**AIRCHECK**

A project Submitted to

University of Mumbai for partial completion of the degree of

Bachelor in Science (Computer Science)

Under the faculty of Science

**By**

MR GAURAV.VINOD.SINGH

**Under the Guidance of**

MRS.HARSHALI MANKAR

**Shri. Vishnu Waman Thakur Charitable Trust’s**

**Bhaskar Waman Thakur College of Science,**

**Yashwant Keshav Patil College of Commerce**

**Vidhya Dayanand Patil College of Arts,**

**VIVA College**

**2019-20**

**Late Shri Vishnu Waman Thakur Charitable Trust’s**

**Bhaskar Waman Thakur College of Science,**

**Yashwant Keshav Patil College of Commerce,**

**Vidhya Dayanand Patil College of Arts.**

**Virar (W).**

CERTIFICATE

This is to certify that project entitled **AIRCHECK** under guidance of MISS **HARSHALI MANKAR** submitted by Mr/Ms **GAURAV VINOD SINGH** Roll no: **78** in partial fulfilment of B.Sc.(CS) degree

(Semester-V) examination had not been submitted for any other examination and does not form any other course undergone by the guide.

It is further certified that he/she has completed all required phases of the project.

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Project Guide Examiner Head of Dept.**

**Date: Date: Date:**

**COLLEGE SEAL:**

**Declaration by learner**

I, here by further declare that all information of this document has been obtained and presented in accordance with academic rules and ethics conduct.

I the undersigned Mr. **GAURAV VINOD SINGH** hereby declare that the work embodied

In this project work title **AIRCHECK** form my own contribution to research carried out under the guidance of Mrs. **HARSHALI MANKAR** is a result of my own research work and has not been previously submitted to any other university.

Wherever reference has made to previous works of others, it has been clearly indicated as such and include in the bibliography.

**Mr**. GAURAV VINOD SINGH

Signature:

Certified by

**Mrs.** HARSHALI MANKAR

Signature:

**INDEX**

|  |  |  |
| --- | --- | --- |
| **SR.NO.** | **PARTICULARS** | **PAGE.NO.** |
| **1** | **INTRODUCTION**   * Preface * Brief | **6** |
| **2** | **PRELIMINARY INVESTIGATION**   * Data Preprocessing * Hardware and Software Requirement * Gantt Chart | **8** |
| **3** | **ANALYSIS AND VISUALIZATION**   * Data * Model | **12** |
| **4** | **MODEL DESIGN**   * ER Diagram * Activity Diagram * Workflow * Overview of model | **16** |
| **5** | **MODEL TESTING**   * Test Cases | **19** |
| **6** | **MODEL LAYOUT** | **20** |
| **7** | **PROGRAM LISTINGS**   * Coding | **21** |
| **8** | **CONCLUSION** | **28** |
| **9** | **BIBLIOGRAPHY** | **29** |

**INTRODUCTION**

**PREFACE**

Let me present you my project, I have made this project with sincerely, hard work, dedication, punctuality and with lots of efforts.

The main aim of my project is to provide a platform to the users from where they can use my machine learning model in order to predict the Air Quality Index of several parts of areas in India.

The whole idea to build this model came to me while I was watching news about the air quality in Delhi which is very danger for the living things, I was wondering and started to think how can I contribute something good to this problem? Awareness and attention is also something I would like to address here as a matter of fact.

Gaurav Singh.

**Brief**

The whole process of building the model starts from collecting data, in this case I collected the data from [www.kaggle.com](http://www.kaggle.com).

Once the data is available we can now start building the model but before doing that remember sometimes the data can be raw, so in order to tackle it we need to perform some cleaning of data or wrangling which we will discuss further.

When we have clean data with us we can start visualising it with the help of some library in python like pandas, matplotlib, seaborns, etc.

Python is a programming language as of now one can say best platform for data science work there are also some other programming languages like R but as campare to Python R is limited.

So once we visualize the data we can see the patter and trends of the data and also we can see the flow of our data which will help us to decide what type of algorithm will fits more to the data.

To check which algorithm is better for the data we keep on using a technique which is called “Trial & Error”, this means that we have to keep applying different algorithms to our data like in this case I have used **Multi-linear regression variance, Linear Discriminant, Naïve Bais** and campare their accuracy to get better algorithm for the model.

