Faculty of engineering

Cairo university

Big Data Project

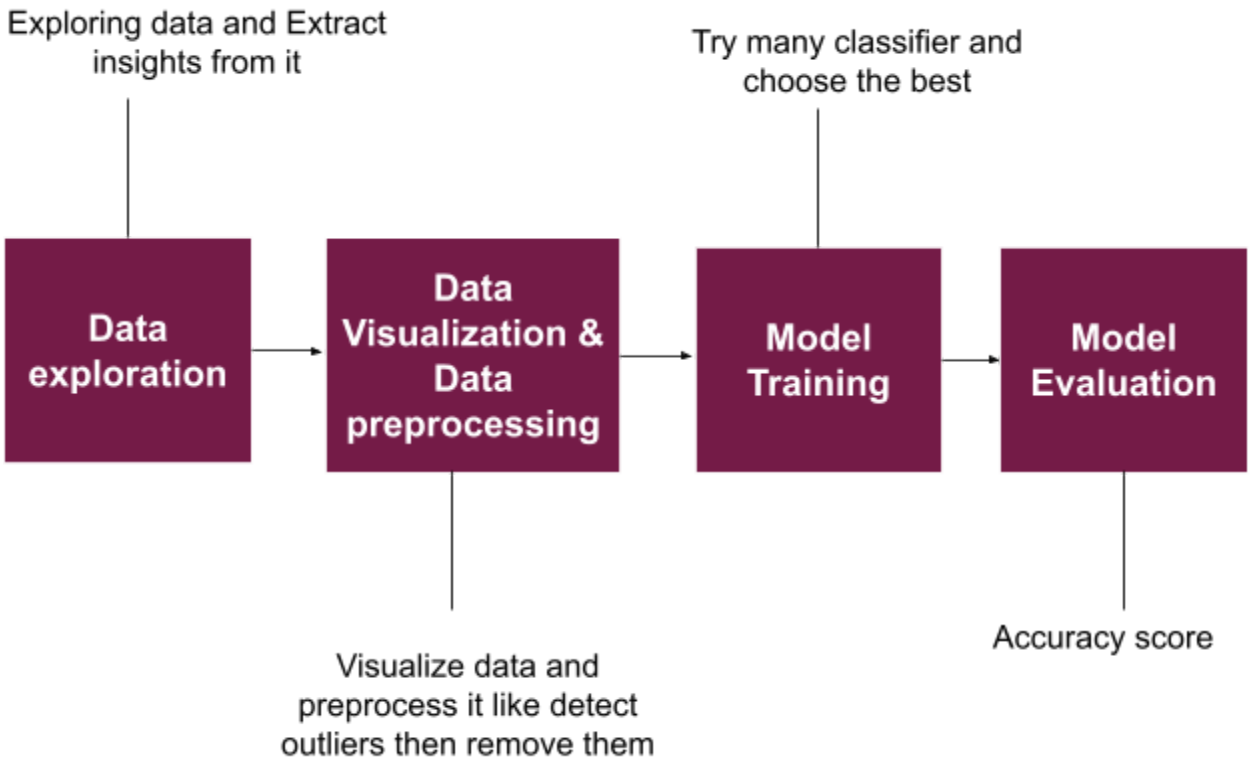
Team 14

1. Brief problem description

Predict if it will rain tomorrow! Clearly, it is a classification problem. The output will be 2 classes - either YES or NO.

Predicting raining tomorrow based on some parameters like Location,temperature, Rainfall, Evaporation, Sunshine, Wind direction, Wind Speed, Humidity, Cloud, Pressure.

2. Project Pipeline



3. Data exploration and extracting insights from data

3.1 Read data set and Create data frame

```
df = pd.read_csv('/content/drive/MyDrive/BigData/weatherAUS.csv')
```

3.2 Checking first 5 rows

```
df.head()
```

Date	Location	MinTemp	MaxTemp	Rainfall	Evaporation	Sunshine	WindGustDir	WindGustSpeed	WindDir9am	WindDir3pm	WindSpeed9am	WindSpeed3pm
2008-12-01	Albury	13.4	22.9	0.6	NaN	NaN	W	44.0	W	WNW	20.0	24.0
2008-12-02	Albury	7.4	25.1	0.0	NaN	NaN	WNW	44.0	NNW	WSW	4.0	22.0
2008-12-03	Albury	12.9	25.7	0.0	NaN	NaN	WSW	46.0	W	WSW	19.0	26.0
2008-12-04	Albury	9.2	28.0	0.0	NaN	NaN	NE	24.0	SE	E	11.0	9.0
2008-12-05	Albury	17.5	32.3	1.0	NaN	NaN	W	41.0	ENE	NW	7.0	20.0

WindSpeed9am	WindSpeed3pm	Humidity9am	Humidity3pm	Pressure9am	Pressure3pm	Cloud9am	Cloud3pm	Temp9am	Temp3pm	RainToday	RainTomorrow
20.0	24.0	71.0	22.0	1007.7	1007.1	8.0	NaN	16.9	21.8	No	No
4.0	22.0	44.0	25.0	1010.6	1007.8	NaN	NaN	17.2	24.3	No	No
19.0	26.0	38.0	30.0	1007.6	1008.7	NaN	2.0	21.0	23.2	No	No
11.0	9.0	45.0	16.0	1017.6	1012.8	NaN	NaN	18.1	26.5	No	No
7.0	20.0	82.0	33.0	1010.8	1006.0	7.0	8.0	17.8	29.7	No	No

3.3 Shape of DataFrame

```
print(f'The number of rows are {df.shape[0] }')
print(f'The number of columns are {df.shape[1] }')
```

```
The number of rows are 145460
The number of columns are 23
```

3.4 Describing the attributes

6 columns are of type 'object' and remaining of type 'float'

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 145460 entries, 0 to 145459
Data columns (total 23 columns):
 #   Column         Non-Null Count   Dtype  
---  ------         --------------   -----  
 0   Date           145460 non-null  object 
 1   Location       145460 non-null  object 
 2   MinTemp        143975 non-null  float64
 3   MaxTemp        144199 non-null  float64
 4   Rainfall       142199 non-null  float64
 5   Evaporation    82670 non-null   float64
 6   Sunshine       75625 non-null   float64
 7   WindGustDir    135134 non-null  object 
 8   WindGustSpeed  135197 non-null  float64
 9   WindDir9am     134894 non-null  object 
 10  WindDir3pm     141232 non-null  object 
 11  WindSpeed9am   143693 non-null  float64
 12  WindSpeed3pm   142398 non-null  float64
 13  Humidity9am    142806 non-null  float64
 14  Humidity3pm    140953 non-null  float64
 15  Pressure9am    130395 non-null  float64
 16  Pressure3pm    130432 non-null  float64
 17  Cloud9am       89572 non-null   float64
 18  Cloud3pm       86102 non-null   float64
 19  Temp9am        143693 non-null  float64
 20  Temp3pm        141851 non-null  float64
 21  RainToday      142199 non-null  object 
 22  RainTomorrow   142193 non-null  object 
dtypes: float64(16), object(7)
memory usage: 25.5+ MB
```

3.5 Description of data

	MinTemp	MaxTemp	Rainfall	Evaporation	Sunshine	WindGustSpeed	WindSpeed9am	WindSpeed3pm
count	143975.000000	144199.000000	142199.000000	82670.000000	75625.000000	135197.000000	143693.000000	142398.000000
mean	12.194034	23.221348	2.360918	5.468232	7.611178	40.035230	14.043426	18.662657
std	6.398495	7.119049	8.478060	4.193704	3.785483	13.607062	8.915375	8.809800
min	-8.500000	-4.800000	0.000000	0.000000	0.000000	6.000000	0.000000	0.000000
25%	7.600000	17.900000	0.000000	2.600000	4.800000	31.000000	7.000000	13.000000
50%	12.000000	22.600000	0.000000	4.800000	8.400000	39.000000	13.000000	19.000000
75%	16.900000	28.200000	0.800000	7.400000	10.600000	48.000000	19.000000	24.000000
max	33.900000	48.100000	371.000000	145.000000	14.500000	135.000000	130.000000	87.000000

Humidity9am	Humidity3pm	Pressure9am	Pressure3pm	Cloud9am	Cloud3pm	Temp9am	Temp3pm
142806.000000	140953.000000	130395.00000	130432.000000	89572.000000	86102.000000	143693.000000	141851.00000
68.880831	51.539116	1017.64994	1015.255889	4.447461	4.509930	16.990631	21.68339
19.029164	20.795902	7.10653	7.037414	2.887159	2.720357	6.488753	6.93665
0.000000	0.000000	980.50000	977.100000	0.000000	0.000000	-7.200000	-5.40000
57.000000	37.000000	1012.90000	1010.400000	1.000000	2.000000	12.300000	16.60000
70.000000	52.000000	1017.60000	1015.200000	5.000000	5.000000	16.700000	21.10000
83.000000	66.000000	1022.40000	1020.000000	7.000000	7.000000	21.600000	26.40000
100.000000	100.000000	1041.00000	1039.600000	9.000000	9.000000	40.200000	46.70000

The above table show for each attribute in the dataset, the Count number of non-NA/null observations (count), mean of values (mean), standard deviation (std), minimum value (min), maximum value (max), and the 25th, 50th, and 75th percentiles (25%, 50%, 75%)

3.6 Showing if in this data null columns

There are many columns that have null values

```
df.isnull()
```

	Date	Location	MinTemp	MaxTemp	Rainfall	Evaporation	Sunshine	WindGustDir	WindGustSpeed	WindDir9am	WindDir3pm	WindSpeed9am	WindSpeed3pm
0	False	False	False	False	False	True	True	False	False	False	False	False	False
1	False	False	False	False	False	True	True	False	False	False	False	False	False
2	False	False	False	False	False	True	True	False	False	False	False	False	False
3	False	False	False	False	False	True	True	False	False	False	False	False	False
4	False	False	False	False	False	True	True	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...
145455	False	False	False	False	False	True	True	False	False	False	False	False	False
145456	False	False	False	False	False	True	True	False	False	False	False	False	False
145457	False	False	False	False	False	True	True	False	False	False	False	False	False
145458	False	False	False	False	False	True	True	False	False	False	False	False	False
145459	False	False	False	True	False	True	True	True	True	False	False	False	False

145460 rows × 23 columns

Humidity9am	Humidity3pm	Pressure9am	Pressure3pm	Cloud9am	Cloud3pm	Temp9am	Temp3pm	RainToday	RainTomorrow
False	False	False	False	False	True	False	False	False	False
False	False	False	False	True	True	False	False	False	False
False	False	False	False	True	False	False	False	False	False
False	False	False	False	True	True	False	False	False	False
False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...
False	False	False	False	True	True	False	False	False	False
False	False	False	False	True	True	False	False	False	False
False	False	False	False	True	True	False	False	False	False
False	False	False	False	False	False	False	False	False	False
False	False	False	False	False	False	False	False	False	True

3.7 Get number of unique value in each column

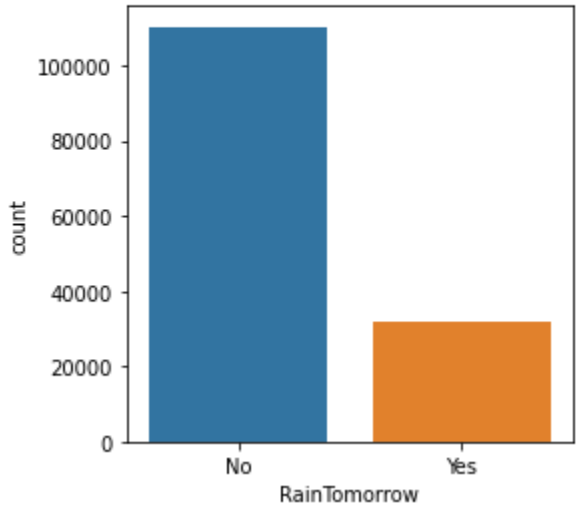
```
df.nunique()
```

```
Date             3436
Location           49
MinTemp           389
MaxTemp           505
Rainfall          681
Evaporation       358
Sunshine          145
WindGustDir        16
WindGustSpeed      67
WindDir9am         16
WindDir3pm         16
WindSpeed9am       43
WindSpeed3pm       44
Humidity9am       101
Humidity3pm       101
Pressure9am       546
Pressure3pm       549
Cloud9am           10
Cloud3pm           10
Temp9am           441
Temp3pm           502
RainToday           2
RainTomorrow        2
dtype: int64
```

3.8 Percentage of will it rain and not and check balancing

- 77.5% data will not rain tomorrow
- Also this figure show that this dataset is imbalanced

