HW 10.1

Yu Fung David Wang

10.1 (b)

```r
library(randomForest)
```

```
## randomForest 4.7-1.1

## Type rfNews() to see new features/changes/bug fixes.
```

```r
# import the dataset
crime_dataset <- read.table("uscrime.txt", header = TRUE)
crime_dataset
```

```
##       M So   Ed  Po1  Po2    LF   M.F Pop   NW    U1  U2 Wealth Ineq     Prob
## 1  15.1  1  9.1  5.8  5.6 0.510  95.0  33 30.1 0.108 4.1   3940 26.1 0.084602
## 2  14.3  0 11.3 10.3  9.5 0.583 101.2  13 10.2 0.096 3.6   5570 19.4 0.029599
## 3  14.2  1  8.9  4.5  4.4 0.533  96.9  18 21.9 0.094 3.3   3180 25.0 0.083401
## 4  13.6  0 12.1 14.9 14.1 0.577  99.4 157  8.0 0.102 3.9   6730 16.7 0.015801
## 5  14.1  0 12.1 10.9 10.1 0.591  98.5  18  3.0 0.091 2.0   5780 17.4 0.041399
## 6  12.1  0 11.0 11.8 11.5 0.547  96.4  25  4.4 0.084 2.9   6890 12.6 0.034201
## 7  12.7  1 11.1  8.2  7.9 0.519  98.2   4 13.9 0.097 3.8   6200 16.8 0.042100
## 8  13.1  1 10.9 11.5 10.9 0.542  96.9  50 17.9 0.079 3.5   4720 20.6 0.040099
## 9  15.7  1  9.0  6.5  6.2 0.553  95.5  39 28.6 0.081 2.8   4210 23.9 0.071697
## 10 14.0  0 11.8  7.1  6.8 0.632 102.9   7  1.5 0.100 2.4   5260 17.4 0.044498
## 11 12.4  0 10.5 12.1 11.6 0.580  96.6 101 10.6 0.077 3.5   6570 17.0 0.016201
## 12 13.4  0 10.8  7.5  7.1 0.595  97.2  47  5.9 0.083 3.1   5800 17.2 0.031201
## 13 12.8  0 11.3  6.7  6.0 0.624  97.2  28  1.0 0.077 2.5   5070 20.6 0.045302
## 14 13.5  0 11.7  6.2  6.1 0.595  98.6  22  4.6 0.077 2.7   5290 19.0 0.053200
## 15 15.2  1  8.7  5.7  5.3 0.530  98.6  30  7.2 0.092 4.3   4050 26.4 0.069100
## 16 14.2  1  8.8  8.1  7.7 0.497  95.6  33 32.1 0.116 4.7   4270 24.7 0.052099
## 17 14.3  0 11.0  6.6  6.3 0.537  97.7  10  0.6 0.114 3.5   4870 16.6 0.076299
## 18 13.5  1 10.4 12.3 11.5 0.537  97.8  31 17.0 0.089 3.4   6310 16.5 0.119804
## 19 13.0  0 11.6 12.8 12.8 0.536  93.4  51  2.4 0.078 3.4   6270 13.5 0.019099
## 20 12.5  0 10.8 11.3 10.5 0.567  98.5  78  9.4 0.130 5.8   6260 16.6 0.034801
## 21 12.6  0 10.8  7.4  6.7 0.602  98.4  34  1.2 0.102 3.3   5570 19.5 0.022800
## 22 15.7  1  8.9  4.7  4.4 0.512  96.2  22 42.3 0.097 3.4   2880 27.6 0.089502
## 23 13.2  0  9.6  8.7  8.3 0.564  95.3  43  9.2 