*Blog on Machine Learning Problem*

Rainfall Prediction- Weather Forecasting

Weather Forecasting is the application of Science and Technology to predict the state of the atmosphere for the future time and given location. Now days, forecasting for accurate atmospheric condition is the major challenge for the meteorologist and poor forecasting has significant impact on our daily lives. This brings the necessity to make research work on forecasting of the weather events.

***Problem Statement****:*

Rain Prediction –Weather forecasting

Weather forecasting is the application of science and technology to predict the conditions of the atmosphere for a given locationand time. Weather forecastsare made by collecting quantitative dataabout the current state ofthe atmosphere at a given place and using meteorology to project how the atmosphere will change.

Rain Dataset is to predict whether or not it will rain tomorrow. The Dataset contains about 10 years of daily weather observations of different locations in Australia.

**Here, predict two things:**

**1. Problem Statement:**

a) Design a predictive model with the use of machine learning algorithms to forecast whether or not it will rain tomorrow.

b)  Design a predictive model with the use of machine learning algorithms to predict how much rainfall could be there.

**Dataset Description:**

Number of columns: **23**

Date  - The date of observation

Location  -The common name of the location of the weather station

MinTemp  -The minimum temperature in degrees celsius

MaxTemp -The maximum temperature in degrees celsius

Rainfall  -The amount of rainfall recorded for the day in mm

Evaporation  -The so-called Class A pan evaporation (mm) in the 24 hours to 9am

Sunshine  -The number of hours of bright sunshine in the day.

WindGustDir- The direction of the strongest wind gust in the 24 hours to midnight

WindGustSpeed -The speed (km/h) of the strongest wind gust in the 24 hours to midnight

WindDir9am -Direction of the wind at 9am

WindDir3pm -Direction of the wind at 3pm

WindSpeed9am -Wind speed (km/hr) averaged over 10 minutes prior to 9am

WindSpeed3pm -Wind speed (km/hr) averaged over 10 minutes prior to 3pm

Humidity9am -Humidity (percent) at 9am

Humidity3pm -Humidity (percent) at 3pm

Pressure9am -Atmospheric pressure (hpa) reduced to mean sea level at 9am

Pressure3pm -Atmospheric pressure (hpa) reduced to mean sea level at 3pm

Cloud9am - Fraction of sky obscured by cloud at 9am.

Cloud3pm -Fraction of sky obscured by cloud

Temp9am-Temperature (degrees C) at 9am

Temp3pm -Temperature (degrees C) at 3pm

RainToday -Boolean: 1 if precipitation (mm) in the 24 hours to 9am exceeds 1mm, otherwise 0

RainTomorrow -The amount of next day rain in mm. Used to create response variable . A kind of measure of the "risk".

We imported the necessary Libraries like pandas, numpy, matplotlib and seaborn. Which is important to any of the machine learning program to execute.

We read the data through pd.read\_csv(‘link of the datatset’). Than our dataset appered on the output box

***Some basic information regarding the dataset***

1. Shape of the dataset:- we will get the shape of the dataset through data.shape(in which data is the name what we have given to the dataset’)

There are 8425 rows and 25 columns

1. data.columns:- which gives all the columns name which are present in the dataset
2. data.info():- which gives us the information that holds the column name, null values if available,and datatype of each column
3. data.describe() :- it gives us the summary of the dataset of all the columns whose datatype is in numeric(mean. Meadian, quantiles of each columns)
4. data.isna().sum() :- checking the null values in whole dataset that gives the individual null values of each column.

