

YK_Final_P2

Yinan Kang

5/13/2019

```
require(dplyr)
require(leaps)
require(neuralnet)
require(readxl)
require(caret)
require(car)
require(rpart)
```

Problem 2

Load Data

```
rm(list=ls())
df <- read.csv("/cloud/project/Question 2.csv")
```

Split Data

```
set.seed(12345)
trainIndex <- createDataPartition(df$Y, p=0.7, list=FALSE)

train.df <- df[trainIndex,]
test.df <- df[-trainIndex,]
```

Fit Model with Development sample (a.k.a. ‘train.df’)

(a)

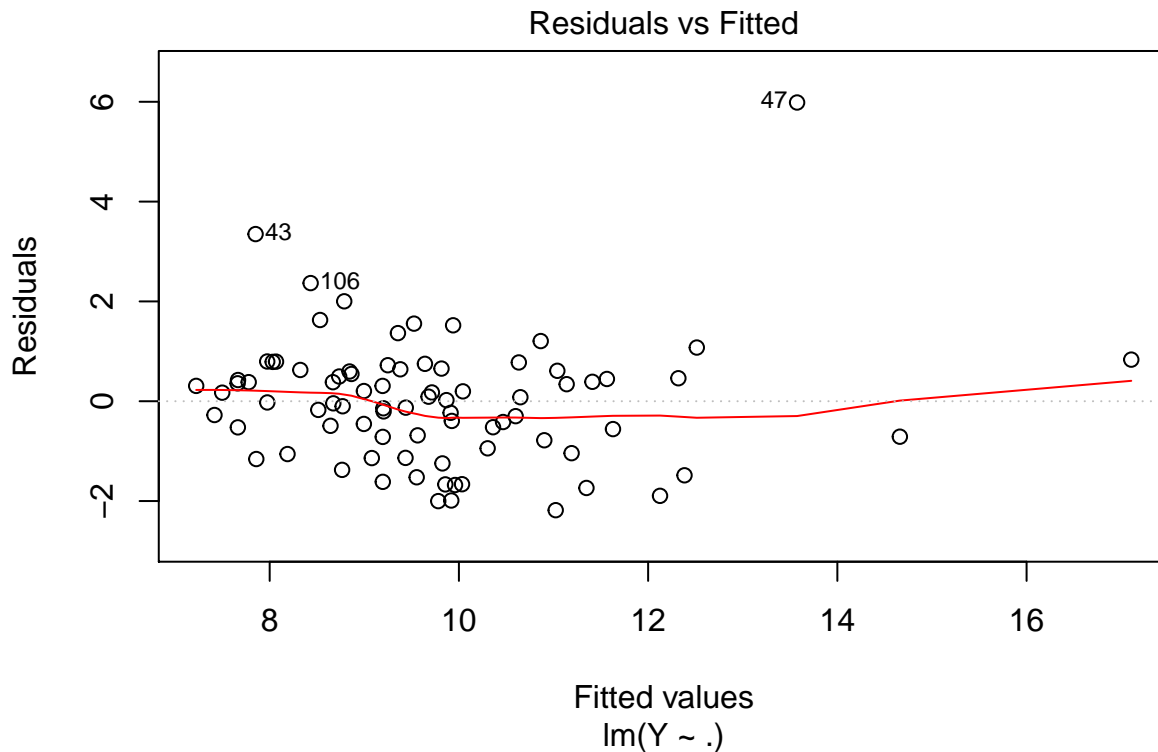
```
model <- lm(Y~. , data=train.df)

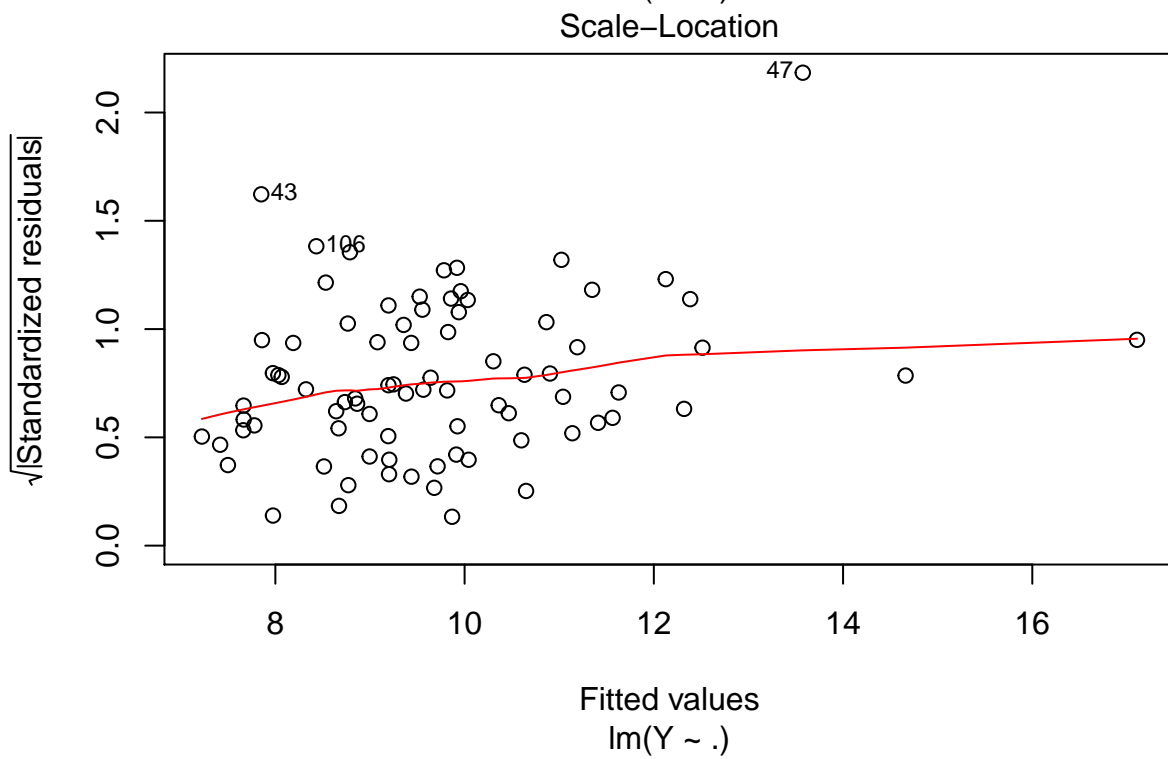
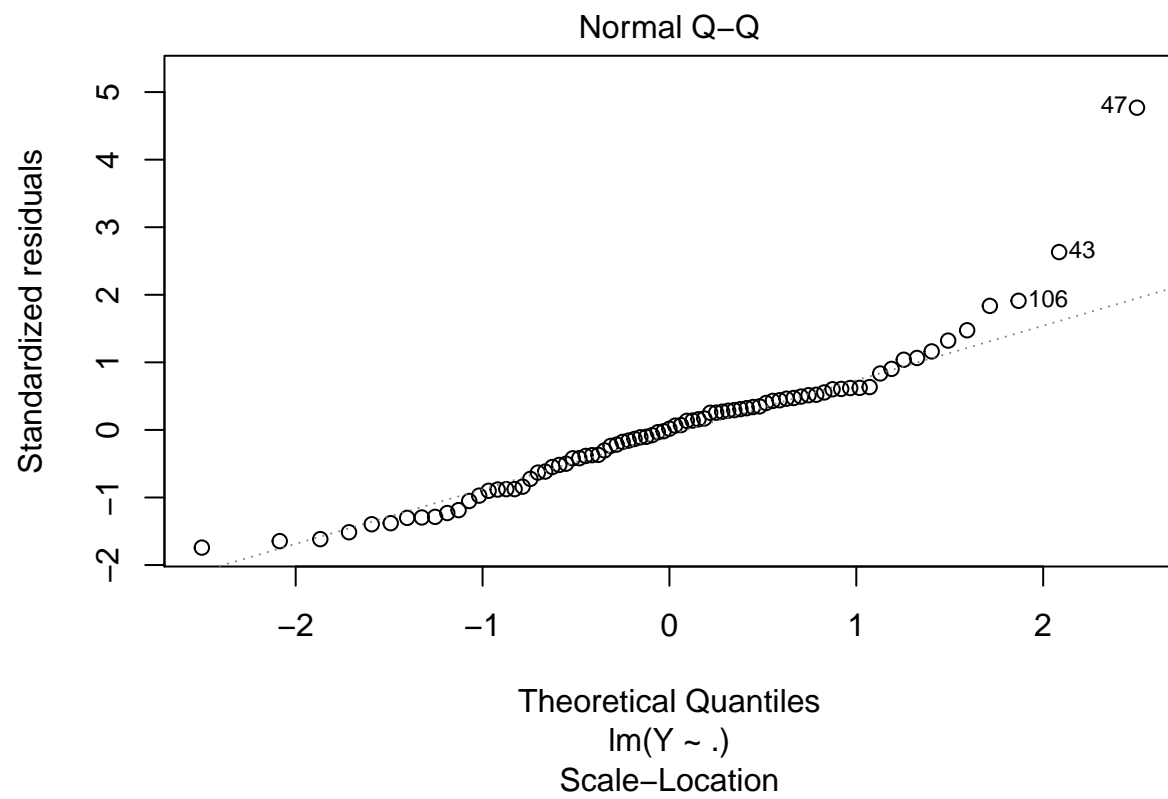
# Residual Analysis and Model Significance
summary(model)
```

