# Australia Next Day Rain Prediction

*Allister Mounsey*

*09 March 2019*

## Contents



# 1 Introduction

In this paper a rain prediction model for various locations in Australia is developed using daily weather data supplied by the Australia Bureau of Meteorology for the period November 2007 to June 2017. Specifically it predicts whether tomorrow's precipitation in a given location will exceed 1 mm - in which case it is labeled **Yes**, otherwise **No**. In selecting a suitable model, binary clasification techniques based on *Logistic Regression* and *Boosted Decision Trees* are explored. The *Receiver Operating Characteristic (ROC)* was the metric used in training these models. Model diagnistics suggests both models are useful predictors with the estimated *Area Under the Curve (AUC)* being 0.892 and 0.894 in favor of the model based on *Boosted Decision Trees*. This suggest that the model based on *Boosted Decision Trees* should be selected based on its (albeit marginally) superior *ROC*. Applying the "preferred" model to the *test dataset* (using the default cut-off of 0.5) result in a *True Positive Rate (Recall)* of 80.9% and a 80.8% *True Negative Rate (Specificity)*.

The remained of the paper is structured as follows. Section 2 outlines the models and techniques employed in greater detail. It also outlines the analysis that assisted in arriving at the articulated models. Section 3 presents the results of the models and conclusions are presented in section 4.

# 2 Methods and Analysis

## 2.1 Data

The was obtained from Kaggle, the following is the link to the dataset: https://www.kaggle.com/jsphyg/weather-dataset-rattle-package/home. The **weather** dataset contains 142,193 daily weather observations from 49 weather stations across Australia over the period November 2007 to June 2017. The following are 24 orginal variables in this dataset:

- **Date**-The date of observation,
- **Location**-The common name of the location of the weather station,

- **MinTemp**-The minimum temperature in degrees celsius,
- **MaxTemp**-The maximum temperature in degrees celsius,
- **Rainfall**-The amount of rainfall recorded for the day in mm,
- **Evaporation**-The so-called Class A pan evaporation (mm) in the 24 hours to 9am,
- **Sunshine**-The number of hours of bright sunshine in the day,
- **WindGustDir**-The direction of the strongest wind gust in the 24 hours to midnight,
- **WindGustSpeed**-The speed (km/h) of the strongest wind gust in the 24 hours to midnight,
- **WindDir9am**-Direction of the wind at 9am,
- **WindDir3pm**-Direction of the wind at 3pm,
- **WindSpeed9am**-Wind speed (km/hr) averaged over 10 minutes prior to 9am,
- **WindSpeed3pm**-Wind speed (km/hr) averaged over 10 minutes prior to 3pm,
- **Humidity9am**-Humidity (percent) at 9am,
- **Humidity3pm**-Humidity (percent) at 3pm,
- **Pressure9am**-Atmospheric pressure (hpa) reduced to mean sea level at 9am,
- **Pressure3pm**-Atmospheric pressure (hpa) reduced to mean sea level at 3pm',
- **Cloud9am**-Fraction of sky obscured by cloud at 9am. This is measured in "oktas", which are a unit of eigths. It records how many eigths of the sky are obscured by cloud. A 0 measure indicates completely clear sky whilst an 8 indicates that it is completely overcast,
- **Cloud3pm**-Fraction of sky obscured by cloud (in "oktas": eighths) at 3pm. See Cload9am for a description of the values,
- **Temp9am**-Temperature (degrees C) at 9am,
- **Temp3pm**-Temperature (degrees C) at 3pm,
- **RainToday**-Boolean: 1 if precipitation (mm) in the 24 hours to 9am exceeds 1mm, otherwise 0,
- ***RISK_MM***-The amount of rain. A kind of measure of the "risk",
- **RainTomorrow**-The target variable. Did it rain tomorrow?

The variable ***RISK_MM*** was deleted from the dataset based on the cautionary note from the Kaggle webpage advising that this variable should be excluded from binary classification models because it will "leak the answers to your model and reduce its predictability".

The variables **WindDir9am/3pm** can legitimately take on NA values in the absence of wind, that is, when **WindSpeed9am/3pm = 0** respectively. This is addressed by creating a new factor level called *NoDir* for these cases. After accounting for the *NoDir* cases in the WindDir variables, approximately 60% of cases (rows) in the dataset contained atleast one or more missing values. Quick examination of these cases reveal a tendency for some stations (**Locations**) to consistently report NAs for some variables,in such cases imputation does seem advisable. All these missing cases were therefore dropped, resulting in the dataset being reduced to 58,079 observations over 26 unique **Locations** as opposed to 49 orginally. The analysis and results that follow is for these 26 locations using the truncated dataset.

Having cleaned the **weather** dataset, it was then randomly split in an 8:2 ratio. The larger dataset was used for training the models and is herein referred to as the **trainSet** while the remaining 20% was reserved exclusively for testing purposed (the **testSet**). All data analysis was conducted only on the **trainSet**.

## 2.2 Data Analysis

In thinking of constructing a rain prediction model for Australia, rudimentary inquiry into rainfall partners across the continent was conducted. Figure 1 represents an insightfully summary of rainfall pattern across the country. The figure shows progressively lower number of raindays as one moves inland from the northern, eastern and southern coasts, with interior and western regions having very few rain days.

Another important weather feature is the seasonal patterns in rainfall. Figure 2 illustrates two readily observed seasonal rain patterns. In some regions like Darwin there is a Summer (Southern Hemisphere) Monsoon - where over the period December to March there is significantly increased rainfall, both in terms of volume and the number of rain days in the month. While in areas like Portland there is a Winter Monsoon starting around May and ending in September.

Figures 1 and 2 points to the possibility that combinations of location and months may be jointly predictive

