# STOR 455 Homework #5

## 40 points - Due 10/20 at 5:00pm

**Directions:** For parts 6 and 9 you may work together, but they should be **submitted individually** by each group member. For parts 7 and 8, you should have only **one submission per group**. There will be separate places on Gradescope to submit the individual vs group work.

**Situation:** Can we predict the selling price of a house in Ames, Iowa based on recorded features of the house? That is your task for this assignment. Each team will get a dataset with information on forty potential predictors and the selling price (in $1,000's) for a sample of homes. The data sets for your group are AmesTrain??.csv and AmesTest??.csv (where ?? corresponds to your group number) A separate file identifies the variables in the Ames Housing data and explains some of the coding.

```
library(readr)
library(corrplot)

## corrplot 0.90 loaded

library(leaps)
library(car)

## Loading required package: carData
```

### Part 6. Cross-validation:

In some situations, a model might fit the peculiarities of a specific sample of data well, but not reflect structure that is really present in the population. A good test for how your model might work on "real" house prices can be simulated by seeing how well your fitted model does at predicting prices that were NOT in your original sample. This is why we reserved an additional 200 cases as a holdout sample in AmesTest??.csv. Import your holdout test data and

```
setwd("C:/Users/adeve/Desktop")
amestrain24 <- read.csv("AmesTrain24.csv")
amestest24 <- read.csv("AmesTest24.csv")
```

- Compute the predicted Price for each of the cases in the holdout test sample, using your model resulting from the initial fit and residual analysis in parts 1 and 2 of Homework #3. This should be done with the same AmesTrain??.csv dataset that you used for homework #3, with your assignment #3 group numbe, and AmesTrain?? also using your assignment #3 group number.

```
allsubmod = lm(Price~Fireplaces+GarageSF+GroundSF, amestrain24)

ames.test.predict <- predict(allsubmod, newdata=amestest24)
```

- Compute the residuals for the 200 holdout cases.

```
ames.test.residual = amestest24$Price - ames.test.predict
```

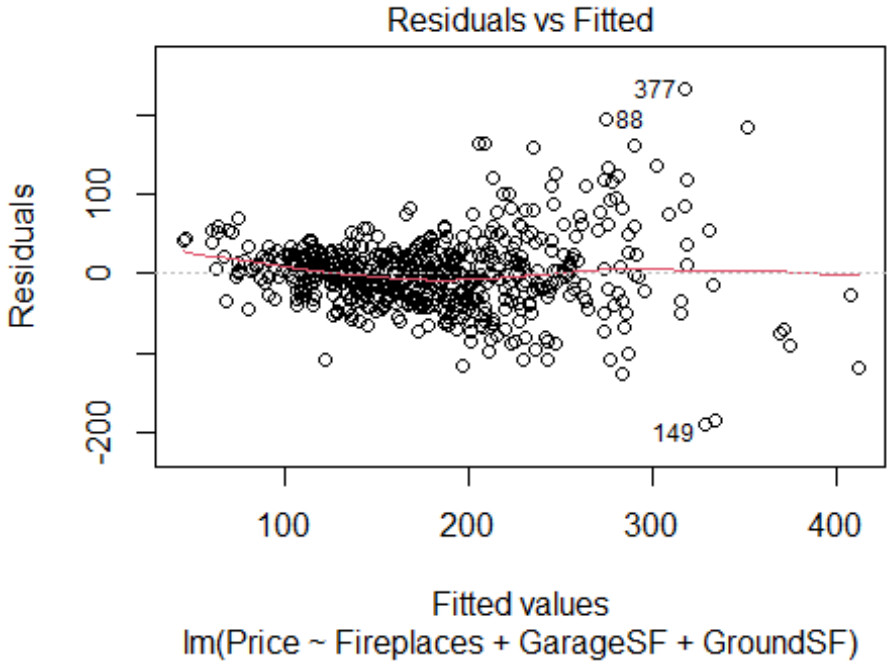
- Compute the mean and standard deviation of these residuals. Are they close to what you expect from the training model?

