Analysis of State dataset

Maria George

December 14, 2015

```r
# Loading all the required libraries
library("dplyr")
library("ggplot2")
library("car") # Contains the scatterplotMatrix function
#install.packages("boot")
library("boot") # Perform crossvalidation
#install.packages("tree")
library("tree")
library("randomForest")
library(pROC) # Useful for computing and plotting classifer metrics
library("ISLR")
# install.packages("gbm")
library(gbm) # To perform boosting
```

The state dataset, available as part of the base R package, contains various data related to the 50 states of the United States of America.

Exploring the relationship between a states Murder rate and other characteristics of the state, for example population, illiteracy rate, and more. Follow the questions below to perform this analysis.

Examining the bivariate relationships present in the data.

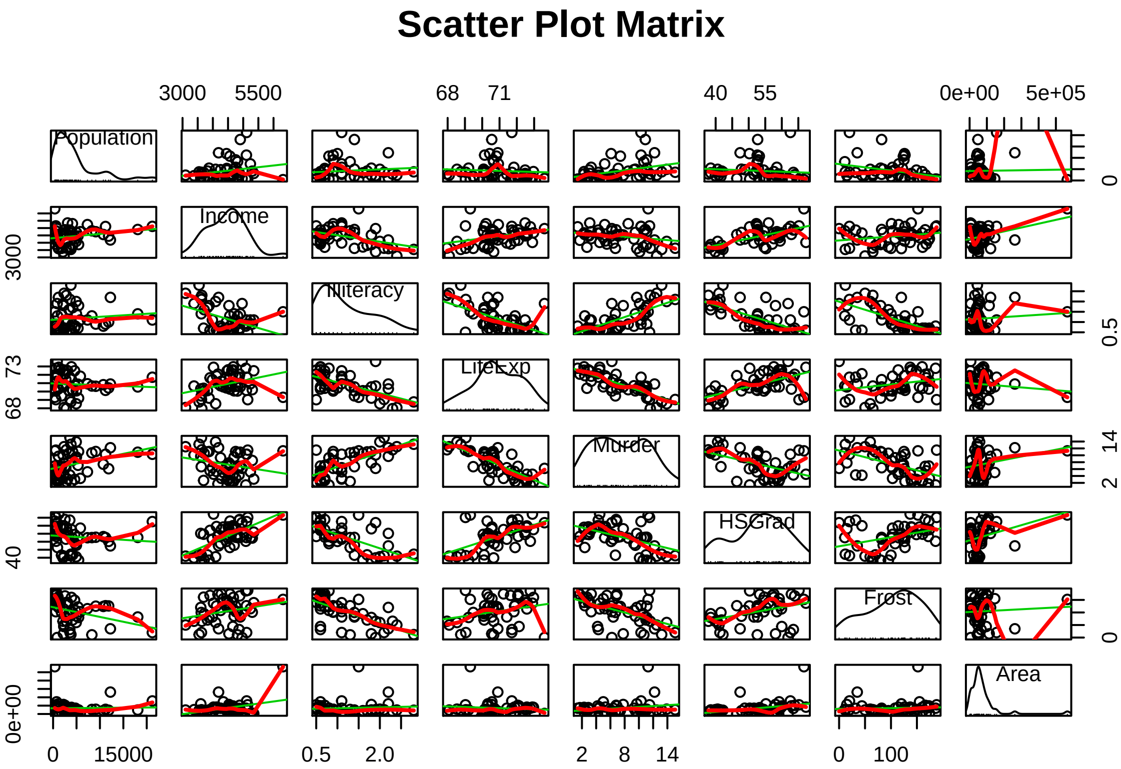
```r
# Loading the state.x77 data into a local variable
state.data <- as.data.frame(state.x77)

# Renaming the column names
colnames(state.data)[colnames(state.data)=="HS Grad"] <- "HSGrad"
colnames(state.data)[colnames(state.data)=="Life Exp"] <- "LifeExp"

# Displaying the correlation matrix
cor(state.data)
```

```
##             Population     Income  Illiteracy     LifeExp     Murder
## Population  1.00000000  0.2082276  0.10762237 -0.06805195  0.3436428
## Income      0.20822756  1.0000000 -0.43707519  0.34025534 -0.2300776
## Illiteracy  0.10762237 -0.4370752  1.00000000 -0.58847793  0.7029752
## LifeExp    -0.06805195  0.3402553 -0.58847793  1.00000000 -0.7808458
## Murder      0.34364275 -0.2300776  0.70297520 -0.78084575  1.0000000
## HSGrad     -0.09848975  0.6199323 -0.65718861  0.58221620 -0.4879710
## Frost      -0.33215245  0.2262822 -0.67194697  0.26206801 -0.5388834
## Area        0.02254384  0.3633154  0.07726113 -0.10733194  0.2283902
##                 HSGrad      Frost        Area
## Population -0.09848975 -0.3321525  0.02254384
## Income      0.61993232  0.2262822  0.36331544
## Illiteracy -0.65718861 -0.6719470  0.07726113
## LifeExp     0.58221620  0.2620680 -0.10733194
## Murder     -0.48797102 -0.5388834  0.22839021
## HSGrad      1.00000000  0.3667797  0.33354187
## Frost       0.36677970  1.0000000  0.05922910
## Area        0.33354187  0.0592291  1.00000000
```

```r
# Plotting the scatterplot matrix to check for bivariate relationships
scatterplotMatrix(state.data, spread=FALSE, lty.smooth=2,
                  main="Scatter Plot Matrix")
```



```
# Ensuring that there are no missing values
state.data <- state.data[complete.cases(state.data), ]
```

