Precipitation Prediction in Australia

Elizabeth Frank

IST 687 Intro to Data Science | Syracuse University Summer 2024

Guide to the Report

Project Overview and Objectives

Planning for the day involves many factors, and one of the most significant is the weather, specifically, whether it will rain. Rain can impact a variety of daily decisions, and while rainy days are unavoidable, the ability to more accurately predict if it will rain tomorrow can make a significant difference. The question becomes: will it rain, and do you need an umbrella?

The main source of weather planning is the local weather stations by TV, radio, or print, with a general accuracy of 80-90% for short term forecasts. This raises the question, is their accuracy based more on knowledge and understanding of the weather patterns or historical data and modeling. Could individuals, without formal meteorological training, be nearly as effective by relying on historical data alone? Let's explore the historical weather data from the Australian Government Bureau of Meteorology utilizing the 49 weather stations across the country, with the aim to understand Australia's weather patterns and develop a precipitation prediction model that answers the question: will it rain tomorrow, yes, or no?

This project follows the traditional approach to building a predictive model for rainfall in Australia. It begins with data preprocessing and cleaning, preparing the dataset for extensive exploratory data analysis and visualization, where insights, patterns, and trends are identified. Next, a variety of models are developed, considering the strengths and weaknesses of the data. The models are then evaluated and compared, with the goal of identifying the most accurate and reliable models for predicting rainy days.

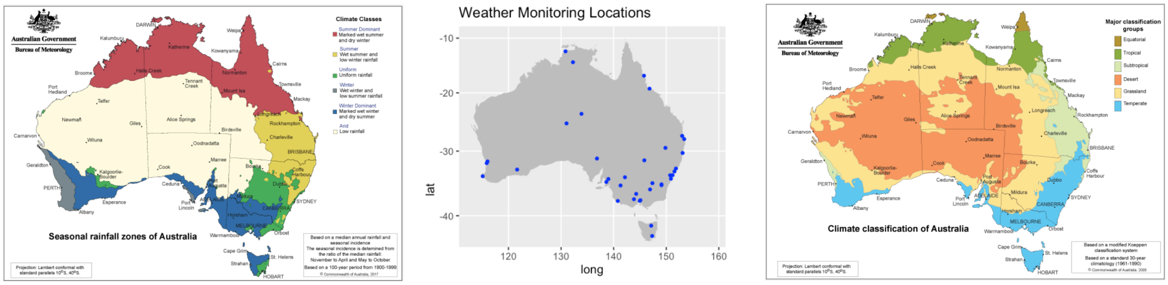
Prediction Goals

- Identify key predictors by building models to not only understand which factors have the most significant impact on whether it will rain tomorrow but also to accurately predict this outcome.
- Assess the predictive power of the models that use historical weather data and consider how much of current local weather forecasts are likely driven by historical patterns versus real-time meteorological insights.

Data Preprocessing and Preparation

The dataset used for this project comes from the Australian Government Bureau of Meteorology, comprised of weather data from 49 stations across the country sourced from Kaggle. Ranging from 2007-2017 with 145,460 rows and 25 attributes, including date, location, minimum and maximum temperature, rainfall, evaporation, sunshine, wind gust direction, wind gust speed, did it rain today, and morning and afternoon wind direction, wind speed, humidity, pressure, cloud coverage and temperature. The dependent variable is rain tomorrow with 77.6% No and 22.4% Yes values.

After inspecting the data and studying the rain zones and climate classifications of the weather stations it became apparent that this information was relevant to the project and verifiable by the Bureau of Meteorology and was added to the list of attributes.



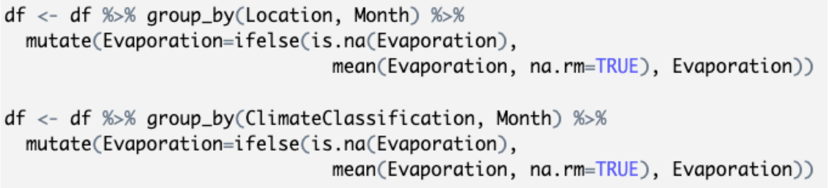
Australian Government
Bureau of Meteorology
Climate Classes
Summer Dominant
🟥 Marked wet summer and dry winter
Summer
🟨 Wet summer and low winter rainfall
Uniform
🟩 Uniform rainfall
Winter
⬛ Wet winter and low summer rainfall
Winter Dominant
🟦 Marked wet winter and dry summer
Arid
⬜ Low rainfall
DARWIN
Weipa
Kalumburu
Katherine
Kowanyama
Broome
Halls Creek
Normanton
Cairns
Port Hedland
Tennant Creek
Mount Isa
Townsville
Mackay
Telfer
Newman
Giles
Alice Springs
Longreach
Rockhampton
Carnarvon
Birdsville
Charleville
Wiluna
Oodnadatta
BRISBANE
Marree
Geraldton
Cook
Bourke
Kalgoorlie-Boulder
Coffs Harbour
Ceduna
Port Augusta
Dubbo
PERTH
Mildura
Esperance
Port Lincoln
ADELAIDE
SYDNEY
Albany
CANBERRA
Horsham
MELBOURNE
Seasonal rainfall zones of Australia
Warrnambool
Orbost
Cape Grim
St. Helens
Strahan
HOBART
Projection: Lambert conformal with standard parallels 10°S, 40°S.
Based on a median annual rainfall and seasonal incidence
The seasonal incidence is determined from the ratio of the median rainfall: November to April and May to October
Based on a 100-year period from 1900-1999
© Commonwealth of Australia, 2017
Weather Monitoring Locations
lat
-10
-20
-30
-40
long
120
130
140
150
160
Major classification groups
Equatorial
Tropical
Subtropical
Desert
Grassland
Temperate
Climate classification of Australia
Based on a modified Koeppen classification system
Based on a standard 30-year climatology (1961-1990)
© Commonwealth of Australia, 2005

Figure 1(a): Seasonal rainfall zones of Australia, (b): Weather monitoring locations, (c): Climate classification of Australia

With the unique rain and climate bands on the country, the non-uniform dispersion of the monitoring locations, and for increased interpretability, the rain zones were adopted as the primary location in the modeling. Instead of the 49 locations, the data was classified into 6 rain zones; arid, summer dominant, summer, uniform, winter dominant, and winter. This increased interpretability allowing individuals to recognize their rain zone and the impact that has.

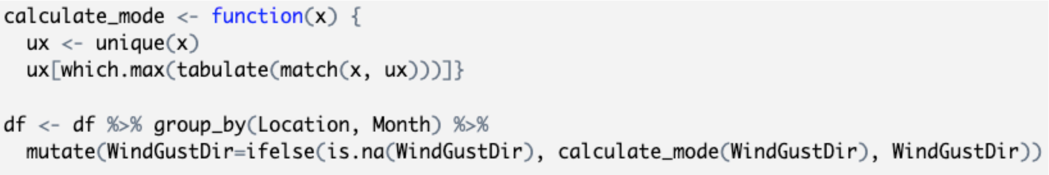
Missing Values

Ranging from 0 to 69,835 NA values in each column, comprising a total of 9.4% of the data, finding an effective and logical method to impute values was essential. If the column was numerical, the mean of the location and month for the variable was utilized, when that did not remove all NA values, the climate classification or rain zone and month was used to clear the remaining missing values. An example:



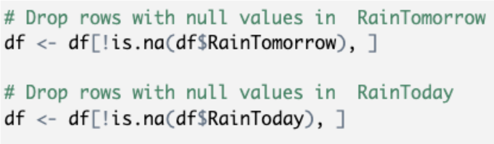
```
df <- df %>% group_by(Location, Month) %>%
  mutate(Evaporation=ifelse(is.na(Evaporation),
                            mean(Evaporation, na.rm=TRUE), Evaporation))

df <- df %>% group_by(ClimateClassification, Month) %>%
  mutate(Evaporation=ifelse(is.na(Evaporation),
                            mean(Evaporation, na.rm=TRUE), Evaporation))
```

Or if the attribute was categorical the mode was used similarly. An example:



```
calculate_mode <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]}

df <- df %>% group_by(Location, Month) %>%
  mutate(WindGustDir=ifelse(is.na(WindGustDir), calculate_mode(WindGustDir), WindGustDir))
```

With the consideration that rain today is a strong predictor of rain tomorrow and the low percentage of missing data at 0.9% the rows missing values were dropped, as well as any missing values in the dependent variable rain tomorrow.



```
# Drop rows with null values in  RainTomorrow
df <- df[!is.na(df$RainTomorrow), ]

# Drop rows with null values in  RainToday
df <- df[!is.na(df$RainToday), ]
```

And while not exactly missing data, during exploratory data analysis it became apparent that not all weather stations came online recording at the same time. There was a wave that came on in 2007 and then it took until the beginning of 2009 for all stations to be recording daily. In order to not unintentionally create a bias in specific rain zones, the dataset will be reduced to 2009 to 2017.

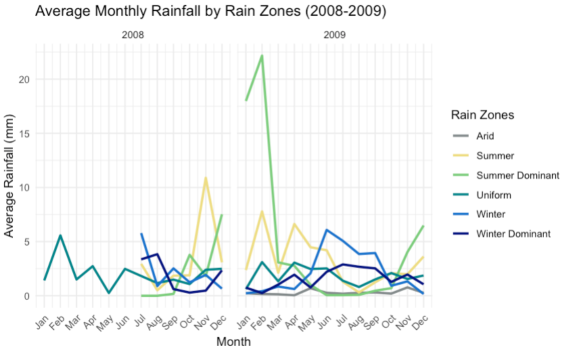


Figure 2: Average monthly rainfall by rain zones

Outliers

The size of the dataset and the quality of the recording, low human error, really lead to minimal variables with extreme outliers. The two variables that needed to be addressed was Rainfall and Evaporation.

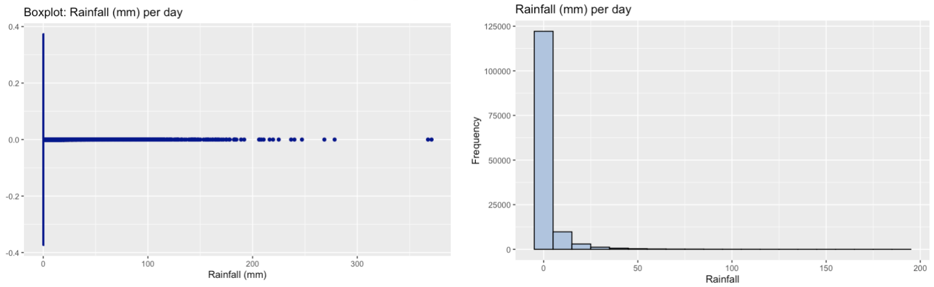


Figure 3(a): Boxplot of rainfall (mm) per day, (b): Histogram of rainfall (mm) per day

Rainfall had 14 values above 200, upon closer inspection these we true rain events that occurred in already high rainfall areas. In order to preserve the accuracy of the model, considering it does have areas of the country that experience seasonally high rainfall times, those values were removed to create a more accurate distribution of rainfall values.

