Classification Tree - Brisbane

Kathryn Weissman

1/4/2022

Classification Tree: Brisbane

The goal is to predict if there will be rain the following day.

```
# The random seed must be set before each call to a function that uses random.
set.seed(1234) # for reproducibility of results
```

Load Train & Test Data

I am loading the same data that was used for the LDA modelling.

```
# Load the data
Btrain <- read.csv("Train_Test_CSVs/df_Brisbane_train.csv", stringsAsFactors = T)
Btest <- read.csv("Train_Test_CSVs/df_Brisbane_test.csv", stringsAsFactors = T)
Btrain$Date <- as.Date(Btrain$Date)
Btest$Date <- as.Date(Btest$Date)
```

Summarize Train Data

```
str(Btrain)
```

```
## 'data.frame':    1431 obs. of  30 variables:
##  $ Date         : Date, format: "2008-07-01" "2008-07-02" ...
##  $ ID           : int  84008 84009 84010 84011 84012 84013 84014 84015 84016 84017 ...
##  $ Year         : int  2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 ...
##  $ Month        : Factor w/ 12 levels "abril","agosto",..: 6 6 6 6 6 6 6 6 6 6 ...
##  $ Day          : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Location     : Factor w/ 1 level "Brisbane": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Evaporation  : num  1.4 2 5.8 1.8 2 5.2 2.4 2 1 4 ...
##  $ Sunshine     : num  9.5 9.8 9.4 1.1 0.3 6.4 1.6 0.6 11.7 10 ...
##  $ WindGustDir  : Factor w/ 17 levels "E","ENE","ESE",..: 16 15 1 13 9 2 5 17 15 15 ...
##  $ WindGustSpeed: int  26 30 22 24 37 31 17 31 43 39 ...
##  $ WindDir9am   : Factor w/ 18 levels "calm","E","ENE",..: 13 16 14 14 10 13 14 14 18 18 ...
##  $ WindDir3pm   : Factor w/ 18 levels "calm","E","ENE",..: 16 17 2 12 13 2 6 9 16 18 ...
##  $ WindSpeed9am : int  6 15 7 9 11 7 9 7 13 17 ...
##  $ WindSpeed3pm : int  15 19 15 7 7 17 9 2 19 17 ...
##  $ Humidity9am  : int  81 41 55 76 81 78 83 94 48 47 ...
##  $ Humidity3pm  : int  37 30 52 53 89 52 63 73 34 34 ...
##  $ Pressure9am  : num  1020 1019 1021 1024 1027 ...
##  $ Pressure3pm  : num  1015 1015 1019 1022 1026 ...
##  $ Cloud9am     : int  0 0 1 7 7 5 7 8 1 0 ...
##  $ Cloud3pm     : int  1 0 4 7 8 2 7 6 1 1 ...
##  $ Temp9am      : num  14.9 16.2 15.4 14.1 16.1 15.9 14.7 14.7 12.4 13.2 ...
```

```
##  $ Temp3pm      : num  24.6 22.4 21.3 19.6 15 20 20.1 17 16.2 18.8 ...
##  $ RainToday    : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 2 1 2 2 1 ...
##  $ RainTomorrow : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 1 2 2 1 1 ...
##  $ TempRange    : num  14.9 14.8 12.7 8.2 4.4 8.9 9 2.9 10.7 9.4 ...
##  $ MaxTemp      : num  25.2 22.9 22.4 20 16.7 21.5 21.9 17.4 16.7 19.7 ...
##  $ MinTemp      : num  10.3 8.1 9.7 11.8 12.3 12.6 12.9 14.5 6 10.3 ...
##  $ Rainfall     : num  0 0 0 0.8 0 16.2 0 24.2 3.8 0.2 ...
##  $ monthID      : Factor w/ 47 levels "2008-agosto",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ Season       : Factor w/ 4 levels "autumn","spring",..: 4 4 4 4 4 4 4 4 4 4 ...
```

```
summary(Btrain)
```

```
##       Date                   ID             Year            Month
##  Min.   :2008-07-01   Min.   :84008   Min.   :2008   agosto   :124
##  1st Qu.:2009-06-23   1st Qu.:84358   1st Qu.:2009   diciembre:124
##  Median :2010-06-16   Median :84708   Median :2010   enero    :124
##  Mean   :2010-06-16   Mean   :84708   Mean   :2010   julio    :124
##  3rd Qu.:2011-06-08   3rd Qu.:85058   3rd Qu.:2011   marzo    :124
##  Max.   :2012-05-31   Max.   :85408   Max.   :2012   mayo     :124
##                       NA's   :30                     (Other)  :687
##       Day            Location     Evaporation        Sunshine
##  Min.   : 1.00   Brisbane:1431   Min.   : 0.000   Min.   : 0.000
##  1st Qu.: 8.00                   1st Qu.: 3.200   1st Qu.: 5.300
##  Median :16.00                   Median : 4.800   Median : 9.200
##  Mean   :15.73                   Mean   : 5.032   Mean   : 7.757
##  3rd Qu.:23.00                   3rd Qu.: 6.800   3rd Qu.:10.500
##  Max.   :31.00                   Max.   :13.800   Max.   :13.500
##
##   WindGustDir  WindGustSpeed     WindDir9am    WindDir3pm   WindSpeed9am
##  E      :246   Min.   :13.00   SW     :315   ENE    :238   Min.   : 0.000
##  ENE    :185   1st Qu.:22.00   SSW    :152   NE     :202   1st Qu.: 4.000
##  W      :168   Median :28.00   WSW    :146   E      :183   Median : 7.000
##  SE     :133   Mean   :29.14   SE     :106   ESE    :170   Mean   : 7.593
##  ESE    :129   3rd Qu.:33.00   SSE    : 95   NNE    :114   3rd Qu.: 9.000
##  NE     :125   Max.   :93.00   W      : 92   W      :111   Max.   :26.000
##  (Other):445                   (Other):525   (Other):413
##   WindSpeed3pm    Humidity9am     Humidity3pm     Pressure9am
##  Min.   : 0.00   Min.   :20.00   Min.   : 8.00   Min.   : 997.8
##  1st Qu.: 9.00   1st Qu.:57.00   1st Qu.:46.00   1st Qu.:1014.1
##  Median :11.00   Median :64.00   Median :54.00   Median :1017.8
##  Mean   :11.56   Mean   :64.94   Mean   :54.83   Mean   :1017.8
##  3rd Qu.:15.00   3rd Qu.:73.00   3rd Qu.:63.00   3rd Qu.:1021.6
##  Max.   :28.00   Max.   :98.00   Max.   :98.00   Max.   :1031.6
##
##   Pressure3pm        Cloud9am        Cloud3pm        Temp9am
##  Min.   : 993.2   Min.   :0.000   Min.   :0.000   Min.   :10.70
##  1st Qu.:1011.1   1st Qu.:1.000   1st Qu.:1.000   1st Qu.:17.90
##  Median :1014.9   Median :4.000   Median :4.000   Median :22.50
##  Mean   :1014.7   Mean   :4.126   Mean   :4.134   Mean   :21.70
##  3rd Qu.:1018.5   3rd Qu.:7.000   3rd Qu.:7.000   3rd Qu.:25.65
##  Max.   :1028.8   Max.   :8.000   Max.   :8.000   Max.   :31.60
##
##     Temp3pm      RainToday  RainTomorrow   TempRange         MaxTemp
##  Min.   :12.00   No :1064   No :1064     Min.   : 0.400   Min.   :12.60
##  1st Qu.:21.60   Yes: 367   Yes: 367     1st Qu.: 7.800   1st Qu.:23.10
```

```
##  Median :24.60                          Median : 9.700   Median :26.40
##  Mean   :24.53                          Mean   : 9.777   Mean   :26.16
##  3rd Qu.:27.40                          3rd Qu.:11.700   3rd Qu.:29.05
##  Max.   :36.30                          Max.   :22.000   Max.   :37.10
##
##     MinTemp         Rainfall                monthID        Season
##  Min.   : 3.70   Min.   :  0.000   2008-agosto   :  31   autumn:368
##  1st Qu.:12.60   1st Qu.:  0.000   2008-diciembre:  31   spring:364
##  Median :17.10   Median :  0.000   2008-julio    :  31   summer:361
##  Mean   :16.39   Mean   :  3.758   2008-octubre  :  31   winter:338
##  3rd Qu.:20.40   3rd Qu.:  1.200   2009-agosto   :  31
##  Max.   :25.20   Max.   :168.400   2009-diciembre:  31
##                                    (Other)       :1245
```