```
No     0.775819
Yes    0.224181
Name: RainTomorrow, dtype: float64
<matplotlib.axes._subplots.AxesSubplot at 0x7f3f6019fb90>
```

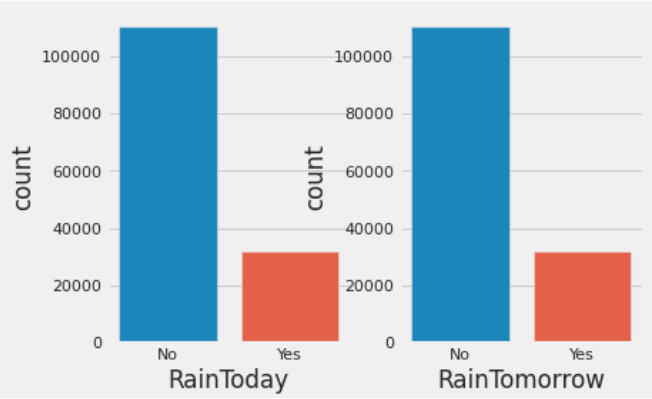


4. Data visualization

4.1 Count of rain today and tomorrow

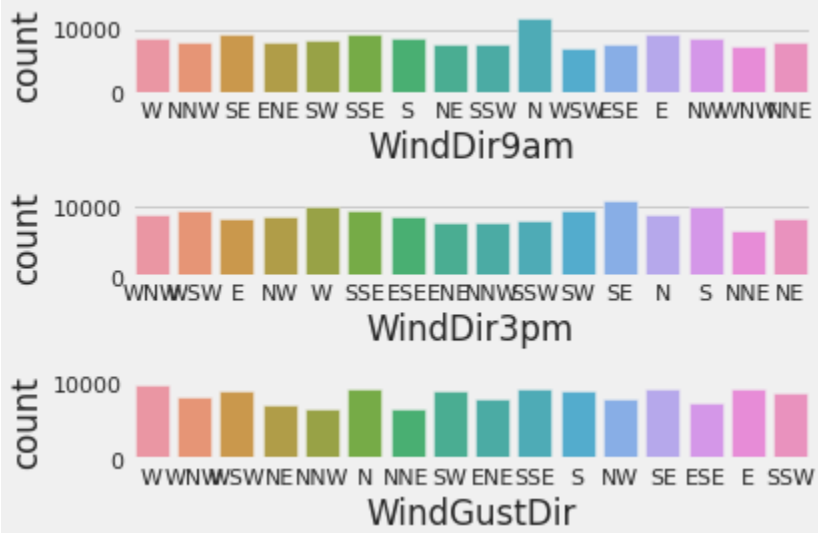
It is obvious that when it rain today it will most probably rain tomorrow