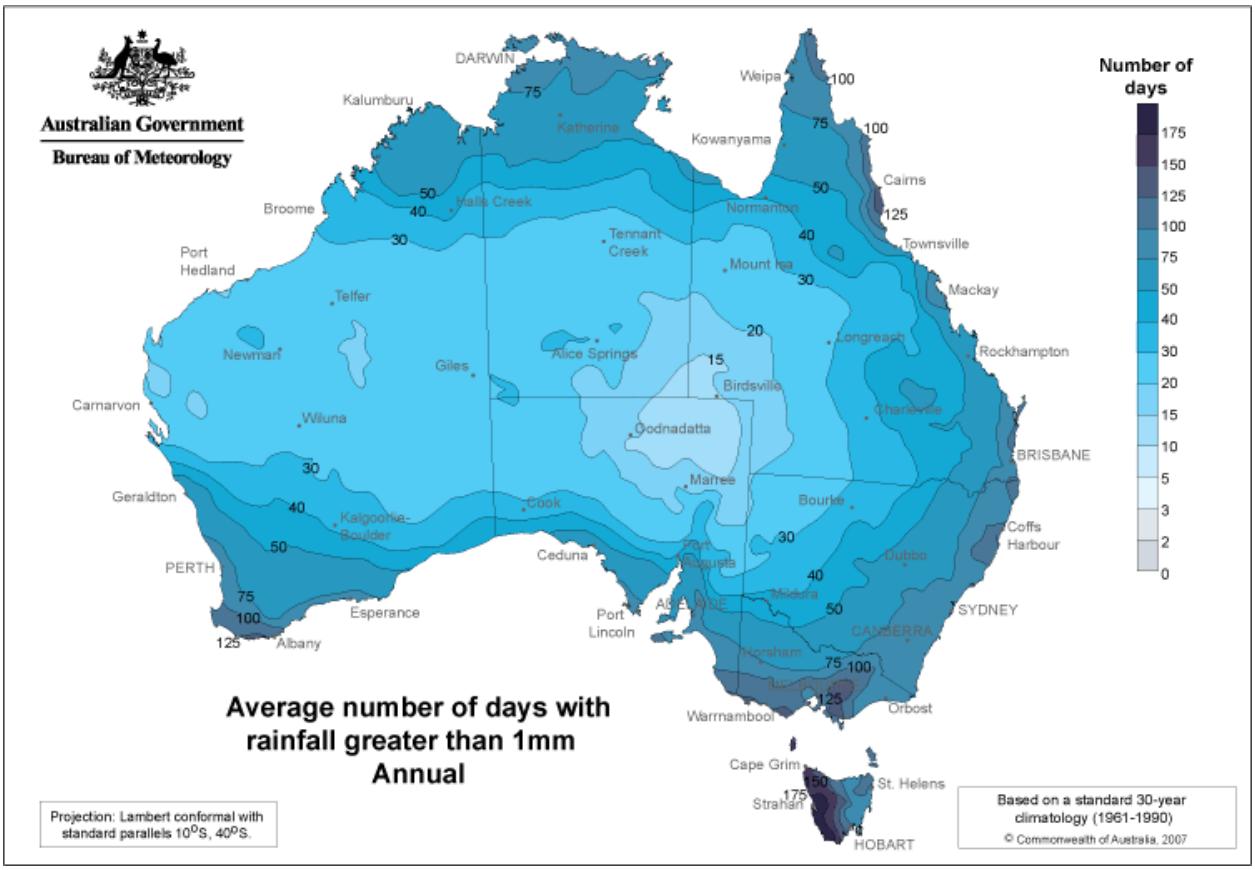


Figure 1: Rain Days across Australia Source: http://www.bom.gov.au/jsp/ncc/climate_averages/raindays/index.jsp

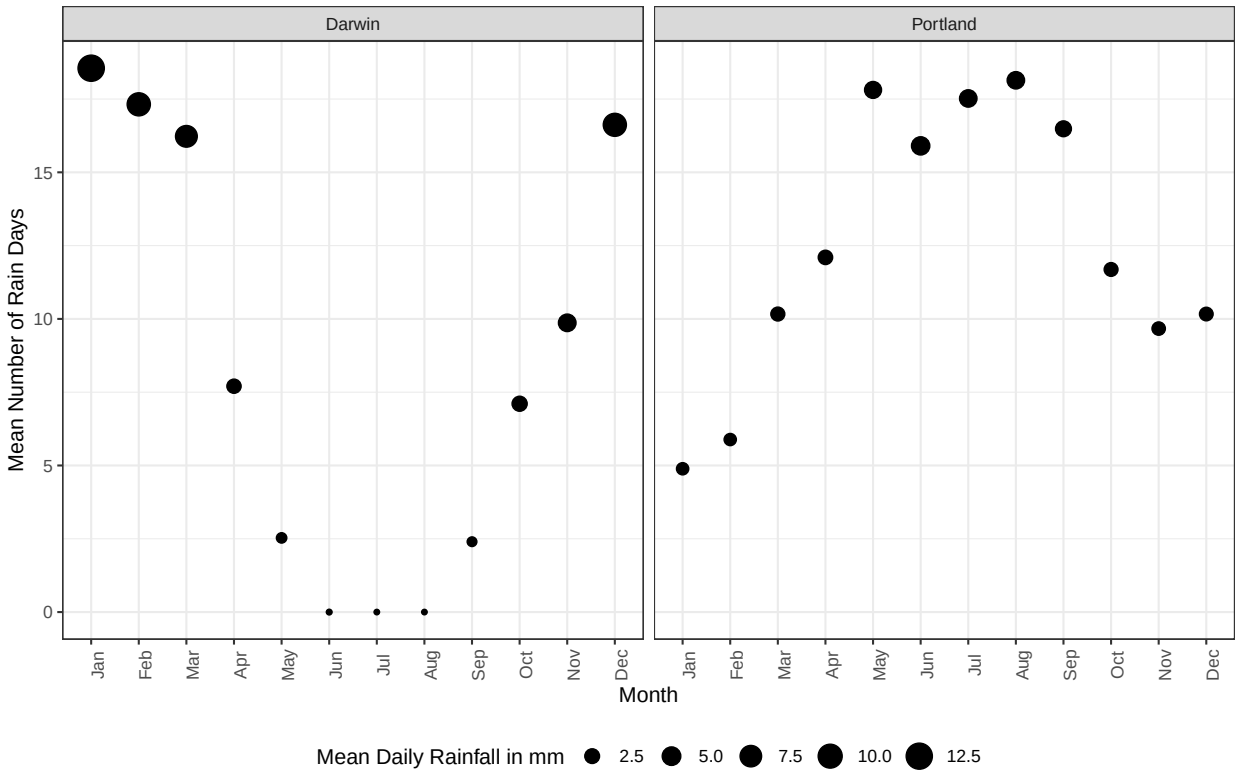


Figure 2: Monthly Rainfall Patterns by Location

of the *label* - **RainTomorrow**. For example, Darwin in the month of January (Darwin-January), any given day typically has a $\frac{20}{30}$ chance of rain where as in Portland-January the chance is typically $\frac{5}{30}$. The contrast is however reversed if Portland-July and Darwin-July are examined. This phenomenon, referenced here as *Location-Month Effect*, will be returned to in the next sub-section.

## 2.3 Feature Selection and Engineering

Each numeric *feature* in the **trainSet** was examine to determine the extent to which it is separated along *label* values - that is the degree to which the *conditional distribution* of *feature* $X$ given **RainTomorrow**= 'Yes' is different from the *conditional distribution* of *feature* $X$ given **RainTomorrow**= 'No'. This examination was conducted at two ways: (1) using *box-plots* and (2) using the *Kolmogorov–Smirnov test (KS test)*. Some *features* are 'well separated' as can be observed in figure 3. This figure shows that the distribution of **Humidity3pm** conditioned on **RainTomorrow**= 'Yes' is shifted to the right (that it typically contains higher values) of that which is conditioned on **RainTomorrow**= 'No'. **Clouds3pm** is similarly well separated.

Some examples of 'not well separated' features are illustrated in figure 4. In figure 4 it is easily observed that the degree of distributional overlap is high across cases conditioned on the outcome of the *Label* -**RainTomorrow**.

The test statistic - $D$ for the *KS test* was used to determine the relative strength (degree) of 'separatedness', as well as, to determine a cut-off point for what should be considered as reasonably 'separated' *features*. The cut-off point choosen was $D = 0.1$. *Features* with $D \geq 0.1$ were considered 'well-separated' and were selected as *features* to be used in constructing the models.

The *Location-Month Effect (LME)*, referred to previously, may prove to be predictive of the *Label*. Since there are 26 locations in the **trainSet**, an attempt to directly leverage the LME would entail creating a total of