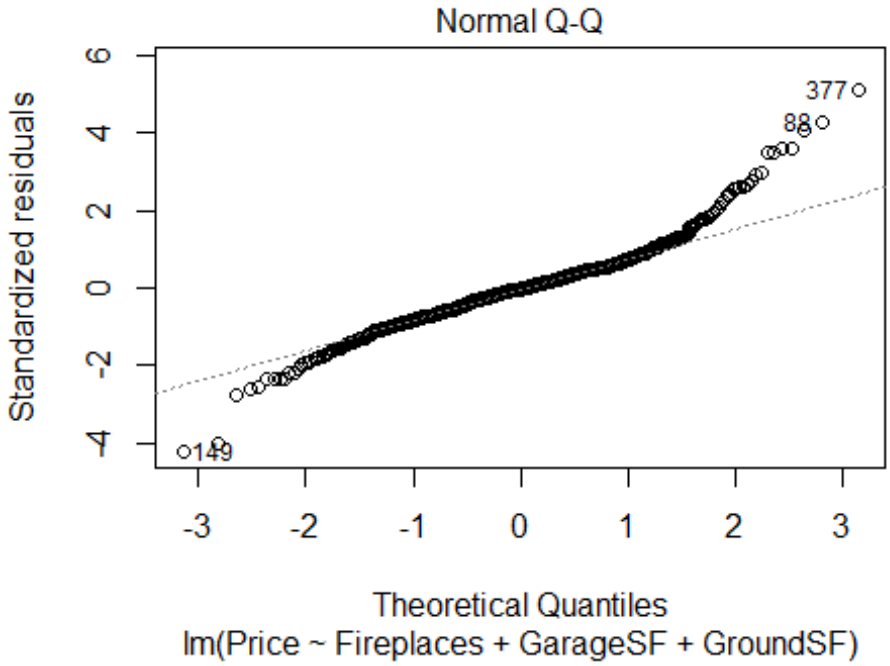
*From the summary of the allsubmod, we would expect a residual standard error of 46.14. Since the ames.test.residual is 37.61 and we are talking about thousands of dollars when referring to houses, the residual is roughly close enough to what we would expect from the training model.*

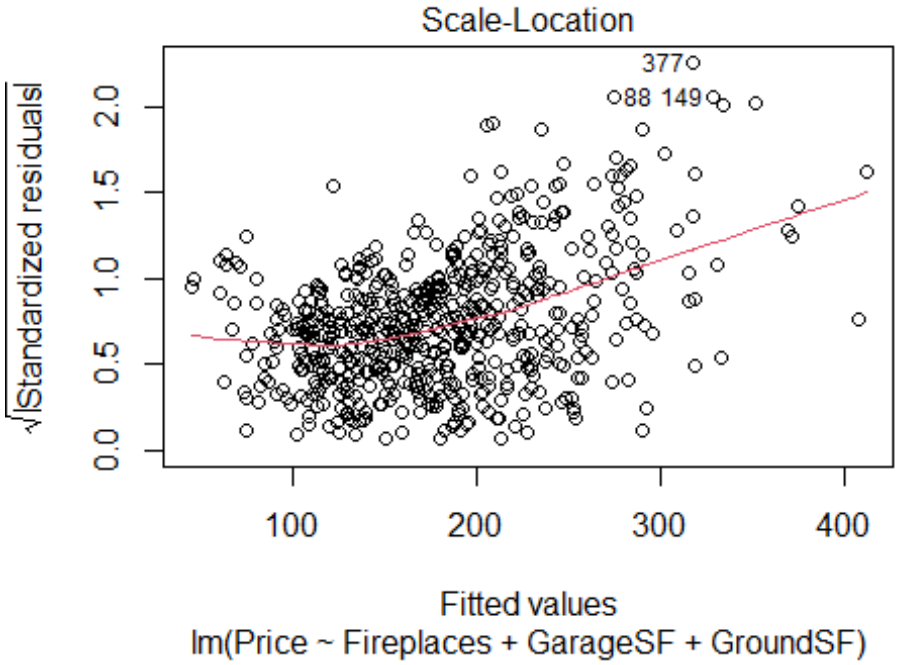
```
mean(ames.test.residual)

## [1] 4.265713

sd(ames.test.residual)

## [1] 37.61052

summary(allsubmod)

##
## Call:
## lm(formula = Price ~ Fireplaces + GarageSF + GroundSF, data = amestrain24)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -189.853  -26.864   -1.401   21.907  233.684
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.891286   6.291353   1.095    0.274
## Fireplaces  19.901124   3.513306   5.665  2.3e-08 ***
## GarageSF     0.142456   0.009959  14.305  < 2e-16 ***
## GroundSF     0.062734   0.004650  13.490  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 46.14 on 596 degrees of freedom
## Multiple R-squared:  0.6301, Adjusted R-squared:  0.6283
## F-statistic: 338.5 on 3 and 596 DF,  p-value: < 2.2e-16
```

