, x_n) &= p(x_1, \ldots, x_n, C_k) \\ &= p(x_1 \mid x_2, \ldots, x_n, C_k)\, p(x_2, \ldots, x_n, C_k) \\ &= p(x_1 \mid x_2, \ldots, x_n, C_k)\, p(x_2 \mid x_3, \ldots, x_n, C_k)\, p(x_3, \ldots, x_n, C_k) \\ &= \cdots \\ &= p(x_1 \mid x_2, \ldots, x_n, C_k)\, p(x_2 \mid x_3, \ldots, x_n, C_k) \cdots p(x_{n-1} \mid x_n, C_k)\, p(x_n \mid C_k)\, p(C_k) \end{aligned}$$

Now the "naïve" conditional independence assumptions come into play: assume that all features in are mutually independent, conditional on the category . Under this assumption,

$$p(x_i \mid x_{i+1}, \ldots, x_n, C_k) = p(x_i \mid C_k) .$$

Thus, the joint model can be expressed as

$$\begin{aligned} p(C_k \mid x_1, \ldots, x_n) &\propto p(C_k, x_1, \ldots, x_n) \\ &\propto p(C_k)\, p(x_1 \mid C_k)\, p(x_2 \mid C_k)\, p(x_3 \mid C_k) \cdots \\ &\propto p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k), \end{aligned}$$

where denotes proportionality.

This means that under the above independence assumptions, the conditional distribution over the class variable is:

$$p(C_k \mid x_1, \ldots, x_n) = \frac{1}{Z} p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)$$

where the evidence is a scaling factor dependent only on , that is, a constant if the values of the feature variables are known.

Advantages:

- It is speedy and easy to build and predict.
- It can predict output for multiple classes or labels therefore also called multiclass prediction.
- We need less training data in this to perform the model training.
- It perform better in case of categorical input variables compared to numerical variables.
- In this we assume that the attributes have no relation with each other which makes it more better and efficient than other algorithms.

Disadvantages:

- Here if the dataset wil have null values then the the data would be 0 assumed by the model and our efficiency could be reduced.
- The probability we get from this algorithm is not much accurate so in real life we donot consider it,for that we need to normalize it.
- The law of independence in our life is not suitable because every attribute relies on other attributes to derive the set of results for the given set of problem.

Used cases of Naive Bayes Algorithms in real world life:

- **Dynamic time Analysis:** Naive Bayes is a algorithm that is very much speedy to calculate the results therfore can be very fast to have dynamic and real time analysis for the given problem.
- **Multiclass Analysis:**When we have more then 2 target variables in our datset then we have the Naive bayes algorithm to help us out in that case.

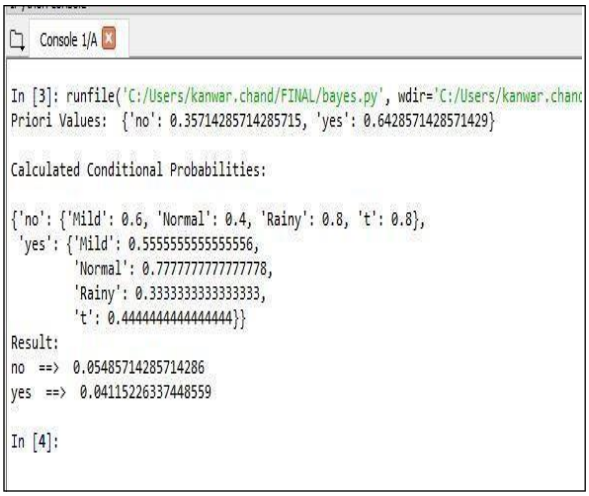
Console 1/A
In [3]: runfile('C:/Users/kanwar.chand/FINAL/bayes.py', wdir='C:/Users/kanwar.chanc
Priori Values: {'no': 0.35714285714285715, 'yes': 0.6428571428571429}
Calculated Conditional Probabilities:
{'no': {'Mild': 0.6, 'Normal': 0.4, 'Rainy': 0.8, 't': 0.8},
'yes': {'Mild': 0.5555555555555556,
'Normal': 0.7777777777777778,
'Rainy': 0.3333333333333333,
't': 0.4444444444444444}}
Result:
no ==> 0.05485714285714286
yes ==> 0.04115226337448559
In [4]:

Fig-16

```
class Classifier():
    data = None
    class_attr = None
    priori = {}
    cp = {}
    hypothesis = None


    def __init__(self,filename=None, class_attr=None ):
        self.data = pd.read_csv(filename, sep=',', header =(0))
        self.class_attr = class_attr

    '''
        probability(class) =    How many  times it appears in cloumn
                                ____________________________________
                                    count of all class attribute
    '''
    def calculate_priori(self):
        class_values = list(set(self.data[self.class_attr]))
        class_data =  list(self.data[self.class_attr])
        for i in class_values:
            self.priori[i]  = class_data.count(i)/float(len(class_data))
        print ("Priori Values: ", self.priori)

    '''
        Here we calculate the individual probabilites
        P(outcome|evidence) =   P(Likelihood of Evidence) x Prior prob of out
                                ____________________________________________
                                                P(Evidence)
    '''
```

Fig-17

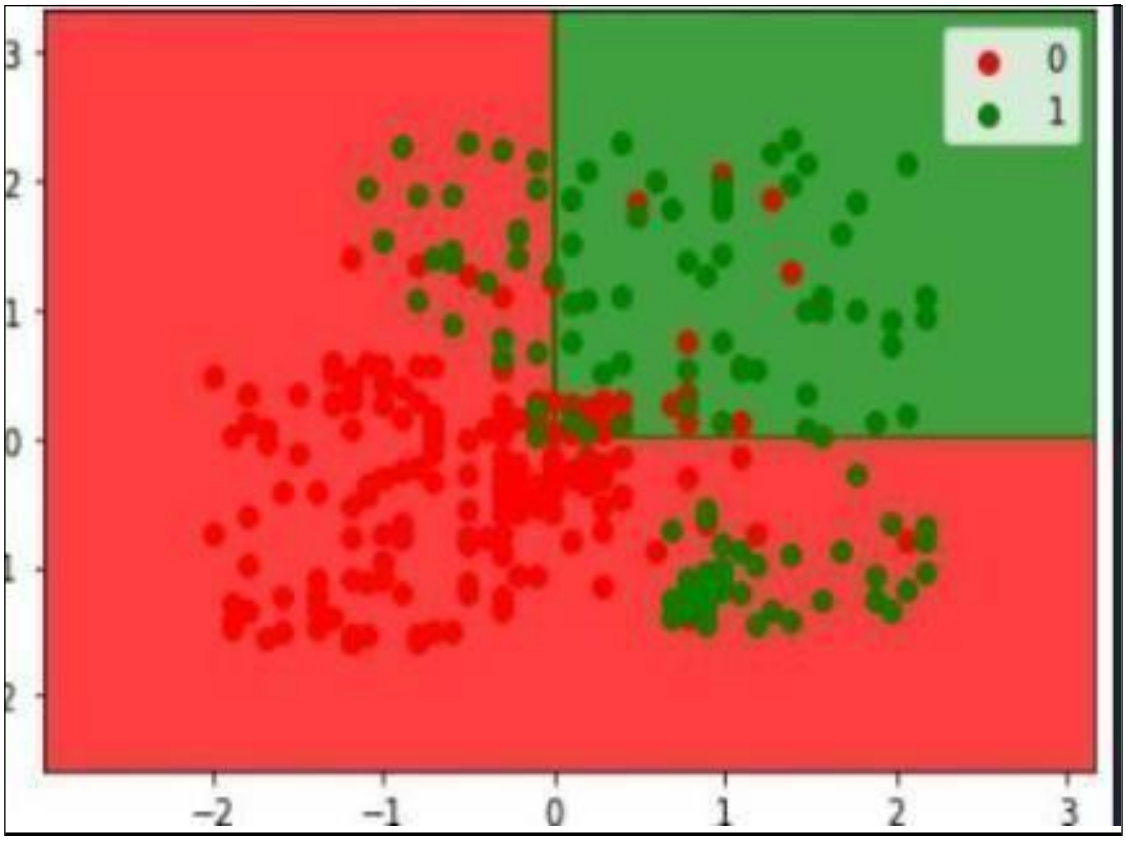

Fig-18

CHAPTER-4

Findings of Phase-3

4.1 Objective:

Here our main objective was to predict the temperature with the most accurate accuracyand display the conditions of the weather with respect to temperature on a web page byusing web framework.

4.2 Our Findings:

Our main findings in this phase was:

I- A good accuracy of our model.

II- The web framework to display the results using Flask Framework.

4.3 Explainations of findings:

4.3.1 Good Accuracy of Model:

We here tried to Improve the Accuracy of our Model to make it more accurate and precise to desired output.

We tried to implement different machine learning models to make the model more accurate like logistic Regression,linear Regression,Decision tree,Random Forest Algorithm.

