Next Day Rain Prediction in Australia.

S15025
C. D. V. P. Basnayake

IS 4007

—

Statistics in Practice II

Abstract

Accurate rainfall prediction plays a crucial role in various aspects of human civilization, such as water resource planning, flood prevention, and agriculture. Machine learning techniques have shown promising results in improving rainfall predictions due to the complex and varying nature of weather patterns. In this study, mainly focused on identifying relevant atmospheric features that influence rainfall occurrence using machine learning and neural network models. The dataset was obtained from the Australian Government's bureau of meteorology, considering Australia's diverse climate conditions. Several models, including Random Forest, Logistic Regression, Decision Tree Regressor, XGBoost Classifier and neural networks, were evaluated. The XGBoost Classifier algorithm demonstrated superior performance, achieving a high accuracy score and outperforming other models in predicting rain tomorrow. (F1 score is 0.9025 & Accuracy is 0.9052) . Feature selection revealed that variables such as Humidity3pm, Rainfall, Humidity9am, WindGustSpeed, Pressure3pm, Pressure9am, and Temp3pm had significant influence on rain predictions. Notably, the XGBoost model showed excellent accuracy even without incorporating the location variable, indicating its potential for predicting rain in other locations as well. The findings underscore the effectiveness of machine learning in enhancing rainfall predictions, enabling better planning and decision-making in relation to weather events.

Contents

List of Figures.

List of Tables.

Introduction

Rainfall plays a major role in various weather events and holds a great importance in human civilization. Predicting rainfall accurately is a challenging task due to its complex nature. However, predicting rainfall is crucial in different aspects, such as water resource planning, flood prevention, sewer management and other human activities. Rainfall is also closely linked to agriculture. Therefore, rain prediction has direct impact on agriculture too.

Various researchers have conducted studies to improve the rain predictions using data belongs to different countries. Australia itself also has a variety of weather conditions. Australia's climate varies greatly throughout the eight states and territories; there are four seasons across most of the country and a wet and dry season in the tropical north. According to the results of the past studies, it shows that machine learning techniques have outperformed the traditional deterministic methods while predicting rain due to this varying nature of the weather.

Several environmental factors directly or indirectly influence rainfall occurrence and intensity. These factors include today rain or not, temperature, relative humidity, sunshine, pressure, and evaporation. Various studies have investigated the correlations between these features and rainfall prediction. For example, temperature, wind, and cyclone were identified as significant features for rainfall prediction in the Indian region.

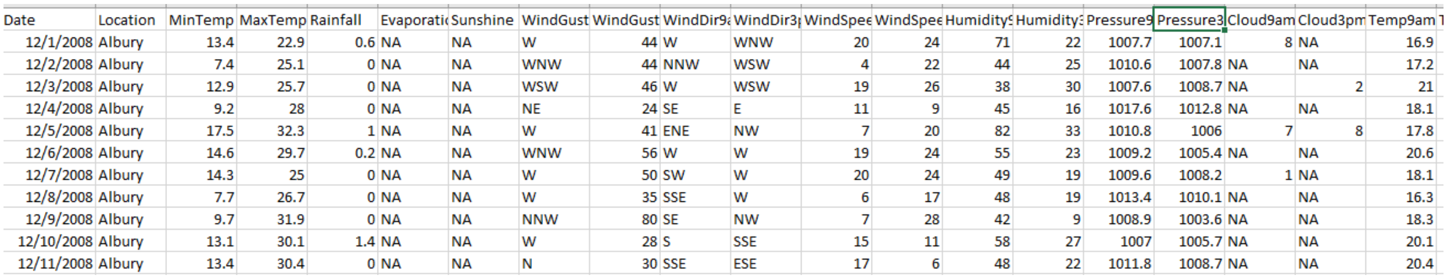
Therefore, this study focuses on identifying relevant atmospheric features that influence rainfall occurrence of the next day, using machine learning techniques and neural networks. The raw data is collected from Australian Government’s bureau of meteorology. The study experimented with machine learning algorithms, including Random Forest, Binary logistic Regression, and Decision Tree Regressor and some neural network models. The Random Forest algorithm outperformed the others, demonstrating better prediction of rain tomorrow in Australia using the selected relevant environmental features, as indicated by the accuracy score and confusion matrix.

The main objective of this study is to predict the next day going to be rain or not.

In the next section (Literature Review) the theories used by past researchers will be discussed. Under theory and methodology part the theories used in this analysis, and how they were used wii be thoroughly discussed. In the data section the dataset, variables and the data preprocessing techniques used are going to be explained. All the analysis will be done under the explanatory and advance analysis topics. Final conclusions of the study will be discussed under the section of general discussion and conclusion.

significance of the study,

- Accurate prediction of rain can significantly contribute to risk reduction measures such as flood control.
- Provide farmers with vital information for making informed decisions related to irrigation, planting, harvesting, and pest management.
- Studying next day rain prediction can contribute to a better understanding of broader climate patterns, such as seasonal variations, long-term trends and climate change impacts.



Date	Location	MinTemp	MaxTemp	Rainfall	Evaporati	Sunshine	WindGust	WindGust	WindDir9a	WindDir3p	WindSpee	WindSpee	Humidity9	Humidity3	Pressure9	Pressure3	Cloud9am	Cloud3pm	Temp9am
12/1/2008	Albury	13.4	22.9	0.6	NA	NA	W	44	W	WNW	20	24	71	22	1007.7	1007.1	8	NA	16.9
12/2/2008	Albury	7.4	25.1	0	NA	NA	WNW	44	NNW	WSW	4	22	44	25	1010.6	1007.8	NA	NA	17.2
12/3/2008	Albury	12.9	25.7	0	NA	NA	WSW	46	W	WSW	19	26	38	30	1007.6	1008.7	NA	2	21
12/4/2008	Albury	9.2	28	0	NA	NA	NE	24	SE	E	11	9	45	16	1017.6	1012.8	NA	NA	18.1
12/5/2008	Albury	17.5	32.3	1	NA	NA	W	41	ENE	NW	7	20	82	33	1010.8	1006	7	8	17.8
12/6/2008	Albury	14.6	29.7	0.2	NA	NA	WNW	56	W	W	19	24	55	23	1009.2	1005.4	NA	NA	20.6
12/7/2008	Albury	14.3	25	0	NA	NA	W	50	SW	W	20	24	49	19	1009.6	1008.2	1	NA	18.1
12/8/2008	Albury	7.7	26.7	0	NA	NA	W	35	SSE	W	6	17	48	19	1013.4	1010.1	NA	NA	16.3
12/9/2008	Albury	9.7	31.9	0	NA	NA	NNW	80	SE	NW	7	28	42	9	1008.9	1003.6	NA	NA	18.3
12/10/2008	Albury	13.1	30.1	1.4	NA	NA	W	28	S	SSE	15	11	58	27	1007	1005.7	NA	NA	20.1
12/11/2008	Albury	13.4	30.4	0	NA	NA	N	30	SSE	ESE	17	6	48	22	1011.8	1008.7	NA	NA	20.4

Figure 1: Data set

Literature Review

In this chapter, it is intended to discuss about previous developed model and findings that has been done by different scholars. A variety of approaches has been employed for weather forecasting. Most of the methods were based on linear models.

(Imon, Manos, & Bhattacharjee, 2012) have made the weather forecasting using Logistic regression approach cause Logistic regression is useful for situations in which we want to be able to predict the presence or absence of an outcome (e.g, rainfall) based on values of a set of predictor variables. They have used a climatic data set from Bihar, India, that has been extensively analyzed by many other researchers. Before fitting the model by a logistic regression, researchers used some recently developed data screening methods like brushing and clustering to identify spurious observations.

