**Introduction:**

This data is from Kaggle with titles “Rain in Australia”. By gathering quantitative information about the atmosphere's current condition at a specific location. To forecast whether or not it will rain tomorrow, using the Rain Dataset. About 10 years' worth of daily weather measurements from several Australia locales are included in the dataset. Speaking of weather predictions, they might have an impact on daily activities as well as industries like the food sector, tourism, emergency healthcare, etc.

There is a target variable that can be either "Yes" or "No”. Yes, if there was at least 0.1 mm of rain that day. The variables in our dataset that are most likely to cause rain to fall are pressure, humidity, clouds, sunlight and etc.

**Research Question**:

Developing models to achieve great accuracy for this dataset which does have a lot null values to process. To add to that, there are multiple factors to classify rain such are humidity, pressure, temp, cloud, etc. Using these features, aim is to identify if there will be rain tomorrow in binary format ‘Yes’ or ‘No’. In addition, determination of wind direction and speed do affect in causation of rain.

**Unit of Analysis:**

Unit of analysis is important for analysis as it also can be called as unit of observation. For our scenario, a sample collected from environment is unit of analysis. Because, observation of environmental phenomenon such pressure, humidity and others at two specific time like 9am and 3pm as well as other observation of clouds, wind speed and direction, rainfall amount are features. Therefore, unit of analysis can be said as observation of atmosphere.

**Description of Data:**

As illustrated in Fig.1 (Metadata),

- 24-hour format is reshaped from 9 am to 9 am for recording the sample.
- The measure of cloud is fraction of sky covered by cloud, so we can assume that the fraction is in percentage.

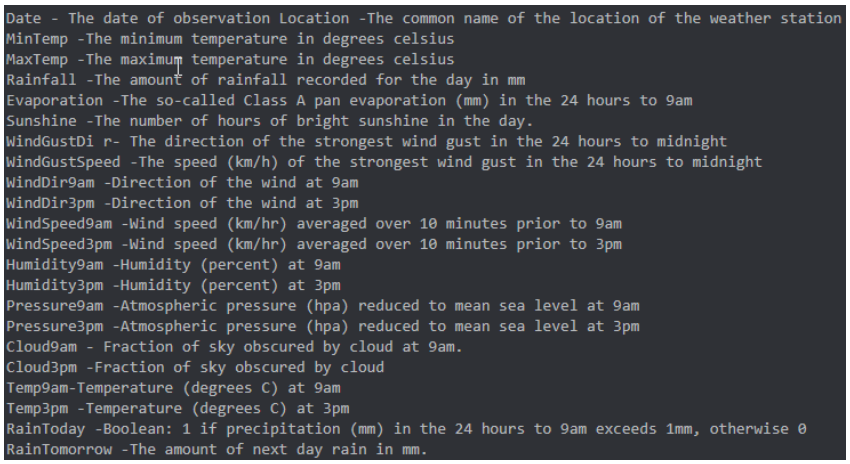



**Fig 1: Metadata**

According to Table 1 (Information of Data types), these are the main 17 feature which eventually can cause a rain. Rain is a natural calamity and we have 145,460 samples. RainTomorrow is out target variable. Due to few numbers of samples in this kind of research problem the performance of the model can be underfitted.

**Table 1 Information of Data type**

| # | Column | Non-Null Count | Dtype |
|---|---|---|---|
| 0 | MinTemp | 145460 non-null | float64 |
| 1 | MaxTemp | 145460 non-null | float64 |
| 2 | Rainfall | 145460 non-null | float64 |
| 3 | Evaporation | 145460 non-null | float64 |
| 4 | Sunshine | 145460 non-null | float64 |
| 5 | WindGustSpeed | 145460 non-null | float64 |
| 6 | WindSpeed9am | 145460 non-null | float64 |
| 7 | WindSpeed3pm | 145460 non-null | float64 |
| 8 | Humidity9am | 145460 non-null | float64 |
| 9 | Humidity3pm | 145460 non-null | float64 |
| 10 | Pressure9am | 145460 non-null | float64 |
| 11 | Pressure3pm | 145460 non-null | float64 |
| 12 | Cloud9am | 145460 non-null | float64 |
| 13 | Cloud3pm | 145460 non-null | float64 |
| 14 | Temp9am | 145460 non-null | float64 |
| 15 | Temp3pm | 145460 non-null | float64 |
| 16 | RainToday | 145460 non-null | int64 |

As demonstrated in the Table 2 (Description of Data),

- the difference in humidity at 9 am and 3 pm is significant as humidity reduce from 69 to 52 while both recorded maximum 100.
- Whereas for pressure, there is slight difference for mean pressure at 9 am to 3pm.
- It has recorded that maximum 9% fraction of sky was covered by cloud at specific time frame.
- 371 mm rainfall has been registered. And this amount of rain can cause flood like disaster.