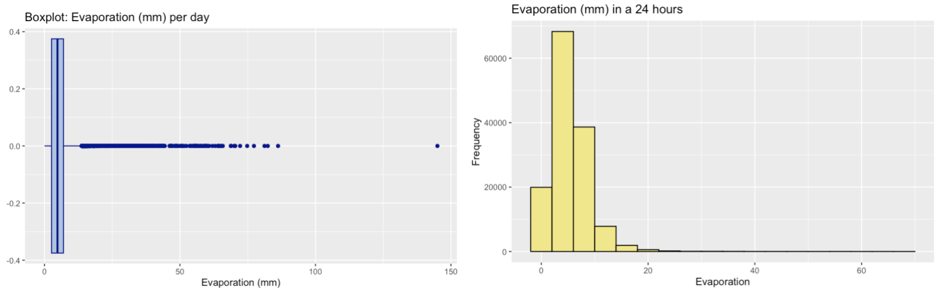


Figure 4(a): Boxplot of evaporation (mm) per day, (b): Histogram of evaporation (mm) per day

Evaporation had one value that was extremely high, and upon further examination was classified as an incorrect recording of data, specifically because the type of climate and rain pattern that day indicates that this was likely human error. In addition, there were 8 other values of extreme evaporation, upon further inspection these values did fall in the summer of the desert and are classified as extreme weather events therefore removed from the dataset, but again, due to the variability of environments including desert climates the data was not stripped of all higher evaporation values.

Feature Engineering

The main feature engineering that was required was addressing the variables with high collinearity. In addition to the repetitive nature of some of the variables, like having 9am and 3 pm readings, reducing the variables, but retaining the information allowed for increased interpretability. Windspeed, Humidity, Pressure, and Cloud morning and afternoon readings were combined for a daily average. Simplifying the data and reducing high attribute correlation.

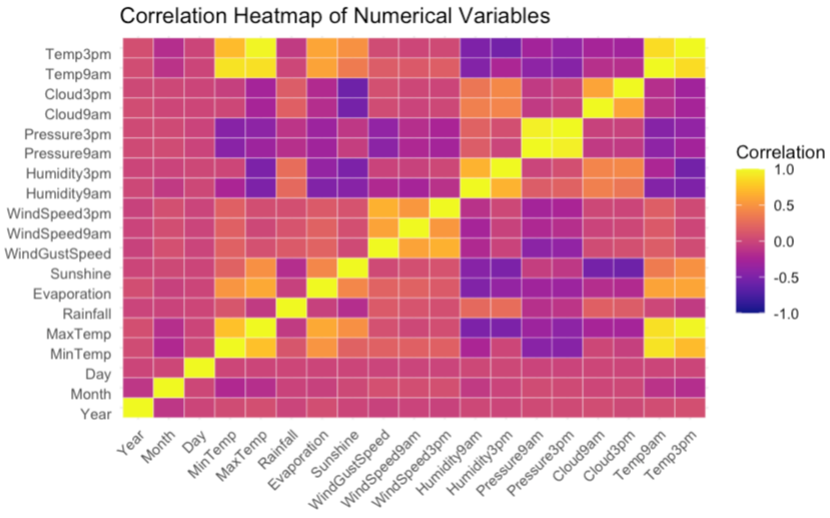


Figure 5: Correlation heatmap of numerical variables

Exploratory Data Analysis and Visualization

Understanding how environmental factors interact and influence the probability of rain tomorrow can increase the interpretability of the model and its effectiveness in use. Many of the factors are inherently obvious, which helps everyone be an effective weather predictor. Exploratory data analysis will examine how 8 years of data parallel those understandings and uncover new correlations that may not have been previously known.

Numerical and Categorical Variable Correlation

Using heatmaps for both numerical and categorical variables can show relationships through positive and negative correlation. Positive correlation indicates variables move in the same direction together while negative indicates they move in opposite directions. For calculation of the categorical variable correlation Cramér's V, which is derived from the Chi-square test of independence, was used for summarizing the measure of association. Similar to correlation, but the range is from 0 (no) to 1 (strong) association.

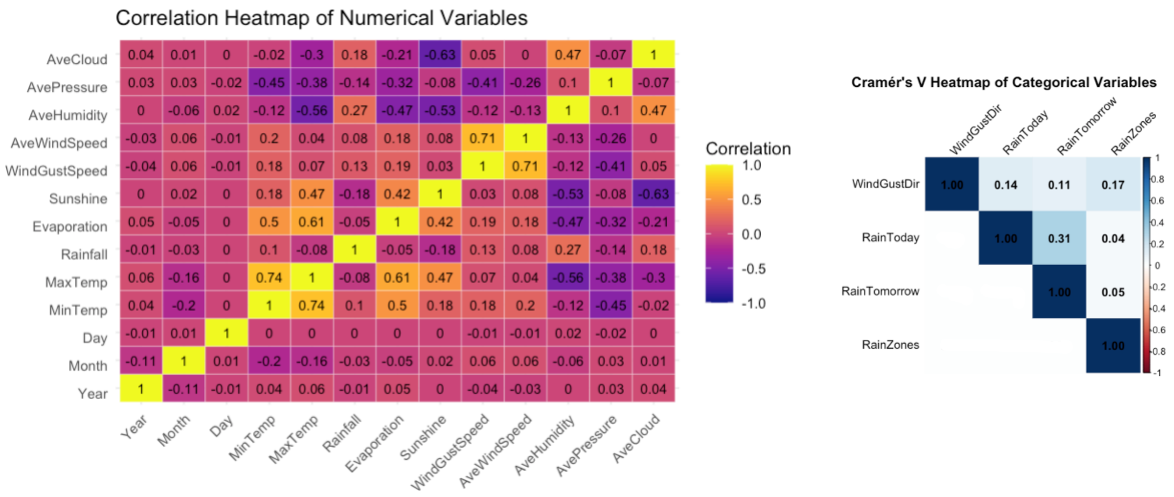


Figure 6(a): Correlation heatmap of numerical variables, (b): Cramér's V Heatmap of categorical variables

The relationships between the variables can be summarized as the following trends:

- *Numerical Variables*
  - Average cloud cover and average humidity are correlated at 0.47, increasing cloud cover relates to increased humidity. And logically, increased average cloud cover indicates reduced hours of sunshine with a -0.63 correlation.
  - Increased average daily pressure translates to a decrease in min and max temperature, evaporation, and both wind gust speed and average daily wind speed ranging from -0.26 to -0.48.
  - As average humidity increases max temp, evaporation and hours of sunshine decrease ranging from -.047 to -0.56. And as expected, as it increases so does the mm of rainfall that day.
  - Average wind speed for the day has a 0.71 correlation with high wind gust speed.
  - High wind gust speed for the day has a 0.19 positive relationship with increased daily evaporation.
  - With increased hours of daily sunshine there is a 0.42 to 0.47 positive correlation with evaporation and max temperature. And logically more hours of sunshine relates to less rainfall that day.
  - Increased daily evaporation is strongly correlated at 0.61 as max temperature of the day increases.
  - And of course, max and min temperature are correlated at 0.74.
- *Categorical Variables*
  - Wind gust direction has a weak association with rain zones, directional winds in the different rain zones of Australia have some relationship. The association between rain today and tomorrow with the wind direction is even weaker and is likely related to the previous association.
  - Rain today and rain tomorrow have a moderate association, indicating that the two events have a relationship.

Univariate Analysis

Analysis of individual variables provides insight in distribution, central tendency, and variability of data. Visualization of the frequency and range allows for better understanding of the data for preprocessing and feature engineering.

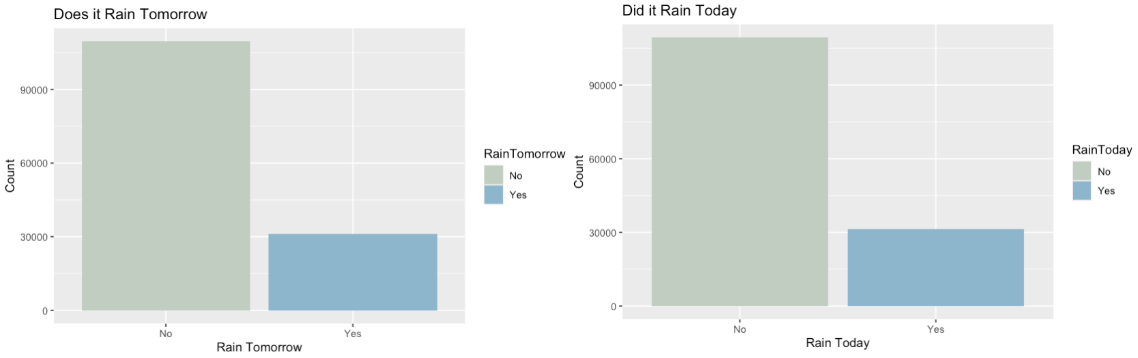


Figure 7(a): Bar chart of rain tommorrow, (b): Bar chart of rain today

Overall, during this 8-year period, the number of days without rain was more than three times greater than the number of days with rain. The imbalanced nature of this dataset indicates that the modeling might be more predisposed to predict no rain more often.

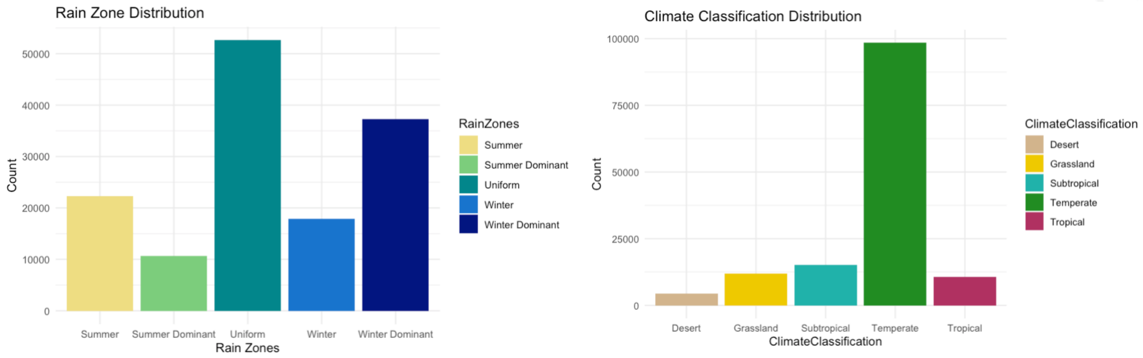


Figure 8(a): Histogram of rain zones, (b): Histogram of climate classifications

The most common types of rain zone and climate classification are both regions with more evenly distributed rainfall patterns. The extreme groups are generally less represented in the data. It's important to note that the weather monitoring stations are not uniformly spread throughout the country, therefore it cannot be interpreted as an indication of the overall prevalence of these zones and climates in Australia.