All of these algorithms had provided us with different accuracy but the one algorithm which provided us the most least error rate was Random Forest Algorithm.

Random Forest Algorithm:

Random forest algorithm or we can say Random decision forests are an ensemble learning method of classification ,regression and other tasks that are constructed by multitude of decision tree during the time of training.

This is the algorithm which can be used for classification,Regression at a same time simentaneously.

This algorithm is basically a Supervised type of classification Algorithm where decision tree concept is used to predict the results.We here usually create a forest and from that forest find the results.

The result accuracy totally depends upon the no. of tress in the forest.As the number of trees in a forest increases the accuracy of the Model also gets increased.

This algorithm is similar as decision tree but not the decision tree to be constructed with Information gain,Entropy etc.

The major difference between the decision tree and the Random forest is that the splitting of trees to get result and finding root node is different,in decision tree it totally depends upon the information gain and entropy while in random forest it runs Randomly.

Advantages of Random Forest Algorithm:

1- The first and important advantage is that it can be **used for both classification and regression techniques.**

2- We above discussed the advantage of having greater number of trees for better results but also it is useful as our model would **not get Overfitted** as other models may get Overfitted.

3- This is the one basic and necessary model which can **effectively handle the missing values,Invalid values and null and blank values too.**

4- The last but not the least advantage is that it can be **used for the modeling of Categorical Fields.**

5- The major advantage of this algorithm is that it is very **Stable**.From stability we means if after training of the data new values or data is being added it adjust the model itself and gets updated on basis of previous training.

Applications:

1- It is basically used to find the Churned customers for a Subscription Of a Telecom company.

2- It can used in the sector of **Banking and finance.**

3- It can be used in the field of **health industry.**

4- It can be used in **online platforms.**

Working Of Random Forest:

The Algorithm is mainly divided into 2 stages:

i) Creating a forest

ii) Determining the results from the forest created with the help of dataset used.

The main functions for **Creating a Forest** are:

1- Randomly, select 'm' features from the given dataset which would be containing total 'n' features that means the attributes or the fields.

2- Using the method called **Best split Point** we have to calculate the Nodal distance ‘d’ from the randomly selected ‘m’ features.

3- Using the Above method **Best Split Point,**split the node into child nodes.

4- Now,repeat the steps from 1-3 until and unless we reached a state where we have ‘l’ number of nodes.

5- For creating the ‘t’ number of total trees repeat all the steps ‘t’ number of times.

After the forest has been created we have to now make predictions out of that which could be made by the following STEPS:

1- Use the features which are take for testing of model,using the rules of particular creating decision tree for predicting results and after that storing the outcome.

2- Now calculate the no of frequency for each value of target.

3- Take the value which is having higher frequency and display as well as store that value.

