QNT 730, Business Data Mining
Professor Ke Yang
Term Project – Team 7

Machine Learning Techniques to Predict Rain Tomorrow

Renuka Mukati, Ruhina Praveen, Shaily Prajapati
March 7, 2023

Abstract

The weather has a major affect on the agricultural industry and because of that, being capable to foretell it assists farmers in their day-to-day decisions as an example how to method efficiently, play down costs maximize yields. The goal is to predict if there is – no Rain or Rain tomorrow. This study introduces a set of experiments that signify the uses of current machine learning that can foretell in case it will rain tomorrow based on weather information for that day in main cities in Australia. The results provide a comparison of various evaluation metrics of these machine learning techniques and their reliability to predict the rainfall by analyzing the weather data. Three techniques explored are – Logistic Regression, Decision Trees, and Random Forests and on evaluation it has been found that with an accuracy of 91.2%, the Logistic Regression model has outperformed other two.

Introduction

A lot of study has been done on knowledge extraction from time series data. Time series data are data that are gathered in a precise fashion using time stamps. This kind of time-oriented data is gathered over a predetermined time period, like hourly, daily, or weekly. Predictions in a variety of fields and contexts, such as stock market movements, energy consumption estimates, and climate change, can be made with success using time series data. To uncover hidden patterns in past data and predict future trends, machine learning and data mining techniques can be used. The task of weather forecasting using historical data is difficult, but it has many advantages.

Literature Review

Throughout the past two decades, numerous researchers have focused on increasing the precision of machine learning algorithms used in weather forecasting. Here, several related studies are covered. Researchers described an ANN-based method to forecast atmospheric conditions. A variety of meteorological characteristics, such as humidity, temperature, and wind speed, were included in the dataset utilized for the prediction. The Back Propagation Network and Hopfield Network were combined in the proposed method to provide the HN with the output of the BPN as input. Investigating the non-linear relationship between historical weather attributes is how this method operates. In [19], scientists utilized ANN to forecast India's monsoon season's monthly average rainfall. The dataset was comprised of 8 months each year.

Two forecasting models for rainfall prediction were created in [28], the first using ANN to predict 1 month in advance and the second using ANN to predict 2 months in advance. For the experiment, a dataset from several regions of north India was employed. Levenberg-Marquardt training was combined with the Feed Forward Neural Network with Back Propagation approach in the model. Mean Square Error and Magnitude of Relative Error were used to examine the performance. The results showed that the 1-month forecasting model performed better than the 2-month model. To forecast rainfall, researchers created the Wavelet Neural Network (WNN) framework in [29]. The proposed remedy used ANN and wavelet technology. Using historical rainfall data, both models (ANN and WNN) were employed for prediction. According to the results, WNN outperformed ANN.

In [30], researchers presented an SVM-based application for the prediction of weather. A time series dataset related to the past n days from a location was analyzed, and then the maximum temperature of that location for the next day was predicted. By using optimal values of the kernel function, the performance of the proposed application was evaluated and found to outperform Multi-Layer Perceptron (MLP), trained with a back-propagation algorithm. To train the SVM, a nonlinear regression method was found to be suitable.

Table 1: Summary of previous related work.

Reference	Method	Accuracy %
D. Gupta et al. [1]	ANN-based classification model, with 10 hidden layers	82.1
D. Gupta et al. [1]	Classification and Regression Tree-based Prediction	80.3
D. Gupta et al. [1]	K nearest neighbor-based prediction, with k = 22	80.7

J. Joseph et al. [2]	ANN-based hybrid technique, integrating classification and clustering techniques	87
V.B. Nikam et al. [3]	Feature selection-based Bayesian classification model	91
N. Prasad et al. [4]	Decision Tree-based supervised learning in quest (SLIQ)	72.3