```
No     110319
Yes     31880
Name: RainToday, dtype: int64
No     110316
Yes     31877
Name: RainTomorrow, dtype: int64
<matplotlib.axes._subplots.AxesSubplot at 0x7ff730ed7dd0>
```



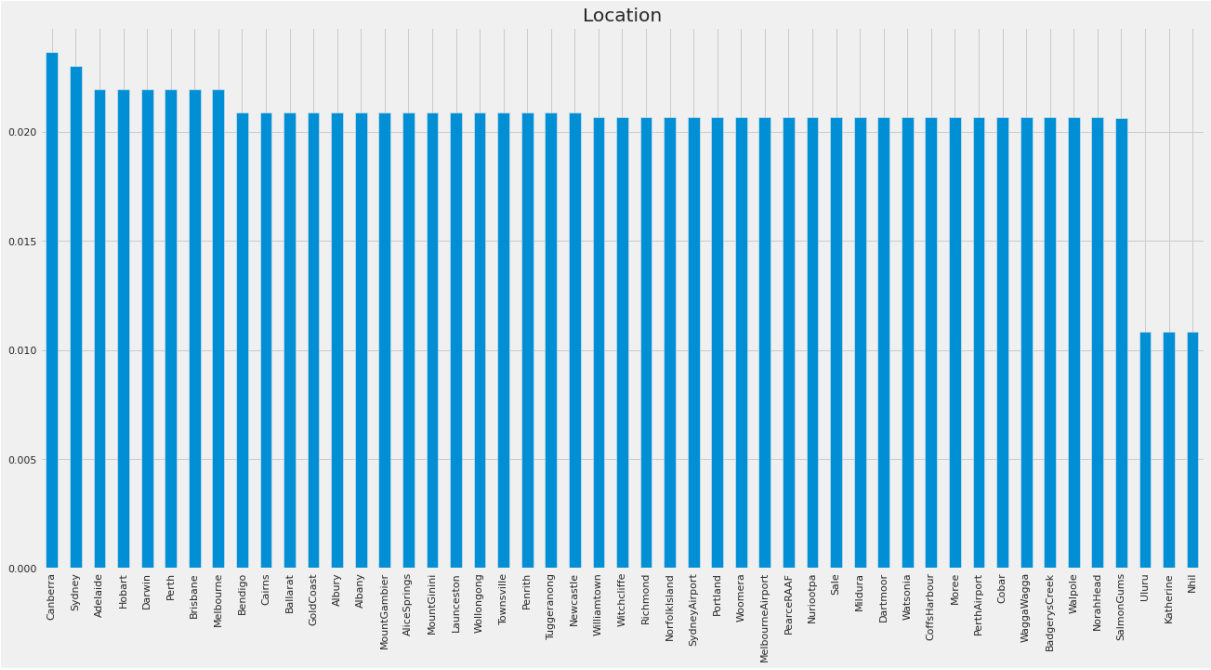
4.2 Direction of wind at 9 am, 3 pm and windGistDir

- Wind at 9 am is highest in direction N.
- Wind at 3 pm is highest in direction SE.
- WindGustDir is highest in direction W.



4.3 Location

The highest location in canberra

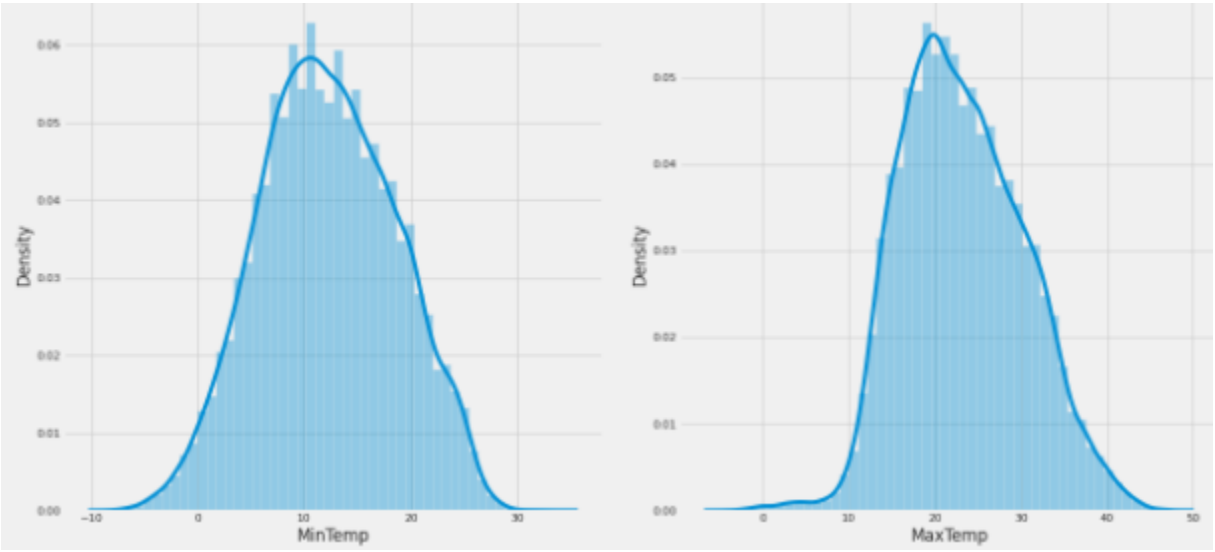


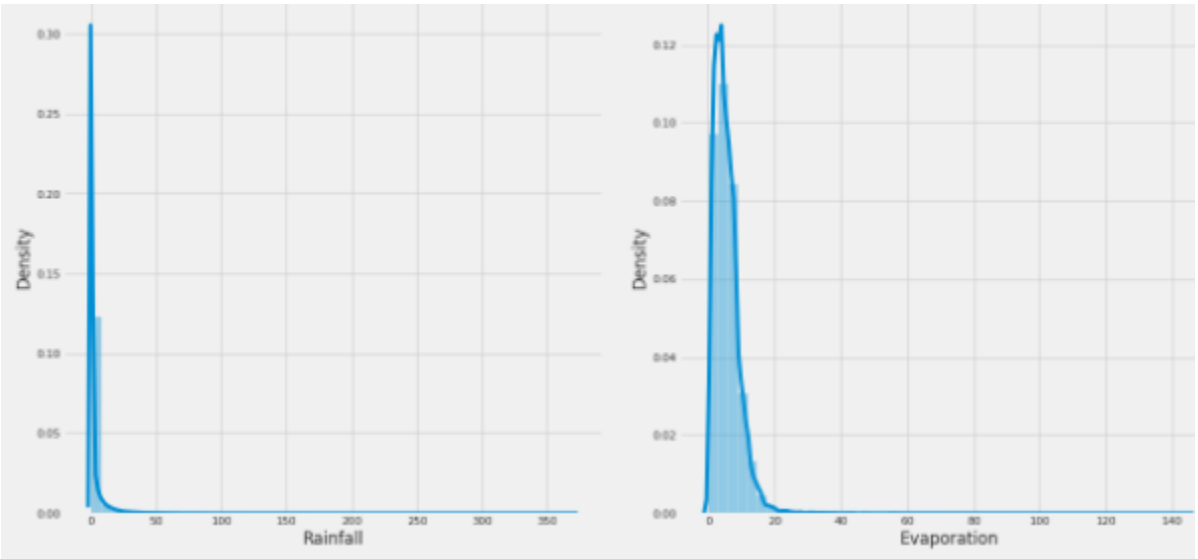
4.4 Distribution plots

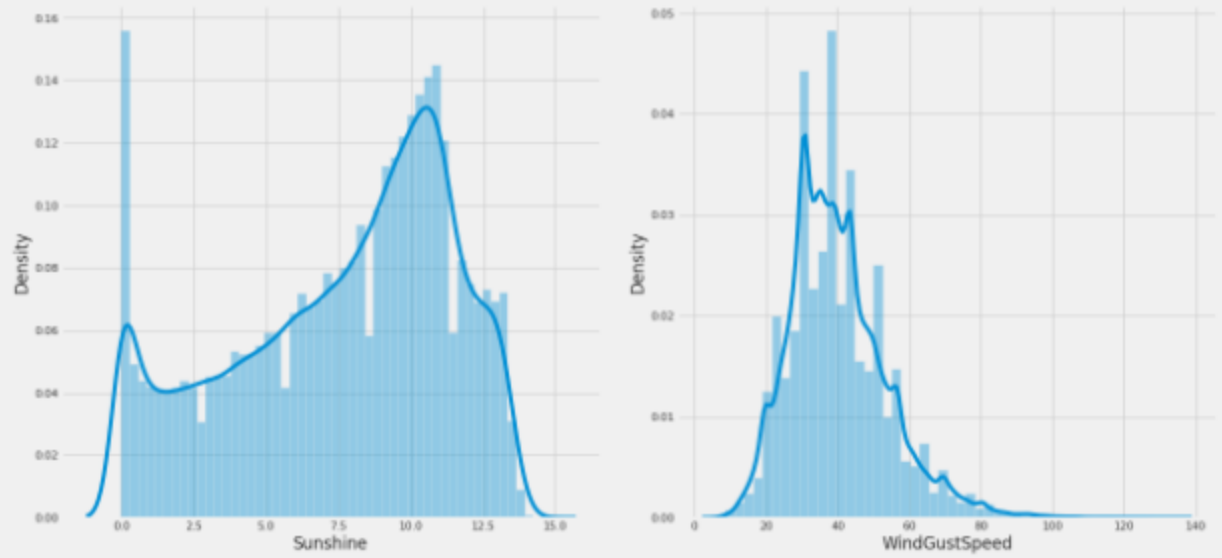
Plot shows the distribution of numerical columns.

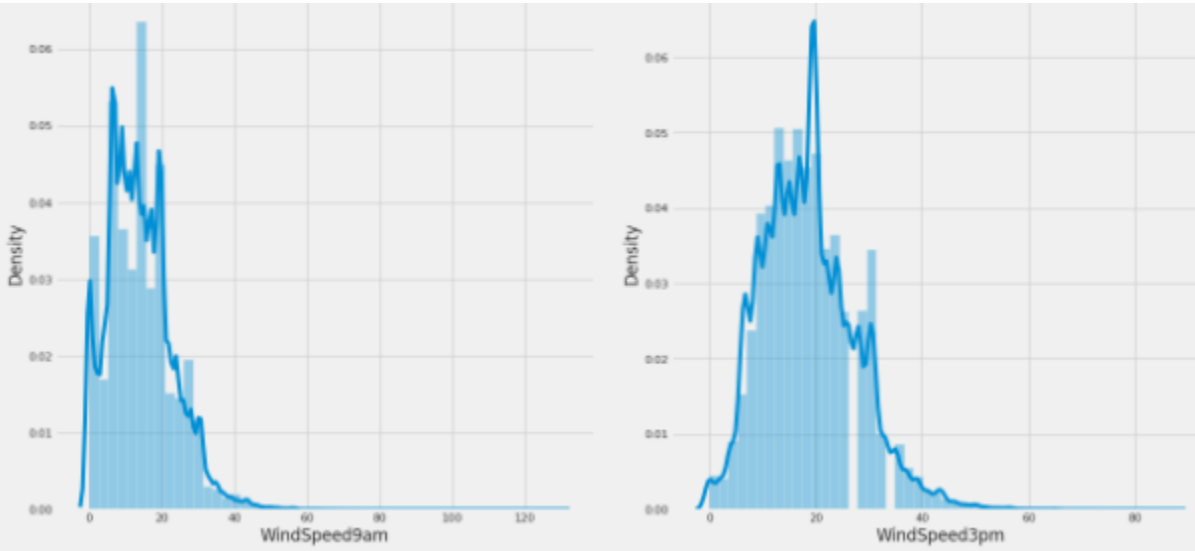
MinTemp: As in the MinTemp column most of the values are between 6 and 19 degrees.

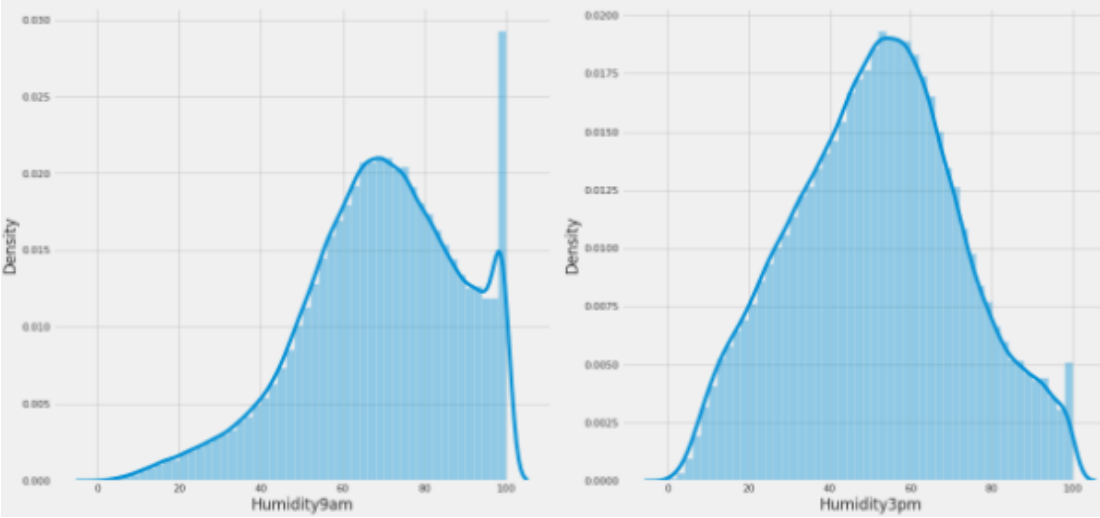
MaxTemp: As in the MinTemp column most of the values are between 20 and 30 degrees.

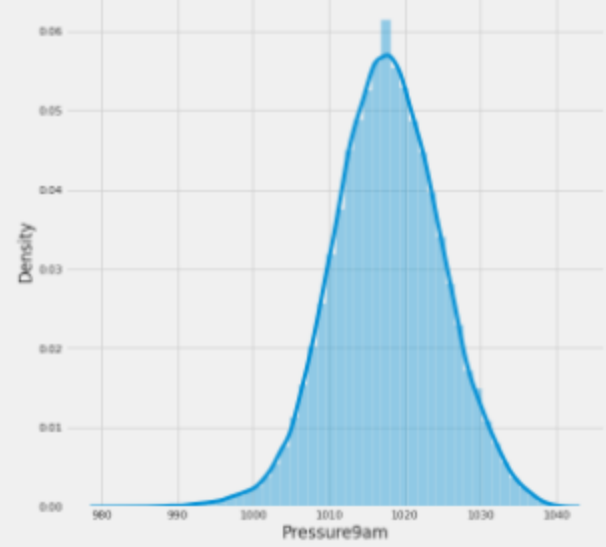


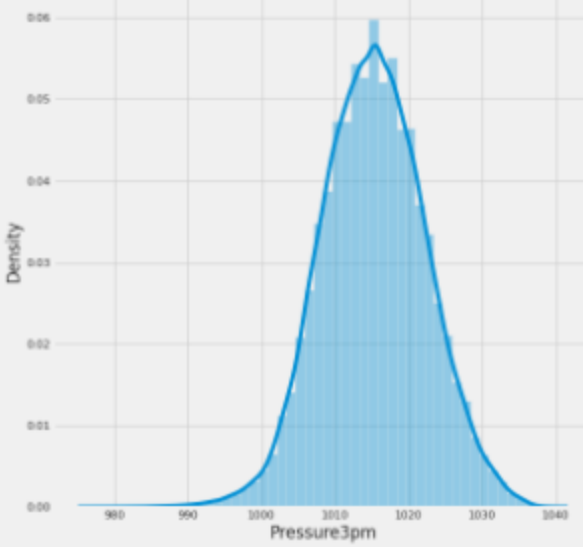


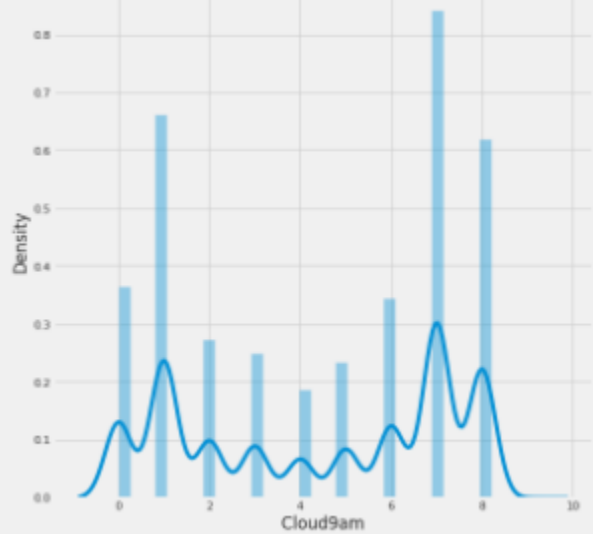

Density
Sunshine
Density
WindGustSpeed

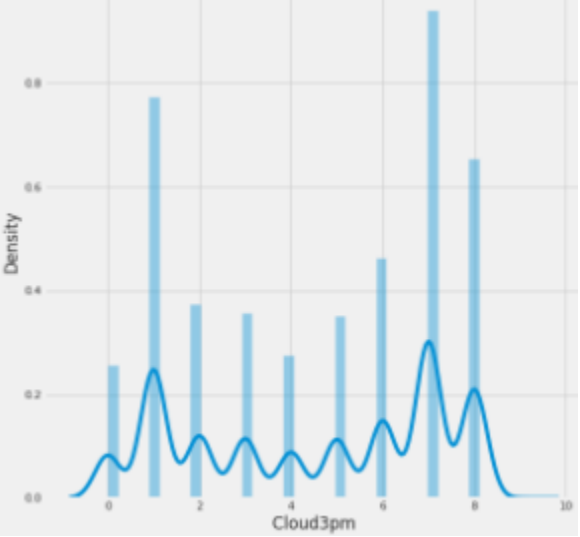

Density
WindSpeed9am
Density
WindSpeed3pm

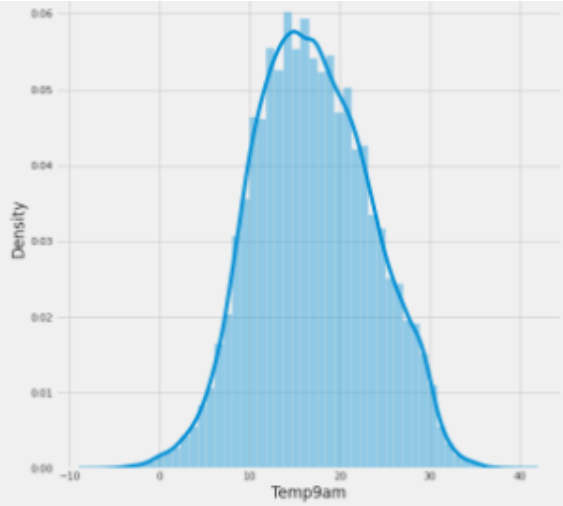

Density
Humidity9am
Density
Humidity3pm

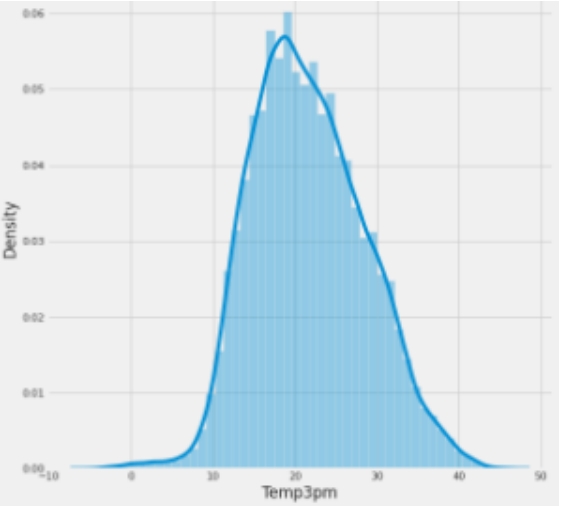

Density
Pressure9am


Density
Pressure3pm


Density
Cloud9am


Density
Cloud3pm

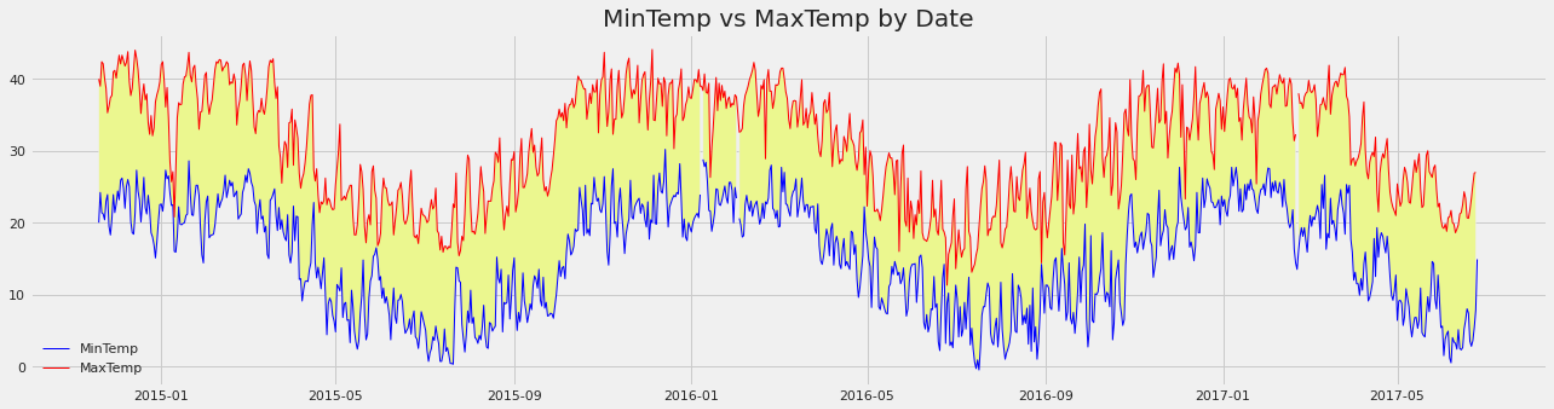

Density
Temp9am


Density
Temp3pm

4.5 Date Plots

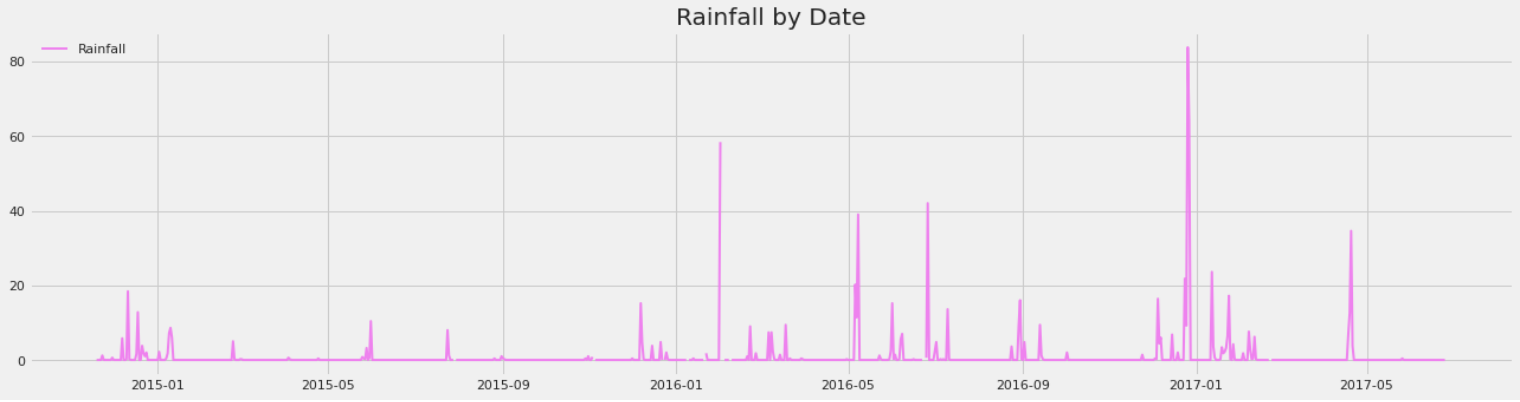
MinTemp and MaxTemp

- The plot shows that the MinTemp and MaxTemp relatively increases and decreases every year.
- The weather conditions are always opposite in the two hemispheres. As Australia is situated in the southern hemisphere. The seasons are a bit different.
- As you can see, December to February is summer; March to May is autumn; June to August is winter; and September to November is spring.



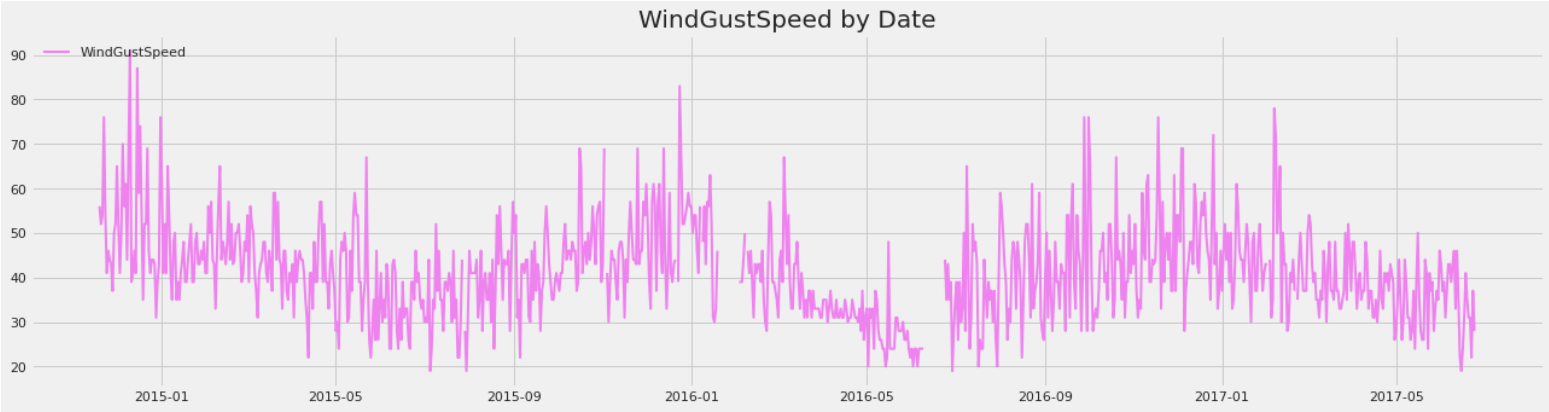
RainFall

- Being situated in the southern hemisphere, the majority of rainfall occurs between December and March.
- We can see that Dec-Jan does get a lot of rainfall but there are months like Jun-Jul when rainfall occurs too.



