Week7Assignment-Final

Pradeepta Das

21 November 2020

Let's see some data.

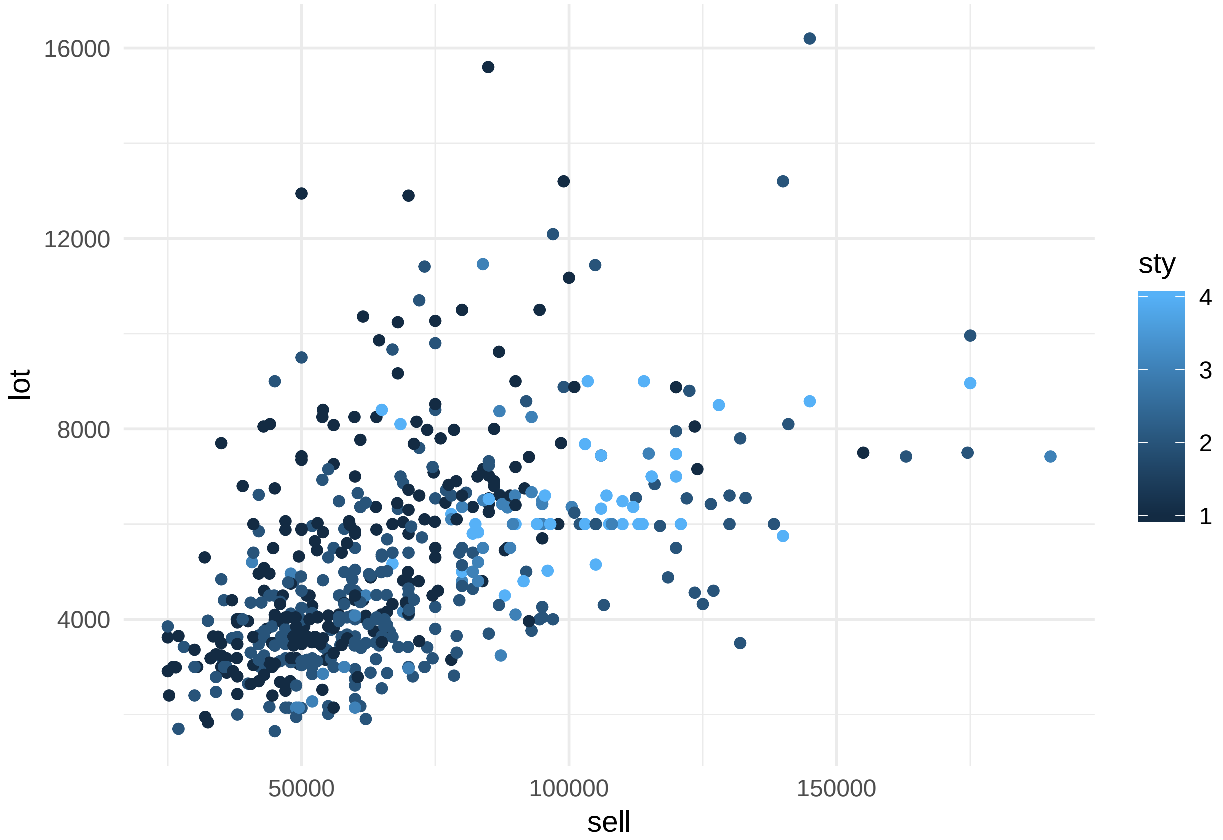
```
head(data)
```

```
##   obs  sell  lot bdms fb sty drv rec ffin ghw ca gar reg sell_LOG  lot_LOG
## 1   1 42000 5850    3  1   2   1   0    1   0  0   1   0 10.64542 8.674197
## 2   2 38500 4000    2  1   1   1   0    0   0  0   0   0 10.55841 8.294050
## 3   3 49500 3060    3  1   1   1   0    0   0  0   0   0 10.80973 8.026170
## 4   4 60500 6650    3  1   2   1   1    0   0  0   0   0 11.01040 8.802372
## 5   5 61000 6360    2  1   1   1   0    0   0  0   0   0 11.01863 8.757784
## 6   6 66000 4160    3  1   1   1   1    1   0  1   0   0 11.09741 8.333270
```

```
summary(data)
```

```
##       obs             sell             lot             bdms      
##  Min.   :  1.0   Min.   : 25000   Min.   : 1650   Min.   :1.000  
##  1st Qu.:137.2   1st Qu.: 49125   1st Qu.: 3600   1st Qu.:2.000  
##  Median :273.5   Median : 62000   Median : 4600   Median :3.000  
##  Mean   :273.5   Mean   : 68122   Mean   : 5150   Mean   :2.965  
##  3rd Qu.:409.8   3rd Qu.: 82000   3rd Qu.: 6360   3rd Qu.:3.000  
##  Max.   :546.0   Max.   :190000   Max.   :16200   Max.   :6.000  
##        fb             sty             drv             rec        
##  Min.   :1.000   Min.   :1.000   Min.   :0.000   Min.   :0.0000  
##  1st Qu.:1.000   1st Qu.:1.000   1st Qu.:1.000   1st Qu.:0.0000  
##  Median :1.000   Median :2.000   Median :1.000   Median :0.0000  
##  Mean   :1.286   Mean   :1.808   Mean   :0.859   Mean   :0.1777  
##  3rd Qu.:2.000   3rd Qu.:2.000   3rd Qu.:1.000   3rd Qu.:0.0000  
##  Max.   :4.000   Max.   :4.000   Max.   :1.000   Max.   :1.0000  
##       ffin             ghw                ca              gar        
##  Min.   :0.0000   Min.   :0.00000   Min.   :0.0000   Min.   :0.0000  
##  1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.0000  
##  Median :0.0000   Median :0.00000   Median :0.0000   Median :0.0000  
##  Mean   :0.3498   Mean   :0.04579   Mean   :0.3168   Mean   :0.6923  
##  3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.:1.0000  
##  Max.   :1.0000   Max.   :1.00000   Max.   :1.0000   Max.   :3.0000  
##       reg            sell_LOG        lot_LOG     
##  Min.   :0.0000   Min.   :10.13   Min.   :7.409  
##  1st Qu.:0.0000   1st Qu.:10.80   1st Qu.:8.189  
##  Median :0.0000   Median :11.03   Median :8.434  
##  Mean   :0.2344   Mean   :11.06   Mean   :8.467  
##  3rd Qu.:0.0000   3rd Qu.:11.31   3rd Qu.:8.758  
##  Max.   :1.0000   Max.   :12.15   Max.   :9.693  
```

```
data %>% ggplot(aes(sell, lot, color= sty)) + geom_point()
```



(a) Consider a linear model where the sale price of a house is the dependent variable and the explanatory variables are the other variables given above. Perform a test for linearity. What do you conclude based on the test result?

```
modelA <- lm(sell ~ lot + bdms + fb + sty + drv + rec + ffin + ghw + ca + gar + reg, data = data)
print(modelA.summary <- summary(modelA))
```

```
##
## Call:
## lm(formula = sell ~ lot + bdms + fb + sty + drv + rec + ffin +
##     ghw + ca + gar + reg, data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -41389  -9307   -591   7353  74875
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4038.3504  3409.4713  -1.184 0.236762
## lot             3.5463     0.3503  10.124  < 2e-16 ***
## bdms         1832.0035  1047.0002   1.750 0.080733 .
## fb          14335.5585  1489.9209   9.622  < 2e-16 ***
## sty          6556.9457   925.2899   7.086 4.37e-12 ***
## drv          6687.7789  2045.2458   3.270 0.001145 **
## rec           4511.2838  1899.9577   2.374 0.017929 *  
## ffin          5452.3855  1588.0239   3.433 0.000642 ***
## ghw          12831.4063  3217.5971   3.988 7.60e-05 ***
## ca           12632.8904  1555.0211   8.124 3.15e-15 ***
## gar           4244.8290   840.5442   5.050 6.07e-07 ***
## reg           9369.5132  1669.0907   5.614 3.19e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 15420 on 534 degrees of freedom
## Multiple R-squared:  0.6731, Adjusted R-squared:  0.6664 
## F-statistic: 99.97 on 11 and 534 DF,  p-value: < 2.2e-16
```

Linearity Test

Its the ramsay's RESET test.

```r
modelA.RESET <- resettest(modelA, power = 2, type = "fitted", data = data)
print(modelA.RESET)
```

```
## 
##  RESET test
## 
## data:  modelA
## RESET = 26.986, df1 = 1, df2 = 533, p-value = 2.922e-07
```

With a statistic of 26.986 and a p-value of ~0.000, the Ramsey's RESET test suggests that the linear model is NOT correctly specified. So we reject H_0.