**Table 2 Description of Data**

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| MinTemp | 145460.0 | 12.194317 | 6.364469 | -5.950000 | 7.700000 | 12.100000 | 16.800000 | 30.450000 |
| MaxTemp | 145460.0 | 23.225145 | 7.067566 | 2.700000 | 18.000000 | 22.700000 | 28.200000 | 43.500000 |
| Rainfall | 145460.0 | 0.381674 | 0.608638 | 0.000000 | 0.000000 | 0.000000 | 0.600000 | 1.500000 |
| Evaporation | 145460.0 | 5.095891 | 1.709594 | 1.797653 | 4.000000 | 5.468232 | 5.468232 | 7.670579 |
| Sunshine | 145460.0 | 7.922535 | 1.386787 | 5.977944 | 7.611178 | 7.611178 | 8.700000 | 10.333234 |
| WindGustSpeed | 145460.0 | 39.716321 | 12.174937 | 8.500000 | 31.000000 | 39.000000 | 46.000000 | 68.500000 |
| WindSpeed9am | 145460.0 | 13.952432 | 8.555347 | 0.000000 | 7.000000 | 13.000000 | 19.000000 | 37.000000 |
| WindSpeed3pm | 145460.0 | 18.576025 | 8.442192 | 0.000000 | 13.000000 | 18.662657 | 24.000000 | 40.500000 |
| Humidity9am | 145460.0 | 68.932605 | 18.703608 | 18.000000 | 57.000000 | 69.000000 | 83.000000 | 100.000000 |
| Humidity3pm | 145460.0 | 51.539116 | 20.471189 | 0.000000 | 37.000000 | 51.539116 | 65.000000 | 100.000000 |
| Pressure9am | 145460.0 | 1017.676878 | 6.568430 | 1001.050000 | 1013.500000 | 1017.649940 | 1021.800000 | 1034.250000 |
| Pressure3pm | 145460.0 | 1015.274311 | 6.528871 | 998.650000 | 1011.100000 | 1015.255889 | 1019.400000 | 1031.850000 |
| Cloud9am | 145460.0 | 4.447461 | 2.265604 | 0.000000 | 3.000000 | 4.447461 | 6.000000 | 9.000000 |
| Cloud3pm | 145460.0 | 4.544125 | 2.026092 | 1.000000 | 4.000000 | 4.509930 | 6.000000 | 9.000000 |
| Temp9am | 145460.0 | 16.991738 | 6.440803 | -1.500000 | 12.300000 | 16.800000 | 21.500000 | 35.300000 |
| Temp3pm | 145460.0 | 21.685669 | 6.812734 | 2.450000 | 16.700000 | 21.400000 | 26.200000 | 40.450000 |
| RainToday | 145460.0 | 0.351430 | 0.477419 | 0.000000 | 0.000000 | 0.000000 | 1.000000 | 1.000000 |

**Exploratory Data Analysis:**

**Table 3: Percentage of Null value in Features.**

```
Percentages of Null values in Features :
Sunshine         48.01
Evaporation      43.17
Cloud3pm         40.81
Cloud9am         38.42
Pressure9am      10.36
Pressure3pm      10.33
WindDir9am        7.26
WindGustDir       7.10
WindGustSpeed     7.06
Humidity3pm       3.10
WindDir3pm        2.91
Temp3pm           2.48
RainTomorrow      2.25
Rainfall          2.24
RainToday         2.24
WindSpeed3pm      2.11
Humidity9am       1.82
Temp9am           1.21
WindSpeed9am      1.21
MinTemp           1.02
MaxTemp           0.87
```

The description of data serves as a brief summary of an individual variable which let us know standard deviation, mean, quartiles null values etc. This helps us to have a basic knowledge of our dataset and also let us know where data cleaning is required.

It is strongly advised to have a clean dataset in order to precede, for that same, we have just figured out the percentage of null values in the research.

The existence of these features, which might have been absent at the time, could be the cause of these null values.