(Hemalatha, Rao, & Kumar, 2021) proposed a Neural Networks based model for weather prediction. The traditional methods of weather prediction sometimes deviate in predicting the weather conditions due to non linear relationship between the input features and output condition was the motivation to use Neural Network approach. The superiority of the proposed model is tested with the weather data collected from Indian metrological Department (IMD). The performance of model is tested with various metrics. It was employed Fully Connected Neural Network (FCNN) model and weather data is supervised and labeled before it is divided in to training and testing. The training data is fed as input to FCNN model through which the learning process is completed. The test data is fed as an input to the FCNN model to validate the performance. During the validation, the weather condition of the test pattern is predicted.

In the research paper, *Smart Weather Prediction Using Machine Learning* (Jayasingh, Mantri, & Pradhan, 2022) used several machine learning models to identify the best fitted model and the approach for weather prediction. It was incorporated Random Forest, Decision Tree, Support Vector Machine, KNN, Adaboost, Xgboost, Gradient Boosting, Naïve Bayes and Logistic Regression methods and checked the accuracy of model to select the best model.

Following table shows the accuracy of each model.

Model	Accuracy
Random Forest	79.52
Decision Tree	71.23
Support Vector Machine	59.33
KNN	77.86
Adaboost	71.43
Xgboost	79.94
Gradient Boosting	81.67
Naïve Bayes	73.09
Logistic Regression	78.14

Table 1:Accuracy of each model; Literature Review

According to the table number, highest accuracy level is recorded in the Gradient Boosting model and logistic regression model’s accuracy level is recorded as 78.14. In conclusion, predicting weather is a challenging task for researchers, many techniques are evolved in time for the prediction of weather since last many years. Here, it was discussed some of the methodologies done by scholars to forecast weather.

Theory and Methodology

The purpose of the study is to predict whether the next day is a rainy day or no when some weather-related data for today is given. Since there exist a target column in the data set, this belongs supervised learning. The target column or the dependent variable is a binary variable. Because of that for the building of model, (which is a forecasting model), set of algorithms such as Binary Logistic Model, machine learning techniques and neural networks were used

Models Applied

Logistic Regression Model

Since the target variable which is next day weather status; Yes or No is a categorical variable, it will be performed Binary Logistic Regression for the prediction. Following is the visual summery of how logistic regression model is structured.

Here what happen is, probabilistically model binary variables. This is an ideal model when an interpretable model is needed. How when comes to performance, due to its simpler nature, logistic regression model lags behind the other above-mentioned models. The assumptions of logistic regression model are,

Dependent variable is binary, No multicollinearity, Linearity between logit and independent variables, Large sample size

Decision Tree

Decision trees are versatile machine learning algorithms used for both regression and classification tasks. They consist of a series of nodes, starting with a root node and leading to decision outcomes at the leaves. The objective is to create a predictive model by learning simple decision rules from the data's features. Decision trees can be seen as approximations using piecewise constant functions.

When using decision trees, the process begins at the root node, comparing attribute values with the record's attributes and traversing the tree accordingly. Decision tree algorithms employ various techniques to determine how to split nodes into sub-nodes. The goal is to increase the homogeneity (or purity) of the resultant sub-nodes in relation to the target variable. The algorithm explores all available variables for splitting nodes and selects the split that leads to the most homogeneous sub-nodes.

The selection of the specific algorithm for decision tree construction depends on the type of target variables being predicted. One challenge in using the decision tree algorithm is determining which variable should be chosen for splitting the tree. To address this, two measures, namely entropy and the Gini index, are utilized. Entropy quantifies the level of randomness or diversity within a set of class values. Higher entropy indicates a more diverse set with less informative aspects. The decision tree aims to find divisions that reduce entropy and increase homogeneity within groups. Conversely, the Gini index measures the probability of misclassifying a randomly selected variable. It ranges from 0 (when all elements belong to a single class) to 1 (when elements are randomly distributed across different classes), with 0.5 indicating equal distribution among classes. When constructing a decision tree, the variable with the smallest Gini index is preferred as the root element.

Random Forest

Random Forest is a machine learning method that combines multiple decision trees to improve prediction accuracy. The key idea is that a group of uncorrelated models (individual decision trees) performs better together than individually. In classification tasks, each tree provides a classification, and the forest selects the class with the majority of votes. In regression tasks, the forest calculates the average of the outputs from all trees.

The strength of Random Forest lies in the low correlation between individual models, reducing errors and providing more reliable predictions. It corrects for overfitting and maintains accuracy even with missing data. Random Forest is efficient for large databases and offers versatility for classification or regression tasks. It is relatively easy to use compared to more complex algorithms like neural networks.

However, Random Forest may require more memory and can be slower due to the use of multiple decision trees. While it addresses overfitting, there can still be some impact on the overall forest. Despite these limitations, the benefits of Random Forest, including ease of use, efficiency and accuracy are the reasons for choosing this.

XGBoost

XGBoost is a highly efficient and flexible machine learning library designed for optimized distributed gradient boosting. It handles missing data well, works efficiently on large datasets, and offers various hyperparameters for tuning. It performs exceptionally in scenarios with large training data and often outperforms deep learning models. XGBoost is widely used for its speed, accuracy, and ability to handle complex relationships in the data.

Neural Networks

Neural networks are AI techniques that mimic the human brain's structure and enable computers to process data. They capture complex relationships and can make intelligent decisions without extensive human intervention. Inputs are assigned weights and combined using activation functions. The network architecture includes layers, such as input, output, and hidden layers, and can be single-layer or multi-layer with different types of connections. Learning algorithms adjust weights based on available information, using supervised or unsupervised learning. Trained neural networks retain learned weights for future use.

Performance Measuring

Accuracy Score

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. In binary classification, accuracy has the following definition:

$$Accuracy = \frac{True\ positives + True\ negatives}{True\ positives + True\ negatives + False\ positives + False\ Negatives}$$

F1 Score

$$Precision = \frac{No. of\ true\ positives}{No. of\ true\ positives + No. of\ False\ positives} \quad Recall = \frac{No. of\ true\ positives}{No. of\ true\ positives + No. of\ False\ negatives}$$

F1 score combines precision and recall into a single metric. The F1 score is specifically designed to handle imbalanced data effectively.

$$F1\ score = 2 * \frac{precision * recall}{precision + recall}$$

Confusion Matrix

It is a table that is used in classification problems to assess where errors in the model were made.
The rows represent the actual classes the outcomes should have been. While the columns represent the predictions we have made. Using this table, it is easy to see which predictions are wrong.

True Positive	False Positive
False Negative	True Negative

Data Preprocessing Techniques

SMOTE

Imbalanced classification means, developing predictive models on datasets where there is a significant class imbalance. One common approach to tackling imbalanced datasets is by oversampling the minority class. The simplest way to do this is by duplicating instances from the minority class, although this doesn't provide any new information to the model. An alternative method is to generate new synthetic examples based on existing ones, which is known as the Synthetic Minority Oversampling Technique (SMOTE).

Methodology

First The dataset is imported into Python. Then data preprocessing was done to ensure the data is in a suitable format for analysis. Once the data is prepared, exploratory data analysis (EDA) is performed, utilizing various statistical graphics and visualization techniques such as box plots, histograms, bar graphs, stacked bar graphs, and heatmaps. These visualizations provide valuable insights and help uncover patterns, trends, and correlations within the dataset.

After completing the EDA, the dataset was split into two separate parts, a training set and a test set. The chosen split ratio is 80% for the training set and 20% for the test set. The training set was utilized to train the machine learning models, enabling them to learn and capture patterns and relationships in the data.

Once the models are trained, they are evaluated to assess their performance on unseen data. The test dataset, which was kept separate from the training dataset, was used for this evaluation. Then the models made predictions on the test set, and these predicted values were then compared with the corresponding values of the dependent variable in the test set. By measuring the accuracy of these predictions, we could gauge how well the models can generalize to new, unseen data.

In the logistic regression model, one important step was to handle correlated variables. Correlated variables can violate the assumptions of the logistic regression model, so they were identified and removed before fitting the model. This was done to enhance the reliability of the model.