Date 0

Location 0

MinTemp 75

MaxTemp 60

Rainfall 240

Evaporation 3512

Sunshine 3994

WindGustDir 991

WindGustSpeed 991

WindDir9am 829

WindDir3pm 308

WindSpeed9am 76

WindSpeed3pm 107

Humidity9am 59

Humidity3pm 102

Pressure9am 1309

Pressure3pm 1312

Cloud9am 2421

Cloud3pm 2455

Temp9am 56

Temp3pm 96

RainToday 240

RainTomorrow 239

Thses are null values that each column holds in the dataset

***Data Engineering on Date Column of dataset :-***

We have seaparated the day, month and year of the Date column into day, month and Year column.

data['Day']=pd.to\_datetime(data['Date'],format='%Y-%m-%d').dt.day

data['Month']=pd.to\_datetime(data['Date'],format='%Y-%m-%d').dt.month

data['Year']=pd.to\_datetime(data['Date'],format='%Y-%m-%d').dt.year

we have used the pd.to\_datetime function to separate the date column of dataset

now we have created the three columns(day, month and year) through Date column than we deleted the Date column because that not in used now.

**Now we can go one by one to each column to see the problem and Rectify it**

1. *Location Column*

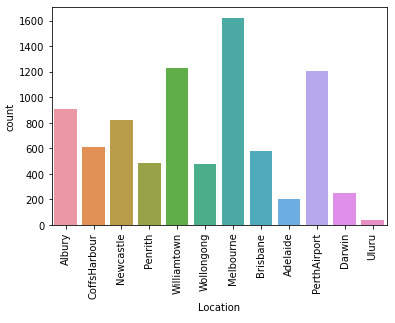
* We have checked the unique values for Location Column on dataset and there are number of location we could find that is

Data(‘Location’).unique()

'Albury', 'CoffsHarbour', 'Newcastle', 'Penrith', 'Williamtown', 'Wollongong',

'Melbourne', 'Brisbane', 'Adelaide', 'PerthAirport’, 'Darwin', 'Uluru'

* We have count the values of each unique value have in location Column of dataset and then we plot the countplot for that to.



* We encoded the values of Location Column because model building cannot understand categorical values it only understand the numerical values.

***MinTemp Column***

* Check the null values of MinTemp Column if there are any null values

Data[‘MinTemp’].isna().sum() :- It will give us the total null values present in MinTemp column

There are 75 Null values in MinTemp Column

* We have checked what kind of data MimTemp column hold and its float type of data so we fill nan values to the mean of the column

Data[‘MinTemp’].fillna(data[‘MinTemp’].mean()

We have treated the null values of MinTemp column and filled it with mean values of that column now the null values is zero

1. ***MaxTemp column***

* Check the null values of MaxTemp Column if there are any null values

Data[‘MaxTemp’].isna().sum() :- It will give us the total null values present in MaxTemp column

There are 60 Null values in MaxTemp Column

* We have checked what kind of data MaxTemp column hold and its float type of data so we fill nan values to the mean of the column

Data[‘MaxTemp’].fillna(data[‘MaxTemp’].mean()

We have treated the null values of MaxTemp column and filled it with mean values of that column now the null values is zero

1. ***Rainfall Column***

* Check the null values of Rainfall Column if there are any null values

Data[‘Rainfall’].isna().sum() :- It will give us the total null values present in Rainfall column

There are 240 Null values in Rainfall Column

* We have checked what kind of data Rainfall column hold and its float type of data so we fill nan values to the mean of the column

Data[‘Rainfall’].fillna(data[‘Rainfall’].mean()

We have treated the null values of Rainfall column and filled it with mean values of that column now the null values is zero

1. ***Evaporation Column***

We have deleted the Evaporation column because the null values in Evaporation column is to high that is 3512, which if we assumed also than the assuming values will be same for all the nan values or for all the missing values so we should better delete this column to make are model more accurate

1. ***Sunshine Column***

We have deleted the Sunshine column because the null values in Sunshine column is to high that is 3994, which if we assumed also than the assuming values will be same for all the nan values or for all the missing values so we should better delete this column to make are model more accurate

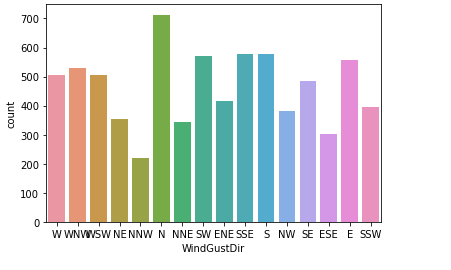
1. ***WindGustDir Column***

* Check the unique values of WindGustDir Column which is:

'W', 'WNW', 'WSW', 'NE', 'NNW', 'N', 'NNE', 'SW', nan, 'ENE', 'SSE', 'S', 'NW', 'SE', 'ESE', 'E', 'SSW'

These are unique values

* Check the null values of WidGustDir Column and there are 991 null values in WindGustDir column so we need to fill them all.
* Then check the count of values in WindGustDir Column



* Now we replaced the null values of WindGustDir Column to the ‘N’ values of the column because the value of ‘N’ are higher than any other values so the chances are higher that it belongs to ‘N’
* After replacing null values filled with ‘N’
* Now we label Encode the values of WindGustDir column into numeric values

1. ***WindGustSpeed Column***

* Check the null values of WindGustSpeed Column which is 991 values
* The values are in numeric form so we fill the null values to the mean value of the column.
* Now the null values is becomes to 0 and filled with mean values of the column

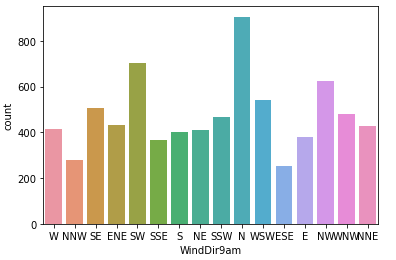
1. ***WindDir9am column***

* Check the Null values of WindDir9am column column which is 829 misiing values of WindDir9am column column.
* Then check the unique values of WindDir9am column column and unique columns are

'W', 'NNW', 'SE', 'ENE', 'SW', 'SSE', 'S', 'NE', nan, 'SSW', 'N',

'WSW', 'ESE', 'E', 'NW', 'WNW', 'NNE'

* Then plot the countplot for WindDir9am column



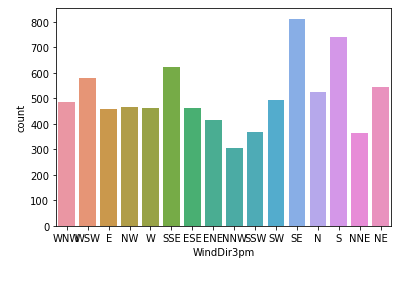
* Now we replaced the null values of WindDir9am Column to the ‘N’ values of the column because the value of ‘N’ are higher than any other values so the chances are higher that it belongs to ‘N’
* After replacing null values filled with ‘N’
* Now we label Encode the values of WindDir9am column into numeric values

1. ***WindDir3pm Column***

* Check the null values of WindDir3pm Column that is 308 null values present in the column
* Check the unique values of WindDir3pm Column which is

'WNW', 'WSW', 'E', 'NW', 'W', 'SSE', 'ESE', 'ENE', 'NNW', 'SSW', 'SW', 'SE', 'N', 'S', 'NNE', nan, 'NE'

* Plot the countplot of WindDir3pm Column to check values count



* Now we replaced the null values of the column to the’SE’ column value because the higherthean a y other value of the column so the chnaces are higher that they belongs to ‘SE’ value of the column

1. ***WindSpeed9am Column***

* Check the null values of WindSpeed9am Column column which is 107 missing values in the column.
* The values that holds the WindSpeed9am Column column is int type value so we need mean value of the column to fill the null values.
* We have filled the null values with the mean of the WindSpeed9am Column column values and null values becomes to Zero.

1. ***Humidity9am Column***

* Check the null values of Humidity9am column which is 102

missing values in the column.

* The values that holds the Humidity9am column is int type value so we need mean value of the column to fill the null values.
* We have filled the null values with the mean of the Humidity9am Column values and null values becomes to Zero.

1. ***Pressure9am Column***

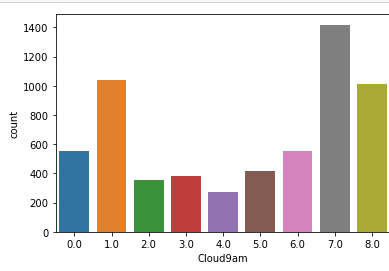
* Check the null values of Pressure9am column which is 1039 missing values in the column.
* The values that holds the Pressure9am column is float type value so we need mean value of the column to fill the null values.
* We have filled the null values with the mean of the Pressure9am Column values and null values becomes to Zero.