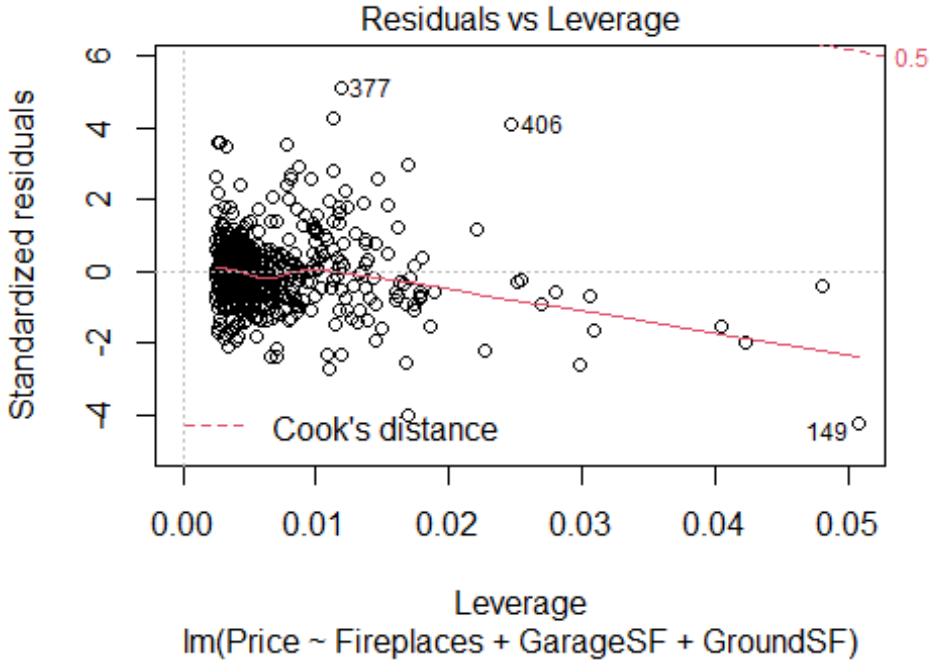
- Construct a plot of the residuals to determine if they are normally distributed. Is this plot what you expect to see considering the training model? *The residuals in the testing data are more spread out than in the training data. Furthermore, the right tail of the QQNorm plot on the testing data is much more prominent than the training data. This suggests that there may be a skew that a model that is fitted to the testing data that may not be accounted for in the other data.*

```
plot(allsubmod)
```