Jarque-Bera (residuals normality)

```r
# Ho: The errors of the model are distributed normal
modelA.JB <- jarque.bera.test(modelA$residuals)
modelA.JB
```

```
## 
##  Jarque Bera Test
## 
## data:  modelA$residuals
## X-squared = 247.62, df = 2, p-value < 2.2e-16
```

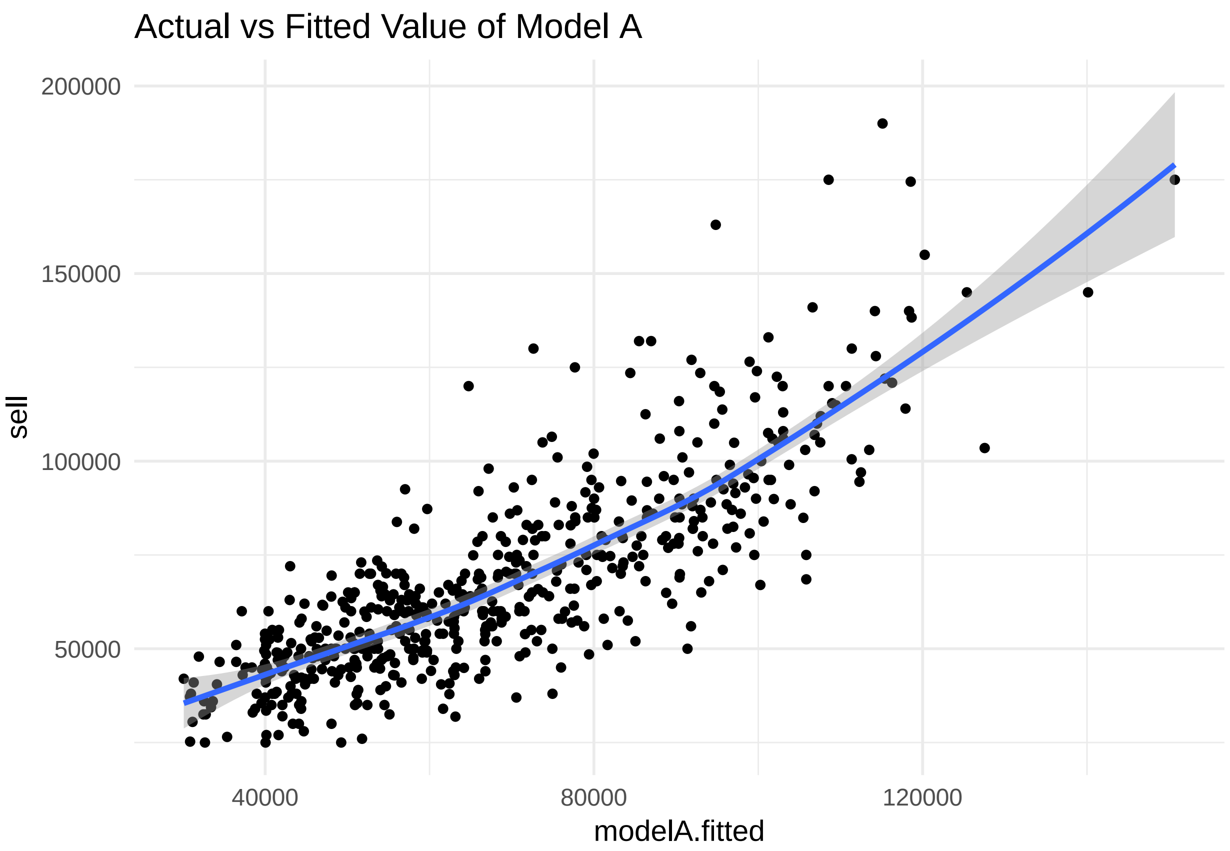
With a statistic of ~247.62 and a p-value of ~0, the Jarque-Bera test suggests that the linear model residuals are NOT normally distributed, therefore the linear model is NOT correctly specified.

Both Ramsey's RESET and Jarque-Bera tests suggest that the considered linear model is NOT correctly specified.

```r
modelA.fitted <- fitted.values(modelA)

data %>% ggplot(aes(modelA.fitted, sell)) +
    geom_point(shape=16) +
    geom_smooth() + ggtitle("Actual vs Fitted Value of Model A")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



(b) Now consider a linear model where the log of the sale price of the house is the dependent variable and the explanatory variables are as before. Perform again the test for linearity. What do you conclude now?

```
modelB <- lm(sell_LOG ~ lot + bdms + fb + sty + drv + rec + ffin + ghw + ca + gar + reg, data = data)
summary(modelB)
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot + bdms + fb + sty + drv + rec + ffin +
##     ghw + ca + gar + reg, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.67865 -0.12211  0.01666  0.12868  0.67737
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.003e+01  4.724e-02 212.210  < 2e-16 ***
## lot         5.057e-05  4.854e-06  10.418  < 2e-16 ***
## bdms        3.402e-02  1.451e-02   2.345  0.01939 *
## fb          1.678e-01  2.065e-02   8.126 3.10e-15 ***
## sty         9.227e-02  1.282e-02   7.197 2.10e-12 ***
## drv         1.307e-01  2.834e-02   4.610 5.04e-06 ***
## rec         7.352e-02  2.633e-02   2.792  0.00542 **
## ffin        9.940e-02  2.200e-02   4.517 7.72e-06 ***
## ghw         1.784e-01  4.458e-02   4.000 7.22e-05 ***
## ca          1.780e-01  2.155e-02   8.262 1.14e-15 ***
## gar         5.076e-02  1.165e-02   4.358 1.58e-05 ***
## reg         1.271e-01  2.313e-02   5.496 6.02e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2137 on 534 degrees of freedom
## Multiple R-squared:  0.6766, Adjusted R-squared:  0.6699
## F-statistic: 101.6 on 11 and 534 DF,  p-value: < 2.2e-16
```

~ zero (0) coefficient of variable lot.

```
modelB.RESET <- resettest(modelB, power = 2, type = "fitted", data = data)
print(modelB.RESET)
```

```
##
##  RESET test
##
## data:  modelB
## RESET = 0.27031, df1 = 1, df2 = 533, p-value = 0.6033
```

With a statistic of ~0.27 and a p-value of ~0.6033, the Ramsey's RESET test suggests that the second linear model might be correctly specified (H_0 of correct/linear specification NOT rejected, at the 5% level of significance).

```
# Ho: The errors of the model are distributed normal
modelB.JB <- jarque.bera.test(modelB$residuals)
modelB.JB
```

