Blog write by Nandini Singh

Rainfall Prediction - Weather Forecasting



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 **location**and **time**. **Weather forecasts**are made by collecting **quantitative data**about the **current state of the 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:**

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

Some Description about :-Weather forecasting Rainfall

Weather forecasting is the task of forecasting weather conditions for a given location and time. With the use of weather data and algorithms, it is possible to predict weather conditions for the next n number of days. and **how much rainfall could be there**

2. Data Analysis.

Weather forecasting is something that interested me. We all get frustrated when it suddenly rains but we are not prepared. Sometimes we can foresee the weather simply by looking at the sky, but our prediction from examining the sky is not always correct. Before working on this project, I have no idea what factors will affect the possibility of rain the next day. But this dataset provided by “Datatrained” to me with useful information to predict the possibility of rain the next day.

The first thing I did was to explore this dataset. I found that The dataset has 8425 rows and 23 columns. column Rain Tommorow is the target column. Out of these 23 columns, 7 of them are categorical data, 1 is a date, and the rest are all continuous data. The numeric data include temperature, wind direction, humidity, cloud, and pressure at different times of the day, as well as rainfall, evaporation, sunshine. Categorical values are location, wind direction at different times of the day, and whether is it rained today or tomorrow. And the target variable in this dataset is the column Rain Tomorrow.

RangeIndex: 8425 entries, 0 to 8424

Data columns (total 23 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Date 8425 non-null object

1 Location 8425 non-null object

2 MinTemp 8350 non-null float64

3 MaxTemp 8365 non-null float64

4 Rainfall 8185 non-null float64

5 Evaporation 4913 non-null float64

6 Sunshine 4431 non-null float64

7 WindGustDir 7434 non-null object

8 WindGustSpeed 7434 non-null float64

9 WindDir9am 7596 non-null object

10 WindDir3pm 8117 non-null object

11 WindSpeed9am 8349 non-null float64

12 WindSpeed3pm 8318 non-null float64

13 Humidity9am 8366 non-null float64

14 Humidity3pm 8323 non-null float64

15 Pressure9am 7116 non-null float64

16 Pressure3pm 7113 non-null float64

17 Cloud9am 6004 non-null float64

18 Cloud3pm 5970 non-null float64

19 Temp9am 8369 non-null float64

20 Temp3pm 8329 non-null float64

21 RainToday 8185 non-null object

22 RainTomorrow 8186 non-null object

dtypes: float64(16), object(7)

memory usage: 1.5+ MB

I checked whether there are any missing values in the dataset. To my surprise, there are many missing values in this dataset. The percentage of missing values in each column range from 0 percent to 48 percent. For instance, there are 41.69% null values in the Evaporation column and there is no null value for the Location column.and max. null values Sunshine 47.41%

Sunshine 47.41

Evaporation 41.69

Cloud3pm 29.14

Cloud9am 28.74

Pressure3pm 15.57

Pressure9am 15.54

WindGustDir 11.76

WindGustSpeed 11.76

WindDir9am 9.84

WindDir3pm 3.66

RainToday 2.85

Rainfall 2.85

RainTomorrow 2.84

WindSpeed3pm 1.27

Humidity3pm 1.21

Temp3pm 1.14

WindSpeed9am 0.90

MinTemp 0.89

MaxTemp 0.71

Humidity9am 0.70

Temp9am 0.66

Location 0.00

Date 0.00

dtype: float64

Since columns such as Sunshine, Evaporation, Cloud9am, and Cloud3pm have more than 10% missing values, I can simply fill in those by the median or mean values.. I also dropped all the missing values in the column RainTomorrow since it is the target variable for my predictions.