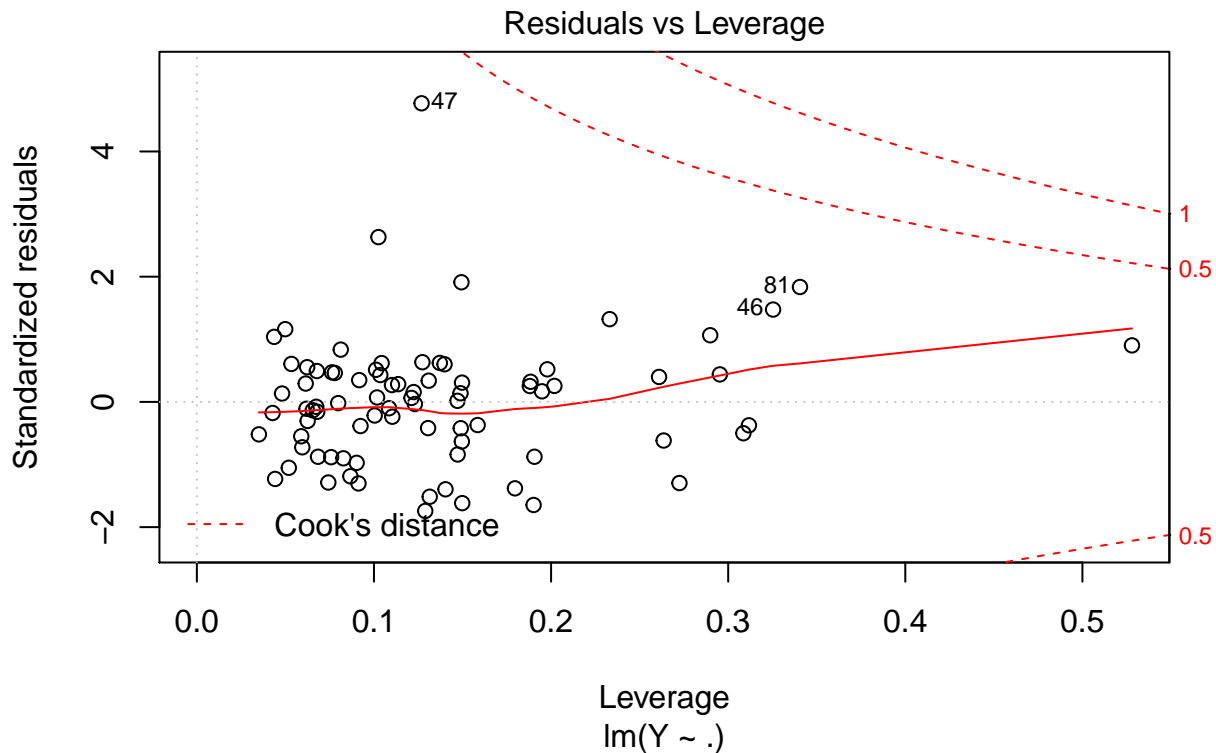
```
##
## Call:
## lm(formula = Y ~ ., data = train.df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.1832 -0.7152  0.0220  0.6097  5.9856
##
## Coefficients:
```

```
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.5097992  2.5712076   0.976  0.33237
## X1           0.0851752  0.0355052   2.399  0.01911 *
## X2           0.4778835  0.1740312   2.746  0.00766 **
## X3           0.0006345  0.0221793   0.029  0.97726
## X4           0.0132255  0.0089821   1.472  0.14539
## X5          -0.0103232  0.0057853  -1.784  0.07869 .
## X6          -0.1159227  0.5899833  -0.196  0.84480
## X7          -0.4528421  0.1938593  -2.336  0.02236 *
## X8           0.0216602  0.0063384   3.417  0.00106 **
## X9          -0.0065853  0.0031066  -2.120  0.03757 *
## X10          0.0078747  0.0191298   0.412  0.68186
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.343 on 70 degrees of freedom
## Multiple R-squared:  0.6363, Adjusted R-squared:  0.5843
## F-statistic: 12.24 on 10 and 70 DF,  p-value: 6.101e-12
```

```
plot(model)
```







```
# Multicollinearity
vif(model)
```

```
##          X1          X2          X3          X4          X5          X6          X7
##  1.105343  2.405521  2.186076  1.570451  53.297022  2.234402  1.560107
##          X8          X9          X10
## 43.156298  8.085746  3.734094
```

```
mean(vif(model))
```

```
## [1] 11.93351
```