Summarize Test Data

```
summary(Btest)
```

```
##       Date                   ID             Year            Month
##  Min.   :2012-06-01   Min.   :85409   Min.   :2012   agosto   : 31
##  1st Qu.:2012-08-31   1st Qu.:85486   1st Qu.:2012   diciembre: 31
##  Median :2012-11-30   Median :85562   Median :2012   enero    : 31
##  Mean   :2012-11-30   Mean   :85562   Mean   :2012   julio    : 31
##  3rd Qu.:2013-03-01   3rd Qu.:85638   3rd Qu.:2013   junio    : 31
##  Max.   :2013-06-01   Max.   :85715   Max.   :2013   marzo    : 31
##                       NA's   :59                     (Other)  :180
##       Day            Location    Evaporation        Sunshine       WindGustDir
##  Min.   : 1.00   Brisbane:366   Min.   : 0.000   Min.   : 0.000   unkn   : 63
##  1st Qu.: 8.00                  1st Qu.: 3.400   1st Qu.: 6.100   E      : 51
##  Median :16.00                  Median : 5.000   Median : 9.000   NE     : 39
##  Mean   :15.68                  Mean   : 5.203   Mean   : 7.898   ENE    : 38
##  3rd Qu.:23.00                  3rd Qu.: 7.000   3rd Qu.:10.400   W      : 34
##  Max.   :31.00                  Max.   :11.800   Max.   :13.200   SE     : 30
##                                                                   (Other):111
##  WindGustSpeed     WindDir9am    WindDir3pm   WindSpeed9am     WindSpeed3pm
##  Min.   :13.00   SW     : 79   unkn   : 59   Min.   : 0.000   Min.   : 0.00
##  1st Qu.:22.00   unkn   : 59   NE     : 46   1st Qu.: 4.000   1st Qu.: 7.00
##  Median :26.00   WSW    : 40   ENE    : 45   Median : 6.000   Median : 9.00
##  Mean   :28.08   SSW    : 28   E      : 41   Mean   : 6.434   Mean   :10.19
##  3rd Qu.:31.00   W      : 19   ESE    : 36   3rd Qu.: 7.000   3rd Qu.:13.00
##  Max.   :70.00   ENE    : 17   NNE    : 27   Max.   :37.000   Max.   :24.00
##                  (Other):124   (Other):112
##   Humidity9am    Humidity3pm     Pressure9am      Pressure3pm
##  Min.   :25.0   Min.   :16.00   Min.   : 998.4   Min.   : 998.1
##  1st Qu.:57.0   1st Qu.:47.00   1st Qu.:1014.4   1st Qu.:1010.9
##  Median :65.0   Median :55.00   Median :1018.1   Median :1015.0
##  Mean   :65.8   Mean   :55.66   Mean   :1018.0   Mean   :1014.8
##  3rd Qu.:74.0   3rd Qu.:63.00   3rd Qu.:1022.0   3rd Qu.:1018.5
##  Max.   :98.0   Max.   :98.00   Max.   :1030.4   Max.   :1027.9
##
##     Cloud9am        Cloud3pm        Temp9am         Temp3pm      RainToday
##  Min.   :0.000   Min.   :0.000   Min.   :10.60   Min.   :14.10   No :280
##  1st Qu.:1.000   1st Qu.:1.000   1st Qu.:17.43   1st Qu.:21.73   Yes: 86
##  Median :4.000   Median :4.000   Median :22.00   Median :24.50
```

```
##  Mean   :3.956   Mean   :4.044   Mean   :21.55   Mean   :24.42
##  3rd Qu.:7.000   3rd Qu.:7.000   3rd Qu.:25.48   3rd Qu.:26.90
##  Max.   :8.000   Max.   :8.000   Max.   :31.90   Max.   :33.70
##
##  RainTomorrow   TempRange        MaxTemp         MinTemp         Rainfall
##  No :282      Min.   : 2.00   Min.   :15.10   Min.   : 4.10   Min.   :  0.000
##  Yes: 84      1st Qu.: 7.80   1st Qu.:23.20   1st Qu.:12.72   1st Qu.:  0.000
##               Median : 9.90   Median :26.60   Median :16.40   Median :  0.000
##               Mean   :10.09   Mean   :26.22   Mean   :16.13   Mean   :  3.377
##               3rd Qu.:12.30   3rd Qu.:29.20   3rd Qu.:20.00   3rd Qu.:  0.600
##               Max.   :18.20   Max.   :37.90   Max.   :26.10   Max.   :145.000
##
##           monthID       Season
##  2012-agosto   : 31   autumn:92
##  2012-diciembre: 31   spring:91
##  2012-julio    : 31   summer:90
##  2012-octubre  : 31   winter:93
##  2013-enero    : 31
##  2013-marzo    : 31
##  (Other)       :180
```

Compare Target Variables for Train and Test Data

It is important that our training data and testing data have similar characteristics in order to optimize the performance of our model. The test set has a slightly lower percentage of rainy days than the training set.

```
print ("Percentage of Days with Rain Tomorrow in Train Data")
```

```
## [1] "Percentage of Days with Rain Tomorrow in Train Data"
```

```
round(prop.table(table(Btrain$RainTomorrow))*100,1)
```

```
##
##   No  Yes
## 74.4 25.6
```

```
print ("Percentage of Days with Rain Tomorrow in Test Data")
```

```
## [1] "Percentage of Days with Rain Tomorrow in Test Data"
```

```
round(prop.table(table(Btest$RainTomorrow))*100,1)
```

```
##
##  No Yes
##  77  23
```

The seasons are mostly balanced between the training and testing data. The testing data has a slightly larger proportion of winter days. This is due to the span of dates in the training data not including one of the full years. The training data spans from July 1, 2008 to May 31, 2012. We don't have data for the month of June 2008 to include in the training set.

```
print ("Percentage of Days in each Season in Train Data")
```

```
## [1] "Percentage of Days in each Season in Train Data"
```

```
round(prop.table(table(Btrain$Season))*100,1)
```

```
##
## autumn spring summer winter
```

```
##   25.7   25.4   25.2   23.6
```

```
print ("Percentage of Days in each Season in Test Data")
```

```
## [1] "Percentage of Days in each Season in Test Data"
```

```
round(prop.table(table(Btest$Season))*100,1)
```

```
##
## autumn spring summer winter
##   25.1   24.9   24.6   25.4
```

Classification Tree

https://cran.r-project.org/web/packages/rpart/vignettes/longintro.pdf

"The rpart programs build classification or regression models of a very general structure using a two stage procedure; the resulting models can be represented as binary trees."

We use two different sets of modeling variables to see if there is a difference in the performance of the model for classifying whether or not there will be rain tomorrow.

```
# We use two different sets of variables for the model to consider

# Set 1 includes "RainToday" and "TempRange"
modeling_vars1 <- c("Evaporation", "Sunshine", "WindGustSpeed", "WindSpeed9am",
                    "WindSpeed3pm", "Humidity9am", "Humidity3pm", "Pressure9am",
                    "Pressure3pm", "Cloud9am", "Cloud3pm", "TempRange",
                    "RainToday", "Season", "RainTomorrow")

# Set 2 includes all temperature variables and "Rainfall" instead of "RainToday"
modeling_vars2 <- c("Evaporation", "Sunshine", "WindGustSpeed", "WindSpeed9am",
                    "WindSpeed3pm", "Humidity9am", "Humidity3pm", "Pressure9am",
                    "Pressure3pm",  "Cloud9am", "Cloud3pm", "Temp9am", "Temp3pm",
                    "TempRange", "MaxTemp", "MinTemp","RainToday", "Rainfall", "Season",
                    "RainTomorrow")

train1 <- Btrain[,modeling_vars1]
test1 <- Btest[,modeling_vars1]

train2 <- Btrain[,modeling_vars2]
test2 <- Btest[,modeling_vars2]
```

SMOTE algorithm for unbalanced classification problems

From the library {performanceEstimation}

"This function handles unbalanced classification problems using the SMOTE method. Namely, it can generate a new"SMOTEd" data set that addresses the class unbalance problem."

Balanced Training Sets 1 and 2 have different observations due to the nearest neighbors defined by the subset of variables contained in each training data set.

```
set.seed(1234) # for reproducibility of results
# Create balanced training data sets
trainBal1 <- smote(RainTomorrow ~., train1, perc.over = 2, k = 5, perc.under = 2)
trainBal2 <- smote(RainTomorrow ~., train2, perc.over = 2, k = 5, perc.under = 2)
```

```
print("Training Data: Count of Rain Tomorrow")
```

```
## [1] "Training Data: Count of Rain Tomorrow"
```

```
(table(Btrain$RainTomorrow))
```

```
##
##   No  Yes
## 1064  367
```

```
print("Balanced Training 1 Data: Count of Rain Tomorrow")
```

```
## [1] "Balanced Training 1 Data: Count of Rain Tomorrow"
```

```
(table(trainBal1$RainTomorrow))
```

```
##
##   No  Yes
## 1468 1101
```

```
print("Balanced Training 1 Data: Percent of Days with Rain Tomorrow")
```

```
## [1] "Balanced Training 1 Data: Percent of Days with Rain Tomorrow"
```

```
round(prop.table((table(trainBal1$RainTomorrow)))*100,2)
```

```
##
##    No   Yes
## 57.14 42.86
```

```
print("Balanced Training 2 Data: Count of Rain Tomorrow")
```

```
## [1] "Balanced Training 2 Data: Count of Rain Tomorrow"
```

```
(table(trainBal2$RainTomorrow))
```

```
##
##   No  Yes
## 1468 1101
```

```
print("Balanced Training 2 Data: Percent of Days with Rain Tomorrow")
```

```
## [1] "Balanced Training 2 Data: Percent of Days with Rain Tomorrow"
```

```
round(prop.table(table(trainBal2$RainTomorrow))*100,2)
```

```
##
##    No   Yes
## 57.14 42.86
```

```
print("Balanced Training 1 Data: Percent of Days in each Season")
```

```
## [1] "Balanced Training 1 Data: Percent of Days in each Season"
```

```
round(prop.table(table(trainBal1$Season))*100,1)
```

```
##
## autumn spring summer winter
##   26.5   23.3   29.2   21.0
```

```
print("Balanced Training 2 Data: Percent of Days in each Season")
```

```
## [1] "Balanced Training 2 Data: Percent of Days in each Season"
```

```
round(prop.table(table(trainBal2$Season))*100,1)
```

```
##
## autumn spring summer winter
##   26.7   24.6   28.2   20.6
```

