STOR 455 Homework #5

40 points - Due Friday 3/11 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.

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

```
library(readr)
AmesTest15 <- read_csv("AmesTest15.csv")

## Rows: 200 Columns: 42

## -- Column specification --------------------------------------------
------------
## Delimiter: ","
## chr (15): LotConfig, HouseStyle, ExteriorQ, ExteriorC, Foundation, B
asementH...
## dbl (27): Order, Price, LotFrontage, LotArea, Quality, Condition, Ye
arBuilt,...

##
## i Use `spec()` to retrieve the full column specification for this da
ta.
## i Specify the column types or set `show_col_types = FALSE` to quiet
this message.

AmesTrain15 <- read_csv("AmesTrain15.csv")

## Rows: 600 Columns: 42
```

```
## -- Column specification --------------------------------------------
------------
## Delimiter: ","
## chr (15): LotConfig, HouseStyle, ExteriorQ, ExteriorC, Foundation, B
asementH...
## dbl (27): Order, Price, LotFrontage, LotArea, Quality, Condition, Ye
arBuilt,...

##
## i Use `spec()` to retrieve the full column specification for this da
ta.
## i Specify the column types or set `show_col_types = FALSE` to quiet
this message.
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
#remove the categorical variables
AmesTrain15num=select_if(AmesTrain15,is.numeric)
AmesTest15num=select_if(AmesTest15,is.numeric)
#remove the Order variable and predictors that are exactly related
AmesTrain15new=subset(AmesTrain15num,select =-c(Order,BasementSF,Ground
SF))
AmesTest15new=subset(AmesTest15num,select =-c(Order,BasementSF,GroundSF
))

mod1 = lm(Price~., data=AmesTrain15new)
MSE = (summary(mod1)$sigma)^2
none = lm(Price~1, data=AmesTrain15new)
stepwise_mod = step(none, scope=list(upper=mod1), scale=MSE,trace=FALSE
)
summary(stepwise_mod)
```

```
##
## Call:
## lm(formula = Price ~ Quality + FirstSF + SecondSF + YearBuilt +
##     BasementFinSF + GarageSF + LotArea + YearRemodel + Condition +
##     ScreenPorchSF + LotFrontage + BasementUnFinSF + EnclosedPorchSF
+
##     Bedroom + Fireplaces, data = AmesTrain15new)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -163.707  -15.529   -1.181   12.460  182.143
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -1.541e+03  1.478e+02 -10.426  < 2e-16 ***
## Quality          1.338e+01  1.356e+00   9.865  < 2e-16 ***
## FirstSF          7.081e-02  6.676e-03  10.607  < 2e-16 ***
## SecondSF         5.619e-02  4.373e-03  12.850  < 2e-16 ***
## YearBuilt        4.340e-01  6.360e-02   6.824 2.22e-11 ***
## BasementFinSF    3.397e-02  5.025e-03   6.760 3.34e-11 ***
## GarageSF         3.602e-02  6.895e-03   5.223 2.45e-07 ***
## LotArea          7.098e-04  1.034e-04   6.862 1.74e-11 ***
## YearRemodel      3.087e-01  7.794e-02   3.961 8.37e-05 ***
## Condition        4.302e+00  1.188e+00   3.622 0.000318 ***
## ScreenPorchSF    5.883e-02  1.782e-02   3.302 0.001017 **
## LotFrontage      1.049e-01  3.711e-02   2.828 0.004851 **
## BasementUnFinSF  1.311e-02  4.612e-03   2.843 0.004632 **
## EnclosedPorchSF  5.098e-02  2.238e-02   2.278 0.023064 *
## Bedroom         -3.549e+00  1.942e+00  -1.827 0.068144 .
## Fireplaces       3.934e+00  2.182e+00   1.803 0.071897 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 27.42 on 584 degrees of freedom
## Multiple R-squared:  0.863,  Adjusted R-squared:  0.8595
## F-statistic: 245.2 on 15 and 584 DF,  p-value: < 2.2e-16
```