Analysis of (a): Upon seeing the overall summary model statistics, we find that overall, the model has a p-value = $6.101e-12$, so overall the model IS statistically significant. The adj-R² = 0.5843, which (without knowing context of dataset) we can generally say is between OK and mediocre variance accountability.

We see from the residual plot and the QQ-Normality plot that: * there is no severe normality issue (though a few noticeable observations - #43, #47)

* overall there is constant variance of the residuals

Multicollinearity: From the Variance Inflation Factor analysis, we find that there is relatively severe multicollinearity with the current model. Predictors X5 and X8 have VIF values >10, and overall the mean VIF = 11.93, which is >1 (benchmark).

(b) Outliers and Influentials

```
summary(influence.measures(model))
```

```
## Potentially influential observations of
## lm(formula = Y ~ ., data = train.df) :
##
```

```
##      dfb.1_ dfb.X1 dfb.X2 dfb.X3 dfb.X4 dfb.X5 dfb.X6 dfb.X7 dfb.X8 dfb.X9
## 11  0.06 -0.05  0.15 -0.09 -0.11 -0.03 -0.04  0.04  0.05 -0.17
## 13  0.00  0.08  0.06  0.05 -0.03  0.01 -0.11 -0.06 -0.05  0.06
## 43  0.49 -0.72 -0.14 -0.26  0.25 -0.14  0.02  0.18  0.17 -0.22
## 47 -0.83  0.66  0.89 -0.72  0.51 -0.43  0.31 -0.47  0.56 -0.33
## 52 -0.04  0.00 -0.01  0.05 -0.11  0.15  0.12 -0.05 -0.15  0.03
## 74 -0.05  0.00  0.04 -0.08  0.03  0.02  0.13  0.00  0.00  0.07
## 81  0.80 -0.59  0.08 -0.32 -0.13  0.49 -0.71 -0.20 -0.43 -0.76
## 104 0.23 -0.17 -0.06  0.07 -0.16  0.03 -0.06 -0.01 -0.06  0.10
## 112 -0.08  0.12 -0.15  0.22 -0.03 -0.33  0.00  0.05  0.60 -0.29
##      dfb.X10 dffit   cov.r   cook.d hat
## 11  0.04 -0.33   1.63_*  0.01  0.31
## 13 -0.04  0.24   1.55_*  0.01  0.26
## 43  0.22  0.93   0.41_*  0.07  0.10
## 47  0.21  2.20_*  0.02_*  0.30  0.13
## 52  0.01  0.28   1.61_*  0.01  0.30
## 74 -0.14 -0.25   1.67_*  0.01  0.31
## 81  0.37  1.34_*  1.03   0.16  0.34
## 104 -0.12 -0.37   1.50_*  0.01  0.26
## 112 -0.09  0.95   2.18_*  0.08  0.53_*
```

