R Notebook

Name: Gabriel Bentley

Date: 10/04/22

Dataset: Gas Turbine Metric Measurements

https://archive.ics.uci.edu/ml/datasets/Gas+Turbine+CO+and+NOx+Emission+Data+Set

Data set information

The dataset contains 14795 instances of 11 sensor measures aggregated over one hour (by means of average or sum) from a gas turbine located in Turkey's north western region for the purpose of studying flue gas emissions, namely CO and NOx (NO + NO2). This data is collected in another data range (01.01.2011 - 31.12.2011) and (01.01.2015 - 31.12.2015), includes gas turbine parameters (such as Turbine Inlet Temperature and Compressor Discharge pressure) in addition to the ambient variables.

Attribute information

The explanations of sensor measurements and their brief statistics are given below.

Variable (Abbr.) Unit Min Max Mean Ambient temperature (AT) C â€“6.23 37.10 17.71 Ambient pressure (AP) mbar 985.85 1036.56 1013.07 Ambient humidity (AH) (%) 24.08 100.20 77.87 Air filter difference pressure (AFDP) mbar 2.09 7.61 3.93 Gas turbine exhaust pressure (GTEP) mbar 17.70 40.72 25.56 Turbine inlet temperature (TIT) C 1000.85 1100.89 1081.43 Turbine after temperature (TAT) C 511.04 550.61 546.16 Compressor discharge pressure (CDP) mbar 9.85 15.16 12.06 Turbine energy yield (TEY) MWH 100.02 179.50 133.51 Carbon monoxide (CO) mg/m3 0.00 44.10 2.37 Nitrogen oxides (NOx) mg/m3 25.90 119.91 65.29

Read in and seperate the data set into train and test

```
df <- read.csv("gt_2011.csv")

df$year <- rep(c("2011"), each = 7411)

colSums(is.na(df))
```

```
##   AT   AP   AH AFDP GTEP  TIT  TAT  TEY  CDP   CO  NOX year
##    0    0    0    0    0    0    0    0    0    0    0    0
```

```
df2 <- read.csv("gt_2015.csv")
df2$year <- rep(c("2015"), each = 7384)

colSums(is.na(df2))
```

```
##   AT   AP   AH AFDP GTEP  TIT  TAT  TEY  CDP   CO  NOX year
##    0    0    0    0    0    0    0    0    0    0    0    0
```

```
df3 <- rbind(df, df2)

df3$year <- factor(df3$year)
```

```r
set.seed(5675)
i <- sample(1:nrow(df3), nrow(df3)*0.80, replace=FALSE)
train <- df3[i,]
test <- df3[-i,]
```