```
##
##  Jarque Bera Test
##
## data:  modelB$residuals
## X-squared = 8.4432, df = 2, p-value = 0.01467
```

With a statistic of ~8.443 and a p-value of ~0.0147, the Jarque-Bera test suggests that the linear model residuals are still NOT normally distributed, therefore the linear model is still NOT correctly specified, althought that the second model's JB statistic is significantly decreased (and therefore the model significantly improved).

Conclusion:

Both Ramsey's RESET and Jarque-Bera tests suggest that the second model is significantly improved than the model considered first.

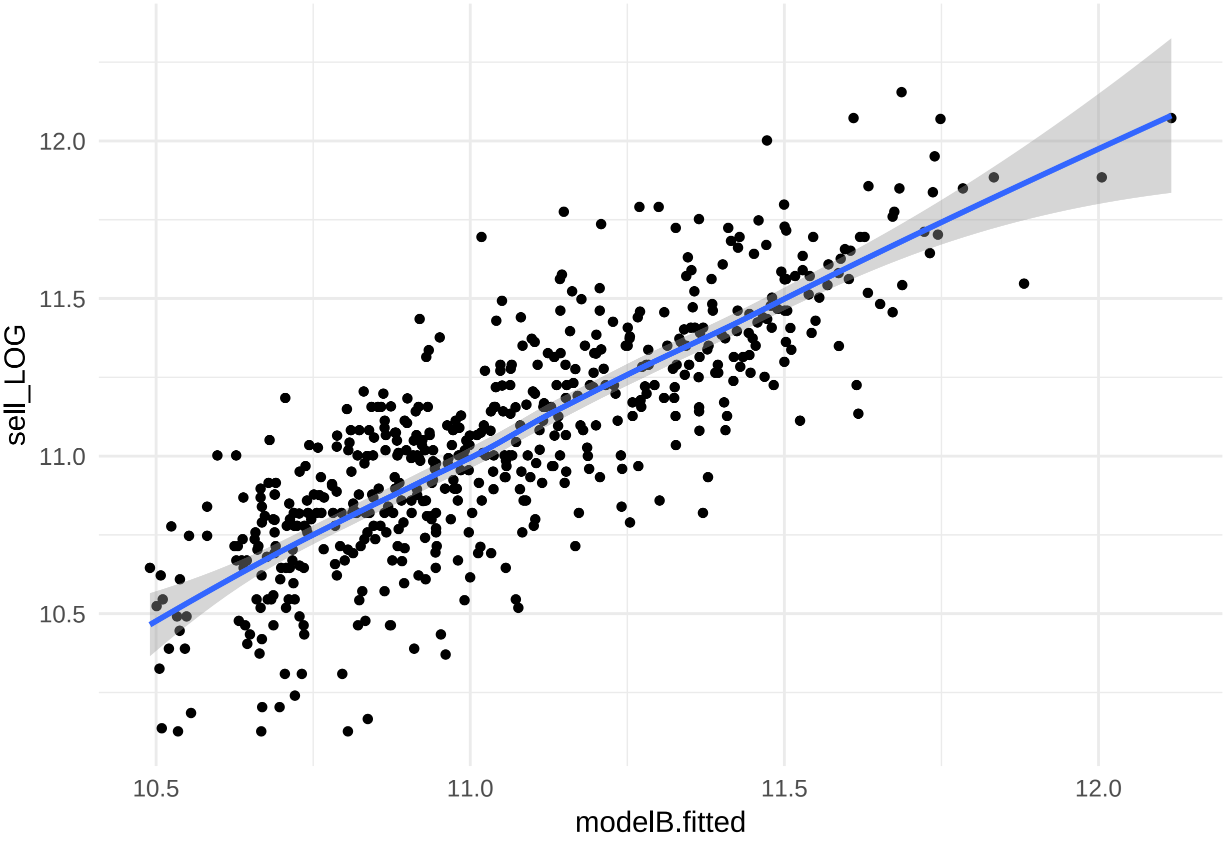
The Ramsey's RESET test suggests that the second linear model might be correctly specified, while the Jarque-Bera test suggests that it is still NOT correctly specified (although significantly improved).

This is also intuitively demonstrated by the second model real to fitted-values diagram shown below (looks much more like a linear relationship than before).

```
modelB.fitted <- fitted.values(modelB)

ggplot(data, aes(x=modelB.fitted, y=sell_LOG)) +
    geom_point(shape=16) +
    geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



(c) Continue with the linear model from question (b). Estimate a model that includes both the lot size variable and its logarithm, as well as all other explanatory variables without transformation. What is your conclusion, should we include lot size itself or its logarithm?

```r
# Estimating third model.
modelC <- lm(sell_LOG ~ lot + lot_LOG + bdms + fb + sty + drv + rec + ffin + ghw + ca + gar + reg,
             data = data)
print(modelC.summary <- summary(modelC))
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot + lot_LOG + bdms + fb + sty + drv +
##     rec + ffin + ghw + ca + gar + reg, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68573 -0.12380  0.00785  0.12521  0.68112
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.150e+00  6.830e-01  10.469  < 2e-16 ***
## lot         -1.490e-05  1.624e-05  -0.918 0.359086
## lot_LOG      3.827e-01  9.070e-02   4.219 2.88e-05 ***
## bdms         3.489e-02  1.429e-02   2.442 0.014915 *
## fb           1.659e-01  2.033e-02   8.161 2.40e-15 ***
## sty          9.121e-02  1.263e-02   7.224 1.76e-12 ***
## drv          1.068e-01  2.847e-02   3.752 0.000195 ***
## rec          5.467e-02  2.630e-02   2.078 0.038156 *
## ffin         1.052e-01  2.171e-02   4.848 1.64e-06 ***
## ghw          1.791e-01  4.390e-02   4.079 5.20e-05 ***
## ca           1.643e-01  2.146e-02   7.657 9.01e-14 ***
## gar          4.826e-02  1.148e-02   4.203 3.09e-05 ***
## reg          1.344e-01  2.284e-02   5.884 7.10e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2104 on 533 degrees of freedom
## Multiple R-squared:  0.687,  Adjusted R-squared:   0.68
## F-statistic: 97.51 on 12 and 533 DF,  p-value: < 2.2e-16
```

Notice the ~ zero (0) coefficient of variable lot.

```r
# Model Ramsey's RESET testing.
modelC.RESET <- resettest(modelC, power = 2, type = "fitted", data = data)
print(modelC.RESET)
```

```
##
##  RESET test
##
## data:  modelC
## RESET = 0.06769, df1 = 1, df2 = 532, p-value = 0.7948
```

With a statistic of ~0.068 and a p-value of ~0.7948, the Ramsey's RESET test suggests that the third linear model might be correctly specified (H0 of correct/linear specification NOT rejected, at the 5% level of significance).

It also suggests that this is the best model constructed so far, as it has the lowest statistic and the highest p-value scored by all Ramsey's RESET tests ran so far.

```r
# Model Jarque-Bera testing.
modelC.JB <- jarque.bera.test(modelC.summary$residuals)
print(modelC.JB)
```

```
##
##  Jarque Bera Test
##
## data:  modelC.summary$residuals
## X-squared = 9.3643, df = 2, p-value = 0.009259
```

With a statistic of ~9.364 and a p-value of ~0.0093, the Jarque-Bera test suggests that the linear model residuals are still NOT normally distributed; therefore the linear model is still NOT correctly specified.

No further model improvement is indicated by the Jarque-Bera residuals normality test; in fact the second model's residuals were slightly more normal than the third's.

Conclusion:

Both Ramsey's RESET and Jarque-Bera tests suggest that the third model is significantly improved than the model considered first, while the Ramsey's RESET test suggests that it is even more improved than the model considered second.

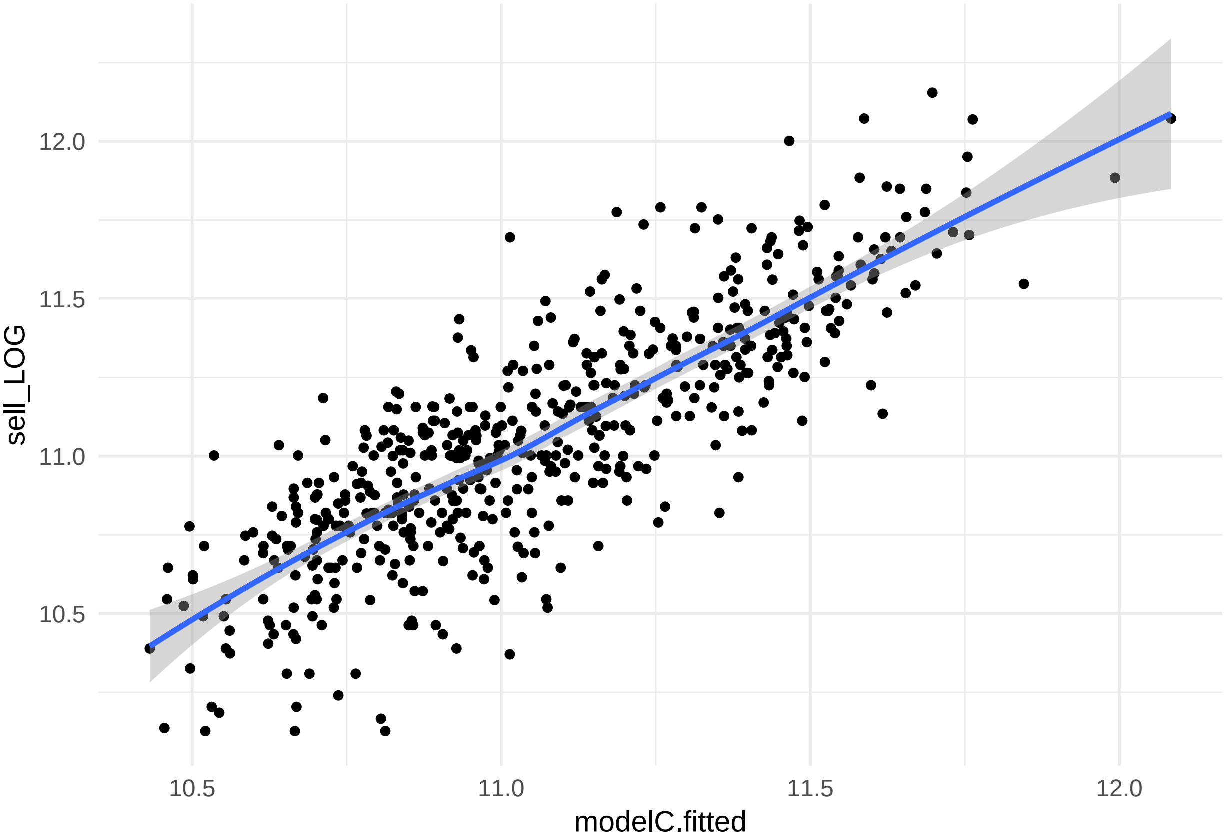
The Ramsey's RESET test suggests that the third linear model might be correctly specified, while the Jarque-Bera test suggests that it is still NOT correctly specified.