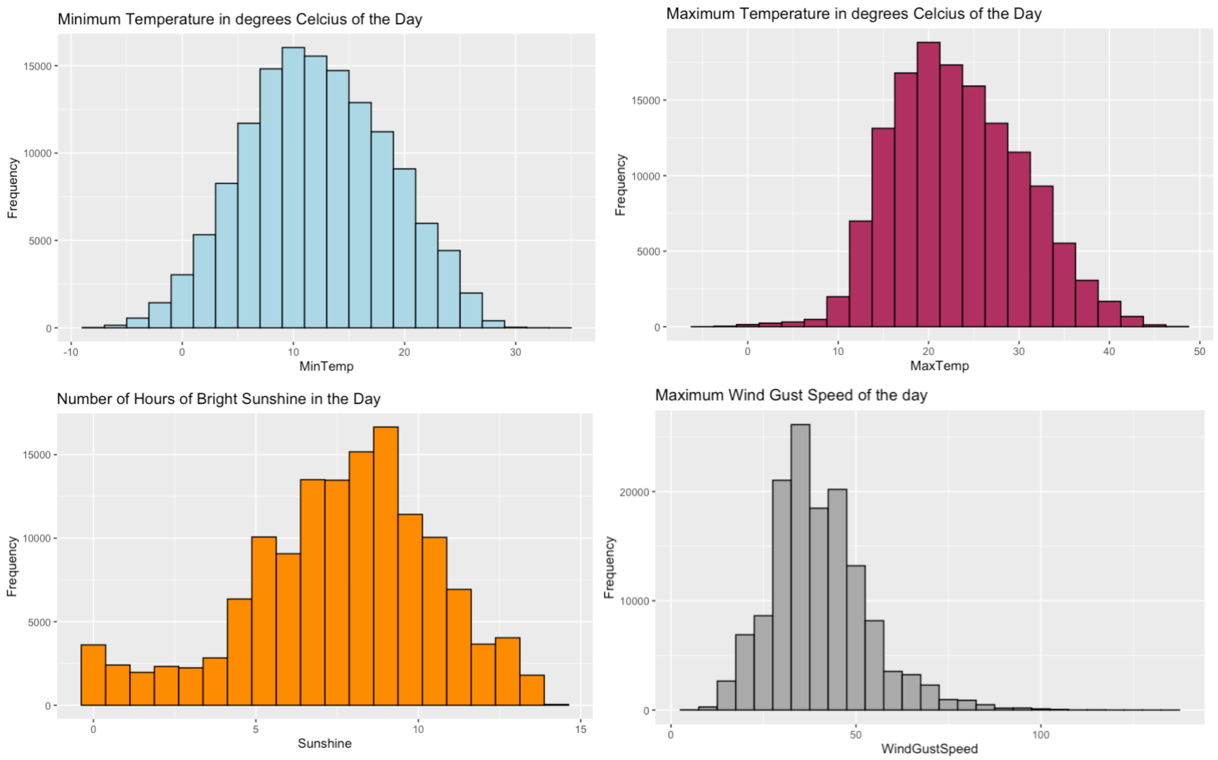


Figure 9(a): Histogram of minimum temperature, (b): Histogram of maximum temperature, (c): Histogram of Sunshine (d): Histogram of wind gust speed

The large size of the dataset has resulted in variables with more defined distributions that won't be as easily influenced by the variability in weather conditions.

- The min and max temp have normal distribution, and an average range of daily temperature is 10°C to 20°C.
- The number of hours of sunshine each day has more variability, some days there is no sunshine, but with a clear average of around 7 hours.
- Wind gust has a right skew to its distribution with most average gusts around 30 km/h but there are records of extreme gusts.

Multivariate Analysis

Exploration of two or more variables shows the correlations between variables and allows for better understanding of how the data interacts. Visualization of the trends provides insight that can lead to better model interpretability.

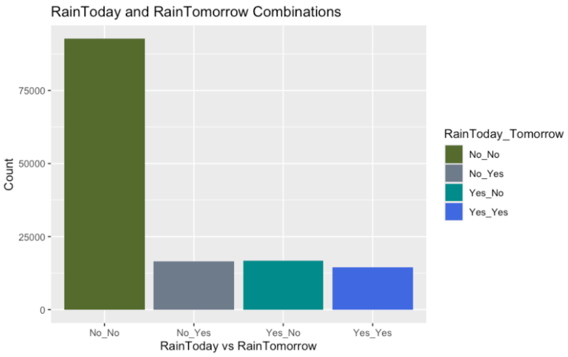


Figure 10: Bar chart of rain today and rain tomorrow combinations

The most frequent combination is two days without rain, and the least common is consecutive days of rain. And while the one day of rain out of two categories are only slightly larger than consecutive days, if combined they have a more substantial number, which indicates that two days of rain in a row is a relatively rare event for Australia.

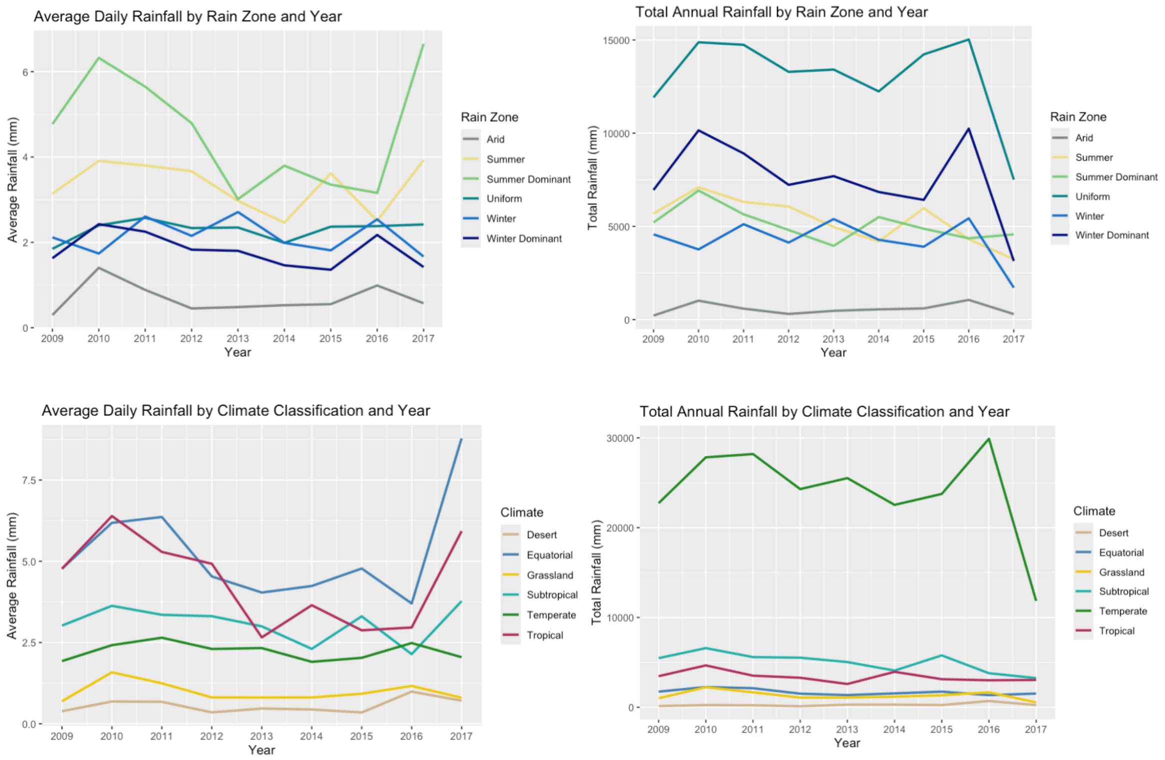


Figure 11(a): Line chart of average daily rainfall by rain zone and year, (b): Line chart of total annual rainfall by rain zone and year, (c): Line chart of average daily rainfall by climate classificationand year (d): Line chart of total annual rainfall by climate classification and year

For a better understanding of the overall trends in the rain zones and climate classifications the average daily rainfall and the cumulative amount of rain per year are paired.

- Summer dominant rain zone has by far the highest average daily rainfall per year, while has on of the lowest total rainfall per year. This indicates a severely dry winter and a very rainy summer.
- Uniform rain zone indicates that even though it has a lower daily average with the highest annual precipitation that even a low consistent daily rain can add up over a year.
- Winter dominant rain zone has the second highest amount of rain per year and nearly the lowest daily average, again indicating very polar seasons, winter is very wet, and summer is very dry.
- Equatorial, tropical, and subtropical are coastal Australia and these not surprisingly also have the highest daily rainfall averages.
- Climate classification associates with daily rainfall in a logical way with dessert and grassland with the lowest daily rainfall averages and tropical and equatorial with the highest.
- Total rainfall per year has temperate climate almost 5 times higher than the relatively similar other climate classifications for total annual rainfall per year.

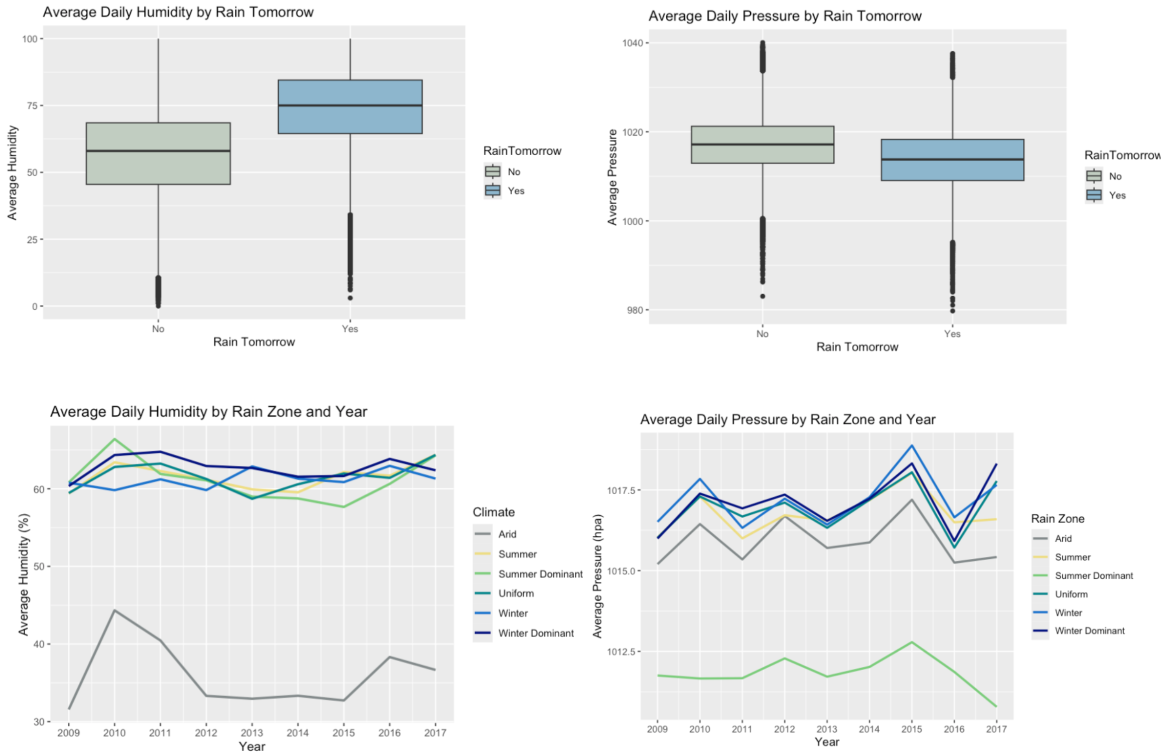


Figure 12(a): Boxplot of average daily humidity by rain tomorrow, (b): Boxplot of average daily pressure by rain tomorrow, (c): Line chart of average daily humidity by rain zone and year (d): Line chart average daily pressue by rain zone and year

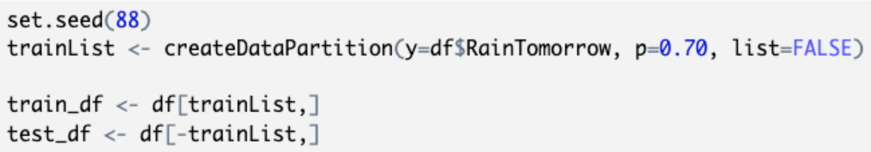
Humidity and pressures are two measurements that can easily be obtained from an inexpensive home monitoring device.