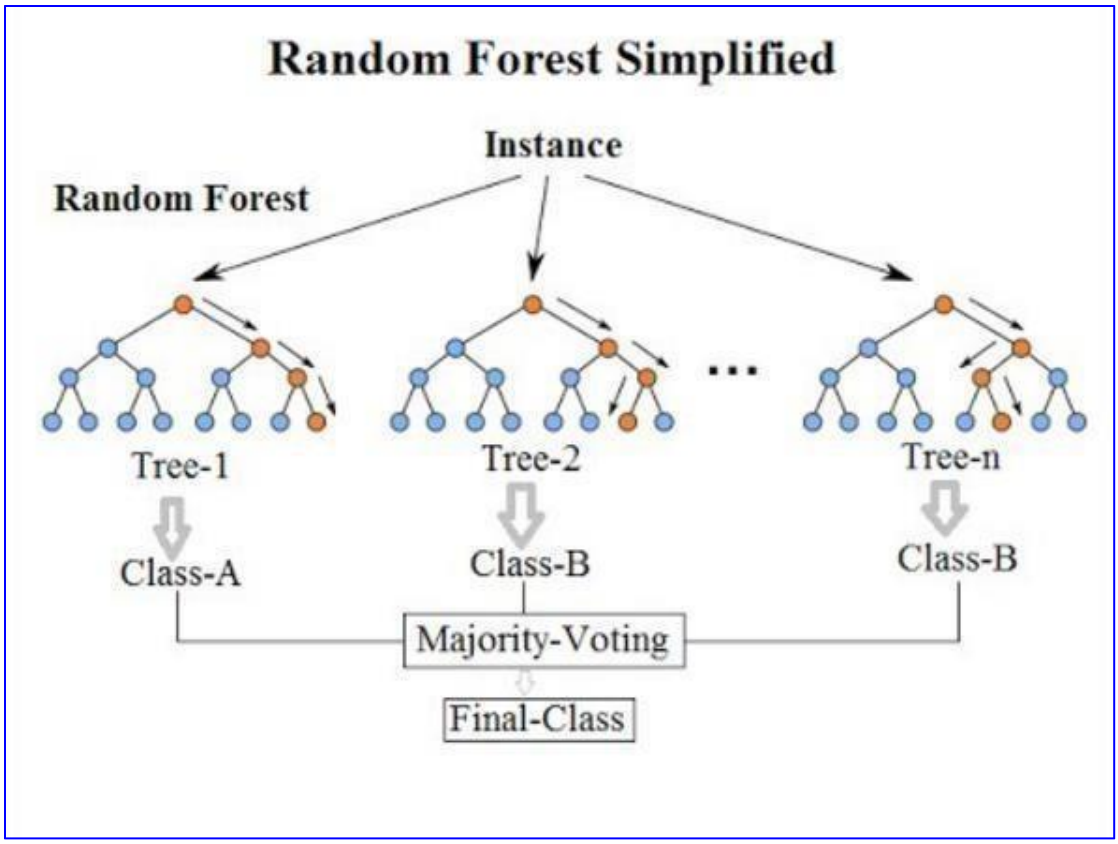

Fig-19

Disadvantages of Using Random Forest:

1- The **time complexity of the prediction** of results in random forests **increases** as the size of trees increases.Mostly,it is likely to create 100 trees in the library of python.

2- It requires **more computational resources** than the decision tree needs.It is very complex to use.

3- When Random forest is compared with a decision tree it **requires more time for the period of training** with respect to the decision tree.

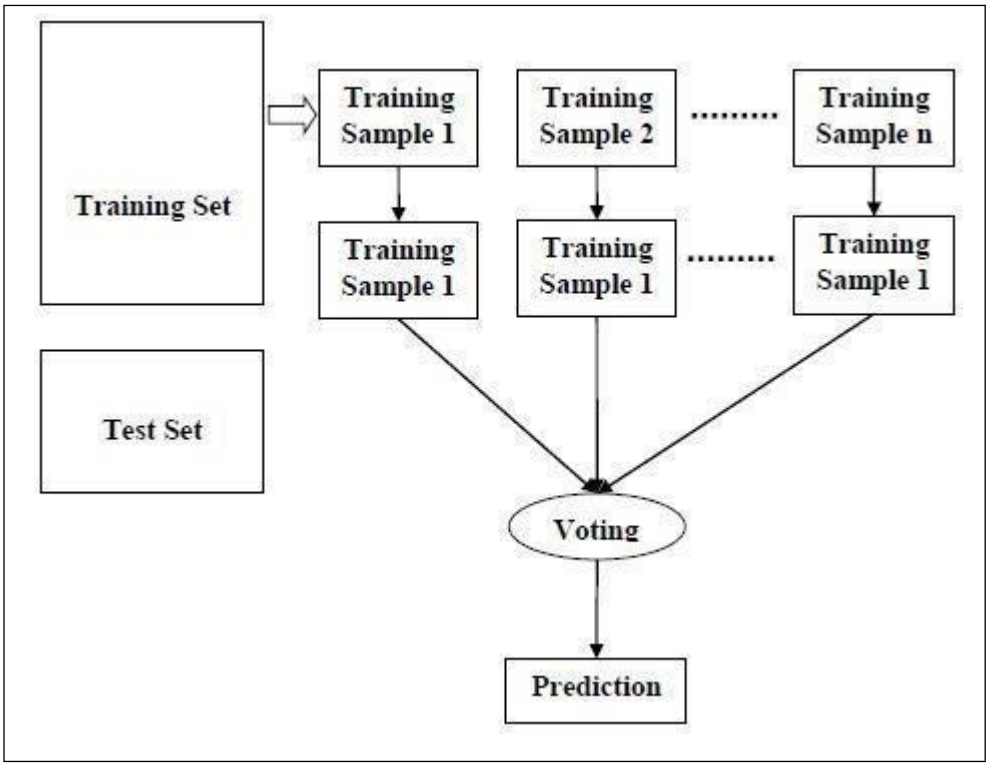

Fig-20

4.3.2 Web Framework FLASK:

After the prediction of results we were to think now what to do with the predicted,how to showcase them in a useful manner to a user.

So,we thought to build a web page application using a framework to showcase our results.For that we need a framework which would be Very flexible,easy to use and build.