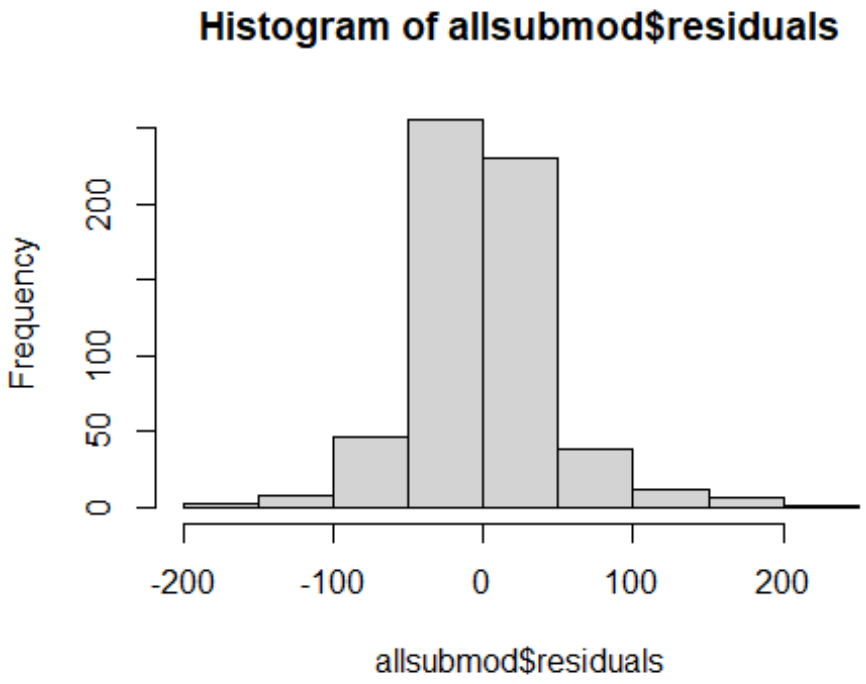

Residuals vs Fitted
Residuals
Fitted values
lm(Price ~ Fireplaces + GarageSF + GroundSF)
377
88
149
100
0
-200
100
200
300
400


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Price ~ Fireplaces + GarageSF + GroundSF)
377
88
149
6
4
2
0
-2
-4
-3
-2
-1
0
1
2
3

