Rainfall Analysis

Task-1 Data Exploration:

What is the proportion of rainy days to fine days?

Proportion of rainy days to fine days is 0.240 as it is just calculated by taking rainy days based on any other given days.

The summary for the dataset is available in the R section. In the same, the values for "WindGustDir", "WindDir9am", "WindDir3pm", "RainToday" and "RainTomorrow" are in factor form so there cannot be any data exploration possible for the above-mentioned variables, not in quantitative way at least. Therefore, they have been removed heading into summary table. Day, Month, Year and Location are the attributes does not have any effect on any other variables, so it is assumed to exclude them to avoid any unnecessary overlapping in the model in future.

Looking at the description tables for the rest of the attributes, the data consists of 744 rows after removing all the null values. The attribute MinTemp, MaxTemp, Pressure9am, Temp3pm have closed to 0 skew meaning they have symmetrical graphs in their distribution and are skewed. There are also variables like Humidity3pm, Cloud9am, Cloud3pm, Pressure3pm, Temp9am which are moderately skewed closely with median. However, the data for Rainfall is positively highly skewed which makes the graph having long tail and makes it difficult for predicting future values. There, the peak for the graph will be on the left of the average values and the data will become inconsistent as it takes long to reach the average value. Hence, it would be best to exclude Rainfall values in the assumption for this dataset as the values do not provide consistency for this case.

From the standard deviation values, we can say that higher the standard deviation is, the sparser data will be. In the other case, the data required for the classification models needs to be varying range to range; it means we would have to have dataset with columns where we could classify and fit the model based on range of values in the data such that it could give results without any limitations of the values in data. Therefore, we have “Humidity3pm”, “Humidity9am”, “WindGustSpeed” which will be one of the most important variables to be used to fit any model for the given dataset. We cannot say anything specific, but they are most important to be used while fitting followed by other variables in the order. There are also other factors needs to be considered at any given stage while fitting the model.

Task-2 Pre-Processing Documentation:

As part of the pre-process of the dataset, the NA values from the dataset have been removed. It is decided to be removed as they could lead to completely different results if we try to alter the values. Imputing any types of values in place of NA can end up with change of model which can be ambiguous at any time while testing the data. By removing rows having NA, we do lose some data, but we have a lot to start with to make the classification predicting models. Therefore, removing rows of Na values from the data is part of the cleaning process.

The attributes Day, Month, Year and Location have not been included in any classification model since we are assuming that it is not best to predict whether or not it will rain tomorrow based on the specific calendar or geographical location. It will lead to many other factors in higher dimensions, alternatively it will drift away the focus from the central perspective, so it is assumed to exclude Day, Month, Year and Location from the models.

There is an importance point is noticed for the Rainfall and RainToday variables. It is assumed to be considered two different variables. From the summary data, they both had their separate interpretation where Rainfall seemed to have skewed data and RainToday is factor value in which we do not know what exact value it falls on. Therefore, it is assumed to keep both variables into model and let the algorithm decide on how further it contributes to the model.

According to the classification models definition, the variable being predicted needs to be in the factorized form so RainTomorrow is converted into factor form just in case it is not already.

All the libraries required for the specific models can be seen being imported at the start of the given section.

Task-3 Training and Testing Dataset

The seed will be set before every time any model is fitted to the training data to avoid any inconsistent results. We will use 70% of the data to train the model and other 30% to test the model.

Task-4 Classification Model Implementation

The task is to implement and fit all the given models such as Decision Tree, Naïve Bayes, Bagging, Boosting and Random Forest.

In task 2, It was made sure that RainTomorrow variable is in factor form.

The steps on how the algorithm is working can be found in the coding part of the document.

Task-5

The confusion matrix for all five models have been created in R after predicting the model for the testing dataset. The accuracy (in %) for the models be below:

1. Decision Tree Accuracy: 76.78%

2. Naive Bayes Accuracy: 82.14%

3. Bagging Accuracy: 83.93%

4. Boosting Accuracy: 82.59%

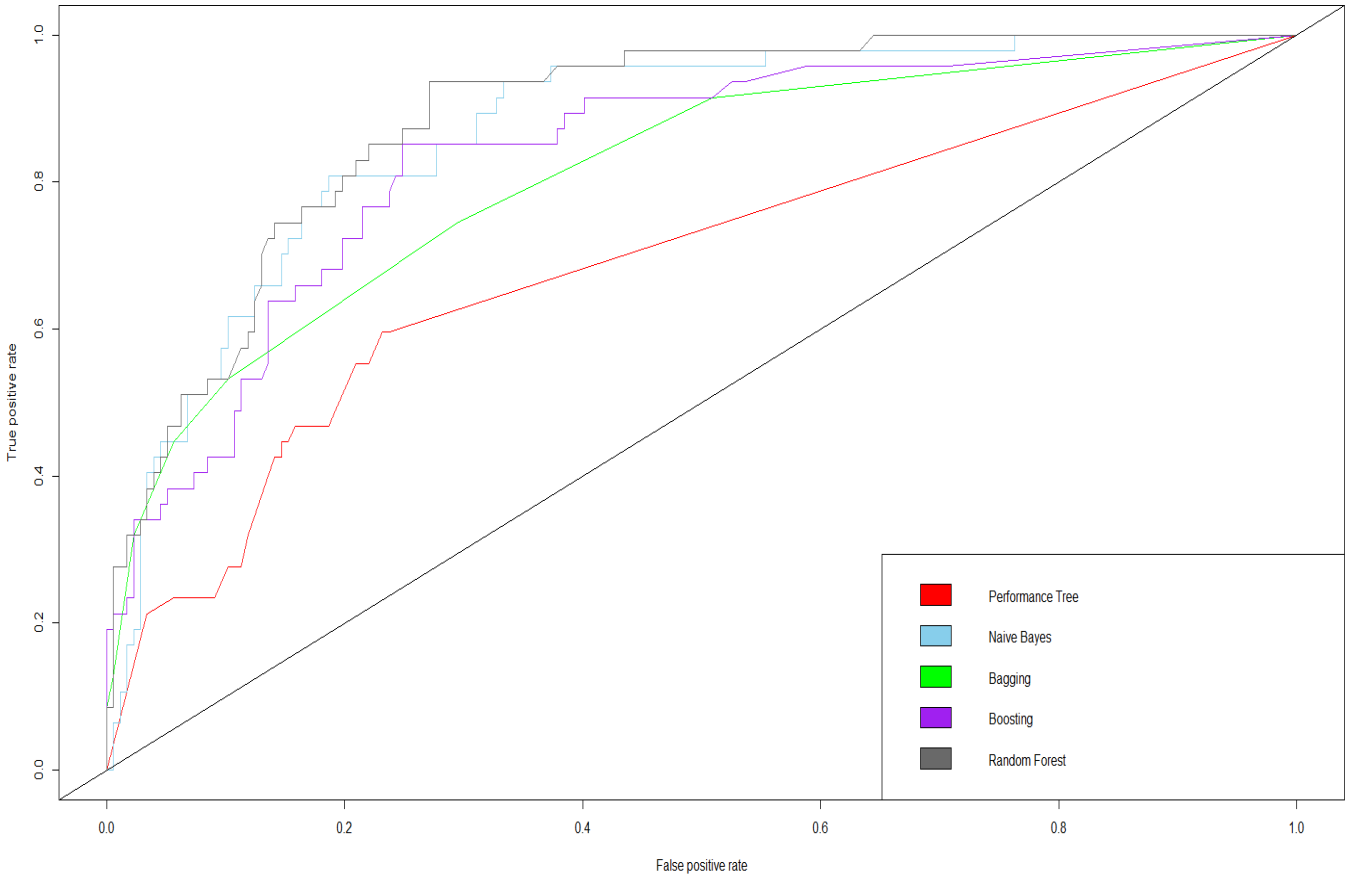
5. Random Forest Accuracy: 83.93%

Since all the preprocessed variables are used in fitting the above five models, we get the highest accuracy for Random Forest and Bagging. Decision tree has relatively lower accuracy but there could be a way to improve it by pruning the tree and running cross validation over the model again which we will learn in the later parts.

Task-6

The confidence of predicting the RainTomorrow variable have been calculated as the model have been predicted in the form such that ROC curve can be constructed and further area under curve is calculated to give more specific results. The details can be seen in the R file.

The constructed ROC curve can be found as follow:



The area under curve is calculated for all five models and it is stated (in %) as below:

1. Decision Tree AUC: 68.96%

2. Naive Bayes AUC: 87.03%

3. Bagging AUC: 81.08%

4. Boosting AUC: 83.61%

5. Random Forest AUC: 88.70%

As it can be seen, Random Forest has relatively high area under curve which helps fitting model better than any other one. Decision tree has lower AUC as compared to other models here.

Task-7

The table comparing the results from both tasks as in comparing the accuracy and area under curve value between models are fitted into table as below:

Number	Model Name	Accuracy (%)	AUC (%)
1.	Decision Tree	76.78	68.96
2.	Naive Bayes	82.14	87.03
3.	Bagging	83.93	81.08
4.	Boosting	82.59	83.61
5.	Random Forest	83.93	88.70

We have calculated the value for Accuracy and Area under curve for all the given models and the result of both is compared for five models as in the table.

As we concluded in the previous two parts as well that Bagging and Random Forest have the joint topmost accuracy (83.93%) among five models. However, the best value of area under curve goes to Random Forest. Which means Random Forest not only has the highest accuracy but also it has the most area under curve values. And that makes Random Forest the most stable model for this dataset which will provide the most accurate results and having AUC of 88.7% is an excellent discrimination. Alternatively, the task of classifying data with appropriate predictors to make sure 'Yes' or 'No' value for RainTomorrow variable goes into correct class is the most important and Random Forest gives us the surety that we are 83.93% confident that the correct value will go into correct class.

For the given testing data, there will be 36 misclassifications among 224 observations while testing model for Random Forest and Bagging. While for Decision Tree, we have accuracy of 76.78% resulting in 52 misclassifications during testing the model with the same testing dataset. For Naïve Bayes and Boosting, we have accuracy of 82.14% and 82.59% which ends up with 40 and 39 misclassifications correspondingly. That is why, Decision Tree has the lowest accuracy, Naïve Bayes and Boosting have the similar accuracy (Boosting slightly better) and Random Forest and Bagging has the best accuracy with only 36 misclassifications.