Once you have a model with better accuracy its time to deploy the model so that everyone can make use of it. Deployment require a good GUI for user and some different approach to deploy the model like by using flask, Django, Docker, etc.

**PRELIMINARY INVESTIGATION**

**Data Preprocessing**

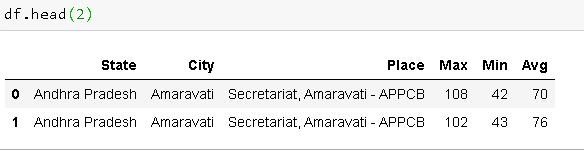
Data Preprocessing involves a lot of process and it is time consuming.

Basically it deals with the data and removes all the duplicates and empty values.

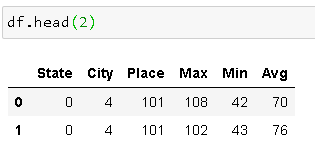
There are many ways to do so and its is simplest in python as are as follows

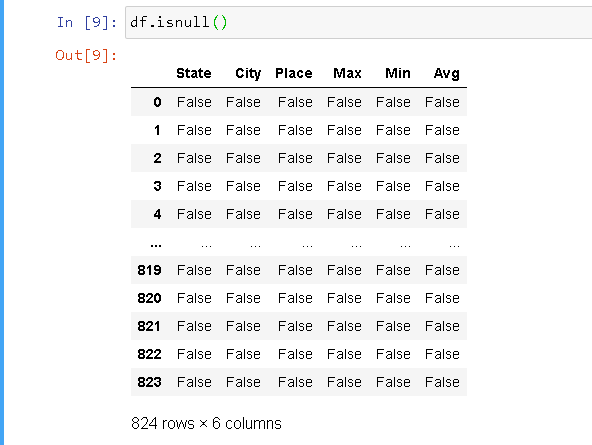
* **Labelencoder()-**it convert the simple text to binary format for further operations

Without Labelencoder



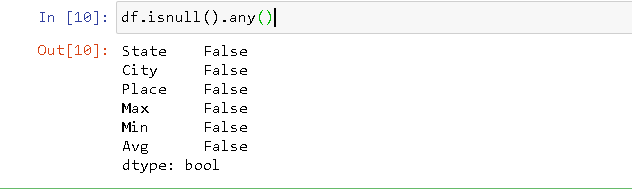
With Labelencoder



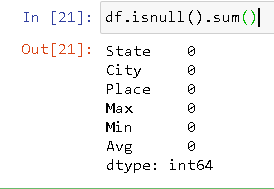
* **isnull()** – it displays all the null values from the dataset

Result in false state means there are no null values in dataset

* **isnull().any()** –it displays the columns which have null values

Result in true state indicates that there are null values in the dataset

* **isnull().sum()**



It gives you the total number of missing values

* **fillna()** –allow to add something to those values which are not available(NA)