These null values were treated by imputing median for numeric values after checking for its normal distribution and the null values for categorical values were treated with mode.

| | Date | Location | MinTemp | MaxTemp | Rainfall | Evaporation | Sunshine | WindGustDir | WindGustSpeed | WindDir9am | WindDir3pm | WindSpeed9am |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 2008-12-01 | Albury | 13.4 | 22.9 | 0.6 | NaN | NaN | W | 44.0 | W | WNW | 20.0 |
| 1 | 2008-12-02 | Albury | 7.4 | 25.1 | 0.0 | NaN | NaN | WNW | 44.0 | NNW | WSW | 4.0 |
| 2 | 2008-12-03 | Albury | 12.9 | 25.7 | 0.0 | NaN | NaN | WSW | 46.0 | W | WSW | 19.0 |
| 3 | 2008-12-04 | Albury | 9.2 | 28.0 | 0.0 | NaN | NaN | NE | 24.0 | SE | E | 11.0 |
| 4 | 2008-12-05 | Albury | 17.5 | 32.3 | 1.0 | NaN | NaN | W | 41.0 | ENE | NW | 7.0 |

| WindSpeed3pm | Humidity9am | Humidity3pm | Pressure9am | Pressure3pm | Cloud9am | Cloud3pm | Temp9am | Temp3pm | RainToday | RainTomorrow |
|---|---|---|---|---|---|---|---|---|---|---|
| 24.0 | 71.0 | 22.0 | 1007.7 | 1007.1 | 8.0 | NaN | 16.9 | 21.8 | No | No |
| 22.0 | 44.0 | 25.0 | 1010.6 | 1007.8 | NaN | NaN | 17.2 | 24.3 | No | No |
| 26.0 | 38.0 | 30.0 | 1007.6 | 1008.7 | NaN | 2.0 | 21.0 | 23.2 | No | No |
| 9.0 | 45.0 | 16.0 | 1017.6 | 1012.8 | NaN | NaN | 18.1 | 26.5 | No | No |
| 20.0 | 82.0 | 33.0 | 1010.8 | 1006.0 | 7.0 | 8.0 | 17.8 | 29.7 | No | No |

**Fig 2: First 5 rows of Data frame**

Fig 2 shows the first 5 rows that are present in the data frame.

In our dataset we are focusing and aligning our resources to a target variable which is: **RainTomorrow** Analyzing data of RainTomorrow we came up with amount of percentage of their values; which are:

No 75.839406

Yes 21.914616

NaN 2.245978

NaN represent not a number which is treated as null and we need to remove these samples containing null s. It's approx. 2% so we can remove it from the data frame. Replacement of null value from our point of v iew should not be done as it is our target variable.

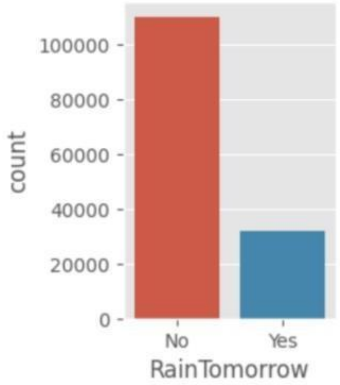



Fig 3 (Count of each value) depicts the count of individual variable of the same attribute Number of count of NOs is relatively more than that of YES.

**Fig 3: Count of each value Eda for Categorical**

**value**

Now, EDA is performed on categorical vales of the dataset. Out of 23 total attribute we have 7 categorical values. Fig 4 represents the attribute in category. We also analysed distributions of WindDir3pm, WindDir9am and WindGustDir have shown same response as uniform distribution. Date and Location are not in consideration for this classification. And wind direction variables are not really impactful to rain.

```
Number of variables:  7
Variables : ['Date', 'Location', 'WindGustDir', 'WindDir9am', 'WindDir3pm', 'RainToday', 'RainTomorrow']
```

**Fig.4: Categorical Variable**

Table 4 displays the top 10 values of the categorical variables.