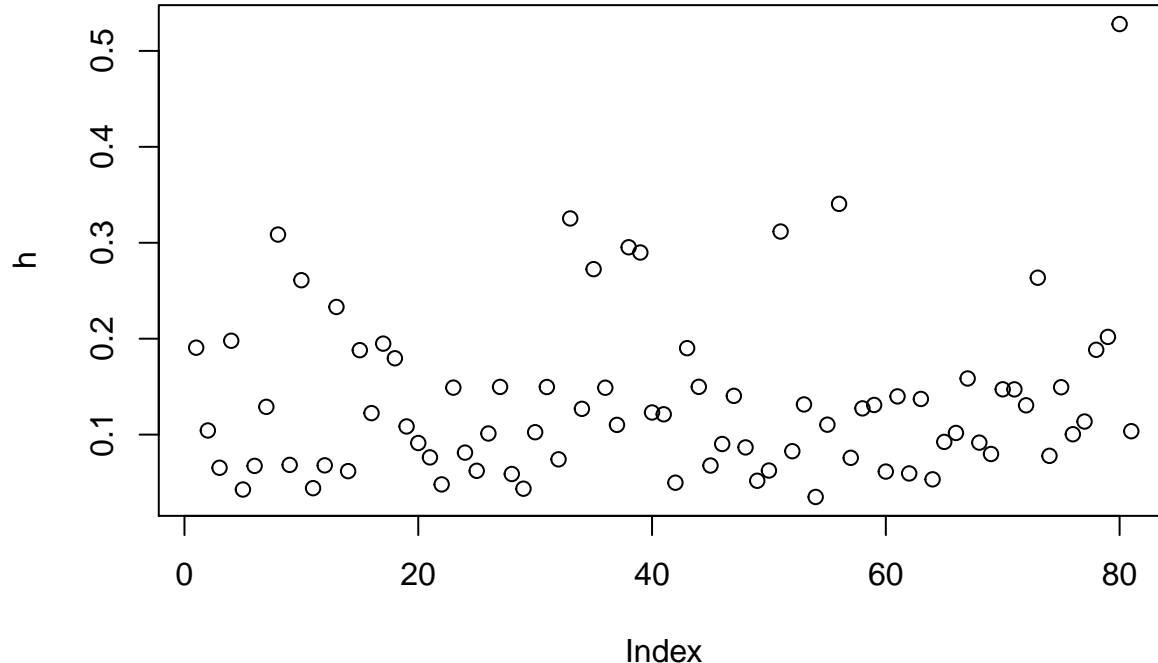
```
# For hat values: 2p/n
```

```
p <- 10
```

```
n <- nrow(train.df)
```

```
h <- lm.influence(model)$hat
```

```
plot(h)
```



Influential Analysis: We displayed the key influential determination statistics (Cook's Distance, DFFITS, Hat, etc.). We also plotted the Hat values for a visual comparison.

The approximate threshold to be considered a highly influential case for Hat values was ' $2p/n$ ' = 0.25. Some observations crossed this threshold slightly, but only Case 112 truly did (noticeable from plot as well, where it's Hat=0.53). For DFFITS, the threshold is '1', but those that crossed this threshold did not show high

leverage in the other metrics.

Thus, I would point to Case 112 as the only really influential case.

(c) Can X5, X6, X7 be dropped?

Note: I am assuming question is asking can X5, X6, X7 be dropped together from the model, as opposed to one-by-one

```
model.light <- lm(Y~X1+X2+X3+X4+X8+X9+X10, data=train.df)
anova(model.light, model)

## Analysis of Variance Table
##
## Model 1: Y ~ X1 + X2 + X3 + X4 + X8 + X9 + X10
## Model 2: Y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + X9 + X10
##   Res.Df    RSS Df Sum of Sq    F Pr(>F)
## 1      73 145.63
## 2      70 126.28  3    19.347 3.5748 0.01816 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Analysis: If we go by the $\alpha = 0.05$ level... NO, we can't drop X5,X6,X7 as p-value < 0.05 , indicating statistical significance.

(But, if we were to utilize the $\alpha = 0.01$ level, then the opposite conclusion would be valid.)

(d) Regression Tree

```
model.reg <- rpart(Y~., data=train.df)
summary(model.reg)

## Call:
## rpart(formula = Y ~ ., data = train.df)
##   n= 81
##
##           CP nsplit rel error   xerror   xstd
## 1 0.38600137     0 1.0000000 1.0264650 0.3458540
## 2 0.09857913     1 0.6139986 0.9862689 0.2675827
## 3 0.03068474     2 0.5154195 0.7569987 0.2040399
## 4 0.01534651     3 0.4847348 0.7790316 0.2047835
## 5 0.01201516     5 0.4540417 0.8028815 0.2049651
## 6 0.01000000     6 0.4420266 0.8013606 0.2049183
##
## Variable importance
##  X2  X3  X5  X9  X8 X10  X1  X4  X7
##  44  12  12  10   9   7   2   2   1
##
## Node number 1: 81 observations,    complexity param=0.3860014
##   mean=9.724568, MSE=4.286047
##   left son=2 (73 obs) right son=3 (8 obs)
##   Primary splits:
##       X2 < 5.85 to the left, improve=0.3860014, (0 missing)
```