Explore the data set

```r
str(train)
```

```
## 'data.frame':    11836 obs. of  12 variables:
##  $ AT  : num  20 32 23.8 14.6 30 ...
##  $ AP  : num  1015 1008 1015 1008 1004 ...
##  $ AH  : num  89 51.3 81.1 62.4 64.4 ...
##  $ AFDP: num  3.68 4.37 3.92 3.29 4.52 ...
##  $ GTEP: num  24.8 29.7 25.5 27.7 29.8 ...
##  $ TIT : num  1089 1100 1091 1079 1100 ...
##  $ TAT : num  550 547 550 550 542 ...
##  $ TEY : num  133 146 134 133 146 ...
##  $ CDP : num  11.9 13.1 12.1 11.9 13 ...
##  $ CO  : num  1.036 2.471 0.393 3.594 1.216 ...
##  $ NOX : num  61 56.7 58.1 65.3 56.5 ...
##  $ year: Factor w/ 2 levels "2011","2015": 1 2 1 2 1 2 2 2 1 2 ...
```

```r
summary(train)
```

```
##        AT               AP               AH              AFDP      
##  Min.   :-6.235   Min.   : 989.4   Min.   : 24.09   Min.   :2.369  
##  1st Qu.:11.031   1st Qu.:1009.8   1st Qu.: 64.22   1st Qu.:3.297  
##  Median :16.858   Median :1013.7   Median : 76.03   Median :3.850  
##  Mean   :17.140   Mean   :1014.3   Mean   : 73.97   Mean   :3.843  
##  3rd Qu.:23.468   3rd Qu.:1018.2   3rd Qu.: 85.07   3rd Qu.:4.321  
##  Max.   :37.103   Max.   :1036.6   Max.   :100.17   Max.   :7.319  
##       GTEP            TIT            TAT             TEY       
##  Min.   :17.70   Min.   :1001   Min.   :512.6   Min.   :100.0  
##  1st Qu.:23.22   1st Qu.:1073   1st Qu.:543.1   1st Qu.:127.8  
##  Median :25.01   Median :1086   Median :549.8   Median :133.8  
##  Mean   :25.88   Mean   :1082   Mean   :545.6   Mean   :134.8  
##  3rd Qu.:29.95   3rd Qu.:1100   3rd Qu.:550.0   3rd Qu.:147.4  
##  Max.   :40.72   Max.   :1101   Max.   :550.6   Max.   :179.5  
##       CDP               CO                NOX           year     
##  Min.   : 9.871   Min.   : 0.00039   Min.   : 27.77   2011:5946  
##  1st Qu.:11.541   1st Qu.: 1.09420   1st Qu.: 55.26   2015:5890  
##  Median :11.978   Median : 1.79050   Median : 61.72              
##  Mean   :12.147   Mean   : 2.34972   Mean   : 63.76              
##  3rd Qu.:13.167   3rd Qu.: 2.96442   3rd Qu.: 70.24              
##  Max.   :15.159   Max.   :43.62200   Max.   :119.68              
```

```r
head(train)
```

```
##           AT     AP     AH   AFDP   GTEP    TIT    TAT    TEY    CDP     CO
## 3017  19.993 1014.9 88.981 3.6784 24.841 1088.9 550.36 133.39 11.935 1.0357
## 11728 31.988 1008.2 51.303 4.3743 29.717 1099.8 546.53 146.43 13.065 2.4708
## 5153  23.750 1015.2 81.058 3.9171 25.531 1091.2 549.90 133.64 12.075 0.3928
## 7609  14.610 1008.2 62.448 3.2890 27.655 1079.0 549.95 132.81 11.908 3.5939
## 4496  29.991 1003.8 64.434 4.5170 29.809 1100.3 542.00 146.39 13.021 1.2164
```

```
## 13388 19.098 1004.0 92.465 3.4557 23.974 1075.9 549.66 128.83 11.716 2.0947
##          NOX year
## 3017  60.974 2011
## 11728 56.734 2015
## 5153  58.068 2011
## 7609  65.259 2015
## 4496  56.536 2011
## 13388 44.677 2015
```

```
names(train)
```

```
##  [1] "AT"   "AP"   "AH"   "AFDP" "GTEP" "TIT"  "TAT"  "TEY"  "CDP"  "CO"
## [11] "NOX"  "year"
```