**Table 4: Top 10 values of Categorical variables**

| | Date | Location | WindGustDir | WindDir9am | WindDir3pm | RainToday | RainTomorrow |
|---|---|---|---|---|---|---|---|
| 0 | 2008-12-01 | Albury | W | W | WNW | No | No |
| 1 | 2008-12-02 | Albury | WNW | NNW | WSW | No | No |
| 2 | 2008-12-03 | Albury | WSW | W | WSW | No | No |
| 3 | 2008-12-04 | Albury | NE | SE | E | No | No |
| 4 | 2008-12-05 | Albury | W | ENE | NW | No | No |
| 5 | 2008-12-06 | Albury | WNW | W | W | No | No |
| 6 | 2008-12-07 | Albury | W | SW | W | No | No |
| 7 | 2008-12-08 | Albury | W | SSE | W | No | No |
| 8 | 2008-12-09 | Albury | NNW | SE | NW | No | Yes |
| 9 | 2008-12-10 | Albury | W | S | SSE | Yes | No |

## Cardinality

The number of distinct values allocated to a dimension is referred to as its cardinality. A certain number of distinct values are specified for some dimensions. Cardinality checking is important because it specifies the relation between other categorical variables also, if we discover high cardinality in a dataset then it will provide an extremely large matrix which in result makes building of model extremely difficult or it will cause under-fitting. Depicting from fig 7 we can see that location has the highest count of unique categories which we will use for advance classification.

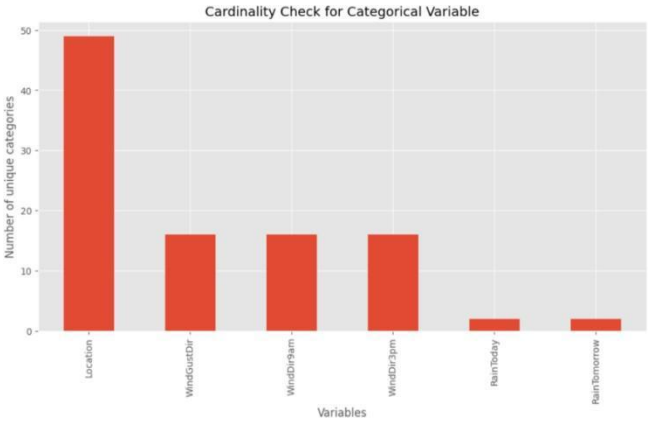



**Fig 5: Cardinality check for Categorical value**

```
Null % in categorical variables:
WindDir9am       7.264
WindGustDir      7.099
WindDir3pm       2.907
RainTomorrow     2.246
RainToday        2.242
Date             0.000
Location         0.000
```

For cleaning Rainfall and RainToday of the categorical variables, we have to be sure that as we cannot just simply impute the data with mode and median only as it depends on the rainfall.

**Fig 6: Null percentage in categorical variables**

**Table 5: Cleaning of rainfall**

| | Rainfall | RainToday |
|---|---|---|
| **0** | 0.6 | Yes |
| **1** | 0.0 | No |
| **2** | 0.0 | No |
| **3** | 0.0 | No |
| **4** | 1.0 | Yes |
| **...** | ... | ... |
| **145455** | 0.0 | No |
| **145456** | 0.0 | No |
| **145457** | 0.0 | No |
| **145458** | 0.0 | No |
| **145459** | 0.0 | No |

As per the procedure, we corrected Null value to median of Rainfall and if the value of Rainfall is greater than 0.0, we are classifying RainToday as yes.

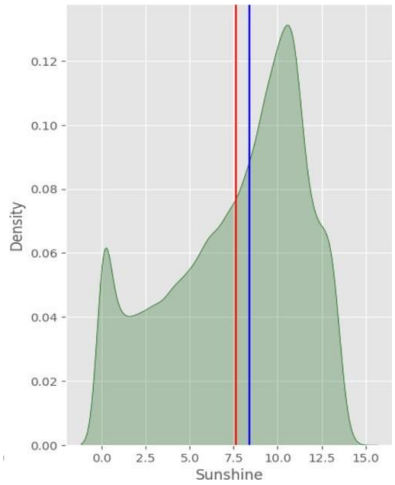
**EDA for Numerical Features:**

Firstly, we display name of the numerical data and thereby along with that we also defined each integer to a data type equal to float64.

```
['MinTemp',
 'MaxTemp',
 'Rainfall',
 'Evaporation',
 'Sunshine',
 'WindGustSpeed',
 'WindSpeed9am',
 'WindSpeed3pm',
 'Humidity9am',
 'Humidity3pm',
 'Pressure9am',
 'Pressure3pm',
 'Cloud9am',
 'Cloud3pm',
 'Temp9am',
 'Temp3pm']
```