*# filling the null value for numerical variables*

df['MinTemp']**=**df['MinTemp']**.**fillna(df['MinTemp']**.**mean())

df['MaxTemp']**=**df['MinTemp']**.**fillna(df['MaxTemp']**.**mean())

df['Rainfall']**=**df['Rainfall']**.**fillna(df['Rainfall']**.**mean())

df['Evaporation']**=**df['Evaporation']**.**fillna(df['Evaporation']**.**mean())

df['Sunshine']**=**df['Sunshine']**.**fillna(df['Sunshine']**.**mean())

df['WindGustSpeed']**=**df['WindGustSpeed']**.**fillna(df['WindGustSpeed']**.**mean())

df['WindSpeed9am']**=**df['WindSpeed9am']**.**fillna(df['WindSpeed9am']**.**mean())

df['WindSpeed3pm']**=**df['WindSpeed3pm']**.**fillna(df['WindSpeed3pm']**.**mean())

df['Humidity9am']**=**df['Humidity9am']**.**fillna(df['Humidity9am']**.**mean())

df['Humidity3pm']**=**df['Humidity3pm']**.**fillna(df['Humidity3pm']**.**mean())

df['Pressure9am']**=**df['Pressure9am']**.**fillna(df['Pressure9am']**.**mean())

df['Pressure3pm']**=**df['Pressure3pm']**.**fillna(df['Pressure3pm']**.**mean())

df['Cloud9am']**=**df['Cloud9am']**.**fillna(df['Cloud9am']**.**mean())

df['Cloud3pm']**=**df['Cloud3pm']**.**fillna(df['Cloud3pm']**.**mean())

df['Temp9am']**=**df['Temp9am']**.**fillna(df['Temp9am']**.**mean())

df['Temp3pm']**=**df['Temp3pm']**.**fillna(df['Temp3pm']**.**mean())

*#Filling the missing values for categorical variables with mode*

df['RainToday']**=**df['RainToday']**.**fillna(df['RainToday']**.**mode()[0])

df['RainTomorrow']**=**df['RainTomorrow']**.**fillna(df['RainTomorrow']**.**mode()[0])

df['WindDir9am'] **=** df['WindDir9am']**.**fillna(df['WindDir9am']**.**mode()[0])

df['WindGustDir'] **=**df['WindGustDir']**.**fillna(df['WindGustDir']**.**mode()[0])

df['WindDir3pm'] **=** df['WindDir3pm']**.**fillna(df['WindDir3pm']**.**mode()[0])

after do the thing I have no null value data

Date 0

Location 0

MinTemp 0

MaxTemp 0

Rainfall 0

Evaporation 0

Sunshine 0

WindGustDir 0

WindGustSpeed 0

WindDir9am 0

WindDir3pm 0

WindSpeed9am 0

WindSpeed3pm 0

Humidity9am 0

Humidity3pm 0

Pressure9am 0

Pressure3pm 0

Cloud9am 0

Cloud3pm 0

Temp9am 0

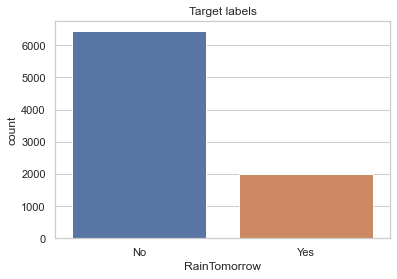
Temp3pm 0

RainToday 0

RainTomorrow 0

dtype: int64

Now there is no null value in the dataset. We have treated all NaN's.after fill all null values analyzing categorical columns.in 'rain today', 'winddir9am','windgustdir','location', 'winddir3pm','raintomorrow'location and rain today have 12 ,2 and other categorical columns have multiple unique values .when thought how many location in dataset 12 different type location .so by grouping the columns by location I observe **Humidity9am is more than Humidity3pm in different location this is clear indication for majorly rain fall happen in morning time between 9am to 3 pm .when I see the target columns (Rain\_tomorrow) have yes aur no variable which say Rain happen aur not .**



In target variable 6434 have no and 1991 yes value for rainfall…

3. EDA Concluding Remark.

Here from the distribution plot we can see most of the columns like mintemp, maxtemp, sunshine, windgustspeed, humidity9am, humidity3pm, pressure9am, pressure3pm, Temp9am and Temp3pm.So the data is not much skewed. It is safe to proceed.when I made countplot from some categorical data in location ,windDir9am , windDir3pm,RainToday,windGustDir “Melbourne” have max possibility of Rain wall.