0.083 3.2   5130 22.7 0.030700
## 24 13.1  0 11.6  7.8  7.3 0.574 103.8   7  3.6 0.142 4.2   5400 17.6 0.041598
## 25 13.0  0 11.6  6.3  5.7 0.641  98.4  14  2.6 0.070 2.1   4860 19.6 0.069197
## 26 13.1  0 12.1 16.0 14.3 0.631 107.1   3  7.7 0.102 4.1   6740 15.2 0.041698
## 27 13.5  0 10.9  6.9  7.1 0.540  96.5   6  0.4 0.080 2.2   5640 13.9 0.036099
## 28 15.2  0 11.2  8.2  7.6 0.571 101.8  10  7.9 0.103 2.8   5370 21.5 0.038201
## 29 11.9  0 10.7 16.6 15.7 0.521  93.8 168  8.9 0.092 3.6   6370 15.4 0.023400
## 30 16.6  1  8.9  5.8  5.4 0.521  97.3  46 25.4 0.072 2.6   3960 23.7 0.075298
## 31 14.0  0  9.3  5.5  5.4 0.535 104.5   6  2.0 0.135 4.0   4530 20.0 0.041999
## 32 12.5  0 10.9  9.0  8.1 0.586  96.4  97  8.2 0.105 4.3   6170 16.3 0.042698
## 33 14.7  1 10.4  6.3  6.4 0.560  97.2  23  9.5 0.076 2.4   4620 23.3 0.049499
```

```
## 34 12.6  0 11.8  9.7  9.7 0.542  99.0  18  2.1 0.102 3.5   5890 16.6 0.040799
## 35 12.3  0 10.2  9.7  8.7 0.526  94.8 113  7.6 0.124 5.0   5720 15.8 0.020700
## 36 15.0  0 10.0 10.9  9.8 0.531  96.4   9  2.4 0.087 3.8   5590 15.3 0.006900
## 37 17.7  1  8.7  5.8  5.6 0.638  97.4  24 34.9 0.076 2.8   3820 25.4 0.045198
## 38 13.3  0 10.4  5.1  4.7 0.599 102.4   7  4.0 0.099 2.7   4250 22.5 0.053998
## 39 14.9  1  8.8  6.1  5.4 0.515  95.3  36 16.5 0.086 3.5   3950 25.1 0.047099
## 40 14.5  1 10.4  8.2  7.4 0.560  98.1  96 12.6 0.088 3.1   4880 22.8 0.038801
## 41 14.8  0 12.2  7.2  6.6 0.601  99.8   9  1.9 0.084 2.0   5900 14.4 0.025100
## 42 14.1  0 10.9  5.6  5.4 0.523  96.8   4  0.2 0.107 3.7   4890 17.0 0.088904
## 43 16.2  1  9.9  7.5  7.0 0.522  99.6  40 20.8 0.073 2.7   4960 22.4 0.054902
## 44 13.6  0 12.1  9.5  9.6 0.574 101.2  29  3.6 0.111 3.7   6220 16.2 0.028100
## 45 13.9  1  8.8  4.6  4.1 0.480  96.8  19  4.9 0.135 5.3   4570 24.9 0.056202
## 46 12.6  0 10.4 10.6  9.7 0.599  98.9  40  2.4 0.078 2.5   5930 17.1 0.046598
## 47 13.0  0 12.1  9.0  9.1 0.623 104.9   3  2.2 0.113 4.0   5880 16.0 0.052802
##       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
## 7  20.6993   963
## 8  24.5988  1555
## 9  29.4001   856
## 10 19.5994   705
## 11 41.6000  1674
## 12 34.2984   849
## 13 36.2993   