After fitting the initial logistic regression model, a technique called backward elimination was employed. This process involved iteratively removing insignificant variables from the model based on specified significance level. By eliminating these variables, the model was refined to include only the most significant predictors. The model was then refitted using this reduced set of significant variables.

In the case of the "RandomForestModel," a different approach was used. After fitting the model, a feature selection technique was applied to identify the best set of features that would construct a more optimized model. Feature selection helps to identify the most relevant features for predicting the target variable. This process ensures that unnecessary or redundant features are not included, which can enhance the efficiency and performance of the model. So, with random forest model, two scenarios were tested. First, the accuracy was assessed using all the variables in the model. Then, the accuracy was tested again using only the best set of features selected through feature selection.

Data

Variables Used

The “weatherAUS” dataset contains 145460 observations with 23 variables. From this data set 140787 observations and 22 variables used for the analysis. Not any column has more than 50% missing values. The variables which were used and their descriptions are given below.

Variable	Description	Variable type
Location	Common name of the location of the weather station	Categorical
MinTemp	Minimum temperature in degrees Celsius	Continuous
MaxTemp	Maximum temperature in degrees Celsius	Continuous
Rainfall	The amount of rainfall recorded for the day in mm	Continuous
WindGustDir	The direction of the strongest wind gust in the 24 hours to midnight	Categorical
WindGustSpeed	The speed (km/h) of the strongest wind gust in the 24 hours to midnight	Continuous
WindDir9am	Direction of the wind at 9am	Categorical
WindDir3pm	Direction of the wind at 3pm	Categorical
WindSpeed9am	Wind speed (km/h) averaged over 10 minutes prior to 9am	Continuous
WindSpeed3pm	Wind speed (km/h) averaged over 10 minutes prior to 9pm	Continuous
Humidity9am	Humidity (percent) at 9am	Continuous
Humidity3pm	Humidity (percent) at 3pm	Continuous
Preasure9am	Atmospheric pressure (hpa) reduced to mean sea level at 9am	Continuous
Preasure3pm	Atmospheric pressure (hpa) reduced to mean sea level at 3pm	Continuous
Temp9am	Temperature (degrees C) at 9am	Continuous
Temp3pm	Temperature (degrees C) at 3pm	Continuous
RainToday	1 if precipitation (mm) in the 24 hours to 9am exceeds 1mm, otherwise 0	Categorical
RainTomorrow	The amount of next day rain in mm. Used to create response variable RainTomorrow. A kind of measure of the "risk".	Categorical

Table 2:Data Descriptions

Data Preparation

Before carry out the study a set of operations carried out on the data

1. Missing Values: A major problem with the real-world data sets are the missing values. Here the *Figure 2: Missing values in each variable* shows the number of missing values in each variable. If the observations which has missing values are eliminated, then a large number of observations are lost, because of that a thorough analysis was done on missing values and below mentioned information are identified.
    - Some variables such as Pressure3pm, Pressure9pm and WindGustSpeed has missing values on some cities like Salmon Gums, Penrith etc. For these variables the missing values are imputed using average of those variables considering all cities.
    - Some observations don't have a value for RainToday and RainTomorrow variables. Since RainTomorrow is the target variable and imputing RainToday with mode is not valid, those observations are removed.
    - More than 35% of the recorded values of Evaporation, Sunshine, Cloud9am, Cloud3pm are missing values. Because of that those variables removed from the analysis.
    - Remaining missing values handling:
        - Missing values which were belonged to categorical variables imputed using mode of each variable.
        - Missing values which were belonged to continuous variables imputed using median of each variable.

2. Removing Variables: The Date column is removed from the data set since it is irrelevant in predicting whether next day is a rainy day or not.

3. Convert Variables: The RainToday and RainTomorrow has string values "Yes" or "No". These values converted into 1 & 0.

4. Data Scaling: When inspecting the data, it was identified that some of the numerical variables and some variables such as Pressure, Humidity has lager range of values and larger mean values. Because of that the variables having larger values can have a greater impact on the machine learning algorithms. To overcome this issue all the numerical values were normalized using min – max scaler. It scales data to the range of 0 to 1.

5. Outliers: In the data set most of the variables have outliers. ZX. To remove the outliers the Inter Quantile Range (IQR) is used.

6. Dummy Variables: For all the categorical variables in the data set, dummy variables were created using the get_dummies() function in python.

7. Imbalanced Classification: Since the target variable "RainTomorrow" was unbalanced with almost 78% of the values were "0" s and only 22% of the values were "1" s. To get rid of this, SMOTE function used and minority class was oversampled.

After all these preprocessing steps final dataset which was used for model building is consist with 140,723 observations and 107 columns.

Exploratory Data Analysis.

The following bar graph, *Figure 2* shows the number of missing values in each variable.

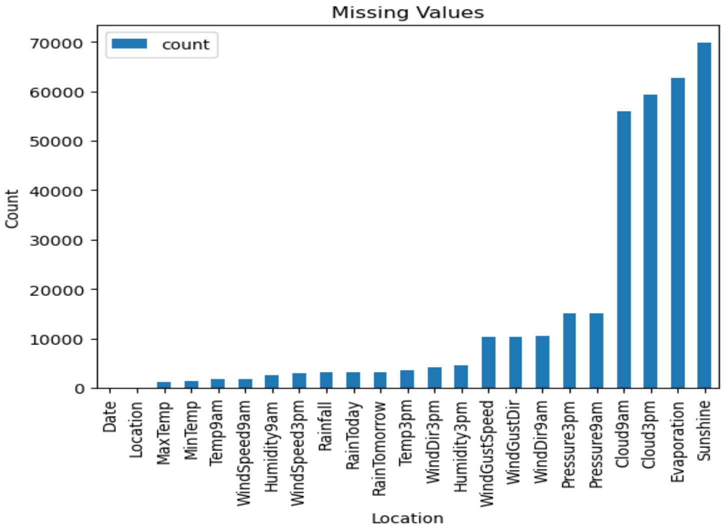


Figure 2: Missing values in each variable

According to above graph it can be identified that Cloud9am, Cloud3pm, Evaporation and Sunshine variables have large number of missing values. The missing value percentages of these variables are:

Variable	Missing value percentage
Cloud9am	38.4%
Cloud3pm	40.8%
Evaporation	48.0%
Sunshine	43.2%

Table 3: Highest missing value percentages

After removing the variables with more than 30% missing values, a set of box-plots were drawn to get the insights about the continuous variables.

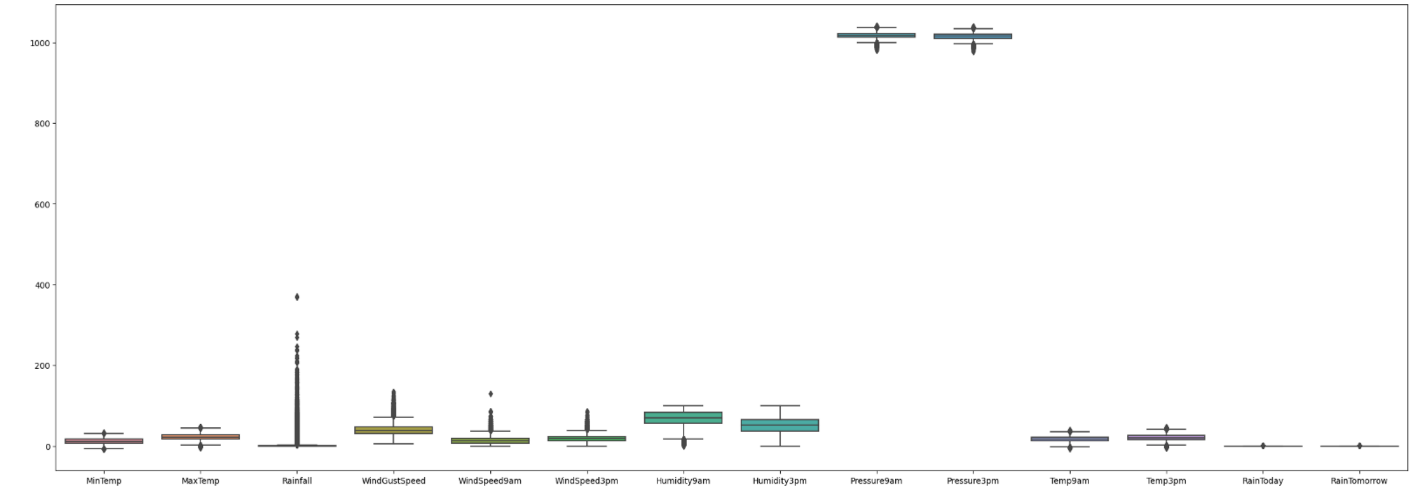


Figure 3: Box plots of used continuous variables before preprocessing

According to the above *Figure 3* it can be identified that most of the continuous variables has outliers. Another main point that could be captured using this graph was most two variables have higher values and higher mean values compared to other variables. Those two variables are Pressure9am and Pressure3pm.