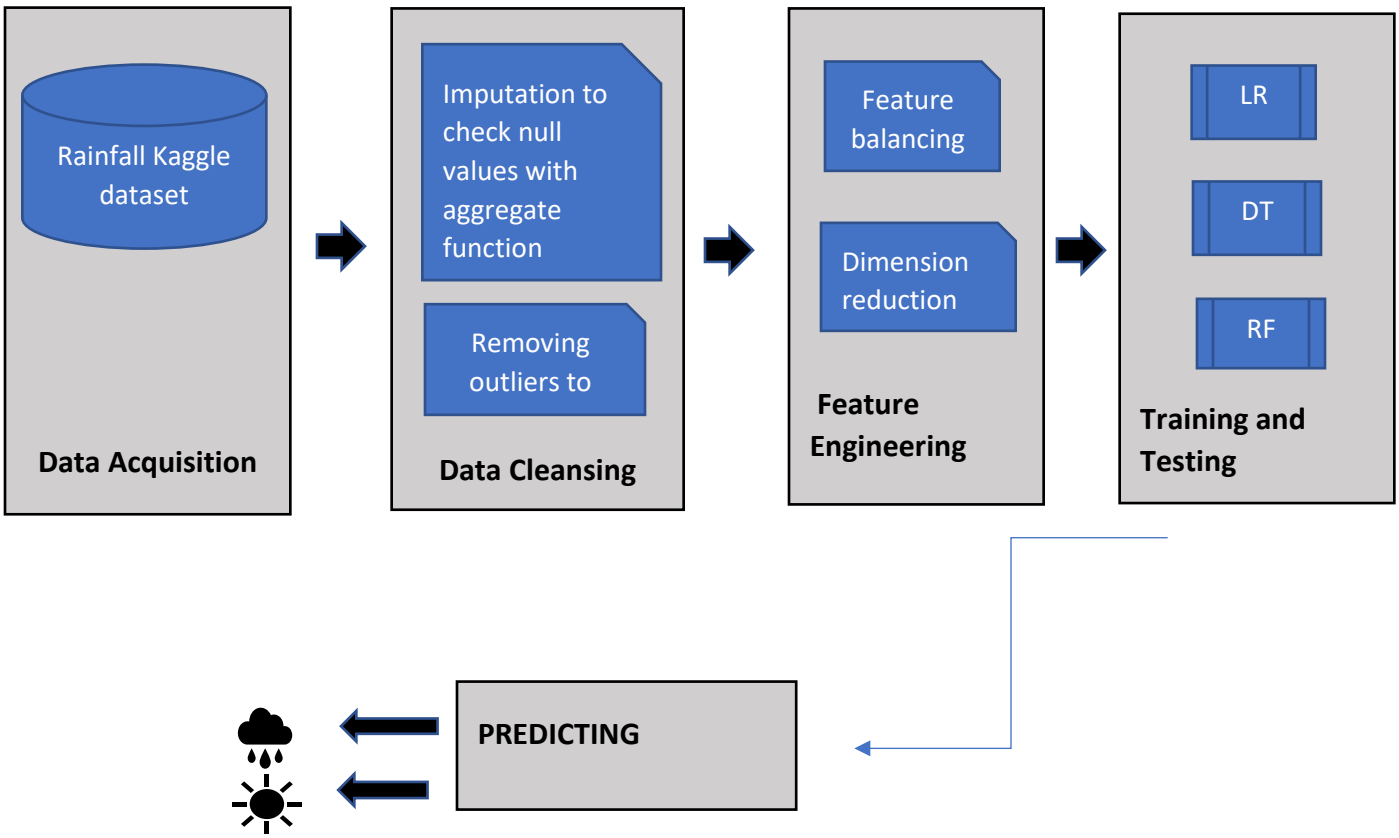
Data and Methodology

In this research, we have extracted a real-time pre-labeled dataset of rainfall prediction from numerous Australian weather stations. The dataset consists of 145,460 instances and 23 features, out of which 22 features are independent and 1 is dependent (output class). [Source]

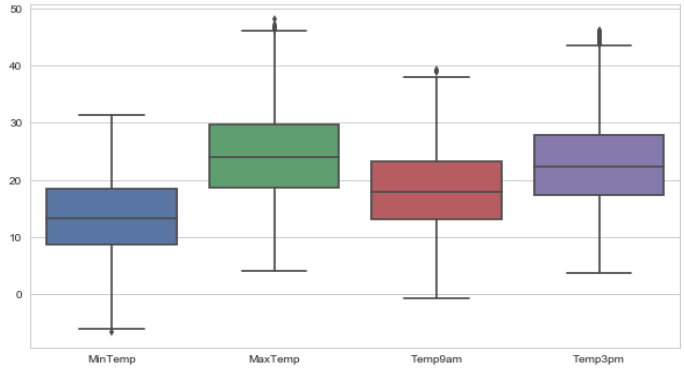
The data pre-processing stage consists of three activities: (1) cleaning, (2) normalization, and (3) splitting. The data cleaning process aims to remove the missing values in the dataset by using the technique of mean imputation. The normalization technique brings the attribute values within a particular range. In the third activity of the pre-processing stage, cleaned and normalized data is divided into two subsets: training data and test data, with a 70:30 ratio of class split rule. Predict whether or not it will rain tomorrow by training a binary classification model on target RainTomorrow. The target variable RainTomorrow means: Did it rain the next day? Yes or No