511
## 14 21.5010   664
## 15 22.7008   798
## 16 26.0991   946
## 17 19.1002   539
## 18 18.1996   929
## 19 24.9008   750
## 20 26.4010  1225
## 21 37.5998   742
## 22 37.0994   439
## 23 25.1989  1216
## 24 17.6000   968
## 25 21.9003   523
## 26 22.1005  1993
## 27 28.4999   342
## 28 25.8006  1216
## 29 36.7009  1043
## 30 28.3011   696
## 31 21.7998   373
## 32 30.9014   754
## 33 25.5005  1072
## 34 21.6997   923
## 35 37.4011   653
## 36 44.0004  1272
## 37 31.6995   831
## 38 16.6999   566
## 39 27.3004   826
```

```
## 40 29.3004  1151
## 41 30.0001   880
## 42 12.1996   542
## 43 31.9989   823
## 44 30.0001  1030
## 45 32.5996   455
## 46 16.6999   508
## 47 16.0997   849
```

```r
# set train and test dataset, 70% as train, 30% as test
train <- sample(1:nrow(crime_dataset), size = floor(0.7*nrow(crime_dataset)), replace = FALSE, prob = r
train
```

```
##  [1] 45 27  1  5 29 34 47 38 22 46 44 21 28 26  6 39 20 35 31  3 24 14 30 37 40
## [26] 10  8 33 12 16  2  9
```

```r
crime_train <- crime_dataset[train,] # train
crime_test <- crime_dataset[-train,] # test
crime_train
```

```
##       M So   Ed  Po1  Po2    LF   M.F Pop   NW    U1  U2 Wealth Ineq     Prob
## 45 13.9  1  8.8  4.6  4.1 0.480  96.8  19  4.9 0.135 5.3   4570 24.9 0.056202
## 27 13.5  0 10.9  6.9  7.1 0.540  96.5   6  0.4 0.080 2.2   5640 13.9 0.036099
## 1  15.1  1  9.1  5.8  5.6 0.510  95.0  33 30.1 0.108 4.1   3940 26.1 0.084602
## 5  14.1  0 12.1 10.9 10.1 0.591  98.5  18  3.0 0.091 2.0   5780 17.4 0.041399
## 29 11.9  0 10.7 16.6 15.7 0.521  93.8 168  8.9 0.092 3.6   6370 15.4 0.023400
## 34 12.6  0 11.8  9.7  9.7 0.542  99.0  18  2.1 0.102 3.5   5890 16.6 0.040799
## 47 13.0  0 12.1  9.0  9.1 0.623 104.9   3  2.2 0.113 4.0   5880 16.0 0.052802
## 38 13.3  0 10.4  5.1  4.7 0.599 102.4   7  4.0 0.099 2.7   4250 22.5 0.053998
## 22 15.7  1  8.9  4.7  4.4 0.512  96.2  22 42.3 0.097 3.4   2880 27.6 0.089502
## 46 12.6  0 10.4 10.6  9.7 0.599  98.9  40  2.4 0.078 2.5   5930 17.1 0.046598
## 44 13.6  0 12.1  9.5  9.6 0.574 101.2  29  3.6 0.111 3.7   6220 16.2 0.028100
## 21 12.6  0 10.8  7.4  6.7 0.602  98.4  34  1.2 0.102 3.3   5570 19.5 0.022800
## 28 15.2  0 11.2  