This is also intuitively demonstrated by the third model real to fitted-values diagram shown at the next page (looks about the same or more like a linear relationship than before).

```
modelC.fitted <- fitted.values(modelC)

ggplot(data, aes(x=modelC.fitted, y=sell_LOG)) +
    geom_point(shape=16) +
    geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Conclusion:

It is concluded that it would be better to include the lot size logarithm in the model, rather than the lot size variable itself, due to the following reasons:

The three models testing performed so far, see Table 4 (above): "Models linearity test results' comparison chart". The Ramsey's RESET tests showed that the lot size logarithm variable significantly improves the model linearity, while the Jarque-Bera tests showed that it produces a satisfactory (so far) level of residuals normality.

The (much better) lot size logarithm variable coefficient p-value (0), compared to the lot size variable itself coefficient p-value (0.359), when used together. See Table 5 (above): "Third model lot related variables' coefficients' comparison chart".

The fact that lot variable ended with a ~zero (0) coefficient anyway at the (improved) second and third models.

(d) Consider now a model where the log of the sale price of the house is the dependent variable and the explanatory variables are the log transformation of lot size, with all other explanatory variables as before. We now consider interaction effects of the log lot size with the other variables. Construct these interaction variables. How many are individually significant?

```r
# Estimating fourth model.
modelD <- lm(sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec + ffin + ghw + ca + gar + reg +
                 lot_LOG * bdms + lot_LOG * fb + lot_LOG * sty + lot_LOG * drv + lot_LOG * rec +
                 lot_LOG * ffin + lot_LOG * ghw + lot_LOG * ca + lot_LOG * gar + lot_LOG * reg,
             data = data)
print(modelD.summary <- summary(modelD))
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec +
##     ffin + ghw + ca + gar + reg + lot_LOG * bdms + lot_LOG *
##     fb + lot_LOG * sty + lot_LOG * drv + lot_LOG * rec + lot_LOG *
##     ffin + lot_LOG * ghw + lot_LOG * ca + lot_LOG * gar + lot_LOG *
##     reg, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68306 -0.11612  0.00591  0.12486  0.65998
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   8.966499   1.070667   8.375 5.09e-16 ***
## lot_LOG       0.152685   0.128294   1.190   0.2345
## bdms          0.019075   0.326700   0.058   0.9535
## fb           -0.368234   0.429048  -0.858   0.3911
## sty           0.488885   0.309700   1.579   0.1150
## drv          -1.463371   0.717225  -2.040   0.0418 *
## rec           1.673992   0.655919   2.552   0.0110 *
## ffin         -0.031844   0.445543  -0.071   0.9430
## ghw          -0.505889   0.902733  -0.560   0.5754
## ca           -0.340276   0.496041  -0.686   0.4930
## gar           0.401941   0.258646   1.554   0.1208
## reg           0.118484   0.479856   0.247   0.8051
## lot_LOG:bdms  0.002070   0.038654   0.054   0.9573
## lot_LOG:fb    0.062037   0.050145   1.237   0.2166
## lot_LOG:sty  -0.046361   0.035942  -1.290   0.1977
## lot_LOG:drv   0.191542   0.087361   2.193   0.0288 *
## lot_LOG:rec  -0.188462   0.076373  -2.468   0.0139 *
## lot_LOG:ffin  0.015913   0.052851   0.301   0.7635
## lot_LOG:ghw   0.081135   0.106929   0.759   0.4483
## lot_LOG:ca    0.059549   0.058024   1.026   0.3052
## lot_LOG:gar  -0.041359   0.030142  -1.372   0.1706
## lot_LOG:reg   0.001515   0.055990   0.027   0.9784
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2095 on 524 degrees of freedom
## Multiple R-squared:  0.6951, Adjusted R-squared:  0.6829
## F-statistic: 56.89 on 21 and 524 DF,  p-value: < 2.2e-16
```

ten (10) interaction variables introduction, between the log lot size and each one of all other variables.

```
# Model Ramsey's RESET testing.
modelD.RESET <- resettest(modelD, power = 2, type = "fitted", data = data)
print(modelD.RESET)
```

```
##
##  RESET test
##
## data:  modelD
## RESET = 0.011571, df1 = 1, df2 = 523, p-value = 0.9144
```

```
# Model Jarque-Bera testing.
modelD.JB <- jarque.bera.test(modelD.summary$residuals)
print(modelD.JB)
```

```
##
##  Jarque Bera Test
##
## data:  modelD.summary$residuals
## X-squared = 8.2029, df = 2, p-value = 0.01655
```

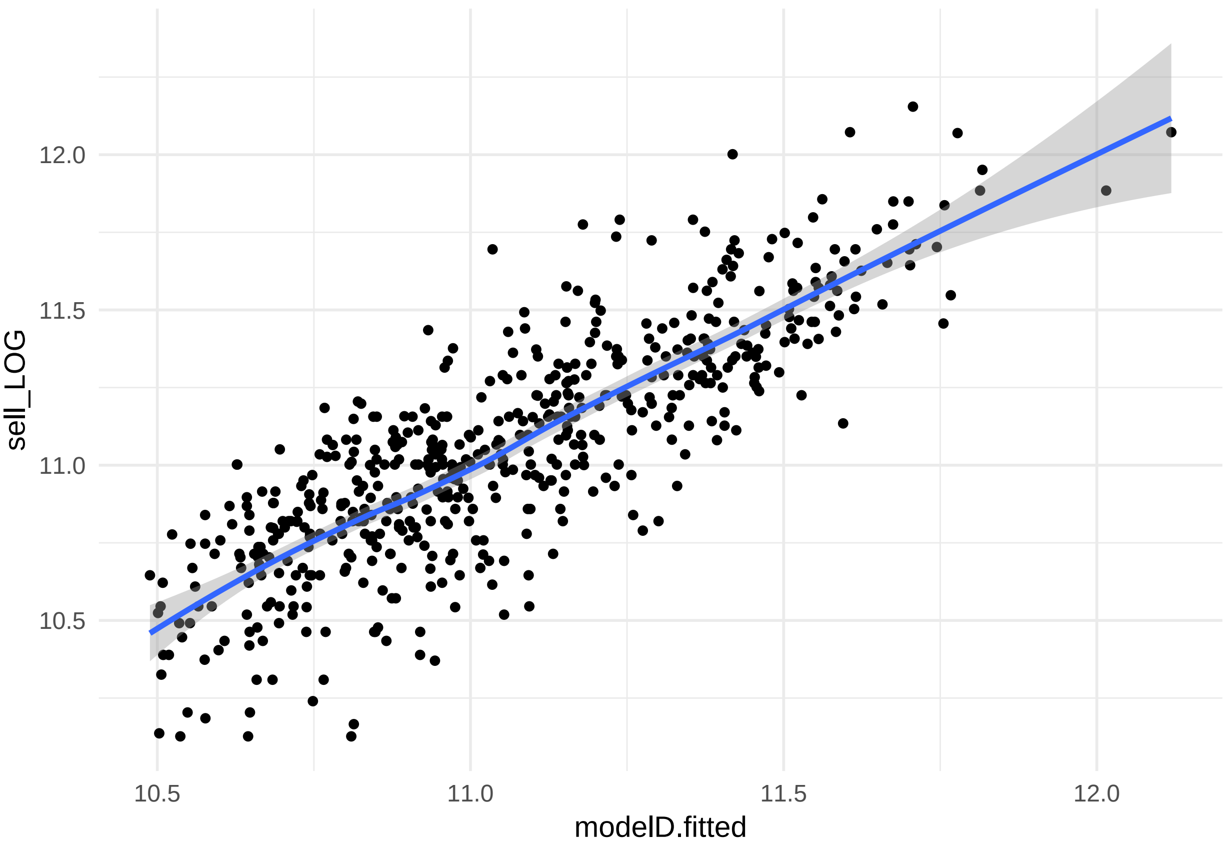
With a statistic of ~8.203 and a p-value of ~0.0165, the Jarque-Bera test suggests that the model residuals are still NOT normally distributed; therefore the model is still NOT correctly specified.

This Jarque-Bera test result, however, is the best scored so far. It seems that the interaction variables introduction slightly improves the (previous best) second model residuals normality.

