Question 10.1

Using the same crime data set uscrime.txt as in Questions 8.2 and 9.1, find the best model you can using (a) a regression tree model, and (b) a random forest model.

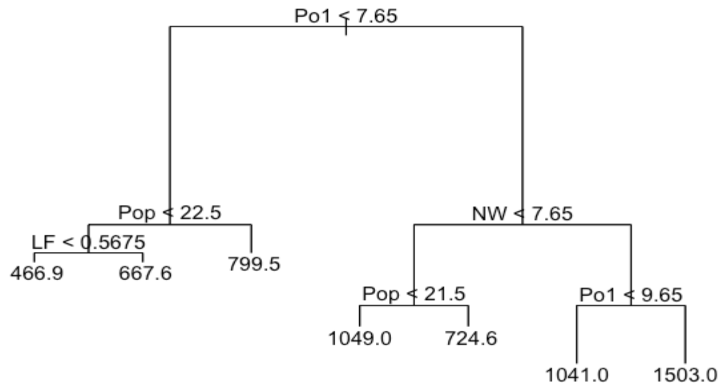
Answer 10.1

#Clear global environment, load and preview dataset

```
> rm(list=ls())
> df1 <- read.table("/Users/…/uscrime.txt", header = T)
> head(df1)
     M So   Ed  Po1  Po2    LF   M.F Pop   NW    U1  U2 Wealth Ineq      Pr
ob
1 15.1  1  9.1  5.8  5.6 0.510  95.0  33 30.1 0.108 4.1   3940 26.1 0.0846
02
2 14.3  0 11.3 10.3  9.5 0.583 101.2  13 10.2 0.096 3.6   5570 19.4 0.0295
99
3 14.2  1  8.9  4.5  4.4 0.533  96.9  18 21.9 0.094 3.3   3180 25.0 0.0834
01
4 13.6  0 12.1 14.9 14.1 0.577  99.4 157  8.0 0.102 3.9   6730 16.7 0.0158
01
5 14.1  0 12.1 10.9 10.1 0.591  98.5  18  3.0 0.091 2.0   5780 17.4 0.0413
99
6 12.1  0 11.0 11.8 11.5 0.547  96.4  25  4.4 0.084 2.9   6890 12.6 0.0342
01
     Time Crime
1 26.2011   791
2 25.2999  1635
3 24.3006   578
4 29.9012  1969
5 21.2998  1234
6 20.9995   682
> library(tree)
> library(randomForest)
> library(caret)
```

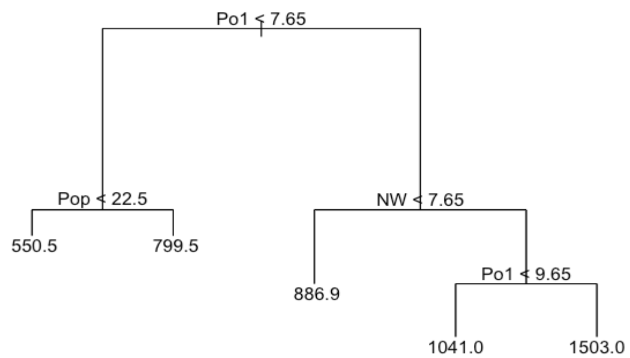
Perform regression tree model using tree package. Plot the model for further studies. Summary of the model suggests four variables Po1 Pop LF and NW are used in the tree model. Graph suggests that Pop and Po1 are used multiple times in branching, with Po1 being the primary branching factor.

```
> model1 <- tree(Crime ~., data = df1)
> summary(model1)

Regression tree:
tree(formula = Crime ~ ., data = df1)
Variables actually used in tree construction:
[1] "Po1" "Pop" "LF"  "NW"
Number of terminal nodes:  7
Residual mean deviance:  47390 = 1896000 / 40
Distribution of residuals:
    Min.  1st Qu.   Median     Mean   3rd Qu.     Max.
-573.900  -98.300   -1.545    0.000  110.600   490.100
> plot(model1)
> text(model1)
```