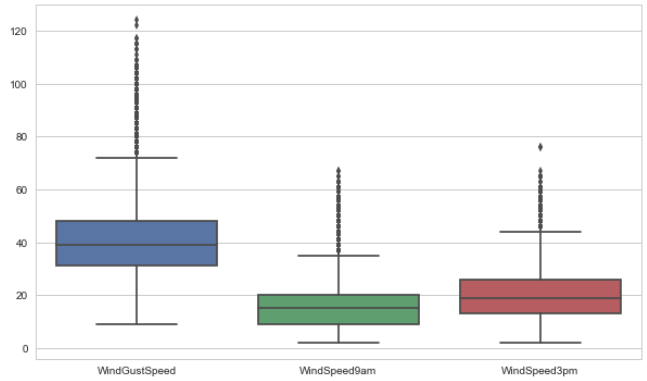
Further the train data has been fit for three classification algorithms:

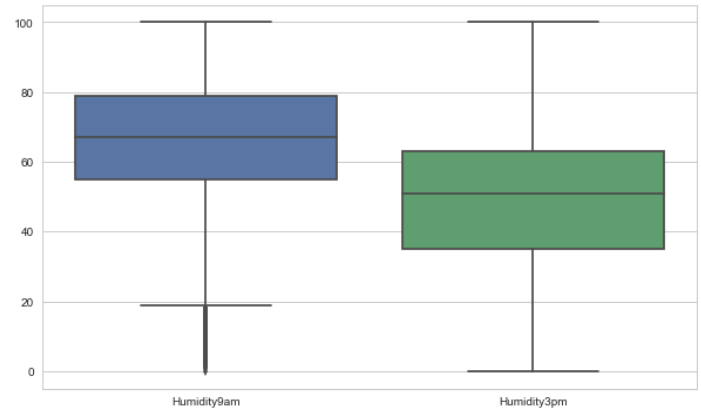
1. Logistic Binary Classification
2. CART
3. Random forest

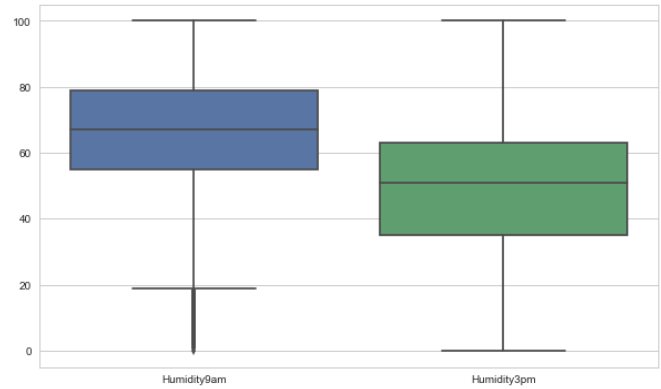


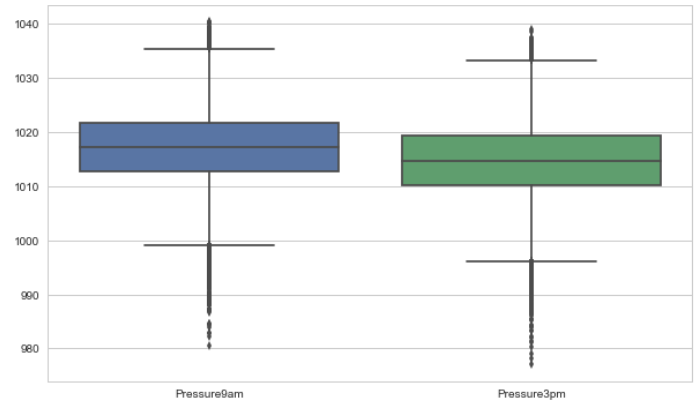
Some outliers represent natural variations in the population. To capture the true nature of data they should not be removed from the dataset as they are true outliers. Other outliers should be removed because they represent measurement errors, data entry or processing or sampling errors.











From the below box plot we can see that all temperature values are meaning full, no outliers found here. All wind speed values also are in sensible ranges. As we can see that there are some humidity values =0% which is almost never possible, hence removing 0 values. Here also there are no outliers, all pressure ranges also normally can happen in nature.After addressing issues with quantitative variables, next step is to process categorical data by feature encoding – Location, WindGustDir, `WindDir9am`, `WindDir3pm`, `RainToday`,`RainTomorrow`.

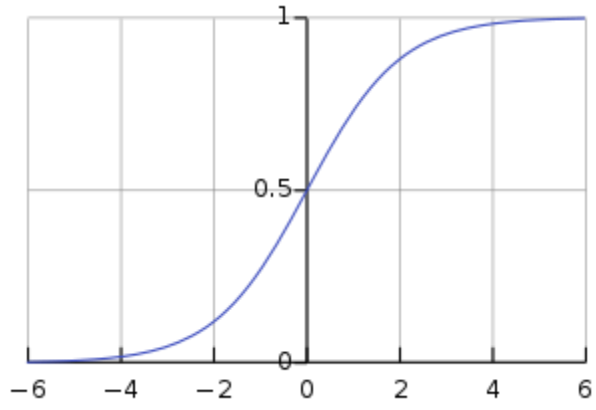
Methodology

Logistic Regression

Logistic Regression is a kind of parametric classification model, despite having the word 'regression' in its name. This means that logistic regression models are models that have a certain fixed number of parameters that depend on the number of input features, and they output categorical prediction, like for example if a plant belongs to a certain species or not. In Logistic Regression, we don't directly fit a straight line to our data like in linear regression. Instead, we fit a S shaped curve, called Sigmoid, to our observations. Sigmoid