1. ***Pressure3pm Column***

* Check the null values of Pressure3am column which is 1312 missing values in the column.
* The values that holds the Pressure3am column is float type value so we need mean value of the column to fill the null values.
* We have filled the null values with the mean of the Pressure3am Column values and null values becomes to Zero.

14, ***Cloud9am column***

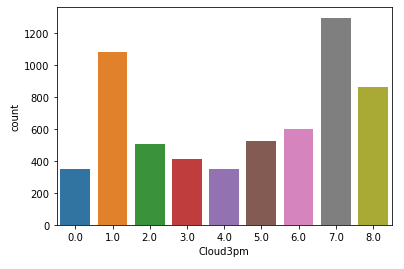
* Check the null values of Cloud9am column which is 2421 missing values present in the Cloud9am column.
* Check the unique values of Cloud9am column i.e 1,2,3,4,5,6,7,8. These are the unique values that Cloud9am column have.
* Plot the countplot to check the value count of the column.



* So we replace the nan values of Cloud9am column to the ‘7’ value of the column because thses are the highest in numbers in the column and chances are higher that they belomgs to that value.

1. ***Cloud3pm Column***

* Check the null values of Cloud3pm Column which is 2455 missing values present in the column.
* Check the unique values of Cloud3am column i.e 1,2,3,4,5,6,7,8. These are the unique values that Cloud9am column have.
* Plot the countplot to check the value count of the column.



* So we replace the nan values of Cloud3am column to the ‘7’ value of the column because thses are the highest in numbers in the column and chances are higher that they belomgs to that value.

1. ***Temp9am Colum***

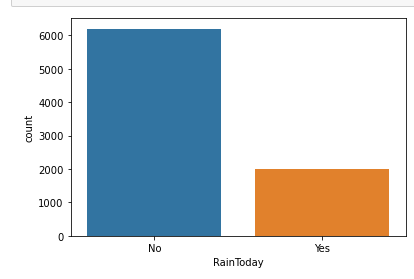
* Check the null values of Temp9am Colum which is near about 56.
* Temp9am Colum holds the float datatype so we filled the null values of the column to the mean of the column.

1. ***Temp3am Colum***

* Check the null values of Temp3am Colum which is near about 96.
* Temp3am Colum holds the float datatype so we filled the null values of the column to the mean of the column.

1. ***RainToday column***

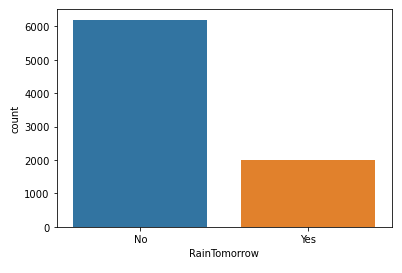
* Check the null values of the RainToday column column which is 240.
* Check the unique values of RainToday column column which is only ‘yes’, ’No’.
* Plot the countplot for RainToday column column to check number of values in the column.



* So the numbers of that ‘No’ has higher number of values so we filled all the missing values of the column to the ‘No’ values of the column.
* Then we encode the values of column yes or no to the numeric form 0,1.