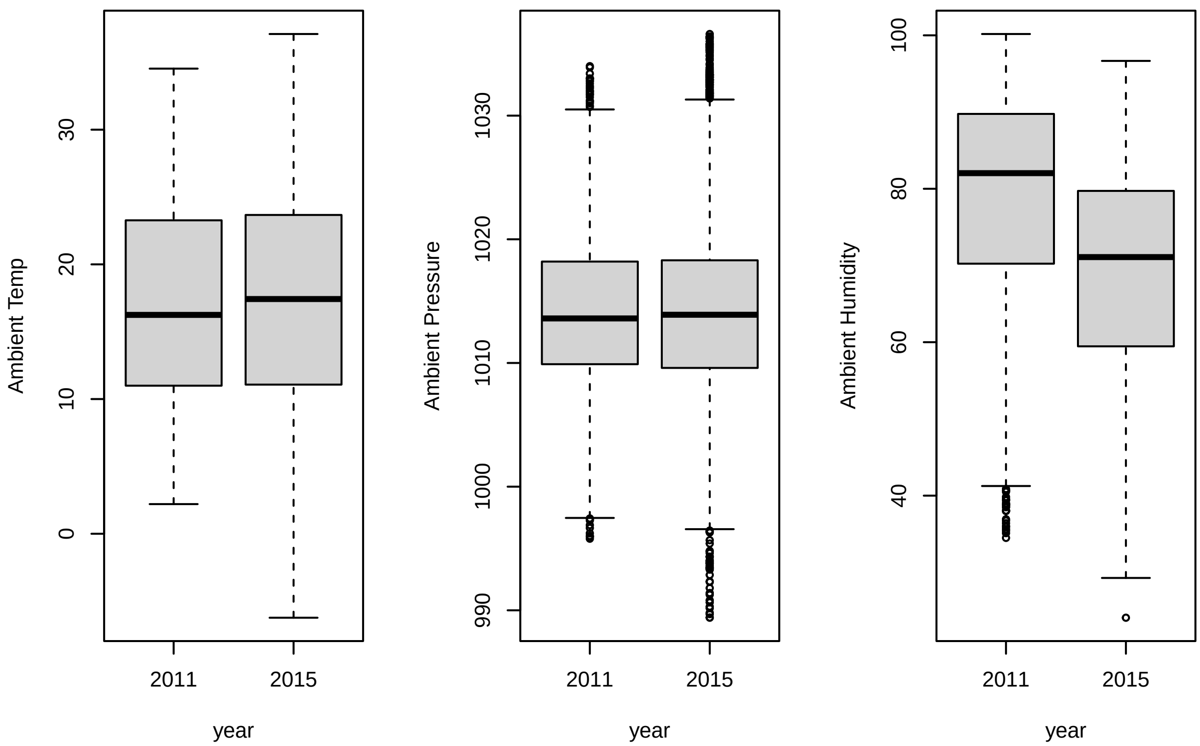
```
colSums(is.na(train))
```

```
##   AT   AP   AH AFDP GTEP  TIT  TAT  TEY  CDP   CO  NOX year
##    0    0    0    0    0    0    0    0    0    0    0    0
```