- Higher humidity days are associated with increased chances of rain tomorrow, with the understanding that there is variability in the humidity on the days before rain events.
- Average daily humidity is very similar in all rain zones apart from the arid zone.
- Lower pressure the day before it rains is a commonly known trend, and the data represents that also, although again the outliers in the data indicates that there is variability.
- Average daily pressure is similar throughout most rain zones, apart from the summer dominant zone which represents the northern third of the country, the more tropical area, and is likely associated with that.

Model Development and Evaluation

A range of models, including Support Vector Machines (SVM), decision trees (rpart), and Random Forests, were developed to explore different approaches to this classification task. Each model was trained to predict the target variable, RainTomorrow, and evaluated by comparing their accuracy, precision, recall, and F1 scores.

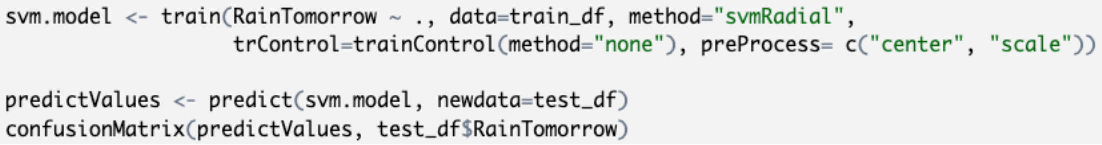
The dataset was split into a 70% training set and a 30% test set to allow for an unbiased evaluation of each model's performance on the unseen data.



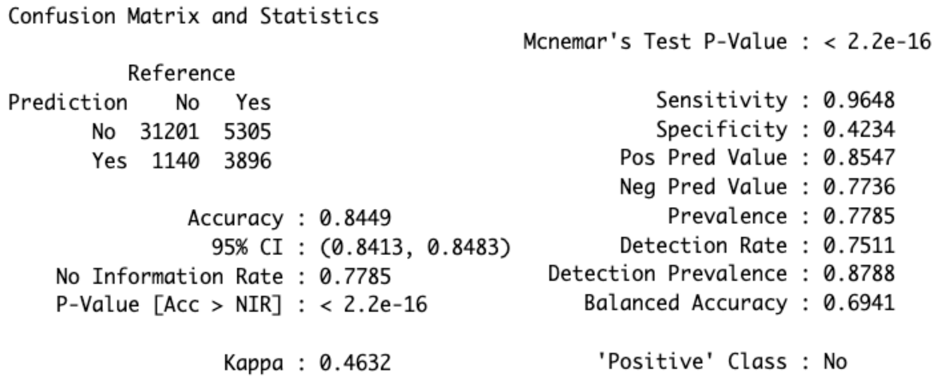
```r
set.seed(88)
trainList <- createDataPartition(y=df$RainTomorrow, p=0.70, list=FALSE)

train_df <- df[trainList,]
test_df <- df[-trainList,]
```

Support Vector Machine (svm)

SVM is a supervised learning algorithm primarily used for classification, and very effective for high dimensional data and can handle both linear and non-linear relationships. With the nature of weather's complex relationships, utilizing the radial function for its increased ability to handle nonlinear separation of data makes this model is a great option.



```r
svm.model <- train(RainTomorrow ~ ., data=train_df, method="svmRadial",
                   trControl=trainControl(method="none"), preProcess= c("center", "scale"))

predictValues <- predict(svm.model, newdata=test_df)
confusionMatrix(predictValues, test_df$RainTomorrow)
```

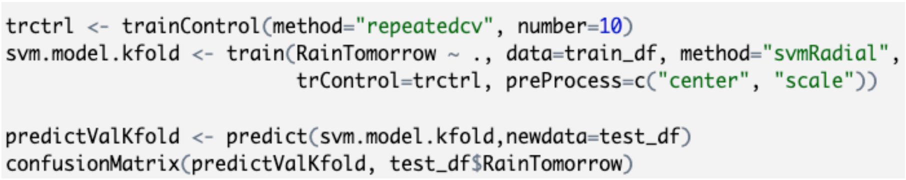


```
Confusion Matrix and Statistics
                                           Mcnemar's Test P-Value : < 2.2e-16
          Reference
Prediction    No   Yes                                Sensitivity : 0.9648
       No  31201  5305                                Specificity : 0.4234
       Yes  1140  3896                             Pos Pred Value : 0.8547
                                                   Neg Pred Value : 0.7736
               Accuracy : 0.8449                       Prevalence : 0.7785
                 95% CI : (0.8413, 0.8483)         Detection Rate : 0.7511
    No Information Rate : 0.7785             Detection Prevalence : 0.8788
    P-Value [Acc > NIR] : < 2.2e-16             Balanced Accuracy : 0.6941

                  Kappa : 0.4632                 'Positive' Class : No
```

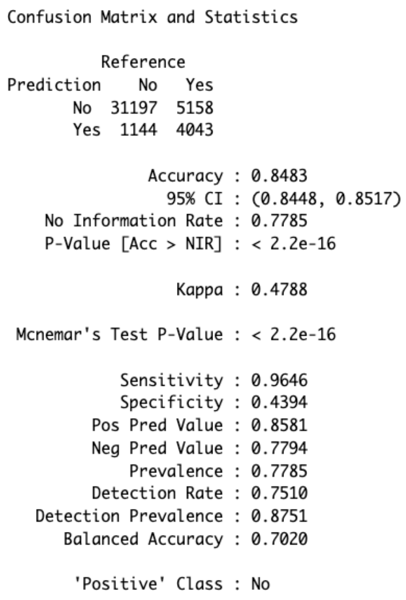
The model’ s accuracy indicates that it correctly predicts whether it will rain tomorrow in approximately 84.5% of cases. The sensitivity (recall) is 96.5%, showing that the model is very effective at predicting days with no rain, although with a specificity of 42.3%, it suggests that the model is less capable of identifying rain tomorrow when it rains. The positive predictive value (precision) predicts when there is no rain tommorrow correct 85.5% of the time.

Support Vector Machine with K-fold Cross Validation (kfold)

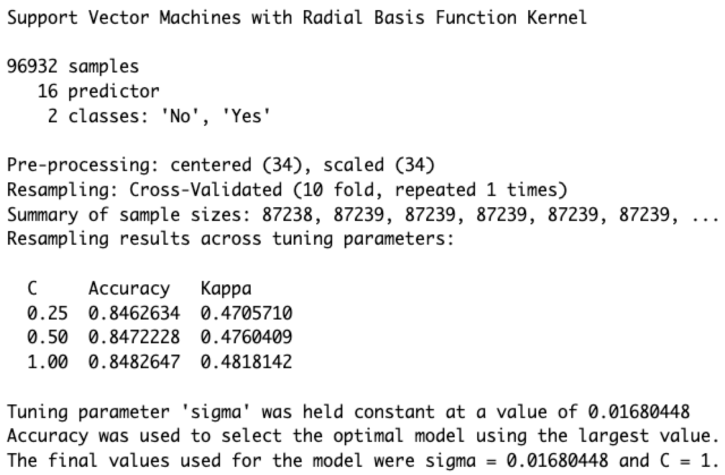
K-fold cross validation is utilized to decreased overfitting, by tuning hyperparameters, by splitting the data into training subsets and testing on the remainder, repeatedly, in this case 10 times, the results are then averaged for the best model.



```
trctrl <- trainControl(method="repeatedcv", number=10)
svm.model.kfold <- train(RainTomorrow ~ ., data=train_df, method="svmRadial",
                        trControl=trctrl, preProcess=c("center", "scale"))

predictValKfold <- predict(svm.model.kfold,newdata=test_df)
confusionMatrix(predictValKfold, test_df$RainTomorrow)
```



```
Confusion Matrix and Statistics

          Reference
Prediction    No   Yes
       No  31197  5158
       Yes  1144  4043

               Accuracy : 0.8483
                 95% CI : (0.8448, 0.8517)
    No Information Rate : 0.7785
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.4788

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9646
            Specificity : 0.4394
         Pos Pred Value : 0.8581
         Neg Pred Value : 0.7794
             Prevalence : 0.7785
         Detection Rate : 0.7510
   Detection Prevalence : 0.8751
      Balanced Accuracy : 0.7020

       'Positive' Class : No
```



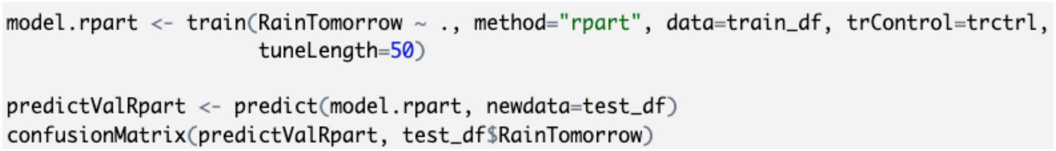
```
Support Vector Machines with Radial Basis Function Kernel

96932 samples
   16 predictor
    2 classes: 'No', 'Yes'

Pre-processing: centered (34), scaled (34)
Resampling: Cross-Validated (10 fold, repeated 1 times)
Summary of sample sizes: 87238, 87239, 87239, 87239, 87239, 87239, ...
Resampling results across tuning parameters:

  C     Accuracy   Kappa
  0.25  0.8462634  0.4705710
  0.50  0.8472228  0.4760409
  1.00  0.8482647  0.4818142

Tuning parameter 'sigma' was held constant at a value of 0.01680448
Accuracy was used to select the optimal model using the largest value.
The final values used for the model were sigma = 0.01680448 and C = 1.
```

The model's accuracy indicates that it correctly predicts whether it will rain tomorrow in approximately 84.8% of cases. The sensitivity (recall) is 96.5%, showing that the model is highly effective at predicting days with no rain. However, with a specificity of 43.9%, it suggests that the model struggles more to identify rainy days when it actually rains. The positive predictive value (precision) predicts when there is no rain tomorrow correctly 85.4% of the time.