The below graph shows distributions of continuous variables after removing outliers and scaling them using Min-Max scaler.

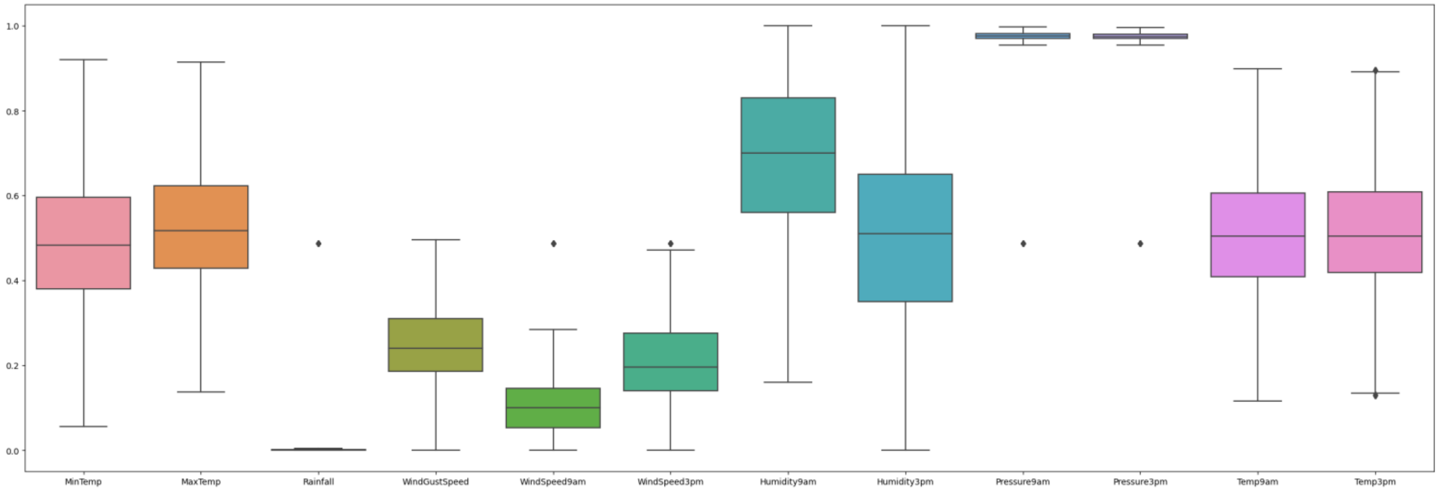


Figure 4: Box plots of the continuous variables in final dataset

The below *Figure 5* shows how the target variable "RainTomorrow" was imbalanced before doing the data preprocessing and the *Figure 6* shows how the "RainTomorrow" after using SMOTE.

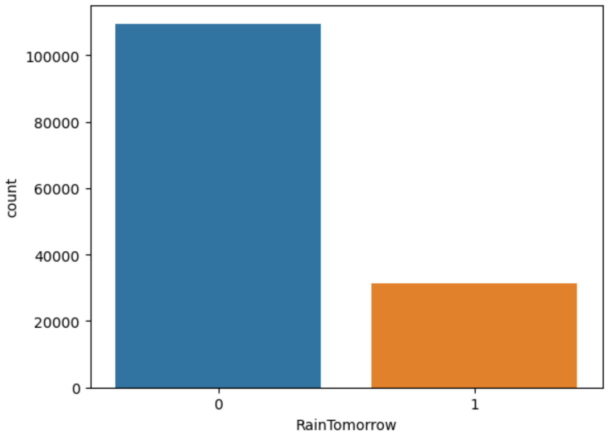


Figure 5: Imbalance target variable

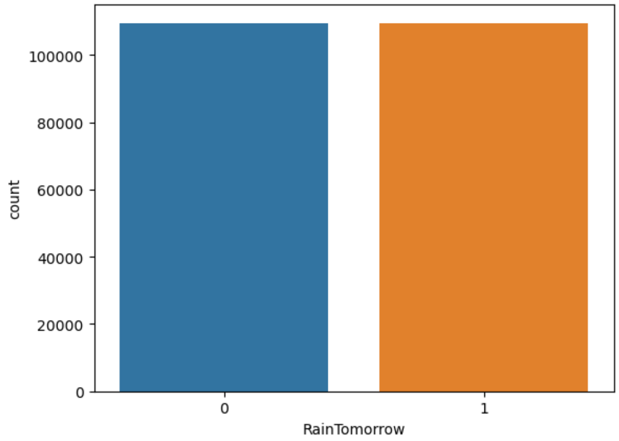


Figure 6:Target variable; After applying SMOTE

It can be identified that after using SMOTE, The RainTomorrow is balanced with same number of 1 s and 0 s.

The below 2 heatmaps shows the details about the multicollinearity among the variables.