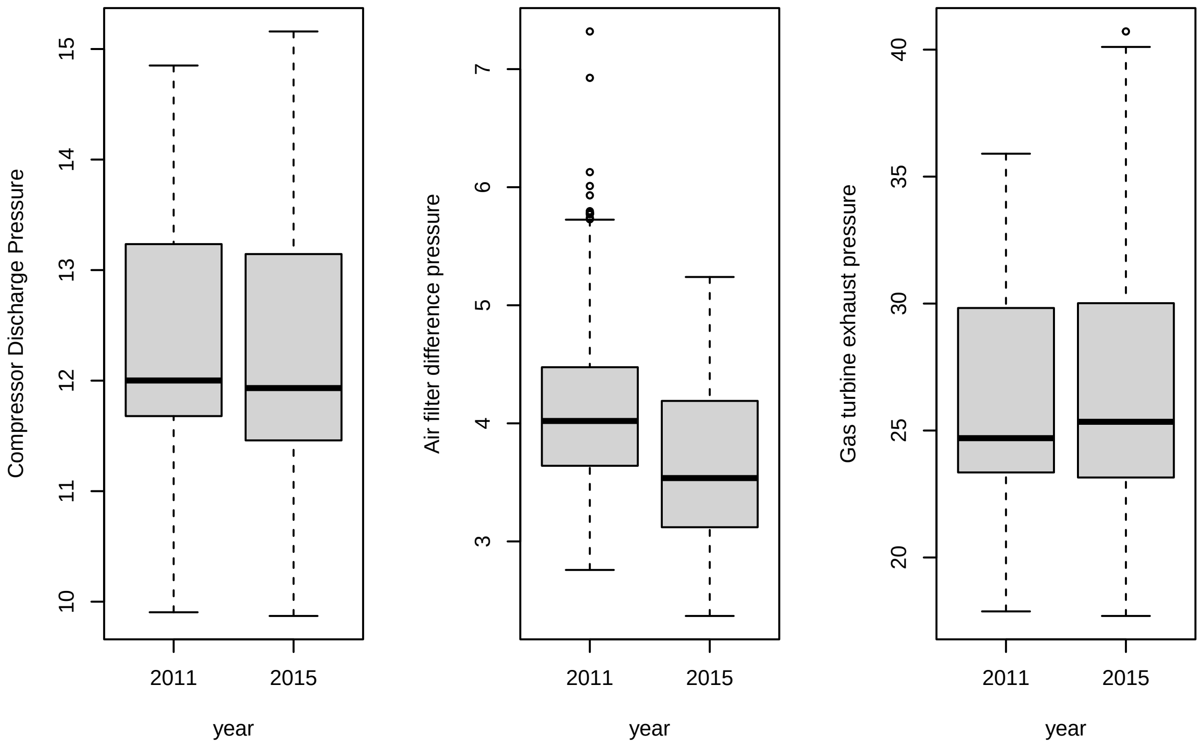
Graphical exploration

Here we plot the year against the other predictors of the data set to determine their relation.

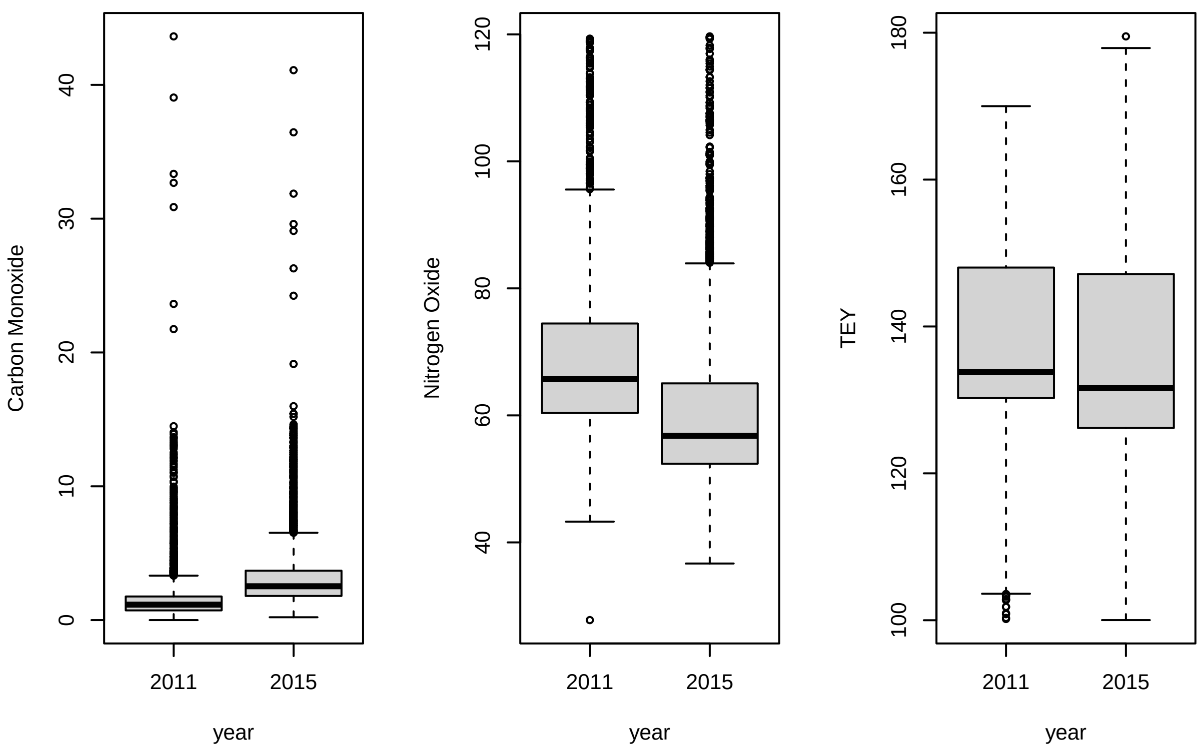
```
par(mfrow=c(1,3))
plot(train$year, train$AT, xlab="year", ylab="Ambient Temp")
plot(train$year, train$AP, xlab="year", ylab="Ambient Pressure")
plot(train$year, train$AH, xlab="year", ylab="Ambient Humidity")
```




```
par(mfrow=c(1,3))
plot(train$year, train$CDP, xlab="year", ylab="Compressor Discharge Pressure")
plot(train$year, train$AFDP, xlab="year", ylab="Air filter difference pressure")
plot(train$year, train$GTEP, xlab="year", ylab="Gas turbine exhaust pressure")
```