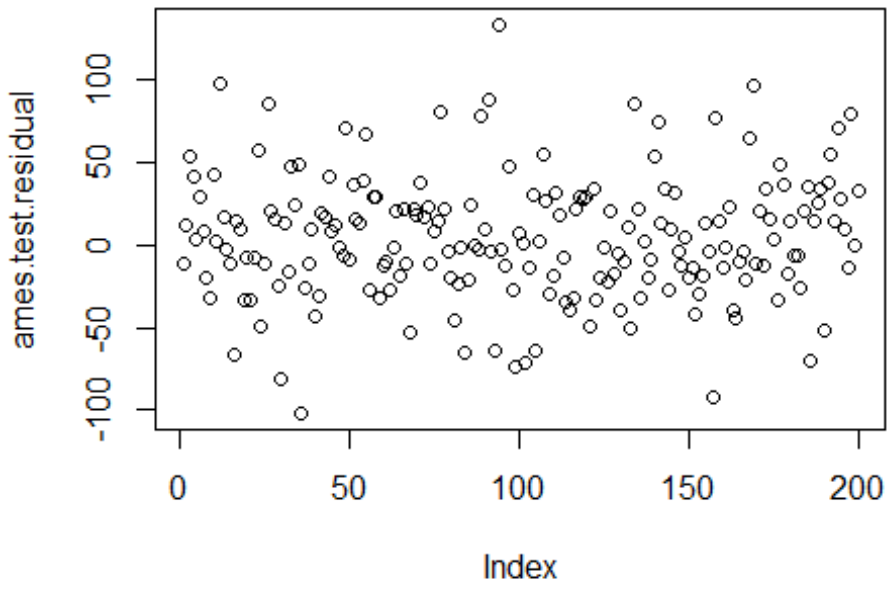


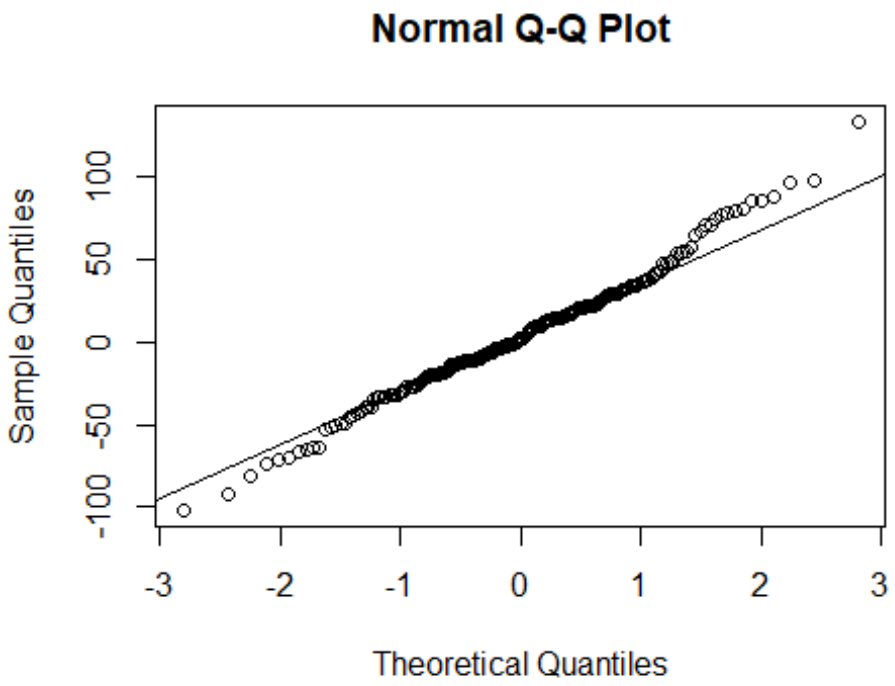

```r
hist(allsubmod$residuals)
```



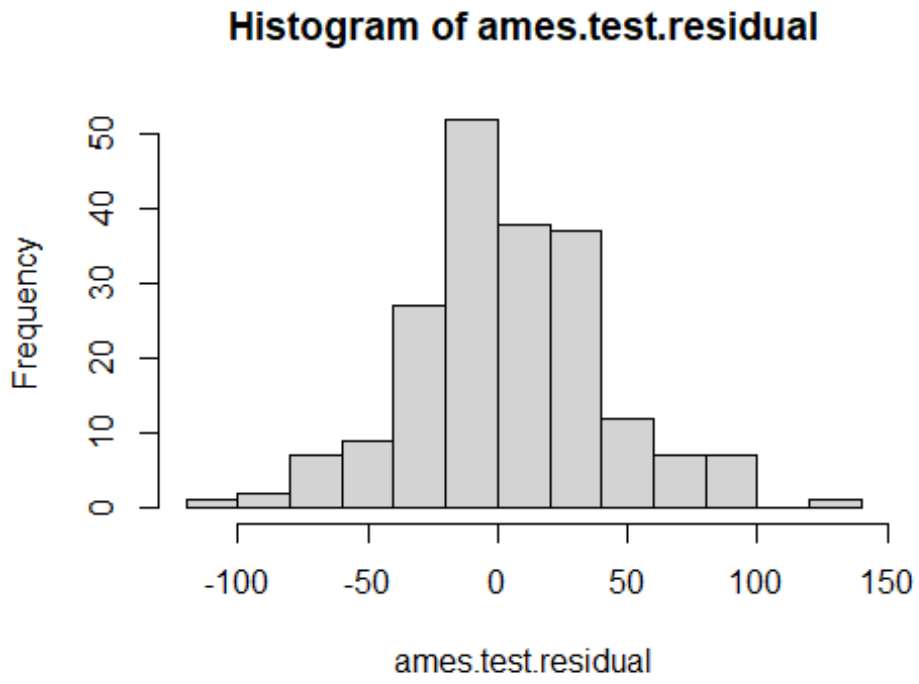

```
plot(ames.test.residual)
```



```
qqnorm(ames.test.residual)
qqline(ames.test.residual)
```




```
hist(ames.test.residual)
```




- Are any holdout cases especially poorly predicted by the training model? If so, identify by the row number(s) in the holdout data. *The biggest holdout case is 94 with a residual value of positive 133.758. Based on the cook's distance plot in the previous question, there does not appear to be any points outside of 0.5 of cook's distance, so that is good.*

```
head(sort(ames.test.residual), decreasing=FALSE, 10)
```

```
##         36        157         30         99        102        186
16
## -101.95333  -91.94209  -81.53233  -73.62176  -71.50196  -69.83581  -
66.29810
##         84         93        105
##  -65.74621  -64.22883  -63.97843
```

```
tail(sort(ames.test.residual), decreasing=TRUE, 10)
```

```
##       158        89       198        77       134        26        91
169
##  76.78719  78.43109  80.17102  80.82564  85.24867  86.12330  88.21376
96.82794
##        12        94
##  98.54538 133.75848
```