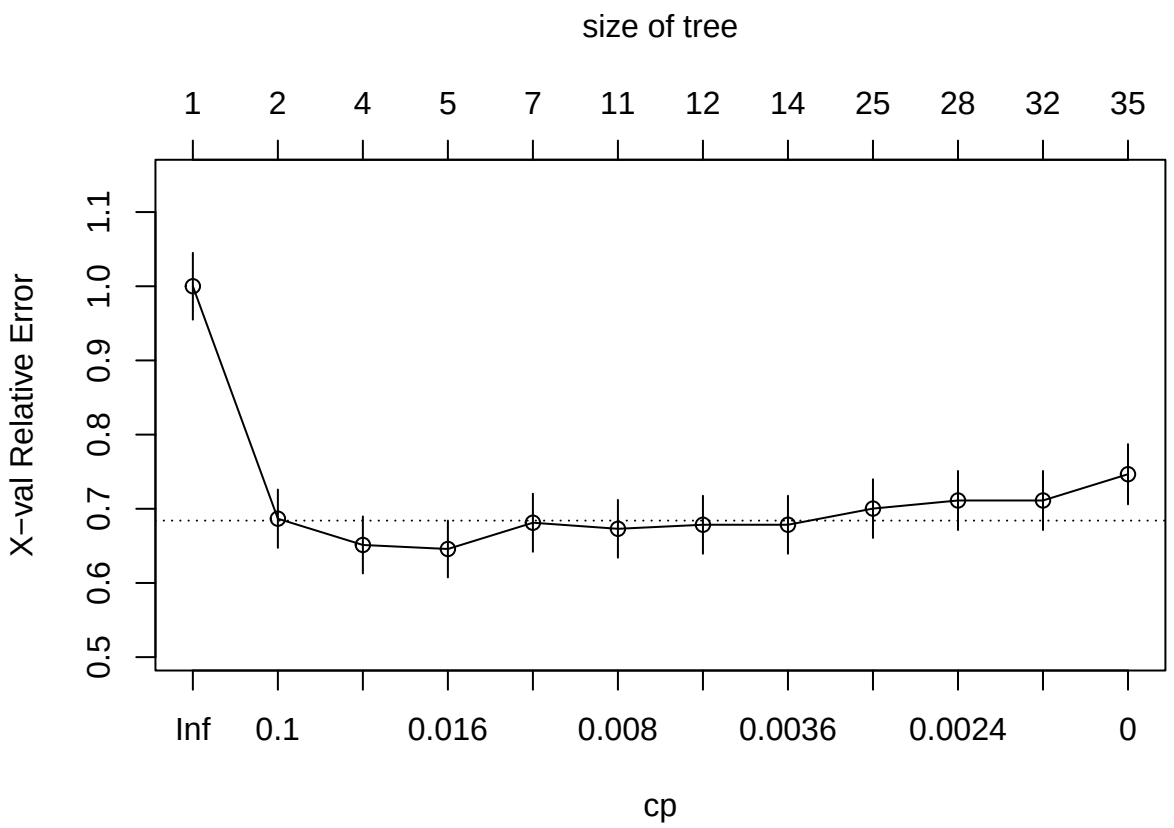
Using Fitting & Pruning Strategy shown in Lab

```
# Best strategy for tree fitting, start with cp = 0, then prune.
set.seed(1234) # for reproducibility of results
treeFit1 <- rpart(RainTomorrow ~., data = train1, method = "class", cp = 0)
printcp(treeFit1)
```

First Set of Variables on Unbalanced Data

```
##
## Classification tree:
## rpart(formula = RainTomorrow ~ ., data = train1, method = "class",
##     cp = 0)
##
## Variables actually used in tree construction:
##  [1] Cloud3pm      Cloud9am      Evaporation   Humidity3pm   Humidity9am
##  [6] Pressure3pm   Pressure9am   RainToday     Season        Sunshine
## [11] TempRange     WindGustSpeed WindSpeed9am
##
## Root node error: 367/1431 = 0.25646
##
## n= 1431
##
##           CP nsplit rel error  xerror     xstd
## 1  0.3324251      0   1.00000 1.00000 0.045011
## 2  0.0313351      1   0.66757 0.68665 0.039262
## 3  0.0190736      3   0.60490 0.65123 0.038446
## 4  0.0136240      4   0.58583 0.64578 0.038317
## 5  0.0118074      6   0.55858 0.68120 0.039139
## 6  0.0054496     10   0.50681 0.67302 0.038953
## 7  0.0040872     11   0.50136 0.67847 0.039077
## 8  0.0032202     13   0.49319 0.67847 0.039077
## 9  0.0027248     24   0.45777 0.70027 0.039565
## 10 0.0020436     27   0.44959 0.71117 0.039804
## 11 0.0018165     31   0.44142 0.71117 0.039804
## 12 0.0000000     34   0.43597 0.74659 0.040556
```

```
plotcp(treeFit1)
```




```
#rpart.plot(treeFit1)
```

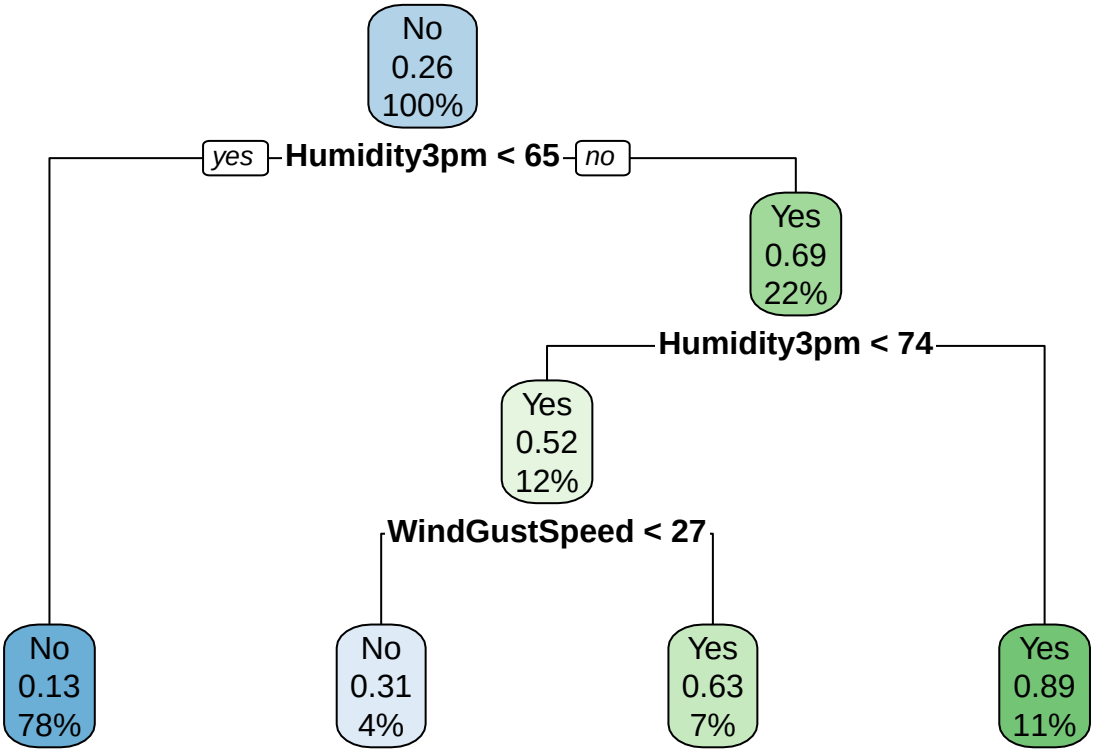
```
# Find the cp with lowest error, then prune.
xerror <- treeFit1$cptable[,"xerror"]
imin.xerror <- which.min(xerror)
treeFit1$cptable[imin.xerror, ]
```

```
##         CP     nsplit  rel error     xerror       xstd
## 0.01362398 4.00000000 0.58583106 0.64577657 0.03831691
```

```
upper.xerror <- xerror[imin.xerror] + treeFit1$cptable[imin.xerror, "xstd"]
icp <- min(which(xerror <= upper.xerror))
cp <- treeFit1$cptable[icp, "CP"]
```

The pruned tree produced using imbalanced training data on the first set of variables is extremely simple, and only uses two variables, Humidity3pm and WindGustSpeed.

```
# prune using cp
tree1 <- prune(treeFit1, cp = cp)
rpart.plot(tree1)
```




```
#Checking important variables
importance1 <- tree1$variable.importance
importance1 <- round(100*importance1/sum(importance1), 1)
importance1[importance1 >= 1]
```

```
##   Humidity3pm   Humidity9am      Sunshine     TempRange      Cloud3pm
##          40.0          12.4          11.9          11.8          10.8
##      Cloud9am  WindSpeed3pm WindGustSpeed
##           8.7           1.9           1.8
```

Confusion Matrix Help for Confusion Matrix: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Recall,Precision and Accuracy should be high as possible

Balanced Accuracy represents area under ROC.

Although the accuracy is fairly high, 84%, the sensitivity is low, below 60%, which is how well the model predicts it will rain on a rainy day. Since the data is imbalanced, we should try using SMOTE sampling for the training data to see if it improves the performance of the model.

```
#Evaluation
#Confusion matrix-train
pred_train1 <- predict(tree1, train1, type = 'class') # using train data
#Make sure to state positive class in the confusion matrix.
confusionMatrix(pred_train1, train1$RainTomorrow, positive="Yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  1009  167
```

```
##        Yes   55  200
##
##                Accuracy : 0.8449
##                  95% CI : (0.8251, 0.8632)
##     No Information Rate : 0.7435
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.548
##
##  Mcnemar's Test P-Value : 9.346e-14
##
##             Sensitivity : 0.5450
##             Specificity : 0.9483
##          Pos Pred Value : 0.7843
##          Neg Pred Value : 0.8580
##              Prevalence : 0.2565
##          Detection Rate : 0.1398
##    Detection Prevalence : 0.1782
##       Balanced Accuracy : 0.7466
##
##        'Positive' Class : Yes
##
```

