Exam 2 - Stat 330

Isaiah Morgan

April 10, 2018

On an annual basis, each county Assessor is required by Utah law to list and value on an assessment roll all property subject to *ad valorem* taxation. Iron County is located in southwest Utah approximately 265 miles south of Salt Lake City, UT and 170 miles north of Las Vegas, NV on the I-15 corridor. The Iron County Assessor's office assesses values on approximately 35,000 parcels of property on approximately 620,000 acres.

The data file 'ironco.txt' contains data on selling price for various properties, as well as information on covariates that may be related to selling price. The columns are described below:

1. price - selling price of the property
2. lot - lot acreage
3. floors - number of floors (not including basement)
4. const - assessed construction quality on a scale of 1 (poor) to 4 (excellent)
5. roof - assessed roof condition on a scale of 1 (poor) to 4 (excellent)
6. build - assessed home condition on a scale of 1 (poor) to 4 (excellent)
7. area - square footage of home
8. yr.built - year the home was built
9. eff.age - evaluation by the assessor of the home's equivalent market age
10. baths - number of full bathrooms
11. gar - indicator for presence of a garage
12. basmt - indicator of presence of a basement

A model is desired for predicting the selling price of a residential property based on property characteristics.

The purpose of this exam is to demonstrate that you can step through the model building process. The exam will be due at 9:30 am on Tuesday, April 10, and should be turned in via email. Please do your work in an .Rmd file so that code will accompany answers. Hand in ***both*** your .Rmd file and either a .pdf or an .html file. Please name the files 'your_last_name_330Exam2Winter2018' with the appropriate extension.

1. Read in the data, and fit a model that estimates price using area, baths, and lot. What are the $\hat{\beta}$'s?

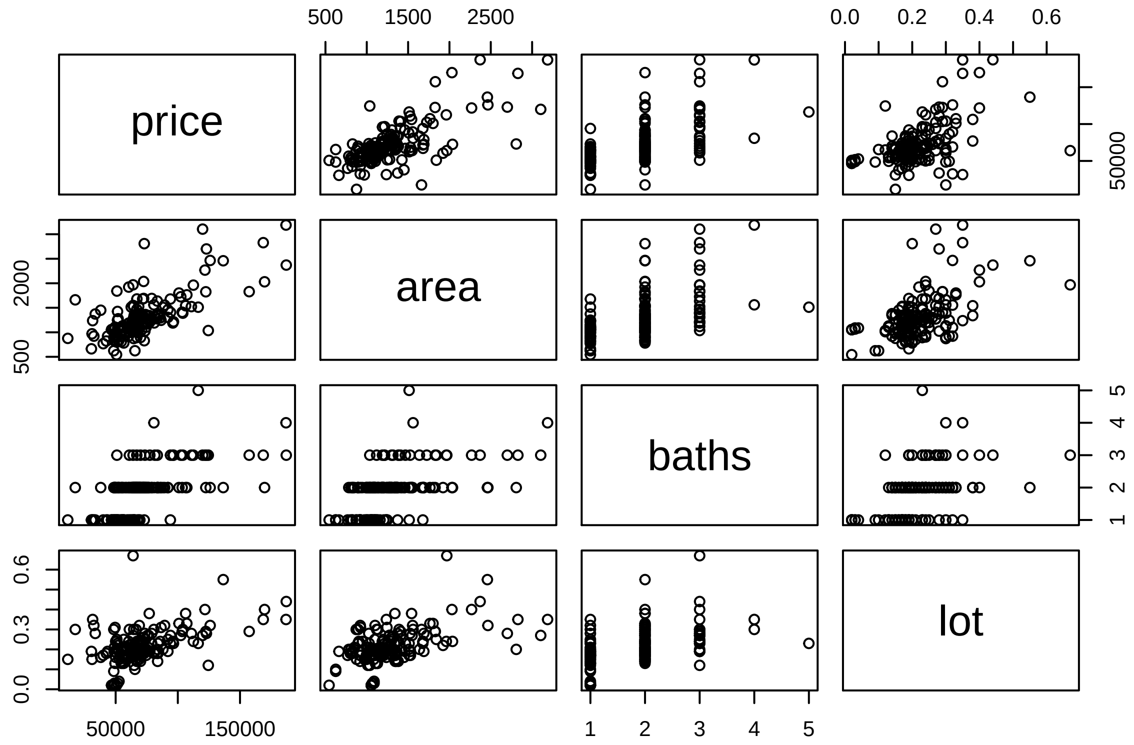
```r
setwd('C:/Users/imoe9/Documents/School Work/STAT PROG/R Files/STAT 330/Exam2')
ironco <- read.table('ironco.txt', header = TRUE)
attach(ironco)

fit <- lm(price ~ area + baths + lot)
coef(fit)
```

```
## (Intercept)        area       baths         lot
##  2749.55026    31.29318 11132.84675 32948.88758
```