```
modelD.fitted <- fitted.values(modelD)

ggplot(data, aes(x=modelD.fitted, y=sell_LOG)) +
    geom_point(shape=16) +
    geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Using the 5% significance level, only two (2) of the ten (10) interaction variables used are individually significant:

LOG(lot)-drv LOG(lot)-rec

(e) Perform an F-test for the joint significance of the interaction effects from question (d).

```
## All Variables:
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec +
##     ffin + ghw + ca + gar + reg + lot_LOG * bdms + lot_LOG *
##     fb + lot_LOG * sty + lot_LOG * drv + lot_LOG * rec + lot_LOG *
##     ffin + lot_LOG * ghw + lot_LOG * ca + lot_LOG * gar + lot_LOG *
##     reg, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68306 -0.11612  0.00591  0.12486  0.65998
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   8.966499   1.070667   8.375 5.09e-16 ***
## lot_LOG       0.152685   0.128294   1.190   0.2345
## bdms          0.019075   0.326700   0.058   0.9535
## fb           -0.368234   0.429048  -0.858   0.3911
## sty           0.488885   0.309700   1.579   0.1150  
## drv          -1.463371   0.717225  -2.040   0.0418 *
## rec           1.673992   0.655919   2.552   0.0110 *
## ffin         -0.031844   0.445543  -0.071   0.9430  
## ghw          -0.505889   0.902733  -0.560   0.5754  
## ca           -0.340276   0.496041  -0.686   0.4930  
## gar           0.401941   0.258646   1.554   0.1208  
## reg           0.118484   0.479856   0.247   0.8051  
## lot_LOG:bdms  0.002070   0.038654   0.054   0.9573  
## lot_LOG:fb    0.062037   0.050145   1.237   0.2166  
## lot_LOG:sty  -0.046361   0.035942  -1.290   0.1977  
## lot_LOG:drv   0.191542   0.087361   2.193   0.0288 *
## lot_LOG:rec  -0.188462   0.076373  -2.468   0.0139 *
## lot_LOG:ffin  0.015913   0.052851   0.301   0.7635  
## lot_LOG:ghw   0.081135   0.106929   0.759   0.4483  
## lot_LOG:ca    0.059549   0.058024   1.026   0.3052  
## lot_LOG:gar  -0.041359   0.030142  -1.372   0.1706  
## lot_LOG:reg   0.001515   0.055990   0.027   0.9784  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.2095 on 524 degrees of freedom
## Multiple R-squared:  0.6951, Adjusted R-squared:  0.6829 
## F-statistic: 56.89 on 21 and 524 DF,  p-value: < 2.2e-16
```

```
## LOG(lot)-reg variable removed:
```

```
## 
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec + 
##     ffin + ghw + ca + gar + reg + lot_LOG * bdms + lot_LOG * 
##     fb + lot_LOG * sty + lot_LOG * drv + lot_LOG * rec + lot_LOG * 
##     ffin + lot_LOG * ghw + lot_LOG * ca + lot_LOG * gar, data = data)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.68292 -0.11619  0.00573  0.12491  0.65976 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)   8.96795    1.06831   8.394 4.37e-16 ***
## lot_LOG       0.15252    0.12802   1.191   0.2341    
## bdms          0.01949    0.32603   0.060   0.9523    
## fb           -0.36774    0.42824  -0.859   0.3909    
## sty           0.48721    0.30316   1.607   0.1086    
## drv          -1.46786    0.69713  -2.106   0.0357 *  
## rec           1.67468    0.65480   2.558   0.0108 *  
## ffin         -0.03494    0.43021  -0.081   0.9353    
## ghw          -0.50427    0.89990  -0.560   0.5755    
## ca           -0.33954    0.49483  -0.686   0.4929    
## gar           0.40234    0.25797   1.560   0.1194    
## reg           0.13145    0.02304   5.705 1.94e-08 ***
## lot_LOG:bdms  0.00202    0.03857   0.052   0.9582    
## lot_LOG:fb    0.06198    0.05005   1.238   0.2161    
## lot_LOG:sty  -0.04617    0.03518  -1.312   0.1900    
## lot_LOG:drv   0.19207    0.08504   2.259   0.0243 *
## lot_LOG:rec  -0.18855    0.07623  -2.473   0.0137 *
## lot_LOG:ffin  0.01629    0.05098   0.319   0.7495
## lot_LOG:ghw   0.08094    0.10658   0.759   0.4479
## lot_LOG:ca    0.05946    0.05788   1.027   0.3047
## lot_LOG:gar  -0.04140    0.03007  -1.377   0.1691
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2093 on 525 degrees of freedom
## Multiple R-squared:  0.6951, Adjusted R-squared:  0.6835
## F-statistic: 59.85 on 20 and 525 DF,  p-value: < 2.2e-16
```

```
## LOG(lot)-bdms variable removed:
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec +
##     ffin + ghw + ca + gar + reg + lot_LOG * fb + lot_LOG * sty +
##     lot_LOG * drv + lot_LOG * rec + lot_LOG * ffin + lot_LOG *
##     ghw + lot_LOG * ca + lot_LOG * gar, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68301 -0.11617  0.00574  0.12490  0.66020
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   8.93486    0.86069  10.381  < 2e-16 ***
## lot_LOG       0.15647    0.10329   1.515   0.1304
## bdms          0.03655    0.01459   2.506   0.0125 *
## fb           -0.37745    0.38563  -0.979   0.3281
## sty           0.48200    0.28614   1.685   0.0927 .
## drv          -1.46253    0.68903  -2.123   0.0343 *
## rec           1.67592    0.65375   2.564   0.0106 *
## ffin         -0.03743    0.42717  -0.088   0.9302
## ghw          -0.50161    0.89761  -0.559   0.5765
## ca           -0.33869    0.49409  -0.685   0.4933
## gar           0.40103    0.25652   1.563   0.1186
## reg           0.13144    0.02302   5.710 1.89e-08 ***
## lot_LOG:fb    0.06313    0.04494   1.405   0.1607
## lot_LOG:sty  -0.04556    0.03321  -1.372   0.1706
## lot_LOG:drv   0.19143    0.08408   2.277   0.0232 *
## lot_LOG:rec  -0.18868    0.07612  -2.479   0.0135 *
## lot_LOG:ffin  0.01658    0.05062   0.328   0.7434
## lot_LOG:ghw   0.08062    0.10631   0.758   0.4486
## lot_LOG:ca    0.05935    0.05779   1.027   0.3049
## lot_LOG:gar  -0.04125    0.02989  -1.380   0.1682
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2091 on 526 degrees of freedom
## Multiple R-squared:  0.6951, Adjusted R-squared:  0.6841
## F-statistic: 63.12 on 19 and 526 DF,  p-value: < 2.2e-16
## LOG(lot)-ffin variable removed:
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec +
##     ffin + ghw + ca + gar + reg + lot_LOG * fb + lot_LOG * sty +
##     lot_LOG * drv + lot_LOG * rec + lot_LOG * ghw + lot_LOG *
##     ca + lot_LOG * gar, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68181 -0.11724  0.00567  0.12594  0.65662
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.87651    0.84134  10.550  < 2e-16 ***
## lot_LOG      0.16359    0.10089   1.621   0.1055
## bdms         0.03655    0.01458   2.507   0.0125 *
## fb          -0.38191    0.38506  -0.992   0.3217
## sty          0.48851    0.28520   1.713   0.0873 .
## drv         -1.45022    0.68742  -2.110   0.0354 *
## rec          1.62140    0.63167   2.567   0.0105 *
## ffin         0.10232    0.02181   4.691 3.47e-06 ***
## ghw         -0.51600    0.89578  -0.576   0.5648
## ca          -0.35449    0.49131  -0.722   0.4709
## gar          0.40146    0.25629   1.566   0.1179
## reg          0.13227    0.02286   5.786 1.24e-08 ***
## lot_LOG:fb   0.06360    0.04487   1.417   0.1570
## lot_LOG:sty -0.04640    0.03308  -1.403   0.1613
## lot_LOG:drv  0.18991    0.08388   2.264   0.0240 *
## lot_LOG:rec -0.18218    0.07343  -2.481   0.0134 *
## lot_LOG:ghw  0.08250    0.10606   0.778   0.4370
## lot_LOG:ca   0.06123    0.05746   1.066   0.2871
## lot_LOG:gar -0.04129    0.02987  -1.383   0.1674
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2089 on 527 degrees of freedom
## Multiple R-squared:  0.695,  Adjusted R-squared:  0.6846
## F-statistic: 66.73 on 18 and 527 DF,  p-value: < 2.2e-16
```

```
## LOG(lot)-ghw variable removed:
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec +
##     ffin + ghw + ca + gar + reg + lot_LOG * fb + lot_LOG * sty +
##     lot_LOG * drv + lot_LOG * rec + lot_LOG * ca + lot_LOG *
##     gar, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68122 -0.11898  0.00738  0.12611  0.65311
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.80857    0.83648  10.530  < 2e-16 ***