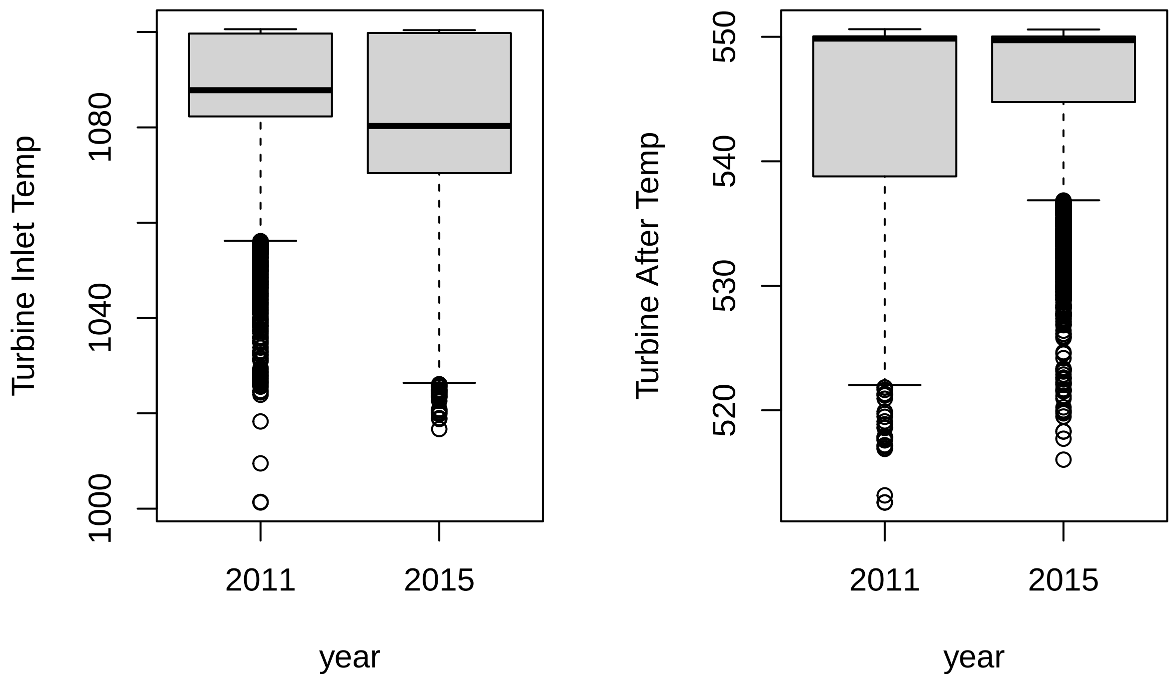



```
par(mfrow=c(1,3))
plot(train$year, train$CO, xlab="year", ylab="Carbon Monoxide")
plot(train$year, train$NOX, xlab="year", ylab="Nitrogen Oxide")
plot(train$year, train$TEY, xlab="year", ylab="TEY")
```




```
par(mfrow = c(1,2))
plot(train$year, train$TIT, xlab="year", ylab="Turbine Inlet Temp")
```

```r
plot(train$year, train$TAT, xlab="year", ylab="Turbine After Temp")
```