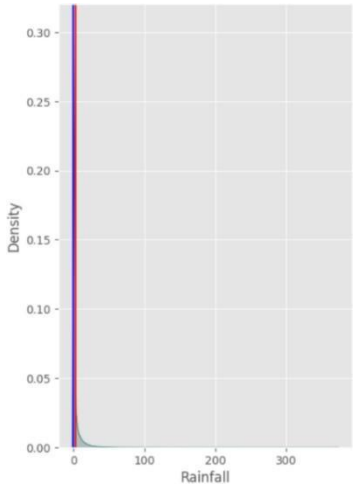
Clean data is necessary which is input for our model. When it comes to imputation of missing values for numeric data, we need to check the distribution. This selection of method is determined on the basis of distribution of each variable.

**Fig 7: Numerical Value**




In Fig 8 (Distribution of sunshine), the graph is unevenly distributed and weighted right side. And we can see that red line which is mean of the sunshine is present at the left side and hence we opt for mean to impute data.

**Fig 8: Distribution of sunshine**




Here in Fig.9 (Distribution of Rainfall), we can see that data is having 0 for a lot of samples. Hence, for this mean and median both on the same line hence any of that is preferred to get imputed with. But replacing median is better choice as 1 mm of rainfall can affect RainToday variable and it should be classified as yes.

**Fig 9: Distribution of rainfall**

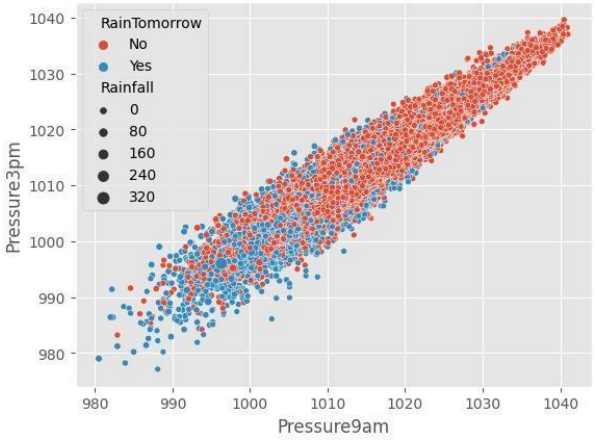



**Fig.10 Pressure at 9am vs 3pm**

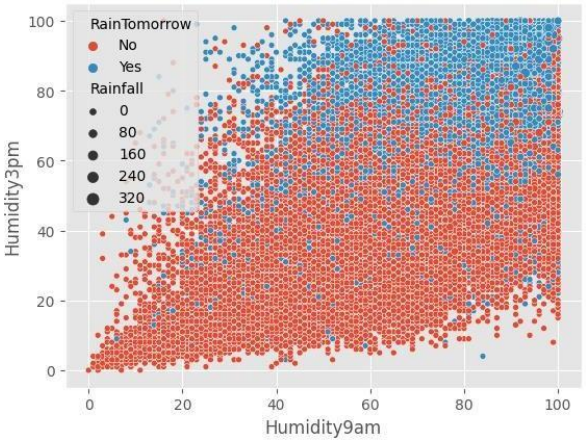



**Fig.11 Humidity at 9am vs 3pm**

Fig.10 and Fig.11 demonstrates some patterns such as pressure less than 1015 at 9am and 3pm can cause a good chance of rain while there are high chances of rain when humidity is greater than 60 at 3pm.

**Collinearity and Feature Engineering**

Primarily, we need to factorise features into numeric. To add to that, we need to plot a heat map or correlation chart, to determine the collinearity between the features.

As these features are realistic there is hardly strong correlation between the variable. So, we need to calculate the variation inflation factor (VIF) to identity multicollinearity between each variable. There is categorical variable with more unique value or cardinality, it can be ambiguous to understand for collinearity.