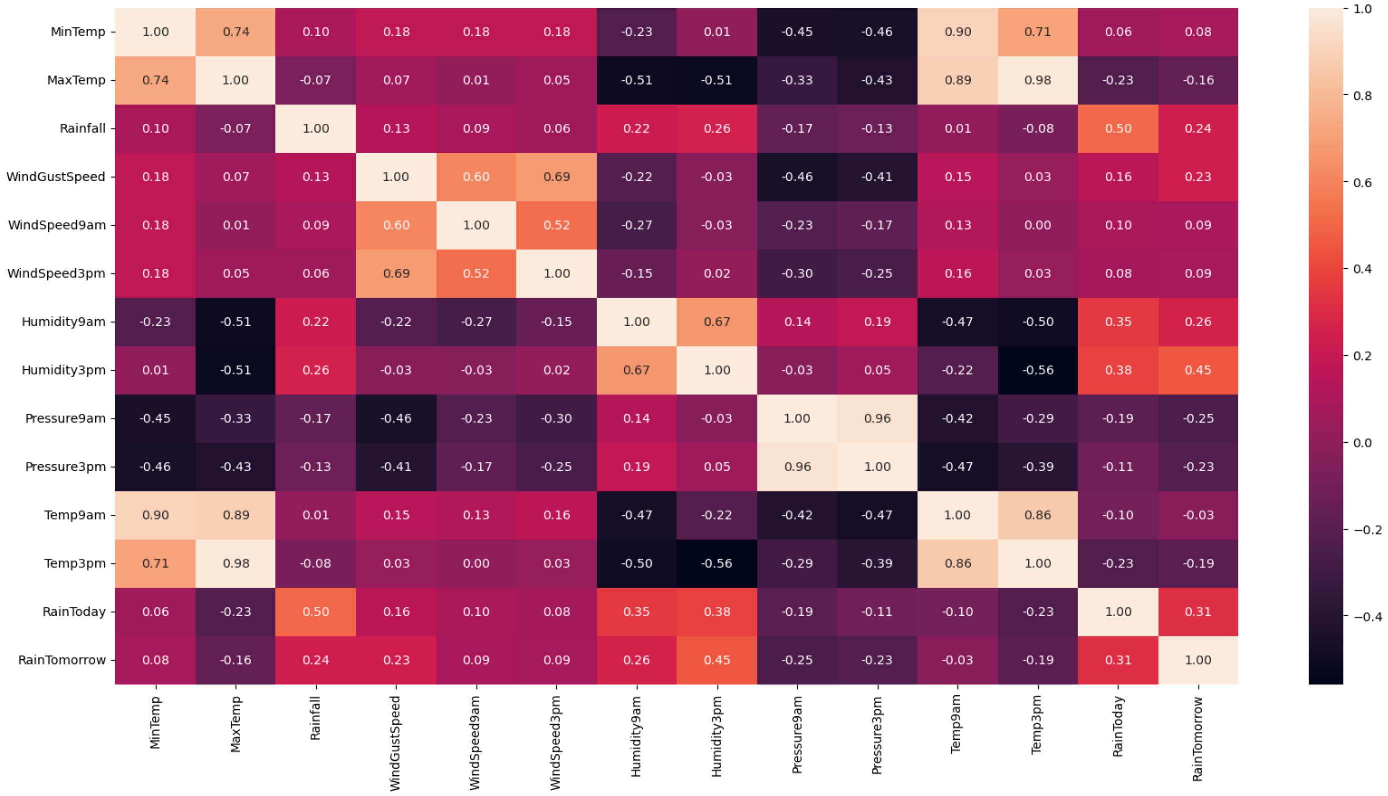


Figure 7: Heatmap for multicollinearity

By inspecting above *Figure 7* it can be concluded that some independent variables have strong correlations among that.

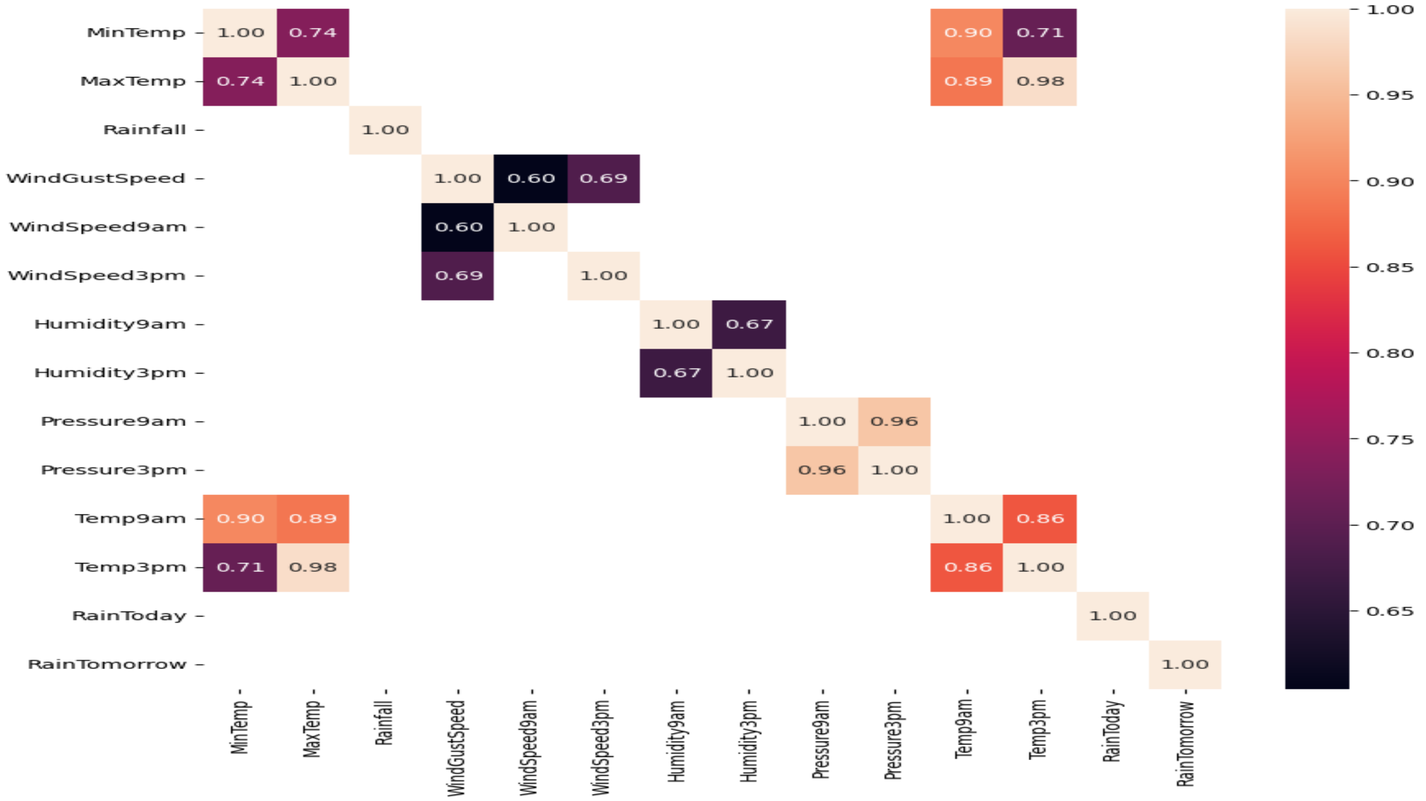


Figure 8: Heatmap for highly correlated variables

The above *Figure 8* shows the info about highly correlated variables. By inspecting, it can be identified that those highly correlated variables are actually related ones. That is because the highly correlated variables are,

- MinTemp, MaxTemp, Temp3pm and Temp9am
- WindSpeed3pm and WindSpeed9pm and WindGustSpeed
- Humidity9am and Humidity3pm

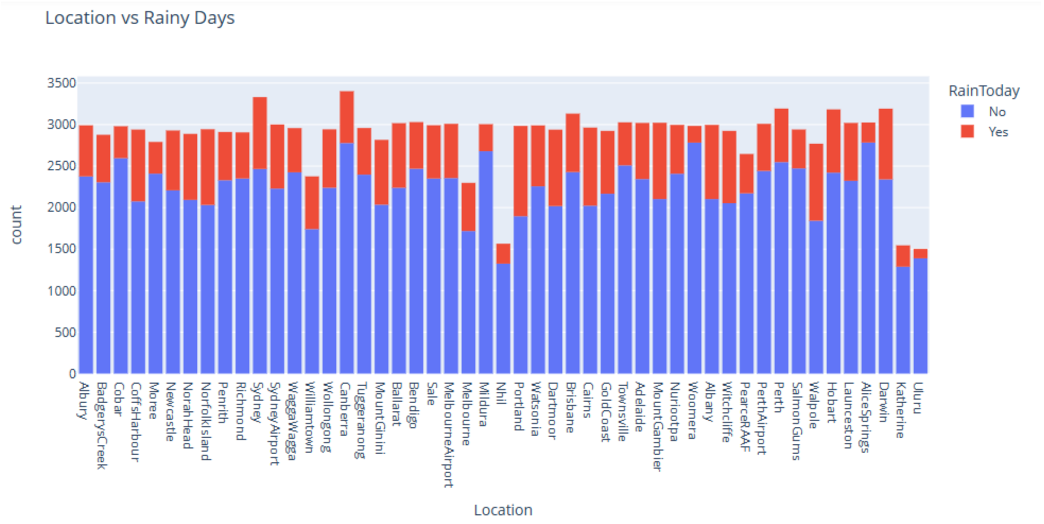


Figure 9:RainToday counts in each city

According to above graph, *Figure 9* it can be concluded that the data is collected uniformly. Most of the cities have data for more than 2000 days. And it seems that RainToday:Yes to RainToday:No ratio is also quite close among most of the cities.

The below map, *Figure 10* shows the locations of the cities in Australia which the data collected and the total rainfall in each city within the observation period.