```

##      X5 < 305   to the left,  improve=0.2580972, (0 missing)
##      X8 < 250   to the left,  improve=0.2571419, (0 missing)
##     X10 < 50    to the left,  improve=0.2402275, (0 missing)
##      X9 < 130.5 to the left,  improve=0.2019305, (0 missing)
## Surrogate splits:
##      X3 < 44    to the left,  agree=0.926, adj=0.25, (0 split)
##
## Node number 2: 73 observations,      complexity param=0.09857913
## mean=9.298767, MSE=1.659033
## left son=4 (24 obs) right son=5 (49 obs)
## Primary splits:
##      X5 < 122.5 to the left,  improve=0.2825850, (0 missing)
##      X9 < 99.5  to the left,  improve=0.2800819, (0 missing)
##      X8 < 66.5  to the left,  improve=0.2545677, (0 missing)
##     X10 < 30    to the left,  improve=0.1981774, (0 missing)
##      X2 < 4.25  to the left,  improve=0.1954625, (0 missing)
## Surrogate splits:
##      X8 < 94    to the left,  agree=0.945, adj=0.833, (0 split)
##      X9 < 83    to the left,  agree=0.918, adj=0.750, (0 split)
##     X10 < 35.7  to the left,  agree=0.863, adj=0.583, (0 split)
##      X2 < 2.85  to the left,  agree=0.795, adj=0.375, (0 split)
##      X1 < 55.95 to the right, agree=0.726, adj=0.167, (0 split)
##
## Node number 3: 8 observations
## mean=13.61, MSE=11.50655
##
## Node number 4: 24 observations,      complexity param=0.01201516
## mean=8.320417, MSE=0.8548457
## left son=8 (16 obs) right son=9 (8 obs)
## Primary splits:
##      X3 < 15.8  to the left,  improve=0.20331650, (0 missing)
##     X10 < 30    to the left,  improve=0.15790380, (0 missing)
##      X8 < 60    to the left,  improve=0.14193620, (0 missing)
##      X1 < 52.75 to the left,  improve=0.10585500, (0 missing)
##      X2 < 2.75  to the left,  improve=0.07611349, (0 missing)
## Surrogate splits:
##      X1 < 50    to the right, agree=0.708, adj=0.125, (0 split)
##      X4 < 81.65 to the left,  agree=0.708, adj=0.125, (0 split)
##      X7 < 1.5   to the right, agree=0.708, adj=0.125, (0 split)
##      X9 < 73    to the left,  agree=0.708, adj=0.125, (0 split)
##     X10 < 32.85 to the left,  agree=0.708, adj=0.125, (0 split)
##
## Node number 5: 49 observations,      complexity param=0.03068474
## mean=9.777959, MSE=1.354477
## left son=10 (20 obs) right son=11 (29 obs)
## Primary splits:
##      X2 < 4.35  to the left,  improve=0.16050800, (0 missing)
##      X4 < 96    to the left,  improve=0.10494870, (0 missing)
##      X8 < 212   to the left,  improve=0.09400911, (0 missing)
##      X5 < 183   to the left,  improve=0.08514898, (0 missing)
##      X7 < 2.5   to the right, improve=0.07494387, (0 missing)
## Surrogate splits:
##      X3 < 7.85  to the left,  agree=0.735, adj=0.35, (0 split)
##      X9 < 127   to the left,  agree=0.735, adj=0.35, (0 split)

```

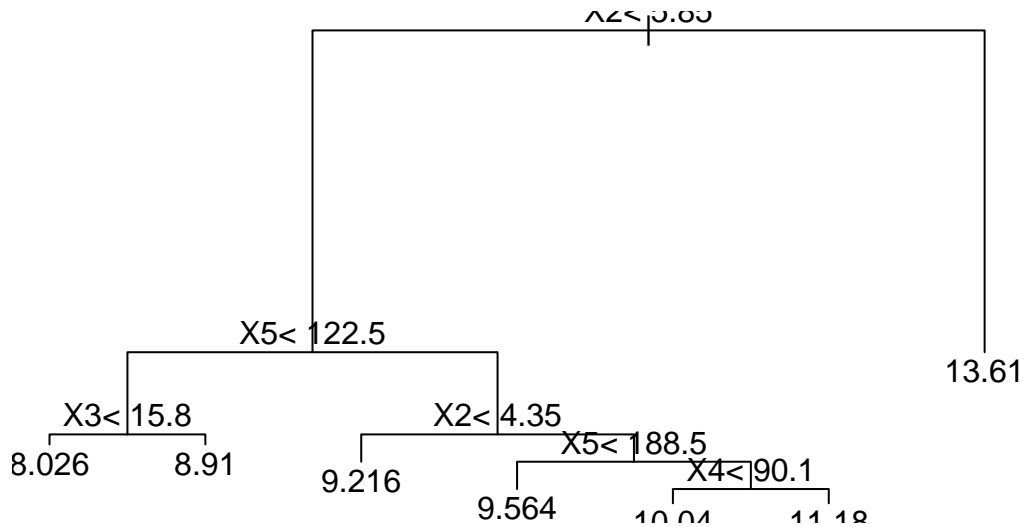
```