ROC curve is used to compare between the models to see which one will work better to avoid the false alarm or false positive rates in the data. Here, having the best AUC would make the model stable. We have Random Forest with the highest AUC of 88.7% and we could see in the confusion matrix of Random Forest that it has the least false positive numbers than any other; plus it does neither underfit nor overfit the data which makes it perfect. While for others, Naïve Bayes has 87.03% AUC followed by Boosting (83.61%), Bagging (81.08%) and Decision Tree (68.96%). Decision tree has the lowest area under curve as having low accuracy (low TPR) is also affecting this value so improving the tree by its accuracy could result in improvement of AUC value.

However, Random Forest has again the best value for Area under curve. Having most accurate value and the most AUC would give more accurate result than any other model.

Finally, Random Forest is the single best classifier.

Task-8 Most Important Variables Identification

The method to find the most important variables have been described in the R file where the important variables are shown directly for the decision tree, bagging provides with the relative comparison of all the contribution made by the variables present in the data, boosting gives the proportional contribution of each variable to information gain in the weights form and random forest has contributions of predictors by creating multiple datasets from the original training data, build a decision tree for each new data and combine the classifiers by taking a majority vote to calculate weights. Unfortunately, Naïve Bayes do not have any approach to check for the important variables in the most so we cannot use that model.

But after looking at the important variables and their weights in the given models which can be found in the code section, the most important variables found in the classification models of Decision Tree, Bagging, Boosting, Random Forests are:

Humidity3pm, Winddir9am, WindgustDir, Sunshine and Winddir3pm.

The decision is taken based on the overall rank of these variables in all four models as above.

The ranks are based on how much better each predictor contributes to the model than others. Humidity3pm is the most important variable which contributes the most in both Bagging and RandomForest and fairly decent in Boosting. Winddir9am contributes the second most as it is ranked first in Boosting and decent in other two models whereas WindgustDir contributes second best in all models. Also, Sunshine, Winddir3pm have their contributions very good as they are ranked fifth and fourth in Bagging, third and fourth in Boosting, third and fifth in Random Forest correspondingly. In short, these five variables contribute much higher than any other variables in all the given models, so they are ranked as the most important variables on predicting the RainTommorow variable.

On the other side, the variables that have been found to be having very little effect on the models are: RainToday, Temp3pm, Cloud9am, WindSpeed3pm, WindgustSpeed. These variables have effect on models as low as less than 1 relative contribution to their models. So, even if these variables are omitted from the dataset, the model would not be affected; instead, it could lead to better results. Furthermore, the remaining variables (neither most important nor with less effect) are fairly important to the model but not as much important as the ones stated as most important variables on predicting the rain tomorrow. Such variables are contributing fairly to the construction of model, so they are not removed.

Task-9

Best tree-based classifier: Using cross validation to improve decision tree and Improving the random forest using the ensemble modelling.

First, decision tree does not perform well on this dataset. So, in an attempt to improve the model, we used cross validation to fit the improved tree. Remember, we have only considered the most important variables found from the last task to fit the pruned decision tree since we want to avoid unimportant variables to end up having more misclassifications. In the same, we prune the tree based on the misclassification numbers while predicting the RainTommorw classes.

The pruned decision tree in the appendix:1.1 graph leads to cleaner result as compared to original decision tree in task-4 in R file.

As from the implementation part, the size 4 leads to the least number of misclassification so we prune the decision tree with the size 4 and summarizing the tree ends up with improved accuracy of 80.35% from 76.78% and AUC of 79.41% from 68.96%. Therefore, we have nearly 4% improvement for accuracy and 10.45% improvement in AUC value when we prune the decision tree.

As it seems, the ROC curve is better in pruned decision tree (appendix: 1.2) as it falls under acceptable discrimination.

However, the pruned tree still does not result in the best tree-based algorithm among the classification models we have implemented in the previous tasks. We have Random Forest being the best model so fa r with 83.93% Accuracy and 88.70% AUC value.

We have tried taking an approach which is to improve the random forest model by simply adding more trees into the model and decreasing the number of nodes at the split of each node. On top of that, we are excluding the zero-effect attribute into the model as it might affect the model in negative way. However, we will include all the remaining attributes as all those attributes contribute fairly equal to the model meaning no big gap into the values. This would mean we will take all those attributes except the ones removed as part of preprocessing and low effect one which is RainToday. By increasing the number of trees into the data, we will be able to increase accuracy as it will provide variety into the data. That means area under curve value can be improved too after resulting into less misclassifications into the false positive class. The number of nodes will be set to 2, the minimum one to control the model to not get overfitted. Therefore, the accuracy is 0.91% improved and AUC value is slightly better which improved by 0.27%.

In the end, we get the improved accuracy of 84.82% which is a little better than 83.93% which is 0.91% improvement from the original model and improved AUC of 88.97% which is better than the original AUC of 88.7%. ROC curve can be seen improved in Appendix-1.8.

The variables used is one of the most important factor while improving the model and number of trees c ould help in positive way on how we could improve the accuracy values into the data and it helped redu cing misclassification to 34 which is slightly better than the original. It is the best model for Random Fore st classification; however, by using ensemble method with combinations of different models, we could e xtract even higher accuracy for the data given.