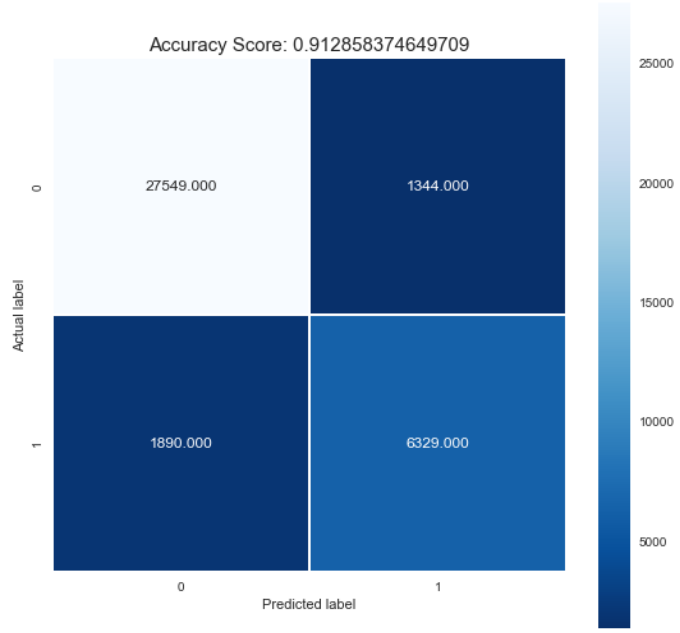
Function is used to scale z values between 0 and 1. But this is not the same thing as the normalization.

Sigmoid function is used for explaining probability. That means our model's prediction's result is 1. Because all y hat values above 0.5 (threshold value) on the graph are 1 in the sigmoid function graph. If we have y hat = 0.4 that means our model's prediction is 0.



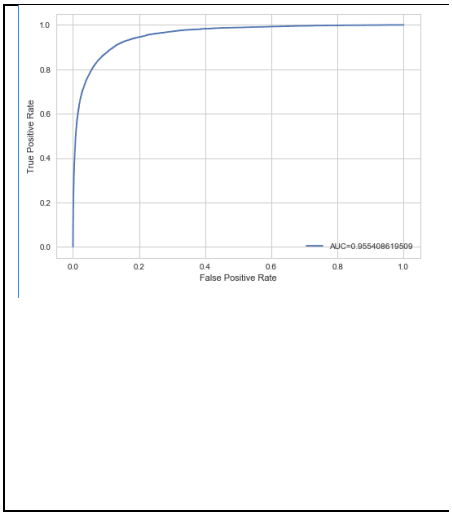
Here is the sigmoid function's graph:

Accuracy is defined as the number of correct predictions over the total predictions.

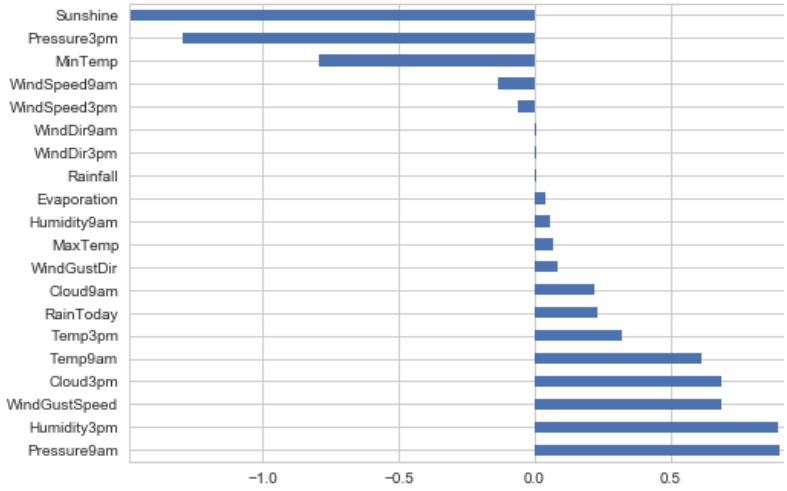


With the confusion matrix the accuracy of Logistic Regression is 91.2%

Area under the curve = 95.6



Feature Importance



Predictors are listed down based on their feature importance. To sum up, the most important feature in predicting whether it will rain tomorrow or not is Sunshine. One unit increase in the sunshine will decrease the odds of rain tomorrow by 1.49 times.
Also, Wind direction at 9 am and 3 pm bear no significance in the prediction model. Taking them out of the model will help reduce the dimension and hence in turn improving the performance.