2. Produce a pairs plot with price, area, baths, and lot.

```r
pairs(cbind(price, area, baths, lot))
```

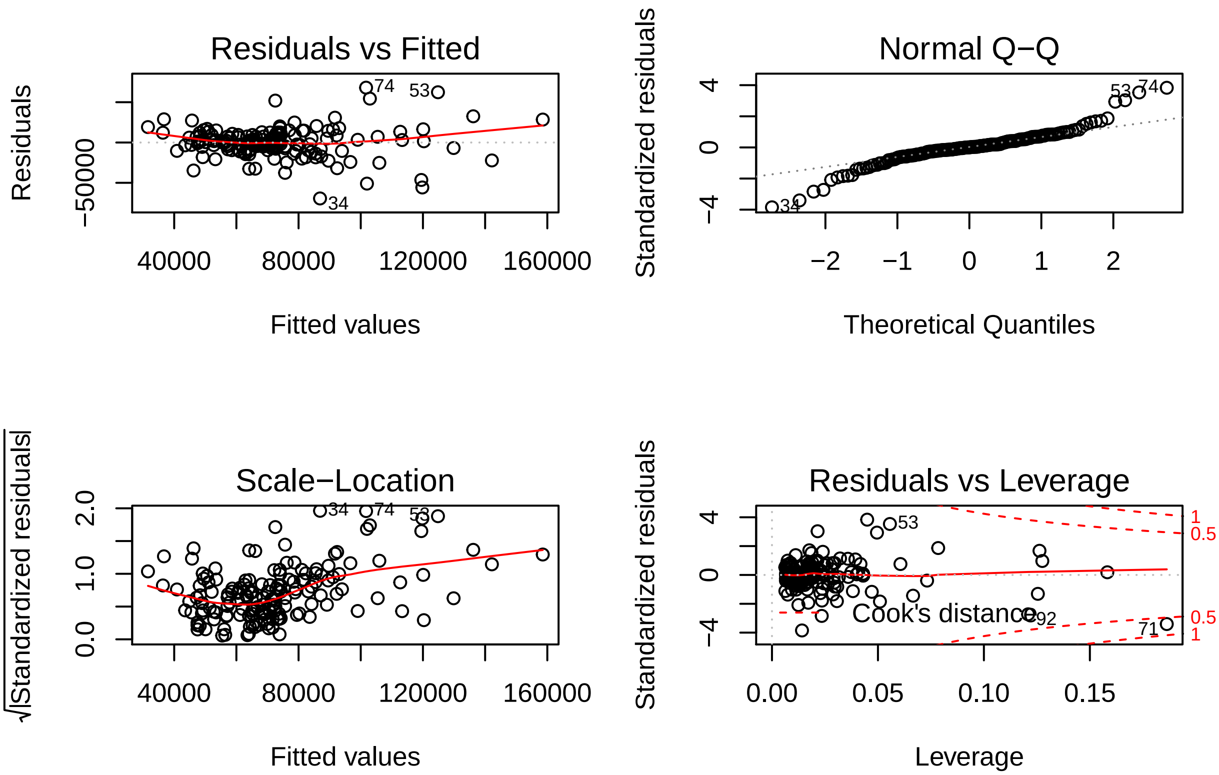


3. Is there anything in the relationship of price with area, or price with lot that might be a concern?

The relationship between price and area appears to be linear; however, it appears that the variance may not be constant. Price with lot does not appear linear which may need to be transformed to fit better

4. Show the four plots that R provides as a default option to examine assumptions about the data.

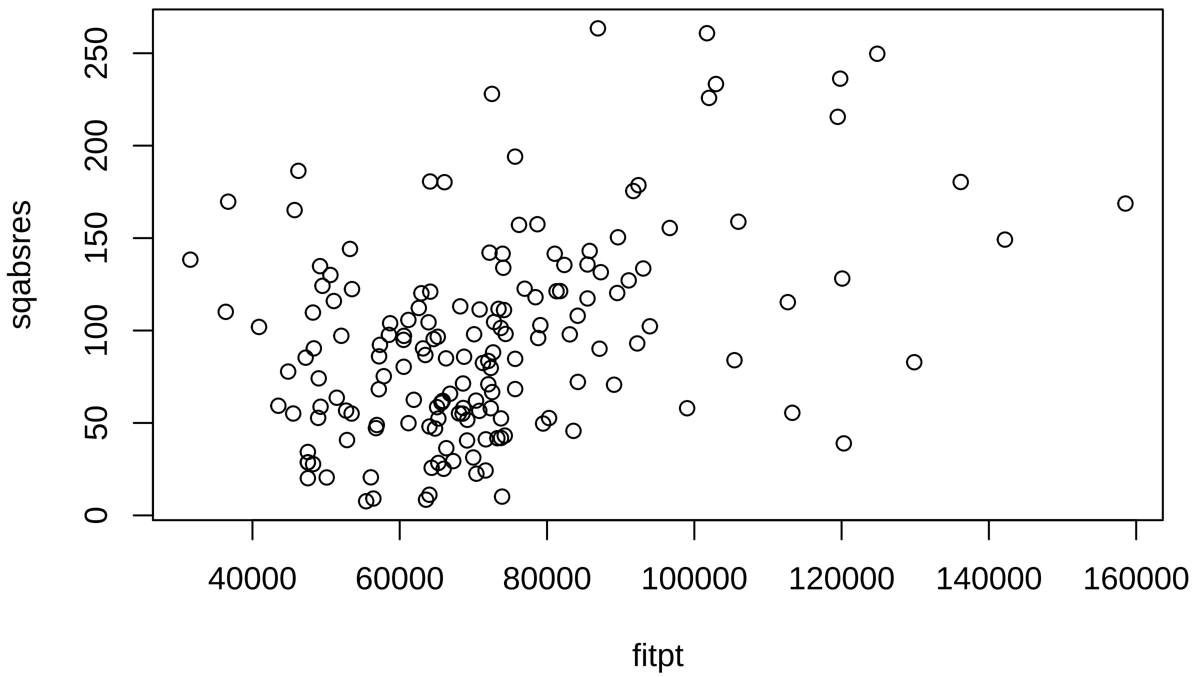
```r
par(mfrow = c(2,2))
plot(fit)
```



5. Find the fitted values and the square root of the absolute value of the standardized residuals for this model. Plot the fitted values on the x-axis and the square root of the absolute value of the standarized residuals on the y-axis.

```r
par(mfrow = c(1,1))
fitpt <- fitted(fit)
fitres <- resid(fit)
sqabsres <- sqrt(abs(fitres))

plot(fitpt, sqabsres)
```



6. Produce a linear model to estimate the square root of the absolute value of the standardized residuals as a function of the fitted values. What are the $\hat{\beta}$'s?

```
fit.asr <- lm(sqabsres ~ fitpt)
coef(fit.asr)
```

```
##  (Intercept)        fitpt
## 18.545892004  0.001059667
```