Another approach taken to improve the accuracy is by using ensemble methods which is a process where different models are combined in a way such that it produces the better outcome. It is believed that combining multiples can produce better results by decreasing generalization error. We will use stacking to allow multiple models to be combined and producing the effective accuracy using Random Forest method. In this model we have chosen the attributes based on the function rfcv() output. The function is useful to check how much misclassification and model fitting error it produces based on the number of attributes you select. As it can be seen in Appendix:1.3, it will provide much less error for variable 5, 10 and 20. But they are all close to each other which means we will take the smallest possible variables. Another reason for taking five variables are that we get the exact same number of variables from the last part while choosing for important variable. Therefore, five variables for the chosen for the ensemble random forest tree are: SunShine, WindgustDir, WindgustDir9am, WindgustDir3pm, Humidity3pm which are also the top-5 most variables contributing to the random forest model.

Since package caretEnsemble() gives the flexibility to use multiple algorithms to be used in number of iterations and repetitions. That way, we have the independent accuracy for each model. We are using repeated cross validation method to assign iterations for the models. The models we are using in the stacked are Recursive Partitioning and regression trees, logistic regression, K- Nearest neighbour classification and support vector machine (which does not use any probabilistic model instead generates hyperplanes to classify data into different regions). We have powerful results for svmradial followed by logistic regression, knn, rpart and then followed by random forest which all are close to each other. So, they all provide great correlation between each other and end up giving good results. But by stacking the models using random forest provides us with the more accurate result of 85.13%. Since it provides different accuracy for each node size, we have the best accuracy of 85.13% for node size=2 (mfinal=2) which was the crucial factor in getting the improved accuracy. So, we get the result 85.13% accuracy for when the classification splits into two nodes. Another factor found quite useful was the logistic regression as it helps combing all the models into linear form and ensemble them to make sure we get the best average accuracy considering the values for all models.

Therefore, Ensemble models with Random Forest improves the accuracy by 0.31% which is the most accurate model among all the models. Hence, Ensemble Random Forest model is the best classifier for this dataset.

In conclusion, the accuracy here is based on the dataset and having more data with more observation can certainly improve the model further which would provide with even more accurate results.

Task-10 Neural Network Implementation

Applying the Artificial Neural Network on this dataset gives the accuracy of 84.56% and AUC value of 69.12%.

First thing to note in the algorithm that we would not need hidden layers in the model as data is not separated non-linearly. Having data into scalar form has made it easy to fit the graph linearly here. As part of pre-processing, we made use of only the most important predictors in the model as found before which are Windgustdir, Windgustdir9am, Windgust3pm and Humidity3pm. The reason for selecting these variables is to stick by the importance found in classification models before and they would make good contributions in neural network as they have previously. Another reason is that they all have great correlation between each other as wind direction attributes are already corelated and similar in their data-exploration section.

All the variables to be used in the model needs to be converted into numerical as neural network only works with numerical data. For that, we converted RainTomorrow to binary variable 0 and 1 and converted variables Windgustdir, Windgustdir9am, Windgustdir3pm into numerical form. All the data needs to be normalized too as some of them have very large units and others have smaller units, so they are all converted into scalar form to make it work equally for every predictor. We are using 80% data to be trained on and remaining 20% to be tested on. As computing function runs on the remaining 20%, we get the prediction into confusion matrix where we get the accuracy of 84.56% with 23 misclassifications. The accuracy here is slight (around 1%) better than Random Forest and Bagging (not better than improved random forest though) we have seen before as neural network only has 23 misclassifications

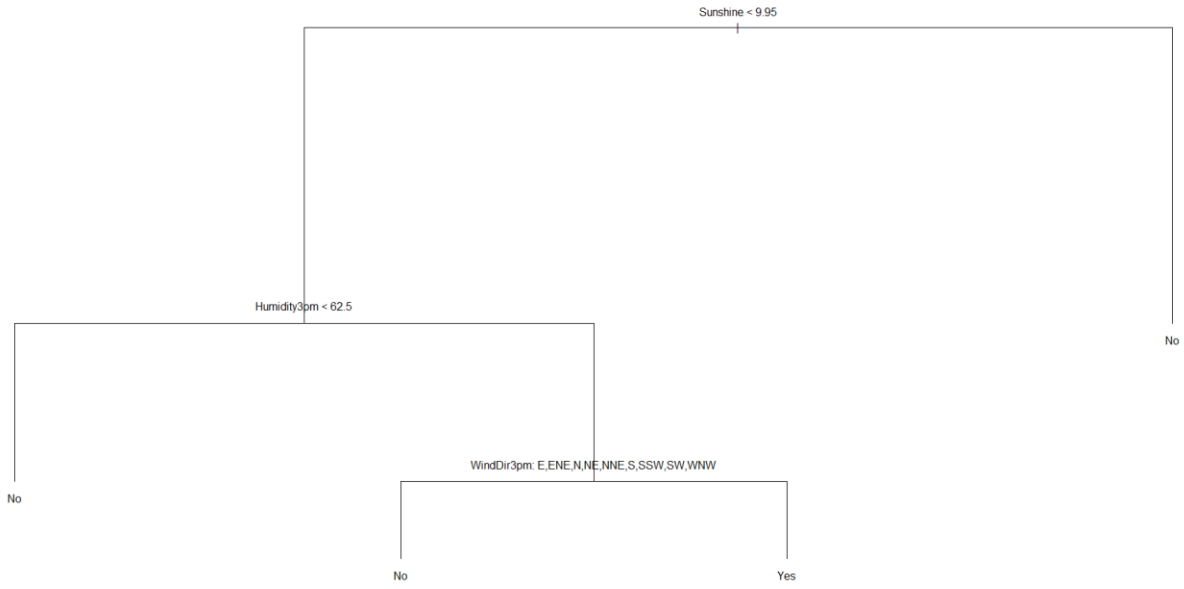
out of 149 observations. However, the biggest issue gets in the Area under curve value where it is found to be 69.12% which is the highly dropped as compared to Random Forest. The issue here is that we have false positive rate gets higher as we try to fit in the model accurately. ANN would underfit the model while trying to increase the accuracy and that is the reason, we get 21 out of 23 misclassifications to be false positive. The ROC curve can be seen in the graph as in Appendix:1.4.