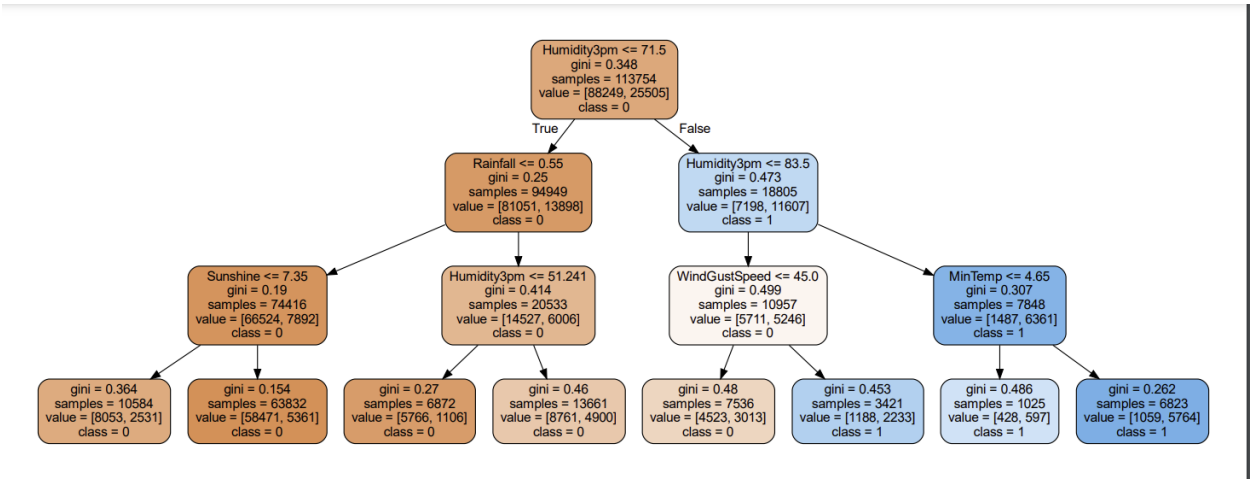
CART Classification

Gini Impurity

The CART algorithm is used to build the classification model using the Gini Impurity as the criterion to evaluate the quality of a split. The Gini Impurity measures the degree of probability of misclassification, which means it measures how often a randomly chosen element from a set would be incorrectly labeled if it were randomly labeled according to the distribution of labels in the subset. This model is trained on all available features.
The CART model was evaluated using the Gini impurity criterion on a test dataset. The model achieved an accuracy score of 0.828, which indicates that the model classified 82.8% of the instances correctly. The precision for class 0 (no rain) was 0.84, indicating that when the model predicted no rain, it was correct 84% of the time. The recall for class 0 was 0.97, indicating that the model correctly identified 97% of all instances of no rain. The f1-score for class 0 was 0.90, which is a harmonic mean of precision and recall.
On the other hand, for class 1 (rain), the precision was 0.76, indicating that when the model predicted rain, it was correct 76% of the time. The recall for class 1 was 0.34, indicating that the model correctly identified 34% of all instances of rain. The f1-score for class 1 was 0.47.
Overall, the model performed well in predicting class 0, with high accuracy, precision, and recall scores. However, the model struggled with predicting class 1, with lower precision

and recall scores. The macro-average f1-score was 0.68, indicating that the model performed moderately well overall.
It is important to note that the model was trained on a training dataset, and it achieved a similar performance on the test dataset, indicating that it did not overfit to the training data. The model had a depth of 3 and 8 leaves, which suggests that it is not a very complex model, but it could be easier to interpret than more complex models.

```
*************** Tree Summary ***************
Classes:  [0 1]
Tree Depth:  3
No. of leaves:  8
No. of features:  17
-----------------------------------------------------------

*************** Evaluation on Test Data ***************
Accuracy Score:  0.828439818559021
              precision    recall  f1-score   support

           0       0.84      0.97      0.90     22067
           1       0.76      0.34      0.47      6372

    accuracy                           0.83     28439
   macro avg       0.80      0.65      0.68     28439
weighted avg       0.82      0.83      0.80     28439
```



Gini Impurity and limited features

In this model, we use feature selection techniques to select a subset of features that are most relevant to predicting the target variable. We train the CART model using the selected features and the Gini Impurity as the splitting criterion. This technique reduces the dimensionality of the data, which leads to a simpler and more interpretable model.

The given metrics show the performance of a CART model trained using Gini impurity on a binary classification problem with two features.