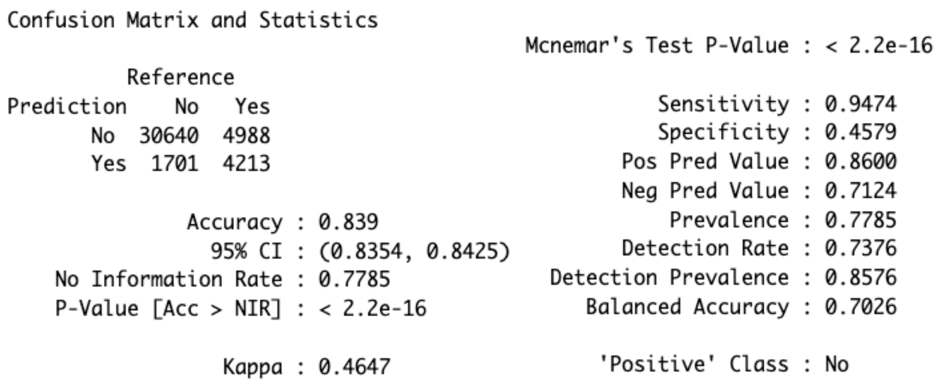
The final model was tuned using C = 1.0 and sigma = 0.0168, which were chosen to maximize accuracy and achieve the best balance between maximizing margin and minimizing misclassification.

Recursive Partitioning and Regression Trees (rpart)

Rpart decision trees are built using the data split into subsets based on the most significant predictors at each node. While decision trees are easy to visualize and interpret, they do not handle non-linear data well without scaling and transformations with reduces interpretability.

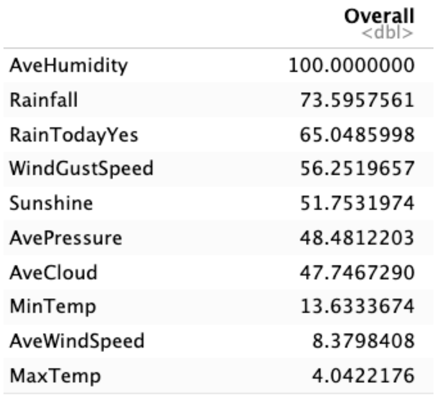


```
model.rpart <- train(RainTomorrow ~ ., method="rpart", data=train_df, trControl=trctrl,
                     tuneLength=50)

predictValRpart <- predict(model.rpart, newdata=test_df)
confusionMatrix(predictValRpart, test_df$RainTomorrow)
```



```
Confusion Matrix and Statistics
                                              Mcnemar's Test P-Value : < 2.2e-16
          Reference
Prediction    No   Yes                                   Sensitivity : 0.9474
       No  30640  4988                                   Specificity : 0.4579
       Yes  1701  4213                                Pos Pred Value : 0.8600
                                                      Neg Pred Value : 0.7124
               Accuracy : 0.839                           Prevalence : 0.7785
                 95% CI : (0.8354, 0.8425)            Detection Rate : 0.7376
    No Information Rate : 0.7785                Detection Prevalence : 0.8576
    P-Value [Acc > NIR] : < 2.2e-16                Balanced Accuracy : 0.7026

                  Kappa : 0.4647                    'Positive' Class : No
```

The model's accuracy indicates that it correctly predicts whether it will rain tomorrow in approximately 83.9% of cases. The sensitivity (recall) is 94.7%, showing that the model is very effective at predicting days with no rain. However, with a specificity of 45.8%, it suggests that the model is less capable of identifying rain tomorrow when it actually rains, a slight improvement from the svm model. The positive predictive value (precision) predicts when there is no rain tomorrow correctly 86.0% of the time.



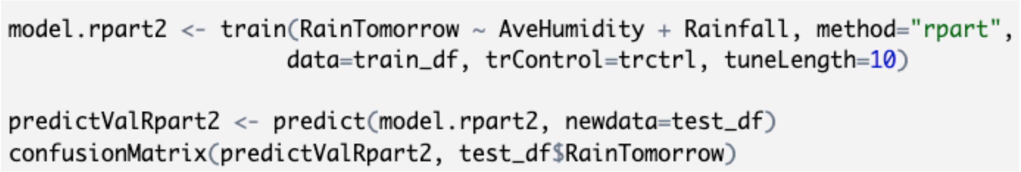
	Overall <dbl>
AveHumidity	100.0000000
Rainfall	73.5957561
RainTodayYes	65.0485998
WindGustSpeed	56.2519657
Sunshine	51.7531974
AvePressure	48.4812203
AveCloud	47.7467290
MinTemp	13.6333674
AveWindSpeed	8.3798408
MaxTemp	4.0422176

Figure 13: Rpart variable importance

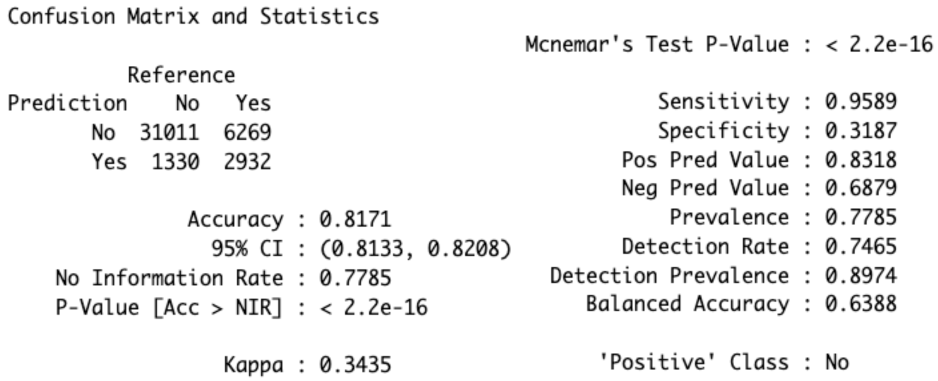
The variable importance for the rpart indicates that average humidity, rainfall, and did it rain today, are the top three variables in the dataset for decision tree splits.

Recursive Partitioning and Regression Trees Simplified (rpart2)

For a citizen-friendly tool that offers simplicity and practical assistance in predicting rain, a simplified rpart model was trained with both 2 and 3 variables. It was determined that using 2 variables provided a higher level of interpretability while still maintaining effectiveness.



```
model.rpart2 <- train(RainTomorrow ~ AveHumidity + Rainfall, method="rpart",
                      data=train_df, trControl=trctrl, tuneLength=10)

predictValRpart2 <- predict(model.rpart2, newdata=test_df)
confusionMatrix(predictValRpart2, test_df$RainTomorrow)
```



```
Confusion Matrix and Statistics
                                                   Mcnemar's Test P-Value : < 2.2e-16
          Reference
Prediction    No   Yes                                        Sensitivity : 0.9589
       No  31011  6269                                        Specificity : 0.3187
       Yes  1330  2932                                     Pos Pred Value : 0.8318
                                                           Neg Pred Value : 0.6879
               Accuracy : 0.8171                               Prevalence : 0.7785
                 95% CI : (0.8133, 0.8208)                 Detection Rate : 0.7465
    No Information Rate : 0.7785                     Detection Prevalence : 0.8974
    P-Value [Acc > NIR] : < 2.2e-16                     Balanced Accuracy : 0.6388

                  Kappa : 0.3435                         'Positive' Class : No
```

The model’s accuracy indicates that it correctly predicts whether it will rain tomorrow in approximately 81.7% of cases. The sensitivity (recall) is 95.9%, showing that the model is very effective at predicting days with no rain, although with a specificity of 31.9%, it suggests that the model is even less capable of identifying rain tomorrow when it rains. The positive predictive value (precision) predicts when there is no rain tomorrow correctly 83.2% of the time. The increased simplicity and interpretability do have a cost in performance but is still a viable model with a high value.

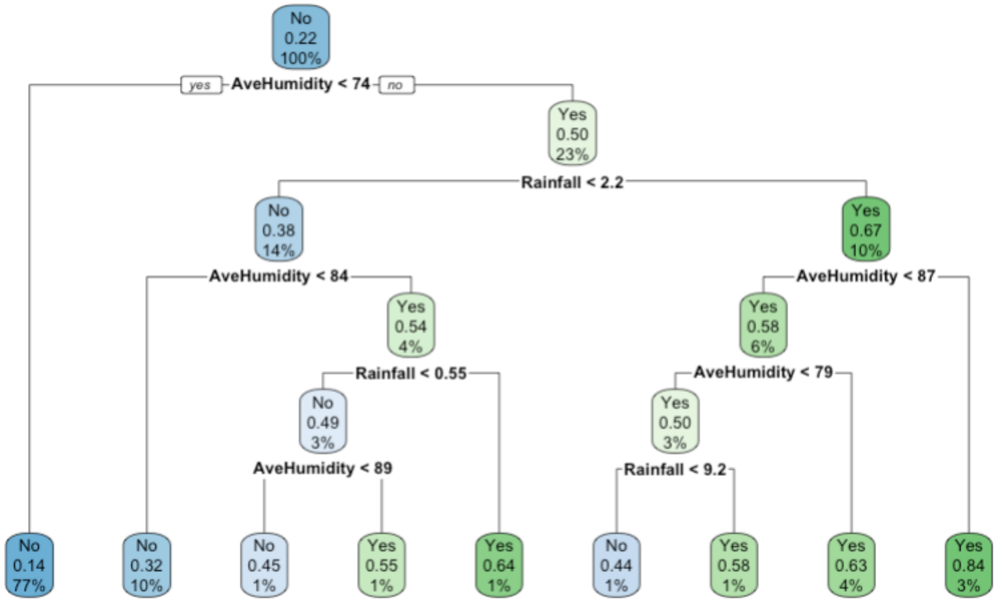


Figure 14: Rpart2 decision tree

Based on two significant variables, this user friendly tree provides clear and interpretable rules for predicting if it will rain tomorrow.

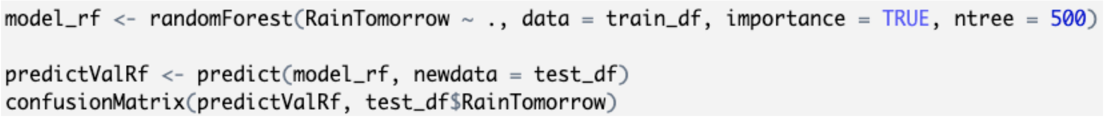
- If average humidity is < 74% the model predicts no rain with a probability of 22%
- If average humidity is > 74% the model moves on to rainfall.
  - If rainfall is < 2.2 mm in the last 24 hours, the model predicts no rain at 38%

    - The average humidity is > 84% and rainfall is < .55 mm and the model predicts rain tomorrow with 54% probability
  - If the rainfall is > 2.2 mm, the model predicts rain tomorrow with a probability of 50%
    - And the average humidity is > 87% the model predicts rain tomorrow at an 84% probability
  - If the rainfall is < 9.2 mm, and the average humidity is < 79% there is a 44% percent probability of no rain tomorrow.