- 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 homework #3 group number, and AmesTest?? also using your homework #3 group number.

```
fitprice=predict(stepwise_mod,newdata=AmesTest15new)
head(fitprice)

##         1         2         3         4         5         6
## 164.81287 258.99289 238.88988  71.21457 232.09380 113.77276
```

- Compute the residuals for the 200 holdout cases.

```
holdoutresid=AmesTest15new$Price - fitprice
head(holdoutresid)

##          1          2          3          4          5          6
## -19.812868 -13.292892  21.110121  31.985434 -25.793799   5.227237
```

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

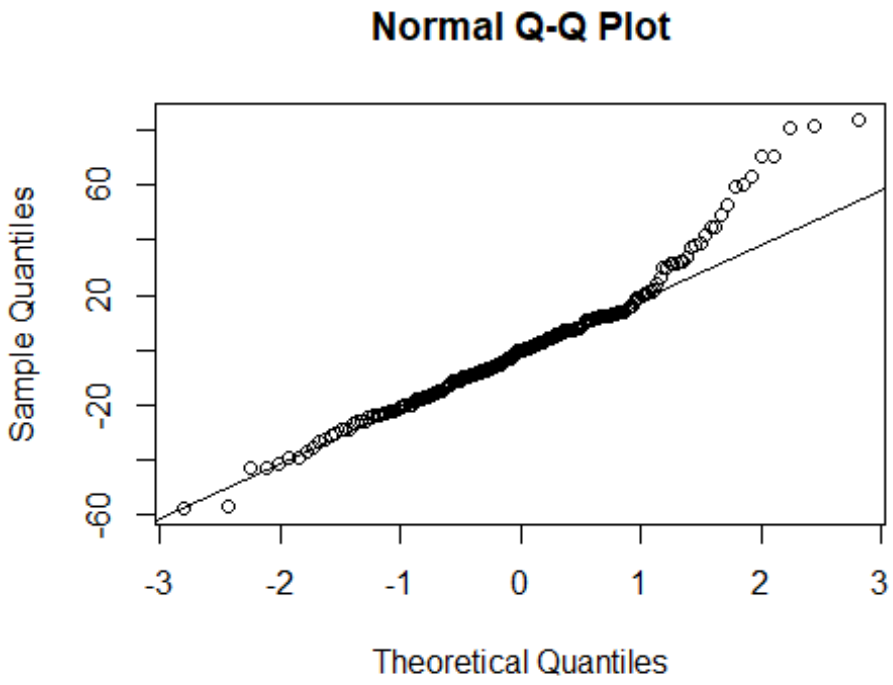
The mean value of residuals in holdout residuals is problematic. It is not close to zero, which can not satisfy one of the linearity conditions. the standard deviation is close in light of the large value of price.

```
mean(holdoutresid)

## [1] 1.116639

mean(stepwise_mod$resid)

## [1] 2.043195e-15

sd(holdoutresid)

## [1] 24.23439

summary(stepwise_mod)$sigma

## [1] 27.41554
```

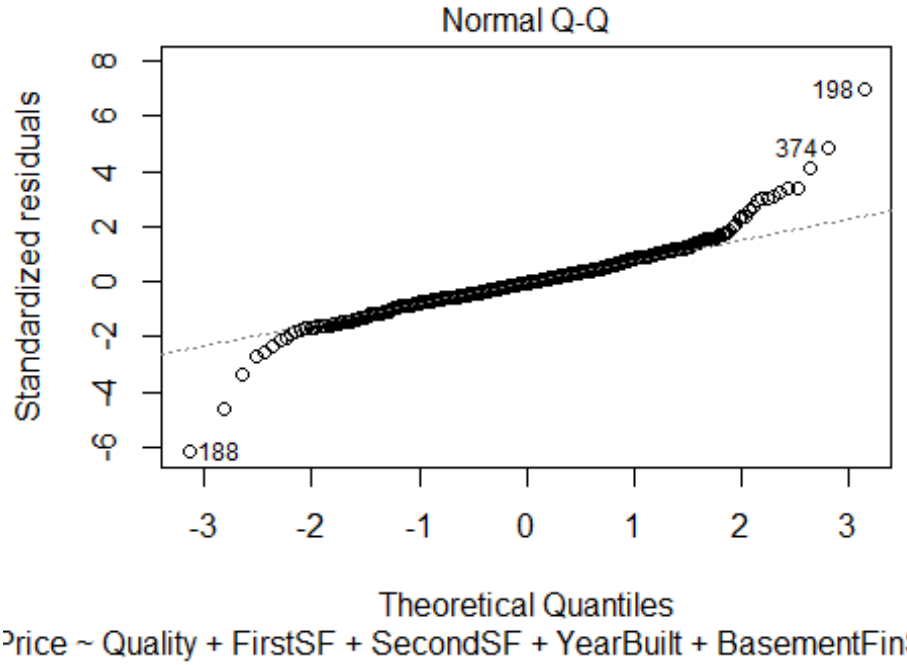
- Construct a plot of the residuals to determine if they are normally distributed. Is this plot what you expect to see considering the training model?