Results observed from the scatter plot:

1. From the scatter plot, we can observe that Murder rate is bimodal and each of the predictor variables are skewed to some extent.
2. Murder rate rises with Population (r=0.344), Illiteracy(0.703) and Area(0.228)
3. Murder rate falls with Income(r=-0.230), LifeExp(-0.781), HSGrad(-0.488) and Frost(-0.539)
4. Murder rate has a strong correlation with Illiteracy and LifeExp, moderate correlation with Frost.
5. We can also observe that Illiteracy falls with HSGrad and Frost.
6. Income rises with HSGrad.

(b) Fit a multiple linear regression model. How much variance in the murder rate across states do the predictor variables explain?

```
# Fitting a multiple linear regression model
state.fit <- lm(Murder ~ Population + Income + Illiteracy + LifeExp +
                  HSGrad +  Frost + Area, data = state.data)

# Displaying the summary statistics of the fitted model
summary(state.fit)
```

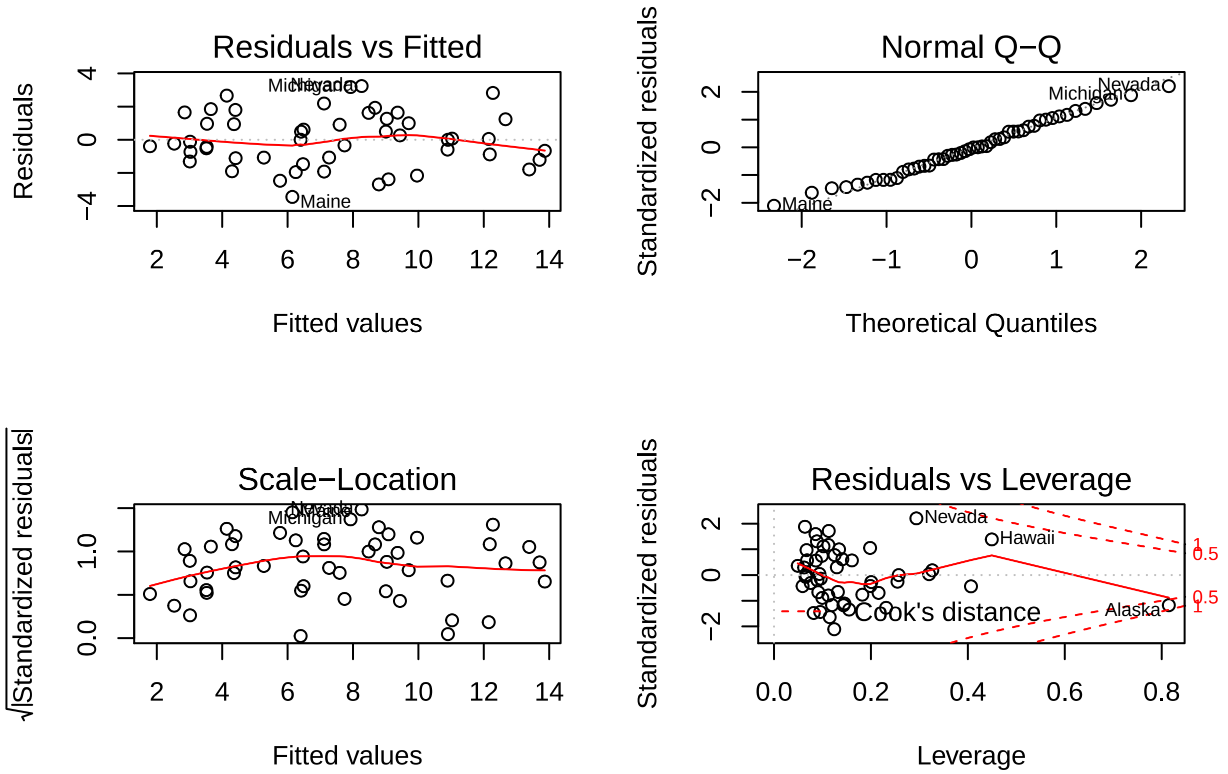
```
##
## Call:
## lm(formula = Murder ~ Population + Income + Illiteracy + LifeExp +
##     HSGrad + Frost + Area, data = state.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4452 -1.1016 -0.0598  1.1758  3.2355
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.222e+02  1.789e+01   6.831 2.54e-08 ***
## Population   1.880e-04  6.474e-05   2.905  0.00584 **
## Income      -1.592e-04  5.725e-04  -0.278  0.78232
## Illiteracy   1.373e+00  8.322e-01   1.650  0.10641
## LifeExp     -1.655e+00  2.562e-01  -6.459 8.68e-08 ***
## HSGrad       3.234e-02  5.725e-02   0.565  0.57519
## Frost       -1.288e-02  7.392e-03  -1.743  0.08867 .
## Area         5.967e-06  3.801e-06   1.570  0.12391
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.746 on 42 degrees of freedom
## Multiple R-squared:  0.8083, Adjusted R-squared:  0.7763
## F-statistic: 25.29 on 7 and 42 DF,  p-value: 3.872e-13
```

R-squared value gives the amount of variance explained by the model. From the summary statistic table, we can observe that Multiple R-squared value is 0.808. This means that 80.8% of the variance in the murder rate can be predicted using the predictor variables.

From the summary statistics of the fitted multiple linear rgression model, we can observe that predictor variable LifeExp is statistically significant at the 0.001 level, Population is significant at the 0.05 level and Frost at the 0.1 level.

Evaluating the statistical assumptions in regression analysis of the above fitted model

```r
# Plotting the fitted model to evaluate the statistical assumptions
par(mfrow = c(2,2))
plot(state.fit)
```



Statistical assumptions behind regression models are: 1. Normality: The dependent variable is normally distributed for fixed values of the independent variables. 2. Independence: The response variable values are independent of each other. 3. Linearity: The dependent variable is linearly related to the independent variables. 4. Homoscedasticity or Constant variance: The variance of the dependent variable doesnt vary with the levels of the independent variables.