8.2  7.6 0.571 101.8  10  7.9 0.103 2.8   5370 21.5 0.038201
## 26 13.1  0 12.1 16.0 14.3 0.631 107.1   3  7.7 0.102 4.1   6740 15.2 0.041698
## 6  12.1  0 11.0 11.8 11.5 0.547  96.4  25  4.4 0.084 2.9   6890 12.6 0.034201
## 39 14.9  1  8.8  6.1  5.4 0.515  95.3  36 16.5 0.086 3.5   3950 25.1 0.047099
## 20 12.5  0 10.8 11.3 10.5 0.567  98.5  78  9.4 0.130 5.8   6260 16.6 0.034801
## 35 12.3  0 10.2  9.7  8.7 0.526  94.8 113  7.6 0.124 5.0   5720 15.8 0.020700
## 31 14.0  0  9.3  5.5  5.4 0.535 104.5   6  2.0 0.135 4.0   4530 20.0 0.041999
## 3  14.2  1  8.9  4.5  4.4 0.533  96.9  18 21.9 0.094 3.3   3180 25.0 0.083401
## 24 13.1  0 11.6  7.8  7.3 0.574 103.8   7  3.6 0.142 4.2   5400 17.6 0.041598
## 14 13.5  0 11.7  6.2  6.1 0.595  98.6  22  4.6 0.077 2.7   5290 19.0 0.053200
## 30 16.6  1  8.9  5.8  5.4 0.521  97.3  46 25.4 0.072 2.6   3960 23.7 0.075298
## 37 17.7  1  8.7  5.8  5.6 0.638  97.4  24 34.9 0.076 2.8   3820 25.4 0.045198
## 40 14.5  1 10.4  8.2  7.4 0.560  98.1  96 12.6 0.088 3.1   4880 22.8 0.038801
## 10 14.0  0 11.8  7.1  6.8 0.632 102.9   7  1.5 0.100 2.4   5260 17.4 0.044498
## 8  13.1  1 10.9 11.5 10.9 0.542  96.9  50 17.9 0.079 3.5   4720 20.6 0.040099
## 33 14.7  1 10.4  6.3  6.4 0.560  97.2  23  9.5 0.076 2.4   4620 23.3 0.049499
## 12 13.4  0 10.8  7.5  7.1 0.595  97.2  47  5.9 0.083 3.1   5800 17.2 0.031201
## 16 14.2  1  8.8  8.1  7.7 0.497  95.6  33 32.1 0.116 4.7   4270 24.7 0.052099
## 2  14.3  0 11.3 10.3  9.5 0.583 101.2  13 10.2 0.096 3.6   5570 19.4 0.029599
## 9  15.7  1  9.0  6.5  6.2 0.553  95.5  39 28.6 0.081 2.8   4210 23.9 0.071697
##       Time Crime
## 45 32.5996   455
## 27 28.4999   342
```

```
## 1  26.2011   791
## 5  21.2998  1234
## 29 36.7009  1043
## 34 21.6997   923
## 47 16.0997   849
## 38 16.6999   566
## 22 37.0994   439
## 46 16.6999   508
## 44 30.0001  1030
## 21 37.5998   742
## 28 25.8006  1216
## 26 22.1005  1993
## 6  20.9995   682
## 39 27.3004   826
## 20 26.4010  1225
## 35 37.4011   653
## 31 21.7998   373
## 3  24.3006   578
## 24 17.6000   968
## 14 21.5010   664
## 30 28.3011   696
## 37 31.6995   831
## 40 29.3004  1151
## 10 19.5994   705
## 8  24.5988  1555
## 33 25.5005  1072
## 12 34.2984   849
## 16 26.0991   946
## 2  25.2999  1635
## 9  29.4001   856
```