Therefore, ANN is not the best model to predict the RainTomorrow as it would not give the best area under curve, not in this dataset at least. Random Forest has succeeded in having both accuracy and auc value to make the best classifier. Therefore, ANN might provide with great accuracy but in this dataset, it comes up with lower auc which would underfit the data with no hidden layer. But it will also overfit if we add a hidden layer which ends up decreasing accuracy.

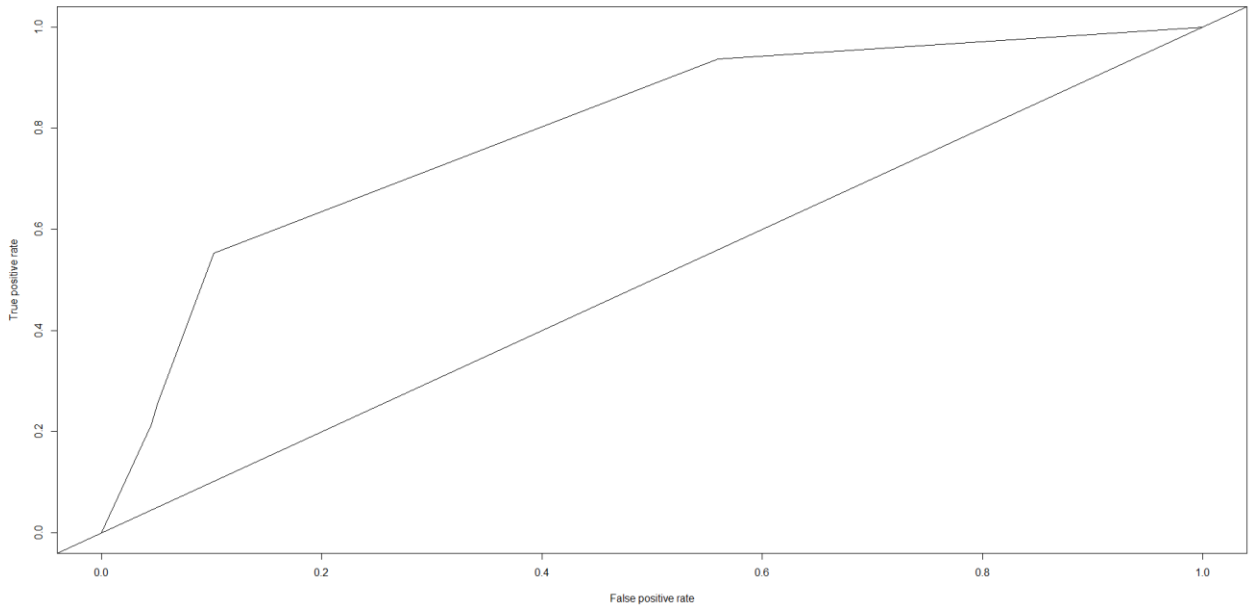
In conclusion, the issue with AUC also occurs due to lack of data we have. We have enough data to fit the model but still, it's not good enough to generate the generalized model to be computed on which means we do not have enough number of observations resulting not enough variety into data.. So, having more data would help neural network improve the Area Under Curve value when predicting whether it will RainTomorrow.

Appendix:

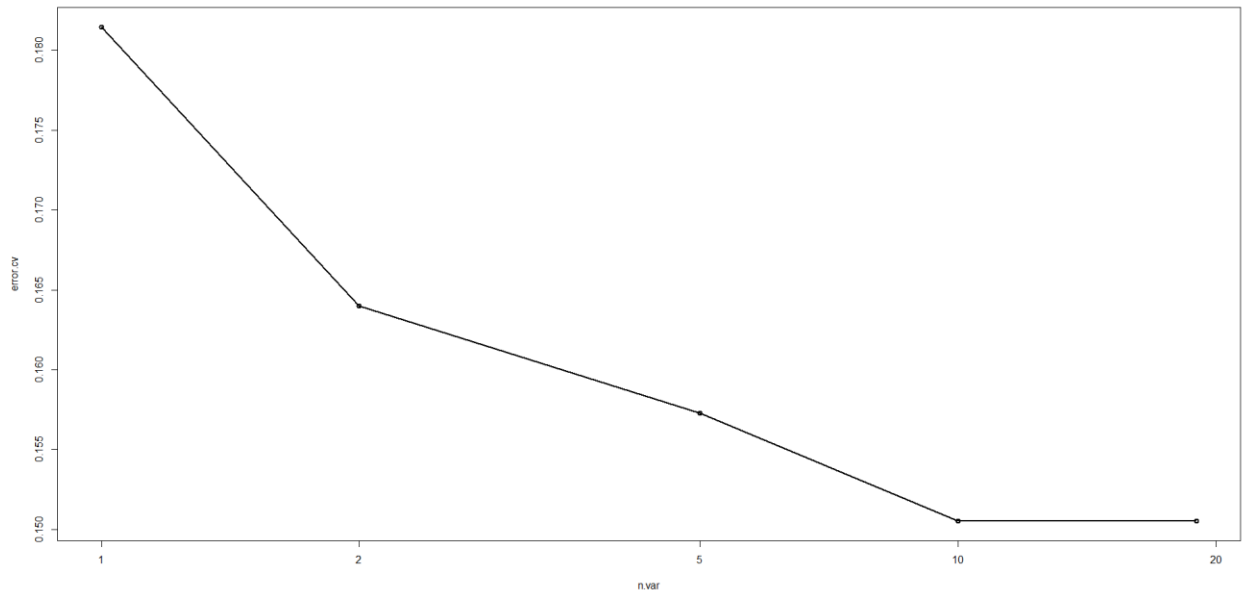
1.1: Improved Decision Tree



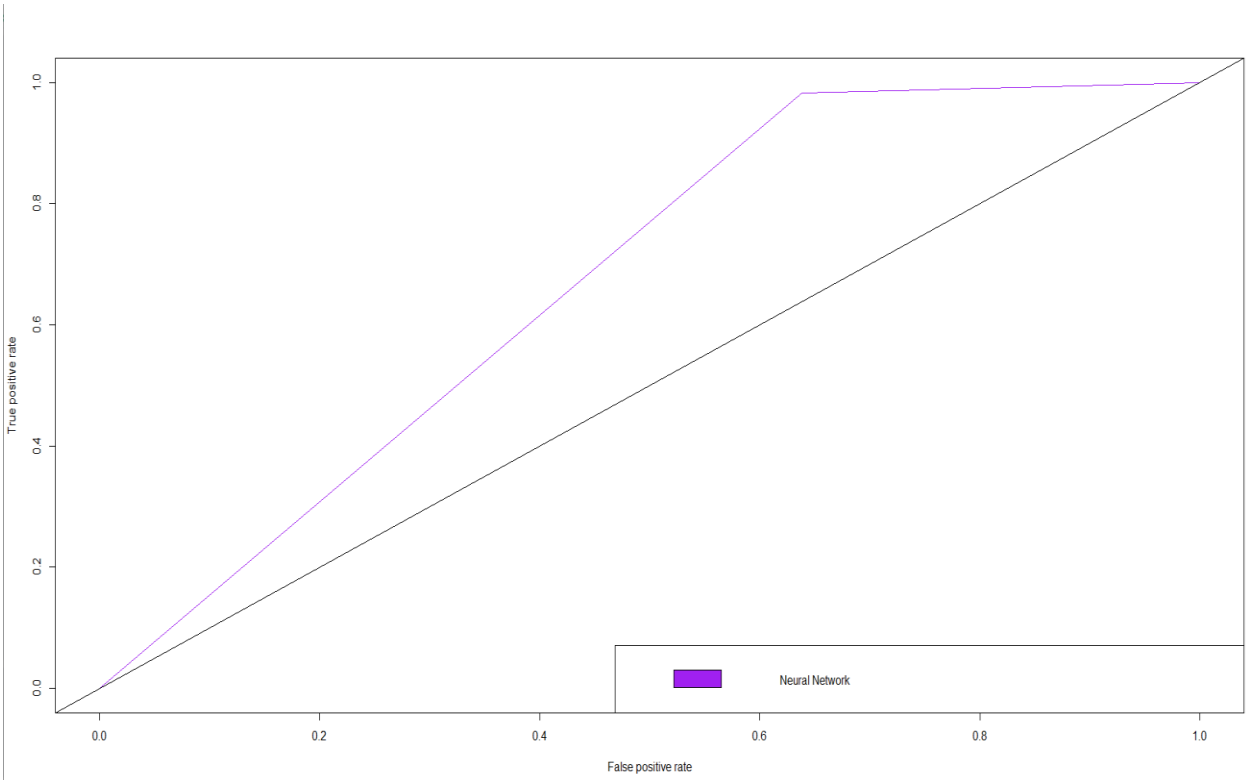
1.2: Improved Decision Tree ROC



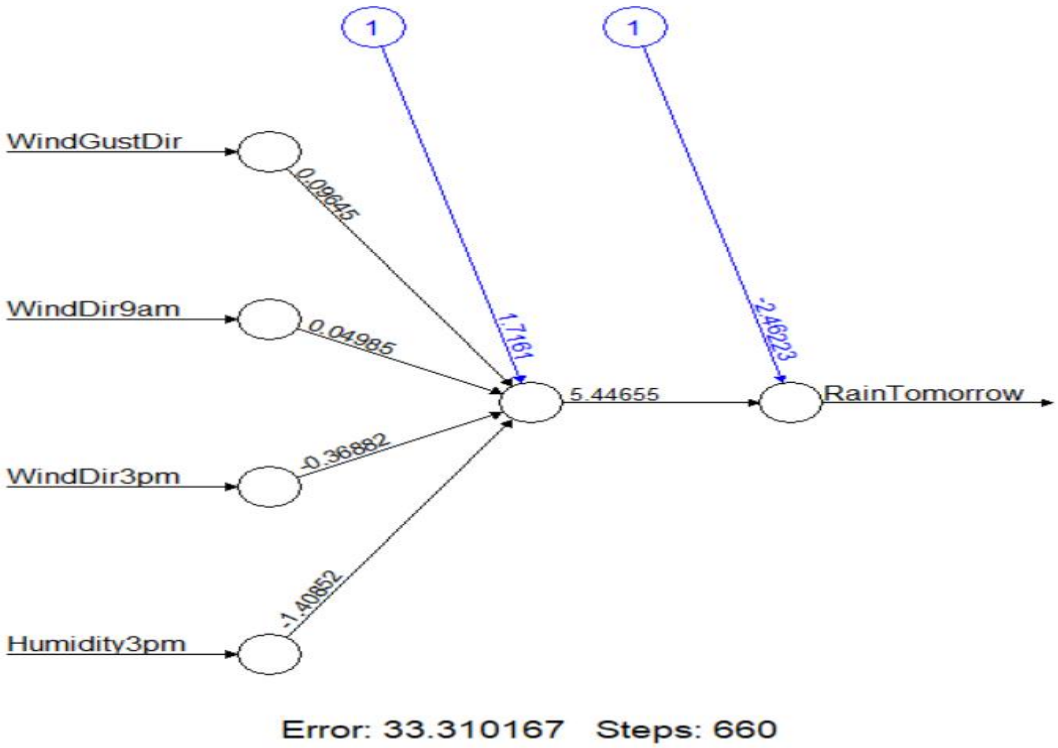
1.3:



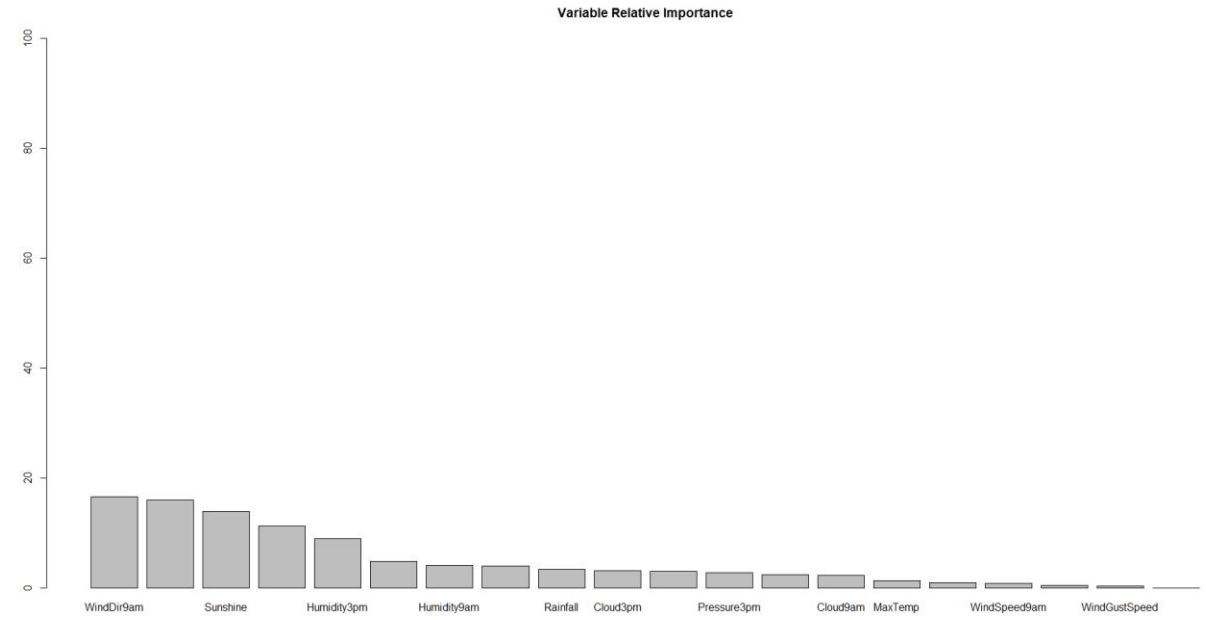
1.4: Neural Network ROC



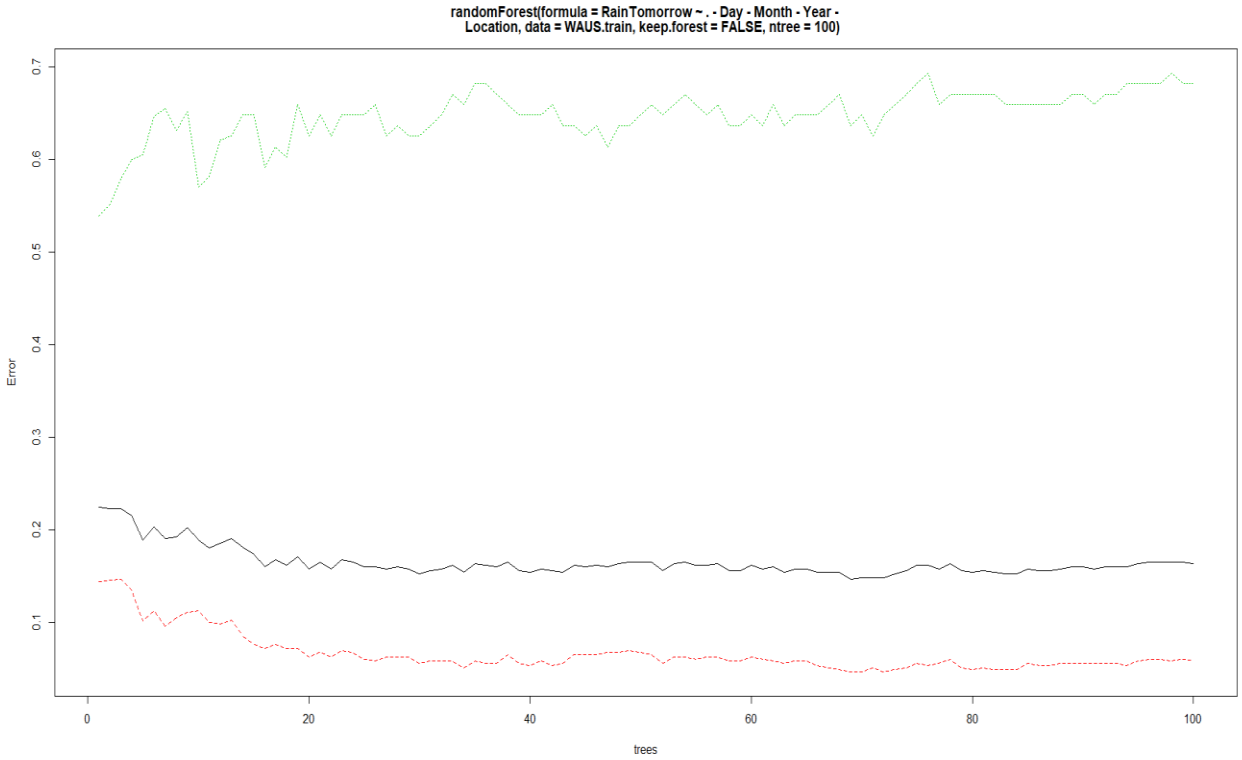
1.5: Neural network diagram:



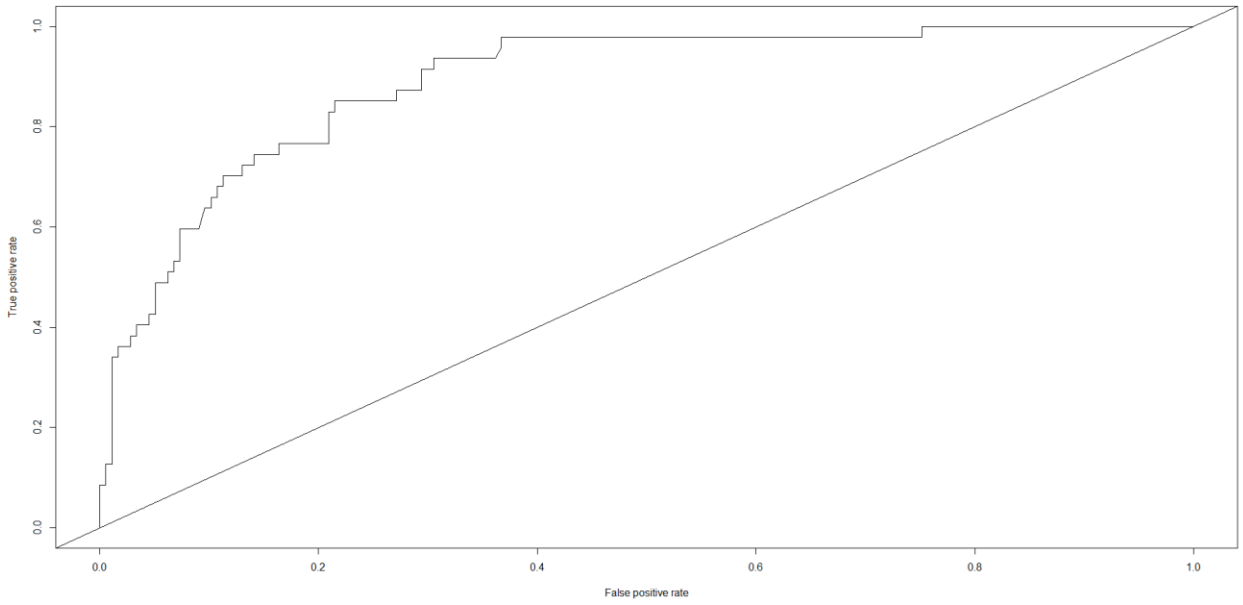
1.6: Boosting information gain for each variable:



1.7: Random Forest error trend for number of trees:



1.8: Improved Random Forest ROC curve



Assumptions

-We are assuming that the day is rainy day if the rain in the given day is more than 1 mm, but it will not be rainy day otherwise.

-We will remove all the NA values from the dataset in the cleaning process.

-Day, Month, Year, Location have not been included in any models.

-Assuming Rainfall and Raintoday can lead to different results from summary table so they are both included

-While choosing for most important variables, the variables having contribution of more than 1 in any model is considered important and less than 1 as very little effect. Top 5 are the most important variables.

-In neural network, the variable Sunshine is removed from the most important predictor in the model since we are assuming that Sunshine would not contribute to the neural because it has relatively lower contribution in all models which might lead to loss of the accuracy values.

-We are taking 80-20 datasets for neural network to make sure we have larger and generalized ANN to fit in diverse values.

-Assuming logistic regression model and rpart will help increasing the accuracy into the ensemble method of random forest

References

1. https://codeburst.io/2-important-statistics-terms-you-need-to-know-in-data-science-skewness-and-kurtosis-388fef94eeaa

2. https://pubmed.ncbi.nlm.nih.gov/16022695/