**Hardware and Software Requirement**

*Hardware Requirements:*

* Pentium III or higher
* 256 MB RAM
* 500 MB hard free drive space

*Software Requirement:*

* Python
* MS-Excel
* Google Chrome
* Command Prompt

**GANTT CHART**

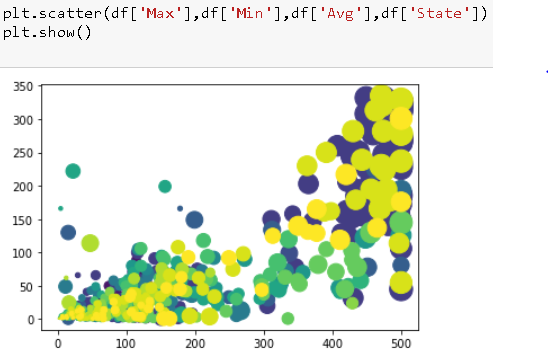
**DATA ANALYSATION AND VISUALIZATION**

**Data**

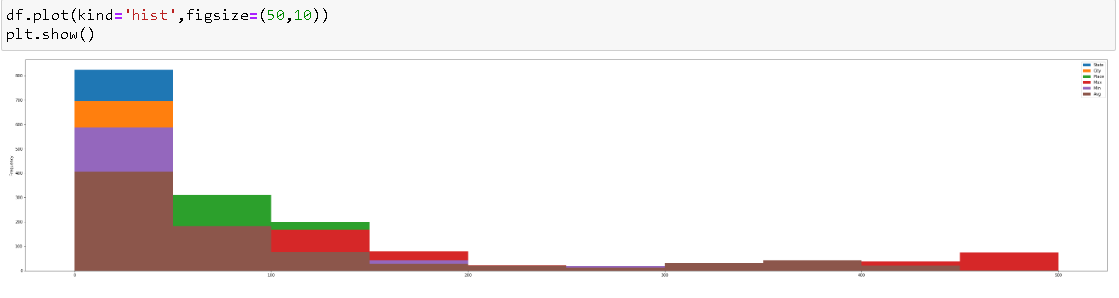
* The **describe()** method is used for calculating some statistical data like percentile, mean and std of the numerical values of the Series or DataFrame. It analyzes both numeric and object series and also the DataFrame column sets of mixed data types.



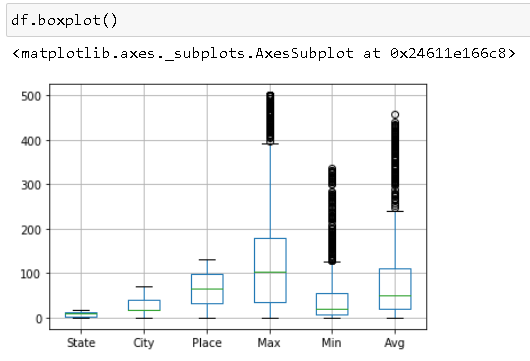
* Scatter plot simply display the data in form of dots and it help the user to see the trends and flows of data and how data is variance.



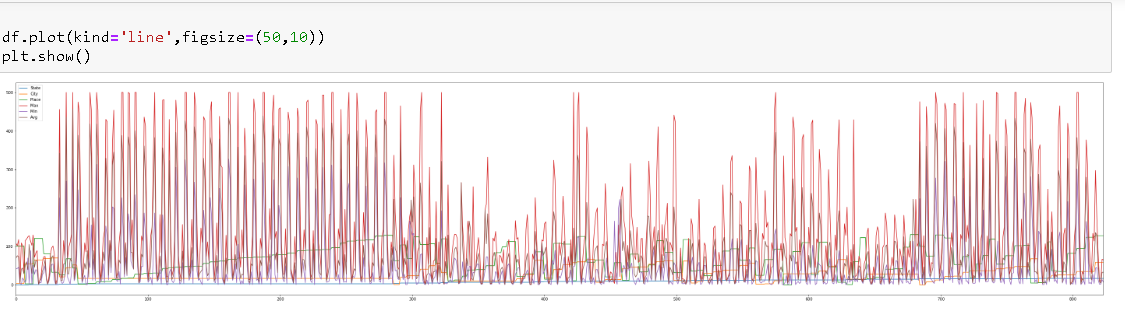
* Histogram allows the developer to visualize the data in the form of histogram



* Boxplot is very important as it display the user to see the outliers in the dataset