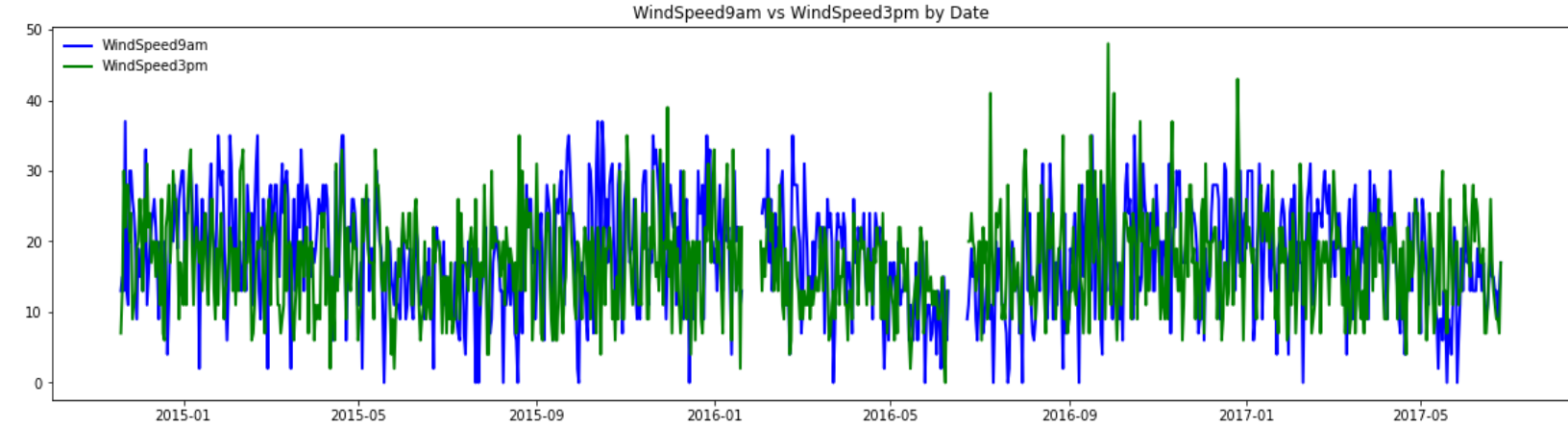
WindGustSpeed

In Australia wind speed is usually moderate. But from the plot we can see that Dec-Feb is the windiest months.



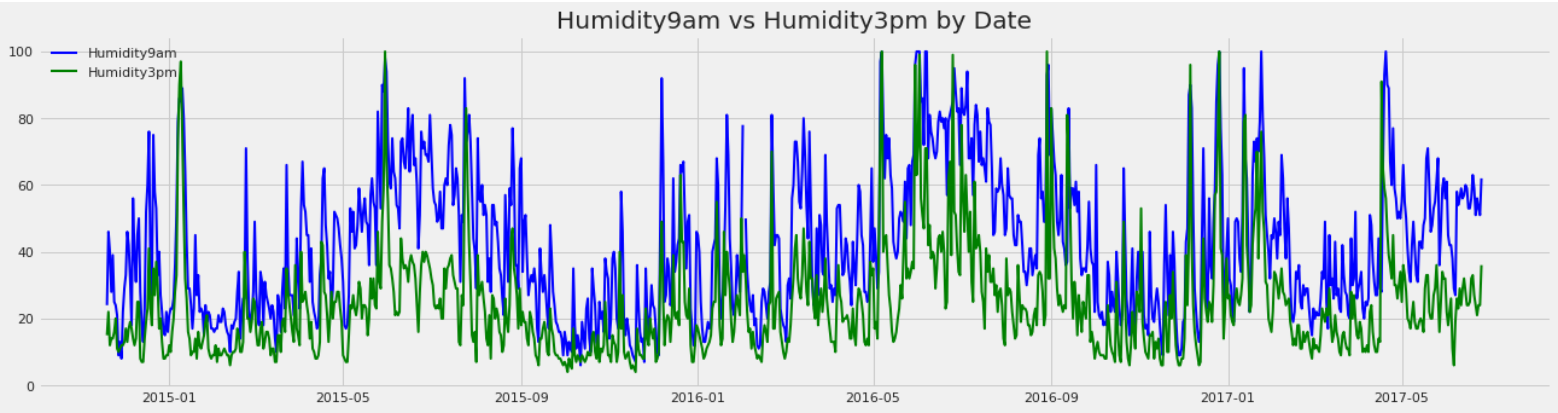
WindSpeed9am and WindSpeed3pm

WindSpeed9am and WindSpeed3pm are relatively the same around certain months.



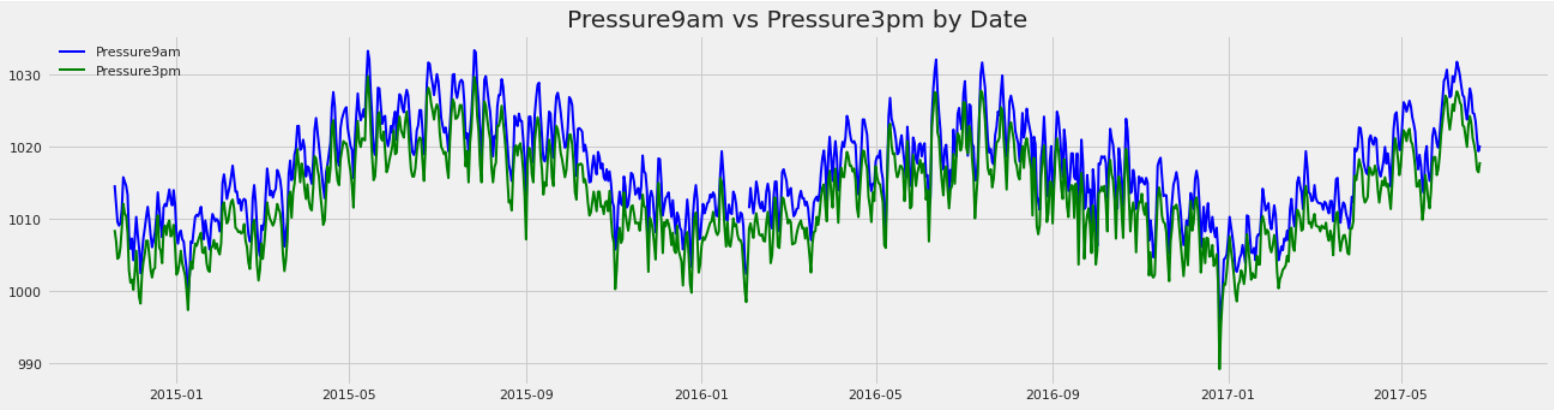
Humidity9am and Humidity3pm

From the plot we can see that the Humidity is high around Jun-Jul and also during that time, there is a good difference between humidity around 9am and 3pm.



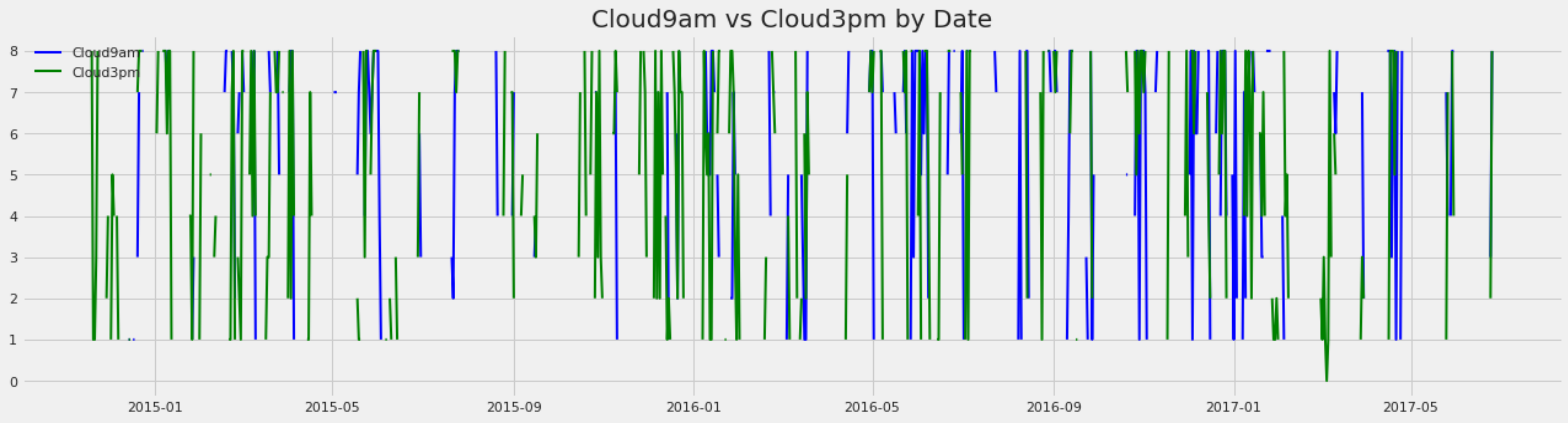
Pressure9am and Pressure3am

- Pressure is high around the months of Jun-Aug and around Dec-Jan you can see that the pressure is low.
- In a low pressure area the rising air cools and this is likely to condense water vapour and form clouds, and consequently rain.



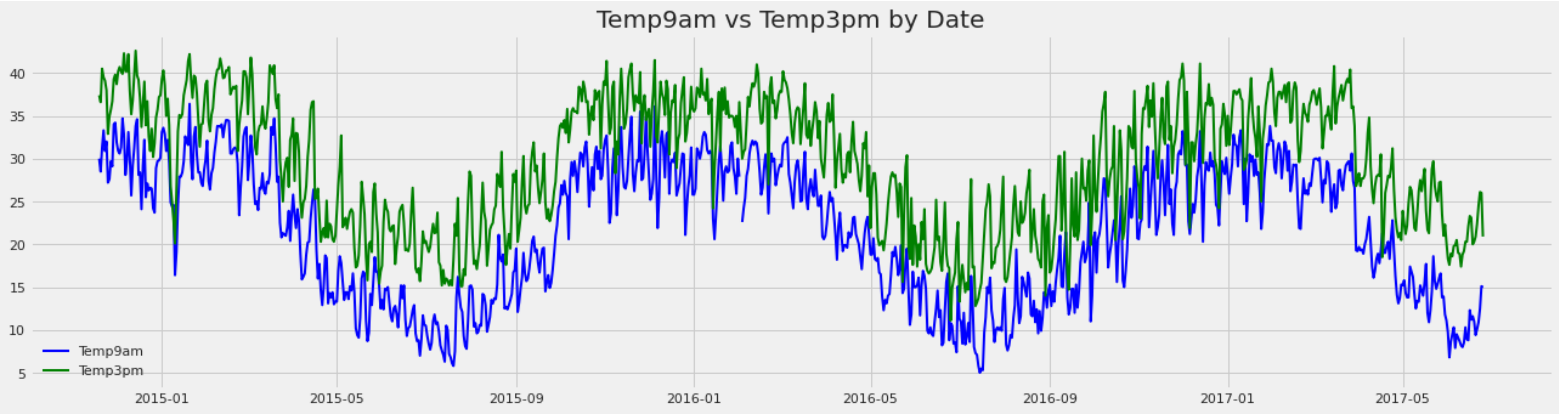
Cloud9am and Cloud3am

Cloud is the same at 5 years but there are certain months when it falls or rises.



Temp9am and Temp3pm

From previous plots we know that Dec-Jan are months when the temperature is high but these are the months when the difference between temperature around 9am and 3pm is less as compared to the months of Jun-Aug when the difference is high.



4.6 Checking Correlationship: HeatMap

From correlation map we can see that:

- MinTemp and MaxTemp features are highly correlated (correlation coefficient = 0.74).
- MinTemp and Temp9am features are highly correlated (correlation coefficient = 0.89).
- MinTemp and Temp3pm features are highly correlated (correlation coefficient = 0.71).

- MaxTemp and Temp9am features are highly correlated (correlation coefficient = 0.89).
- MaxTemp and Temp3pm features are highly correlated (correlation coefficient = 0.98).
- Pressure9am and Pressure3pm features are highly correlated (correlation coefficient = 0.96).
- Temp9am and Temp3pm features are highly correlated (correlation coefficient = 0.86).

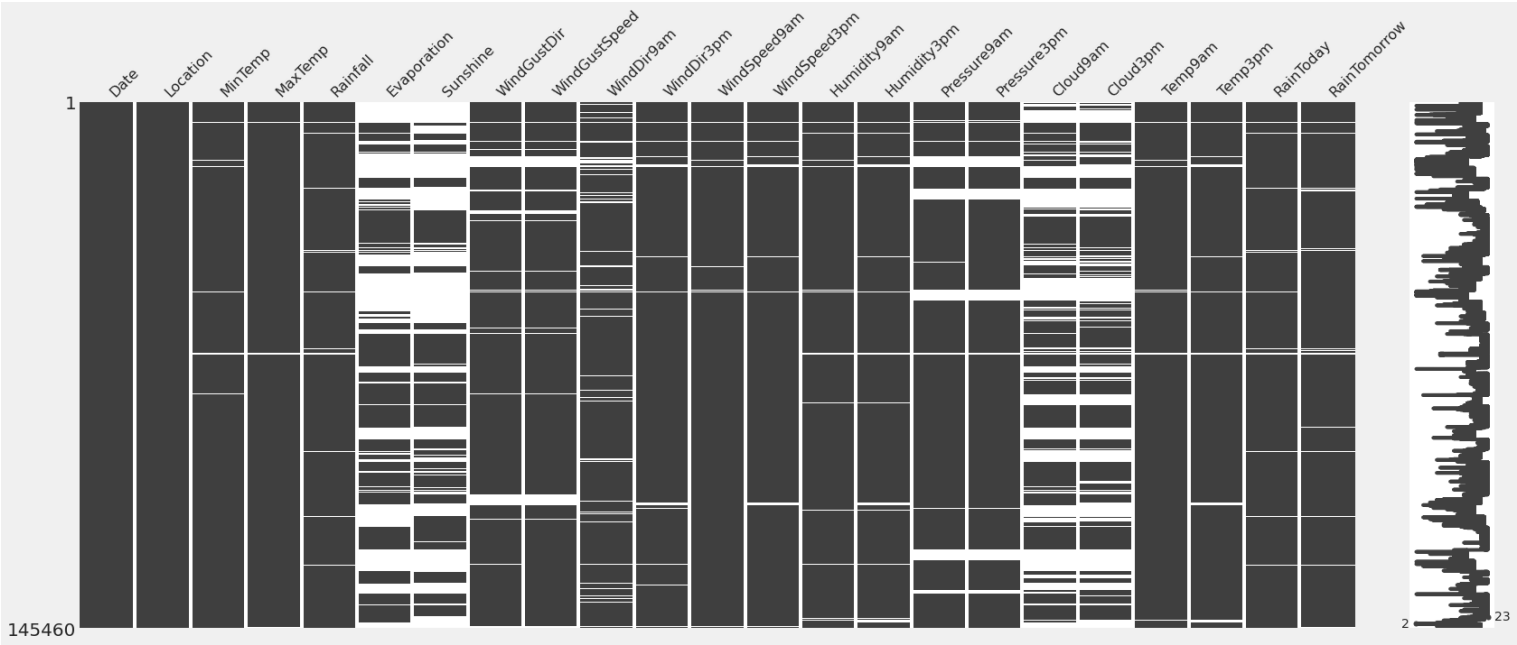
5. Data Visualization and Data preprocessing

5.1 Checking Null values : The following figure show the count of null values in each column as we see in MinTemp column there are 1485 null values