## lot_LOG      0.17159    0.10033   1.710   0.0878 .
## bdms         0.03661    0.01457   2.513   0.0123 *
## fb          -0.37636    0.38485  -0.978   0.3286
## sty          0.49092    0.28508   1.722   0.0857 .
## drv         -1.43262    0.68679  -2.086   0.0375 *
## rec          1.63058    0.63133   2.583   0.0101 *
## ffin         0.10361    0.02174   4.766 2.44e-06 ***
## ghw          0.17991    0.04391   4.098 4.83e-05 ***
## ca          -0.33972    0.49076  -0.692   0.4891
## gar          0.39730    0.25614   1.551   0.1215
## reg          0.13113    0.02281   5.750 1.51e-08 ***
## lot_LOG:fb   0.06302    0.04485   1.405   0.1606
## lot_LOG:sty -0.04669    0.03306  -1.412   0.1585
## lot_LOG:drv  0.18782    0.08380   2.241   0.0254 *
## lot_LOG:rec -0.18320    0.07339  -2.496   0.0129 *
## lot_LOG:ca   0.05932    0.05738   1.034   0.3017
## lot_LOG:gar -0.04085    0.02985  -1.368   0.1718
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2088 on 528 degrees of freedom
## Multiple R-squared:  0.6947, Adjusted R-squared:  0.6849
## F-statistic: 70.67 on 17 and 528 DF,  p-value: < 2.2e-16
```

```
## LOG(lot)-ca variable removed:
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec +
##     ffin + ghw + ca + gar + reg + lot_LOG * fb + lot_LOG * sty +
##     lot_LOG * drv + lot_LOG * rec + lot_LOG * gar, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.67934 -0.12004  0.00644  0.12660  0.64601
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.78218    0.83615  10.503  < 2e-16 ***
## lot_LOG      0.17484    0.10028   1.743   0.0818 .
## bdms         0.03523    0.01451   2.428   0.0155 *
## fb          -0.38030    0.38485  -0.988   0.3235
## sty          0.47196    0.28451   1.659   0.0977 .
## drv         -1.42861    0.68683  -2.080   0.0380 *
## rec          1.55669    0.62731   2.482   0.0134 *
## ffin         0.10341    0.02174   4.756 2.55e-06 ***
## ghw          0.17721    0.04383   4.043 6.06e-05 ***
## ca           0.16716    0.02121   7.880 1.87e-14 ***
## gar          0.33850    0.24976   1.355   0.1759
## reg          0.13298    0.02274   5.848 8.69e-09 ***
## lot_LOG:fb   0.06359    0.04485   1.418   0.1569
## lot_LOG:sty -0.04432    0.03299  -1.344   0.1797
## lot_LOG:drv  0.18733    0.08381   2.235   0.0258 *
## lot_LOG:rec -0.17463    0.07293  -2.395   0.0170 *
## lot_LOG:gar -0.03385    0.02908  -1.164   0.2448
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2088 on 529 degrees of freedom
## Multiple R-squared:  0.6941, Adjusted R-squared:  0.6848
## F-statistic: 75.01 on 16 and 529 DF,  p-value: < 2.2e-16
```

```
## LOG(lot)-gar variable removed:
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec +
##     ffin + ghw + ca + gar + reg + lot_LOG * fb + lot_LOG * sty +
##     lot_LOG * drv + lot_LOG * rec, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68420 -0.12071  0.00669  0.12322  0.64513
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.77393    0.83640  10.490  < 2e-16 ***
## lot_LOG      0.17584    0.10031   1.753   0.0802 .
## bdms         0.03530    0.01451   2.432   0.0153 *
## fb          -0.34021    0.38344  -0.887   0.3753
## sty          0.46819    0.28459   1.645   0.1005
## drv         -1.23688    0.66702  -1.854   0.0642 .
## rec          1.51405    0.62645   2.417   0.0160 *
## ffin         0.10279    0.02174   4.727 2.92e-06 ***
## ghw          0.18002    0.04378   4.112 4.55e-05 ***
## ca           0.16697    0.02122   7.869 2.02e-14 ***
## gar          0.04802    0.01143   4.200 3.13e-05 ***
## reg          0.12990    0.02259   5.750 1.51e-08 ***
## lot_LOG:fb   0.05903    0.04469   1.321   0.1872
## lot_LOG:sty -0.04392    0.03300  -1.331   0.1837
## lot_LOG:drv  0.16448    0.08150   2.018   0.0441 *
## lot_LOG:rec -0.16943    0.07281  -2.327   0.0203 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2089 on 530 degrees of freedom
## Multiple R-squared:  0.6933, Adjusted R-squared:  0.6846
## F-statistic: 79.87 on 15 and 530 DF,  p-value: < 2.2e-16
```

```
## LOG(lot)-fb variable removed:
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec +
##     ffin + ghw + ca + gar + reg + lot_LOG * sty + lot_LOG * drv +
##     lot_LOG * rec, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68209 -0.11831  0.00758  0.12350  0.63856
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.29846    0.75549  10.984  < 2e-16 ***
## lot_LOG      0.23088    0.09131   2.529   0.0117 *
## bdms         0.03623    0.01451   2.497   0.0128 *
## fb           0.16549    0.02061   8.030 6.30e-15 ***
## sty          0.38420    0.27758   1.384   0.1669
## drv         -1.25462    0.66735  -1.880   0.0607 .
## rec          1.47254    0.62610   2.352   0.0190 *
## ffin         0.10042    0.02168   4.631 4.58e-06 ***
## ghw          0.18093    0.04381   4.130 4.21e-05 ***
## ca           0.16623    0.02123   7.831 2.64e-14 ***
## gar          0.04751    0.01143   4.155 3.79e-05 ***
## reg          0.13126    0.02258   5.812 1.06e-08 ***
## lot_LOG:sty -0.03402    0.03216  -1.058   0.2906
## lot_LOG:drv  0.16690    0.08154   2.047   0.0412 *
## lot_LOG:rec -0.16467    0.07278  -2.263   0.0241 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2091 on 531 degrees of freedom
## Multiple R-squared:  0.6923, Adjusted R-squared:  0.6842
## F-statistic: 85.33 on 14 and 531 DF,  p-value: < 2.2e-16
```

```
## LOG(lot)-sty variable removed:
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec +
##     ffin + ghw + ca + gar + reg + lot_LOG * drv + lot_LOG * rec,
##     data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.67934 -0.12225  0.00849  0.12259  0.65051
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.74189    0.62863  13.906  < 2e-16 ***
## lot_LOG      0.17906    0.07707   2.323  0.02053 *
## bdms         0.03881    0.01430   2.714  0.00686 **
## fb           0.16145    0.02025   7.971 9.62e-15 ***
## sty          0.09083    0.01254   7.242 1.56e-12 ***
## drv         -1.18996    0.66462  -1.790  0.07395 .
## rec          1.50253    0.62553   2.402  0.01665 *
## ffin         0.10276    0.02157   4.763 2.46e-06 ***
## ghw          0.18448    0.04368   4.223 2.83e-05 ***
## ca           0.16526    0.02121   7.792 3.48e-14 ***
## gar          0.04690    0.01142   4.107 4.65e-05 ***
## reg          0.13260    0.02255   5.880 7.24e-09 ***
## lot_LOG:drv  0.15943    0.08124   1.962  0.05024 .
## lot_LOG:rec -0.16826    0.07270  -2.314  0.02103 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2091 on 532 degrees of freedom
## Multiple R-squared:  0.6916, Adjusted R-squared:  0.6841
## F-statistic: 91.79 on 13 and 532 DF,  p-value: < 2.2e-16
```

```
## LOG(lot)-drv variable removed:
```

```
##
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec +
##     ffin + ghw + ca + gar + reg + lot_LOG * rec, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68111 -0.12208  0.00593  0.12731  0.66275
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.59071    0.22656  33.505  < 2e-16 ***
## lot_LOG      0.32024    0.02770  11.562  < 2e-16 ***
## bdms         0.03842    0.01434   2.680   0.0076 **
## fb           0.16318    0.02029   8.043 5.71e-15 ***
## sty          0.09080    0.01258   7.220 1.80e-12 ***
## drv          0.11312    0.02815   4.018 6.72e-05 ***
## rec          1.44313    0.62646   2.304   0.0216 *
## ffin         0.10450    0.02161   4.835 1.74e-06 ***
## ghw          0.18429    0.04380   4.208 3.03e-05 ***
## ca           0.16593    0.02126   7.804 3.19e-14 ***
## gar          0.04810    0.01144   4.206 3.05e-05 ***
## reg          0.13373    0.02260   5.917 5.89e-09 ***
## lot_LOG:rec -0.16112    0.07281  -2.213   0.0273 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2096 on 533 degrees of freedom
## Multiple R-squared:  0.6894, Adjusted R-squared:  0.6824
## F-statistic: 98.59 on 12 and 533 DF,  p-value: < 2.2e-16
```