Logistic Regression model

```r
library(caret)
```

```
## Loading required package: ggplot2
## Loading required package: lattice
```

```r
glm1 <- glm(year~TEY+CO*NOX+AH+AFDP+TAT*TIT, data=train, family = "binomial")
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```r
summary(glm1)
```

```
##
## Call:
## glm(formula = year ~ TEY + CO * NOX + AH + AFDP + TAT * TIT,
##     family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -8.4904  -0.0431  -0.0001   0.0153   3.7755
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.956e+03  4.430e+02  11.186  < 2e-16 ***
## TEY          2.154e+00  7.953e-02  27.084  < 2e-16 ***
## CO          -7.712e-01  1.297e-01  -5.947 2.73e-09 ***
## NOX         -3.974e-01  1.634e-02 -24.322  < 2e-16 ***
## AH          -1.775e-01  8.855e-03 -20.046  < 2e-16 ***
## AFDP        -4.295e+00  3.912e-01 -10.980  < 2e-16 ***
## TAT         -6.644e+00  7.859e-01  -8.455  < 2e-16 ***
```

```
## TIT         -5.680e+00  4.209e-01 -13.494  < 2e-16 ***
## CO:NOX       1.826e-02  1.815e-03  10.058  < 2e-16 ***
## TAT:TIT      7.757e-03  7.348e-04  10.557  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 16407.9  on 11835  degrees of freedom
## Residual deviance:  1056.1  on 11826  degrees of freedom
## AIC: 1076.1
##
## Number of Fisher Scoring iterations: 10
```

```
probsLR <- predict(glm1, newdata=test, type="response")
predLR <- ifelse(probsLR>0.5, 2015, 2011)
accLR <- mean(predLR == test$year)

cat("accuracy: ", accLR)
```

```
## accuracy:  0.9868199
```

```
table(predLR, test$year)
```

```
##
## predLR 2011 2015
##   2011 1451   25
##   2015   14 1469
```

```
confusionMatrix(as.factor(predLR), reference = test$year)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction 2011 2015
##       2011 1451   25
##       2015   14 1469
##
##                Accuracy : 0.9868
##                  95% CI : (0.982, 0.9906)
##     No Information Rate : 0.5049
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.9736
##
##  Mcnemar's Test P-Value : 0.1093
##
##             Sensitivity : 0.9904
##             Specificity : 0.9833
##          Pos Pred Value : 0.9831
##          Neg Pred Value : 0.9906
##              Prevalence : 0.4951
##          Detection Rate : 0.4904
##    Detection Prevalence : 0.4988
##       Balanced Accuracy : 0.9869
##
```

```
##        'Positive' Class : 2011
##
```

kNN Classification model

```
library(caret)
library(class)

predKNN <- knn(train=train, test=test, cl=train$year, k = 3)
results <- predKNN == test$year
accKNN <- length(which(results == TRUE))/length(results)
cat("\nkNN Classification\naccuracy: ", accKNN)
```

```
##
## kNN Classification
## accuracy:  0.9935789
```

```
table(predKNN, test$year)
```