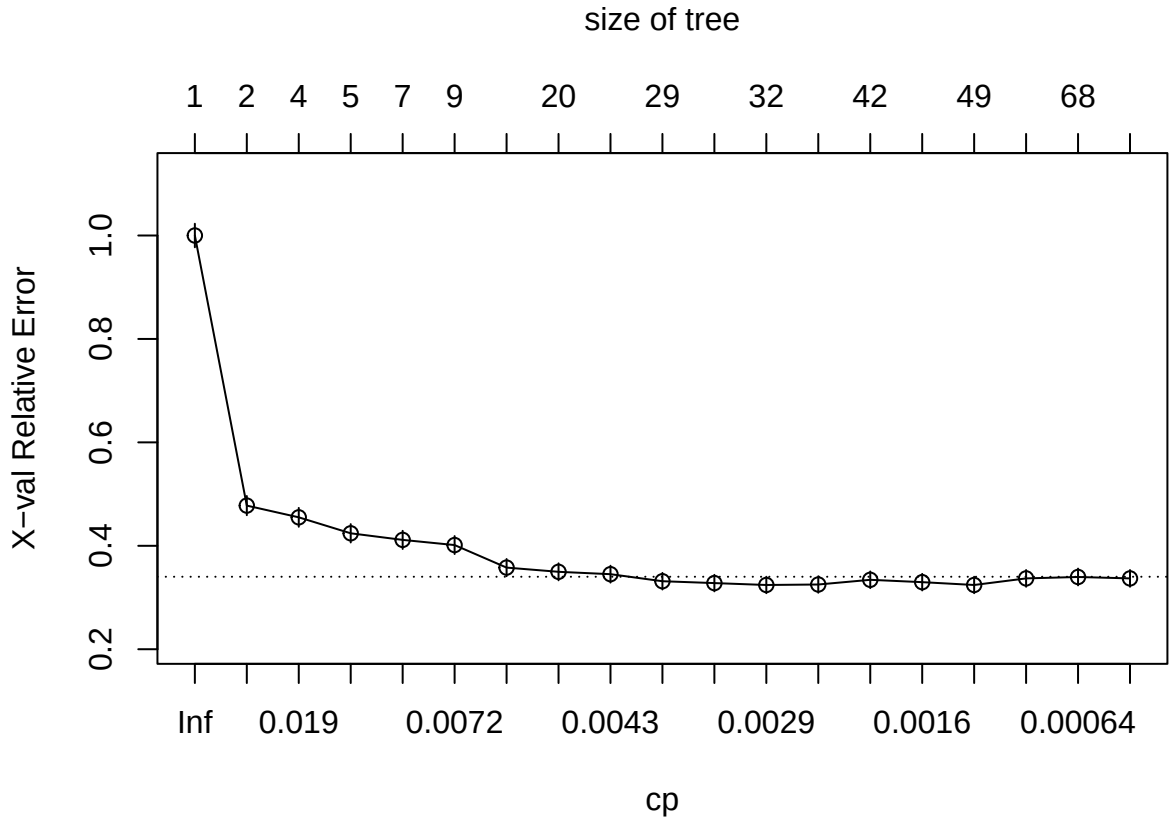
```
# Best strategy for tree fitting, start with cp = 0, then prune.
set.seed(1234) # for reproducibility of results
treeFitBal1 <- rpart(RainTomorrow ~., data = trainBal1, method = "class", cp = 0)
printcp(treeFitBal1)
```

First Set of Variables on Balnced Training Data using SMOTE

```
##
## Classification tree:
## rpart(formula = RainTomorrow ~ ., data = trainBal1, method = "class",
##     cp = 0)
##
## Variables actually used in tree construction:
##  [1] Cloud3pm      Cloud9am      Evaporation   Humidity3pm   Humidity9am
##  [6] Pressure3pm   Pressure9am   Season        Sunshine      TempRange
## [11] WindGustSpeed WindSpeed3pm  WindSpeed9am
##
## Root node error: 1101/2569 = 0.42857
##
## n= 2569
##
##            CP nsplit rel error  xerror     xstd
## 1  0.52316076      0   1.00000 1.00000 0.022782
## 2  0.02134423      1   0.47684 0.47775 0.018576
## 3  0.01725704      3   0.43415 0.45504 0.018240
## 4  0.00862852      4   0.41689 0.42416 0.017754
## 5  0.00817439      6   0.39964 0.41144 0.017544
## 6  0.00635786      8   0.38329 0.40145 0.017375
## 7  0.00544959     16   0.31789 0.35786 0.016589
## 8  0.00454133     19   0.30154 0.34968 0.016432
## 9  0.00408719     21   0.29246 0.34514 0.016344
```

```
## 10 0.00363306     28   0.25613 0.33152 0.016072
## 11 0.00317893     29   0.25250 0.32788 0.015999
## 12 0.00272480     31   0.24614 0.32425 0.015924
## 13 0.00242204     38   0.22707 0.32516 0.015943
## 14 0.00181653     41   0.21980 0.33424 0.016127
## 15 0.00145322     43   0.21617 0.32970 0.016036
## 16 0.00102180     48   0.20890 0.32425 0.015924
## 17 0.00090827     56   0.20073 0.33697 0.016182
## 18 0.00045413     67   0.19074 0.33969 0.016236
## 19 0.00000000     71   0.18892 0.33697 0.016182
```

```
plotcp(treeFitBal1)
```




```
#rpart.plot(treeFitBal1)
```

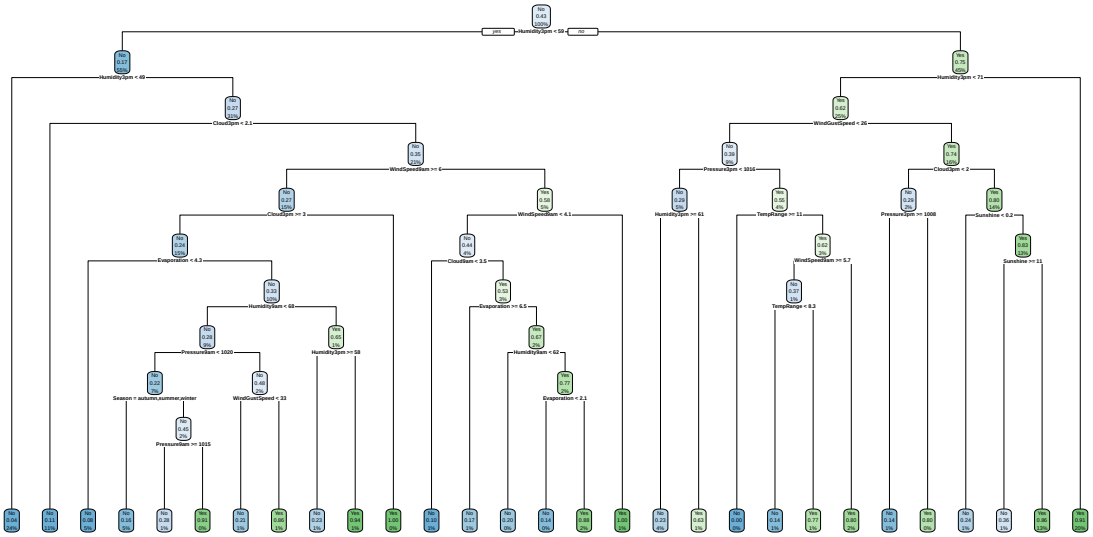
```
# Find the cp with lowest error, then prune.
xerror <- treeFitBal1$cptable[,"xerror"]
imin.xerror <- which.min(xerror)
treeFitBal1$cptable[imin.xerror, ]
```

```
##           CP       nsplit    rel error       xerror         xstd
##  0.002724796 31.000000000  0.246139873  0.324250681  0.015924188
```

```
upper.xerror <- xerror[imin.xerror] + treeFitBal1$cptable[imin.xerror, "xstd"]
icp <- min(which(xerror <= upper.xerror))
cp <- treeFitBal1$cptable[icp, "CP"]
```