##      X1 < 45.35 to the left,  agree=0.653, adj=0.15, (0 split)
##      X4 < 57.95 to the left,  agree=0.653, adj=0.15, (0 split)
##      X5 < 138.5 to the left,  agree=0.653, adj=0.15, (0 split)
##
## Node number 8: 16 observations
##   mean=8.025625, MSE=0.4226621
##
## Node number 9: 8 observations
##   mean=8.91, MSE=1.1978
##
## Node number 10: 20 observations
##   mean=9.2165, MSE=1.313343
##
## Node number 11: 29 observations,    complexity param=0.01534651
##   mean=10.16517, MSE=1.015508
##   left son=22 (9 obs) right son=23 (20 obs)
##   Primary splits:
##     X5 < 188.5 to the left,  improve=0.1599134, (0 missing)
##     X8 < 152   to the left,  improve=0.1599134, (0 missing)
##     X2 < 5.4   to the left,  improve=0.1343526, (0 missing)
##     X10 < 50   to the left,  improve=0.1336582, (0 missing)
##     X4 < 94.55 to the left,  improve=0.1310009, (0 missing)
##   Surrogate splits:
##     X8 < 152   to the left,  agree=1.000, adj=1.000, (0 split)
##     X9 < 164   to the left,  agree=0.897, adj=0.667, (0 split)
##     X10 < 41.45 to the left,  agree=0.793, adj=0.333, (0 split)
##     X1 < 57.7  to the right, agree=0.759, adj=0.222, (0 split)
##     X3 < 25.8  to the right, agree=0.724, adj=0.111, (0 split)
##
## Node number 22: 9 observations
##   mean=9.564444, MSE=0.2281802
##
## Node number 23: 20 observations,    complexity param=0.01534651
##   mean=10.4355, MSE=1.134335
##   left son=46 (13 obs) right son=47 (7 obs)
##   Primary splits:
##     X4 < 90.1  to the left,  improve=0.26210460, (0 missing)
##     X1 < 52.65 to the left,  improve=0.19582880, (0 missing)
##     X7 < 2.5   to the right, improve=0.09871465, (0 missing)
##     X10 < 50   to the left,  improve=0.06785769, (0 missing)
##     X2 < 5.1   to the left,  improve=0.06575969, (0 missing)
##   Surrogate splits:
##     X9 < 176   to the right, agree=0.80, adj=0.429, (0 split)
##     X10 < 40    to the right, agree=0.80, adj=0.429, (0 split)
##     X3 < 26.15 to the left,  agree=0.75, adj=0.286, (0 split)
##     X5 < 202.5 to the right, agree=0.75, adj=0.286, (0 split)
##     X7 < 1.5   to the right, agree=0.75, adj=0.286, (0 split)
##
## Node number 46: 13 observations
##   mean=10.03538, MSE=1.009733
##
## Node number 47: 7 observations
##   mean=11.17857, MSE=0.5162694

```



```
plot(model.reg)
text(model.reg)
```



(d) and (e) Using holdout sample to score the model, and also to compare with regression tree model

```
model.pred <- predict(model,newdata=test.df)
model.ei <- model.pred - test.df$Y

tree.pred <- predict(model.reg, newdata=test.df)
tree.ei <- tree.pred - test.df$Y

# Using MAE as criteria
mean(abs(model.ei))
```

```
## [1] 0.8360067
```

```
mean(abs(tree.ei))
```

```
## [1] 1.208188
```

(d) **second part:** We see that, using mean absolute error as the criteria, the linear model in (a) performed better than the regression tree model.

Re-calibrate model and compare with model in (c)

Confused by the wording... I am going to re-calibrate model from (a) with holdout data to detect stability, then fit holdout data with model in (c) and compare.

[Main confusion... what is meant by ‘final model’ in (c)? We never decided (c) was ‘final’ model.]