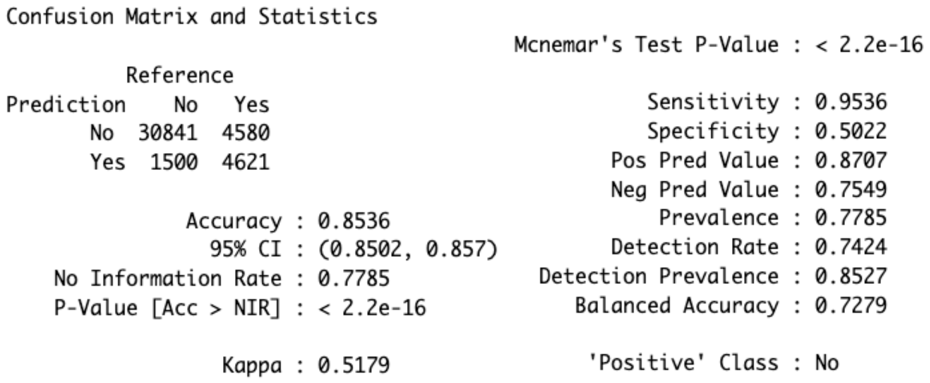
The general trend of the tree indicates that for most days when the average humidity is low it is more likely that there will be no rain tomorrow and cases where the average humidity is high and the presence of rain today there is a higher likelihood of rain tomorrow.

Random Forest (rf)

Random Forest is an ensemble model that builds multiple decision trees on subsets of the data and then averages the results to improve the predictive power in accuracy while controlling overfitting. By bagging, the random selection of a subset of data for each tree in the forest, with a random subset of feature considered at each split, the aggregated results are more stable and yield more accurate predictions.



```r
model_rf <- randomForest(RainTomorrow ~ ., data = train_df, importance = TRUE, ntree = 500)

predictValRf <- predict(model_rf, newdata = test_df)
confusionMatrix(predictValRf, test_df$RainTomorrow)
```



```
Confusion Matrix and Statistics
                                           Mcnemar's Test P-Value : < 2.2e-16
          Reference
Prediction    No   Yes                                Sensitivity : 0.9536
       No  30841  4580                                Specificity : 0.5022
       Yes   1500  4621                            Pos Pred Value : 0.8707
                                                   Neg Pred Value : 0.7549
               Accuracy : 0.8536                       Prevalence : 0.7785
                 95% CI : (0.8502, 0.857)          Detection Rate : 0.7424
    No Information Rate : 0.7785             Detection Prevalence : 0.8527
    P-Value [Acc > NIR] : < 2.2e-16             Balanced Accuracy : 0.7279

                  Kappa : 0.5179                 'Positive' Class : No
```

The model’s accuracy indicates that it correctly predicts whether it will rain tomorrow in approximately 85.4% of cases. The sensitivity (recall) is 95.4%, showing that the model is very effective at predicting days with no rain. With a specificity of 50.2%, it suggests that the model is better than previous models at identifying rain tomorrow when it rains. The positive predictive value (precision) predicts when there is no rain tomorrow correctly 87.1% of the time.

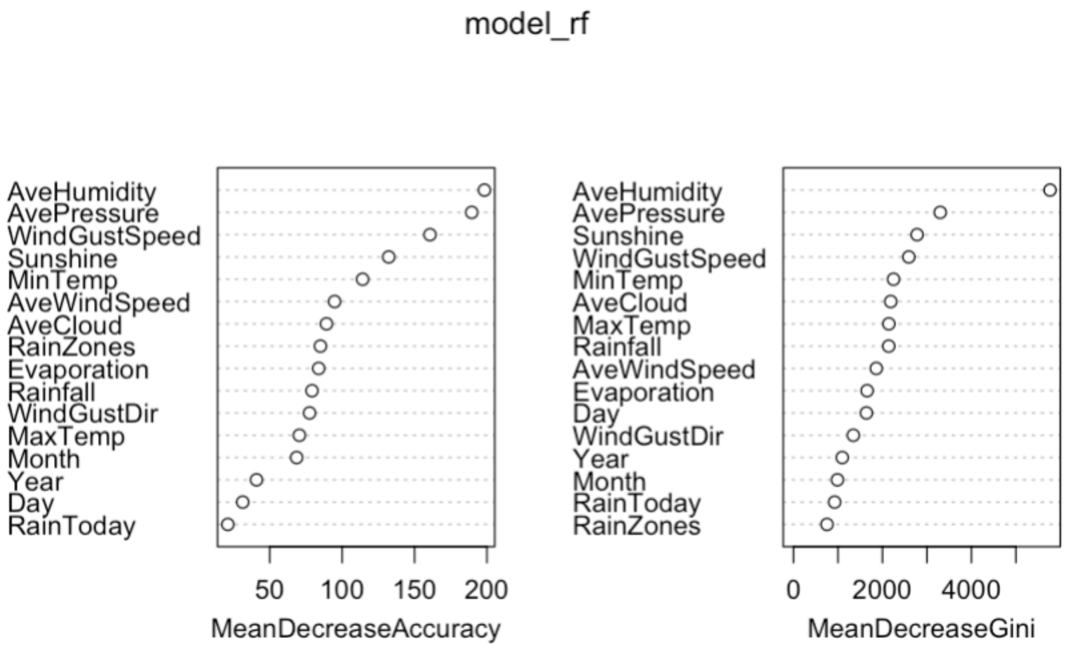


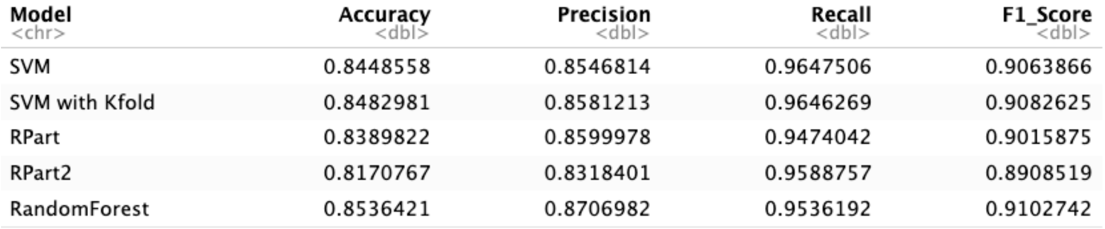
Figure 15: Random forest variable importance

Variable significance is measure in two ways by random forest, mean decrease in accuracy indicates how much the overall accuracy would decrease with the removal of a specific variable. And the mean decrease in Gini indicates how much the variable decreases the uncertainty at each split in the trees.

- Average humidity, average pressure, and high wind gust speed would have the greatest effect on accuracy if removed.
- Average humidity has the largest impact on reducing the Gini impurity.

Model Performance and Comparison

Evaluation and comparison in the performance of the machine learning models used to predict whether it will rain tomorrow was done by assessing the accuracy, precision, recall, and F1 score, which provides a balanced overview of the model's ability to make accurate predictions while handling the imbalanced data. The models are also compared to evaluate the balance between predictive power and interpretability, to determine the most effective approach for daily use in predicting rain tomorrow.



Model	Accuracy	Precision	Recall	F1_Score
<chr>	<dbl>	<dbl>	<dbl>	<dbl>
SVM	0.8448558	0.8546814	0.9647506	0.9063866
SVM with Kfold	0.8482981	0.8581213	0.9646269	0.9082625
RPart	0.8389822	0.8599978	0.9474042	0.9015875
RPart2	0.8170767	0.8318401	0.9588757	0.8908519
RandomForest	0.8536421	0.8706982	0.9536192	0.9102742

Accuracy is the proportion of correctly classified observations out of total observations. Meaning it predicts correct rain and no rain most of the time. Due to the imbalanced nature of the dataset, with far more no rain than rain days, only having a high accuracy doesn't indicate the best performing model.

- Random Forest has the highest accuracy at 85.4%, followed by SVM with Kfold at 84.8% and SVM at 84.5%.

Precision measures how many of the predicted rain days were actually correct. Meaning it minimizes false positives, predicting rain when it doesn't actually happen.

- Random Forest has the highest precision at 87.1%, followed closely by Rpart 86.0% and SVM with Kfold at 85.8%.

Recall is the measurement of how many actual rain events were correctly predicted by the model. Meaning that it minimizes false negatives, prioritizing catching all rain events at the cost of incorrectly indicating rain when it doesn't actually rain.

- SVM has the highest recall at 96.48%, but nearly same is SVM with Kfold at 96.46%.

F1-Score balances the precision and recall, best utilized as a performance metric with imbalanced data. Meaning it minimizes both false positives and negative and has higher predictive power predicting rain events.

- Random Forest again yields the highest F1-score at 91.0%, followed by SVM with Kfold at 90.8% and SVM at 90.7%.

Random Forest performs the best overall in terms of accuracy, precision, and F1-score. While SVM and SVM with Kfold have slightly higher recall, the difference is minimal. Random Forest's leading precision and F1-score suggest a more balanced performance, handling both false positives and false negatives effectively. Therefore, when interpretability isn't a factor, the Random Forest model has the best performance in predicting rain tomorrow. Given the results, local weather stations, could use this machine learning model to handle the dynamic nature of predicting weather with confidence.

Although Rpart2 is not the best-performing model based on accuracy or F1-score, its simplicity makes it the best choice for a real world application, with minor differences in performance, it is the most valuable in terms of practicality and usability. Having a small list of questions to answer to determine if it will likely rain allows for everyday use, and considering that the accuracy of local weather teams for short term predictions is 80-90% with this model at 81.7% it is comparable to that of which we rely on daily.

Interpretation and Key Insights

Considering the weather, particularly rainfall, is a task that influences daily activities like commuting, sports, or family outings. The goal of this project is to develop machine learning models that can accurately predict whether it will rain tomorrow using historical weather data. By leveraging advanced techniques such as Support Vector Machines (SVM), decision trees (rpart), and Random Forests, this project not only aims to understand local weather forecasting methods and their predictive accuracy using historical data but also seeks to find a balance between precision and interpretability for everyday use.

Through exploratory data analysis and modeling, key features for predicting rain have emerged. Rather than relying solely on a weather forecast made days ago, which might have used models like Random Forest to predict a 7-day outlook, individuals can make real-time decisions with simple tools.

By using a home hygrometer and barometer, you can create your own reliable forecast without watching the news. These devices can empower you to monitor conditions like humidity and pressure that influence weather patterns, allowing for you to make informed decisions. Start by asking yourself:

- What climate and rain zone do you live in? Does your area experience strong seasonal patterns?
- What is the average humidity, and is the day overcast?
- Has it rained and how much in the last 24 hours?
- What is the average pressure?