```
Date               0
Location           0
MinTemp         1485
MaxTemp         1261
Rainfall        3261
Evaporation    62790
Sunshine       69835
WindGustDir    10326
WindGustSpeed  10263
WindDir9am     10566
WindDir3pm      4228
WindSpeed9am    1767
WindSpeed3pm    3062
Humidity9am     2654
Humidity3pm     4507
Pressure9am    15065
Pressure3pm    15028
Cloud9am       55888
Cloud3pm       59358
Temp9am         1767
Temp3pm         3609
RainToday       3261
RainTomorrow    3267
dtype: int64
```

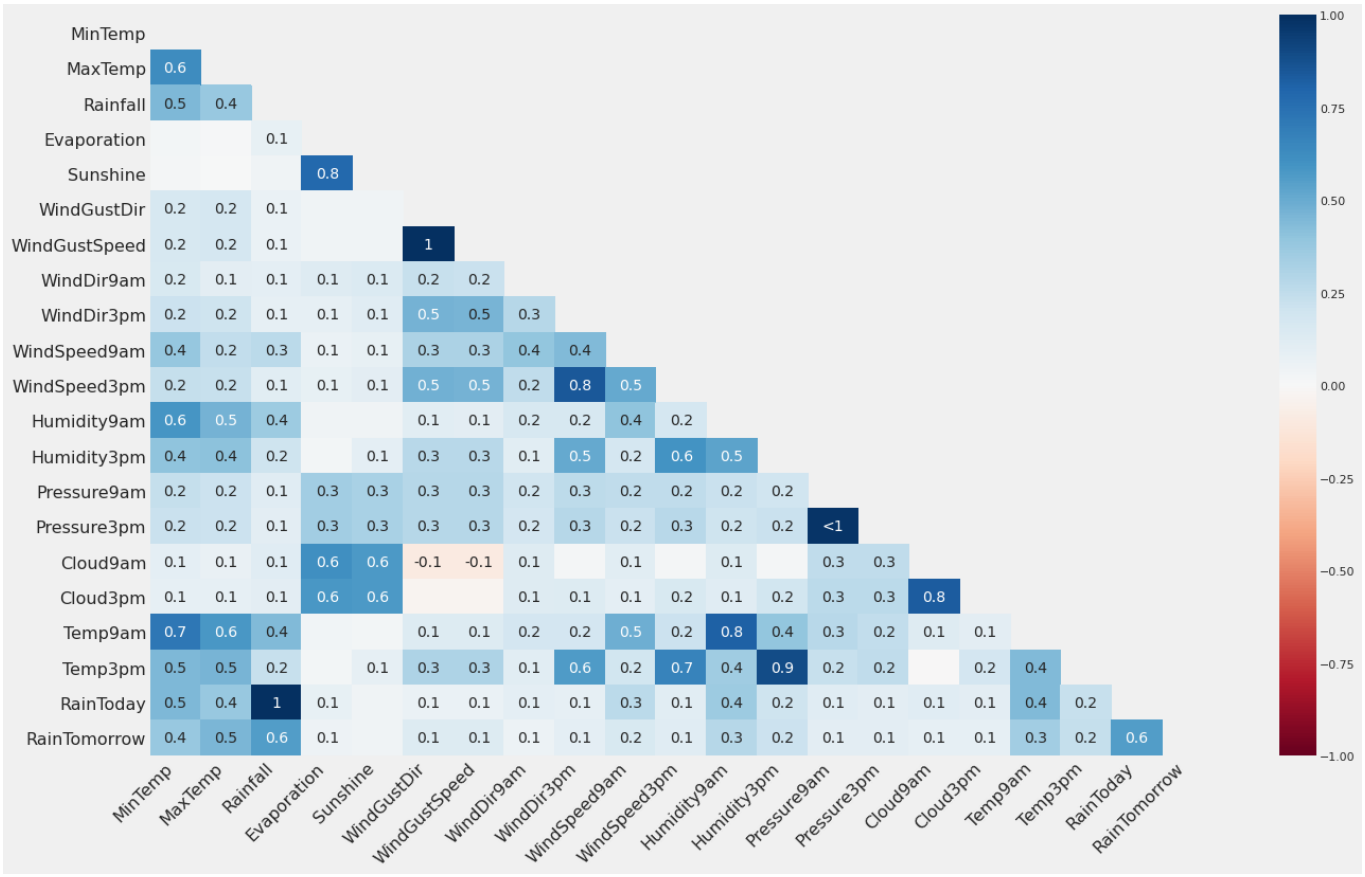
5.2 Visualizing the missing values

It can be visually seen that Evaporation, Sunshine, Cloud9am and Cloud3pm have a lot of missing values.



5.3 Draw heatmap

The graphs show that the number of missing values are high in: Sunshine, Evaporation, Cloud3pm and Cloud9am.



5.4 Dealing with the missing values

1. Check percentage of missing data in every column