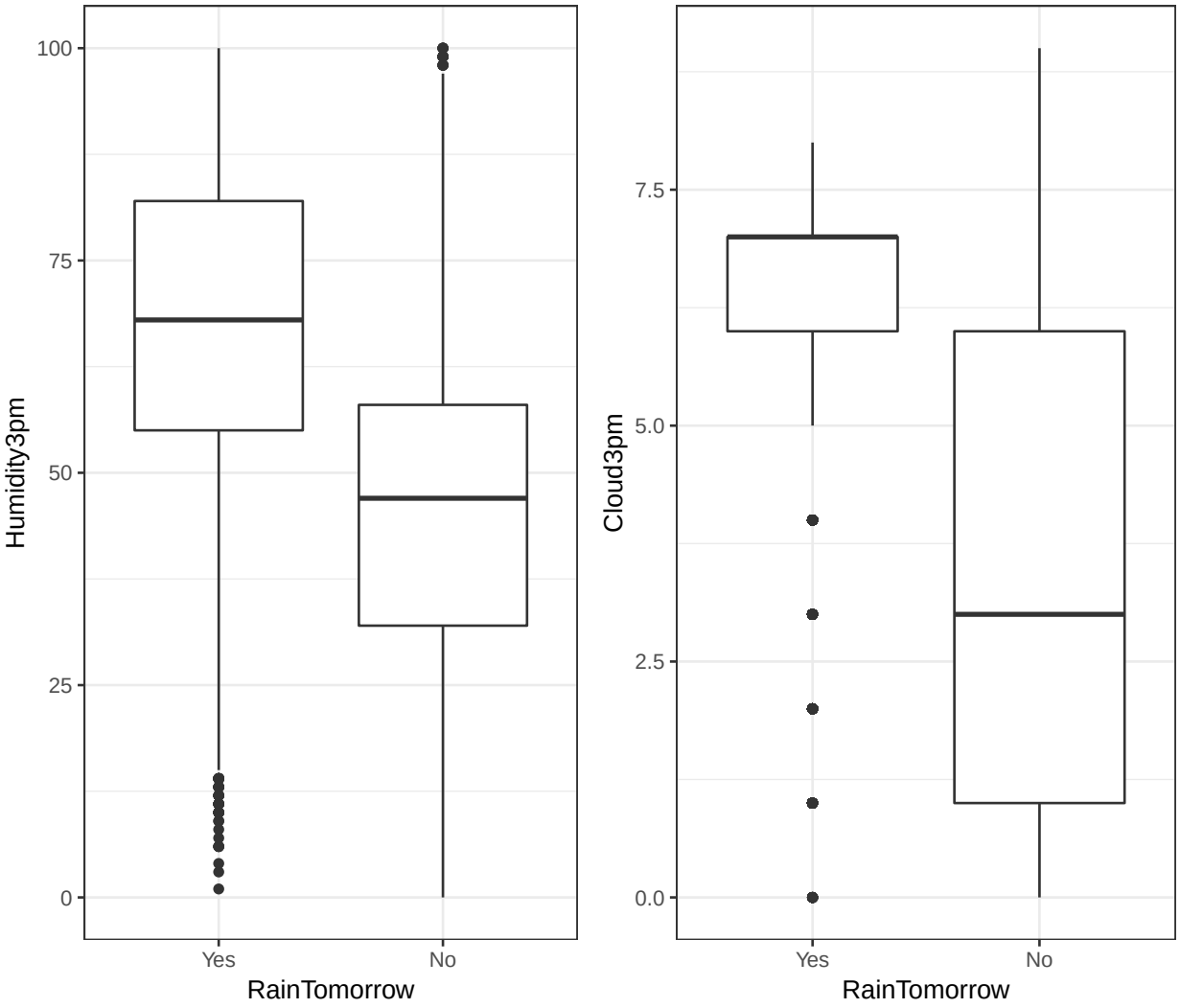


Figure 3: Examples of Some Well Separated Cases

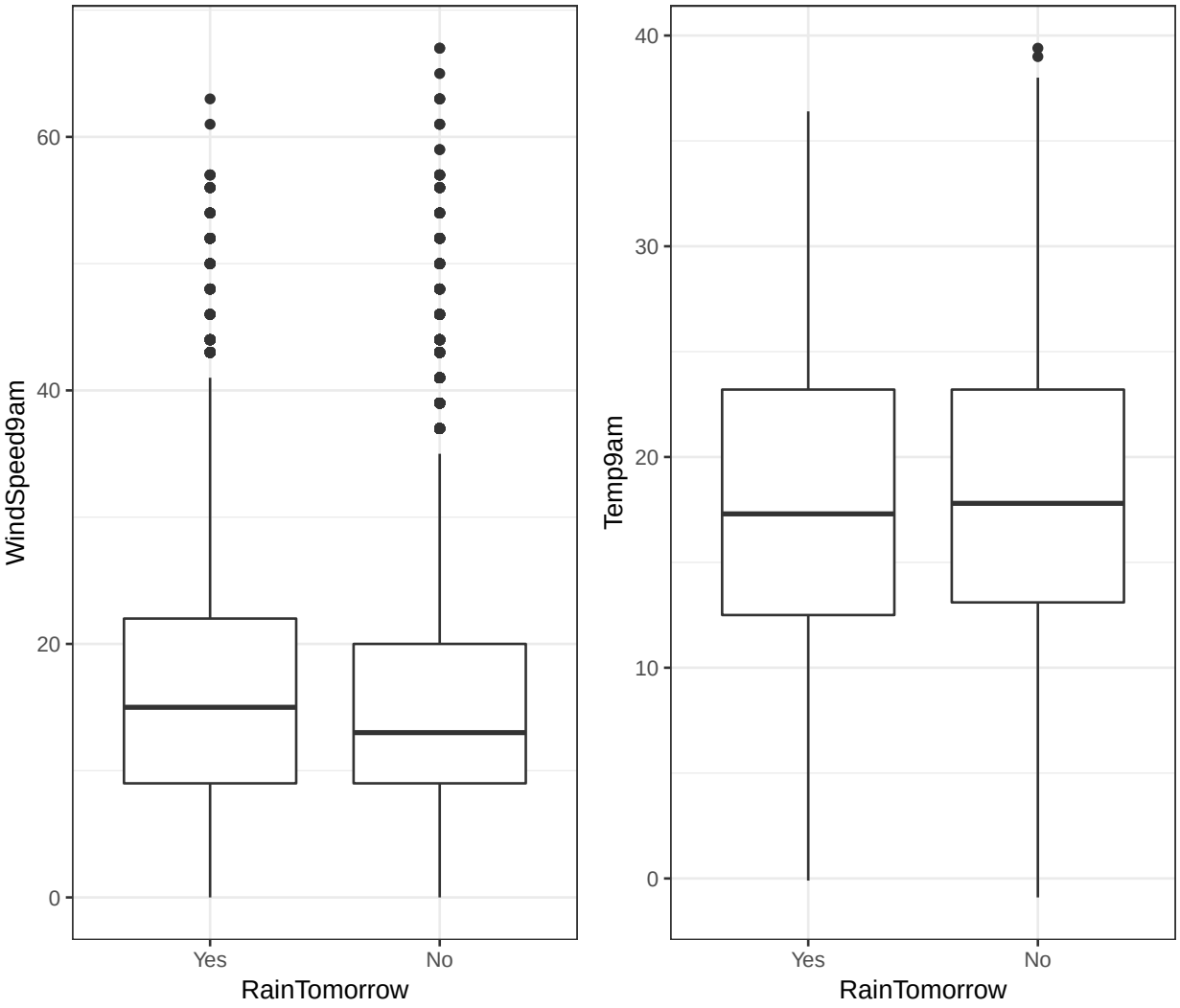


Figure 4: Examples of Some **NOT** Well Separated Cases

Table 1: Results of Kolmogorov-Smirnov test on Conditional Distributions (RainTomorrow = No vs. RainTomorrow = Yes)

| NumericVariable | D |
|---|---|
| MinTemp | 0.08 |
| MaxTemp | 0.14 |
| Rainfall | 0.35 |
| Evaporation | 0.14 |
| Sunshine | 0.47 |
| WindGustSpeed | 0.22 |
| WindSpeed9am | 0.08 |
| WindSpeed3pm | 0.08 |
| Humidity9am | 0.29 |
| Humidity3pm | 0.47 |
| Pressure9am | 0.25 |
| Pressure3pm | 0.22 |
| Cloud9am | 0.34 |
| Cloud3pm | 0.41 |
| Temp9am | 0.04 |
| Temp3pm | 0.18 |

312 ($26 \times 12$) factor levels and more critically estimating 311 of them. To overcome this computational cost, a *clustering* approach is employed to simultaneously leverage the LME while reducing computational load.

The follow steps outline the procedure employed in computing these clusters:

- A dataset of the number of Rain Days and its standard deviation in each *Location-Month* group is computed by grouping the *trainSet* by **Location** and **Month** (the variable **Month** is created by extracted the month from each **Date** observation).
- K-means cluster is performed on these 312 paired observations.

The number of clusters ($K$) was selected using the *within cluster sums of squares* and *average silhouette* methods as guides. Figure 5, shows the results using a maximum cut-off of 10. An 'elbow' at $K = 3$ is identified in the *within cluster sums of squares*, while the *average silhouette* suggest $K = 10$. Ultimate $K = 6$ was selected because it yeilded an *average silhouette width* that was closest to that of the suggested value and it is closer to 3 (the 'elbow'in the *within cluster sums of squares*) than the suggested $K$ from the *average silhouette width* method.

Figure 6 shows the number of rain days and the variability of rain days across these clusters. Table 2 provides **Location** and **Month** composition of each cluster.