At last we decided and concluded to use the Framework of Flask.

FLASK(A micro Web Framework):

Flask is a micro web framework which is usually written and encoded in python with some special qualities and features in it.

It can also be called a Micro web Framework as it does not requires any Tools or libraries as it is a framework,contains all in it.The framework has no storage purpose that means it donot have any particular storage format,no session concept in this,but we can create or export the data from flask to database by invoking any database in flask.

For form validations we need some extensions to be used here for which we have permission in this.It was basically developed by Armin Ronacher in 2010.

This framework is based on some projects like Pocoo,Werkzeug etc.

It consists of many Features in it:

1- It comprimise of server debugging and a debugger.

2- It uses the jinja project templates to build a web page.

3- It s totally Unicode .

4- It is an integrated support for unit testing of the problem.

5- It has the ability to store the client side cookies that means client sidesessions.

Advantages of Using Flask:

1- The Model or web page built with this application takes up little space in memory or we can say it is very ligtweighted.

2- This framework is very modular therefore we can change it to any form anytime we need it.

3- We can have addition of extensions in it whenever we need it like addign of database at any time.

4- The framework is very structured.User can easily understand how the code is working.

5- It store Client side sessions therfore user can see when the user logged in,how much time he spent, what things he selected in the application,when he logged out of the application

6- It is very easy to build the flask application and deploying it too is very easy.

7- It can Handle the HTTP requests very easily with the help of many functions.

8- It is very flexible when compare to using the Django.

Applications And Uses:

1- We can even obtain Graph using Flask to get Real time analysis.

2- Flask can be used in full stack development.

3- It can be used for debugging.

4- It can be used for Unit testing.

Disadvantages:

1- Without the knowledge of HTML,CSS the user cannot work efficiently with Flask.

2- The flask can handle One request at a time that means it does not work on the principle of Muti -threading.

3- It is not Scalable.

4- When we use extensions it means there is 3rd party involvement in it which can be a major threat to security for the application.

4.4 CODE LIBRARIES AND FUNCTIONS SECTION:

In this section using the code we had tried to explain the libraries and functions used in the project.

```
import warnings
warnings.filterwarnings('ignore')
import os
import numpy as np
import pandas as pd

from matplotlib import pyplot as plt
import pickle

import sklearn
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score


from sklearn import preprocessing

```

Fig:21

Numpy:

Using the library we can have support for the large calculations like matrices multiplication,basically used for mathematical functions invoking to solve the equations.

Pandas:

Using this library we can create Dataframes to store the inputs and outputs.

Matplotlib:

Using this library we can plot the graphical representations the data given by calculating the values using numpy library

Pickle:

Using this library we can serialize the data and its output to use it further,or to import the data to another application.

Here Pickle in our project is helping us to import the output to the flask using the **Dump** function.

Sklearn:

It is a library which is free for invoking the methods to imply the machine learning algorithms.

Sklearn.model_selection:

This library gives us an description of using the data and to measure the new data.It also helps to perform the training and testing of the dataset.

Accuracy_score:

This helps to perform the accuracy metrics on the model to get us know the proper accuracy and error rate of the model we build.

```
train_X,test_X,train_y,test_y=train_test_split(weather_x,weather_y,test_size=0.2,random_state=4)
train_X.head()
regr=RandomForestRegressor(max_depth=50,random_state=0,n_estimators=100)
regr.fit(train_X,train_y)
prediction=regr.predict(test_X)
np.mean((prediction-test_y)**2)
pd.DataFrame({'actual':test_y,
              'prediction':prediction,
              'diff':(test_y-prediction)})

pickle.dump(regr,open('bestfinal.pkl','wb'))
```

Fig-22

```
from flask import Flask,request, url_for, redirect, render_template
import pickle
import numpy as np

app=Flask(__name__,template_folder='template')

model=pickle.load(open('bestfinal.pkl','rb'))
```

.**Fig-23**

Flask:

This is the instance of the flask application to used in further in the code.

Request:

This is the library used for the HTTP requests invoking for the user friendly environment and keeping the records for the client side sessions.

Url_for:

This is the library for the passing of the url to the application where we are using the problem statement to execute it.

Redirect:

This library is used to redirect the application to a given link using the **url_for** library.

Render_template:

This is basically a function based on JINJA engine that redirects the output to the required location.It gives an error also when the code doesnot work properly.

Pickle.load():

This is the function which helps us to deserialize the output from the pickle file which we had created in the machine model to store the results.