```
Date             0.000000
Location         0.000000
MinTemp          1.020899
MaxTemp          0.866905
Rainfall         2.241853
Evaporation     43.166506
Sunshine        48.009762
WindGustDir      7.098859
WindGustSpeed    7.055548
WindDir9am       7.263853
WindDir3pm       2.906641
WindSpeed9am     1.214767
WindSpeed3pm     2.105046
Humidity9am      1.824557
Humidity3pm      3.098446
Pressure9am     10.356799
Pressure3pm     10.331363
Cloud9am        38.421559
Cloud3pm        40.807095
Temp9am          1.214767
Temp3pm          2.481094
RainToday        2.241853
RainTomorrow     2.245978
dtype: float64
```

2. Filling the missing values (null values)

1. Filling the missing values for continuous variables with mean
2. Filling the missing values for categorical variables with mode

3. Check percentage of missing data in every column

```
Date             0.0
Location         0.0
MinTemp          0.0
MaxTemp          0.0
Rainfall         0.0
Evaporation      0.0
Sunshine         0.0
WindGustDir      0.0
WindGustSpeed    0.0
WindDir9am       0.0
WindDir3pm       0.0
WindSpeed9am     0.0
WindSpeed3pm     0.0
Humidity9am      0.0
Humidity3pm      0.0
Pressure9am      0.0
Pressure3pm      0.0
Cloud9am         0.0
Cloud3pm         0.0
Temp9am          0.0
Temp3pm          0.0
RainToday        0.0
RainTomorrow     0.0
dtype: float64
```

5.5 Change Categorical columns to numerical

1. Show datatypes of columns

We have 7 categorical columns

```
Date             object
Location         object
MinTemp         float64
MaxTemp         float64
Rainfall        float64
Evaporation     float64
Sunshine        float64
WindGustDir      object
WindGustSpeed   float64
WindDir9am       object
WindDir3pm       object
WindSpeed9am    float64
WindSpeed3pm    float64
Humidity9am     float64
Humidity3pm     float64
Pressure9am     float64
Pressure3pm     float64
Cloud9am        float64
Cloud3pm        float64
Temp9am         float64
Temp3pm         float64
RainToday        object
RainTomorrow     object
dtype: object
```

2. Change yes and no to 1 and 0 in RainTomorrow and RainToday columns

```
print(df.RainToday)
print(df.RainTomorrow)
```

```
0         0
1         0
2         0
3         0
4         0
         ..
145455    0
145456    0
145457    0
145458    0
145459    0
Name: RainToday, Length: 145460, dtype: int64
0         0
1         0
2         0
3         0
4         0
         ..
145455    0
145456    0
145457    0
145458    0
145459    0
Name: RainTomorrow, Length: 145460, dtype: int64
```

3. **Encoding the categorical variables with Label Encoding**
4. **Show data types of columns**

All columns become numerical