From the plot of the fitted model: 1. Normality: If the dependent variable is normally distributed for a fixed set of predictor values, then the residual values should be normally distributed with a mean of 0. If this condition is met, the points in the Normal Q-Q plot, will fall on the 45 degree straight line. This is true for the fitted model. Thus the normality condition is satisfied.

2. Independence: This is judged using how the data was collected. There is no reason to believe that the murder rate in once state influences the murder rate in another state. If not, the the assumption of independence has to be adjusted.

3. Linearity: If the dependent variable is linearly related to the independent variables, there should be no systematic relationship between the residuals and the fitted values. Residual vs Fitted graph will not have patterns and will be randomly distributed, which is true in this case. Thus the linearity assumption is satisfied.

4. Homoscedasticity: If the constant variance assumption is met, the points in the Scale-Location graph (bottom left) should be a random band around a horizontal line, which is true in this case. Thus the constant variance assumption is satisfied.

However, there are some concerns about the model:

1. There are a few outliers. Oultliers are observations that are not well by the model, thereby resulting in large postive or negative residuals. From the Residuals vs Fitted graph, we can see that Nevada, Michigan and Maine have high residual values.
2. High leverage points are observations with unusual value fo r predictor variables.
3. Collinearity: From the scatter plots, we have observed that predictor variables Income and HSGrad are correlated. Also, Illiteracy and HSGrad are correlated. Thus with this model, it is difficult to interpret the individual effec of these predictor variables on the response variable. Thus the accuracy of the estimates of the regression coefficients is reduced. This reduces the t-statistic, thereby failing to identify predictor variables with non-zero coefficients. From the summary statistics of the fitted model, it can be observed that only the predictor variables LifeExp and Population are statistically significant.

Using a stepwise model selection procedure of your choice to obtain a best fit model.

```
# Using stepwise model selection to obtain a best fit model
state.best.fit <- step(state.fit, data = state.data, direction = "backward")
```

```
## Start:  AIC=63.01
## Murder ~ Population + Income + Illiteracy + LifeExp + HSGrad +
##     Frost + Area
##
##              Df Sum of Sq    RSS    AIC
## - Income      1     0.236 128.27 61.105
## - HSGrad      1     0.973 129.01 61.392
## <none>                    128.03 63.013
## - Area        1     7.514 135.55 63.865
## - Illiteracy  1     8.299 136.33 64.154
## - Frost       1     9.260 137.29 64.505
## - Population  1    25.719 153.75 70.166
## - LifeExp     1   127.175 255.21 95.503
##
## Step:  AIC=61.11
## Murder ~ Population + Illiteracy + LifeExp + HSGrad + Frost +
##     Area
##
##              Df Sum of Sq    RSS    AIC
## - HSGrad      1     0.763 129.03 59.402
## <none>                    128.27 61.105
## - Area        1     7.310 135.58 61.877
## - Illiteracy  1     8.715 136.98 62.392
## - Frost       1     9.345 137.61 62.621
## - Population  1    27.142 155.41 68.702
## - LifeExp     1   127.500 255.77 93.613
##
## Step:  AIC=59.4
## Murder ~ Population + Illiteracy + LifeExp + Frost + Area
##
##              Df Sum of Sq    RSS    AIC
## <none>                    129.03 59.402
## - Illiteracy  1     8.723 137.75 60.672
## - Frost       1    11.030 140.06 61.503
## - Area        1    15.937 144.97 63.225
## - Population  1    26.415 155.45 66.714
## - LifeExp     1   140.391 269.42 94.213
```

```
# Displaying the summary statistics of the model
summary(state.best.fit)
```

```
##
## Call:
## lm(formula = Murder ~ Population + Illiteracy + LifeExp + Frost +
##     Area, data = state.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2976 -1.0711 -0.1123  1.1092  3.4671
##
```

```
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  1.202e+02  1.718e+01   6.994 1.17e-08 ***
## Population   1.780e-04  5.930e-05   3.001  0.00442 ** 
## Illiteracy   1.173e+00  6.801e-01   1.725  0.09161 .  
## LifeExp     -1.608e+00  2.324e-01  -6.919 1.50e-08 ***
## Frost       -1.373e-02  7.080e-03  -1.939  0.05888 .  
## Area         6.804e-06  2.919e-06   2.331  0.02439 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.712 on 44 degrees of freedom
## Multiple R-squared:  0.8068, Adjusted R-squared:  0.7848 
## F-statistic: 36.74 on 5 and 44 DF,  p-value: 1.221e-14
```