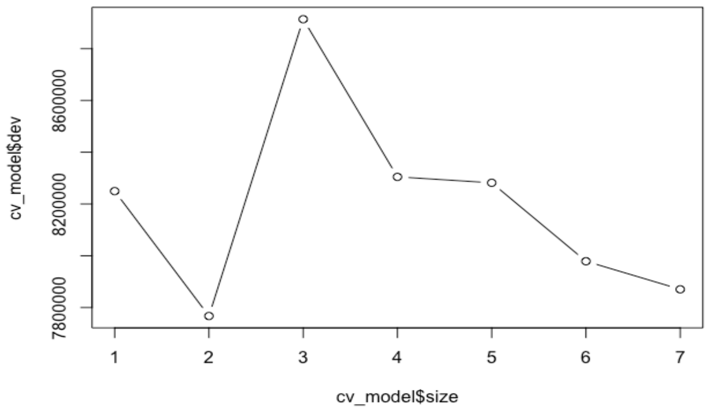
Pruning was further performed on the dataset. Notice the residual mean deviance increases comparing to the model without pruning, and thus prune the tree may not be the better practice.

```
> prune_model1 <- prune.tree(model1, best = 5)
> summary(prune_model1)

Regression tree:
snip.tree(tree = model1, nodes = c(4L, 6L))
Variables actually used in tree construction:
[1] "Po1" "Pop" "NW"
Number of terminal nodes:  5
Residual mean deviance:  54210 = 2277000 / 42
Distribution of residuals:
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 -573.9  -107.5    15.5     0.0   122.8   490.1
> plot(prune_model1)
> text(prune_model1)
```



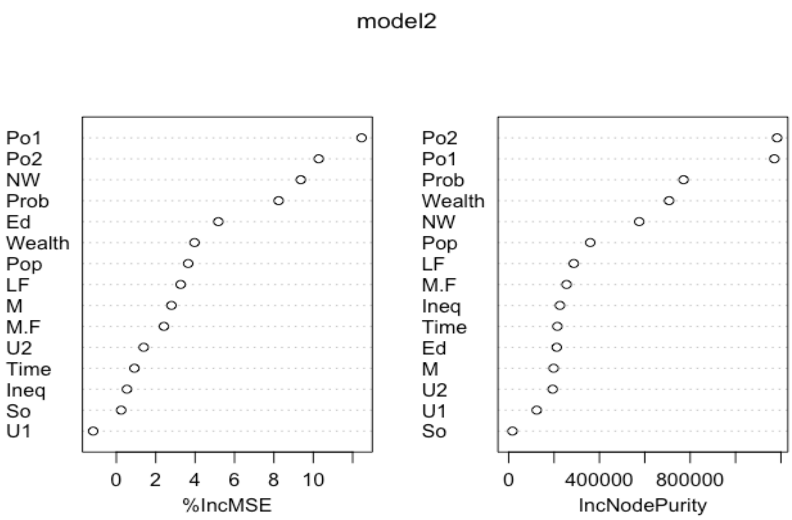
Perform cross validation on the model without prune to measure the model's performance. Graph suggests overfitting takes place since some parameters may not be significant. Overall, the model with Po1 < 7.65 can explain around half of the dataset.

```
> cv_model <- cv.tree(model1)
> plot(cv_model$size, cv_model$dev, type = 'b')
```



Random forest is performed on the dataset. According to the %IncMSE plot, Po1 Po2 NW and Prob are relatively more significant predictors, and overfitting may play a role in this model as well since many other predictors provide limited increase in %IncMSE.

```
> model2 <- randomForest(Crime ~., data=df1, importance = TRUE, nodesize =
  5)
> importance(model2)
           %IncMSE IncNodePurity
M        2.7974536     198168.45
So       0.2436021      16687.76
Ed       5.1795354     211800.62
Po1     12.4522580    1170907.63
Po2     10.2756302    1183029.44
LF       3.2656375     287012.29
M.F      2.4168656     255127.95
Pop      3.6478250     359536.39
NW       9.3658870     575339.37
U1      -1.1764591     123545.48
U2       1.3896550     193963.97
Wealth   3.9686152     707121.97
Ineq     0.5373144     225989.83
Prob     8.2409816     770869.77
Time     0.9190329     214652.53
> varImpPlot(model2)
```



Question 10.2

Describe a situation or problem from your job, everyday life, current events, etc., for which a logistic regression model would be appropriate. List some (up to 5) predictors that you might use.

Answer 10.2

In manufacturing industry, we analyze possible root causes from part failures and adjust plan accordingly. We can use logistic regression to analyze the potential causes of part failure, with time stored in inventory, machine maintenance frequency, unit production per hour for each machine as predictors. Results can be taken into account when implementing production schedule to optimize resources and reduce costs of the production process.

Question 10.3

1. Using the GermanCredit data set germancredit.txt from http://archive.ics.uci.edu/ml/machine-learning-databases/statlog/german / (description at http://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29), use logistic regression to find a good predictive model for whether credit applicants are good credit risks or not. Show your model (factors used and their coefficients), the software output, and the quality of fit.

2. Because the model gives a result between 0 and 1, it requires setting a threshold probability to separate between "good" and "bad" answers. In this data set, they estimate that incorrectly identifying a bad customer as good, is 5 times worse than incorrectly classifying a good customer as bad. Determine a good threshold probability based on your model.