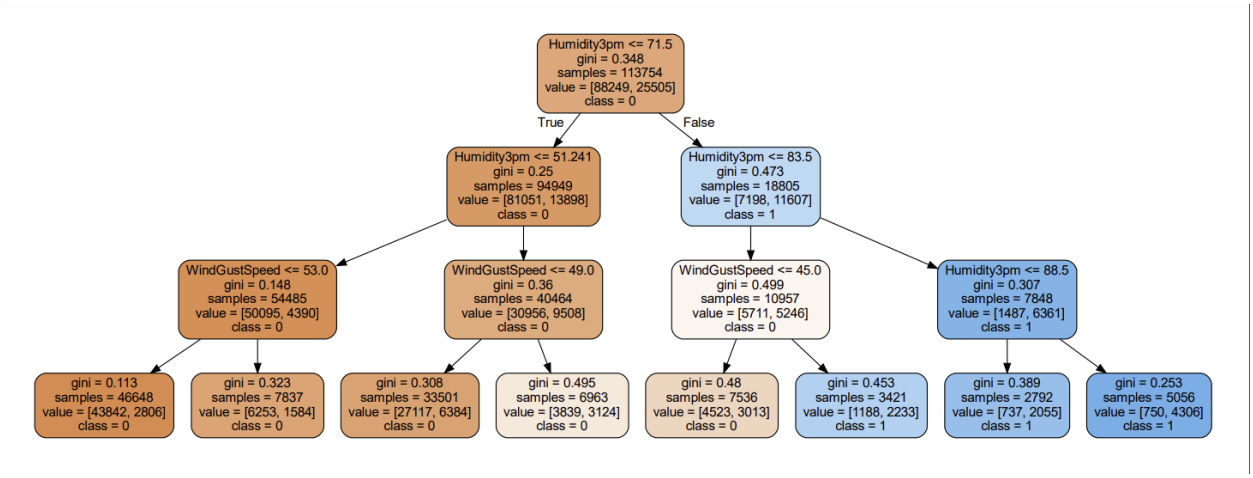
The model has a tree depth of 3 and 8 leaves, indicating that it is not too complex and prone to overfitting. The number of features used to train the model is only 2, which could suggest that the model is underfitting and could benefit from more features.

The accuracy score on both the training and test data is around 83%, which is relatively good. However, the precision, recall, and F1-score for the positive class (1) are relatively low compared to the negative class (0) on both the training and test data. This suggests that the model is better at identifying the negative class than the positive class.

The macro-average F1-score on the test data is 0.68, indicating that the model's performance is slightly above average. The weighted average F1-score is 0.80, which takes into account the class imbalance and indicates the overall performance of the model.

Overall, the model has performed reasonably well, but it could benefit from more features and further tuning to improve its performance on the positive class.

```
*************** Tree Summary ***************
Classes:  [0 1]
Tree Depth:  3
No. of leaves:  8
No. of features:  2
---------------------------------------------------------

*************** Evaluation on Test Data ***************
Accuracy Score:  0.828439818559021
              precision    recall  f1-score   support

           0       0.84      0.97      0.90     22067
           1       0.76      0.34      0.47      6372

    accuracy                           0.83     28439
   macro avg       0.80      0.65      0.68     28439
weighted avg       0.82      0.83      0.80     28439

---------------------------------------------------------
```



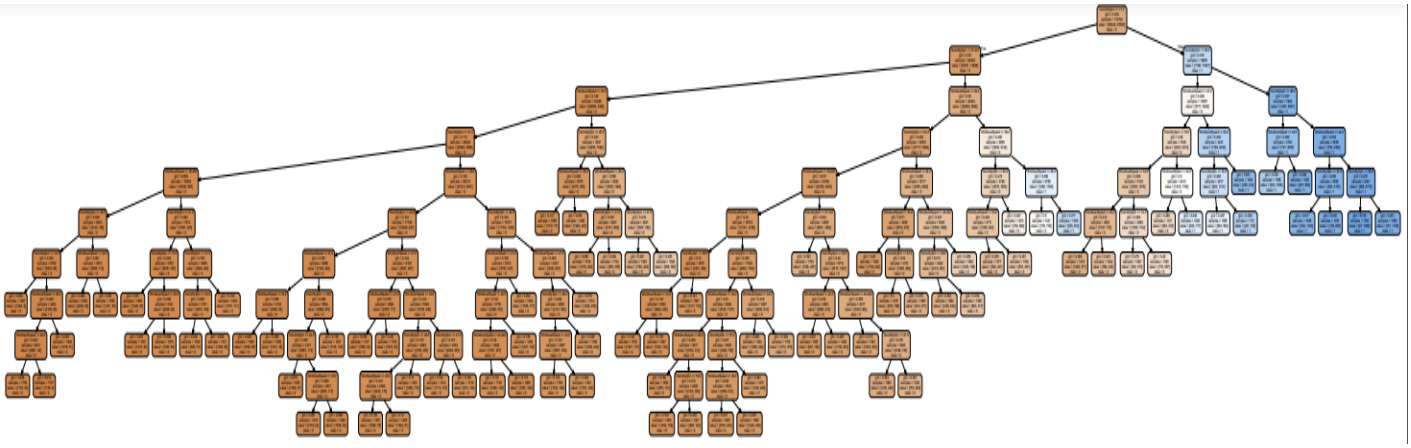
Unlimited tree depth

In this model, we allow the tree to grow to its maximum depth, which means we do not restrict the complexity of the model. This model is trained on all available features.