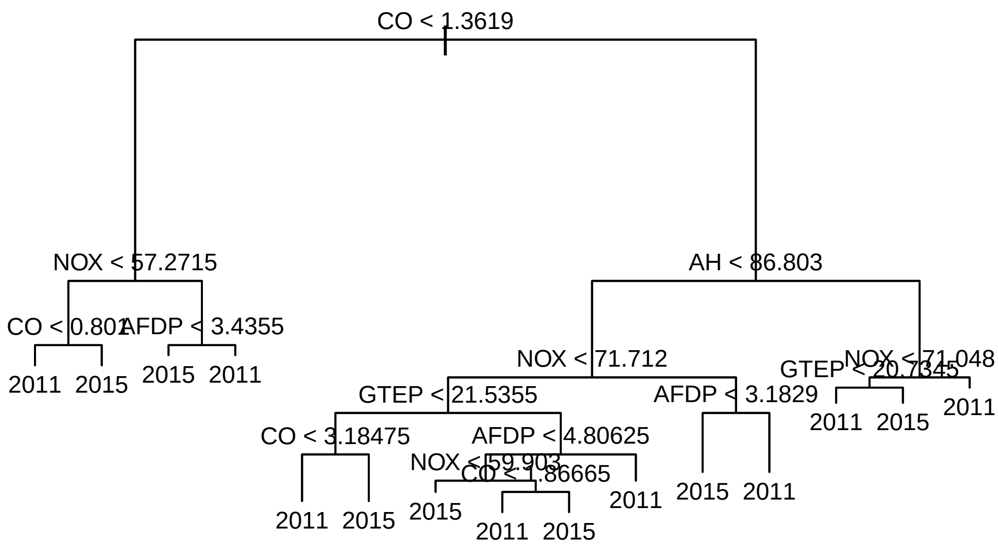
```
##
## predKNN 2011 2015
##    2011 1457   11
##    2015    8 1483
```

Decision tree Classification model

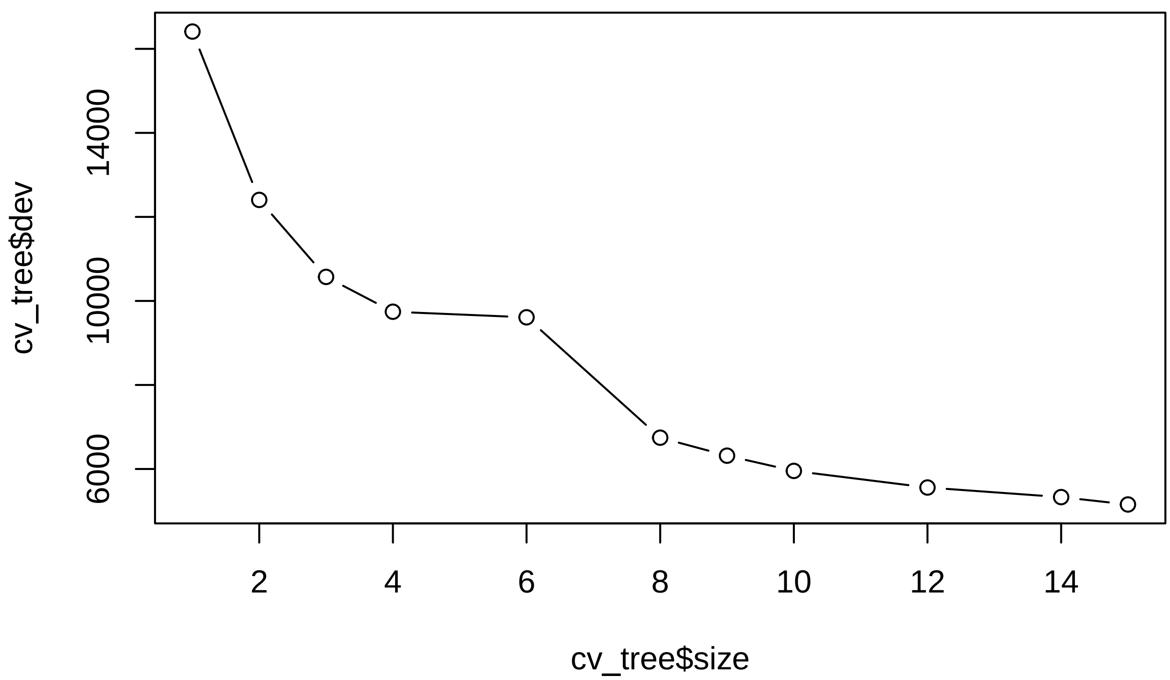
```
library(tree)
tree1 <- tree(year~., data=train)
summary(tree1)
```

```
##
## Classification tree:
## tree(formula = year ~ ., data = train)
## Variables actually used in tree construction:
## [1] "CO"   "NOX"  "AFDP" "AH"   "GTEP"
## Number of terminal nodes:  15
## Residual mean deviance:  0.3857 = 4560 / 11820
## Misclassification error rate: 0.0637 = 754 / 11836
```

```
plot(tree1)
text(tree1, cex=0.75, pretty=0)
```




```
cv_tree <- cv.tree(tree1)
plot(cv_tree$size, cv_tree$dev, type='b')
```