Answer 10.3

#Clear global environment, load and preview dataset

```
> rm(list=ls())
> df1 <- read.table("/Users/…/germancredit.txt", header = F)
> head(df1)
   V1 V2  V3  V4   V5  V6  V7 V8  V9  V10 V11  V12 V13  V14  V15 V16  V17
V18  V19  V20 V21
1 A11  6 A34 A43 1169 A65 A75  4 A93 A101   4 A121  67 A143 A152   2 A173
  1 A192 A201   1
2 A12 48 A32 A43 5951 A61 A73  2 A92 A101   2 A121  22 A143 A152   1 A173
  1 A191 A201   2
3 A14 12 A34 A46 2096 A61 A74  2 A93 A101   3 A121  49 A143 A152   1 A172
  2 A191 A201   1
4 A11 42 A32 A42 7882 A61 A74  2 A93 A103   4 A122  45 A143 A153   1 A173
  2 A191 A201   1
5 A11 24 A33 A40 4870 A61 A73  3 A93 A101   4 A124  53 A143 A153   2 A173
  2 A191 A201   2
6 A14 36 A32 A46 9055 A65 A73  2 A93 A101   4 A124  35 A143 A153   1 A172
  2 A192 A201   1
```

Convert response variable into 0 and 1, create train and validation dataset with 80% and 20% of the dataset respectively. Logistic regression is performed on the train dataset. Summary of the model suggests 17 predictors are significant given .05 α.

```
> df1$V21[df1$V21==1] <- 0
> df1$V21[df1$V21==2] <- 1
> df1_trim <- createDataPartition(df1$V21, times = 1, p = 0.8, list=FALSE)
> train <- df1[df1_trim,]
> valid <- df1[-df1_trim,]
> model1 <- glm(V21 ~ ., data = train, family=binomial(link="logit"))
> summary(model1)

Call:
glm(formula = V21 ~ ., family = binomial(link = "logit"), data = train)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-2.3512  -0.6945  -0.3484   0.7337   2.6693  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept)  1.432e+00  1.249e+00   1.147 0.251374    
V1A12       -5.023e-01  2.479e-01  -2.027 0.042692 *  
V1A13       -8.480e-01  4.015e-01  -2.112 0.034683 *  
V1A14       -1.909e+00  2.643e-01  -7.221 5.15e-13 ***
V2           3.596e-02  1.103e-02   3.261 0.001112 ** 
V3A31       -4.282e-01  6.539e-01  -0.655 0.512630    
V3A32       -1.053e+00  5.141e-01  -2.048 0.040598 *  
V3A33       -1.271e+00  5.608e-01  -2.266 0.023446 *  
V3A34       -1.855e+00  5.251e-01  -3.533 0.000412 ***
V4A41       -1.431e+00  4.154e-01  -3.445 0.000572 ***
V4A410      -1.375e+00  8.800e-01  -1.562 0.118259    
V4A42       -7.551e-01  2.981e-01  -2.533 0.011295 *  
V4A43       -8.477e-01  2.828e-01  -2.997 0.002723 ** 
V4A44       -1.186e-01  8.178e-01  -0.145 0.884721    
V4A45       -4.695e-01  6.211e-01  -0.756 0.449700    
V4A46        1.288e-01  4.464e-01   0.289 0.772872    
V4A48       -1.874e+00  1.230e+00  -1.523 0.127751    
V4A49       -3.306e-01  3.771e-01  -0.877 0.380596    
V5           1.004e-04  5.058e-05   1.985 0.047113 *  
V6A62       -4.162e-01  3.273e-01  -1.272 0.203472    
V6A63       -8.752e-01  5.012e-01  -1.746 0.080794 .  
V6A64       -1.418e+00  6.000e-01  -2.363 0.018108 *  
V6A65       -8.741e-01  2.964e-01  -2.949 0.003185 ** 
V7A72        1.672e-02  4.820e-01   0.035 0.972329    
V7A73       -2.073e-01  4.639e-01  -0.447 0.655034    
V7A74       -7.307e-01  5.039e-01  -1.450 0.147003    
V7A75       -3.347e-01  4.745e-01  -0.705 0.480543    
V8           2.834e-01  1.005e-01   2.818 0.004827 ** 
V9A92       -2.622e-01  4.391e-01  -0.597 0.550486    
V9A93       -6.109e-01  4.294e-01  -1.422 0.154882    
V9A94       -3.101e-01  5.088e-01  -0.609 0.542216    
V10A102      8.446e-01  4.696e-01   1.798 0.072102 .  
V10A103     -9.269e-01  4.640e-01  -1.998 0.045768 *  
V11         -5.709e-03  9.783e-02  -0.058 0.953465    
V12A122      2.243e-02  2.815e-01   0.080 0.936478    
V12A123      1.060e-01  2.630e-01   0.403 0.686869    
```

```
V12A124      2.405e-01  4.878e-01   0.493 0.622049
V13         -1.912e-02  1.060e-02  -1.804 0.071219 .
V14A142     -4.323e-01  4.514e-01  -0.958 0.338220
V14A143     -9.103e-01  2.765e-01  -3.292 0.000996 ***
V15A152     -3.546e-01  2.621e-01  -1.353 0.176071
V15A153     -3.018e-01  5.467e-01  -0.552 0.580935
V16          3.701e-01  2.156e-01   1.716 0.086118 .
V17A172      1.192e-01  7.498e-01   0.159 0.873714
V17A173      3.099e-01  7.223e-01   0.429 0.667875
V17A174      1.245e-01  7.399e-01   0.168 0.866401
V18          2.821e-01  2.790e-01   1.011 0.312099
V19A192     -3.380e-01  2.277e-01  -1.485 0.137668
V20A202     -1.634e+00  7.347e-01  -2.223 0.026195 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 982.41  on 799  degrees of freedom
Residual deviance: 707.93  on 751  degrees of freedom
AIC: 805.93

Number of Fisher Scoring iterations: 5
```