Table 3 relates to the town of Cobar. It provides the conditional probability of rain tomorrow (Yes) given the wind gust direction, as well as, the conditional probability of no rain tomorrow (No) given the wind gust direction. Representing the probability of rain tomorrow as $P(R)$ and the conditional probability of rain tomorrow given a wind gust direction of $D$ as $P(R/D)$, one notes that in some directions (such as NE, NNE and ESE) the $P(R/D) \geq P(R)$. This means, that knowing the wind gust is coming from these directions, the chances of correctly guessing $R$ - rain tomorrow is higher that correctly guessing $R$ when the wind gust direction in not revealed. Wind gust from these directions therefore increases the probability of having rain tomorrow.

Similar tables were constructed for other locations and for other wind direction variables (that is **WindDir9am** and **WindDir3am**). Binary classification variables **rainDir9am**, **rainDir3pm** and **rainDirGust** were created with wind directions where $P(R/D) \geq P(R)$ catalogued as 'Yes'. Tables 4, 5 and 6 show the probabilities of rain tomorrow conditioned on **rainDir9am**, **rainDir3pm** and **rainDirGust** respectively.

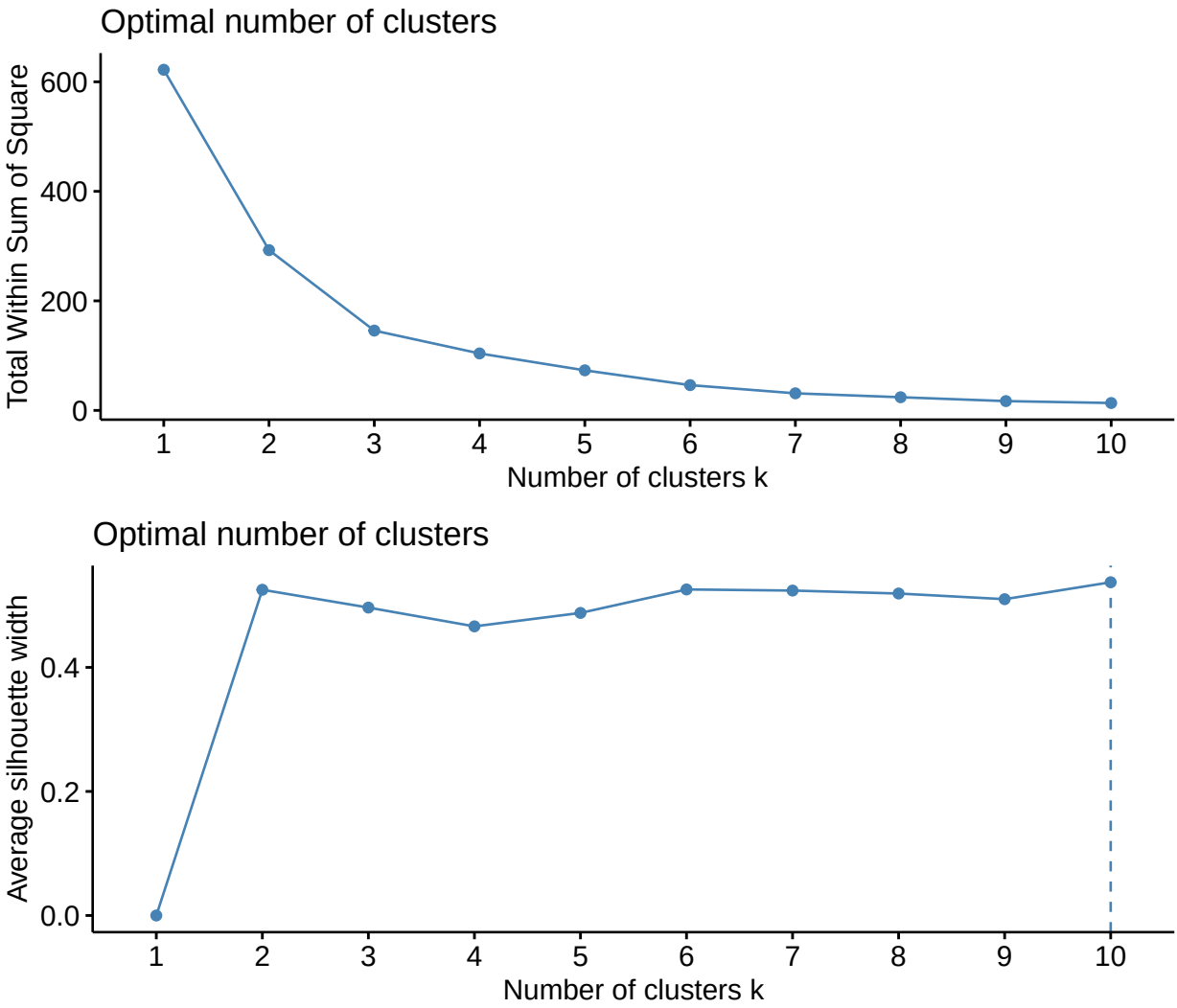


Figure 5: Finding Optimal Number for Location - Month Clustering

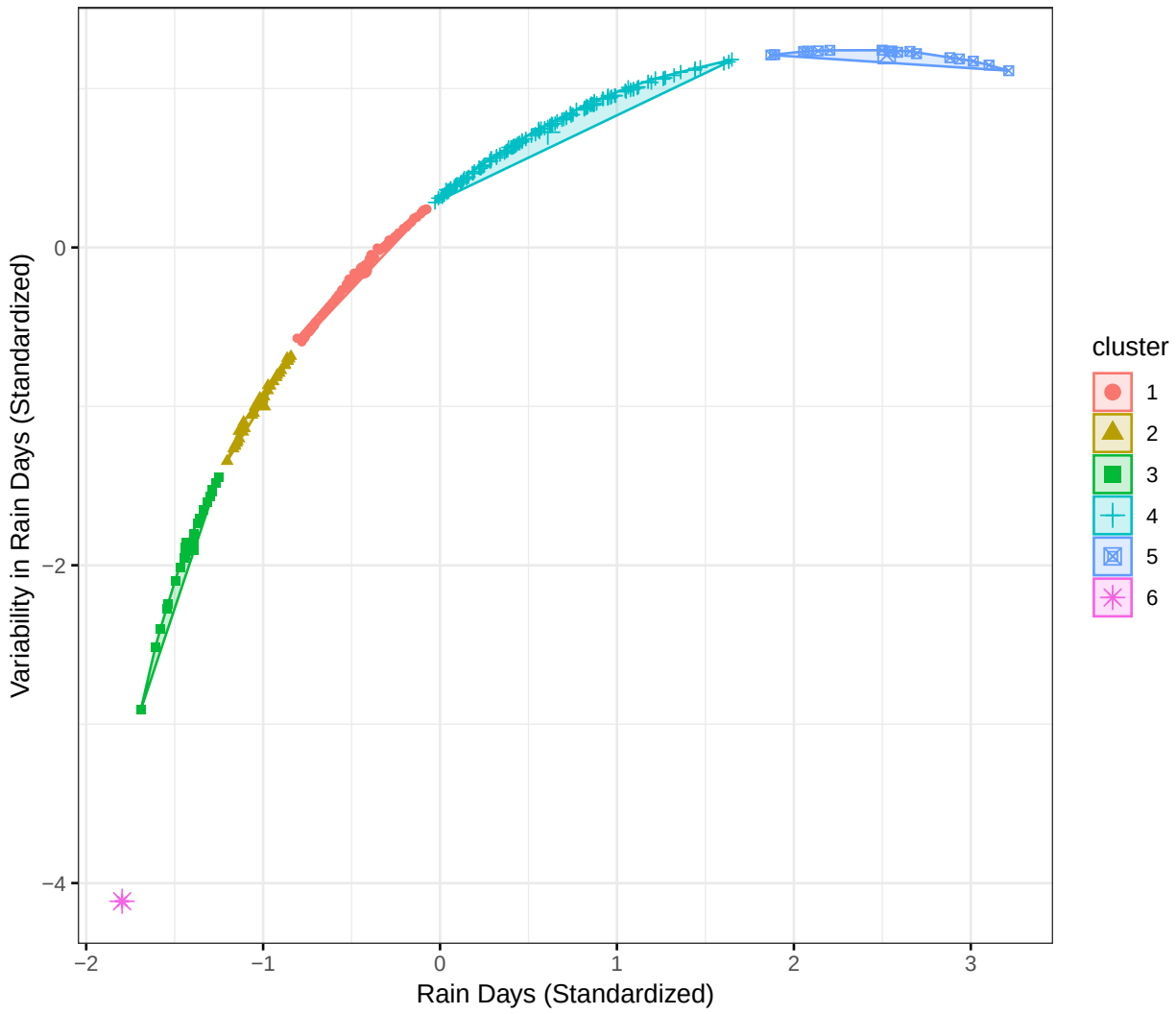


Figure 6: Number of Rain Days and Variability in Rain Days across Clusters

Table 2: Location-Month Clusters

| Location | Jan | Feb | Mar | Apr | May | Jun | Jul | Aug | Sep | Oct | Nov | Dec |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AliceSprings | 1 | 1 | 2 | 3 | 3 | 3 | 2 | 3 | 3 | 3 | 2 | 1 |
| Brisbane | 4 | 4 | 4 | 1 | 1 | 4 | 1 | 2 | 1 | 1 | 4 | 4 |
| Cairns | 5 | 5 | 5 | 4 | 4 | 1 | 1 | 2 | 2 | 4 | 4 | 4 |
| Canberra | 1 | 4 | 1 | 1 | 2 | 1 | 1 | 4 | 1 | 1 | 4 | 4 |
| Cobar | 2 | 1 | 2 | 1 | 1 | 1 | 2 | 6 | 3 | 2 | 2 | 1 |
| CoffsHarbour | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 1 | 1 | 4 | 4 | 4 |
| Darwin | 5 | 5 | 5 | 4 | 2 | 6 | 6 | 6 | 2 | 4 | 4 | 5 |
| Hobart | 1 | 1 | 4 | 1 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| Melbourne | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| MelbourneAirport | 1 | 2 | 1 | 4 | 1 | 4 | 4 | 4 | 4 | 4 | 4 | 1 |
| Mildura | 2 | 3 | 2 | 2 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| Moree | 4 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 4 |
| MountGambier | 1 | 1 | 4 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 1 | 1 |
| NorfolkIsland | 4 | 4 | 4 | 4 | 5 | 4 | 5 | 4 | 4 | 1 | 1 | 1 |
| Nuriootpa | 2 | 2 | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 1 | 1 | 1 |
| Perth | 3 | 3 | 3 | 2 | 4 | 4 | 4 | 4 | 4 | 1 | 2 | 3 |
| PerthAirport | 3 | 3 | 2 | 1 | 4 | 4 | 4 | 4 | 4 | 1 | 2 | 3 |
| Portland | 1 | 1 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 4 |
| Sale | 1 | 4 | 4 | 1 | 4 | 4 | 1 | 4 | 4 | 4 | 1 | 1 |
| Sydney | 1 | 4 | 4 | 4 | 1 | 4 | 4 | 1 | 1 | 1 | 4 | 4 |
| SydneyAirport | 4 | 4 | 4 | 4 | 1 | 4 | 4 | 1 | 1 | 4 | 4 | 4 |
| Townsville | 4 | 5 | 4 | 1 | 2 | 2 | 3 | 3 | 3 | 3 | 1 | 1 |
| WaggaWagga | 2 | 1 | 1 | 1 | 1 | 4 | 4 | 4 | 1 | 1 | 1 | 1 |
| Watsonia | 1 | 1 | 1 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 1 |
| Williamtown | 1 | 4 | 4 | 4 | 4 | 4 | 4 | 1 | 1 | 4 | 4 | 4 |
| Woomera | 3 | 2 | 3 | 2 | 2 | 3 | 2 | 3 | 3 | 2 | 2 | 3 |

Table 3: Conditional Probabilities No Rain Tomorrow vs. Rain Tomorrow Given WindGust Direction (Location = Cobar)