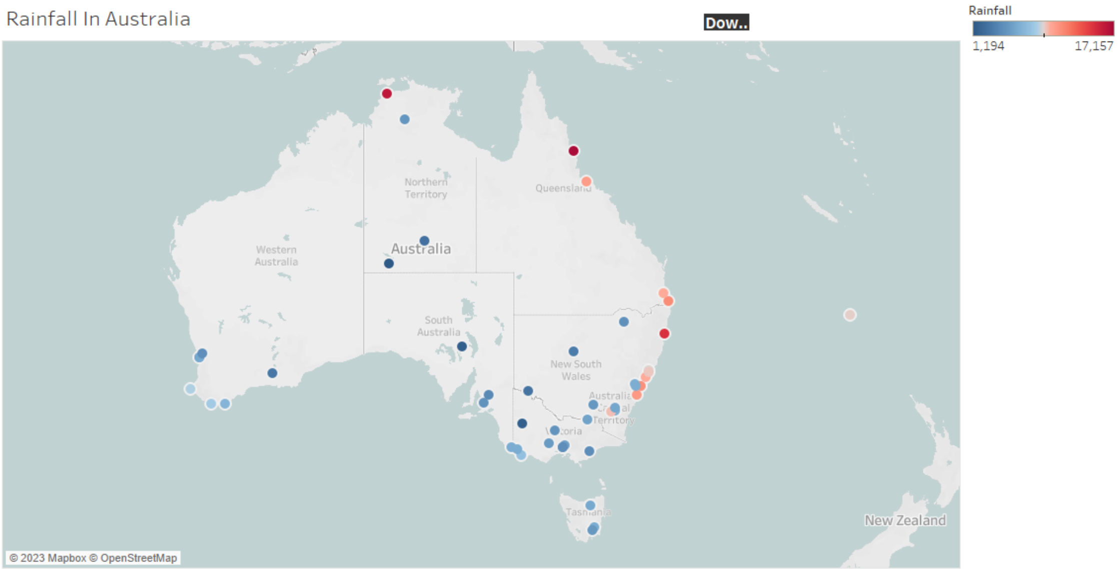


Figure 10: Location and Total Raifall of each city

According to above map, it can be identified that from the cities which were used for data gathering, cities in the North, North-East and East costal areas had higher rainfall during the observation period.

Below graph shows how data collected throughout the year.

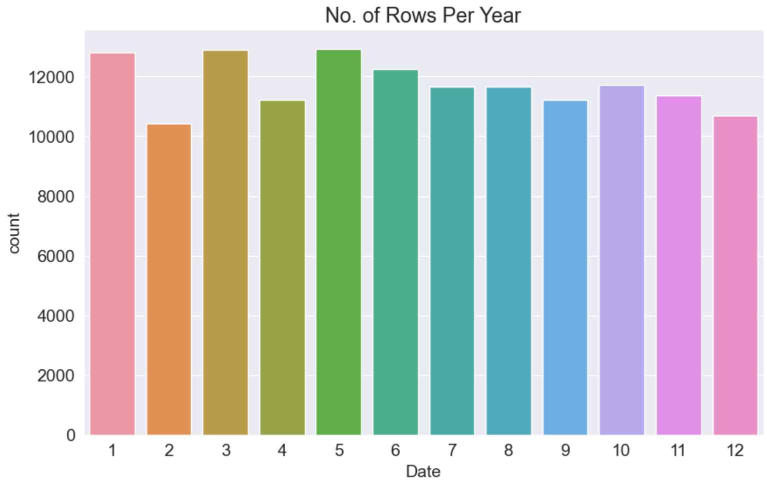


Figure 11:No. of observations for each month

Normally the rainfall and raining conditions is not same for all the months. So, if the data collected from only a set of months or more data collected from specific set of moths the analysis could have gone wrong. But from the above plot, *Figure 11* it can be identified that the number of observations from each month is not much different when comparing to the average number of observations from each month.

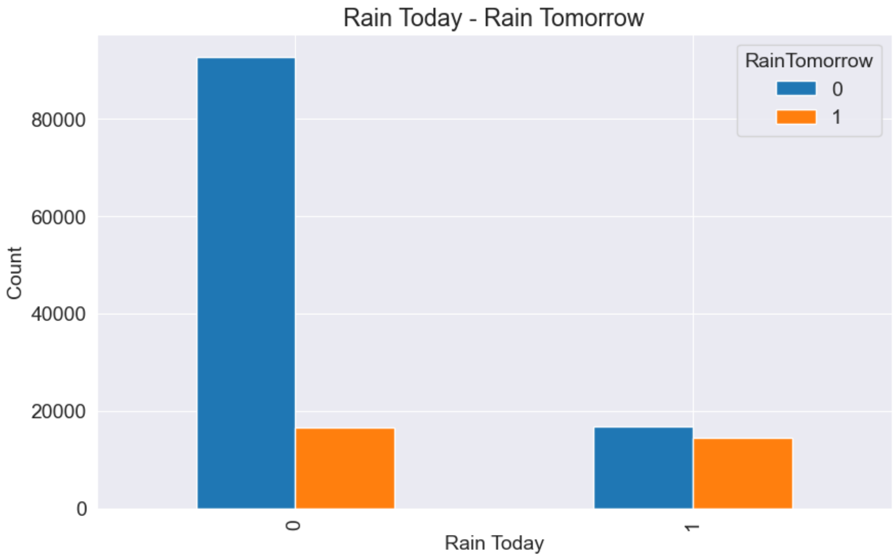
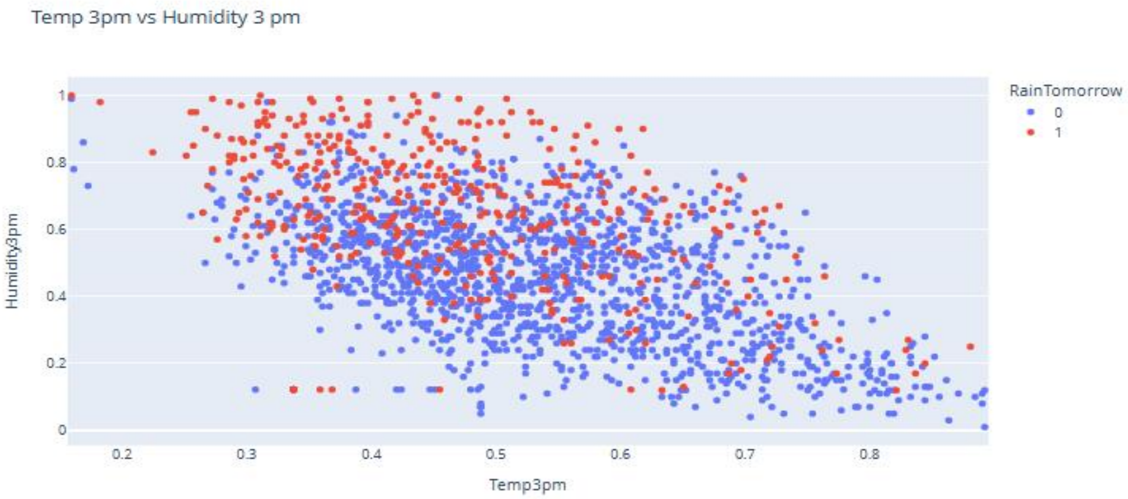


Figure 12:RainToday - RainTomorrow counts

The above clustered bar chart, *Figure 12* is drawn before applying SMOTE. There it can be clearly seen that when today is not a rainy day the probability of tomorrow not being a rainy day is clearly higher than the probability of tomorrow become not rainy when today is a rainy day. According to that, despite having class imbalance it can be seen that there is higher chance of raining tomorrow when today is a rainy day.



The above scatterplot shows that RainTomorrow has a relationship with Humidity3pm and Temp3pm. It can be identified as, when temperature at 3pm is low and Humidity at 3pm is higher, there is a higher chance of raining in the next day.