The CART model with unlimited tree depth has a tree depth of 10 and 82 leaves. It was evaluated on both the training and test data sets. The model has 2 features which it used to make predictions.

On the test data set, the model achieved an accuracy score of 0.8335. This means that the model predicted the correct class label for 83.35% of the observations in the test data set. The model has a precision of 0.85 for class 0 and 0.73 for class 1. This means that when the model predicts an observation as belonging to class 0, it is correct 85% of the time. Similarly, when it predicts an observation as belonging to class 1, it is correct 73% of the time. The recall for class 0 is 0.96, which means that the model correctly identifies 96% of the observations belonging to class 0. The recall for class 1 is 0.40, which means that the model only correctly identifies 40% of the observations belonging to class 1. The F1-score is a weighted average of precision and recall, with values ranging between 0 and 1, where 1 is the best possible score. The F1-score for class 0 is 0.90, and for class 1 is 0.52.

On the training data set, the model achieved an accuracy score of 0.8317. This means that the model predicted the correct class label for 83.17% of the observations in the training data set. The model has a precision of 0.85 for class 0 and 0.73 for class 1. This means that when the model predicts an observation as belonging to class 0, it is correct 85% of the time. Similarly, when it predicts an observation as belonging to class 1, it is correct 73% of the time. The recall for class 0 is 0.96, which means that the model correctly identifies 96% of the observations belonging to class 0. The recall for class 1 is 0.40, which means that the model only correctly identifies 40% of the observations belonging to class 1. The F1-score for class 0 is 0.90, and for class 1 is 0.52.



Overall, the model has performed reasonably well with an accuracy score of around 83% on both training and test data. The precision is higher for class 0 and lower for class 1, while the recall is higher for class 0 and lower for class 1. The unlimited tree depth may have caused overfitting, and the model may not generalize well to new data. Therefore, it may be necessary to apply some regularization techniques to avoid overfitting and improve the model's performance.

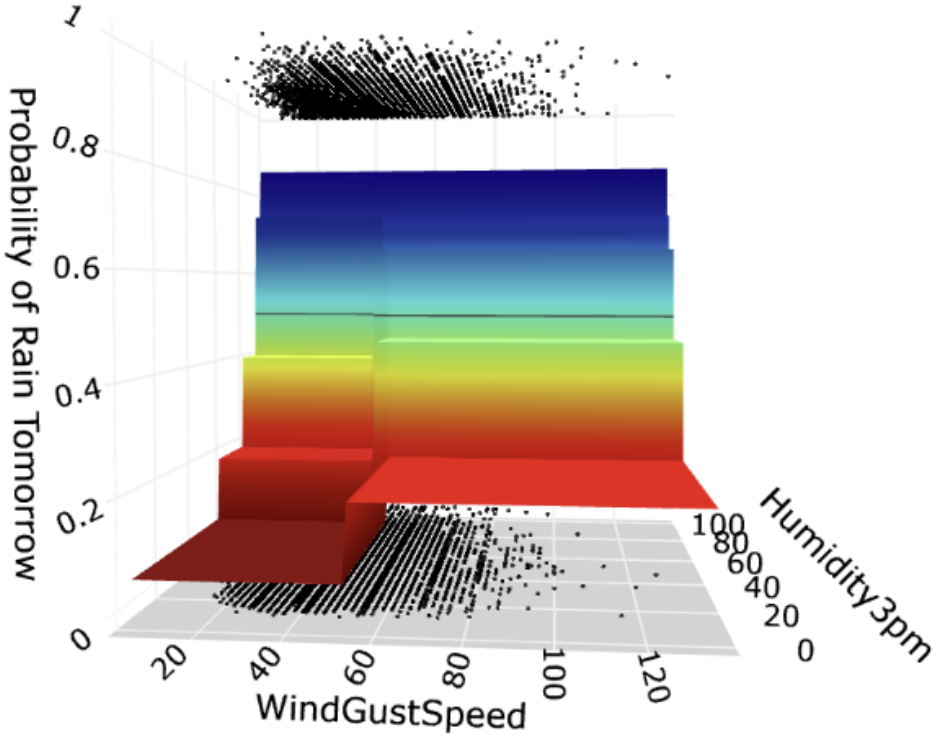
```
*************** Tree Summary ***************
Classes:  [0 1]
Tree Depth:  10
No. of leaves:  82
No. of features:  2
---------------------------------------------------------

*************** Evaluation on Test Data ***************
Accuracy Score:  0.8335032877386688
              precision    recall  f1-score   support

           0       0.85      0.96      0.90     22067
           1       0.73      0.40      0.52      6372

    accuracy                           0.83     28439
   macro avg       0.79      0.68      0.71     28439
weighted avg       0.82      0.83      0.81     28439
```