The “best” fit model is different from the full model. Here the step function helps in building the bst fit model based on lowest AIC. The full model only had Population and LifeExp as the predictor variables that are statistically significant with p-value <0.05. However, in the “best fit” model, predictor variables Population, LifeExp and Area are found to be statistically significant. Also, when compared to the full model, the best fit model only has 5 predictor variables in the model,three of them statistically significant(p-value <0.05) and the Illiteracy and Frost significant at 0.1 level.

We can also observe that, 1. Residual standard error has reduced from 1.75 (full model) to 1.71 (best fit model). 2. Adjusted R-squared value has improved from 0.776(full model) to 0.785(best fit model). 3. F-statistic has improved from 25.3(full model) to 36.7(best fit model).

Assessing the model for generalizability. Performing a 10-fold cross validation to estimate model performance

```r
# Fitting the full model using glm
state.fit <- glm(Murder ~ Population + Income + Illiteracy + LifeExp +
                   HSGrad +  Frost + Area, data = state.data)

# Fitting the best fit model using stepwise model selection procedure
state.best.fit <- step(state.fit, data = state.data, direction = "backward")
```

```
## Start:  AIC=206.91
## Murder ~ Population + Income + Illiteracy + LifeExp + HSGrad +
##     Frost + Area
##
##              Df Deviance    AIC
## - Income      1   128.27 205.00
## - HSGrad      1   129.01 205.29
## <none>            128.03 206.91
## - Area        1   135.55 207.76
## - Illiteracy  1   136.33 208.05
## - Frost       1   137.29 208.40
## - Population  1   153.75 214.06
## - LifeExp     1   255.21 239.40
##
## Step:  AIC=205
## Murder ~ Population + Illiteracy + LifeExp + HSGrad + Frost +
##     Area
##
##              Df Deviance    AIC
## - HSGrad      1   129.03 203.30
## <none>            128.27 205.00
## - Area        1   135.58 205.77
## - Illiteracy  1   136.98 206.29
## - Frost       1   137.61 206.51
## - Population  1   155.41 212.60
## - LifeExp     1   255.77 237.51
##
## Step:  AIC=203.3
## Murder ~ Population + Illiteracy + LifeExp + Frost + Area
##
##              Df Deviance    AIC
## <none>            129.03 203.30
## - Illiteracy  1   137.75 204.57
## - Frost       1   140.06 205.40
## - Area        1   144.97 207.12
## - Population  1   155.45 210.61
## - LifeExp     1   269.42 238.11
```

```r
# Calulating the error rate of the best fit model on the entire data
mean((state.data$Murder-predict(state.best.fit, state.data))^2)
```

```
## [1] 2.580632
```

```r
# Performing 10 fold cross validation
set.seed(1)
```

```r
cv.state <- cv.glm(state.data, state.best.fit, K = 10)

# Displaying the cross validation results
cv.state$delta
```

```
## [1] 3.842053 3.755144
```

The cv.glm() functionproduces a list with several components. One of them is delta. The two numbers of delta represent the cross validation results.The first value, 3.84 is the standard K-fold CV estimate while the second one, 3.76 is the bias corrected version.