  Both of them have some deviations at each of the tails of the plot, showing some skewness of possible concern given the sample size of the data, so roughly they are identical.

```
qqnorm(holdoutresid)
qqline(holdoutresid)
```



```
plot(stepwise_mod,2)
```



Price ~ Quality + FirstSF + SecondSF + YearBuilt + BasementFinSF +

- Are any holdout cases especially poorly predicted by the training model? If so, identify by the row number(s) in the holdout data.

```
head(sort(holdoutresid, decreasing=TRUE),n=30)
```

```
##       77       12       49      140       35       94       55
141
## 83.82793 81.76708 80.81078 70.71247 70.54166 63.00661 60.45080 59.68
012
##      198       54      134      194       91      135       26
120
## 53.05879 49.12220 44.87337 44.54535 41.92624 38.55057 37.92810 37.54
159
##      158       89        4      177      137      122      155
50
## 34.08997 32.21871 31.98543 31.88169 31.41236 31.30337 30.13797 29.82
775
##      169       75      146       13        3      118
## 26.46902 24.32833 21.66742 21.11996 21.11012 20.74599
```

- 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 multiple 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 looks good because the calue is small. And the model is better when applied into the holdout sample because the shrinkage is a negative number.

```
crosscorr=cor(AmesTest15new$Price,fitprice)
crosscorr^2

## [1] 0.8785538

shrinkage = summary(stepwise_mod)$r.squared-crosscorr^2
shrinkage

## [1] -0.01557954
```

Part 7. Find a "fancy model":

Please see the group work. Use AmesTrain??.csv, where ?? corresponds to your *new* homework #5 group number. In addition to the quantitative predictors, you may now consider models with

- Categorical variables - Just put these in the model and let R take care of making the indicator predictors (and picking one category to leave out). Use factor( ) to treat a numeric variable as categorical. You'll see the coefficients for each indicator when you look at the summary( ) and they will be grouped together in the ANOVA. Be careful, since adding a single categorical variable with a lot of categories might actually be adding a lot of new indicator terms.

- Transformations of predictors - You can include functions of quantitative predictors. Probably best to use the I( ) notation so you don't need to create new columns when you run the predictions for the test data.

- Transformations of the response - You might address curvature or skewness in residual plots by transforming the response prices with a function like log(Price), sqrt(Price), Price^2, etc.. These should generally not need the I( ) notation to make these adjustments. IMPORTANT: If you transform Price, be sure to reverse the transformation when making final predictions!

- Combinations of variables - This might include interactions or other combinations. You do not need the I( ) notation when making an interaction using a categorical predictor (e.g. GroundSF*CentralAir).

Keep general track of the approaches you try and explain what guides your decisions as you select a new set of predictors (but again you don't need to give full details of every model you consider). Along the way you should consider some residual analysis.