7. Is the slope of the estimated line in number 6 significantly different from 0? What is the t-value of the test of this null hypothesis?

```
summary(fit.asr)
```

```
##
## Call:
## lm(formula = sqabsres ~ fitpt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -107.07  -35.51   -5.46   24.21  152.82
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.855e+01  1.386e+01   1.338    0.183
## fitpt       1.060e-03  1.846e-04   5.739 4.57e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 49.63 on 162 degrees of freedom
## Multiple R-squared:  0.169,  Adjusted R-squared:  0.1638
## F-statistic: 32.94 on 1 and 162 DF,  p-value: 4.57e-08
```

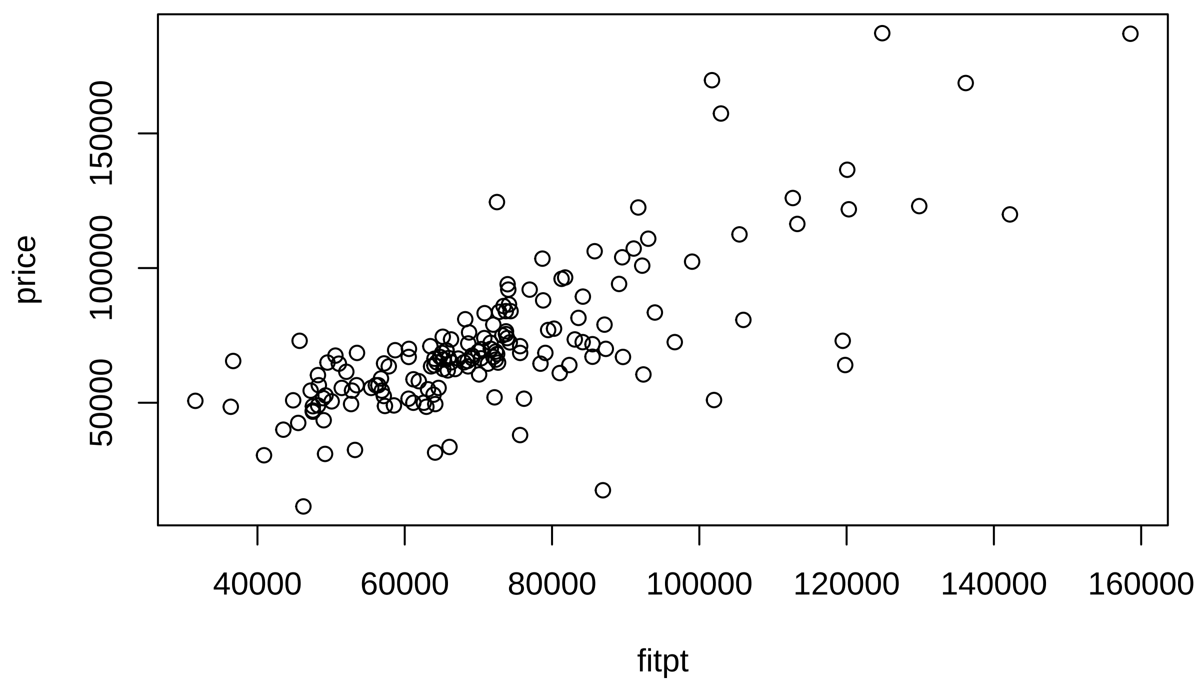
The slope of the line is significantly different from 0 ($p < 0.0001$) with a t-stat of 5.739

8. What does the result from 7 indicate might be a problem?

This significant slope indicates expanding variance which invalidates the model.

9. Plot the fitted values (on the x-axis) against the actual values of the price.

```r
plot(fitpt, price)
```

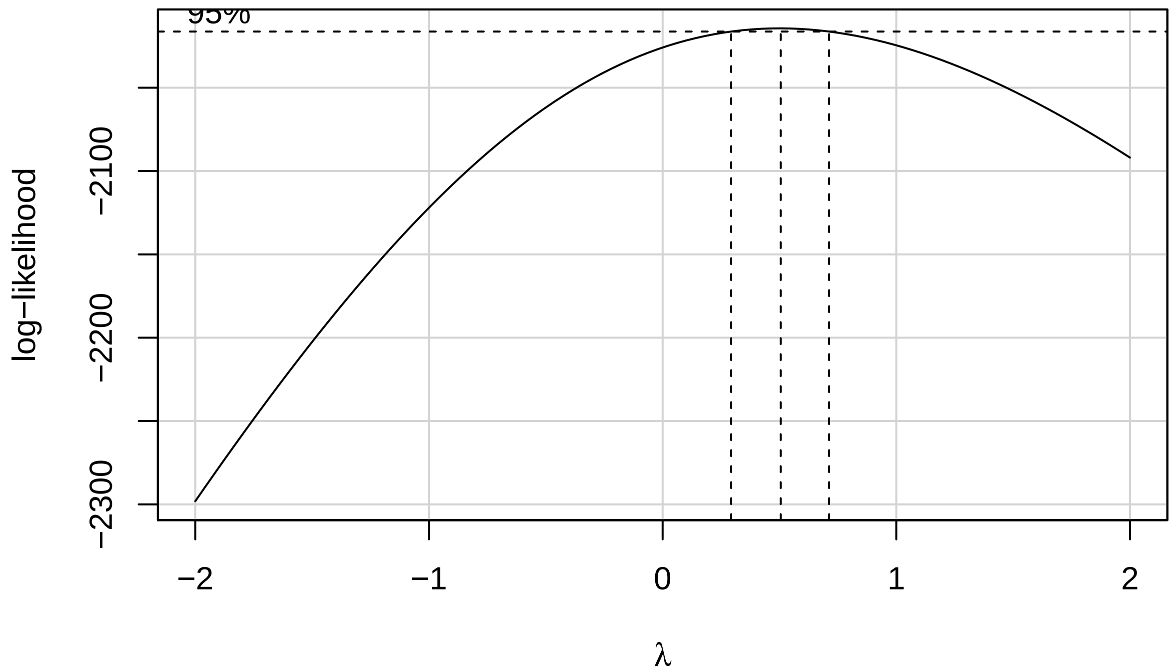


10. Given the results we have seen thus far, we might consider tranforming some of the variables. Does the Box-Cox procedure indicate a transformation on price might be a good idea? If so, what transformation would you suggest?

```r
library(alr3)
```

```
## Loading required package: car
```

```r
boxCox(fit)
```



The BoxCox suggests that we should take the sqrt of the data.