Random Forest

Decision trees are constructed by analyzing a set of training examples for which the class labels are known. [5] Random Forests essentially just a CART algorithm where it creates an ensemble of many trees. Random forests can handle both classification and regression. For the purpose of this study, we focus only on classification. Using classification, we attempt to predict a class label. We will use decision tree to predict a variable that has two categories = whether it will rain tomorrow or not using sklearn, ensemble and RandomForestClassifier. For building the random forest model, we use Bootstrap aggregation(random sampling with replacement,) and feature randomness.

After splitting the data, we set model parameters and train the model. Then we predict the class labels on both the train and test data and general summary statistics for the model. We used an ensemble of 1000 decision trees for this model for which the base is a CART model – previously used algorithm of choice. Below is the summary of the model's performance.

```
Model Summary

Evaluation on Test Data
Accuracy Score:  0.8345933401315095
              precision    recall  f1-score   support

           0       0.84      0.98      0.90     22067
           1       0.80      0.35      0.48      6372

    accuracy                           0.83     28439
   macro avg       0.82      0.66      0.69     28439
weighted avg       0.83      0.83      0.81     28439


Evaluation on Training Data
Accuracy Score:  0.8326036886614976
              precision    recall  f1-score   support

           0       0.84      0.97      0.90     88249
           1       0.79      0.34      0.48     25505

    accuracy                           0.83    113754
   macro avg       0.81      0.66      0.69    113754
weighted avg       0.83      0.83      0.81    113754
```



With an overall accuracy score of around 83% on both training and test data, this model has done well. We have obtained an accuracy level of 83.46% for evaluation on test data and an accuracy score of 83.26% for the training data. Similar to the CART model, the precision is higher for class 0 and lower for class 1while the recall is higher for class 0 and lower for class 1. This suggests that the model is better at identifying observations belonging to class 0 than class 1. However, the f1-score is low for class 1, which means that the model is not as good at predicting observations belonging to class 1. The summary statistics for the random forest model are striking similar to that of the CART model.

Conclusion

The task of predicting weather through historical data is tricky and often unreliable owing to many other factors of influence like the decade in which the data was collected. However, weather predictions are becoming increasingly important as these predictions now has economical ramifications. From planning holidays to business decisions, weather is considered. We used three models to predict the weather data out of which Logistic Regression yielded the most accuracy by 91.2% followed by a tie between the CART and Random Forest at about 83%. Also notice that the summary statistics for the random forest model are striking similar to that of the CART model. The results are a bit surprising as Logistic Regression models perform better with numerical data in the target feature rather than categorical. However, since the base of the Random Forest model was classification

and CART algorithm, it is not surprising that both the model's accuracies are comparable. We modelled Random Forest using only classification as our preliminary hypothesis was that the decision tree classification models would perform better than the logistic regression one. For future enhancements, we would like to also model the Random Forest using regression to be able to compare it to the logistic regression model.

References

1. Gupta D., Ghose U. A comparative study of classification algorithms for forecasting rainfall; Proceedings of the 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) Trends and Future Directions; Noida, India. 2–4 September 2015; pp. 1–6. [Google Scholar]

2. Joseph J. Rainfall Prediction using Data Mining Techniques. *Int. J. Comput. Appl.* 2013;**83**:11–15. doi: 10.5120/14467-2750. [CrossRef] [Google Scholar]

3. Nikam V.B., Meshram B.B. Modeling rainfall prediction using data mining method: A bayesian approach; Proceedings of the International Conference on Computational Intelligence, Modelling and Simulation; Bangkok, Thailand. 24–25 September 2013; pp. 132–136. [Google Scholar]

4. Prasad N., Kumar P., Naidu M.M. An Approach to Prediction of Precipitation Using Gini Index in SLIQ Decision Tree; Proceedings of the 4th International Conference on Intelligent Systems, Modeling and Simulation; Bangkok, Thailand. 29–31 January 2013; pp. 56–60. [Google Scholar]

5. Nat Biotechnol. What are decision trees?; National Library of Medicine, National Center for Biotechnology Information. 26 September 2008. [NCBI]