Started with the most general model, including as many variables as are at hand. Then, checked whether one or more variables can be removed from the model. This can be based on individual t-tests, or a joint F-test in case of multiple variables. In case you remove one variable at a time, the variable with the lowest absolute t-value is removed from the model. The model is estimated again without that variable, and the procedure is repeated. The procedure continues until all remaining variables are significant.

Variables elimination after regression, one at a time, produced the following results:

After regression round #1: LOG(lot)-reg interaction variable was chosen to be removed.

After regression round #2: LOG(lot)-bdms interaction variable was chosen to be removed.

After regression round #3: LOG(lot)-ffin interaction variable was chosen to be removed.

After regression round #4: LOG(lot)-ghw interaction variable was chosen to be removed.

After regression round #5: LOG(lot)-ca interaction variable was chosen to be removed.

After regression round #6: LOG(lot)-gar interaction variable was chosen to be removed.

After regression round #7: LOG(lot)-fb interaction variable was chosen to be removed.

After regression round #8: LOG(lot)-sty interaction variable was chosen to be removed.

After regression round #9: LOG(lot)-drv interaction variable was chosen to be removed.

After regression round #10: all remaining variables were found to be significant; variables removal stops here. Conclusively, the only interaction variable found to be significant is LOG(lot)-rec.

```
modelF <- lm(sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec + ffin + ghw + ca + gar + reg +
                 lot_LOG * rec,
             data = data)
print(modelF.summary<-summary(modelF))
```

```
## 
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec + 
##     ffin + ghw + ca + gar + reg + lot_LOG * rec, data = data)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.68111 -0.12208  0.00593  0.12731  0.66275 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  7.59071    0.22656  33.505  < 2e-16 ***
## lot_LOG      0.32024    0.02770  11.562  < 2e-16 ***
## bdms         0.03842    0.01434   2.680   0.0076 ** 
## fb           0.16318    0.02029   8.043 5.71e-15 ***
## sty          0.09080    0.01258   7.220 1.80e-12 ***
## drv          0.11312    0.02815   4.018 6.72e-05 ***
## rec          1.44313    0.62646   2.304   0.0216 *  
## ffin         0.10450    0.02161   4.835 1.74e-06 ***
## ghw          0.18429    0.04380   4.208 3.03e-05 ***
## ca           0.16593    0.02126   7.804 3.19e-14 ***
## gar          0.04810    0.01144   4.206 3.05e-05 ***
## reg          0.13373    0.02260   5.917 5.89e-09 ***
## lot_LOG:rec -0.16112    0.07281  -2.213   0.0273 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.2096 on 533 degrees of freedom
## Multiple R-squared:  0.6894, Adjusted R-squared:  0.6824 
## F-statistic: 98.59 on 12 and 533 DF,  p-value: < 2.2e-16
```

```
# Model Ramsey's RESET testing.
modelF.RESET <- resettest(modelF, power = 2, type = "fitted",
                          data = data)
print(modelF.RESET)
```

```
## 
##  RESET test
## 
## data:  modelF
## RESET = 0.43102, df1 = 1, df2 = 532, p-value = 0.5118
```

```
# Model Jarque-Bera testing.
modelF.JB <- jarque.bera.test(modelF.summary$residuals)
print(modelF.JB)
```

```
##
##  Jarque Bera Test
##
## data:  modelF.summary$residuals
## X-squared = 10.348, df = 2, p-value = 0.005661
```

Conclusion:

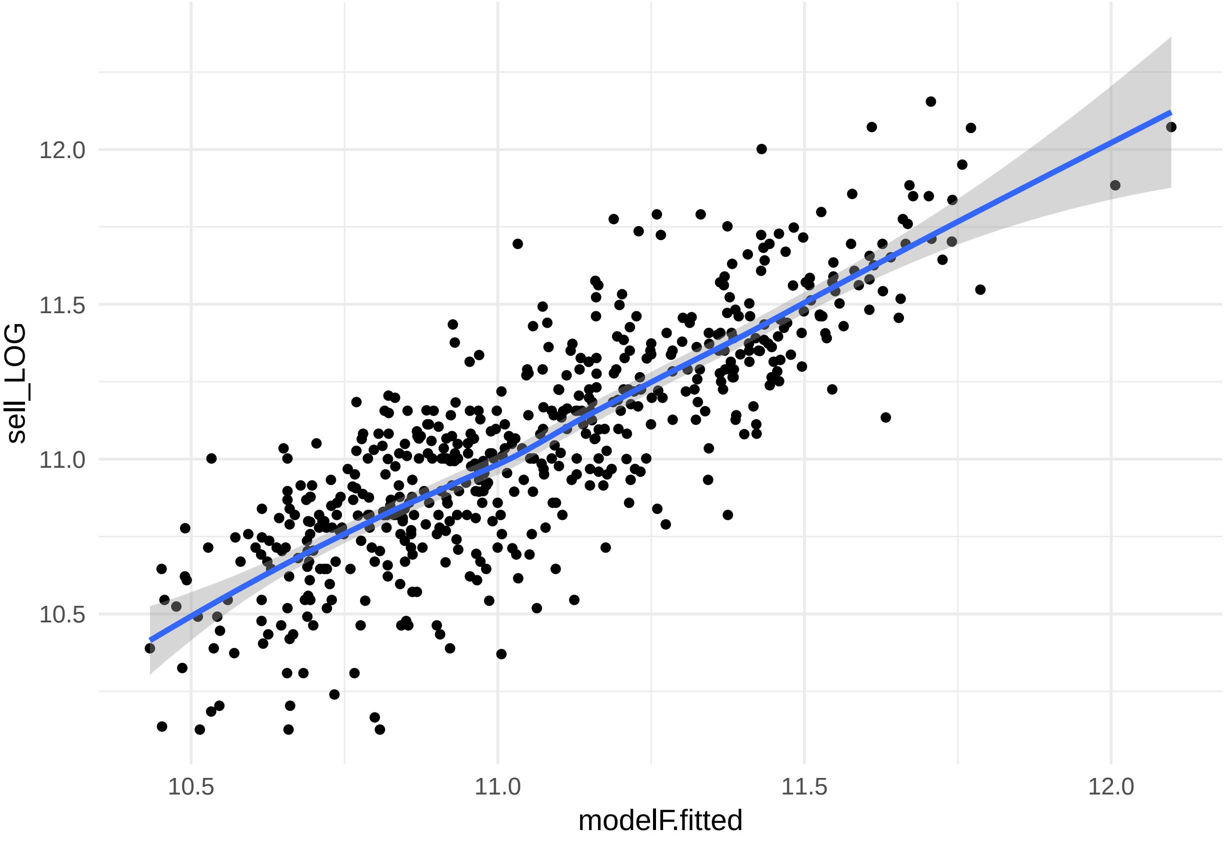
With a statistic of ~10.348 and a p-value of ~0.0057, the Jarque-Bera test suggests that the model residuals are still NOT normally distributed; therefore the model is still NOT correctly specified.