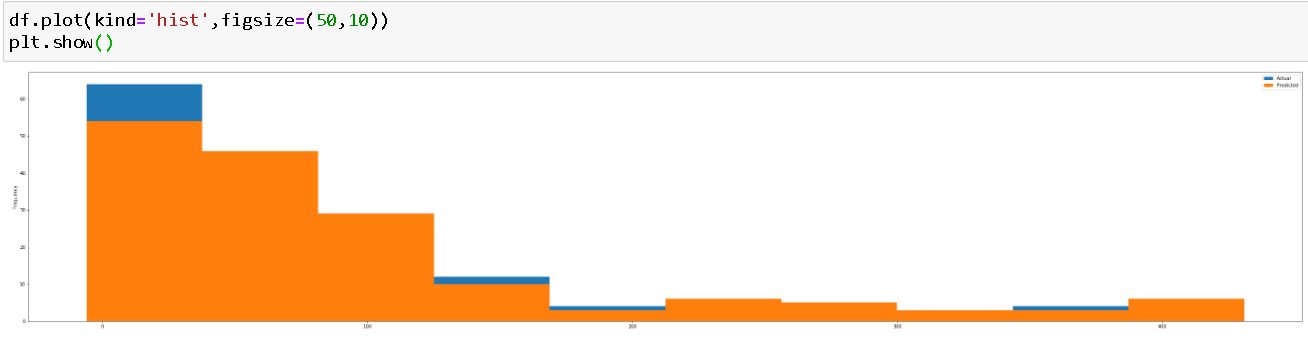


* The data is displayed using line graph

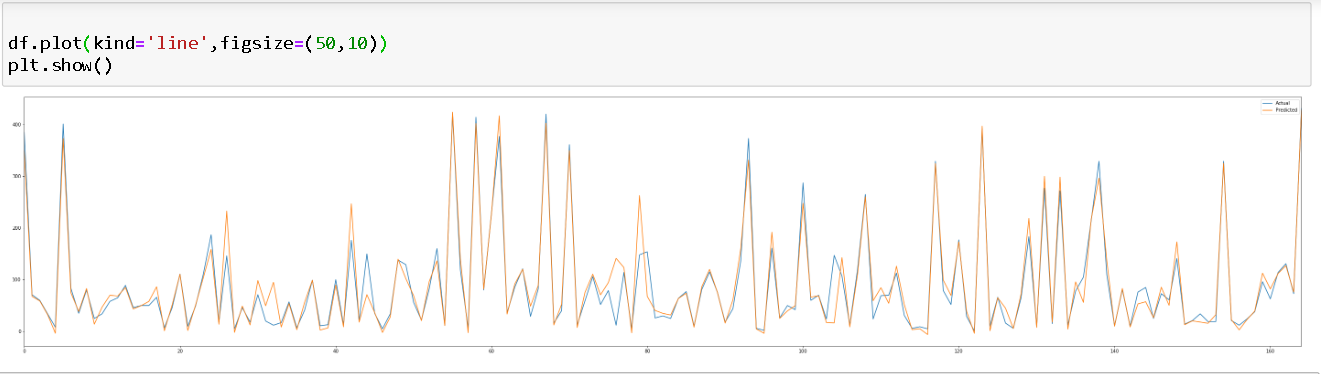


**Model**

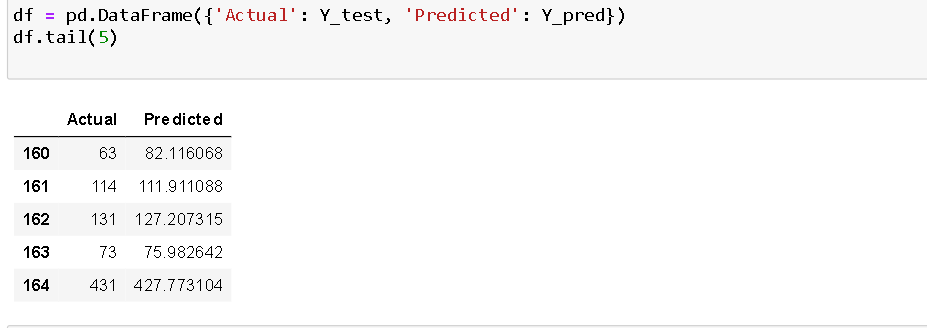
* Histogram representation of the model



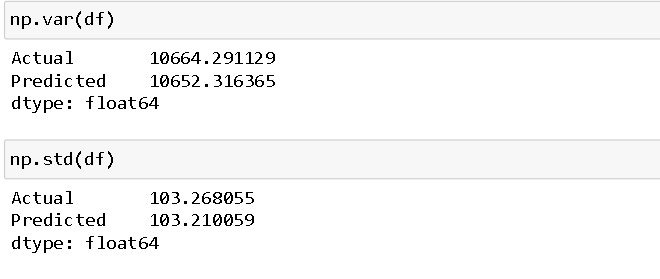
Line representation of model



* The difference between actual and predicted values



* The standard deviation and variance of model

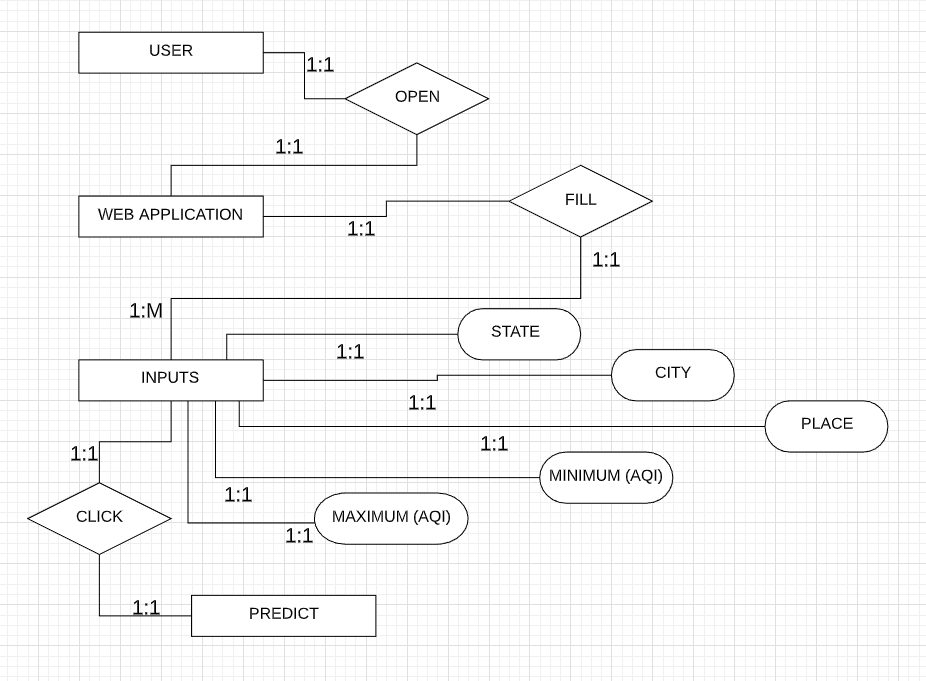


* The Bar Graph representation of actual and predicted values

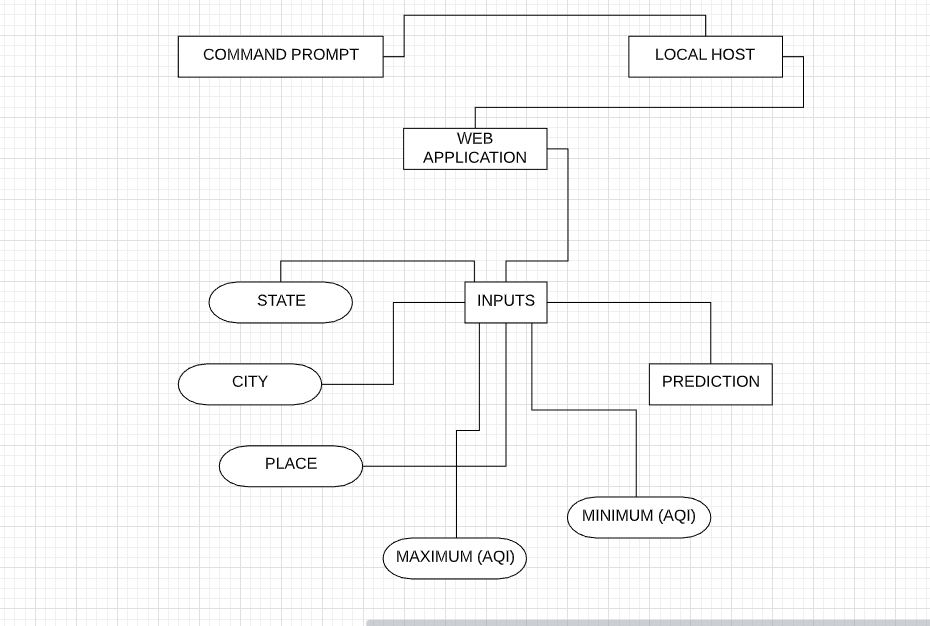


**SYSTEM ANALYSIS**

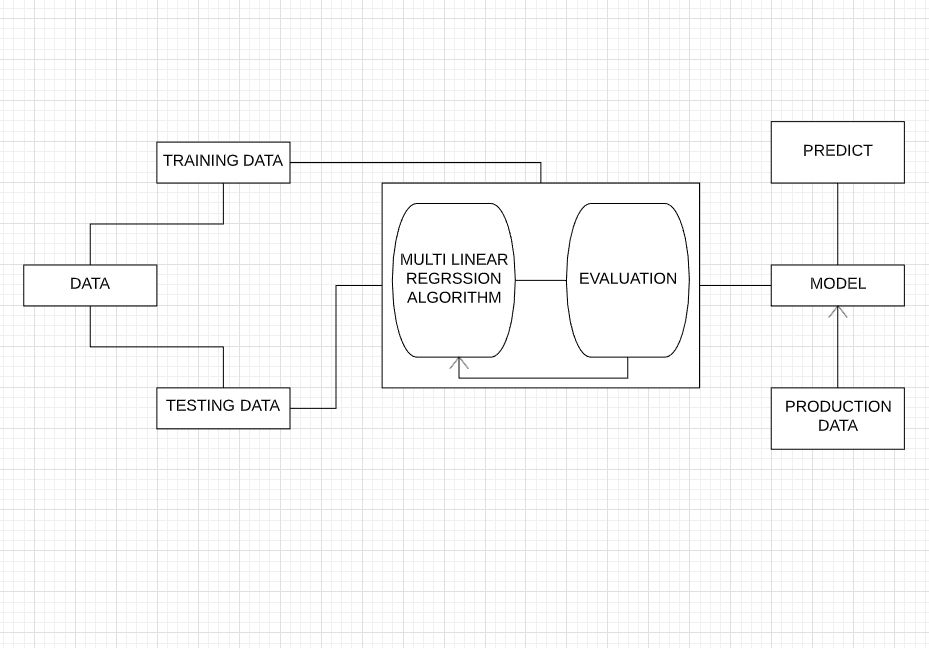
**ENTITY RELATIONSHIP (ER)**



**ACTIVITY DIAGRAM**

****

**WORKFLOW**

****

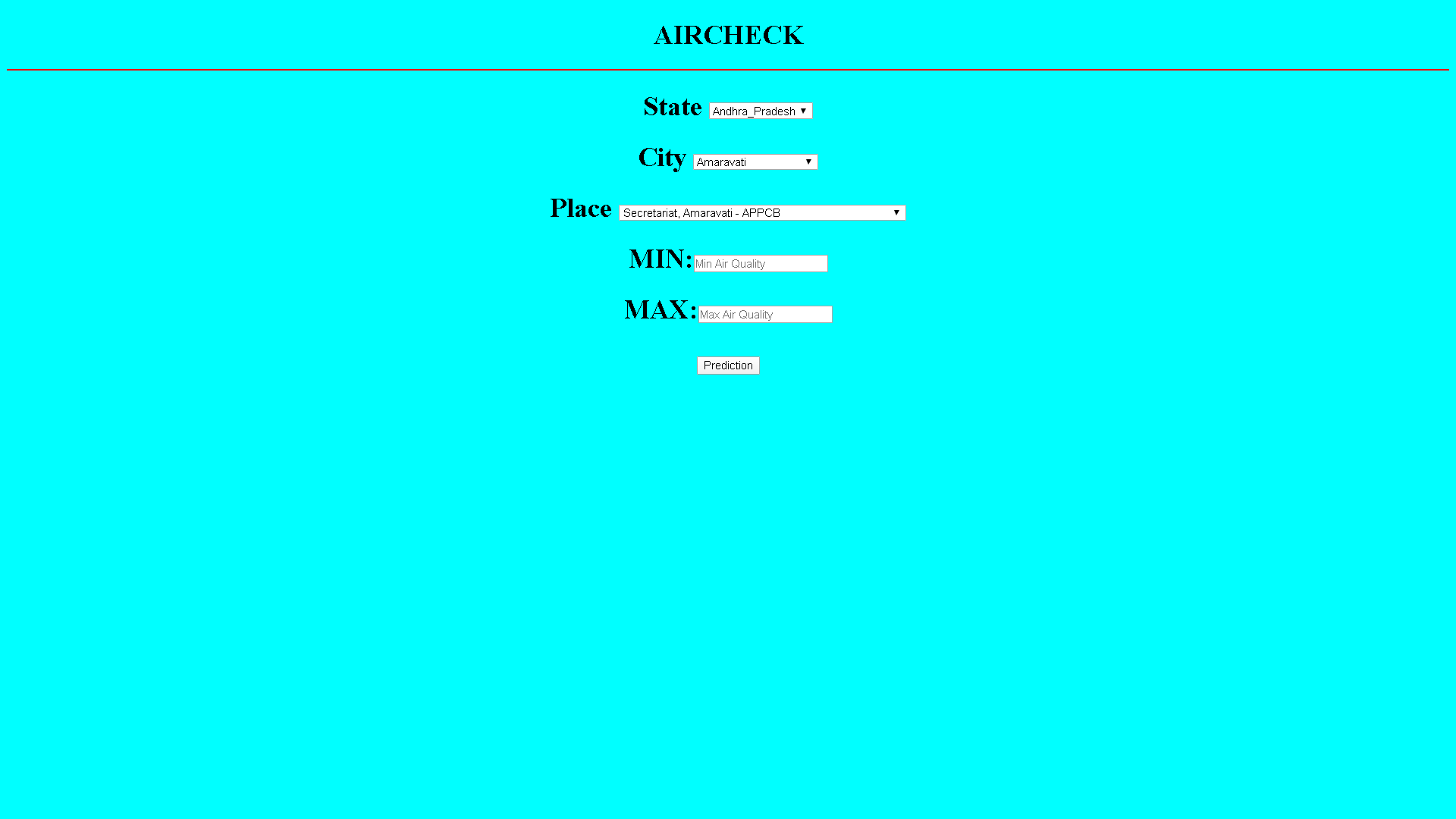
**SYSTEM TESTING**

**Test Cases**

Below is a format for entering input into the model