With the help of these simple tools, you can easily achieve at least an 82% accuracy in predicting if tomorrow will rain, allowing you to plan your day with confidence. This usability makes these tools not only practical but also easy to incorporate into daily life, without relying solely on external sources.

Future Work and Applications

While the models explored during this project yielded strong results, there are several avenues for improving their predictive capabilities. One of the key challenges encountered was the inherently imbalanced nature of rainfall data—statistically, there are simply more "no rain" days than "rain" days. This imbalance can skew the models' predictions, and it may have been compounded by the unequal distribution of weather stations across the country. Addressing this imbalance by using oversampling or under sampling techniques could improve model performance. Additionally, segmenting distinct rain zones into separate datasets might have increased the ability to predict rain events more accurately, as each region has its own unique weather patterns.

Another potential area for improvement lies in more extensive hyperparameter tuning. While basic tuning was conducted, exploring a broader range of hyperparameters might have yielded even better model performance. Furthermore, when recommending the use of personal devices such as hygrometers and barometers, integrating these devices with smartphone apps that provide real-time data could significantly enhance the accuracy of individuals rain predictions. This would not only make the predictions more accessible but also improve practicality for everyday use.

In the broader context of weather forecasting, we are witnessing what has been termed the "quiet revolution." Over the past 20 years, the advancements in weather prediction have been profound, with the quality of forecasts improving dramatically. Gone are the days of relying solely on almanacs and weathervanes—our future in weather prediction lies in advanced AI technologies. Forecasts that once took up to six hours, with updates only four times a day, can now be produced in under a minute.

So, what makes this new wave of forecasting different? The inclusion of historical weather data has played a transformative role. By combining this data with AI, specifically through machine and deep learning techniques, along with real time space-based radar data, predictions have become faster and more accurate than ever before. This shift is revolutionizing not only the meteorology industry but also the many industries that rely heavily on accurate weather forecasts.

The potential positive impacts on society and industries previously affected by inaccurate forecasts are immense. While it's difficult to fully quantify these benefits just yet, at the rapid pace of these advancements, the outcomes will soon become our reality.

Appendix

Libraries utilized during project:

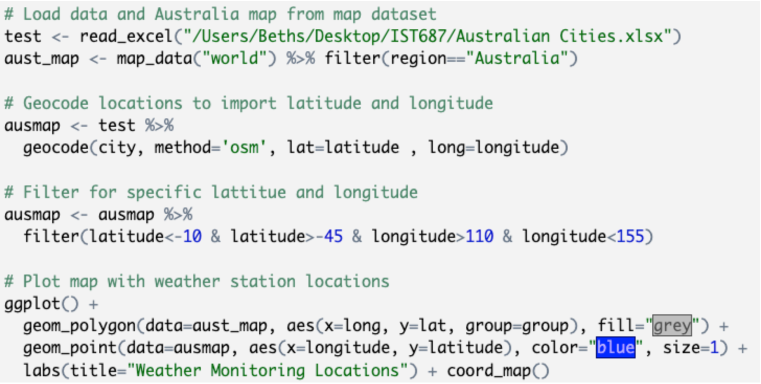
- library(readr)
- library(tidyr)
- library(dplyr)
- library(readxl)
- library(ggmap)
- library(ggplot2)
- library(tidygeocoder)
- library(reshape2)
- library(viridisLite)
- library(viridis)
- library(vcd)
- library(corrplot)
- library(caret)
- library(rpart)
- library(rpart.plot)
- library(randomForest)

R Code used for figures:

Figure 1

(a): Seasonal rainfall zones of Australia: http://www.bom.gov.au/climate/maps/averages/climate-classification/?maptype=seasgrpb

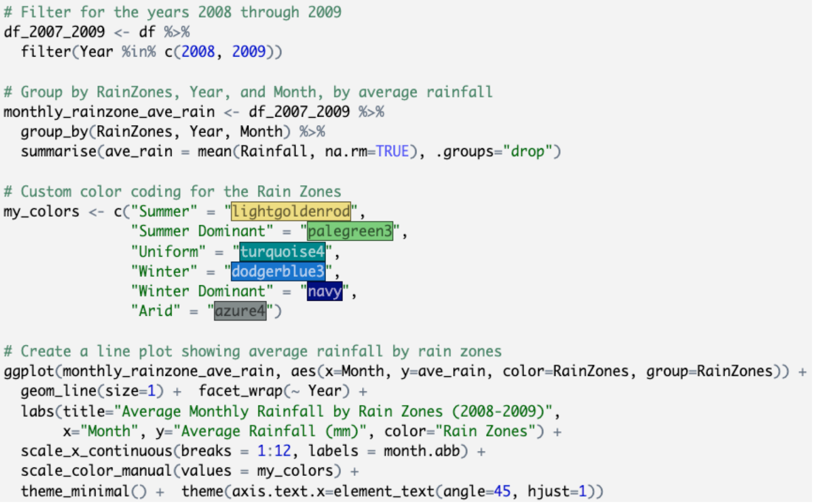
(b): Weather monitoring locations:



```
# Load data and Australia map from map dataset
test <- read_excel("/Users/Beths/Desktop/IST687/Australian Cities.xlsx")
aust_map <- map_data("world") %>% filter(region=="Australia")

# Geocode locations to import latitude and longitude
ausmap <- test %>%
  geocode(city, method='osm', lat=latitude , long=longitude)

# Filter for specific lattitue and longitude
ausmap <- ausmap %>%
  filter(latitude<-10 & latitude>-45 & longitude>110 & longitude<155)

# Plot map with weather station locations
ggplot() +
  geom_polygon(data=aust_map, aes(x=long, y=lat, group=group), fill="grey") +
  geom_point(data=ausmap, aes(x=longitude, y=latitude), color="blue", size=1) +
  labs(title="Weather Monitoring Locations") + coord_map()
```

(c): Climate classification of Australia: http://www.bom.gov.au/climate/maps/averages/climate-classification/

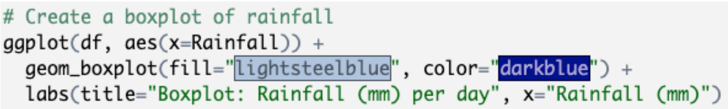
Figure 2: Average monthly rainfall by rain zones



```
# Filter for the years 2008 through 2009
df_2007_2009 <- df %>%
  filter(Year %in% c(2008, 2009))

# Group by RainZones, Year, and Month, by average rainfall
monthly_rainzone_ave_rain <- df_2007_2009 %>%
  group_by(RainZones, Year, Month) %>%
  summarise(ave_rain = mean(Rainfall, na.rm=TRUE), .groups="drop")

# Custom color coding for the Rain Zones
my_colors <- c("Summer" = "lightgoldenrod",
               "Summer Dominant" = "palegreen3",
               "Uniform" = "turquoise4",
               "Winter" = "dodgerblue3",
               "Winter Dominant" = "navy",
               "Arid" = "azure4")

# Create a line plot showing average rainfall by rain zones
ggplot(monthly_rainzone_ave_rain, aes(x=Month, y=ave_rain, color=RainZones, group=RainZones)) +
  geom_line(size=1) +  facet_wrap(~ Year) +
  labs(title="Average Monthly Rainfall by Rain Zones (2008-2009)",
       x="Month", y="Average Rainfall (mm)", color="Rain Zones") +
  scale_x_continuous(breaks = 1:12, labels = month.abb) +
  scale_color_manual(values = my_colors) +
  theme_minimal() +  theme(axis.text.x=element_text(angle=45, hjust=1))
```

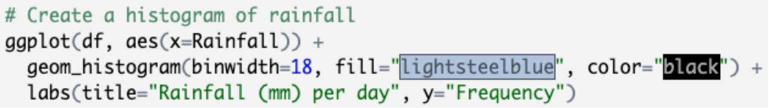
Figure 3(a): Boxplot of rainfall (mm) per day, (b): Histogram of rainfall (mm) per day
Figure 4(a): Boxplot of evaporation (mm) per day, (b): Histogram of evaporation (mm) per day

Example of (a):



```r
# Create a boxplot of rainfall
ggplot(df, aes(x=Rainfall)) +
  geom_boxplot(fill="lightsteelblue", color="darkblue") +
  labs(title="Boxplot: Rainfall (mm) per day", x="Rainfall (mm)")
```

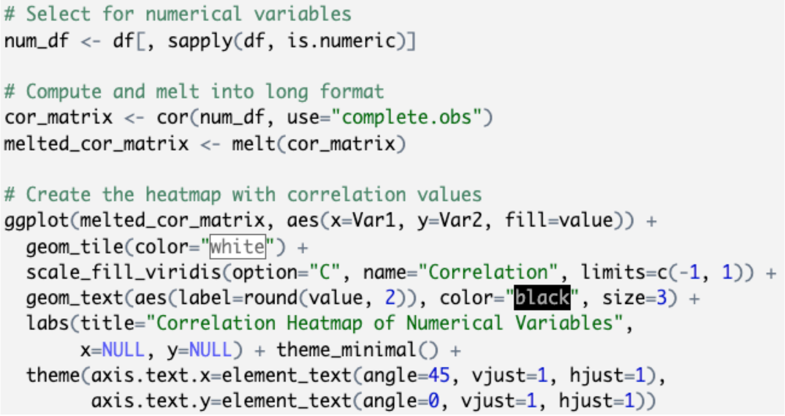
Example of (b):



```r
# Create a histogram of rainfall
ggplot(df, aes(x=Rainfall)) +
  geom_histogram(binwidth=18, fill="lightsteelblue", color="black") +
  labs(title="Rainfall (mm) per day", y="Frequency")
```

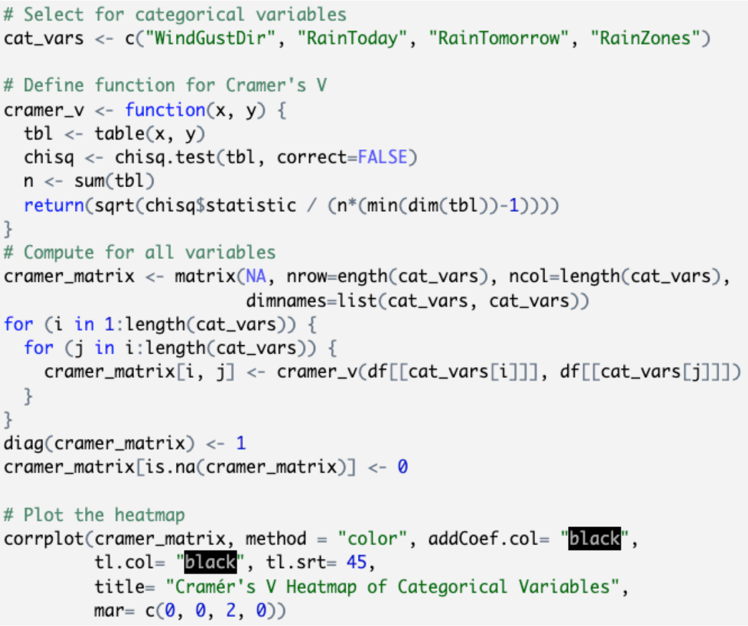
Figure 5: Correlation heatmap of numerical variables
Figure 6(a): Correlation heatmap of numerical variables