| | E | ENE | ESE | N | NE | NNE | NNW | NW | S | SE | SSE | SSW | SW | W | WNW | WSW | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Yes | 0.18 | 0.1 | 0.19 | 0.05 | 0.32 | 0.26 | 0.2 | 0.18 | 0.07 | 0.07 | 0 | 0.03 | 0.07 | 0 | 0.18 | 0.11 | 0.12 |
| No | 0.82 | 0.9 | 0.81 | 0.95 | 0.68 | 0.74 | 0.8 | 0.82 | 0.93 | 0.93 | 1 | 0.97 | 0.93 | 1 | 0.82 | 0.89 | 0.88 |
| Total | 1.00 | 1.0 | 1.00 | 1.00 | 1.00 | 1.00 | 1.0 | 1.00 | 1.00 | 1.00 | 1 | 1.00 | 1.00 | 1 | 1.00 | 1.00 | 1.00 |

Table 4: Probabilites of Rain Tomorrow Conditioned on rainDir9am

| | RainTomorrow | | |
|---|---|---|---|
| | Yes | No | Total |
| **rainDir9am** | | | |
| No | 0.14 | 0.86 | 1 |
| Yes | 0.31 | 0.69 | 1 |
| Total | 0.22 | 0.78 | 1 |

These four engined features (**cluster**, **rainDir9am**, **rainDir3pm** and **rainDirGust** ) along with the twelve numeric features selected on the basis of 'separatedness', are used in developing predictive models for whether

Table 5: Probabilites of Rain Tomorrow Conditioned on rainDir3pm

| | RainTomorrow | | |
|---|---|---|---|
| | Yes | No | Total |
| **rainDir3pm** | | | |
| No | 0.14 | 0.86 | 1 |
| Yes | 0.32 | 0.68 | 1 |
| Total | 0.22 | 0.78 | 1 |

Table 6: Probabilites of Rain Tomorrow Conditioned on rainDirGust

| | RainTomorrow | | |
|---|---|---|---|
| | Yes | No | Total |
| **rainDirGust** | | | |
| No | 0.13 | 0.87 | 1 |
| Yes | 0.32 | 0.68 | 1 |
| Total | 0.22 | 0.78 | 1 |

or not there will be rain tomorrow. Before this done however, each numeric variable is normalized by 'centering' (subtracting its mean from each observation) and 'scaling' by its standard deviation.

## 2.4 Models and Estimation Techniques

### 2.4.1 Logistic Model

A logistic model is motivated from an assumption that the log *odds* of an (*label*) event, that is $ln\frac{p}{1-p}$, can be modeled as a function of a vector of *features* and a vector of parameters associated with these *features*. This function is linear in its paramters. Equation (1) represents this assumption in algebraic form.

$$ln\frac{p(x_{i,1}, x_{i,2}, ..., x_{i,M})}{1 - p(x_{i,1}, x_{i,2}, ..., x_{i,M})} = \sum_{j=0}^{M} b_j x_{i,j}$$

alternatively in vector notation

$$ln\frac{p(\mathbf{x_i})}{1 - p(\mathbf{x_i})} = \mathbf{x_i}\beta \tag{1}$$

where:

$x_{i,0} = 1$

$x_{i,j}$ represents the $i^{th}$ observation feature $j$, $\forall\ j \in (1, M)$

$\mathbf{x_i}$ is a row-vector for observation $i$

$\beta$ is column-vector of parameters

Staying with vector notation and solving for $p(\mathbf{x_i})$ in (1) one gets:

$$p(\mathbf{x_i}) = \frac{e^{\mathbf{x_i}\beta}}{1 + e^{\mathbf{x_i}\beta}} = \frac{1}{1 + e^{-\mathbf{x_i}\beta}} \tag{2}$$

Equation (2) shows that with the assumption made in (1) above $p(\mathbf{x_i})$ follows a logistic distribution. It is therefore possible to use parameter estimates from (1) to generate probability estimates via the *link* function in (2).

Using an appropriate *cut-off* ($K^*$) these probabilities can be converted to predictions of the *label* ($\hat{y}$). That is:

$$\hat{y} = \begin{cases} 1 \;\; p(\mathbf{x_i}) \geq K^* \\ \\ 0 \;\; p(\mathbf{x_i}) < K^* \end{cases} \tag{3}$$

An initial logistic model was developed using the 16 selected features (12 numeric and the 4 engineered dummy variables). This model was *trained* in a *10-fold nested cross validation* framework where optimal values for *hyperparameters* were obtained and futher *feature* selction (elimination) was conducted.