1. ***RainTomorrow Column***

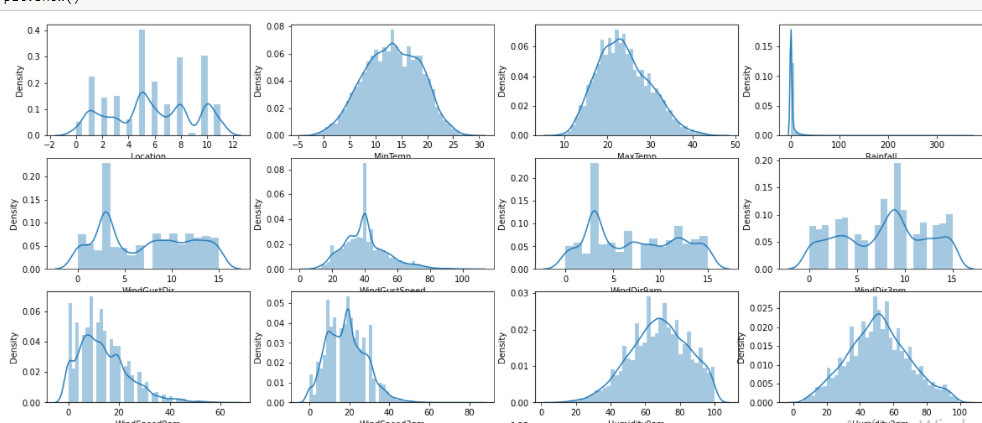
* There are three unique values in RainTomorrow Column i.e ‘No’, ‘Yes’, and nan values.
* Check the value count through countplot of RainTomorrow Column.

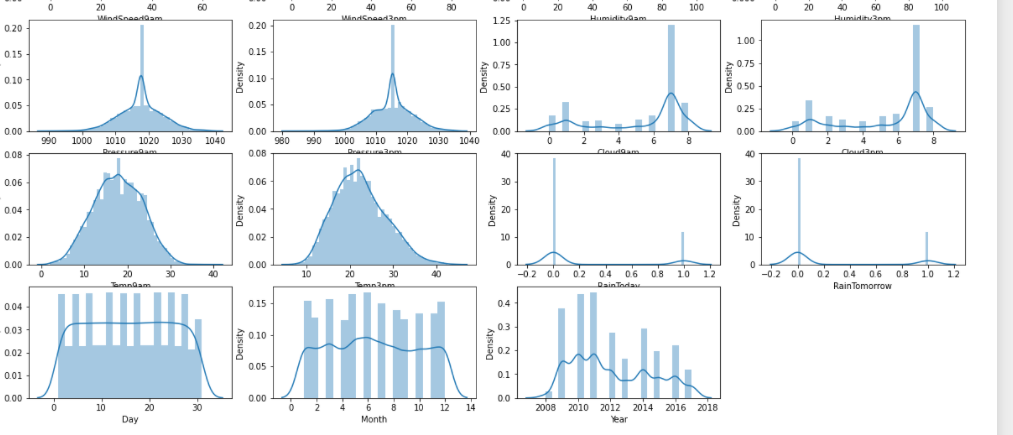


* Value of ‘No’ is higher than value of ‘yes’ so we filled the nan values of the column to the ‘No’ values because chances are higher that they belong to ‘No’.
* After filling missing values we encode the values of column to numeric form where’No’ becomes 0 and ‘yes’ becomes 1.

***Check the normal distribution***

We have checked the distribution to see whether the data are normally distributed or not.





Rainfall are not normally distributed and many more and rainfall have continuous data and other who have nonnormally distribution have 1.2.3. etc that we encoded after label encoder.

***Checking the correlation***

Now we checking the correlation between the columns that if any column correlated to another one so that they share some relationship and we could delete one.

MinTemp and Temp9an are highly correlated which is 89%.

We have check the outliers using boxplot.

And even used the skew() to check the skewness of each column and rectify it if the skewness is more than 0.55.

Rainfall and raintomorrow have the massive outliers in it so we used the condition that if any column of the data have more than 0.55 skewness than it will reduced to it.

Now when the data skewness is rectify so now we can proceed further for splitting the data.

***Splitting the variable for Classification Problem***

We have split the target variable into y and feature variable into x

Than we just checked the shape of x and y by simply using the shape

X=all column except’RainTomorrow’

Y= ‘RainTomorrow’.

***Scaling the x variable***

We have scaled the values of x through standard scaler.

***Train Test Split***

Then we have given the percentage for train\_test\_split methos that how much we have given to the test function in this we have 40% test size.