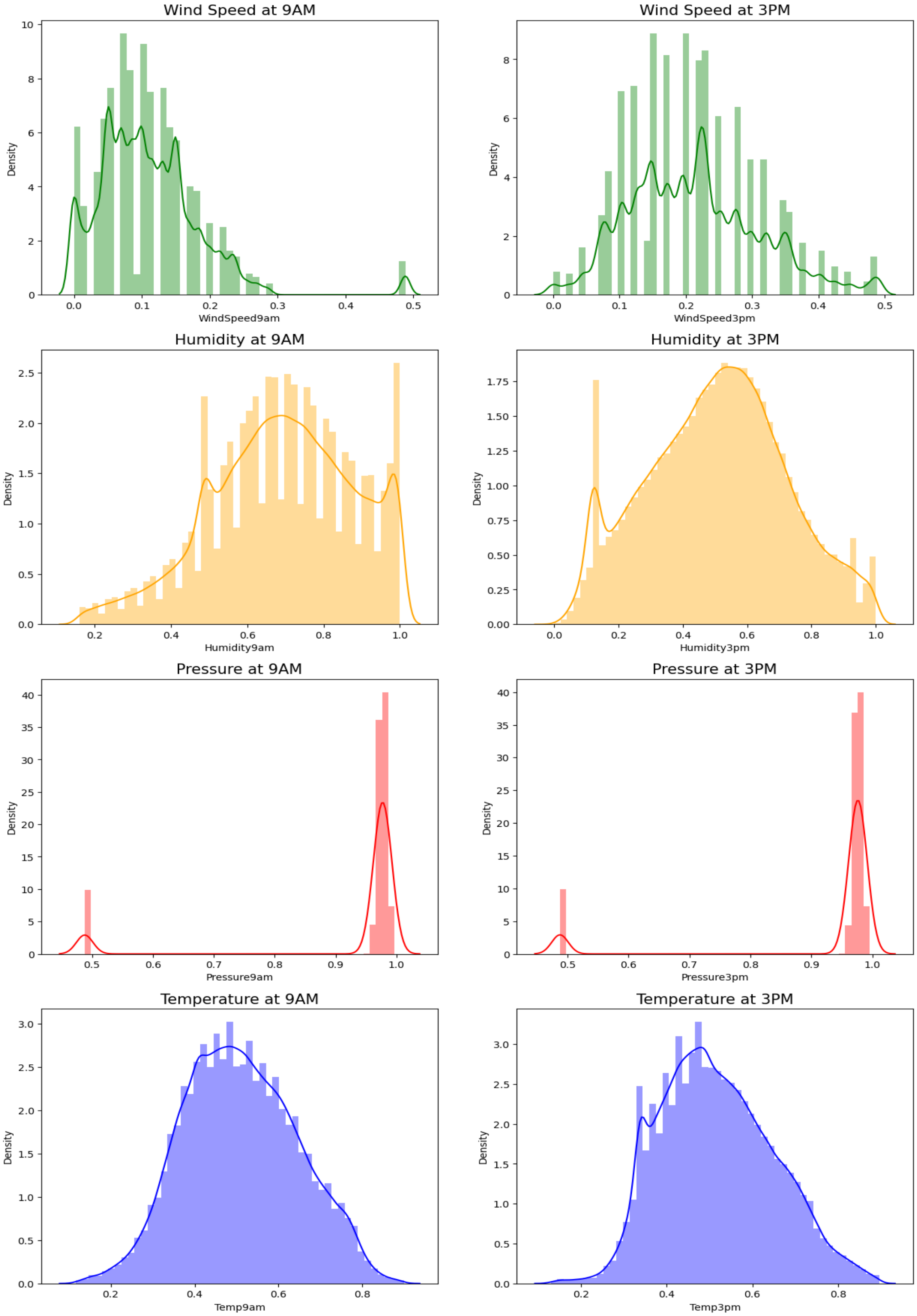


Figure 14: Histograms of Wind speed, Humidity, Pressure and Temperature at 9am and 3pm

The above set of histograms in *Figure 14* shows that WindSpeed9am is left skewed but WindSpeed3pm is somewhat normally distributed. Both Temp9am and Temp3pm are normally distributed. Humidity9am is left skewed while Humidity3pm is normally distributed. Pressure9am and Pressure3pm are not normal but both of them shows the same distribution.

The below histogram shows how RainTomorrow changes with Humidity3pm.

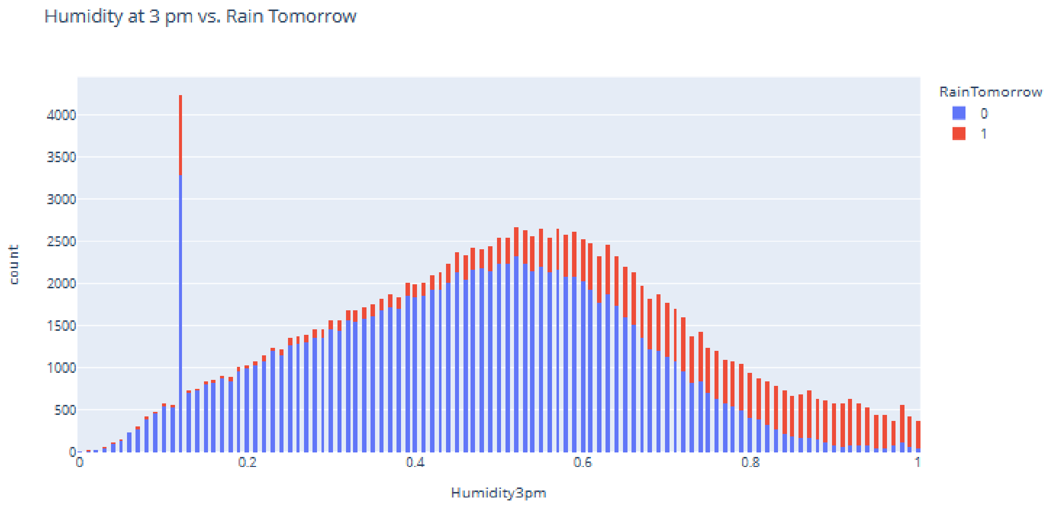


Figure 15: Histogram of Humidity3pm

The above histogram *Figure 15*, also shows that when humidity is high there is a higher chance of raining tomorrow.

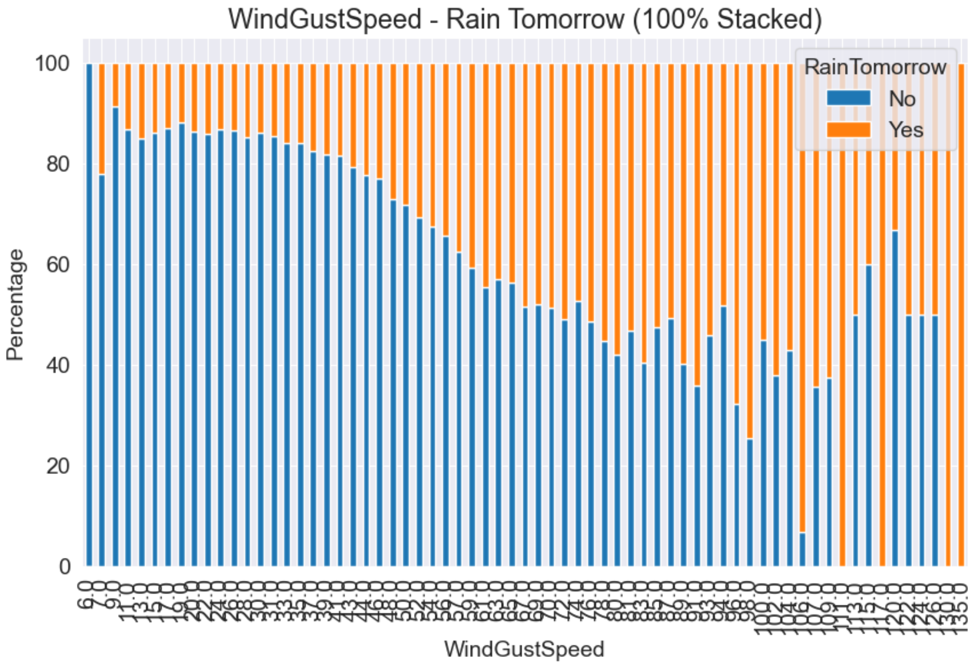


Figure 16: 100% stacked bar graph of Wind speed and Rain tomorrow

The above stacked bar graph indicated that when wind speed is higher the probability if raining tomorrow is getting higher. So it seems that there is a connection between raining tomorrow and today wind speed. When wind speed is above 7.2km/h there is a more than 50% of chance of raining tomorrow.