The pruned tree using the balanced data is much more complex than the tree produced using imbalanced data.

```
# prune using cp
treeBal1 <- prune(treeFitBal1, cp = cp)
rpart.plot(treeBal1)
```



```
#Classification Rules
rpart.rules(treeBal1, style = "tall")
```

```
## RainTomorrow is 0.00 when
##     Humidity3pm is 59 to 71
##     WindGustSpeed < 26
##     Pressure3pm >= 1016
##     TempRange >= 10.9
##
## RainTomorrow is 0.04 when
##     Humidity3pm < 49
##
## RainTomorrow is 0.08 when
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 3.0
##     WindSpeed9am >= 6.0
##     Evaporation < 4.3
##
## RainTomorrow is 0.10 when
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 2.1
##     WindSpeed9am < 4.1
##     Cloud9am < 3.5
##
## RainTomorrow is 0.11 when
##     Humidity3pm is 49 to 59
##     Cloud3pm < 2.1
##
## RainTomorrow is 0.14 when
##     Humidity3pm is 59 to 71
##     WindSpeed9am >= 5.7
##     WindGustSpeed < 26
##     Pressure3pm >= 1016
```

```
##     TempRange < 8.3
##
## RainTomorrow is 0.14 when
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 2.1
##     WindSpeed9am < 4.1
##     Evaporation < 2.1
##     Humidity9am >= 62
##     Cloud9am >= 3.5
##
## RainTomorrow is 0.14 when
##     Humidity3pm is 59 to 71
##     Cloud3pm < 2.0
##     WindGustSpeed >= 26
##     Pressure3pm >= 1008
##
## RainTomorrow is 0.16 when
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 3.0
##     WindSpeed9am >= 6.0
##     Evaporation >= 4.3
##     Humidity9am < 68
##     Pressure9am < 1020
##     Season is autumn or summer or winter
##
## RainTomorrow is 0.17 when
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 2.1
##     WindSpeed9am < 4.1
##     Evaporation >= 6.5
##     Cloud9am >= 3.5
##
## RainTomorrow is 0.20 when
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 2.1
##     WindSpeed9am < 4.1
##     Evaporation < 6.5
##     Humidity9am < 62
##     Cloud9am >= 3.5
##
## RainTomorrow is 0.21 when
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 3.0
##     WindSpeed9am >= 6.0
##     WindGustSpeed < 33
##     Evaporation >= 4.3
##     Humidity9am < 68
##     Pressure9am >= 1020
##
## RainTomorrow is 0.23 when
##     Humidity3pm is 61 to 71
##     WindGustSpeed < 26
##     Pressure3pm < 1016
##
```

```
## RainTomorrow is 0.23 when
##     Humidity3pm is 58 to 59
##     Cloud3pm >= 3.0
##     WindSpeed9am >= 6.0
##     Evaporation >= 4.3
##     Humidity9am >= 68
##
## RainTomorrow is 0.24 when
##     Humidity3pm is 59 to 71
##     Cloud3pm >= 2.0
##     WindGustSpeed >= 26
##     Sunshine < 0.2
##
## RainTomorrow is 0.28 when
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 3.0
##     WindSpeed9am >= 6.0
##     Evaporation >= 4.3
##     Humidity9am < 68
##     Pressure9am is 1015 to 1020
##     Season is spring
##
## RainTomorrow is 0.36 when
##     Humidity3pm is 59 to 71
##     Cloud3pm >= 2.0
##     WindGustSpeed >= 26
##     Sunshine >= 10.9
##
## RainTomorrow is 0.63 when
##     Humidity3pm is 59 to 61
##     WindGustSpeed < 26
##     Pressure3pm < 1016
##
## RainTomorrow is 0.77 when
##     Humidity3pm is 59 to 71
##     WindSpeed9am >= 5.7
##     WindGustSpeed < 26
##     Pressure3pm >= 1016
##     TempRange is 8.3 to 10.9
##
## RainTomorrow is 0.80 when
##     Humidity3pm is 59 to 71
##     Cloud3pm < 2.0
##     WindGustSpeed >= 26
##     Pressure3pm < 1008
##
## RainTomorrow is 0.80 when
##     Humidity3pm is 59 to 71
##     WindSpeed9am < 5.7
##     WindGustSpeed < 26
##     Pressure3pm >= 1016
##     TempRange < 10.9
##
## RainTomorrow is 0.86 when
```

```
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 3.0
##     WindSpeed9am >= 6.0
##     WindGustSpeed >= 33
##     Evaporation >= 4.3
##     Humidity9am < 68
##     Pressure9am >= 1020
##
## RainTomorrow is 0.86 when
##     Humidity3pm is 59 to 71
##     Cloud3pm >= 2.0
##     WindGustSpeed >= 26
##     Sunshine is 0.2 to 10.9
##
## RainTomorrow is 0.88 when
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 2.1
##     WindSpeed9am < 4.1
##     Evaporation is 2.1 to 6.5
##     Humidity9am >= 62
##     Cloud9am >= 3.5
##
## RainTomorrow is 0.91 when
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 3.0
##     WindSpeed9am >= 6.0
##     Evaporation >= 4.3
##     Humidity9am < 68
##     Pressure9am < 1015
##     Season is spring
##
## RainTomorrow is 0.91 when
##     Humidity3pm >= 71
##
## RainTomorrow is 0.94 when
##     Humidity3pm is 49 to 58
##     Cloud3pm >= 3.0
##     WindSpeed9am >= 6.0
##     Evaporation >= 4.3
##     Humidity9am >= 68
##
## RainTomorrow is 1.00 when
##     Humidity3pm is 49 to 59
##     Cloud3pm is 2.1 to 3.0
##     WindSpeed9am >= 6.0
##
## RainTomorrow is 1.00 when
##     Humidity3pm is 49 to 59
##     Cloud3pm >= 2.1
##     WindSpeed9am is 4.1 to 6.0
```

In the Imbalanced Training Data for the first set of variables, Humidity3pm, Humidity9am, Sunshine, TempRange, Cloud3pm, Cloud9am, WindSpeed3pm, and WindGustSpeed were the only variables with important greater than 1%. Using the balanced training data spreads out the importance to more variables

including WindSpeed9am, Pressure variables, Evaporation, and Season.

```
#Checking important variables
importanceBal1 <- treeBal1$variable.importance
importanceBal1 <- round(100*importanceBal1/sum(importanceBal1), 1)
importanceBal1[importanceBal1 >= 1]
```

```
##   Humidity3pm      Cloud3pm      Sunshine     TempRange   Humidity9am
##          25.2          14.0          13.4          12.7          11.3
##      Cloud9am  WindSpeed9am WindGustSpeed   Pressure3pm   Evaporation
##          10.7           2.6           2.4           1.8           1.7
##  WindSpeed3pm   Pressure9am        Season
##           1.5           1.3           1.3
```

Using the model created by balancing the data produces better results when checking predictions on the training data. Accuracy improved from 84.5% to 86.2%. Sensitivity improved from 54.5% to 78.2%. Specificity decreased from 94.8% to 89%, but Balanced Accuracy (Area under ROC) improved from 74.6% to 83.6%

```
#Evaluation of model created with balanced data
#Confusion matrix-train
pred_trainBal1 <- predict(treeBal1, train1, type = 'class') # using original train data
#Make sure to state positive class in the confusion matrix.
confusionMatrix(pred_trainBal1, train1$RainTomorrow, positive="Yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  No Yes
##        No  947  80
##        Yes 117 287
##
##                Accuracy : 0.8623
##                  95% CI : (0.8434, 0.8798)
##     No Information Rate : 0.7435
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.6506
##
##  Mcnemar's Test P-Value : 0.01032
##
##             Sensitivity : 0.7820
##             Specificity : 0.8900
##          Pos Pred Value : 0.7104
##          Neg Pred Value : 0.9221
##              Prevalence : 0.2565
##          Detection Rate : 0.2006
##    Detection Prevalence : 0.2823
##       Balanced Accuracy : 0.8360
##
##        'Positive' Class : Yes
##
```

The default probability threshold is 50% for classification. We can use trial and error to determine if a different probability threshold improves results.

In this case, lowering the probability threshold to 30% did not improve our model evaluation metrics.

When the tree produces a probability of greater than 30% chance of rain tomorrow, predicting that it will

rain lowers the balanced accuracy of the model.

We can use the default threshold of 50%.

```
#Train Set Evaluation of Balanced Model with probabilities
#Confusion matrix-train
pred_trainBal1_prob <- predict(treeBal1, train1, type = 'prob') # using imbalanced training data
# predict rain if chance of rain is more than 30% (default is 50%)
pred_trainBal1_prob30 <- ifelse(pred_trainBal1_prob[,2]>0.3,"Yes","No")
confusionMatrix(data= as.factor(pred_trainBal1_prob30), train1$RainTomorrow, positive="Yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  No Yes
##        No  937  77
##        Yes 127 290
##
##                Accuracy : 0.8574
##                  95% CI : (0.8382, 0.8752)
##     No Information Rate : 0.7435
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.6422
##
##  Mcnemar's Test P-Value : 0.0006021
##
##             Sensitivity : 0.7902
##             Specificity : 0.8806
##          Pos Pred Value : 0.6954
##          Neg Pred Value : 0.9241
##              Prevalence : 0.2565
##          Detection Rate : 0.2027
##    Detection Prevalence : 0.2914
##       Balanced Accuracy : 0.8354
##
##        'Positive' Class : Yes
##
```

Some of our key metrics decrease slightly when expanded to the test set, which could be an indicator of overfitting to the training data, but it is not too different.

Accuracy is 80%, Sensitivity is 79.7%, Specificity is 80%, and Balanced Accuracy is 80%.

```
#Test Set Evaluation of Balanced Model 1
#Confusion matrix-test
pred_testBal1 <- predict(treeBal1, test1, type = 'class') # using testing data
confusionMatrix(pred_testBal1, test1$RainTomorrow, positive="Yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  No Yes
##        No  226  17
##        Yes  56  67
##
##                Accuracy : 0.8005
```

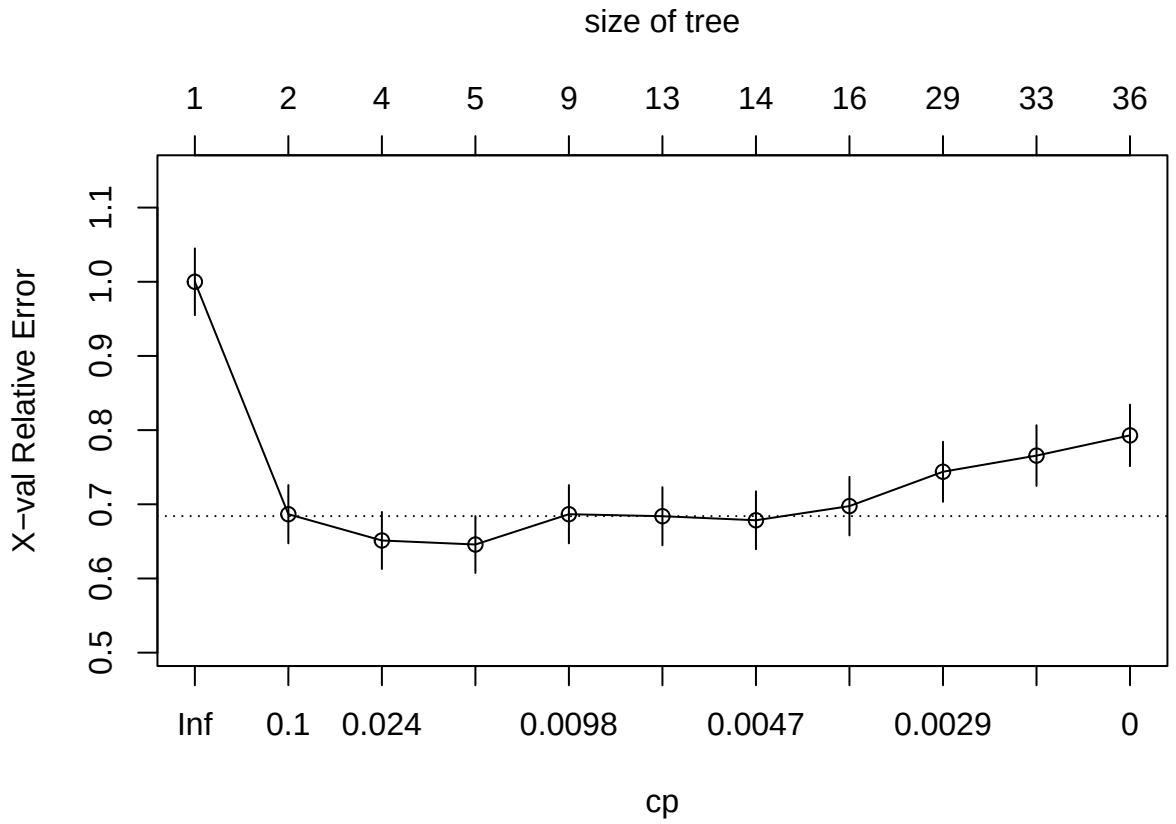
```
##                  95% CI : (0.7559, 0.8403)
##     No Information Rate : 0.7705
##     P-Value [Acc > NIR] : 0.09445
##
##                   Kappa : 0.5151
##
##  Mcnemar's Test P-Value : 8.685e-06
##
##             Sensitivity : 0.7976
##             Specificity : 0.8014
##          Pos Pred Value : 0.5447
##          Neg Pred Value : 0.9300
##              Prevalence : 0.2295
##          Detection Rate : 0.1831
##    Detection Prevalence : 0.3361
##       Balanced Accuracy : 0.7995
##
##        'Positive' Class : Yes
##
```