Notes/Tips:

- WARNING: When using a categorical predictor with multiple categories in regsubsets( ), R will create indicators and treat them as separate predictors when deciding which to put into a model. So you might get a model with quantitative predictors like LotArea and GroundSF along with specific indicators like GarageQTA and HouseStyle1Story. This may not be very useful, since we should generally use all indicators for a categorical predictor if we include one in the model. On the other hand, when using the step( ) function, R will generally keep the multiple indicators for different categories of the same variable together as a unit. 不要使用子集法，用stepwise会包含分类变量的全部类别。

- In some cases the indicators created for different categorical variables will have identical values. For example, if you include both GarageC and GarageQ in a model, R will produce values for each of the indicators. The indicators for GarageQNone and GarageCNone (equal to one only for houses that don't have a garage) will be identical. This may be handled differently in R depending on the procedure. regsubsets( ) may give a "warning" about variables being linearly dependent. You can still use the results, just be aware that some variables are completely dependent. lm( ) might give output with coefficients (and tests) of some predictors listed as NA. This is not a problem, R is just automatically deleting one of the redundant variables. If you are predicting for a house with no garage you might have a coefficient to use for GarageQNone but then you don't need to worry about having one for GarageCNone. 不要担心完全相关或者高度相关的变量，R会自动处理并移除他们。

- If your residual analysis from homework #3 or an early model here suggest you might want to do a transformation for the response variable (Price), do so *before* fitting a lot more models. No sense fine tuning a set of predictors for Price, then deciding you should be predicting log(Price) or Price^2. So make that decision fairly early, but don't get too picky and expect to get perfect plot of residuals versus fits or an exact normal quantile plot. 尽早转换因变量，没必要微调预测变量，最后线性条件不完美也没事。

- Similarly, if you decide that some data cases should be dropped from the training set, don't wait until late in the process to do so. For example, if you spot a *very* large residual you should look at the characteristics for that house to see if it should be deleted. Don't forget about the value of simple plots (like a scatterplot of Price vs. LotArea) for helping to see what is going on and recognize extreme cases. Be sure to document any adjustments you make in the final report. 先做残差分析，并且观察每个特征和因变量的plot图集，早点删除含影响点的特征。

- Comparing C_p from different predictor pools - While Mallow's C_p is a useful tool for comparing models from the same pool of predictors. You should not use it to compare models based on different predictor pools. For example, if you add a bunch of categorical variables to all the quantitative predictors from homework #3 to make a new "full" model, then find C_p from a model that you fit in homework #3, it will be worse than it was before. If you look at the formula for calculating C_p, you will see that all that has changed is MSE for the full model after adding the new batch of predictors.
不要比较自变量集不同的MLCP，比如移植hw3的预测变量并从中找mlcp,因为这么做只是让MSE改变了（看MLCP公式）

- I should be able to follow the steps you use when selecting a model. I certainly don't need to see every bit of output, but it might help to include more of the R commands you use. For example, saying you used backward elimination is not very helpful when I don't know what you start with for the full model or pool of predictors (e.g. did you include Condition and Quality as numeric predictors? or did you decide to eliminate one of GroundSF, FirstSF, or SecondSF due to redundancy?). The easiest way to convey this in many cases is to show the R command you used. It is fine to abbreviate the output (for example, delete many steps in a stepwise procedure using trace=FALSE), but it would be helpful if you identified the parts you do include. For example, a sentence like "After 12 steps of the stepwise procedure, we have the output below for the fitted model." Similarly, I don't need to see 600 residuals, using head and sort can show the important ones. 每一步R命令要写清楚，可以省略不必要的处理过程（选择方法）或者大量数据（不显示所有，而是用head只显示部分残差值），让阅卷人能顺利跟踪建模的思路。

- Once you have settled on a response, made adjustments to the data (if needed), and chosen a set of predictors, be sure to include the summary( )for your "fancy" model at this stage. 最终的fancy model务必包含summary.