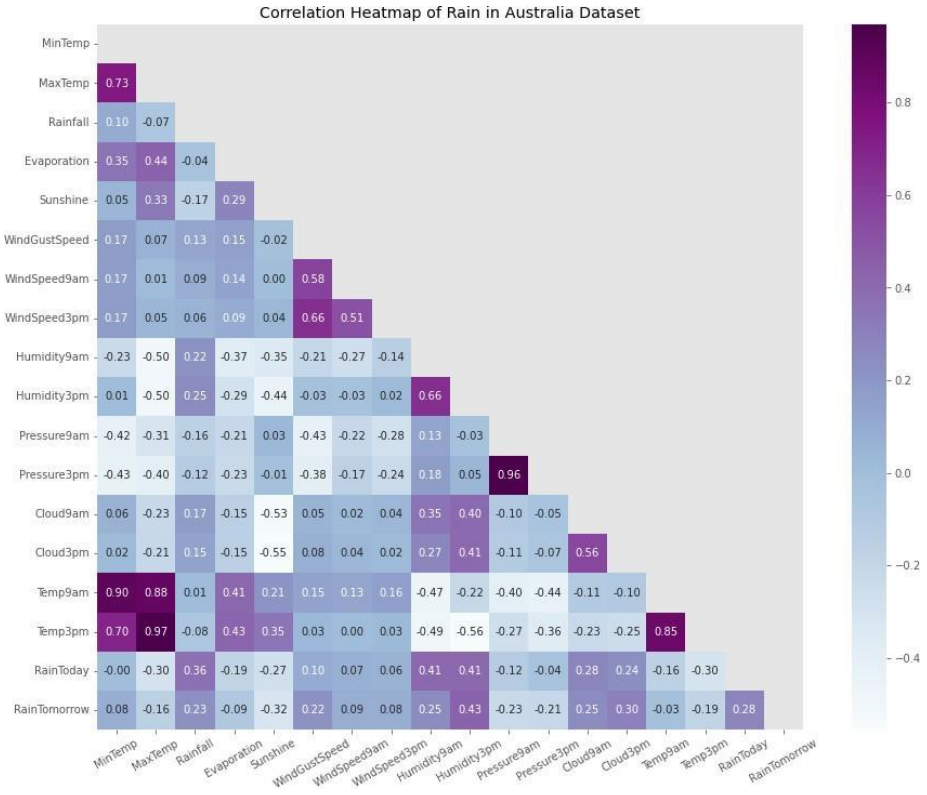



**Fig 12: Heatmap**

From the Fig 12, we can see different collinearity between the variable, there are variables which are having high positive collinearity and cause a rain such as Humidity, Temp and pressure and there are some negative such as sunshine and evaporation.

**Table 6: VIF Factor**

| | VIF | Features |
|---|---|---|
| 0 | 32040.544068 | Intercept |
| 1 | 10.028495 | MinTemp |
| 2 | 43.550294 | MaxTemp |
| 3 | 1.159974 | Rainfall |
| 4 | 2.201594 | Evaporation |
| 5 | 3.243247 | Sunshine |
| 6 | 4.027628 | Humidity9am |
| 7 | 6.611909 | Humidity3pm |
| 8 | 19.692239 | Pressure9am |
| 9 | 19.812638 | Pressure3pm |
| 10 | 2.221865 | Cloud9am |
| 11 | 2.275500 | Cloud3pm |
| 12 | 52.071299 | Temp3pm |
| 13 | 22.818551 | Temp9am |

If value of VIF greater than 5 then it indicates higher potential correlation between the variables. This is not reliable for categorical data if that has cardinality.

VIF is generally used for regression models but for learning co linearity we can use it.

**Methodology for Research Problems:**

Variables of Interest are as illustrated in Fig.13 which are also out dependent variable

```
['MinTemp', 'MaxTemp', 'Rainfall', 'Evaporation', 'Sunshine',
 'WindGustSpeed', 'WindSpeed9am', 'WindSpeed3pm', 'Humidity9am',
 'Humidity3pm', 'Pressure9am', 'Pressure3pm', 'Cloud9am', 'Cloud3pm',
 'Temp9am', 'Temp3pm', 'RainToday'],
```

**Fig 13: Variable of Interest**

As our independent variable is RainTomorrow which has only 2 unique values such as "yes" and "*no"* and achieving our research question classification method such as Logistic Regression, Random Forest and Gradient boosting are suitable. All of them are classification algorithm.

XG Boost, Gradient boosting and Random Forest are advanced approaches to the decision tree algorithm. There is significant difference in all algo which can help to achieve greater accuracy.

**Predictive Analysis:**

he data of 145460 samples were initially divided into train and test splits with a threshold of 0.3, meaning that the index selected 30% of the dataframe for testing and was chosen at random from the dataframe for the index. As a result, the models will be trained using 70% of the samples.

We will use this data to input 3 different models such as Random Forest Classifier, XGBoost and Gradient Boosting Classifier.