Second Set of Variables This set includes more variables than the first set. Set 1 included "RainToday", but set 2 also includes "Rainfall". Set 1 included "TempRange", but set 2 includes all temperature related varaiables including TempRange.

```
# Best strategy for tree fitting, cp = 0
set.seed(1234) # for reproducibility of results
treeFit2 <- rpart(RainTomorrow ~., data = train2, method = "class", cp = 0)
printcp(treeFit2)
```

```
##
## Classification tree:
## rpart(formula = RainTomorrow ~ ., data = train2, method = "class",
##     cp = 0)
##
## Variables actually used in tree construction:
##  [1] Cloud3pm      Evaporation   Humidity3pm   Humidity9am   MinTemp
##  [6] Pressure3pm   Pressure9am   Rainfall      RainToday     Sunshine
## [11] Temp3pm       TempRange     WindGustSpeed WindSpeed3pm  WindSpeed9am
##
## Root node error: 367/1431 = 0.25646
##
## n= 1431
##
##           CP nsplit rel error  xerror     xstd
## 1  0.3324251      0   1.00000 1.00000 0.045011
## 2  0.0313351      1   0.66757 0.68665 0.039262
## 3  0.0190736      3   0.60490 0.65123 0.038446
## 4  0.0118074      4   0.58583 0.64578 0.038317
## 5  0.0081744      8   0.53406 0.68665 0.039262
## 6  0.0054496     12   0.49591 0.68392 0.039201
## 7  0.0040872     13   0.49046 0.67847 0.039077
## 8  0.0031141     15   0.48229 0.69755 0.039505
## 9  0.0027248     28   0.43869 0.74387 0.040500
## 10 0.0018165     32   0.42779 0.76567 0.040946
## 11 0.0000000     35   0.42234 0.79292 0.041487
```

```
plotcp(treeFit2)
```




```
#rpart.plot(treeFit2)
```

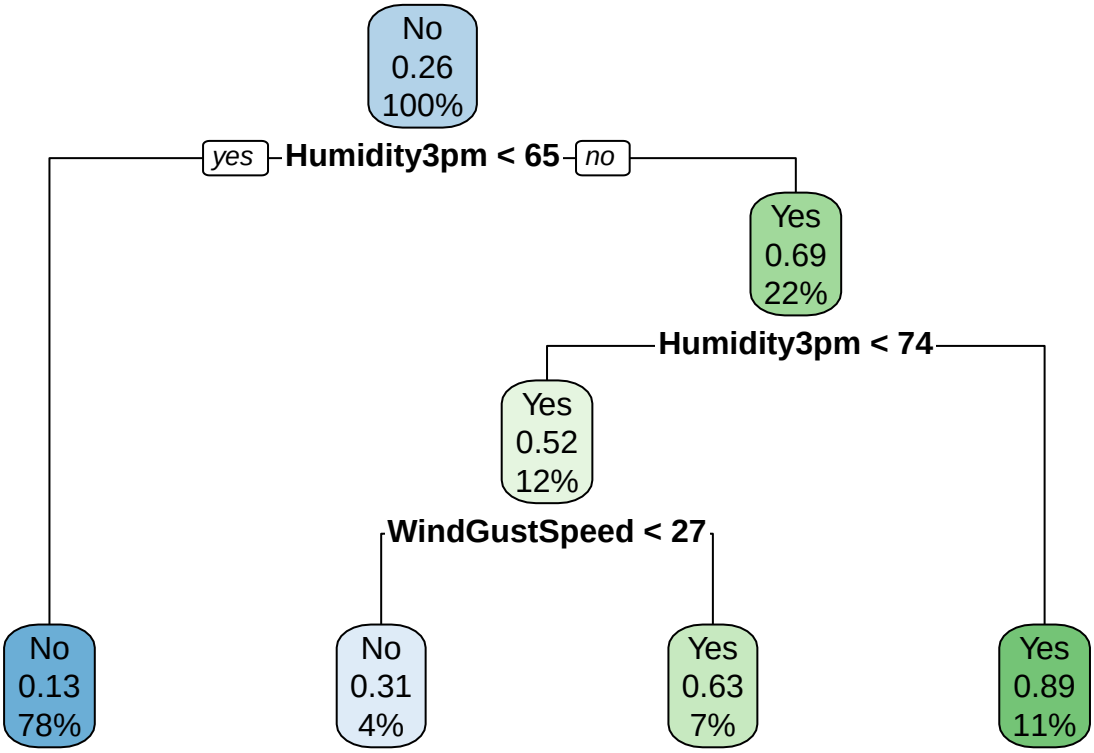
```
xerror <- treeFit2$cptable[,"xerror"]
imin.xerror <- which.min(xerror)
treeFit2$cptable[imin.xerror, ]
```

```
##         CP     nsplit  rel error     xerror       xstd
## 0.01180745 4.00000000 0.58583106 0.64577657 0.03831691
```

```
upper.xerror <- xerror[imin.xerror] + treeFit2$cptable[imin.xerror, "xstd"]
icp <- min(which(xerror <= upper.xerror))
cp <- treeFit2$cptable[icp, "CP"]
```

After pruning, the trees for both sets of variables that use the imbalanced training data are the identical for Brisbane, and they only use the Humidity3pm and WindGustSpeed variable to make a prediction.

```
tree2 <- prune(treeFit2, cp = cp)
rpart.plot(tree2)
```



The important variables are the same for tree1 and tree2, even though some different variables were added to the second training set. Humidity3pm is the most important.

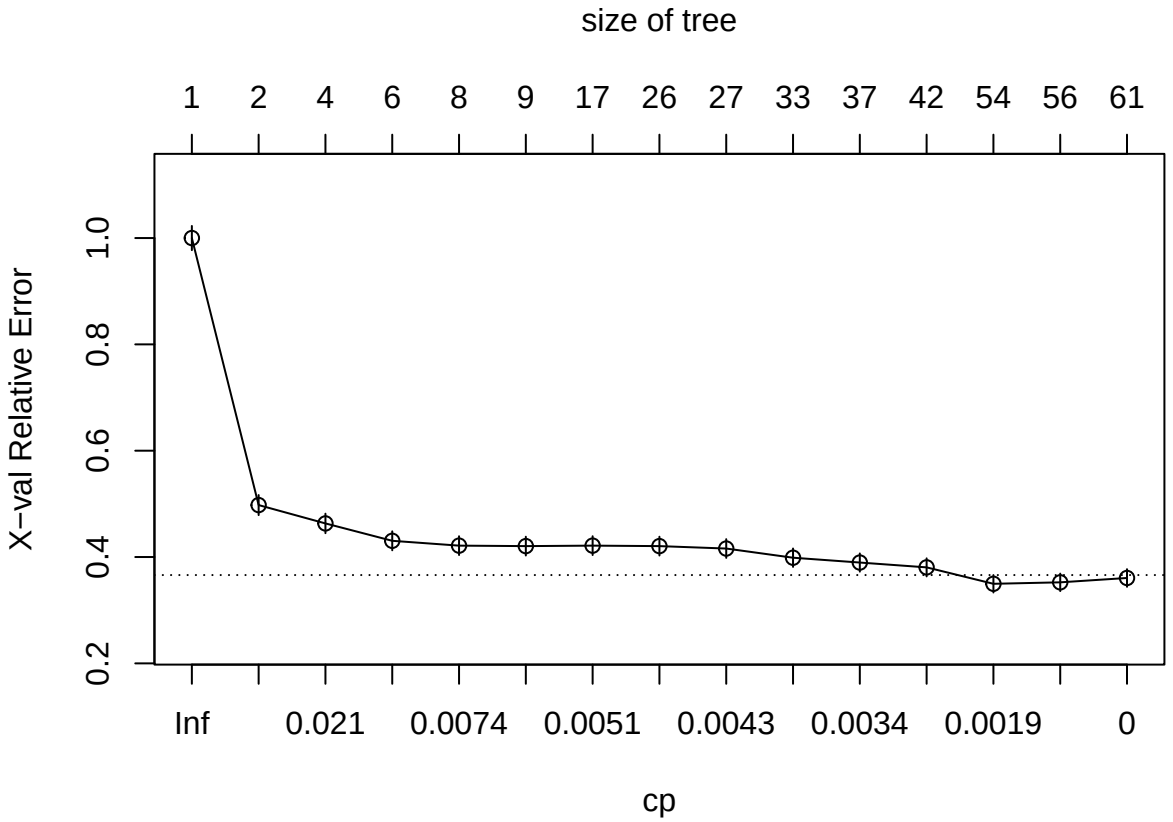
Second Set of Variables on Balnced Training Data using SMOTE The next step is to create the tree using a more balanced training set with the second set of variables to see if our model performance improves.

```
# Best strategy for tree fitting, cp = 0
set.seed(1234) # for reproducibility of results
treeBalFit2 <- rpart(RainTomorrow ~., data = trainBal2, method = "class", cp = 0)
printcp(treeBalFit2)
```

```
##
## Classification tree:
## rpart(formula = RainTomorrow ~ ., data = trainBal2, method = "class",
##     cp = 0)
##
## Variables actually used in tree construction:
##  [1] Cloud3pm      Cloud9am      Evaporation   Humidity3pm   Humidity9am
##  [6] MinTemp       Pressure3pm   Pressure9am   Rainfall      Season
## [11] Sunshine      Temp3pm       TempRange     WindGustSpeed WindSpeed3pm
## [16] WindSpeed9am
##
## Root node error: 1101/2569 = 0.42857
##
## n= 2569
##
##            CP nsplit rel error  xerror     xstd
## 1  0.50681199      0   1.00000 1.00000 0.022782
## 2  0.02861035      1   0.49319 0.49773 0.018858
## 3  0.01589464      3   0.43597 0.46322 0.018363
```

```
## 4  0.00862852      5   0.40418 0.43052 0.017857
## 5  0.00635786      7   0.38692 0.42144 0.017710
## 6  0.00514684      8   0.38056 0.42053 0.017695
## 7  0.00499546     16   0.33697 0.42144 0.017710
## 8  0.00454133     25   0.28792 0.42053 0.017695
## 9  0.00408719     26   0.28338 0.41599 0.017620
## 10 0.00363306     32   0.25704 0.39873 0.017328
## 11 0.00326975     36   0.24251 0.38965 0.017170
## 12 0.00272480     41   0.22616 0.38056 0.017008
## 13 0.00136240     53   0.19346 0.34968 0.016432
## 14 0.00090827     55   0.19074 0.35241 0.016484
## 15 0.00000000     60   0.18619 0.36058 0.016640
```

```
plotcp(treeBalFit2)
```




```
#rpart.plot(treeBalFit2)
```