Part 8: Cross-validation for your "fancy" model

Please see the group work. Redo the cross-validation analysis with your test data for your new fancy model. Use AmesTest??.csv, where ?? corresponds to your new group number from homework #5. Discuss how the various measures (mean of residuals, std. dev of residuals, shape of the distributions of residuals, cross-validation correlation, and shrinkage) compare to the results you had for your basic model. Don't worry about looking for poorly predicted cases this time. If you transformed the response variable, consider how to take this into account for your residual analysis. In order to compare residuals they should have the same units! 交叉检验时，如果trans了因变量，分析残差最后要统一单位。

Note on missing categories:

测试数据中的分类变量的类别可能在训练数据中没有，这时try whatever R uses as the “left out” reference category.

> When creating the predictions using predict(yourmodel,AmesTest) you may see an error like:
>
> Error in model.frame.default(Terms, newdata, na.action = na.action, xlev = object$xlevels) : factor HouseStyle has new levels 1.5Unf, 2.5Fin, 2.5Unf
>
> This occurs because the holdout sample has a value for the categorical variable that was not present in your training sample, so there is no indicator in your model to handle that case. To get a prediction for that house, you’ll need to switch the category to one that is in your training data. In the example above you might choose to replace the “2.5Fin” house style with “2Story”. If you are not sure what category to use, try whatever R uses as the “left out” reference category. Be sure to record any changes like this that you make.

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 11 room home, built in 1983 and remodeled in 1999 on a 21540 sq. ft. lot with 400 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 757 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 2432 sq. ft. of living space above ground, 1485 on the first floor and 947 on the second, with 4 bedrooms, 2 full and one half baths, and 1 fireplace. The 2 car, built-in garage has 588 sq. ft. of space and is average (TA) for both quality and construction. The only porches or decks is a 384 sq. ft. open porch in the front.

In terms of the plot of our group fancy model, the linearity condition is pretty good. The red line seems to be roughly horizontal at zero for the residuals.

Constant variance seems to be good. From the residuals vs fitted values plot we can see a similar spread above/below the red curve for all fitted values.

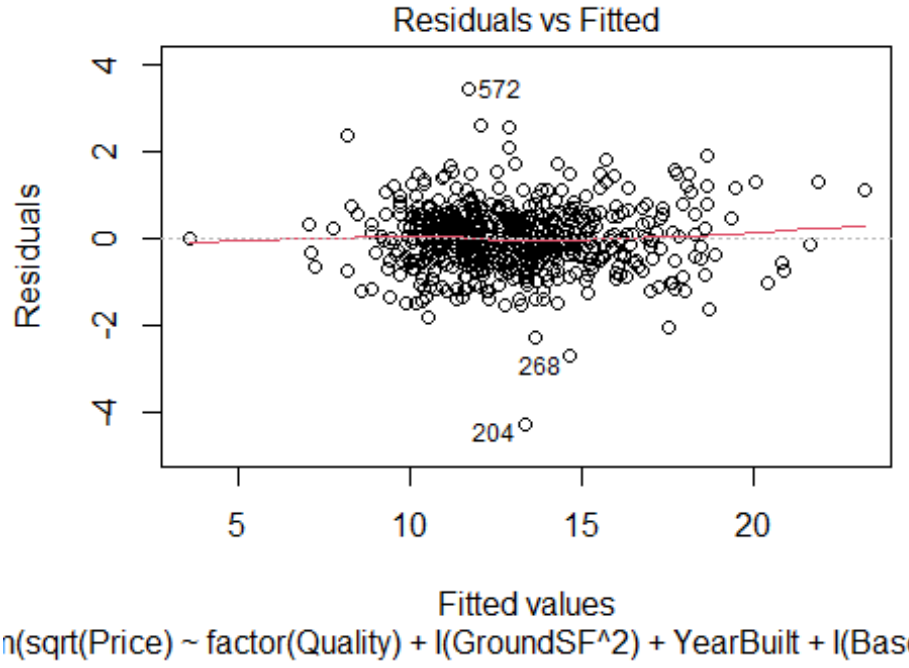
Normality of residuals is good since the data in the qqplot roughly fit the qqline. There are some deviations at each of the tails of the plot, showing some skewness of possible concern given the larger sample size of the data.