```
model.end <- lm(Y~., data=test.df)
summary(model.end) # Holdout
```

```
##
## Call:
## lm(formula = Y ~ ., data = test.df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.1463 -0.4101  0.0541  0.2869  2.0243
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.484464   2.645779   2.451 0.023096 *
## X1           0.065061   0.037822   1.720 0.100108
## X2           0.378929   0.150200   2.523 0.019777 *
## X3           0.017632   0.021746   0.811 0.426563
## X4           0.003781   0.011132   0.340 0.737524
## X5           0.001210   0.003670   0.330 0.744929
## X6          -0.121216   0.714414  -0.170 0.866892
## X7          -0.676796   0.146053  -4.634 0.000143 ***
## X8           0.004417   0.006257   0.706 0.487951
## X9          -0.002534   0.003387  -0.748 0.462671
## X10          -0.035624   0.017592  -2.025 0.055772 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8064 on 21 degrees of freedom
## Multiple R-squared:  0.7738, Adjusted R-squared:  0.6661
## F-statistic: 7.185 on 10 and 21 DF, p-value: 7.767e-05
```

```
summary(model) # Original
```

```
##
## Call:
## lm(formula = Y ~ ., data = train.df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.1832 -0.7152  0.0220  0.6097  5.9856
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.5097992  2.5712076   0.976 0.33237
## X1           0.0851752  0.0355052   2.399 0.01911 *
## X2           0.4778835  0.1740312   2.746 0.00766 **
## X3           0.0006345  0.0221793   0.029 0.97726
## X4           0.0132255  0.0089821   1.472 0.14539
## X5          -0.0103232  0.0057853  -1.784 0.07869 .
## X6          -0.1159227  0.5899833  -0.196 0.84480
## X7          -0.4528421  0.1938593  -2.336 0.02236 *
## X8           0.0216602  0.0063384   3.417 0.00106 **
## X9          -0.0065853  0.0031066  -2.120 0.03757 *
## X10          0.0078747  0.0191298   0.412 0.68186
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.343 on 70 degrees of freedom
```

```
## Multiple R-squared:  0.6363, Adjusted R-squared:  0.5843
## F-statistic: 12.24 on 10 and 70 DF,  p-value: 6.101e-12
```

Analysis: We see that using the holdout sample, certain X predictors gained/lost significance, indicating potential instability with our modeling.

Comparing with Model in (c)

```
model.light.end <- lm(Y~X1+X2+X3+X4+X8+X9+X10, data=test.df)
summary(model.light.end) # Model in (c)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3 + X4 + X8 + X9 + X10, data = test.df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.86074 -0.66180 -0.02693  0.49506  2.21545
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.894681   2.920243   1.334   0.1948
## X1           0.069574   0.049457   1.407   0.1723
## X2           0.429827   0.197128   2.180   0.0393 *
## X3           0.037291   0.024062   1.550   0.1343
## X4           0.009690   0.014584   0.664   0.5127
## X8           0.007155   0.004329   1.653   0.1114
## X9          -0.002571   0.004039  -0.637   0.5304
## X10          -0.050809   0.023010  -2.208   0.0370 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.093 on 24 degrees of freedom
## Multiple R-squared:  0.5251, Adjusted R-squared:  0.3866
## F-statistic: 3.791 on 7 and 24 DF,  p-value: 0.006588
```

```
summary(model.end) # Model in (e) with holdout sample
```

```
##
## Call:
## lm(formula = Y ~ ., data = test.df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.1463 -0.4101  0.0541  0.2869  2.0243
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.484464   2.645779   2.451 0.023096 *
## X1           0.065061   0.037822   1.720 0.100108
## X2           0.378929   0.150200   2.523 0.019777 *
## X3           0.017632   0.021746   0.811 0.426563
## X4           0.003781   0.011132   0.340 0.737524
```

```

## X5          0.001210    0.003670    0.330 0.744929
## X6          -0.121216    0.714414   -0.170 0.866892
## X7          -0.676796    0.146053   -4.634 0.000143 ***
## X8           0.004417    0.006257    0.706 0.487951
## X9          -0.002534    0.003387   -0.748 0.462671
## X10         -0.035624    0.017592   -2.025 0.055772 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8064 on 21 degrees of freedom
## Multiple R-squared:  0.7738, Adjusted R-squared:  0.6661
## F-statistic: 7.185 on 10 and 21 DF,  p-value: 7.767e-05

```

When both used the holdout sample, the Full model (all X's included) performed much better than the model in (c).

Full model adj-R2: 0.67, (c) model adj-R2: 0.39.