```
Date              int64
Location          int64
MinTemp         float64
MaxTemp         float64
Rainfall        float64
Evaporation     float64
Sunshine        float64
WindGustDir       int64
WindGustSpeed   float64
WindDir9am        int64
WindDir3pm        int64
WindSpeed9am    float64
WindSpeed3pm    float64
Humidity9am     float64
Humidity3pm     float64
Pressure9am     float64
Pressure3pm     float64
Cloud9am        float64
Cloud3pm        float64
Temp9am         float64
Temp3pm         float64
RainToday         int64
RainTomorrow      int64
dtype: object
```

5. **Show first 5 rows**

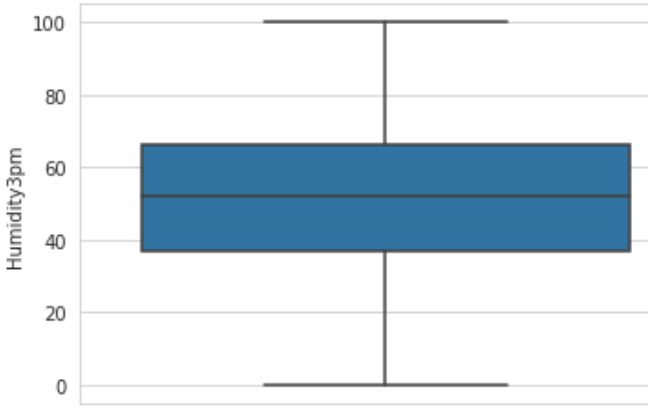
	Date	Location	MinTemp	MaxTemp	Rainfall	Evaporation	Sunshine	WindGustDir	WindGustSpeed	WindDir9am
0	396	2	13.4	13.4	0.6	5.468232	7.611178	13	44.0	13
1	397	2	7.4	7.4	0.0	5.468232	7.611178	14	44.0	6
2	398	2	12.9	12.9	0.0	5.468232	7.611178	15	46.0	13
3	399	2	9.2	9.2	0.0	5.468232	7.611178	4	24.0	9
4	400	2	17.5	17.5	1.0	5.468232	7.611178	13	41.0	1

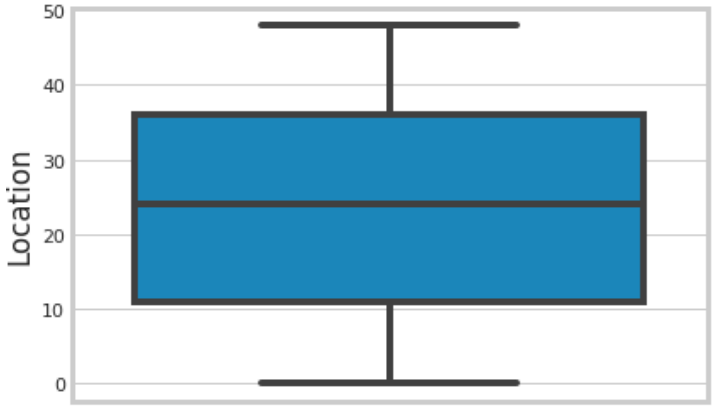
WindSpeed3pm	Humidity9am	Humidity3pm	Pressure9am	Pressure3pm	Cloud9am	Cloud3pm	Temp9am	Temp3pm	RainToday	RainTomorrow
24.0	71.0	22.0	1007.7	1007.1	8.000000	4.50993	16.9	21.8	0	0
22.0	44.0	25.0	1010.6	1007.8	4.447461	4.50993	17.2	24.3	0	0
26.0	38.0	30.0	1007.6	1008.7	4.447461	2.00000	21.0	23.2	0	0
9.0	45.0	16.0	1017.6	1012.8	4.447461	4.50993	18.1	26.5	0	0
20.0	82.0	33.0	1010.8	1006.0	7.000000	8.00000	17.8	29.7	0	0

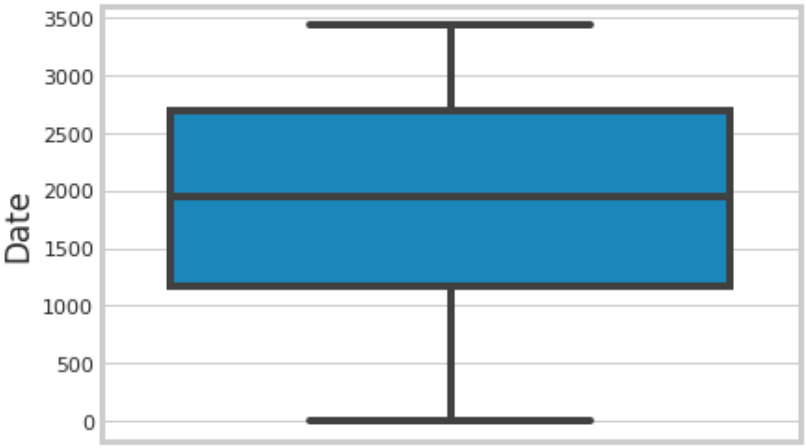
5.6 Check for outliers by boxplots

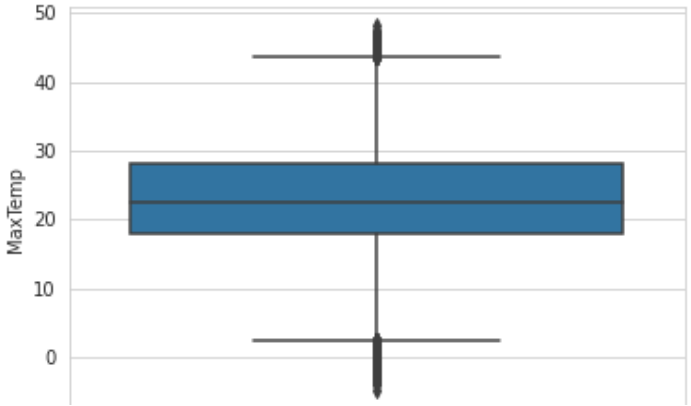
There are outliers in MinTemp, MaxTemp, Rainfall,Evaporation, WindGustSpeed, WindSpeed9am, WindSpeed3pm, Humidity9am, Pressure9am, Pressure3pm, Temp9am, Temp3pm columns.

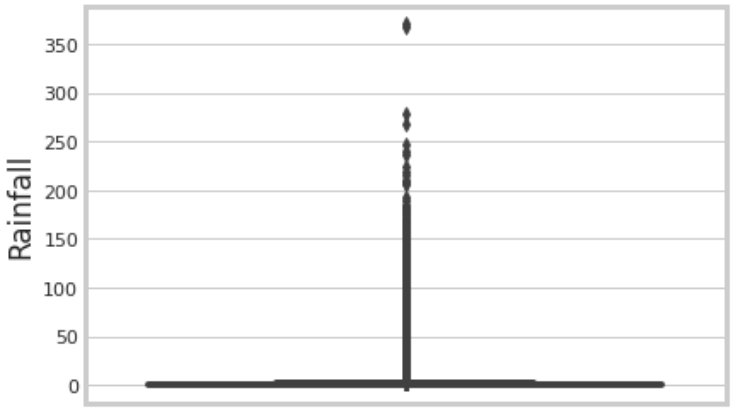
There are no outliers in Date, Location, Humidity9am, Sunshine, Humidity3pm, Cloud9am, Cloud3pm columns.

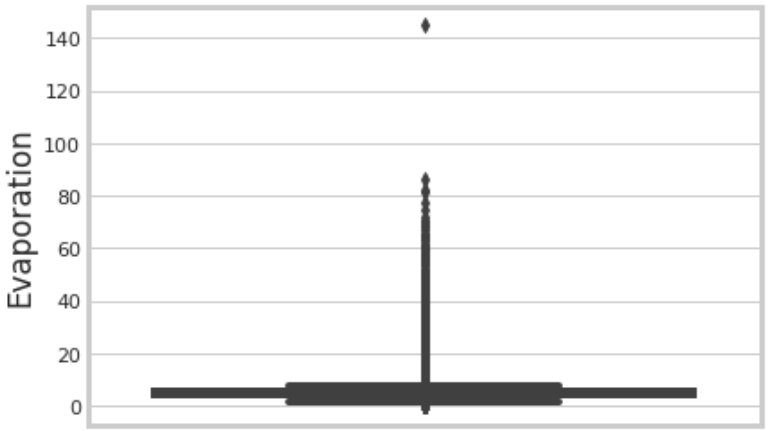


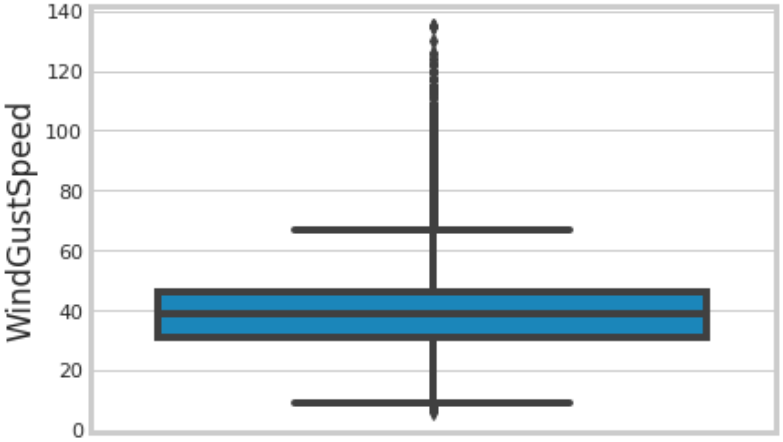


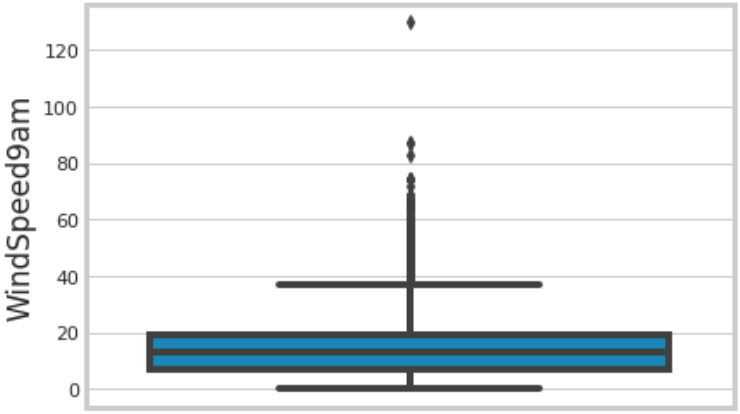


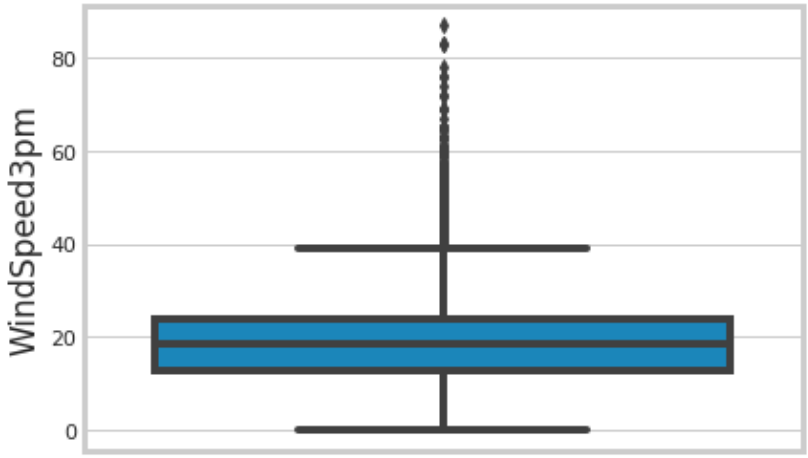


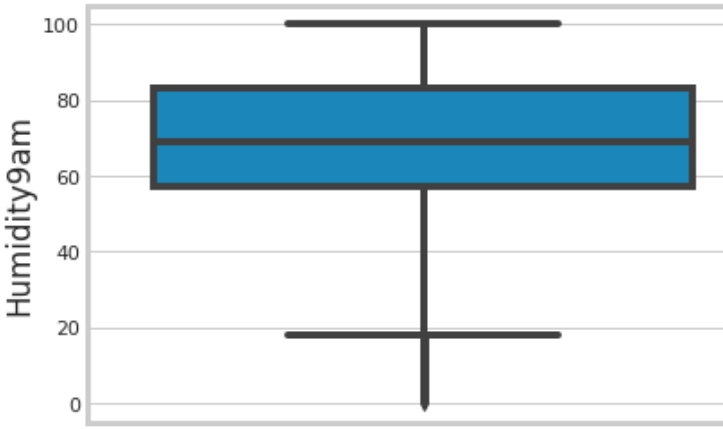

Rainfall
350
300
250
200
150
100
50
0

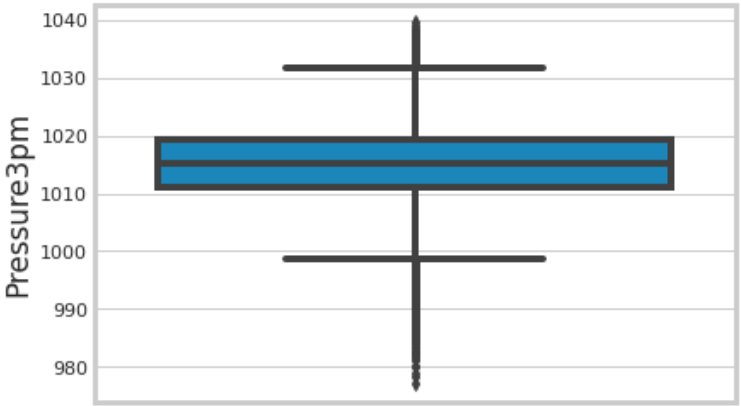

Evaporation
140
120
100
80
60
40
20
0

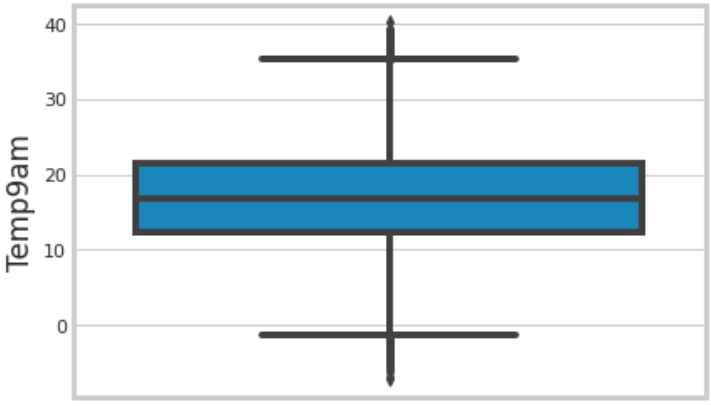

WindGustSpeed
140
120
100
80
60
40
20
0

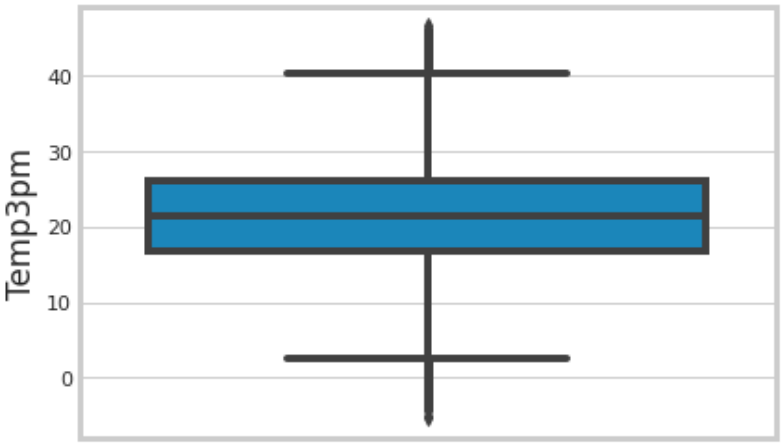

WindSpeed9am
120
100
80
60
40
20
0

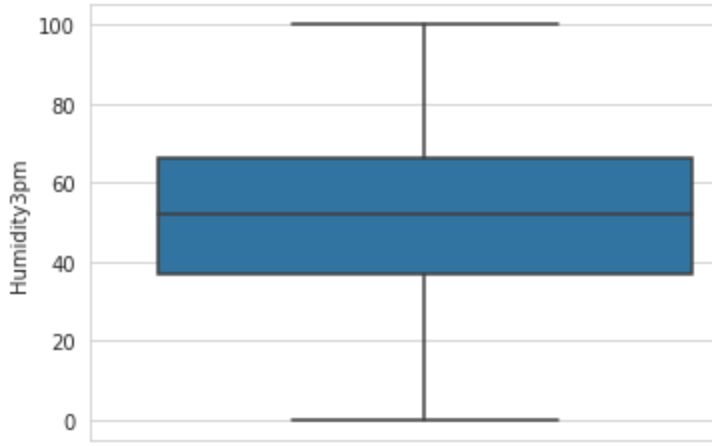

WindSpeed3pm
80
60
40
20
0

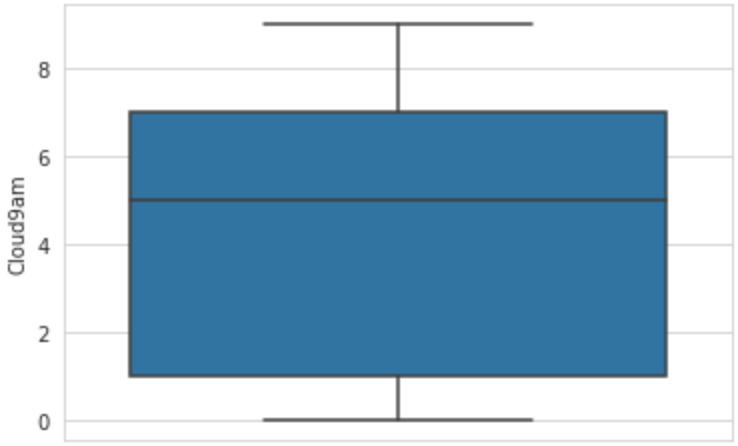

Humidity9am
100
80
60
40
20
0

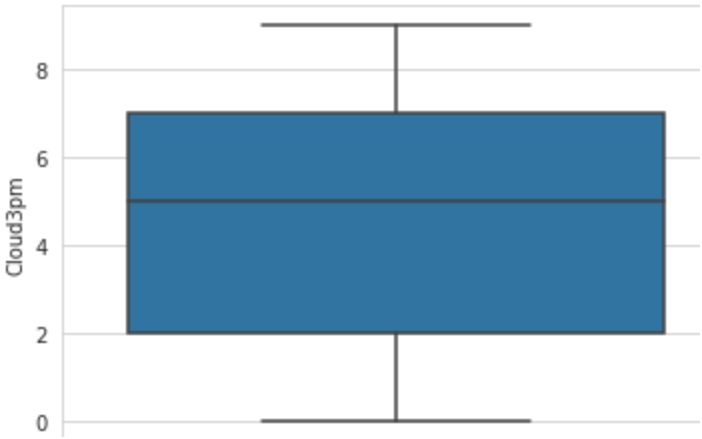

Pressure3pm
1040
1030
1020
1010
1000
990
980

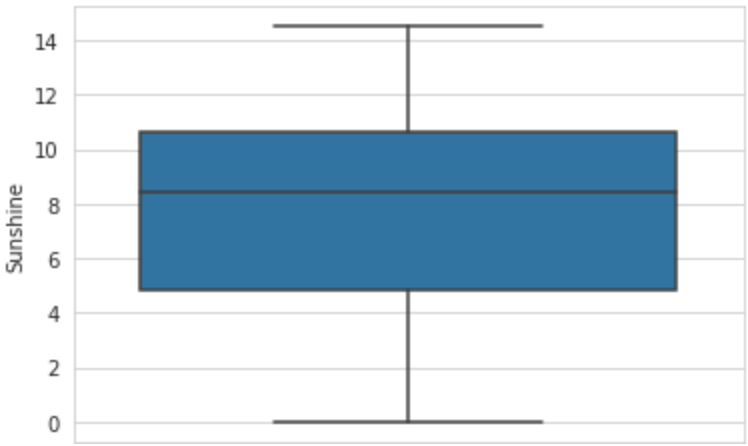

Temp9am
40
30
20
10
0

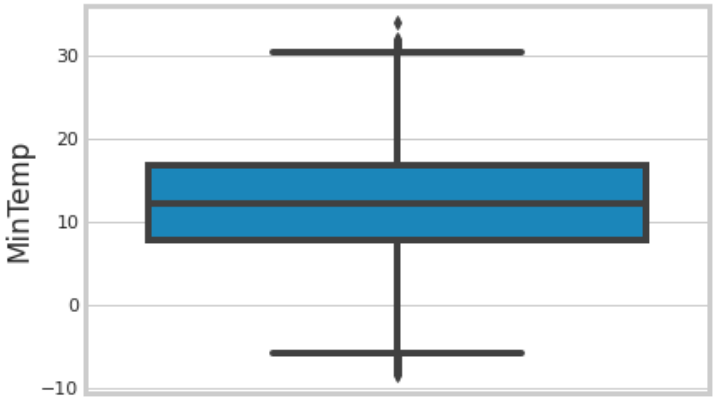

Temp3pm
40
30
20
10
0


Humidity3pm
100
80
60
40
20
0


Cloud9am
8
6
4
2
0

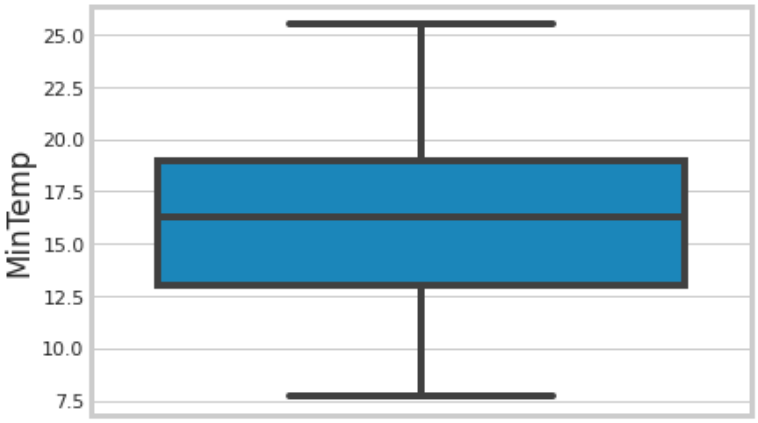

Cloud3pm
8
6
4
2
0

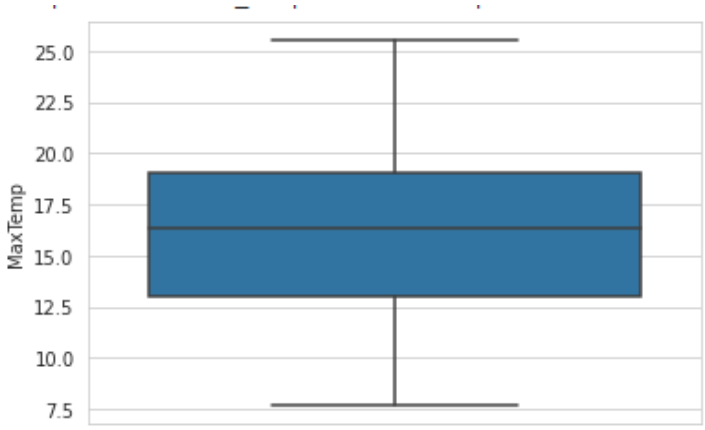

Sunshine
14
12
10
8
6
4
2
0


MinTemp
30
20
10
0
−10

5.7 Remove outliers

Check for MinTemp, MaxTemp column after removing outliers

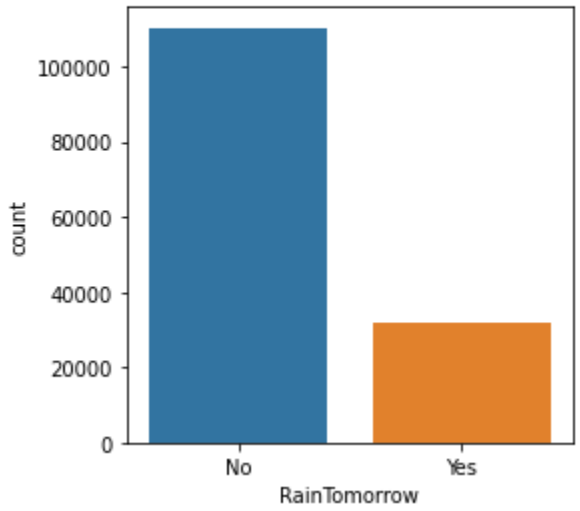




5.8 Balancing the data using SMOTE

- 77.5% data will not rain tomorrow
- Also this figure show that this dataset is imbalanced