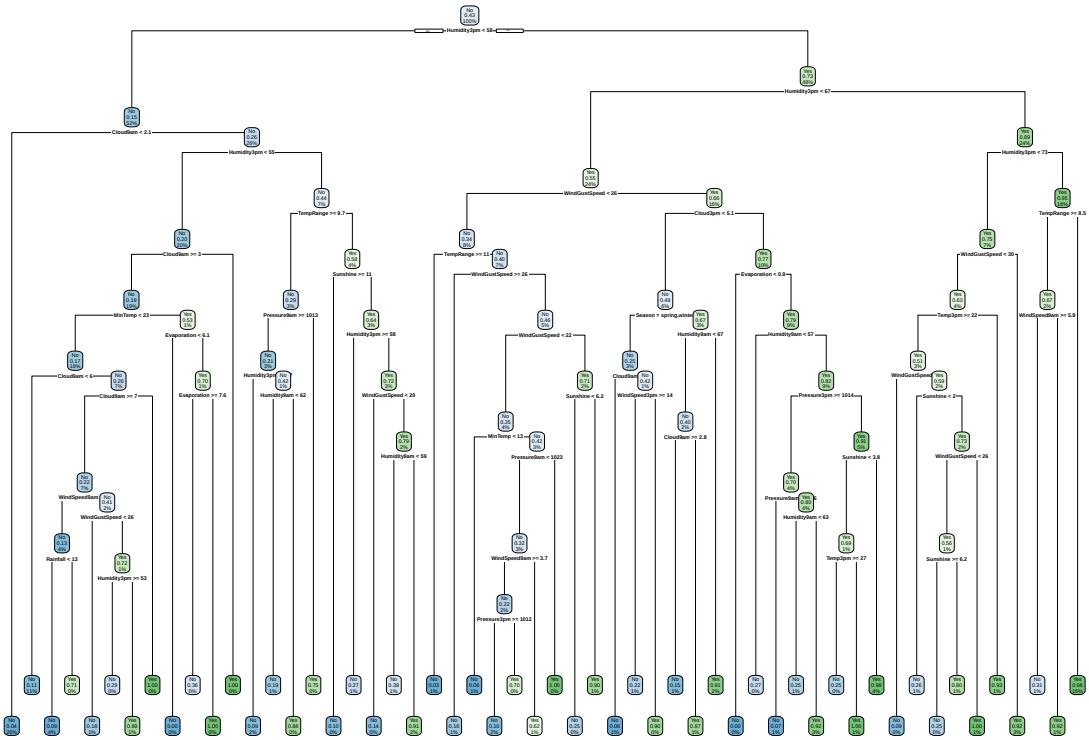
```
xerror <- treeBalFit2$cptable[,"xerror"]
imin.xerror <- which.min(xerror)
treeBalFit2$cptable[imin.xerror, ]
```

```
##           CP       nsplit    rel error       xerror         xstd
##  0.001362398 53.000000000  0.193460490  0.349682107  0.016431882
```

```
upper.xerror <- xerror[imin.xerror] + treeBalFit2$cptable[imin.xerror, "xstd"]
icp <- min(which(xerror <= upper.xerror))
cp <- treeBalFit2$cptable[icp, "CP"]
```

```
treeBal2 <- prune(treeBalFit2, cp = cp)
rpart.plot(treeBal2)
```

```
## Warning: labs do not fit even at cex 0.15, there may be some overplotting
```



```
#Classification Rules
rpart.rules(treeBal2, style = "tall")
```

```
## RainTomorrow is 0.00 when
##     Humidity3pm < 55
##     Cloud9am >= 3.0
##     MinTemp >= 23
##     Evaporation < 6.1
##
## RainTomorrow is 0.00 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud3pm >= 5.1
##     Evaporation < 0.9
##
## RainTomorrow is 0.03 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed < 26
##     TempRange >= 11.1
##
## RainTomorrow is 0.04 when
##     Humidity3pm < 58
##     Cloud9am < 2.1
##
## RainTomorrow is 0.06 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed < 22
##     TempRange < 11.1
##     MinTemp < 13
```

```
##
## RainTomorrow is 0.07 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud3pm >= 5.1
##     Humidity9am >= 57
##     Evaporation >= 0.9
##     Pressure9am < 1016
##     Pressure3pm >= 1014
##
## RainTomorrow is 0.08 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud9am < 4.0
##     Cloud3pm < 5.1
##     Season is spring or winter
##
## RainTomorrow is 0.09 when
##     Humidity3pm < 55
##     Cloud9am >= 7.0
##     MinTemp < 23
##     WindSpeed9am >= 7.0
##     Rainfall < 13
##
## RainTomorrow is 0.09 when
##     Humidity3pm is 67 to 73
##     WindGustSpeed < 20
##     Temp3pm >= 22
##
## RainTomorrow is 0.09 when
##     Humidity3pm is 57 to 58
##     Cloud9am >= 2.1
##     TempRange >= 9.7
##     Pressure9am >= 1013
##
## RainTomorrow is 0.10 when
##     Humidity3pm is 55 to 58
##     Cloud9am >= 2.1
##     TempRange < 9.7
##     Sunshine >= 10.6
##
## RainTomorrow is 0.10 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed < 22
##     TempRange < 11.1
##     MinTemp >= 13
##     Pressure9am < 1023
##     WindSpeed9am >= 3.7
##     Pressure3pm >= 1012
##
## RainTomorrow is 0.11 when
##     Humidity3pm < 55
##     Cloud9am is 3.0 to 6.0
##     MinTemp < 23
```

```
##
## RainTomorrow is 0.14 when
##     Humidity3pm is 55 to 58
##     WindGustSpeed < 20
##     Cloud9am >= 2.1
##     TempRange < 9.7
##     Sunshine < 10.6
##
## RainTomorrow is 0.15 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud9am >= 2.8
##     Cloud3pm < 5.1
##     Humidity9am < 67
##     Season is autumn or summer
##
## RainTomorrow is 0.16 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed is 26 to 26
##     TempRange < 11.1
##
## RainTomorrow is 0.18 when
##     Humidity3pm < 55
##     WindGustSpeed < 26
##     Cloud9am >= 7.0
##     MinTemp < 23
##     WindSpeed9am < 7.0
##
## RainTomorrow is 0.19 when
##     Humidity3pm is 55 to 57
##     Cloud9am >= 2.1
##     TempRange >= 9.7
##     Humidity9am < 62
##     Pressure9am >= 1013
##
## RainTomorrow is 0.22 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud9am >= 4.0
##     Cloud3pm < 5.1
##     Season is spring or winter
##     WindSpeed3pm >= 14
##
## RainTomorrow is 0.25 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed is 22 to 26
##     TempRange < 11.1
##     Sunshine < 6.2
##
## RainTomorrow is 0.25 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud3pm >= 5.1
##     Humidity9am is 57 to 63
```

```
##     Evaporation >= 0.9
##     Pressure9am >= 1016
##     Pressure3pm >= 1014
##
## RainTomorrow is 0.25 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud3pm >= 5.1
##     Sunshine < 3.8
##     Humidity9am >= 57
##     Evaporation >= 0.9
##     Pressure3pm < 1014
##     Temp3pm >= 27
##
## RainTomorrow is 0.25 when
##     Humidity3pm is 67 to 73
##     WindGustSpeed is 20 to 26
##     Sunshine >= 6.2
##     Temp3pm >= 22
##
## RainTomorrow is 0.26 when
##     Humidity3pm is 67 to 73
##     WindGustSpeed is 20 to 30
##     Sunshine < 2.0
##     Temp3pm >= 22
##
## RainTomorrow is 0.27 when
##     Humidity3pm is 58 to 58
##     Cloud9am >= 2.1
##     TempRange < 9.7
##     Sunshine < 10.6
##
## RainTomorrow is 0.27 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud3pm >= 5.1
##     Humidity9am < 57
##     Evaporation >= 0.9
##
## RainTomorrow is 0.29 when
##     Humidity3pm is 53 to 55
##     WindGustSpeed >= 26
##     Cloud9am >= 7.0
##     MinTemp < 23
##     WindSpeed9am < 7.0
##
## RainTomorrow is 0.31 when
##     Humidity3pm >= 73
##     TempRange >= 8.5
##     WindSpeed9am >= 5.9
##
## RainTomorrow is 0.36 when
##     Humidity3pm < 55
##     Cloud9am >= 3.0
```

```
##     MinTemp >= 23
##     Evaporation >= 7.6
##
## RainTomorrow is 0.38 when
##     Humidity3pm is 55 to 58
##     WindGustSpeed >= 20
##     Cloud9am >= 2.1
##     TempRange < 9.7
##     Sunshine < 10.6
##     Humidity9am < 59
##
## RainTomorrow is 0.62 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed < 22
##     TempRange < 11.1
##     MinTemp >= 13
##     Pressure9am < 1023
##     WindSpeed9am < 3.7
##
## RainTomorrow is 0.70 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed < 22
##     TempRange < 11.1
##     MinTemp >= 13
##     Pressure9am < 1023
##     WindSpeed9am >= 3.7
##     Pressure3pm < 1012
##
## RainTomorrow is 0.71 when
##     Humidity3pm < 55
##     Cloud9am >= 7.0
##     MinTemp < 23
##     WindSpeed9am >= 7.0
##     Rainfall >= 13
##
## RainTomorrow is 0.75 when
##     Humidity3pm is 55 to 58
##     Cloud9am >= 2.1
##     TempRange >= 9.7
##     Pressure9am < 1013
##
## RainTomorrow is 0.80 when
##     Humidity3pm is 67 to 73
##     WindGustSpeed is 20 to 26
##     Sunshine is 2.0 to 6.2
##     Temp3pm >= 22
##
## RainTomorrow is 0.87 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud9am < 2.8
##     Cloud3pm < 5.1
##     Humidity9am < 67
##     Season is autumn or summer
```

```
##
## RainTomorrow is 0.88 when
##     Humidity3pm is 55 to 57
##     Cloud9am >= 2.1
##     TempRange >= 9.7
##     Humidity9am >= 62
##     Pressure9am >= 1013
##
## RainTomorrow is 0.89 when
##     Humidity3pm < 53
##     WindGustSpeed >= 26
##     Cloud9am >= 7.0
##     MinTemp < 23
##     WindSpeed9am < 7.0
##
## RainTomorrow is 0.90 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed is 22 to 26
##     TempRange < 11.1
##     Sunshine >= 6.2
##
## RainTomorrow is 0.90 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud9am >= 4.0
##     Cloud3pm < 5.1
##     Season is spring or winter
##     WindSpeed3pm < 14
##
## RainTomorrow is 0.91 when
##     Humidity3pm is 55 to 58
##     WindGustSpeed >= 20
##     Cloud9am >= 2.1
##     TempRange < 9.7
##     Sunshine < 10.6
##     Humidity9am >= 59
##
## RainTomorrow is 0.91 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud3pm < 5.1
##     Humidity9am >= 67
##     Season is autumn or summer
##
## RainTomorrow is 0.92 when
##     Humidity3pm >= 73
##     TempRange >= 8.5
##     WindSpeed9am < 5.9
##
## RainTomorrow is 0.92 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud3pm >= 5.1
##     Humidity9am >= 63
```

```
##     Evaporation >= 0.9
##     Pressure9am >= 1016
##     Pressure3pm >= 1014
##
## RainTomorrow is 0.92 when
##     Humidity3pm is 67 to 73
##     WindGustSpeed >= 30
##
## RainTomorrow is 0.93 when
##     Humidity3pm is 67 to 73
##     WindGustSpeed < 30
##     Temp3pm < 22
##
## RainTomorrow is 0.98 when
##     Humidity3pm >= 73
##     TempRange < 8.5
##
## RainTomorrow is 0.98 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud3pm >= 5.1
##     Sunshine >= 3.8
##     Humidity9am >= 57
##     Evaporation >= 0.9
##     Pressure3pm < 1014
##
## RainTomorrow is 1.00 when
##     Humidity3pm < 55
##     Cloud9am is 6.0 to 7.0
##     MinTemp < 23
##
## RainTomorrow is 1.00 when
##     Humidity3pm < 55
##     Cloud9am >= 3.0
##     MinTemp >= 23
##     Evaporation is 6.1 to 7.6
##
## RainTomorrow is 1.00 when
##     Humidity3pm < 55
##     Cloud9am is 2.1 to 3.0
##
## RainTomorrow is 1.00 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed < 22
##     TempRange < 11.1
##     MinTemp >= 13
##     Pressure9am >= 1023
##
## RainTomorrow is 1.00 when
##     Humidity3pm is 58 to 67
##     WindGustSpeed >= 26
##     Cloud3pm >= 5.1
##     Sunshine < 3.8
##     Humidity9am >= 57
```

```
##     Evaporation >= 0.9
##     Pressure3pm < 1014
##     Temp3pm < 27
##
## RainTomorrow is 1.00 when
##     Humidity3pm is 67 to 73
##     WindGustSpeed is 26 to 30
##     Sunshine >= 2.0
##     Temp3pm >= 22
```