Make prediction on the validation dataset with .5 threshold. Result from confusion matrix suggests there are significant presence of false positive, hence misclassification on current logit fit. Choosing a right threshold may improve the model classification.

```
> df1_pred <- predict(model1, newdata=valid[,-21], type="response")
> table(valid$V21, round(df1_pred))

      0   1
  0 120  23
  1  30  27
```

Train and validation dataset are preprocessed fit the new logit model. Logit model is fitted using significant variables with .01 threshold. Summary of the model suggests all variables are significant.

```
> train$V1A14[train$V1 == "A14"] <- 1
> train$V1A14[train$V1 != "A14"] <- 0
> train$V3A34[train$V3 == "A34"] <- 1
> train$V3A34[train$V3 != "A34"] <- 0
> train$V4A41[train$V4 == "A41"] <- 1
> train$V4A41[train$V4 != "A41"] <- 0
> train$V4A43[train$V4 == "A43"] <- 1
> train$V4A43[train$V4 != "A43"] <- 0
> valid$V1A14[valid$V1 == "A14"] <- 1
> valid$V1A14[valid$V1 != "A14"] <- 0
> valid$V3A34[valid$V3 == "A34"] <- 1
> valid$V3A34[valid$V3 != "A34"] <- 0
> valid$V4A41[valid$V4 == "A41"] <- 1
> valid$V4A41[valid$V4 != "A41"] <- 0
> valid$V4A43[valid$V4 == "A43"] <- 1
> valid$V4A43[valid$V4 != "A43"] <- 0
> model2 <- glm(V21 ~ V1A14+V2+V3A34+V4A41+V4A43, data = train, family=bin
omial(link="logit"))
```

```
> summary(model2)

Call:
glm(formula = V21 ~ V1A14 + V2 + V3A34 + V4A41 + V4A43, family = binomial
(link = "logit"),
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7966  -0.8352  -0.4731   1.0011   2.6035

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.908817   0.192341  -4.725 2.30e-06 ***
V1A14       -1.690561   0.209742  -8.060 7.62e-16 ***
V2           0.047930   0.007556   6.343 2.25e-10 ***
V3A34       -0.645947   0.206352  -3.130 0.001746 **
V4A41       -1.111124   0.335142  -3.315 0.000915 ***
V4A43       -0.684534   0.205471  -3.332 0.000864 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 982.41  on 799  degrees of freedom
Residual deviance: 817.35  on 794  degrees of freedom
AIC: 829.35

Number of Fisher Scoring iterations: 5
```

With the new model, we first make prediction on the validation dataset with .5 threshold. Notice that the result is similar to the prediction made with the previous model. Since the cost of false positive is five times the cost of false negative, so after trying different threshold <u>by targeting to reduce false positive at the expense of false negative, the result with .7 threshold meets our expectation.</u> False positive decreases from 24 to 7, with false negative increases from 36 to 47.

```
> df1_pred2 <- predict(model2, newdata=valid[,-21], type="response")
> as.matrix(table(round(df1_pred2), valid$V21))

      0   1
  0 119  36
  1  24  21
> as.matrix(table(round(df1_pred2 > .7), valid$V21))

      0   1
  0 136  47
  1   7  10
```