```
crime_test
```

```
##       M So   Ed  Po1  Po2    LF  M.F Pop   NW    U1  U2 Wealth Ineq     Prob
## 4  13.6  0 12.1 14.9 14.1 0.577 99.4 157  8.0 0.102 3.9   6730 16.7 0.015801
## 7  12.7  1 11.1  8.2  7.9 0.519 98.2   4 13.9 0.097 3.8   6200 16.8 0.042100
## 11 12.4  0 10.5 12.1 11.6 0.580 96.6 101 10.6 0.077 3.5   6570 17.0 0.016201
## 13 12.8  0 11.3  6.7  6.0 0.624 97.2  28  1.0 0.077 2.5   5070 20.6 0.045302
## 15 15.2  1  8.7  5.7  5.3 0.530 98.6  30  7.2 0.092 4.3   4050 26.4 0.069100
## 17 14.3  0 11.0  6.6  6.3 0.537 97.7  10  0.6 0.114 3.5   4870 16.6 0.076299
## 18 13.5  1 10.4 12.3 11.5 0.537 97.8  31 17.0 0.089 3.4   6310 16.5 0.119804
## 19 13.0  0 11.6 12.8 12.8 0.536 93.4  51  2.4 0.078 3.4   6270 13.5 0.019099
## 23 13.2  0  9.6  8.7  8.3 0.564 95.3  43  9.2 0.083 3.2   5130 22.7 0.030700
## 25 13.0  0 11.6  6.3  5.7 0.641 98.4  14  2.6 0.070 2.1   4860 19.6 0.069197
## 32 12.5  0 10.9  9.0  8.1 0.586 96.4  97  8.2 0.105 4.3   6170 16.3 0.042698
## 36 15.0  0 10.0 10.9  9.8 0.531 96.4   9  2.4 0.087 3.8   5590 15.3 0.006900
## 41 14.8  0 12.2  7.2  6.6 0.601 99.8   9  1.9 0.084 2.0   5900 14.4 0.025100
## 42 14.1  0 10.9  5.6  5.4 0.523 96.8   4  0.2 0.107 3.7   4890 17.0 0.088904
## 43 16.2  1  9.9  7.5  7.0 0.522 99.6  40 20.8 0.073 2.7   4960 22.4 0.054902
##       Time Crime
## 4  29.9012  1969
## 7  20.6993   963
## 11 41.6000  1674
## 13 36.2993   511
## 15 22.7008   798
```

```
## 17 19.1002   539
## 18 18.1996   929
## 19 24.9008   750
## 23 25.1989  1216
## 25 21.9003   523
## 32 30.9014   754
## 36 44.0004  1272
## 41 30.0001   880
## 42 12.1996   542
## 43 31.9989   823
```

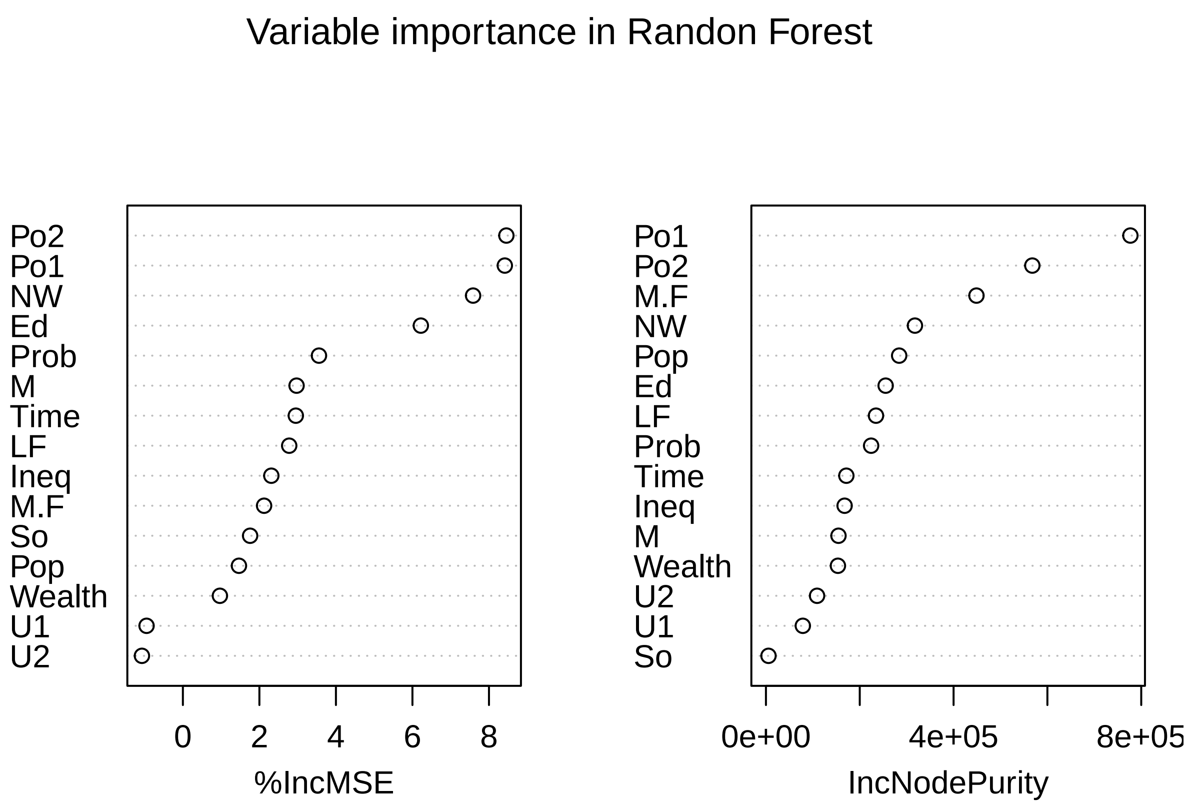
```
# Setup the random forest model
rf_model <- randomForest(Crime~., data = crime_train, importance=TRUE)
print(rf_model)
```

```
##
## Call:
##  randomForest(formula = Crime ~ ., data = crime_train, importance = TRUE)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 5
##
##           Mean of squared residuals: 101226.1
##                     % Var explained: 24.31
```

```