When I make scatterplot on maxTemp , MinTemp for target columns Rain fall happen possibility increase gradually and same thing happen on humidity9am to Temp9am for rain tomorrow .for know the relation between target columns and other columns in data set I run heatmap to understand the correlation on columns I se humidity9am and cloud9am positively correlated with target columns and pressure9am negatively relate.and There is some multicollinearity among the columns. So we will apply VIF Checking outliers in Numerical columns make boxplot. I see same data have outliers then for batter result Removing Outliers from

Numerical Columns.then after Converting categorical columns using label encoder

4. Pre-Processing Pipeline. And 5. Building Machine Learning Models.Now it is time to make predictions with this dataset. To predict whether it will rain tomorrow, I have used mximum different models to conduct this classification process. They are 1) Logistic Regression, 2) Decision Tree, and 3) Random Forest.4)SVC 5) KNeighborsClassifier6) AdaBoostClassifier7) AdaBoostClassifier All the packages are from sklearn.There are a few steps that I took before I train my models and make predictions. First, I converted all the categorical values to dummy variables. Then, I split my data into training and testing with a 70% and 30% ratio. It turned out that there are 106 columns in total in my training and testing dataset, 79,121 rows of data in my training dataset, and 33,948 in my testing dataset. Since the range is different in each column, each feature's importance is impacted by the scale. For example, if one feature is larger than another feature, then the larger feature has more weight compared to the smaller feature. To deal with this, I used MinMaxScaler to scale the feature range between 1 and 0.

Next, I ran confusion matrix also The accuracy of each model came out as the following:

1.Accuracy score of LogisticRegression() is:0.7666407666407666

2. Accuracy score of GaussianNB() is: 0.7733747733747733

3. Accuracy score of SVC() is:0.8026418026418026

4. Accuracy score of DecisionTreeClassifier() is:0.8425278425278425

5. Accuracy score of KNeighborsClassifier() is:0.851333851333851

6. GradientBoostingClassifier() is:89.12198912198912

After that Let's use Ensemble Technique to boostup score when I This model is overfitting as train score is very high and variance between train and test score is also high.

From above, we can see that the GradientBoosting model has the highest accuracy among these three with an approximately 89.12% accuracy. The KNeighbors model performed well too, only about 3.98% lowest LogisticRegression model. The performance of the all models is GradientBoosting > KNeighborsClassifier > DecisionTreeClassifier> SVC> GaussianNB> LogisticRegression.

To figure out the best parameters for each model. I checking best parameter which suited most for the model validation to determine various parameters to get the highest accuracy. For the Logistic Regression model, I tried different regularization strengths ranging from 0.1 to 100, and the result shows that the accuracy is the highest w For the Decision model. Cross validate of RandomForestClassifier .820 and main score is .74 standard deviation score is .049 Here also after hypertuning, train score is 1 and train score and test score variance is high. This is a case of overfitting when hypertuning is done.so we will select GradiantBoostingClassifier as our final model with 89.12 accuracy score. To illustrate the best model which is the Random Forest model, I have plotted the confusion matrix. There are 1707 true negative values, 208false negative values, 198 false positive values, and 1748 true positive values.

I also found the top 7 most important features of the Gradiant Boosting Classifier model. We can see from the chart below that the most important feature in determining whether it will rain tomorrow is humidity at 3 pm (Humidity3pm). The other important features that also have impacts are wind speed (WindGustSpeed) and pressure at 3 pm (Pressure3pm).

6. Concluding Remarks.

In conclusion, the Gradiant Boosting Classifier model is better in the sense it yields higher accuracy than other models