With 95% confidence I predict that the price of this house is between $207.3721 and $314.8616

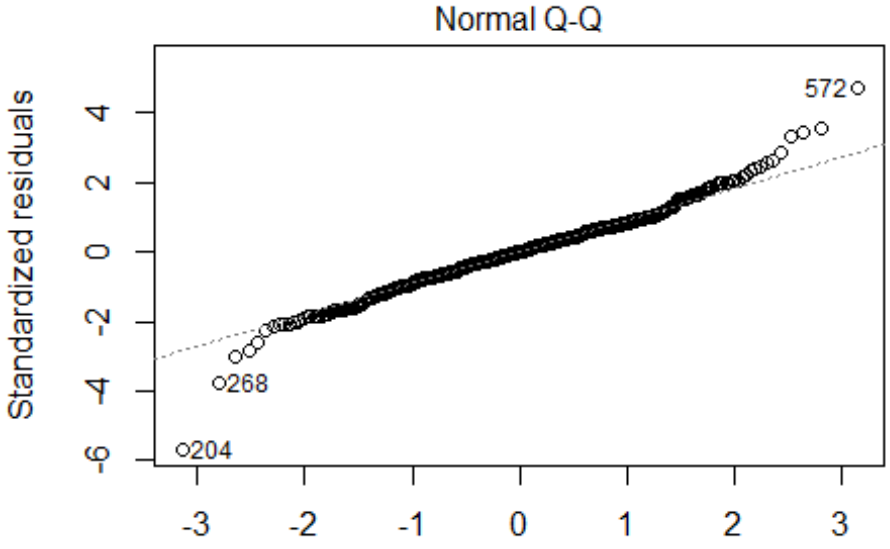
```
AmesTrain2 <- read.csv("AmesTrain2.csv")
Adjusted_TrainData = select(AmesTrain2, !Order & !BasementFinSF &
!BasementUnFinSF & !FirstSF & !SecondSF)

fancymodel = lm(formula = sqrt(Price) ~ factor(Quality) + I(GroundSF^2)
+ YearBuilt + I(BasementSF^2) + factor(Condition) + LotArea + BasementF
Bath + GarageSF + Fireplaces + YearRemodel + ScreenPorchSF + EnclosedPo
rchSF + LotFrontage + factor(Condition) * YearRemodel + GroundSF * Base
mentSF, data = Adjusted_TrainData)

summary(fancymodel)

##
## Call:
## lm(formula = sqrt(Price) ~ factor(Quality) + I(GroundSF^2) +
##     YearBuilt + I(BasementSF^2) + factor(Condition) + LotArea +
##     BasementFBath + GarageSF + Fireplaces + YearRemodel + ScreenPorc
hSF +
##     EnclosedPorchSF + LotFrontage + factor(Condition) * YearRemodel
+
##     GroundSF * BasementSF, data = Adjusted_TrainData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.2720 -0.4579  0.0023  0.4448  3.4327
##
## Coefficients: (1 not defined because of singularities)
##                                  Estimate Std. Error t value Pr(>|t|
)
## (Intercept)                    -8.131e+01  1.486e+02  -0.547 0.58449
7
## factor(Quality)3                1.491e+00  6.928e-01   2.152 0.03183
2 *
## factor(Quality)4                1.814e+00  6.509e-01   2.786 0.00551
3 **
## factor(Quality)5                2.156e+00  6.489e-01   3.322 0.00095
1 ***
## factor(Quality)6                2.197e+00  6.548e-01   3.356 0.00084
5 ***
## factor(Quality)7                2.678e+00  6.626e-01   4.041 6.07e-0
5 ***
## factor(Quality)8                3.357e+00  6.694e-01   5.015 7.12e-0
7 ***
## factor(Quality)9                4.624e+00  6.941e-01   6.662 6.46e-1
1 ***
## factor(Quality)10               5.489e+00  8.729e-01   6.287 6.49e-1
0 ***
## I(GroundSF^2)                   3.008e-07  1.460e-07   2.060 0.03981
4 *
## YearBuilt                       1.687e-02  2.146e-03   7.860 1.97e-1
```

```
4 ***
## I(BasementSF^2)                  1.409e-07  1.875e-07   0.751 0.45268
1
## factor(Condition)2              -3.578e+02  2.153e+02  -1.662 0.09711
4 .
## factor(Condition)3               4.063e+02  1.608e+02   2.528 0.01176
0 *
## factor(Condition)4               3.335e+01  1.503e+02   0.222 0.82445
4
## factor(Condition)5               1.326e+01  1.484e+02   0.089 0.92886
5
## factor(Condition)6               3.380e+01  1.489e+02   0.227 0.82054
8
## factor(Condition)7               4.433e+01  1.492e+02   0.297 0.76648
4
## factor(Condition)8               5.928e+01  1.502e+02   0.395 0.69330
0
## factor(Condition)9               2.635e-01  3.973e+00   0.066 0.94713
8
## LotArea                          2.012e-05  3.041e-06   6.617 8.55e-1
1 ***
## BasementFBath                    3.661e-01  6.456e-02   5.671 2.27e-0
8 ***
## GarageSF                         1.366e-03  1.964e-04   6.956 9.75e-1
2 ***
## Fireplaces                       4.073e-01  6.133e-02   6.641 7.38e-1
1 ***
## YearRemodel                      2.774e-02  7.637e-02   0.363 0.71650
7
## ScreenPorchSF                    1.966e-03  5.074e-04   3.876 0.00011
9 ***
## EnclosedPorchSF                  1.069e-03  6.426e-04   1.664 0.09661
4 .
## LotFrontage                      1.749e-03  1.045e-03   1.673 0.09487
0 .
## GroundSF                         2.188e-04  3.796e-04   0.576 0.56456
1
## BasementSF                      -3.042e-04  4.141e-04  -0.735 0.46294
8
## factor(Condition)2:YearRemodel   1.793e-01  1.096e-01   1.636 0.10237
5
## factor(Condition)3:YearRemodel -2.083e-01  8.241e-02  -2.528 0.01175
2 *
## factor(Condition)4:YearRemodel -1.692e-02  7.705e-02  -0.220 0.82623
8
## factor(Condition)5:YearRemodel -6.483e-03  7.611e-02  -0.085 0.93215
2
## factor(Condition)6:YearRemodel -1.677e-02  7.636e-02  -0.220 0.82626
5
## factor(Condition)7:YearRemodel -2.202e-02  7.650e-02  -0.288 0.77359
```