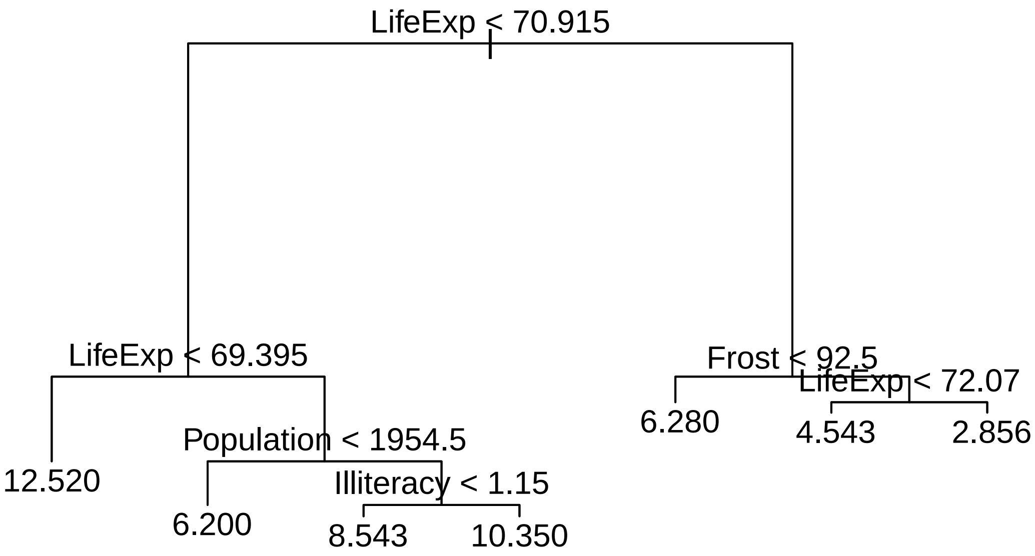
As expected, the CV error estimates is slightly higher than the error rate from linear regression above (2.58) indicating that the error rate obtained from the linear regression model under estimates the test error rate, thereby leading to overfitting of the data.

Fitting a regression tree using the same covariates in the best fit model. Using cross validation to select the best tree.

```r
# Fitting a regression tree with the same covariates as the best fit model
# state.best.fit
tree.state <- tree(Murder ~ Population + Illiteracy + LifeExp + Frost + Area,
                   data = state.data)
# Displaying the summary statistics
summary(tree.state)
```

```
##
## Regression tree:
## tree(formula = Murder ~ Population + Illiteracy + LifeExp + Frost +
##     Area, data = state.data)
## Variables actually used in tree construction:
## [1] "LifeExp"    "Population" "Illiteracy" "Frost"
## Number of terminal nodes:  7
## Residual mean deviance:  2.813 = 121 / 43
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -3.50000 -1.18900  0.02222  0.00000  0.74290  4.02000
```