11. Perhaps we may want to consider transforming some of the x-variables. Using only those variables in the command, what transformations, if any, would you suggest for lot, area, eff.age, and baths?

```
powerTransform(cbind(lot,area,eff.age,baths))
```

```
## Estimated transformation parameters
##        lot       area    eff.age      baths
##  0.6950876 -0.4233065  0.4169122  0.3397706
```

The power transform of those variables suggests that a sqrt of both lot and eff.age would be appropriate; as well as a negative sqrt for area and a third root of baths.

12. Now fit a model to predict the square root of price (sqprice) with the following x's: lot, area, square root of baths (sqbaths), gar, floors, basmt, const, roof, build, square root of effective age (sqage). What term has the largest p-value and what is the p-value?

```
fitfull <- lm(sqrt(price) ~ lot + area + sqrt(baths) + gar + floors + basmt + const
              + roof + build + sqrt(eff.age))
summary(fitfull)
```

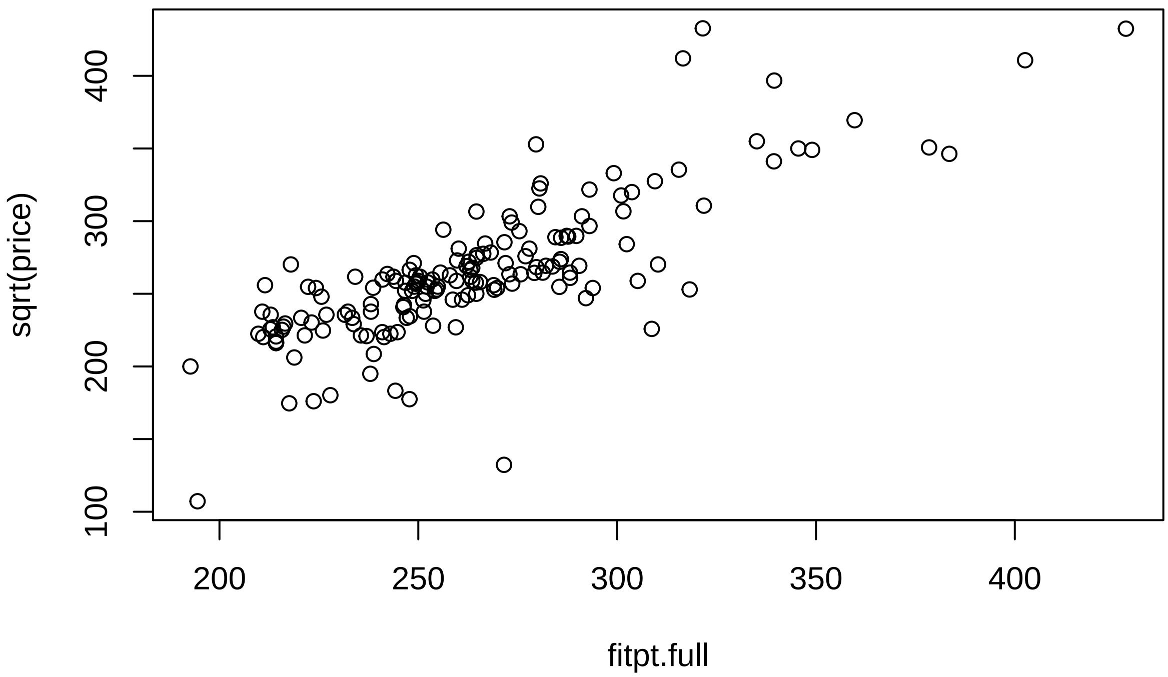
```
##
## Call:
## lm(formula = sqrt(price) ~ lot + area + sqrt(baths) + gar + floors +
##     basmt + const + roof + build + sqrt(eff.age))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -139.262  -13.504    3.263   12.931  111.132
```

```
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   46.459375  41.340889   1.124 0.262853
## lot           43.341477  40.973867   1.058 0.291821
## area           0.041587   0.008986   4.628 7.82e-06 ***
## sqrt(baths)   38.909920  12.211919   3.186 0.001747 **
## gar           10.620408   6.479562   1.639 0.103255
## floors         0.759572  10.841700   0.070 0.944237
## basmt         23.445718   6.766475   3.465 0.000688 ***
## const          8.162178  12.248324   0.666 0.506165
## roof          26.685005   9.362799   2.850 0.004973 **
## build          2.877433   7.080874   0.406 0.685041
## sqrt(eff.age) -3.845694   2.078479  -1.850 0.066207 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29.85 on 153 degrees of freedom
## Multiple R-squared:  0.6417, Adjusted R-squared:  0.6183
## F-statistic:  27.4 on 10 and 153 DF,  p-value: < 2.2e-16
```