Here in this the output is stored in **model** named variablewhich will be used fuurther to extract the results.

```
@app.route('/')
def home():
    return render_template("index.html")


@app.route('/predict',methods=['GET','POST'])
def predict():
    int_features=[float(x) for x in request.form.values()]
    final=[np.array(int_features)]


    output=model.predict(final)
    if format(output[0])>str(22.9):
        return render_template('index.html',prediction_text='Temperature is {} IT may Be sunny'.format(output))
    else:
        return render_template('index.html',prediction_text='Temperature is {}It may be cloudy'.format(output))


if __name__ == "__main__":
    app.run(debug=True)
```

Fig-24

@app.route():

This function routes the address of the location to the given location.

App.run(debug=True):

This function helps us to initialize the process.

In this code the program first reaches to the page the **index.html** which is our basic starting web page and then collecting the inputs from the webpage in the form of a from it stores it into the **int_features** and then it computes the mathematicl functions to it.

Now, after storing the inputs it then uses the **model** variable which was having the data to predict the result then uses that to predict the temperature conditions.

After, the results are being predicted it basically uses **if-else** conditions to predict the conditions of the day.

4.5 WORKING OF THE PROJECT:

8- First of all we had created the machine learning model to have an proper accuracy and less error rate.

```
In [49]: prediction=regr.predict(test_X)
         np.mean((prediction-test_y)**2)

Out[49]: 0.0018652798160000048

In [50]: pd.DataFrame({'actual':test_y,
                       'prediction':prediction,
                       'diff':(test_y-prediction)})
```

Out[50]:

	actual	prediction	diff
37443	-2.288889	-2.277389	-1.150000e-02
86534	8.861111	8.860056	1.055555e-03
2082	9.805556	9.827222	-2.166667e-02
53130	27.222222	27.210722	1.150000e-02
45196	17.705556	17.705556	3.552714e-15
57822	3.888889	3.888889	9.325873e-15
26754	17.777778	17.777778	-1.065814e-14
53177	28.977778	28.897778	8.000000e-02

Fig:25

9- After creating the model,we had dumped it into the pickle file for serializing for flask.

```
In [53]: import pickle

In [54]: with open('model_pickle','wb') as f:
             pickle.dump(regr,f)
```

Fig-26

10- After the dumping the target was to where to use the flask application,so for that we created a web page that contains a form to accept the parameter and predict the results.

11- Now the web page was created then we created the flask file and then connected the flask file with the html web page by passing address of the page in form method.

Fig-

12- Now we connected both and then for that we had to make a proper arrangement of the files and folder to run it.

13- Now using command prompt we used to RUN the application.

14- In command prompt when we ran the python file using **Python app.py**(flask file) then we prompted to use a url to reach to a application.