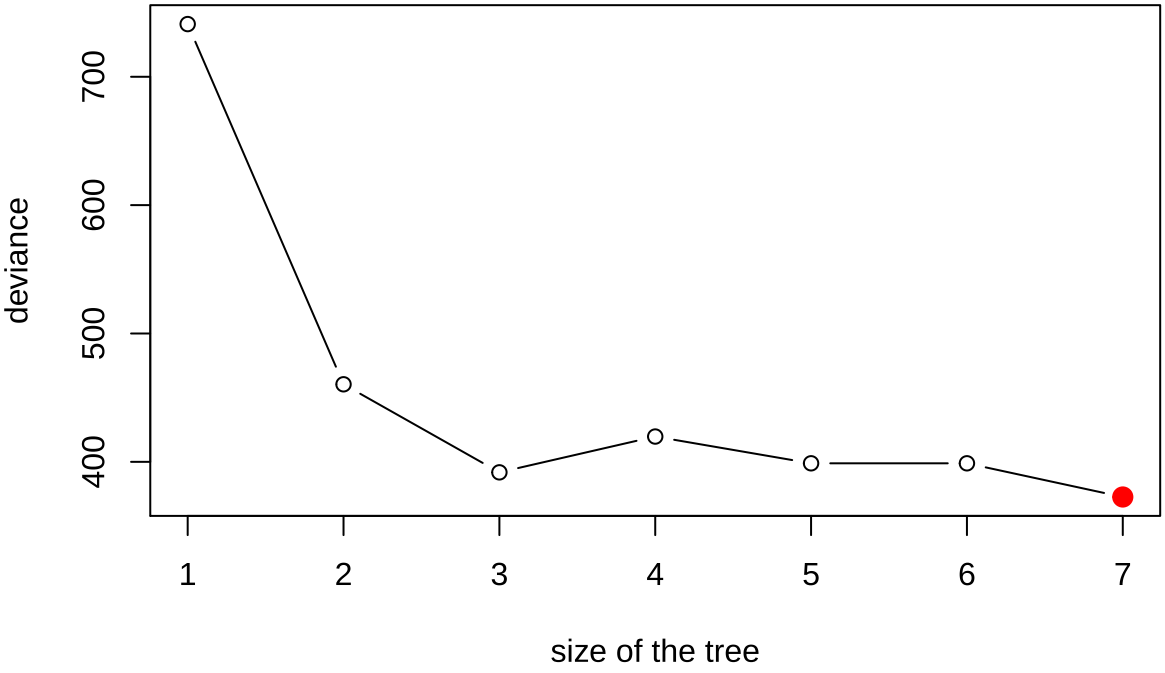
```r
# Plotting the tree
plot(tree.state)
text(tree.state, pretty =0)
```



```r
# Performing cross validation
set.seed(1)
cv.tree.state <- cv.tree(tree.state)
# Displaying the summary statistics of the decision tree
summary(cv.tree.state)
```

```
##        Length Class  Mode
## size   7      -none- numeric
## dev    7      -none- numeric
## k      7      -none- numeric
## method 1      -none- character
```

```
# Plotting the tree to find the determine optimal tree size
plot(cv.tree.state$size, cv.tree.state$dev, type = "b",
     xlab = "size of the tree", ylab = "deviance")
tree.min <- which.min(cv.tree.state$dev)
points(cv.tree.state$size[tree.min], min(cv.tree.state$dev),
       col = "red", cex = 2, pch = 20)
```



```
# Pruning the tree based on the best tree size obtained from cross validation
prune.state <- prune.tree(tree.state, best = 7)
# Displaying the summary statistics of the pruned tree
summary(prune.state)
```

```
##
## Regression tree:
## tree(formula = Murder ~ Population + Illiteracy + LifeExp + Frost +
##     Area, data = state.data)
## Variables actually used in tree construction:
## [1] "LifeExp"    "Population" "Illiteracy" "Frost"
## Number of terminal nodes:  7
## Residual mean deviance:  2.813 = 121 / 43
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -3.50000 -1.18900  0.02222  0.00000  0.74290  4.02000
```

Decision tree is fitted using the same covariates (Population, Illiteracy, LifeExp, Frost, Area) as the best fit model obtained using stepwise model selection procedure, state.best.fit.

After performing cross validation, it was observed that the tree of size 7 results in the lowest deviance (440). Thus a tree of size 7 is the best fit tree. This can also observed from the graph. The trees is pruned to size 7 to obatin the best fit tree.

Comparing the models based on their performance

```
# Calulating the error rate of the best fit model obatined from (d)
# on the entire data
mean((state.data$Murder-predict(state.best.fit, state.data))^2)
```

```
## [1] 2.580632
```

```
# Calulating the error rate of the pruned tree on the entire data
mean((state.data$Murder-predict(prune.state, state.data))^2)
```

```
## [1] 2.41919
```

Mean Squared Error is calculated for both the models (state.best.fit and pruned tree) on the entire data. It can be seen that the train MSE of pruned tree (2.42) is less than the MSE of the linear best fit model (2.58). Thus pruned tree (model obrained in part(f)) is preferred to the linear best fit model(model obrained in part(d)).