Floors has the largest p value at $p = 0.944$

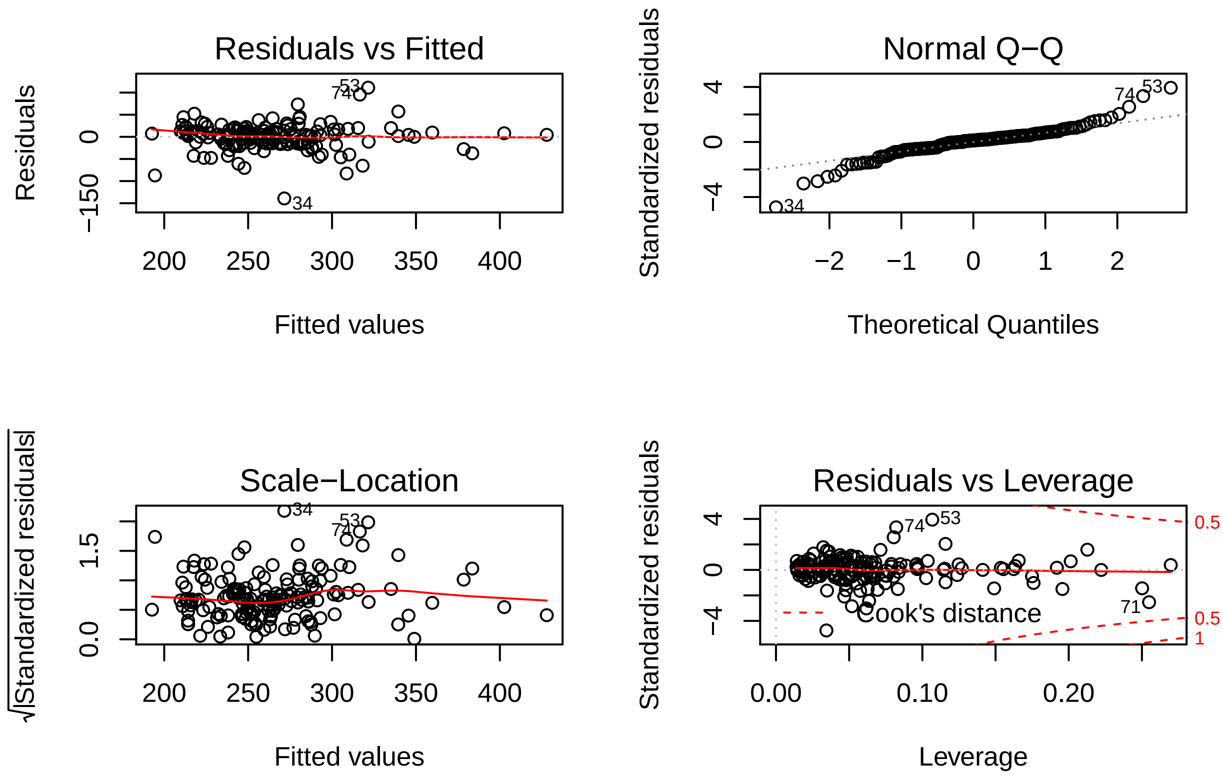
13. Now plot the fitted values from this model (on the x-axis) and the actual square root of price on the y-axis.

```
fitpt.full <- fitted(fitfull)
plot(fitpt.full, sqrt(price))
```



14. Show the four plots that R provides as a default option to examine assumptions about the data.

```
par(mfrow=c(2,2))
plot(fitfull)
```



15. The normal qqplot of the standardized residuals shows something that is a concern. What is it?

The QQ plot indicates especially large tails in this data which could invalidate our predictions based off a normal distribution.

16. Are there any observations for which the values of Cook's distance that might be a concern?

There are no points which exceed cooks distance given the leverage plot above. However, point 71 is close to exceeding.

17. Are there any x variables with variance inflation factors that are a concern?

```
vif(fitfull)
```

```
##          lot         area  sqrt(baths)          gar        floors
##     2.341278     2.945781     1.778291     1.780481      1.919862
##        basmt        const         roof        build sqrt(eff.age)
##     1.444250     1.465488     1.421012     1.785798      2.129004
```

None of these variance inflation factors exceed the standard limit of 5, so I would say that none of them are of great concern.

18. Now I want you to split the data into a test set and a training set. Use the following commands to split the data randomly into two groups. For my example code, I will assume your data set is called 'newdata'.

```
set.seed(0)
aa <- 1:164
trainset <- sample(aa,100)
traindata <- ironco[trainset,]
```

```r
testdata <- ironco[-trainset,]
detach(ironco)
```

These lines of code will give you two data sets. One with 100 observations that we will use to develop possible models that we would use to predict, and one with 64 observations that we will use to test the models. Using the training data set, run the full model. Show a summary of the model.

```r
sqrtprice <- sqrt(traindata$price)
sqrtbaths <- sqrt(traindata$baths)
sqrtage <- sqrt(traindata$eff.age)
fulltrain <- lm(sqrtprice ~ lot + area + sqrtbaths + gar + floors + basmt + const + roof +
                build + sqrtage, data = traindata)
summary(fulltrain)
```

```
##
## Call:
## lm(formula = sqrtprice ~ lot + area + sqrtbaths + gar + floors +
##     basmt + const + roof + build + sqrtage, data = traindata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -153.234  -10.232    0.206   12.584   74.612
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  83.03120   50.79799   1.635  0.10568
## lot         198.95941   71.01398   2.802  0.00624 **
## area          0.04607    0.01239   3.719  0.00035 ***
## sqrtbaths    14.44307   15.43760   0.936  0.35202
## gar           2.73256    8.51001   0.321  0.74889
## floors       18.31770   13.74648   1.333  0.18609
## basmt        33.28997    9.87608   3.371  0.00111 **
## const        -7.88959   16.17904  -0.488  0.62700
## roof         36.83987   11.97476   3.076  0.00278 **
## build        -6.56054    9.41757  -0.697  0.48785
## sqrtage      -6.10603    2.82521  -2.161  0.03336 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29.88 on 89 degrees of freedom
## Multiple R-squared:  0.6693, Adjusted R-squared:  0.6322
## F-statistic: 18.01 on 10 and 89 DF,  p-value: < 2.2e-16
```