- Compute the correlation between the predicted values and actual prices for the holdout sample. This is known as the cross-validation correlation. We don't expect the training model to do better at predicting values different from those that were used to build it (as reflected in the original $R^2$), but an effective model shouldn't do a lot worse

at predicting the holdout values. Square the cross-validation correlation to get an $R^2$ value and subtract it from the original $R^2$ of the training sample. This is known as the shrinkage. We won't have specific rules about how little the shrinkage should be, but give an opinion on whether the shrinkage looks OK to you or too large in your situation.

*The shrinkage is 0.3279954, which is pretty bad It could definitely be better. If it was closer to 0, it would mean that the model is going a better job at predicting the new data compared to the original data. We should aim for a shrinkage under 1%, and this is at least 32%, which is pretty bad.*

```
crosscor = cor(amestest24$Price, ames.test.residual)

summary(allsubmod)$r.squared

## [1] 0.6301369

crosscor^2

## [1] 0.3021415

Shrinkage = summary(allsubmod)$r.squared - crosscor^2
Shrinkage

## [1] 0.3279954
```

## Part 9. Final Model

Again, you may choose to make some additional adjustments to your model after considering the final residual analysis. If you do so, please explain what (and why) you did and provide the summary() for your new final model.

Suppose that you are interested in a house in Ames, Iowa that has characteristics listed below and want to find a 95% prediction interval for the price of this house.

A 2 story 9 room home, built in 1995 and remodeled in 2003 on a 17450 sq. ft. corner lot with 300 feet of road frontage. Overall quality is good (7) and condition is average (5). The quality and condition of the exterior are both good (Gd) and it has a poured concrete foundation. There is an 875 sq. foot basement that has excellent height, but is completely unfinished and has no bath facilities. Heating comes from a gas air furnace that is in excellent condition and there is central air conditioning. The house has **2147 sq. ft.** (fixed from homework #3) of living space above ground, 1164 on the first floor and 983 on the second, with 3 bedrooms, 2 full and one half baths, and 1 fireplace. The 1 car, built-in garage has 304 sq. ft. of space and is average (TA) for both quality and construction. The only porches or decks is a 274 sq. ft. open porch in the front.

*With 95% confidence we can predict based on our model that the price of the described house is between 198.21 thousand dollars and 298.9756 thousand dollars*

```
Mod1 = lm(formula = log(Price) ~ factor(Quality) + I(log(GroundSF)) +
    YearBuilt + I(log(LotArea)) + factor(Condition) + BasementFinSF +
```

```
    GarageCars + Porch + BasementSF + Fireplaces + KitchenQ +
    CentralAir + Bedroom + SecondSF + HouseStyle + HeatingQC +
    GarageC + LotFrontage + factor(Quality):YearBuilt +
YearBuilt:factor(Condition),
    data = AmesTrain5)

newpoint.mod1 = data.frame(Quality=7, GroundSF=2147, YearBuilt=1995,
LotArea=17450, Condition=5, BasementFinSF=0, GarageCars=1, Porch=274,
BasementSF=875, Fireplaces=1, KitchenQ="TA", CentralAir="Y", Bedroom=3,
SecondSF=983, HouseStyle="2Story", HeatingQC="Ex", GarageC= "TA",
LotFrontage=300)

predict.lm(Mod1, newpoint.mod1, interval = "prediction", level = 0.95)

## Warning in predict.lm(Mod1, newpoint.mod1, interval = "prediction", level
=
## 0.95): prediction from a rank-deficient fit may be misleading

##        fit      lwr      upr
## 1 5.494844 5.289327 5.700362

exp(5.289327)

## [1] 198.21

exp(5.700362)

## [1] 298.9756
```