1. **Logistic Regression**

Training score of Logistic Regression Model- 81%

Testing score of Logistic Regression Model- 80%

* ***Confusion matrix***

array([[2395, 166],

[ 428, 381]]

* **Classification report for Logistic Model**

precision recall f1-score support

0 0.85 0.94 0.89 2561

1 0.70 0.47 0.56 809

accuracy 0.82 3370

macro avg 0.77 0.70 0.73 3370

weighted avg 0.81 0.82 0.81 3370

* Cross Val Score -81% that means it accurate model.

1. ***KNN Model***
2. Training Accuracy of KNN Model -87%
3. Testing Accuracy of KNN Model -82%

After Hyperparameter Tuning of KNN Model

Testing Score for hyoerparameter tuning is 82% which remains the same

* ***Confusion Matrix***

array([[2402, 159],

[ 443, 366]]

* ***Classification Report***

precision recall f1-score support

0 0.84 0.94 0.89 2561

1 0.70 0.45 0.55 809

accuracy 0.82 3370

macro avg 0.77 0.70 0.72 3370

weighted avg 0.81 0.82 0.81 3370

3. ***Decision tree Classifier Model***

Training score of Decision Tree Model is 100%

Testing Score of Decision Tree Model is 81%

Hyperparameter Tuning for decision Tree Classifier Model is 80%

* ***Confusion Matrix***

array([[2449, 112],

[ 541, 268]]

* ***Classification report***

precision recall f1-score support

0 0.82 0.96 0.88 2561

1 0.71 0.33 0.45 809

accuracy 0.81 3370

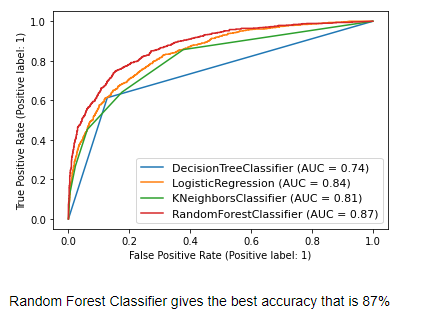
macro avg 0.76 0.64 0.67 3370

weighted avg 0.79 0.81 0.78 3370

1. ***Random forest Classifier***
2. Training Score Random Forest Model: 100%
3. Testing Score of Random Forest Model is : 87%

After hyperparameter tuning the accuracy is 84% that means tuning doesn’t help to improve the accuracy

***Plot Roc curve for all the fitted model***

******

*According to roc curve for the fitted model Random Forest Classifier is the best suitable model for this classification problem.*

**Regression problem: To predict Rainfall.**

--Splitting the data into x and y

X=all the columns except Rainfall

Y= only rainfall column

--scaling the data of x

Through standard sclimg

1. ***Linear Regression Model***

Linear regression Model Accuracy

Training Accuracy= 74%

Testing accuracy =76%

***Model evaluation For Linear Model***

Mean Absolute Error of Linear Regression Model is : 0.2894077860384599

Mean Squared error of Linear Regression Model is : 0.2274315391888936

Root Mean squared error of Linear Regression Model is : 0.4768978288783601

1. ***Decision Tree Regression Model***

Accuarcy of Decision Tree Regressor Model is

Training accuracy= 100%

Testing accuracy = 67%

Hyperparameter tuning -75%

***Model Evalution for Decision Tree Regression Model***

Mean Absolute Error of Decision Tree Regression Model is : 0.24599298083060825

Mean Squared error of Decision Tree Regression Model is : 0.3236132121872063

Root Mean squared error of Decision Tree Regression Model is : 0.5688701189087069

1. ***Random Forest Regressor Model***

Accuarcy of Random Forest Regressor Model is

Training accuracy= 97%

Testing accuracy = 83%

Hyparameter tuning score=80%

***Model Evalution For Random Forest Regressor Model***

Mean Absolute Error of Random forest Regression Model is : 0.21421325229418886

Mean Squared error of Random forest Regression Model is : 0.15641909994315642

Root Mean squared error of Random forest Regression Model is : 0.39549854606958595

In both the cases Random forest Model gives best result so we save the model witb random Forwst model name using pickle

Import pickle

Give file name

Ans pickle.dump to save the file