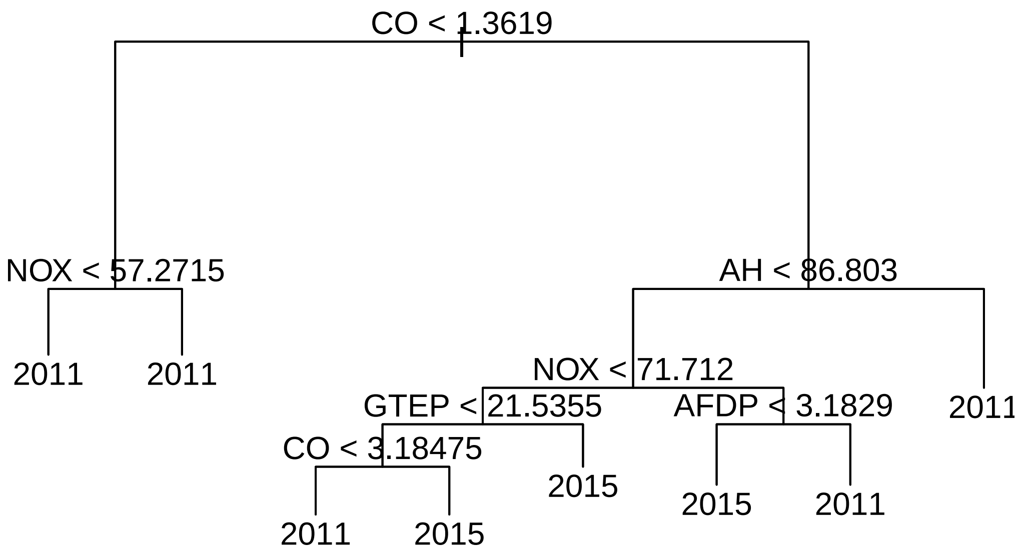

```
predDT <- predict(tree1, newdata=test, type="class")

table(predDT, test$year)
```

```
##
## predDT 2011 2015
##   2011 1366   88
##   2015   99 1406
```

```
mean(predDT==test$year)
```

```
## [1] 0.936803
```

```r
tree_pruned <- prune.tree(tree1, best = 8)
plot(tree_pruned)
text(tree_pruned, pretty = 0)
```




```r
predPrune <- predict(tree_pruned, newdata=test, type="class")

table(predPrune, test$year)
```

```
##
## predPrune 2011 2015
##      2011 1375  217
##      2015   90 1277
```

```r
mean(predPrune==test$year)
```

```
## [1] 0.8962487
```

Analysis of results

Of the three models kNN was the most accurate with an accuracy of 99.3%. The least accurate model was the decision tree model at 93.9%. I looked at each predictor ploted against year and the predictors that had the greatest changes between 2011 and 2015 were used as predictors for the logistic regression model. By using predictors that show noticable differences over the years the logistic regression model greatly increased in accuracy. For the kNN model I decided to use a k value of 3 because it proved to be the value that gave the kNN model the most accuracy. The Decision tree model produced a hard to read tree when plotted, when pruning the tree to the 8 best leaves I was given a more legible tree able to understand that many of the predictors were involved in the branching of the tree. Pruning the tree resulted in a less accurate model with only around 90% accuracy.

In the end all three models worked well when predicting the classification of given data points as coming from 2011 or 2015, with all models having an accuracy higher than 90%. The reason why kNN and logistic regression preformed well is becasue the values in 2011 and the values in 2015 are noticable seperable based of the predictors. The reason why decision tree was not as accurate is probably because it overfit to the training data.