Advanced Analysis.

Logistic Regression

This model is implemented using python, LogisticRegression function defined in scikitlearn library. First the model fitted and then predicted the values and compared them,

This below table shows the values of the evaluated metrics,

Solver used = lillinear

Accuracy	Mean absolute error	Precision	F1 score
0.8045	0.1955	0.8059	0.8027

Table 4:Evaluation Values for Logistic Regression

Logistic Regression with Feature Selection

This model is implemented using python, LogisticRegression function defined in scikitlearn library. After model fitting backward elimination was used to remove insignificant variables from the model. After that model was refitted with the values selected from the random forest and then predicted the values and compared them,

This below table shows the values of the evaluated metrics,

Random_state = 1
Solver used = ‘sag’

Accuracy	Mean absolute error	Precision	F1 score
0.8	0.1959	0.8056	0.8024

Table 5:Evaluation values for Logistic Regression with feature selection

Logit Regression

This model is implemented using python, LogisticRegression function defined in statmodels library. First the model fitted and then predicted the values and compared them,

This below table shows the values of the evaluated metrics,

Random State = 40

Accuracy	**Mean absolute error**	**Precision**	**F1 score**
0.8041	0.1959	0.8056	0.8024

Table 6: Evaluation values for Logit Regression

Decision Tree Classifier

This model is implemented using python, DecisionTreeClassifier function defined in scikitlearn library. First the model fitted and then predicted the values and compared them,

This below table shows the values of the evaluated metrics,

Random State = 1

Accuracy	**Mean absolute error**	**Precision**	**F1 score**
0.8328	0.1675	0.8250	0.8333

Table 7:Evaluation values for Decision Tree Classifier

Neural Network with Single Layer

This model is implemented using python, Perceptron function defined in scikitlearn library. Perceptron is a machine-based algorithm used for supervised learning of various binary sorting tasks. First the model fitted and then predicted the values and compared them,

This below table shows the values of the evaluated metrics,

Random State = 40

Accuracy	**Mean absolute error**	**Precision**	**F1 score**
0.7263	0.2737	0.6763	0.7583

Table 8:Evaluation values for Perceptron

Neural Network with Multiple Layers

This model is implemented using python, MLPClassifier function defined in scikitlearn library. It is a technique of feed-forward artificial neural networks using a back propagation learning method to classify the target variable used for supervised learning. MLPClassifier relies on an underlying Neural Network to perform the task of classification. First the model fitted and then predicted the values and compared them,

This below table shows the values of the evaluated metrics,

Hidden Layer Size = 30,30,30
Max iterations = 1000

Accuracy	Mean absolute error	Precision	F1 score
0.8548	0.1452	0.87	0.8509

Table 9:Evaluation values for MLPClassifier

Random Forest Model

This model is implemented using python, RandomForestClassifier function defined in scikitlearn library. First the model fitted and then predicted the values and compared them,

This below table shows the values of the evaluated metrics,

N_estimator = 1000
Max_depth = 40
Random_state = 1

Accuracy	Mean absolute error	Precision	F1 score
0.8921	0.1060	0.8830	0.8928

Table 10: Evaluation values for Random Forest

Random Forest Model with Feature Selection

This model is implemented using python, RandomForestClassifier function defined in scikitlearn library. After fitting the model then using freature_importances_ function most important features were selected. Then the model refitted using only the most important 12 features. Then predicted the values and compared them,

This below table shows the values of the evaluated metrics,

N_estimator = 1000
Max_depth = 40
Random_state = 1

Accuracy	Mean absolute error	Precision	F1 score
0.8820	0.1180	0.8841	0.8810

Table 11:Evaluation Values for Random Forest with feature selection

XGBoost Model

This model is implemented using python, XGBClassifier function defined in xgboost library. Then the model fitted. Then predicted the values and compared them.
When the model fitted using only 5 features, Humidity3pm, Rainfall, Humidity9am, WindGustSpeed, Pressure3pm, Pressure9am, Temp3pm still get an accuracy of 0.9052 and f1 score of 0.9025.

This below table shows the values of the evaluated metrics,

Accuracy	Mean absolute error	Precision	F1 score
0.9052	0.0948	0.9328	0.9025

Conclusion.

- The analysis was done without removing outliers and after removing outliers. The models' accuracy increased after removing the outliers. So, its concluded that in this model its better to remove outliers.

- The F1 score and accuracy of the developed models were compared to find the best model.

Model	F1 score	Accuracy
Logistic Regression	0.8027	0.8044
Logit	0.2024	0.8041
Decision Tree	0.8333	0.8324
Random Forest	0.8928	0.8921
Perceptron	0.7583	0.7263
MLP Classifier	0.8509	0.8548
Logistic Regression with feature selection	0.7530	0.7555
Random Forest with feature selection	0.8810	0.8820
XGBoost	0.9025	0.9052

Table 12: Model comparison

After comparison it was concluded that XGBoost is the best model for predicting the rain tomorrow with the given variables. Its accuracy is 0.9052 and F1 score is 0.9025. Model is fitted with max depth of 40 and 100 was selected as n_estimator.

- For further clarification a confusion matrix and a heatmap was drawn and it also showed that XGBoost has good performance in predicting Rain Tomorrow with the given variables.

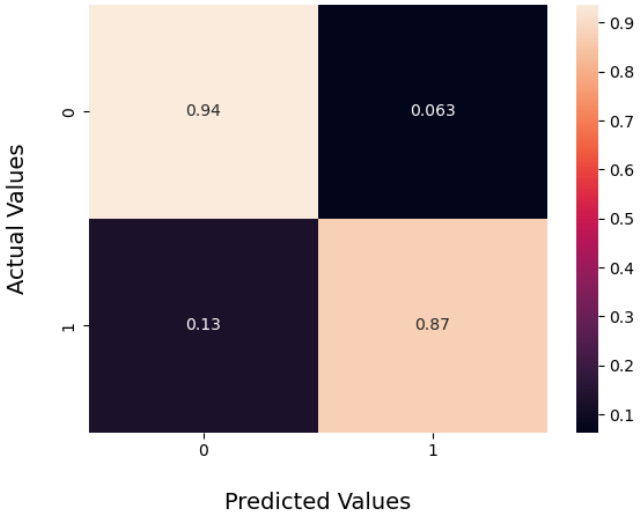


Figure 17:Heat map for confusion matrix

- By the feature selection, it was identified that Humidity3pm, Rainfall, Humidity9am, WindGustSpeed, Pressure3pm, Pressure9am, Temp3pm has more influence on rain tomorrow than the rest of the variables. (Using only these variables the XGBoost model able to predict with 0.8948 accuracy)

- The machine learning model XGBoost outperform the used neural network models in predicting rain tomorrow.

- Since the model built using XGBoost have a higher accuracy of predicting RainTomorrow even without using the location variable, this built model can be used to predict the rain in other locations too.

References

Hemalatha, G., Rao, S. K., & Kumar, A. D. (2021). Weather Prediction using Advanced Machine Learning Techniques. *Journal of Physics: Conference Series*.

Imon, R. A., Manos, R. C., & Bhattacharjee, S. (2012). Prediction of Rainfall Using Logistic Regression. *Statistics in the Twenty-First Century*, 665-667.

Jayasingh, S. K., Mantri, J. K., & Pradhan, S. (2022). Smart Weather Prediction Using Machine Learning.