Example of Correlation heatmap:



```r
# Select for numerical variables
num_df <- df[, sapply(df, is.numeric)]

# Compute and melt into long format
cor_matrix <- cor(num_df, use="complete.obs")
melted_cor_matrix <- melt(cor_matrix)

# Create the heatmap with correlation values
ggplot(melted_cor_matrix, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile(color="white") +
  scale_fill_viridis(option="C", name="Correlation", limits=c(-1, 1)) +
  geom_text(aes(label=round(value, 2)), color="black", size=3) +
  labs(title="Correlation Heatmap of Numerical Variables",
       x=NULL, y=NULL) + theme_minimal() +
  theme(axis.text.x=element_text(angle=45, vjust=1, hjust=1),
        axis.text.y=element_text(angle=0, vjust=1, hjust=1))
```

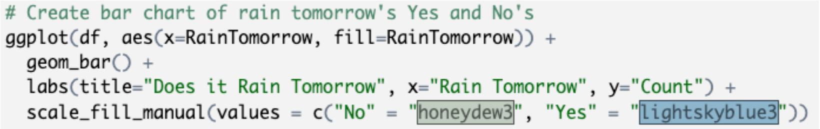
(b): Cramér's V Heatmap of categorical variables



```r
# Select for categorical variables
cat_vars <- c("WindGustDir", "RainToday", "RainTomorrow", "RainZones")

# Define function for Cramer's V
cramer_v <- function(x, y) {
  tbl <- table(x, y)
  chisq <- chisq.test(tbl, correct=FALSE)
  n <- sum(tbl)
  return(sqrt(chisq$statistic / (n*(min(dim(tbl))-1))))
}
# Compute for all variables
cramer_matrix <- matrix(NA, nrow=ength(cat_vars), ncol=length(cat_vars),
                        dimnames=list(cat_vars, cat_vars))
for (i in 1:length(cat_vars)) {
  for (j in i:length(cat_vars)) {
    cramer_matrix[i, j] <- cramer_v(df[[cat_vars[i]]], df[[cat_vars[j]]])
  }
}
diag(cramer_matrix) <- 1
cramer_matrix[is.na(cramer_matrix)] <- 0

# Plot the heatmap
corrplot(cramer_matrix, method = "color", addCoef.col= "black",
         tl.col= "black", tl.srt= 45,
         title= "Cramér's V Heatmap of Categorical Variables",
         mar= c(0, 0, 2, 0))
```

Figure 7(a): Bar chart of rain tommorrow, (b): Bar chart of rain today

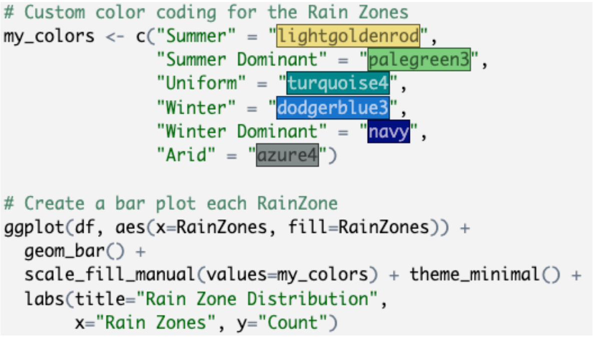
Example:



```r
# Create bar chart of rain tomorrow's Yes and No's
ggplot(df, aes(x=RainTomorrow, fill=RainTomorrow)) +
  geom_bar() +
  labs(title="Does it Rain Tomorrow", x="Rain Tomorrow", y="Count") +
  scale_fill_manual(values = c("No" = "honeydew3", "Yes" = "lightskyblue3"))
```

Figure 8(a): Histogram of rain zones, (b): Histogram of climate classifications

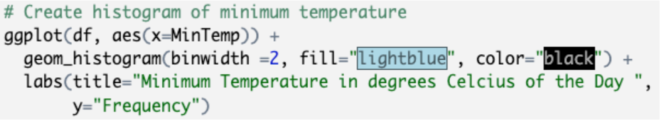
Example:



```r
# Custom color coding for the Rain Zones
my_colors <- c("Summer" = "lightgoldenrod",
               "Summer Dominant" = "palegreen3",
               "Uniform" = "turquoise4",
               "Winter" = "dodgerblue3",
               "Winter Dominant" = "navy",
               "Arid" = "azure4")

# Create a bar plot each RainZone
ggplot(df, aes(x=RainZones, fill=RainZones)) +
  geom_bar() +
  scale_fill_manual(values=my_colors) + theme_minimal() +
  labs(title="Rain Zone Distribution",
       x="Rain Zones", y="Count")
```

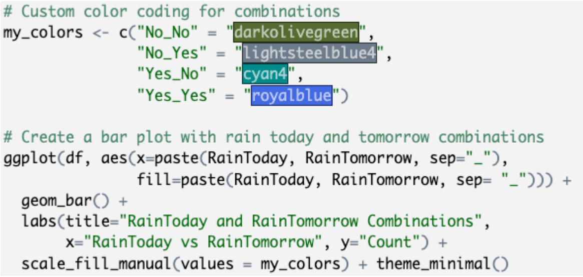
Figure 9(a): Histogram of minimum temperature, (b): Histogram of maximum temperature, (c): Histogram of Sunshine (d): Histogram of wind gust speed

Example:



```r
# Create histogram of minimum temperature
ggplot(df, aes(x=MinTemp)) +
  geom_histogram(binwidth =2, fill="lightblue", color="black") +
  labs(title="Minimum Temperature in degrees Celcius of the Day ",
       y="Frequency")
```

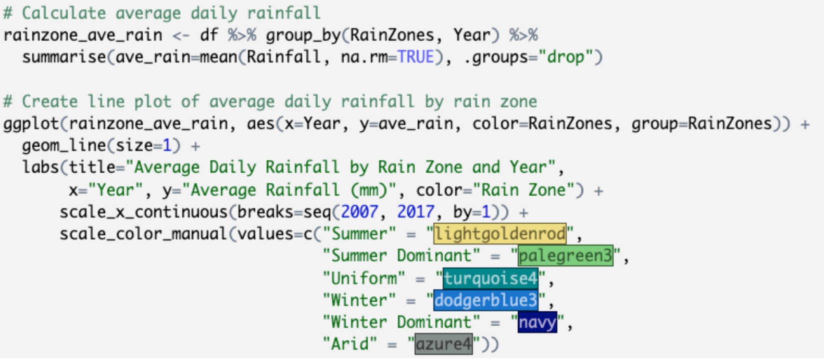
Figure 10: Bar chart of rain today and rain tomorrow combinations



```r
# Custom color coding for combinations
my_colors <- c("No_No" = "darkolivegreen",
               "No_Yes" = "lightsteelblue4",
               "Yes_No" = "cyan4",
               "Yes_Yes" = "royalblue")

# Create a bar plot with rain today and tomorrow combinations
ggplot(df, aes(x=paste(RainToday, RainTomorrow, sep="_"),
               fill=paste(RainToday, RainTomorrow, sep= "_"))) +
  geom_bar() +
  labs(title="RainToday and RainTomorrow Combinations",
       x="RainToday vs RainTomorrow", y="Count") +
  scale_fill_manual(values = my_colors) + theme_minimal()
```

Figure 11(a): Line chart of average daily rainfall by rain zone and year, (b):): Line chart of total annual rainfall by rain zone and year, (c):): Line chart of average daily rainfall by climate classificationand year (d): Line chart of total annual rainfall by climate classification and year

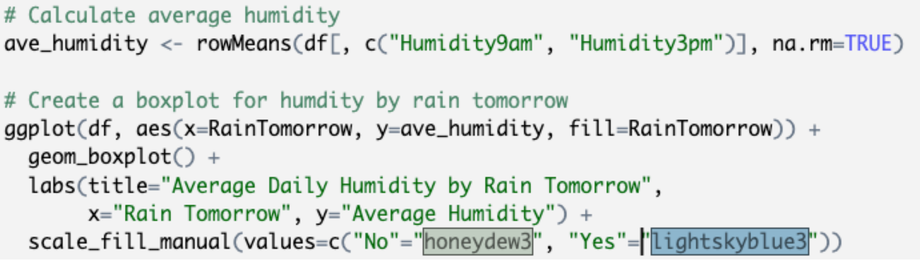
Example:



```r
# Calculate average daily rainfall
rainzone_ave_rain <- df %>% group_by(RainZones, Year) %>%
  summarise(ave_rain=mean(Rainfall, na.rm=TRUE), .groups="drop")

# Create line plot of average daily rainfall by rain zone
ggplot(rainzone_ave_rain, aes(x=Year, y=ave_rain, color=RainZones, group=RainZones)) +
  geom_line(size=1) +
  labs(title="Average Daily Rainfall by Rain Zone and Year",
       x="Year", y="Average Rainfall (mm)", color="Rain Zone") +
      scale_x_continuous(breaks=seq(2007, 2017, by=1)) +
      scale_color_manual(values=c("Summer" = "lightgoldenrod",
                                  "Summer Dominant" = "palegreen3",
                                  "Uniform" = "turquoise4",
                                  "Winter" = "dodgerblue3",
                                  "Winter Dominant" = "navy",
                                  "Arid" = "azure4"))
```

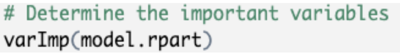
Figure 12(a): Boxplot of average daily humidity by rain tomorrow, (b): Boxplot of average daily pressure by rain tomorrow, (c): Line chart of average daily humidity by rain zone and year (d): Line chart average daily pressue by rain zone and year

Example boxplot:



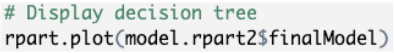
```
# Calculate average humidity
ave_humidity <- rowMeans(df[, c("Humidity9am", "Humidity3pm")], na.rm=TRUE)

# Create a boxplot for humdity by rain tomorrow
ggplot(df, aes(x=RainTomorrow, y=ave_humidity, fill=RainTomorrow)) +
  geom_boxplot() +
  labs(title="Average Daily Humidity by Rain Tomorrow",
       x="Rain Tomorrow", y="Average Humidity") +
  scale_fill_manual(values=c("No"="honeydew3", "Yes"="lightskyblue3"))
```

Figure 13: Rpart variable importance



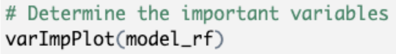
```
# Determine the important variables
varImp(model.rpart)
```

Figure 14: Rpart2 decision tree



```
# Display decision tree
rpart.plot(model.rpart2$finalModel)
```

Figure 15: Random forest variable importance



```
# Determine the important variables
varImpPlot(model_rf)
```