The *ROC* was used as the metric on which the model *trained*. The *ROC* is a curve that maps the trade-off between *Sensitivity* and *Specificity*. When optimizing using the *ROC*, the model that produces a *ROC* with the greatest *area under the curve (AUC)* is generally preferred.

*Training* the model involved:

1. Determining optimal values of *hyperparameters*. For logistic models there are two *hyperparameters*:
    - $\alpha$ - *the mixing percentage* that controls how of each type of *penalty* is applied to the model. There two types of *penalties*, they ar used for avoiding overfitting, reducing variance of the prediction error and dealing with correlated predictors (H2O.ai, n.d.). *Least Absolute Shrinkage and Selection Operator (LASSO)* usually referred to as $l_1$ *regularization*- penalizes the sum of absolute coefficients (the $l_1$ norm) . With a high *tuning parameter value* $\lambda$ it will result significant (possibly all) coefficients being set to zero- that is *features* removed from the model (H2O.ai, n.d.). *Ridge Regression* usually referred as $l_2$ *regularization*- penalizes the sum of squares of the coefficients (the $l_2$ norm). $l_2$ *regularization* reduces coefficients simultaneously without setting to zero, the extent of reduction is determined by the *tuning parameter* - $\lambda$.
    - $\lambda$ -sometimes called the *tuning/regularization/shrinkage parameter* controls the strength of *penalty* to be applied to the model.
2. If appropriate further reducing in the number *features* used.

The (final) *trained* model was then evaluated in the last stage of the *nested cross validation* to determine expected model performance. The expected performance of the *trained* logistic model was compared to that from the *Boosted Decision Tree* classifier to determine if it merits *testing* (final evaluation using the *testSet*) and probably ultimately being placed into production.

### 2.4.2 Boosted Decision Tree Model

*Decision trees* splits the observations of a set of *features* recursively to make predictions about a *label* (see figure 7 for an example). Each split minimizes *entropy* - roughly speaking: the degree of mixing of ojects with different *label* values (Ricaud 2017). Splitting continues until some stop criterion is met. *Decision tree* algorithms generally produce weak classifiers - meaning they are slightly better than guessing. *Boosting* is a method of producing and combining many weak classifers to make a strong classifier (Woodruff 2017). In *Gradient boosting* trees are created iteratively. The weak classifier (the current decision tree) trains on the pseudo-residuals of the *strong/base* learner (the model resulting from aggregating past trees). The contribution of the weak learner to the (new) *strong/base learner* is determined by a *gradient descent* optimization process, where its contribution is, that which minimizes the overall error of the *strong/base* learner (Moise 2017). *Stochastic gradient boosting*- the technique applied here - is a modification to *gradient boosting* where at each iteration a sample of the training data is selected randomly without replacement. This randomly selected sample is then used to fit the *base learner* and compute the model update for the current iteration'(Friedman

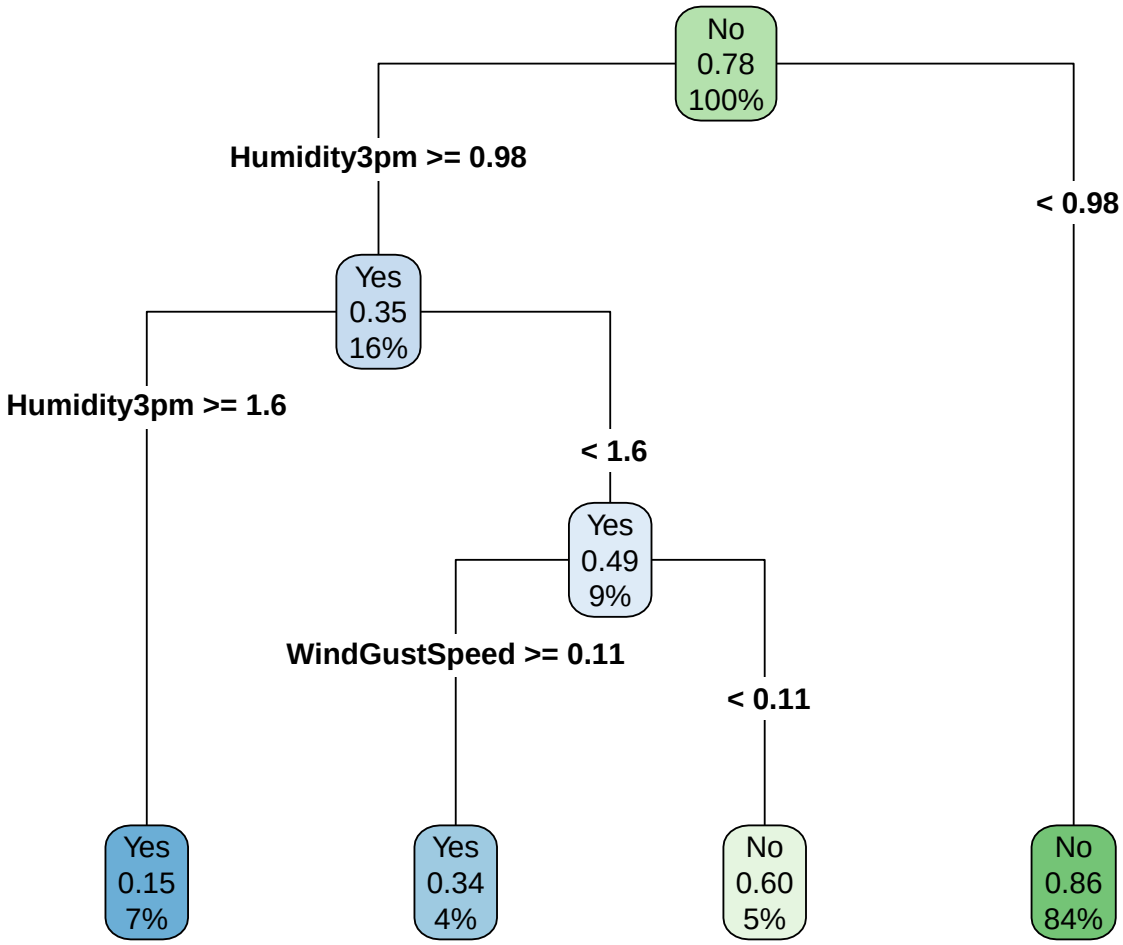


Figure 7: Example of a Decision Tree

2002). Friedman (2002) suggests that the introduction of randomization substantially improves accuracy of *gradient boosting*.

The boosted decision tree model was *trained* in a *5-fold nested cross validation* framework where optimal values for *hyperparameters* were obtained. As *boosting* seems less prone to *overfitting* (Khan 2016; Parkes 2012; Kun 2015), no attempt was made (unlike in the Logistic model) to futher reduce the number of *features* in the model. Again, the *ROC* was used as the optimizing metric in *training*. The *Stochastic Gradient Boosting* algorithm implimented via the *gbm* (Greenwell et al. 2018) and *caret* (Kuhn 2018) packages, has a default value of 0.1 for the shrinkage ($\lambda$) *hyperparameter*, this was was maintained. The minimum number of observation in each node (n.minobsinnode) was also kept at its default value of of 10 observations. The *ROC* was optimized by altering interaction depth and number of trees.

# 3 Results

## 3.1 Expected Performance - Logistic Model

Table 7 shows the result of *Hyperparameter* tuning on the full (all 16 *features*) Logistic model. It indicates that $l_2$ regularization ($\alpha = 1$) should be applied with a relatively small *regularization parameter* ($\lambda = 0.0004$). The importance of individual variables to the logistic model was also examined.

Table 7: Optimal Hyperparameter Selection Using Logistic Model 10-fold nested Cross Validation

| | alpha | lambda |
|---|---|---|
| Optimal | 1 | 0.0003768 |

Figure 8 shows the relative value of the coefficients (relative to the coefficient with the highest value) in the model (direct comparision of coefficient is possible as *features* have been *normalized*). Features with a variable importance of less than 3 (**Temp3pm**, **Rainfall**, **Humidity9am** and **Evaporation** ), were dropped to produce a *reduced model*. The *reduced model* was again trained using *cross-validation*. The *ROC* for the *reduced model* with the best tuned *hyperparameters* was very close to that of the *full model* and the best tuned *hyperparameters* were identical to those in table 7. On the basis of the similarity in *ROC*, the reduced model was preferred as in theory it reduces the chance of *overfitting*.