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Case ID** | **Test scenario** | **Test Steps** | **Test Data** | **Expected**  **Result** | **Actual Result** | **Pass/Fail** |
| **TU01** | Check the State input with valid data | Enter State name | Delhi | Acceptance of value | Value should be accepted | PASS |
| **TU02** | Check the State input with invalid data | Enter State name | Delhii | Rejection of value | Value should be Rejected | PASS |
| **TU03** | Check the City input with valid data | Enter City name | Delhi | Acceptance of value | Value should be accepted | PASS |
| **TU04** | Check the City input with invalid data | Enter City name | 10 | Rejection of value | Value should be Rejected | PASS |
| **TU05** | Check the Place input with valid data | Enter Place name | Anand Vihar | Acceptance of value | Value should be accepted | PASS |
| **TU06** | Check the Place input with invalid data | Enter Place name | Anand | Rejection of value | Value should be Rejected | PASS |
| **TU07** | Check the minimum air quality index input with valid data | Enter min air index value | 43 | Acceptance of value | Value should be accepted | PASS |
| **TU08** | Check the minimum air quality index input with invalid data | Enter min air index value | & | Value should be Rejected | Value should be Rejected | PASS |
| **TU09** | Check the maximum air quality index input with valid data | Enter max air index value | 108 | Acceptance of value | Value should be accepted | PASS |
| **TU10** | Check the maximum input with invalid data | Enter max air index value | Delhi | Value should be Rejected | Value should be Rejected | PASS |

**SCREEN LAYOUT**



**PROGRAM LISTINGS**

**Coding**

**Model.py**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from sklearn.model\_selection import train\_test\_split

from sklearn import metrics

from sklearn.metrics import r2\_score

from sklearn.linear\_model import LinearRegression

from sklearn.preprocessing import LabelEncoder

from sklearn.metrics import mean\_squared\_error

from math import sqrt

import pickle

%matplotlib inline

df=pd.read\_csv(r"Downloads/air.csv")

le=LabelEncoder()

df['State'] = le.fit\_transform(df.State.values)

df['City'] = le.fit\_transform(df.City.values)

df['Place'] = le.fit\_transform(df.Place.values)

df.dtypes

df.head(1)

df.isnull().any()