19. Using this model as the base model, do a stepwise procedure going backward and using AIC as the criterion. What terms are in the best model?

```r
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:alr3':
##
##     forbes
```

```r
stepAIC(fulltrain, direction = 'backward')
```

```
## Start:  AIC=689.77
## sqrtprice ~ lot + area + sqrtbaths + gar + floors + basmt + const +
##     roof + build + sqrtage
##
##             Df Sum of Sq   RSS    AIC
## - gar        1      92.0 79544 687.89
## - const      1     212.3 79664 688.04
## - build      1     433.2 79885 688.32
## - sqrtbaths  1     781.4 80233 688.75
## - floors     1    1585.2 81037 689.75
## <none>                   79452 689.77
## - sqrtage    1    4169.9 83622 692.89
## - lot        1    7007.4 86459 696.23
## - roof       1    8449.2 87901 697.88
## - basmt      1   10143.1 89595 699.79
## - area       1   12343.8 91795 702.21
##
## Step:  AIC=687.89
## sqrtprice ~ lot + area + sqrtbaths + floors + basmt + const +
##     roof + build + sqrtage
##
##             Df Sum of Sq   RSS    AIC
## - const      1     203.9 79748 686.15
## - build      1     388.7 79932 686.38
## - sqrtbaths  1     846.5 80390 686.95
## - floors     1    1503.3 81047 687.76
## <none>                   79544 687.89
## - sqrtage    1    5695.1 85239 692.80
## - lot        1    6950.1 86494 694.27
## - roof       1    9284.3 88828 696.93
## - basmt      1   11004.1 90548 698.85
## - area       1   13140.9 92685 701.18
##
## Step:  AIC=686.15
## sqrtprice ~ lot + area + sqrtbaths + floors + basmt + roof +
##     build + sqrtage
##
##             Df Sum of Sq   RSS    AIC
## - build      1     573.9 80322 684.86
## - sqrtbaths  1     938.8 80686 685.32
## - floors     1    1299.4 81047 685.76
## <none>                   79748 686.15
## - sqrtage    1    6123.4 85871 691.54
## - lot        1    6917.5 86665 692.46
## - roof       1    9116.6 88864 694.97
## - basmt      1   10887.4 90635 696.94
## - area       1   13120.9 92868 699.38
##
## Step:  AIC=684.86
## sqrtprice ~ lot + area + sqrtbaths + floors + basmt + roof +
##     sqrtage
##
##             Df Sum of Sq   RSS    AIC
## - sqrtbaths  1    1091.4 81413 684.21
```

```
## - floors     1    1155.6 81477 684.29
## <none>                   80322 684.86
## - sqrtage    1    6162.0 86484 690.25
## - lot        1    6359.7 86681 690.48
## - roof       1    8632.0 88954 693.07
## - basmt      1   10493.1 90815 695.14
## - area       1   13519.2 93841 698.42
##
## Step:  AIC=684.21
## sqrtprice ~ lot + area + floors + basmt + roof + sqrtage
##
##           Df Sum of Sq   RSS    AIC
## - floors   1     827.6 82241 683.22
## <none>                 81413 684.21
## - sqrtage  1    6399.5 87812 689.78
## - lot      1    7262.8 88676 690.76
## - roof     1   10856.0 92269 694.73
## - basmt    1   12686.0 94099 696.69
## - area     1   17885.7 99299 702.07
##
## Step:  AIC=683.22
## sqrtprice ~ lot + area + basmt + roof + sqrtage
##
##           Df Sum of Sq    RSS    AIC
## <none>                  82241 683.22
## - lot      1    6470.0  88710 688.80
## - sqrtage  1    7528.5  89769 689.98
## - roof     1   11171.9  93412 693.96
## - basmt    1   12290.0  94531 695.15
## - area     1   26322.0 108563 708.99
```

```
##
## Call:
## lm(formula = sqrtprice ~ lot + area + basmt + roof + sqrtage,
##     data = traindata)
##
## Coefficients:
## (Intercept)          lot         area        basmt         roof
##    68.61821    163.29981      0.05644     33.82146     38.67859
##     sqrtage
##    -5.97251
```

From this procedure the best model includes lot, area, basmt, roof, and sqrtage.

20. Print a summary of this model.

```r
fit1 <- lm(sqrtprice ~ lot + area + basmt + roof + sqrtage, data = traindata)
summary(fit1)
```

```
##
## Call:
## lm(formula = sqrtprice ~ lot + area + basmt + roof + sqrtage,
##     data = traindata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -154.975  -11.922    0.341   13.686   79.904
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  68.61821   25.80206   2.659 0.009203 **
## lot         163.29981   60.05014   2.719 0.007790 **
## area          0.05644    0.01029   5.485 3.47e-07 ***
## basmt        33.82146    9.02393   3.748 0.000308 ***
## roof         38.67859   10.82395   3.573 0.000558 ***
## sqrtage      -5.97251    2.03601  -2.933 0.004211 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29.58 on 94 degrees of freedom
## Multiple R-squared:  0.6577, Adjusted R-squared:  0.6395
## F-statistic: 36.13 on 5 and 94 DF,  p-value: < 2.2e-16
```