As we can deduct from Table 6, XGBoost's accuracy and F1 score are the greatest among the three, and as it represents overall model performance by assessing the model's precision and recall, it should be preferred. Additionally, false positives and false negatives are scored along with precision and recall.

F1 score has value in range of 0 to 1 and higher is better for model.

**Table 6: Models with Accuracy and F1 Score**

| | Classifier | Accuracy | F1 Score |
|---|---|---|---|
| 0 | Random Forest | 85.31 | 0.595 |
| 1 | Hypertuned Random Forest | 85.51 | 0.601 |
| 2 | XGBoost | 85.72 | 0.617 |
| 3 | Gradient Boosting | 85.08 | 0.592 |

We also used some hyper parameter tuning in order to get higher accruary

**Confusion matrixes:**

As seen in Figs. 14, 15, 16 and 17, there are roughly 10–11% false negatives and 3–4% false positives, which is a significant amount given that external factors, such as global warming and deforestation, might have negative influence the cause of rain.

There is very less prediction difference in Random Forest (RF) and Gradient Boosting (GB). From RF to XGBoost got improved accuracy for classifying rain.

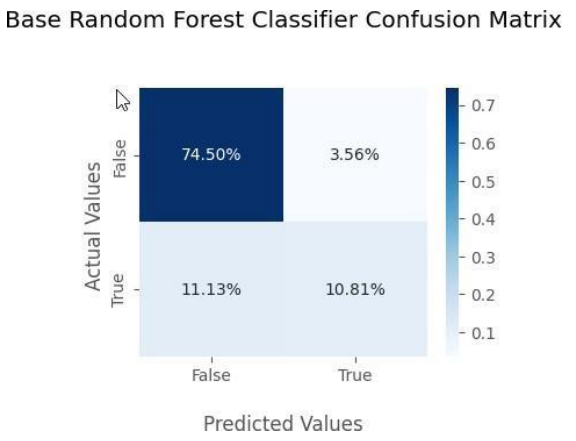



**Fig: 14 CF of Random Forest**

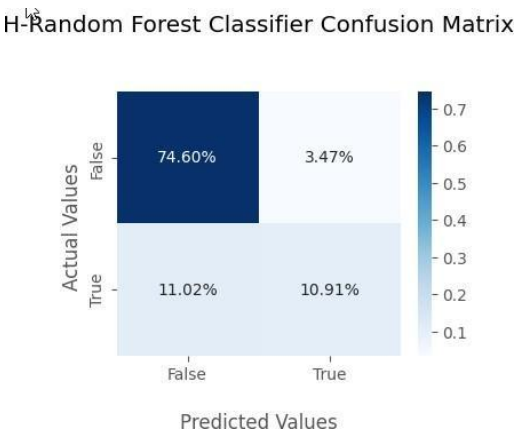



**Fig: 15 CF of Hyper tunned Random Forest**

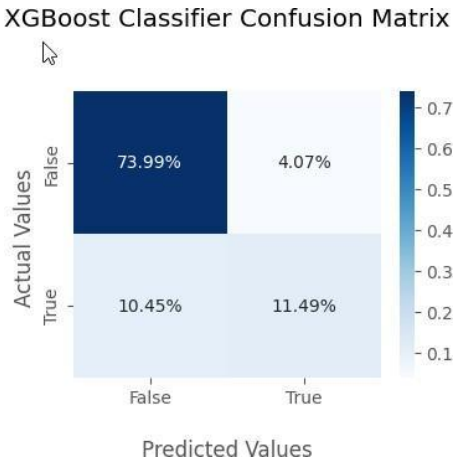



**Fig: 16 CF of XG Boost**

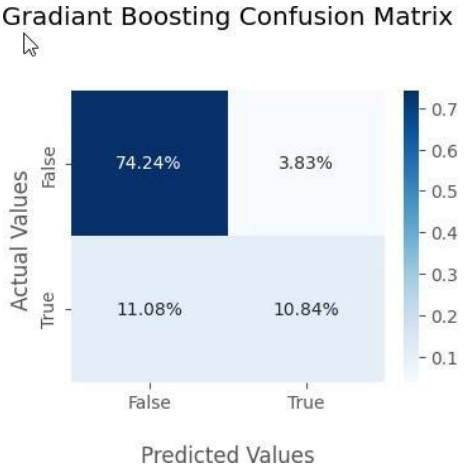



**Fig: 17 CF of Gradient Boosting**

**AUC curves:**

The ROC (Receiver Operating Characteristic) curve can be used to determine the true positive or true negative rate of a prediction made using a model. This makes assessing how well a regression model fits

the data simpler for us. The ROC curve's AUC is used to evaluate the model's specificity and sensitivity (Area under Curve). The closer the AUC value is to 1, the better the model matches the data.