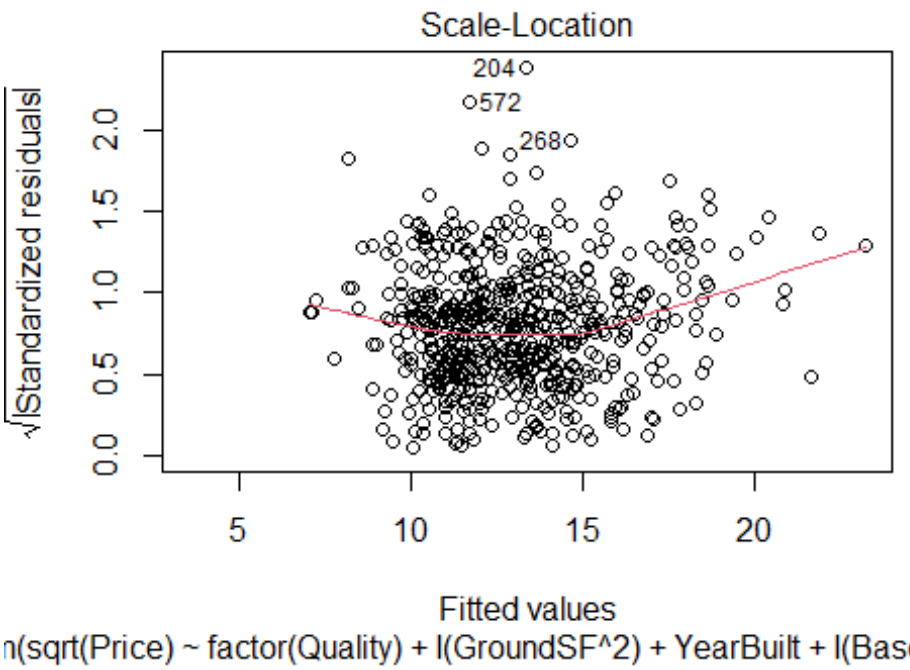
```
8
## factor(Condition)8:YearRemodel -2.950e-02  7.701e-02  -0.383 0.70186
1
## factor(Condition)9:YearRemodel         NA         NA      NA       N
A
## GroundSF:BasementSF             5.368e-07  3.080e-07   1.743 0.08194
2 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7592 on 562 degrees of freedom
## Multiple R-squared:  0.9202, Adjusted R-squared:  0.915
## F-statistic: 175.2 on 37 and 562 DF,  p-value: < 2.2e-16

plot(fancymodel)

## Warning: not plotting observations with leverage one:
##   409, 526
```