21. Run the same stepwise procedure going backward and use BIC as the criterion. What terms are in the best model now?

```
stepAIC(fulltrain, k = log(100), direction = 'backward')
```

```
## Start:  AIC=718.43
## sqrtprice ~ lot + area + sqrtbaths + gar + floors + basmt + const +
##     roof + build + sqrtage
##
##             Df Sum of Sq   RSS    AIC
## - gar        1      92.0 79544 713.94
## - const      1     212.3 79664 714.09
## - build      1     433.2 79885 714.37
## - sqrtbaths  1     781.4 80233 714.80
## - floors     1    1585.2 81037 715.80
## <none>                   79452 718.43
## - sqrtage    1    4169.9 83622 718.94
## - lot        1    7007.4 86459 722.28
## - roof       1    8449.2 87901 723.93
## - basmt      1   10143.1 89595 725.84
## - area       1   12343.8 91795 728.27
##
## Step:  AIC=713.94
## sqrtprice ~ lot + area + sqrtbaths + floors + basmt + const +
##     roof + build + sqrtage
##
##             Df Sum of Sq   RSS    AIC
## - const      1     203.9 79748 709.59
## - build      1     388.7 79932 709.82
## - sqrtbaths  1     846.5 80390 710.39
## - floors     1    1503.3 81047 711.21
## <none>                   79544 713.94
## - sqrtage    1    5695.1 85239 716.25
## - lot        1    6950.1 86494 717.71
## - roof       1    9284.3 88828 720.38
## - basmt      1   11004.1 90548 722.29
## - area       1   13140.9 92685 724.63
##
```

```
## Step:  AIC=709.59
## sqrtprice ~ lot + area + sqrtbaths + floors + basmt + roof +
##     build + sqrtage
##
##             Df Sum of Sq   RSS    AIC
## - build      1     573.9 80322 705.70
## - sqrtbaths  1     938.8 80686 706.16
## - floors     1    1299.4 81047 706.60
## <none>                   79748 709.59
## - sqrtage    1    6123.4 85871 712.38
## - lot        1    6917.5 86665 713.31
## - roof       1    9116.6 88864 715.81
## - basmt      1   10887.4 90635 717.78
## - area       1   13120.9 92868 720.22
##
## Step:  AIC=705.7
## sqrtprice ~ lot + area + sqrtbaths + floors + basmt + roof +
##     sqrtage
##
##             Df Sum of Sq   RSS    AIC
## - sqrtbaths  1    1091.4 81413 702.45
## - floors     1    1155.6 81477 702.53
## <none>                   80322 705.70
## - sqrtage    1    6162.0 86484 708.49
## - lot        1    6359.7 86681 708.72
## - roof       1    8632.0 88954 711.31
## - basmt      1   10493.1 90815 713.38
## - area       1   13519.2 93841 716.65
##
## Step:  AIC=702.45
## sqrtprice ~ lot + area + floors + basmt + roof + sqrtage
##
##           Df Sum of Sq   RSS    AIC
## - floors   1     827.6 82241 698.85
## <none>                 81413 702.45
## - sqrtage  1    6399.5 87812 705.41
## - lot      1    7262.8 88676 706.39
## - roof     1   10856.0 92269 710.36
## - basmt    1   12686.0 94099 712.32
## - area     1   17885.7 99299 717.70
##
## Step:  AIC=698.85
## sqrtprice ~ lot + area + basmt + roof + sqrtage
##
##           Df Sum of Sq    RSS    AIC
## <none>                  82241 698.85
## - lot      1    6470.0  88710 701.82
## - sqrtage  1    7528.5  89769 703.01
## - roof     1   11171.9  93412 706.99
## - basmt    1   12290.0  94531 708.18
## - area     1   26322.0 108563 722.02

##
## Call:
```

```
## lm(formula = sqrtprice ~ lot + area + basmt + roof + sqrtage,
##     data = traindata)
##
## Coefficients:
## (Intercept)          lot         area        basmt         roof
##    68.61821    163.29981      0.05644     33.82146     38.67859
##     sqrtage
##    -5.97251
```

This procedure says the best model is the one including lot, area, basmt, roof, and sqrtage which is the same as the previous procedure.

22. Now run the stepwise procedure building up from only the intercept and using BIC as the criterion. What are the terms in the best model now?

```r
mintrain <- lm(sqrtprice ~ 1)
attach(traindata)
stepAIC(mintrain, k = log(100), direction = 'forward', scope=list(lower = ~1,
        upper = ~lot + area + sqrtbaths + gar + floors + basmt + const + roof +
        build + sqrtage))
```

```
## Start:  AIC=783.04
## sqrtprice ~ 1
##
##             Df Sum of Sq    RSS    AIC
## + area       1    104049 136221 730.90
## + lot        1     83307 156963 745.07
## + sqrtbaths  1     71184 169086 752.51
## + gar        1     42511 197759 768.17
## + roof       1     35111 205159 771.85
## + build      1     25685 214585 776.34
## + basmt      1     18334 221936 779.71
## <none>                   240270 783.04
## + const      1      8444 231827 784.07
## + sqrtage    1      8335 231935 784.11
## + floors     1      2336 237934 786.67
##
## Step:  AIC=730.9
## sqrtprice ~ area
##
##             Df Sum of Sq    RSS    AIC
## + basmt      1   27575.8 108645 712.88
## + lot        1   20676.4 115544 719.04
## + sqrtbaths  1   16129.5 120091 722.90
## + gar        1   12100.3 124120 726.20
## + roof       1    8255.0 127966 729.25
## + build      1    7245.7 128975 730.04
## <none>                   136221 730.90
## + sqrtage    1    2749.2 133472 733.46
## + floors     1    1925.1 134296 734.08
## + const      1    1225.1 134996 734.60
##
## Step:  AIC=712.88
## sqrtprice ~ area + basmt
##
##             Df Sum of Sq    RSS    AIC
```

```
## + roof       1   15808.7  92836 701.76
## + sqrtage    1    7782.6 100862 710.05
## + gar        1    6809.6 101835 711.01
## <none>                   108645 712.88
## + sqrtbaths  1    4721.2 103924 713.04
## + build      1    4022.6 104622 713.71
## + lot        1    2878.6 105766 714.80
## + floors     1     591.0 108054 716.94
## + const      1     389.8 108255 717.13
##
## Step:  AIC=701.76
## sqrtprice ~ area + basmt + roof
##
##             Df Sum of Sq   RSS    AIC
## <none>                   92836 701.76
## + sqrtage    1    4125.7 88710 701.82
## + lot        1    3067.2 89769 703.01
## + gar        1    1872.5 90964 704.33
## + sqrtbaths  1    1645.8 91190 704.58
## + build      1    1247.0 91589 705.02
## + floors     1     191.5 92645 706.16
## + const      1     108.7 92728 706.25
```

```
##
## Call:
## lm(formula = sqrtprice ~ area + basmt + roof)
##
## Coefficients:
## (Intercept)         area        basmt         roof
##    37.38971      0.07518     45.61003     45.00238
```

This procedure suggests that the best model includes only area, basmt, and roof.