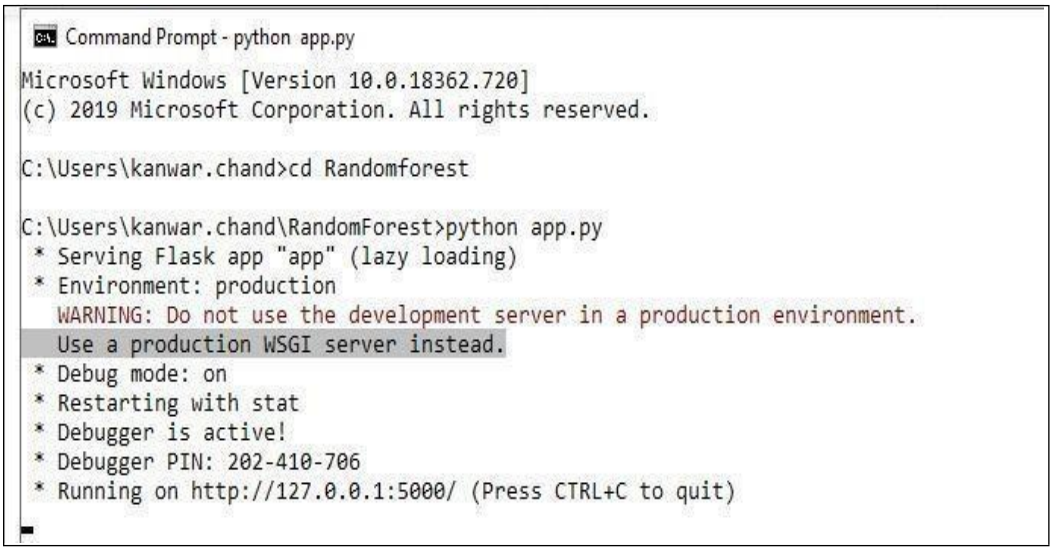

Fig -28

4.6 Working outputs of the Project:

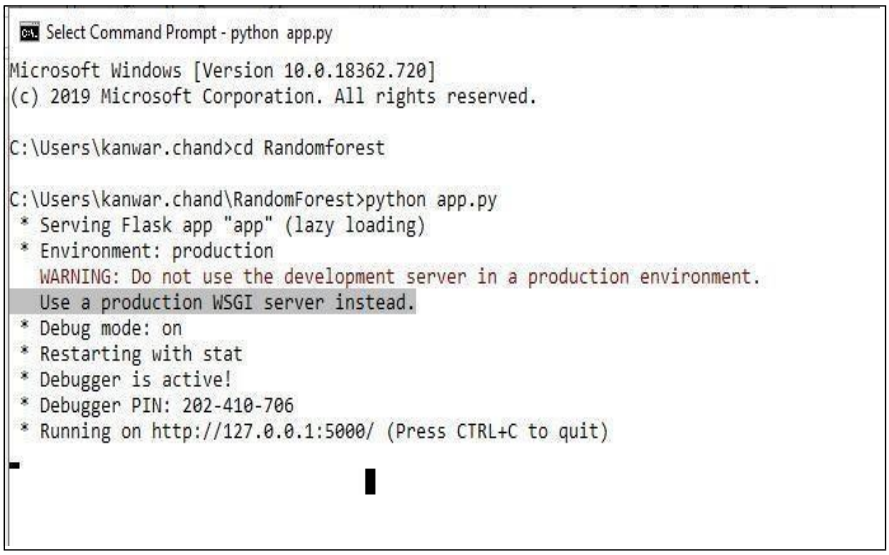

Fig -29

This is the interface for command prompt where we can see the a localhost address where our application would be running.

In this debugger is active,any fault would lead to the error message.

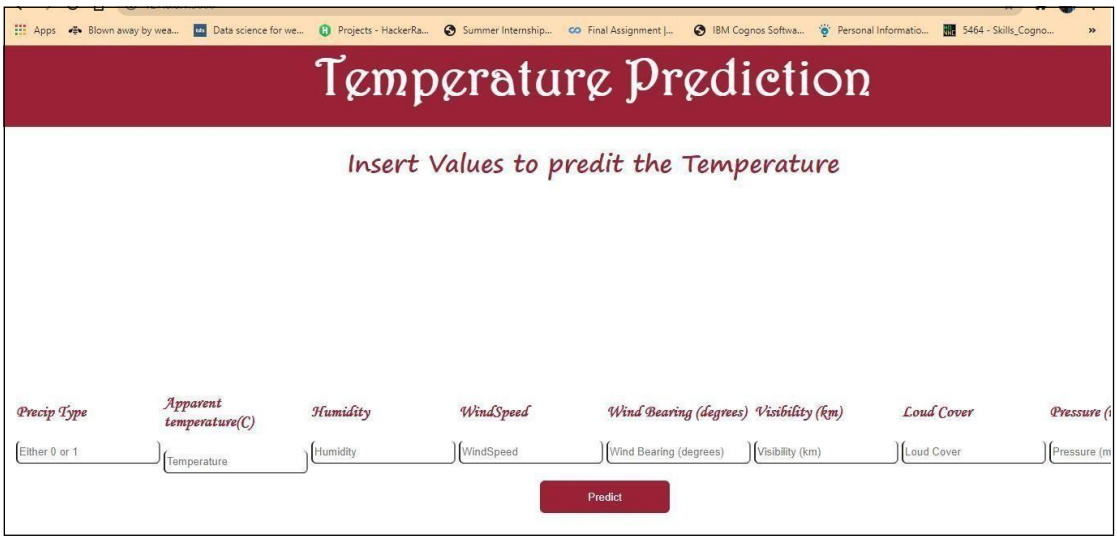

Fig:--30

This was our officially web page for the which was passed in the route() function in flask program.

It consists of a form which has some fields to be input by the user then based on the training data it is predicting the results.

Here the predict button is the one where we had linked the flask program application.