As seen by the AUC curves in Figs. 17, 18, and 19, they are nearly identical with only tiny differences visible; nevertheless, the AUC of XGBoost is 0.7384, which is noteworthy.

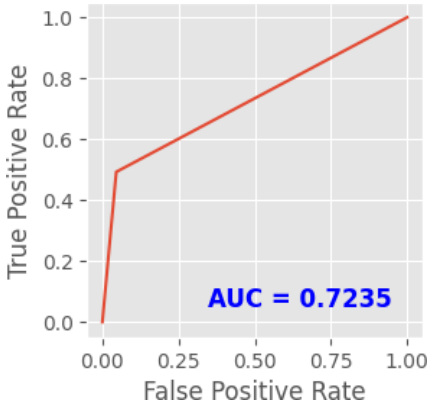



**Fig.17 AUC of Random Forest**

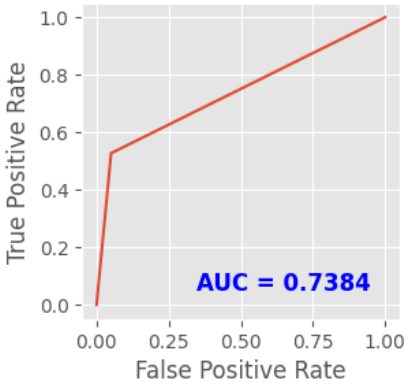



**Fig.18 AUC of XG Boost**

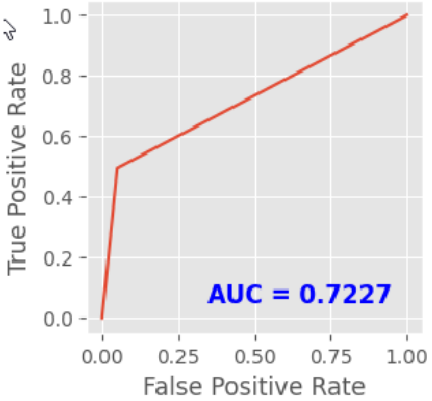



**Fig.19 AUC of Gradient Boosting**

**Table 7 Performance of XGBoost**

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.88 | 0.95 | 0.91 | 34066 |
| 1 | 0.75 | 0.53 | 0.62 | 9572 |
| accuracy | | | 0.86 | 43638 |
| macro avg | 0.81 | 0.74 | 0.76 | 43638 |
| weighted avg | 0.85 | 0.86 | 0.85 | 43638 |

## Conclusion

As we evaluate 3 different model having different statistical approaches and by tuning parameters and we achieved highest accuracy of 85.72 where as F1 score suggests perfectly classified values by XGBoost. This process included base Random Forest (without parameters), tuned Random Forest, XGBoost and Gradient Boosting. For more accuracy, tried out wind direction variables making dummies but didn't improved. Due to high number of samples having 'No' which is denoted as '0' in Table 7 as raining, model is able to classify it to 91% and in 91%, 88% are perfectly predicted (Table 7).

In nutshell, model considering XGBoost took time to train and get that accuracy, it was worth as it is having really good score. There are very less number of falses classification so it fits well considering quality of data as it had a lot of null values.

**References**

[1] M. (2022, October 9). *GitHub - Mxnxn/Rain_Forecasting_Australia*. GitHub. October 30th 2022, from https://github.com/Mxnxn/Rain_Forecasting_Australia.

[2] Z., & posts by Zach, V. A. (2020, July 20). *How to Calculate VIF in Python - Statology*. Statology. Retrieved October 30th 2022, from. https://www.statology.org/variance-inflation-factor-r/.

[3] Manish Pathak (Nov 2019), Using XGBoost in Python Tutorial, Retrieved on October 30th 2022, from https://www.datacamp.com/tutorial/xgboost-in-python.

[4] Vagif Aliyev ( Sep 5, 2020), Gradient Boosting Classification explained through Python, October 30th 2022, from https://towardsdatascience.com/gradient-boosting-classification-explained-through-python-60cc980eeb3d.