23. Using an all possible subsets regression with Adj R^2 as the criterion, what terms are in the best model?

```r
library(leaps)
```

```
## Warning: package 'leaps' was built under R version 3.4.4
```

```r
y <- sqrtprice
x <- as.matrix(cbind(lot, area, sqrtbaths, gar, floors, basmt, const, roof, build, sqrtage))
names <- c('lot', 'area', 'sqrtbaths', 'gar', 'floors', 'basmt', 'const',
           'roof', 'build', 'sqrtage')


leaps(x,y,nbest = 1, names = names, method = 'adjr2')
```

```
## $which
##      lot area sqrtbaths   gar floors basmt const  roof build sqrtage
## 1  FALSE TRUE     FALSE FALSE  FALSE FALSE FALSE FALSE FALSE   FALSE
## 2  FALSE TRUE     FALSE FALSE  FALSE  TRUE FALSE FALSE FALSE   FALSE
## 3  FALSE TRUE     FALSE FALSE  FALSE  TRUE FALSE  TRUE FALSE   FALSE
## 4  FALSE TRUE     FALSE FALSE  FALSE  TRUE FALSE  TRUE FALSE    TRUE
## 5   TRUE TRUE     FALSE FALSE  FALSE  TRUE FALSE  TRUE FALSE    TRUE
## 6   TRUE TRUE     FALSE FALSE   TRUE  TRUE FALSE  TRUE FALSE    TRUE
## 7   TRUE TRUE      TRUE FALSE   TRUE  TRUE FALSE  TRUE FALSE    TRUE
## 8   TRUE TRUE      TRUE FALSE   TRUE  TRUE FALSE  TRUE  TRUE    TRUE
```

```
## 9   TRUE TRUE      TRUE FALSE   TRUE  TRUE  TRUE  TRUE  TRUE    TRUE
## 10  TRUE TRUE      TRUE  TRUE   TRUE  TRUE  TRUE  TRUE  TRUE    TRUE
##
## $label
##  [1] "(Intercept)" "lot"         "area"        "sqrtbaths"   "gar"
##  [6] "floors"      "basmt"       "const"       "roof"        "build"
## [11] "sqrtage"
##
## $size
##  [1]  2  3  4  5  6  7  8  9 10 11
##
## $adjr2
##  [1] 0.4272663 0.5384984 0.6015427 0.6152427 0.6395097 0.6393003 0.6402675
##  [8] 0.6389130 0.6358346 0.6321690
```

This procedure suggests that the model should include lot, area, basmt, roof, anad sqrtage.

24. Now we are going to compare three models, to see which one predicts sqprice best in the test set. Model 1 has lot, area, sqbaths, floors, basmt, roof, and sqage. Fit this model, and use it to predict sqprice in the test data set. What is the error sum of squares for the true values minus the predicted values?

```r
detach(traindata)
attach(testdata)
tsqrtprice <- sqrt(testdata$price)
tsqrtbaths <- sqrt(testdata$baths)
tsqrtage <- sqrt(testdata$eff.age)

model1 <- lm(sqrtprice ~ lot + area + sqrtbaths + floors + basmt + roof + sqrtage, data = traindata)

X1test <- cbind(1, lot, area, tsqrtbaths, floors, basmt, roof, tsqrtage)
yhattest1 <- X1test %*% coef(model1)
m1SSE <- sum((tsqrtprice - yhattest1)^2)
m1SSE
```

```
## [1] 82393.27
```

25. Model 2 has lot, area, basmt, roof, sqage. Fit this model and use it to predict sqprice in the test data set. What is the error sum of squares (SSE) for the true values minus the predicted values?

```r
model2 <- lm(sqrtprice ~ lot + area + basmt + roof + sqrtage, data = traindata)

X2test <- cbind(1, lot, area, basmt, roof, tsqrtage)
yhattest2 <- X2test %*% coef(model2)
m2SSE <- sum((tsqrtprice - yhattest2)^2)
m2SSE
```

```
## [1] 84536.34
```

26. Model 3 has area, basmt, and roof. Fit this model and use it to predict sqprice in the test data set. What is the error sum of squares for the true values minus the predicted values?

```r
model3 <- lm(sqrtprice ~area + basmt + roof, data = traindata)

X3test <- cbind(1, area, basmt, roof)
yhattest3 <- X3test %*% coef(model3)
m3SSE <- sum((tsqrtprice - yhattest3)^2)
m3SSE
```

```
## [1] 82719.5
```

27. Now find the median absolute deviation for model 1.

```
medabsd1 <- median(tsqrtprice - yhattest1)
medabsd1
```

```
## [1] 2.845203
```

28. Now find the median absolute deviation for model 2.

```
medabsd2 <- median(tsqrtprice - yhattest2)
medabsd2
```

```
## [1] 1.971632
```

29. Now find the median absolute deviation (MAD) for model 3.

```
medabsd3 <- median(tsqrtprice - yhattest3)
medabsd3
```

```
## [1] 4.792247
```

30. Which model predicts the best using SSE?

```
which.min(c(m1SSE, m2SSE, m3SSE))
```

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

According to SSE method, the first model is the best predictor.

31. Which model predicts the best using MAD?

```
which.min(c(medabsd1,medabsd2,medabsd3))
```

```
## [1] 2
```

According to MAD, the second model is the best predictor