# get the importance of each independent variable
importance(rf_model)
```

```
##           %IncMSE IncNodePurity
## M       2.9712328    154642.022
## So      1.7575320      5487.329
## Ed      6.2195108    255204.930
## Po1     8.4140689    776898.382
## Po2     8.4548784    567853.034
## LF      2.7792085    234608.728
## M.F     2.1222664    448751.905
## Pop     1.4655741    284013.373
## NW      7.5846707    317640.815
## U1     -0.9491941     78621.611
## U2     -1.0706394    109003.352
## Wealth  0.9677632    153428.074
## Ineq    2.3113879    167742.712
## Prob    3.5567894    224227.974
## Time    2.9521419    171383.051
```

```
# using graph show the importance of each independent variable
varImpPlot(rf_model, main = "Variable importance in Randon Forest")
```



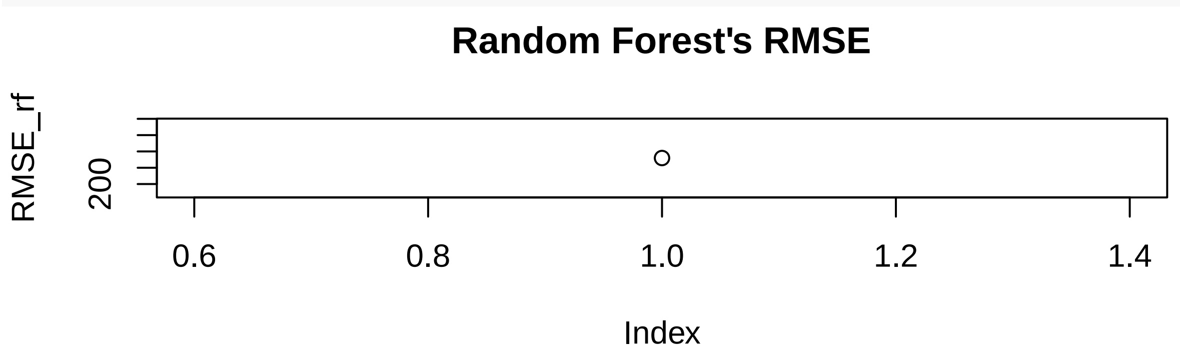

```r
# predict
predict_rf <- predict(rf_model, crime_test)
predict_rf
```

```
##         4         7        11        13        15        17        18        19
## 1293.0589 1003.9858 1127.4177  761.0796  736.2457  697.8636 1046.8947 1018.8984
##        23        25        32        36        41        42        43
## 1017.7701  743.6295  981.9428  976.3890  837.4165  636.3823  852.8904
```

```r
RMSE_rf <- sqrt(mean((crime_test$Crime-predict_rf)^2))
RMSE_rf
```

```
## [1] 279.7685
```

```r
# plot the Random Forest's R^2 and RMSE
par(mfrow=c(2, 1))
plot(RMSE_rf, main="Random Forest's RMSE")
```



This is a random forest regression model with 500 trees. Each split has a random subset of 5 variables considered. 34.17% of the variability in crime was explained by the random forest. The variable that is most important is

Po1 and in relative importance, the others are U2, M, Wealth, and Pop. Po1 is police expenditure, which supports the result from the above tree in section 10.1. Police expenditures also account for the highest %IncMSE and IncNodePurity. This highly suggests that police expenditures have a large impact on crime rates. The next best model is previous crime history, which is also the second best model in the trees above. The RMSE is 186.85, which is more accurate than the trees above. Thus, the random forest outperforms the regression tree models.