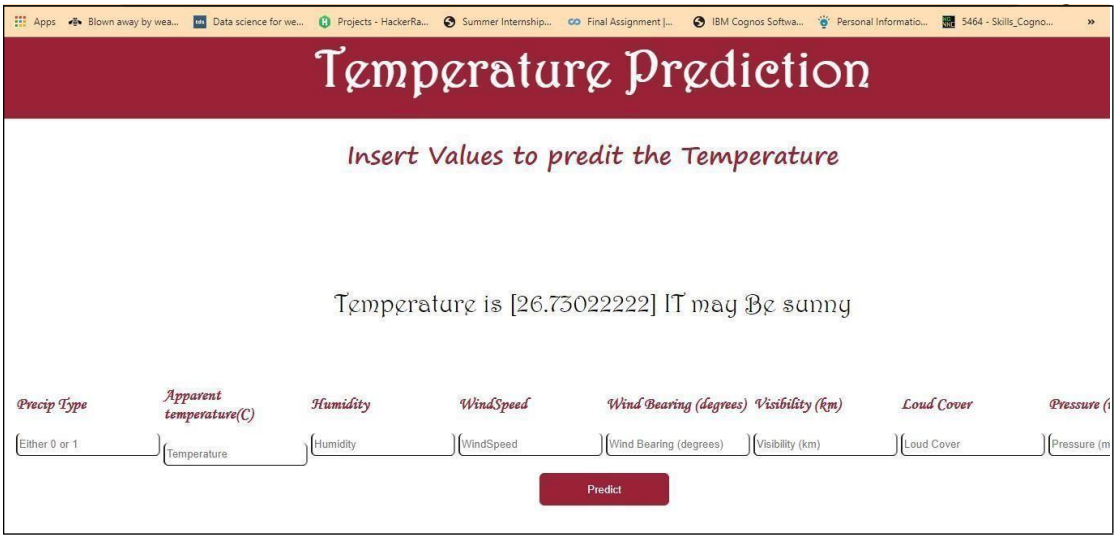

Fig:31

Based on our passed Inputs we had got the answer with the Prevailing conditions of the weather according to the Temperature.

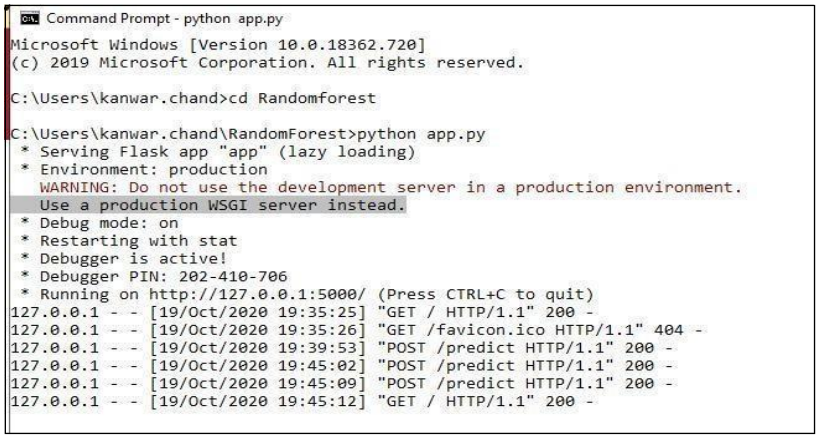

Fig:-32

As we had earlier discussed that in Flask we usually able to receive the client side sessions,so in the above image we could see the sessions information on our command prompt.

Here,we are able to see the date,time,get,post method used by client and also the application address.

CHAPTER-5

Future scope and conclusions

5.1 Conclusions:

As we are advancing in the world we are leaving everything behind us,but forgetting that somethings needs to be taken along with us so that in future we may not regret to let us not carry that thing.

As oxygen is necessary for living and inhaling like that nature is the integral part of the ecosystem which needs to be understood and should not be taken lightly as it has the integral powers to make us realize that importance.

Therefore our motive is to build a alarming system such that whenever a natural disaster would be heading towards any area our model would circulate a message for making a warning.

As our model is having less accurate so we are sure that our model would generate a good set of results for the prediction and would be working in a good and excellent condition.

5.2 Future Scope:

There are many future scopes of this weather prediction project.The future scopes are:

* It can be used to implement it in the field of natural disaster management like drought,floods,cyclone,heavy rainfall,snowfall,Avalanche etc.

* It can be used to know the Soil type of a particular region,to know the structure of soil in that area.

* It can be used to know what type of vegetation belt is found in any geographical location in world.

* It can used in the field of the agriculture,so that the farmers could know which crops to be grown in which season.

* It can be used to detect diseases like malaria,dengue.

* It can be used to detect the shortest path on Google maps Based the conditions of aparticular day

BIBLIOGRAPHY

https://flask.palletsprojects.com/en/1.1.x/

https://palletsprojects.com

https://www.fullstackpython.com/flask.html