The tree using balanced training data with the second set of variables identified more variables with importance greater than 1% than the tree using the first set of variables, and they are also in a different order, however Humidity3pm is still the most important variable. All of the temperature variables have importance greater than 1% and Rainfall has importance of 1%.

```
#Checking important variables
importanceBal2 <- treeBal2$variable.importance
importanceBal2 <- round(100*importanceBal2/sum(importanceBal2), 1)
importanceBal2[importanceBal2 >= 1]
```

```
##   Humidity3pm      Sunshine      Cloud9am     TempRange      Cloud3pm
##          20.3          13.1          11.7          11.7          11.2
##   Humidity9am WindGustSpeed       MinTemp       Temp9am   Pressure9am
##          11.0           3.1           2.3           1.9           1.8
##  WindSpeed3pm       MaxTemp   Evaporation       Temp3pm   Pressure3pm
##           1.7           1.6           1.6           1.6           1.4
##  WindSpeed9am        Season      Rainfall
##           1.4           1.1           1.0
```

The model using the second set of variables performs slightly better than the model created with the first set when evaluating the predictions on the same set of training data. This indicates that the rpart algorithm did a good job of choosing the important variables to use in the model.

Accuracy improves from 86.2% to 89.6%. Sensitivity improves from 78.2% to 79.8%. Specificity improves from 89% to 93%. Balanced accuracy improves from 83.6% to 86.4%

```
#Evaluation of second model using Training Set
#Confusion matrix-train
pred_trainBal2 <- predict(treeBal2, train2, type = 'class') # using unbalanced train data
#Make sure to state positive class in the confusion matrix.
confusionMatrix(pred_trainBal2, train2$RainTomorrow, positive="Yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  No Yes
##        No  989  74
##        Yes  75 293
##
##                Accuracy : 0.8959
##                  95% CI : (0.8789, 0.9112)
##     No Information Rate : 0.7435
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.7272
##
##  Mcnemar's Test P-Value : 1
```

```
##
##             Sensitivity : 0.7984
##             Specificity : 0.9295
##          Pos Pred Value : 0.7962
##          Neg Pred Value : 0.9304
##              Prevalence : 0.2565
##          Detection Rate : 0.2048
##    Detection Prevalence : 0.2572
##       Balanced Accuracy : 0.8639
##
##        'Positive' Class : Yes
##
```

Again we will leave the default of 50% probability to make the prediction, because when lowering the probability to 30%, the model performs slightly worse.

```
#Train Set Evaluation with probabilities
#Confusion matrix-train
pred_trainBal2_prob <- predict(treeBal2, train2, type = 'prob') # using train data
# predict rain if chance of rain is more than 30% (default is 50%)
pred_trainBal2_prob30 <- ifelse(pred_trainBal2_prob[,2]>0.3,"Yes","No")
confusionMatrix(data= as.factor(pred_trainBal2_prob30), train2$RainTomorrow, positive="Yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  No Yes
##        No  966  67
##        Yes  98 300
##
##                Accuracy : 0.8847
##                  95% CI : (0.867, 0.9008)
##     No Information Rate : 0.7435
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.7058
##
##  Mcnemar's Test P-Value : 0.01952
##
##             Sensitivity : 0.8174
##             Specificity : 0.9079
##          Pos Pred Value : 0.7538
##          Neg Pred Value : 0.9351
##              Prevalence : 0.2565
##          Detection Rate : 0.2096
##    Detection Prevalence : 0.2781
##       Balanced Accuracy : 0.8627
##
##        'Positive' Class : Yes
##
```

Both balanced models with the two different sets of variables performed similarly when evaluated with the test data set. Accuracy decreased from 80% to 79% and Balanced Accuracy decreased from 80% to 75%. The sensitivity of the first model was better, with 80% compared to the second model's 69%, but the specificity was a little higher with the second model at 82% compare to 80%.

```r
#Test Set Evaluation of Balanced Model 2
#Confusion matrix-test
pred_testBal2 <- predict(treeBal2, test2, type = 'class') # using testing data
confusionMatrix(pred_testBal2, test2$RainTomorrow, positive="Yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  No Yes
##        No  231  26
##        Yes  51  58
##
##                Accuracy : 0.7896
##                  95% CI : (0.7442, 0.8303)
##     No Information Rate : 0.7705
##     P-Value [Acc > NIR] : 0.210684
##
##                   Kappa : 0.4614
##
##  Mcnemar's Test P-Value : 0.006237
##
##             Sensitivity : 0.6905
##             Specificity : 0.8191
##          Pos Pred Value : 0.5321
##          Neg Pred Value : 0.8988
##              Prevalence : 0.2295
##          Detection Rate : 0.1585
##    Detection Prevalence : 0.2978
##       Balanced Accuracy : 0.7548
##
##        'Positive' Class : Yes
##
```

Balanced Model 1 performs better with the test data and should be the model that is implemented.