Using the *reduced model* and its best tuned *hyperparameters*, *10-fold cross validation* was employed to determine what the expected performance of the best tuned model might be. Table 8 shows the summary statistics. The table shows that the *ROC* is expected to have an *AUC* of 0.8915 with a stardard deviation of 0.0039 and that using the default cut-off (that is, a positive label if $P(\mathbf{x}) >= 0.5$) *sensitivity* or *recall* is expected to be 0.8069 and *specificity* 0.8076 with standard deviations of 0.0061 and 0.0096 respectively.

Table 8: Expected Performance of the Logistic Model

| alpha | lambda | ROC | Sens | Spec | ROCSD | SensSD | SpecSD |
|---|---|---|---|---|---|---|---|
| 1 | 0.0003768 | 0.8915102 | 0.8068762 | 0.8076284 | 0.0038726 | 0.0060766 | 0.0095778 |

## 3.2 Expected Performance- Boosted Decision Tree Model

Table 9 shows the best tuned *hyperparameters* from the *boosted decision tree* model estimated through *stochastic gradient boosting*. Optimization was performed on the number of trees (n.trees) and the interaction depth, which were tuned at 150 and 3 respectively. *Shrinkage* and the minimum number of observations in each were held constant at default values.

Table 9: Optimal Hyperparameter Selection for Boosted Decision Tree Model Using 5-fold nested Cross Validation

| | n.trees | interaction.depth | shrinkage | n.minobsinnode |
|---|---|---|---|---|
| Optimal | 150 | 3 | 0.1 | 10 |

The 'best tuned' *hyperparameters* values were used in *5-fold cross validation* framework to determine the expected performance of the model . Table 10 shows the summary statistics from this exercise. The *ROC* is expected to have an *AUC* of 0.8938 with a stardard deviation of 0.0044 and that using the default cut-off (that is, a positive label if $P(\mathbf{x}) >= 0.5$) *sensitivity* or *recall* is expected to be 0.8083 and *specificity* 0.8081 with standard deviations of 0.0068 and 0.0066 respectively.

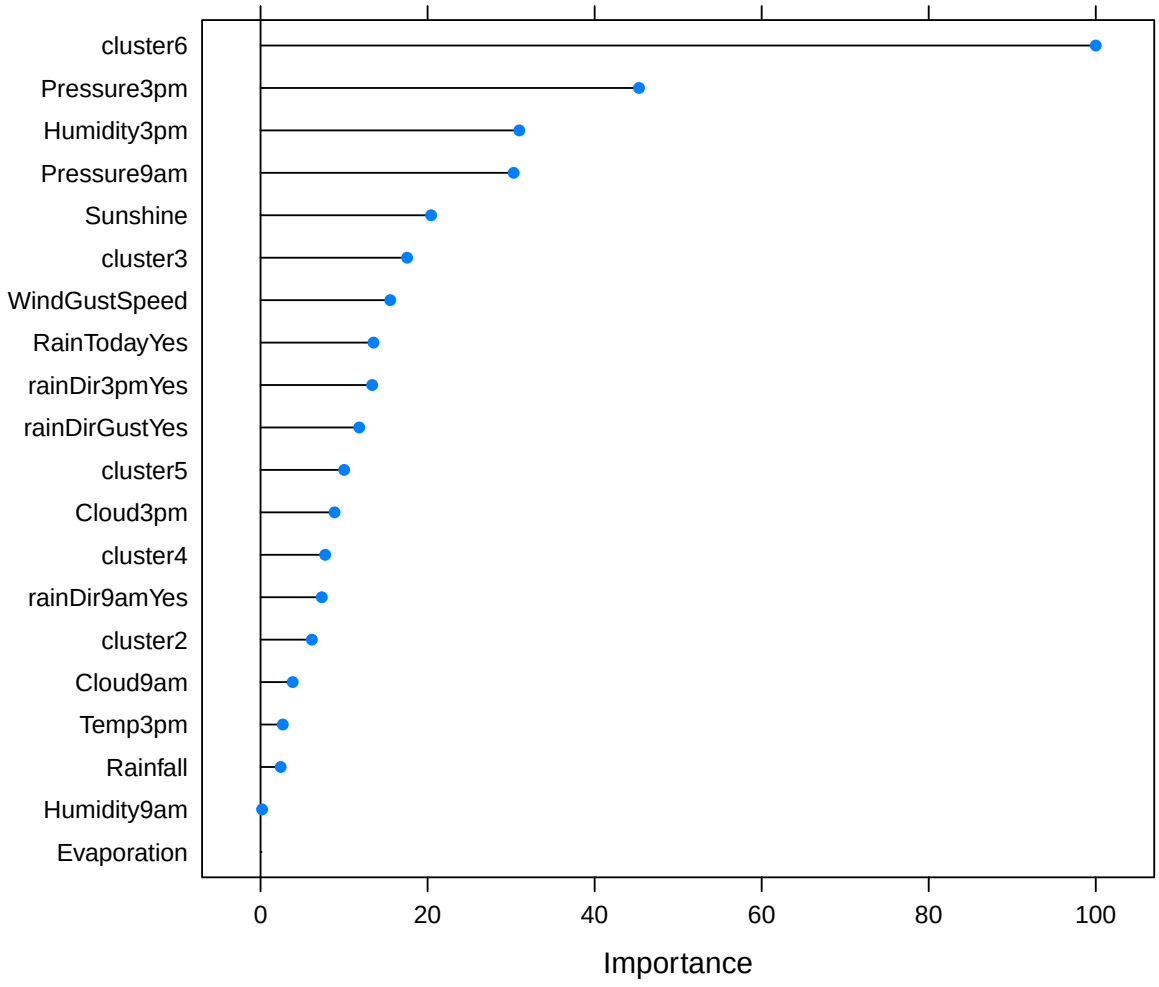


Figure 8: Variable Importance for Logistic Model

Table 10: Expected Performance of the Boosted Decision Tree Model

| n.trees | interaction.depth | ROC | Sens | Spec | ROCSD | SensSD | SpecSD |
|---|---|---|---|---|---|---|---|
| 150 | 3 | 0.8938409 | 0.8083497 | 0.808097 | 0.0043946 | 0.0068181 | 0.0066165 |

### 3.3 Comparing Models

Figure 9 shows the estimated (expected) *ROCs* from the *cross-validation* exercises described above. The labeled points on the chart indicate the cut-off points(and the corresponding *specificity*, *sensitivity* vector) that will maximize *accurracy*. From the figure the expected *ROCs* are almost identical with that for the *boosted decision tree* model being marginally higher. Unlike the logisitic model, the default cut-off (that is, a positive label if $P(\mathbf{x}) >= 0.5$) is not the cut-off point that generates the highest *accuracy* in the *boosted decision tree* model. This notwithstanding, the default cut-off are maintained in the remainder of this assessment because it seems to produce a results with an equal balance between *recall* and *specificity* which has a sort of 'natural' appeal.

Results from *nested cross-validation* - as interesting as they may be - are not the ones that ultimately matter. The 'ultimate' test, is a model's performance against the **testSet**, attention is now turned to this matter.

Tables 11 and 12 present the *confusion matrix* for the Logistic and the Boosted Decision Tree models respectively. Both models performed creditably, correctly predicting approximately 81% of both the *positive cases* (**RainTomorrow** = Yes) and *negative cases*.