```
No     0.775819
Yes    0.224181
Name: RainTomorrow, dtype: float64
<matplotlib.axes._subplots.AxesSubplot at 0x7f3f6019fb90>
```



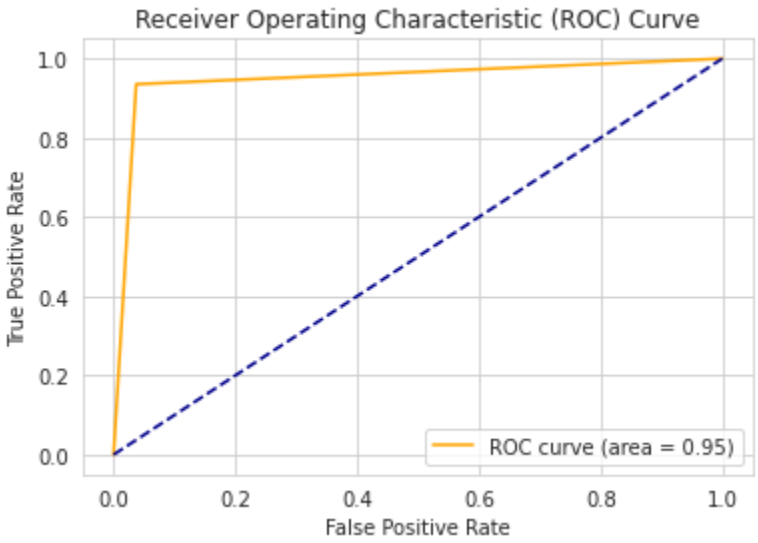
As we show above that the data is imbalanced so we use SMOT to balance it.

6. Model/Classifier training

For modeling we tried to use different models with different parameters:

CatBoostClassifier: was the classifier with best accuracy 94.9% = 95%, tried different parameters with depth (10,15,8) but best was 10.

Cat boost was the model with highest accuracy, that's why we chose it, cat boost is a machine learning algorithm for categorical, numerical and text data and it uses gradient boosting. It's used for forecasting and fraud detection. It works well with small data and doesn't require large amounts of data.



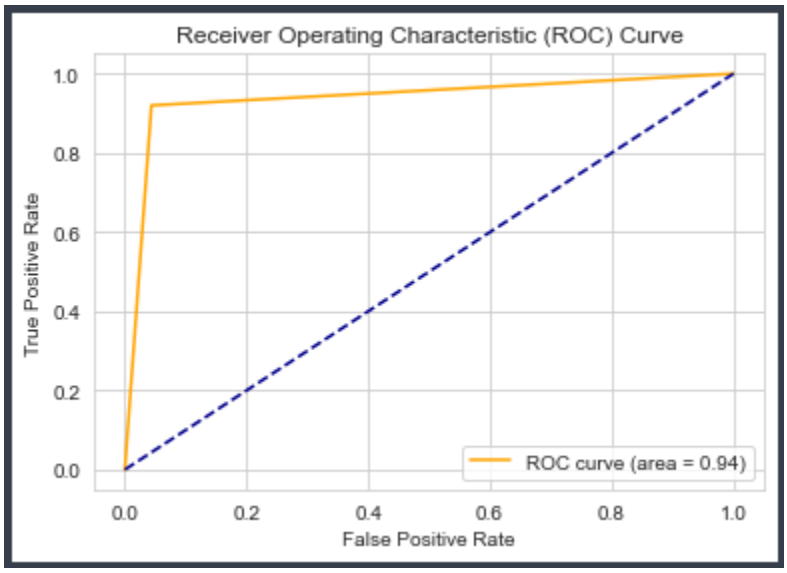
7. Results and Evaluations

- Validation accuracy= val_acc: 0.954368932038835
- Train accuracy = train_acc: 0.9980007497188554
- Test accuracy = Test_acc: 0.9496043315285297

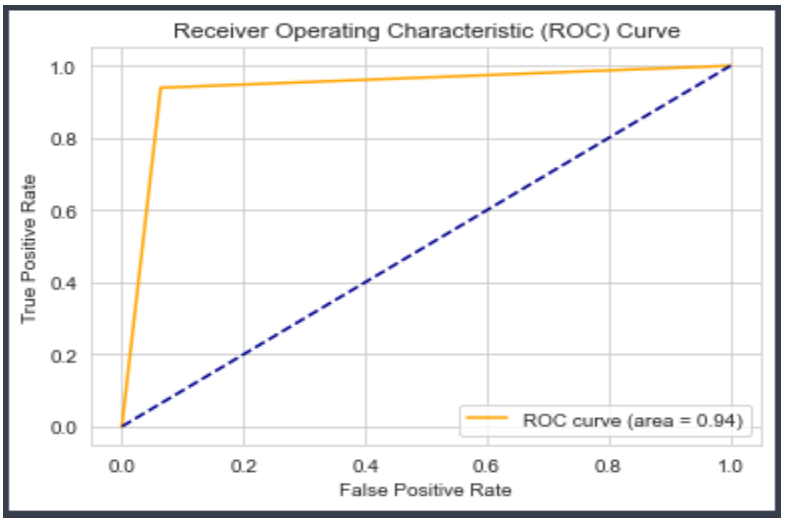
8. Unsuccessful trials that were not included in the final solution

For modeling we tried to use different models with different parameters:

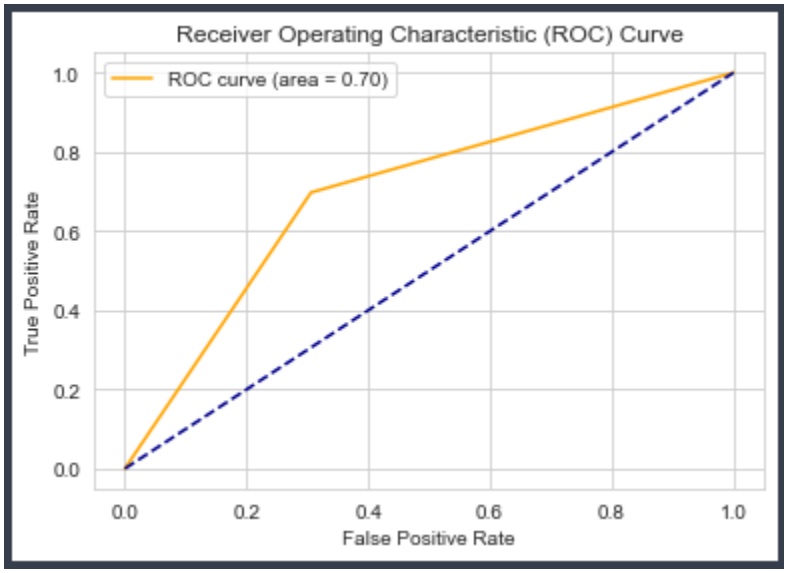
XGBClassifier: tried different parameters with max_depth (8,16,10) and n_estimators (500,300,250) best accuracy with 93.5%.



RandomForestClassifier: tried different parameters with criterion ('entropy', 'gini') and min_sample (100,200) but the best accuracy was 93.7%



LogisticRegression: with different parameters in penalty('l1','l2'), fit_intercept('true','false') best accuracy was 69.5%.



9. Any Enhancements and future work

- Will try other models/ classifiers to get higher accuracy.
- Try to visualize the dataset more so we can find things that need to be fixed like outliers, missing data, unbalanced dataset as we fixed in this project.
- Try to make an android application that uses this model.

10. Business value of this project

This project is very important since it predicts if it will rain tomorrow or not.

Can be used in Weather Forecasting.

1. Helps people prepare for how to dress (i.e. rainy weather)

2. On the other hand, Airlines is another region where weather plays a vital role. They need to recognise the local weather conditions in order to schedule flights.
3. Helps businesses and people plan for power production and how much power to use (i.e. power companies, where to set thermostat)
4. Helps people prepare if they need to take extra gear to prepare for the weather (i.e. umbrella, rain coat)
5. Helps people plan outdoor activities (i.e. to see if rain will impact outdoor event)
6. Helps businesses plan for transportation hazards that can result from the weather (relates to driving and flying for example)
7. Helps farmers and gardeners plan for crop irrigation and protection (irrigation scheduling, freeze protection)
8. Helps people involved with certain activities to know if conditions will be good (i.e. skiing, boating, ballooning)
9. Helps people plan for when to do certain activities that are influenced by weather
10. Helps people know if they need to leave early for work