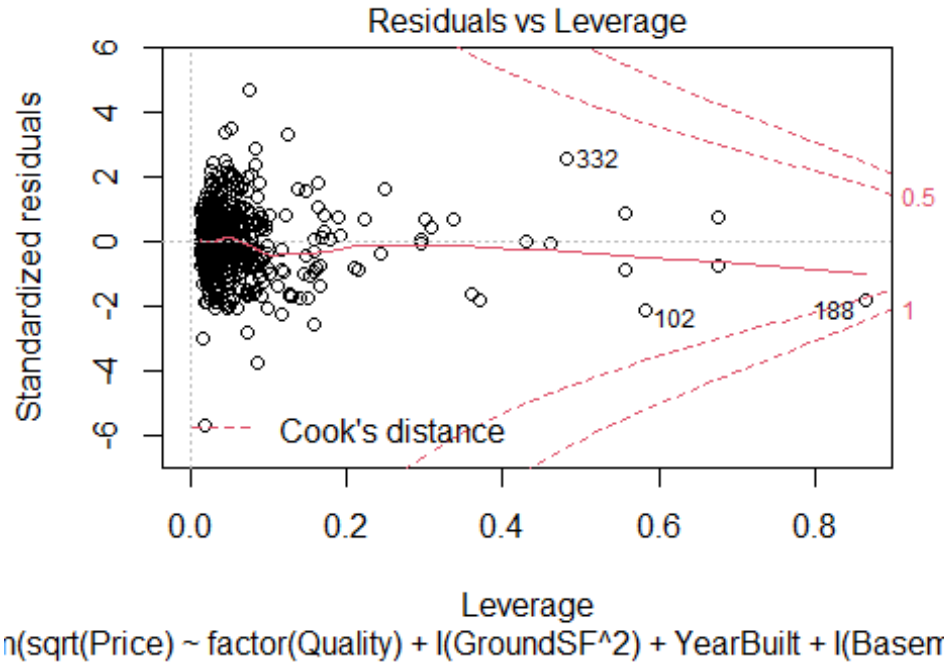


n(sqrt(Price) ~ factor(Quality) + I(GroundSF^2) + YearBuilt + I(Baseme



n(sqrt(Price) ~ factor(Quality) + I(GroundSF^2) + YearBuilt + I(Baseme






```
finalmodel = fancymodel

newxs=data.frame(YearBuilt=1983, YearRemodel=1999, LotArea=21540, LotFr
ontage=400, Quality=7, Condition=5, BasementSF=757, BasementFBath=0,Gro
```

```
undSF=2432, Fireplaces=1, GarageSF=588, ScreenPorchSF=0,EnclosedPorchSF
=0)
predict.lm(finalmodel, newxs, interval="prediction", level=0.95)
```

```
## Warning in predict.lm(finalmodel, newxs, interval = "prediction", le
vel = 0.95):
## prediction from a rank-deficient fit may be misleading
```

```
##        fit      lwr      upr
## 1 16.07238 14.40042 17.74434
```

```
14.40042^2
```

```
## [1] 207.3721
```

```
17.74434^2
```

```
## [1] 314.8616
```