Table 11: Confusion Matrix for Logistic Model

| | Reference | |
|---|---|---|
| | Yes | No |
| **Prediction** | | |
| Yes | 2054 | 1748 |
| No | 491 | 7322 |

Table 12: Confusion Matrix for Boosted Decision Tree Model

| | Reference | |
|---|---|---|
| | Yes | No |
| **Prediction** | | |
| Yes | 2059 | 1739 |
| No | 486 | 7331 |

Table 13 readily allows for a more clinical assessment of the usefulness of both models. The table confirms the usefulness of both models. The accuracy of both models beat the naive accuracy (assigning all observation to the modal class of the *label* variable). More important, *recall* and *specificity* are high- approximately 81% each in both models. Both models have a tendancy to *over detect* positive case with *Detection Prevalance* being about 32.7% in both models as opposed to (the true) *Prevalence* of 21.9%.

## 4 Conclusion

Both models perform similiarly well. the *Sensitivity* and *Specificity* of both models are high - approximately 81%. Both models however have the tendancy to *over detect* positive cases, resulting in low *Positive Prediction Rate* of approximately 54%. If the net cost associated with a *false positive* is low relative the net cost of *false negative* (eg. carrying an umbrella when it didn't rain vs. not carrying and umbrella when it did rain) this may not be a significant challenge to the usefulness of the model . The *negative prediction value* is quite

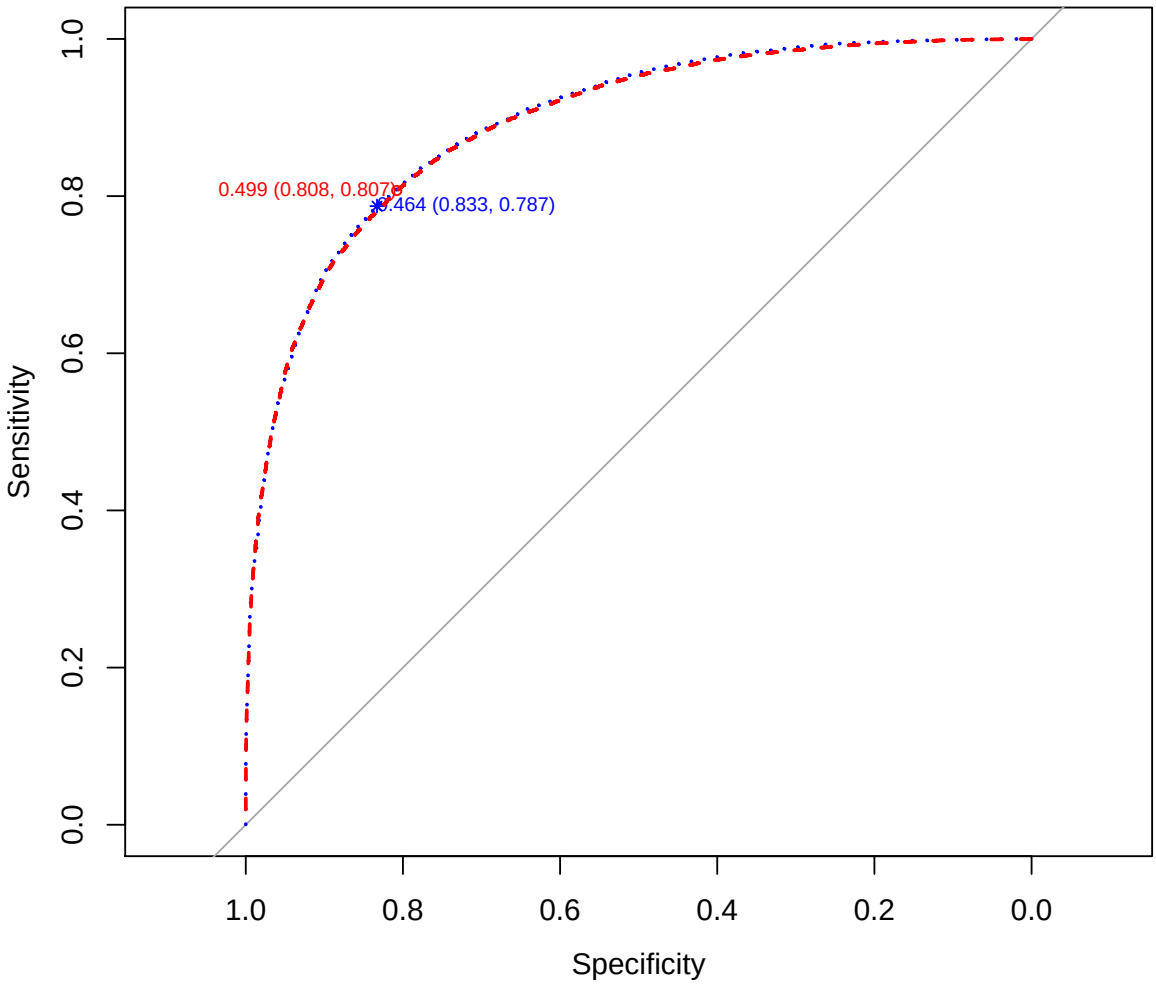


Figure 9: ROC for Logistic (Red-Dashed) and ROC for Boosted Decision Tree (Blue-Dotted) Models

Table 13: Summary of Model Performance

| | Model | |
|---|---|---|
| | Logistic | Boosted_Decision_Tree |
| **Accuracy Tests** | | |
| Accuracy | 0.8072 | 0.8084 |
| Kappa | 0.5217 | 0.5244 |
| AccuracyLower | 0.7999 | 0.8012 |
| AccuracyUpper | 0.8144 | 0.8156 |
| AccuracyNull | 0.7809 | 0.7809 |
| AccuracyPValue | 0.0000 | 0.0000 |
| McnemarPValue | 0.0000 | 0.0000 |
| **Other Metrics** | | |
| Sensitivity | 0.8071 | 0.8090 |
| Specificity | 0.8073 | 0.8083 |
| Pos Pred Value | 0.5402 | 0.5421 |
| Neg Pred Value | 0.9372 | 0.9378 |
| Precision | 0.5402 | 0.5421 |
| Recall | 0.8071 | 0.8090 |
| F1 | 0.6472 | 0.6492 |
| Prevalence | 0.2191 | 0.2191 |
| Detection Rate | 0.1768 | 0.1773 |
| Detection Prevalence | 0.3273 | 0.3270 |
| Balanced Accuracy | 0.8072 | 0.8087 |

high (approx. 94%), meaning there is only about a 6% chance of a negative prediction (no rain tomorrow) being wrong. Model performance statistics suggest that the Boosted Decision Tree model should be preferred. However, as it only marginally out performs the Logistic model, it is conceivable to prefer the Logistic model if interpretability of results is important.

# References


Friedman, Jerome H. 2002. "Stochastic Gradient Boosting." *Comput. Stat. Data Anal.* 38 (4). Amsterdam, The Netherlands, The Netherlands: Elsevier Science Publishers B. V.: 367–78. doi:10.1016/S0167-9473(01)00065-2.

Greenwell, Brandon, Bradley Boehmke, Jay Cunningham, and GBM Developers. 2018. *Gbm: Generalized Boosted Regression Models*. https://CRAN.R-project.org/package=gbm.

H2O.ai. n.d. "Regularization." http://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/glm.html#regularization.

Khan, Shehroz. 2016. "Why Is the Boosting Algorithm Robust to Overfitting?" https://www.quora.com/Why-is-the-boosting-algorithm-robust-to-overfitting.

Kuhn, Max. 2018. *Caret: Classification and Regression Training*. https://CRAN.R-project.org/package=caret.

Kun, Jeremy. 2015. "The Boosting Margin, or Why Boosting Doesn't Overfit." https://jeremykun.com/2015/09/21/the-boosting-margin-or-why-boosting-doesnt-overfit/.

Moise, Remi. 2017. "What Is the Difference Between Gradient Boosting and Adaboost?" https://www.quora.com/What-is-the-difference-between-gradient-boosting-and-adaboost.

Parkes, Shea. 2012. "Does Ensembling (Boosting) Cause Overfitting?" https://stats.stackexchange.com/

questions/20714/does-ensembling-boosting-cause-overfitting.

Ricaud, Benjamin. 2017. “A Simple Explanation of Entropy in Decision Trees.” https://bricaud.github.io/personal-blog/entropy-in-decision-trees/.

Woodruff, Katherine. 2017. “Introduction to Boosted Decision Trees.” https://indico.fnal.gov/event/15356/contribution/1/material/slides/0.pdf.