X=df.iloc[:,:5].values

Y=df.iloc[:,-1].values

plt.scatter(df['Max'],df['Min'],df['Avg'],df['State'])

plt.show()

df.plot(kind='hist',figsize=(50,10))

plt.show()

df.boxplot()

df.plot(kind='line',figsize=(50,10))

plt.show()

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size=0.2, random\_state=0)

regressor = LinearRegression()

regressor.fit(X\_train,Y\_train)

Y\_pred = regressor.predict(X\_test)

pickle.dump(regressor, open('model.pkl','wb'))

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

print(model.predict([[0,4,101,108,42]]))

print('Coefficients: \n', regressor.coef\_)

df = pd.DataFrame({'Actual': Y\_test, 'Predicted': Y\_pred})

df.tail(5)

print('Root Mean Squared Error:', np.sqrt(metrics.mean\_squared\_error(Y\_test, Y\_pred)))

np.var(df)

np.std(df)

df.plot(kind='bar',figsize=(500,40))

plt.grid(which='major', linestyle='-', linewidth='1', color='green')

plt.grid(which='minor', linestyle=':', linewidth='0.5', color='black')

plt.show()

score=r2\_score(Y\_test,Y\_pred)

print(score)

**App.py**

import numpy as np

from flask import Flask, request, jsonify, render\_template

import pickle

app = Flask(\_\_name\_\_)

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

@app.route('/')

def home():

return render\_template('index.html')

@app.route('/predict',methods=['POST'])

def predict():

int\_features = [request.form.to\_string ]

final\_features = [np.array(int\_features)]

prediction = model.predict(final\_features)

output = round(prediction[0], 2)

return render\_template('index.html', prediction\_text='Air Quality Index Should be {}'.format(output))

if \_\_name\_\_ == "\_\_main\_\_":

app.run(debug=True)

**Index.html**

**just a sample**

<!DOCTYPE html>

<html>

<head>

<title>AirCheck</title>

<h1 align="center">AIRCHECK</h1>

</head>

<body bgcolor="cyan">

<div class="login">

<form action="{{ url\_for('predict')}}"method="post">

<hr color="red">

<h1>

<center>State

<select>

<option>Andhra\_Pradesh</option>

<option>Bihar</option>

<option>Delhi</option>

</select>

</center>

</h1>

<h1>

<center>City

<select>

<option>Amaravati

</option>

<option>Rajamahendravaram

</option>

</center>

</h1>

<h1>

<center>Place

<select>

<option>Secretariat, Amaravati - APPCB

</option>

<option>Anand Kala Kshetram, Rajamahendravaram - APPCB

</option>

</select>

</center>

</h1>

<h1>

<center>

MIN:<input type="number" name="Min" placeholder="Min Air Quality" required="required" />

</center>

</h1>

<h1>

<center>

MAX:<input type="number" name="Max" placeholder="Max Air Quality" required="required" />

</center>

</h1>

<h1>

<center>

<button type="submit" value="submit as normal">Prediction</button>

</center>

</h1>

</form>

</div>

{{ prediction\_text }}

</body>

</html>

**CONCLUSION**

These days machine learning techniques is widely being used across all over the world, so when I came to know the problem of air pollution in Delhi I was sure that I am gonna contribute something good as a step to solve the problem and thst’s how I started building this machine learning module named “AirCheck”.

I have used three algorithm which are multi linear regression, Linear Discriminant, Naive Bayes and the accuracy are 93%, 92%, 89% respectively.

Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regression (MLR) is to model the linear relationship between the explanatory (independent) variables and response (dependent) variable.

For the betterment I choosed are multi linear regression with highest accuracy so that model should work well.

The model is very much of predicting the air quality index of your area with the accuracy of almost 94%, which assures that this model is very good and can be helpful for people to know the situation of their area, city, place and then public can be more aware of these situation and then come up with the ideas to solve this this problem.

At the end I would like to conclude that if we do not take major steps regarding this problem than this could cause tremendous amount of harm to us and ofcourse to our next generation.

**BIBLIOGRAPY**

* [www.Kaggle.com](http://www.Kaggle.com)
* [www.towardsdatascience.com](http://www.towardsdatascience.com)
* [www.geeksforgeeks.com](http://www.geeksforgeeks.com)
* [www.stackoverflow.com](http://www.stackoverflow.com)