This Jarque-Bera test result is not the best scored so far. All the previous models (except from the first) related test had indicated an even better residuals normality.

```
modelF.fitted <- fitted.values(modelF)

ggplot(data, aes(x=modelF.fitted, y=sell_LOG)) +
    geom_point(shape=16) +
    geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



(g) One may argue that some of the explanatory variables are endogenous and that there may be omitted variables. For example, the 'condition' of the house in terms of how it is maintained is not a variable (and difficult to measure) but will affect the house price. It will also affect, or be reflected in, some of the other variables, such as whether the house has an air conditioning (which is mostly in newer houses). If the condition of the house is missing, will the effect of air conditioning on the (log of the) sale price be over- or underestimated? (For this question no computer calculations are required.)

The effect of the air conditioning ca variable on the logarithm of the sale price LOG(sell) variable will be overestimated, because it is usually affected by the age (and therefore the condition) of houses both of which (logically) affect the house selling price positively.

So, the effect of the age and condition house properties (which are not available to our models as variables) is partially included in the air conditioning ca variable. And since that effect is expected to be positive on the house sale price (and its logarithm), it will increase the effect of the air-conditioning ca variable in our models (thus, its estimated effect is overestimated).

Finally we analyze the predictive ability of the model. Consider again the model where the log of the sale price of the house is the dependent variable and the explanatory variables are the log transformation of lot size, with all other explanatory variables in their original form (and no interaction effects). Estimate the parameters of the model using the first 400 observations. Make predictions on the log of the price and calculate the MAE for the other 146 observations. How good is the predictive power of the model (relative to the variability in the log of the price)?

```r
# Separating the data sample in two groups
data1 <- data[which(data$obs <= 400), ]
n1 <- nrow(data1)
print(paste("Data group#1 has", n1, "entries."))
```

```
## [1] "Data group#1 has 400 entries."
```

```r
summary(data1)
```

```
##       obs             sell             lot              bdms
##  Min.   :  1.0   Min.   : 25000   Min.   : 1650   Min.   :1.00
##  1st Qu.:100.8   1st Qu.: 46150   1st Qu.: 3495   1st Qu.:2.00
##  Median :200.5   Median : 59250   Median : 4180   Median :3.00
##  Mean   :200.5   Mean   : 64977   Mean   : 4905   Mean   :2.95
##  3rd Qu.:300.2   3rd Qu.: 78000   3rd Qu.: 6000   3rd Qu.:3.00
##  Max.   :400.0   Max.   :190000   Max.   :16200   Max.   :6.00
##        fb             sty             drv              rec
##  Min.   :1.000   Min.   :1.000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:1.000   1st Qu.:1.000   1st Qu.:1.0000   1st Qu.:0.0000
##  Median :1.000   Median :2.000   Median :1.0000   Median :0.0000
##  Mean   :1.278   Mean   :1.718   Mean   :0.8125   Mean   :0.1625
##  3rd Qu.:1.000   3rd Qu.:2.000   3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :4.000   Max.   :4.000   Max.   :1.0000   Max.   :1.0000
##       ffin             ghw             ca              gar
##  Min.   :0.0000   Min.   :0.00   Min.   :0.000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.00   1st Qu.:0.000   1st Qu.:0.0000
##  Median :0.0000   Median :0.00   Median :0.000   Median :0.0000
##  Mean   :0.3475   Mean   :0.05   Mean   :0.285   Mean   :0.6925
##  3rd Qu.:1.0000   3rd Qu.:0.00   3rd Qu.:1.000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :1.00   Max.   :1.000   Max.   :3.0000
##       reg           sell_LOG        lot_LOG
##  Min.   :0.000   Min.   :10.13   Min.   :7.409
##  1st Qu.:0.000   1st Qu.:10.74   1st Qu.:8.159
##  Median :0.000   Median :10.99   Median :8.338
##  Mean   :0.105   Mean   :11.01   Mean   :8.420
##  3rd Qu.:0.000   3rd Qu.:11.26   3rd Qu.:8.700
##  Max.   :1.000   Max.   :12.15   Max.   :9.693
```

```
data2 <- data[which(data$obs > 400), ]
n2 <- nrow(data2)
print(paste("Data group#2 has", n2, "entries."))
```

```
## [1] "Data group#2 has 146 entries."
```

```
summary(data2)
```

```
##       obs              sell              lot              bdms      
##  Min.   :401.0   Min.   : 31900   Min.   : 1950   Min.   :2.000  
##  1st Qu.:437.2   1st Qu.: 60000   1st Qu.: 4678   1st Qu.:3.000  
##  Median :473.5   Median : 72750   Median : 6000   Median :3.000  
##  Mean   :473.5   Mean   : 76737   Mean   : 5821   Mean   :3.007  
##  3rd Qu.:509.8   3rd Qu.: 91125   3rd Qu.: 6652   3rd Qu.:3.000  
##  Max.   :546.0   Max.   :174500   Max.   :12944   Max.   :5.000  
##        fb             sty             drv              rec        
##  Min.   :1.000   Min.   :1.000   Min.   :0.0000   Min.   :0.0000  
##  1st Qu.:1.000   1st Qu.:1.000   1st Qu.:1.0000   1st Qu.:0.0000  
##  Median :1.000   Median :2.000   Median :1.0000   Median :0.0000  
##  Mean   :1.308   Mean   :2.055   Mean   :0.9863   Mean   :0.2192  
##  3rd Qu.:2.000   3rd Qu.:3.000   3rd Qu.:1.0000   3rd Qu.:0.0000  
##  Max.   :2.000   Max.   :4.000   Max.   :1.0000   Max.   :1.0000  
##       ffin             ghw                ca              gar        
##  Min.   :0.0000   Min.   :0.00000   Min.   :0.0000   Min.   :0.0000  
##  1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.0000  
##  Median :0.0000   Median :0.00000   Median :0.0000   Median :0.0000  
##  Mean   :0.3562   Mean   :0.03425   Mean   :0.4041   Mean   :0.6918  
##  3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.:1.0000  
##  Max.   :1.0000   Max.   :1.00000   Max.   :1.0000   Max.   :3.0000  
##       reg           sell_LOG        lot_LOG     
##  Min.   :0.000   Min.   :10.37   Min.   :7.576  
##  1st Qu.:0.000   1st Qu.:11.00   1st Qu.:8.451  
##  Median :1.000   Median :11.19   Median :8.700  
##  Mean   :0.589   Mean   :11.21   Mean   :8.595  
##  3rd Qu.:1.000   3rd Qu.:11.42   3rd Qu.:8.803  
##  Max.   :1.000   Max.   :12.07   Max.   :9.468
```

```
# Estimating third model.
modelH <- lm(sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec + ffin + ghw + ca + gar + reg,
             data = data1)
print(modelH.summary <- summary(modelH))
```

```
## 
## Call:
## lm(formula = sell_LOG ~ lot_LOG + bdms + fb + sty + drv + rec + 
##     ffin + ghw + ca + gar + reg, data = data1)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.66582 -0.13906  0.00796  0.14694  0.67596
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.67309    0.29240  26.241  < 2e-16 ***
## lot_LOG      0.31378    0.03615   8.680  < 2e-16 ***
## bdms         0.03787    0.01744   2.172 0.030469 *
## fb           0.15238    0.02469   6.170 1.71e-09 ***
## sty          0.08824    0.01819   4.850 1.79e-06 ***
## drv          0.08641    0.03141   2.751 0.006216 **
## rec          0.05465    0.03392   1.611 0.107975
## ffin         0.11471    0.02673   4.291 2.25e-05 ***
## ghw          0.19870    0.05301   3.748 0.000205 ***
## ca           0.17763    0.02724   6.521 2.17e-10 ***
## gar          0.05301    0.01480   3.583 0.000383 ***
## reg          0.15116    0.04215   3.586 0.000378 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2238 on 388 degrees of freedom
## Multiple R-squared:  0.6705, Adjusted R-squared:  0.6611
## F-statistic: 71.77 on 11 and 388 DF,  p-value: < 2.2e-16
```

```
# Model Ramsey's RESET testing.
modelH.RESET <- resettest(modelH, power = 2, type = "fitted",
                          data = data1)
print(modelH.RESET)
```

```
##
##  RESET test
##
## data:  modelH
## RESET = 0.03955, df1 = 1, df2 = 387, p-value = 0.8425
```

```
# Model Jarque-Bera testing.
modelH.JB <- jarque.bera.test(modelH.summary$residuals)
print(modelH.JB)
```

```
##
##  Jarque Bera Test
##
## data:  modelH.summary$residuals
## X-squared = 0.69757, df = 2, p-value = 0.7055
```

With a statistic of ~0.698 and a p-value of ~0.7055, the Jarque-Bera test suggests that the model residuals are normally distributed; therefore the model is considered correctly specified.

This Jarque-Bera test result is the best scored so far, and indicates a sufficient residuals normality.

Conclusion

Both Ramsey's RESET and Jarque-Bera tests suggest that the seventh model is sufficiently linear and with good residuals normality.

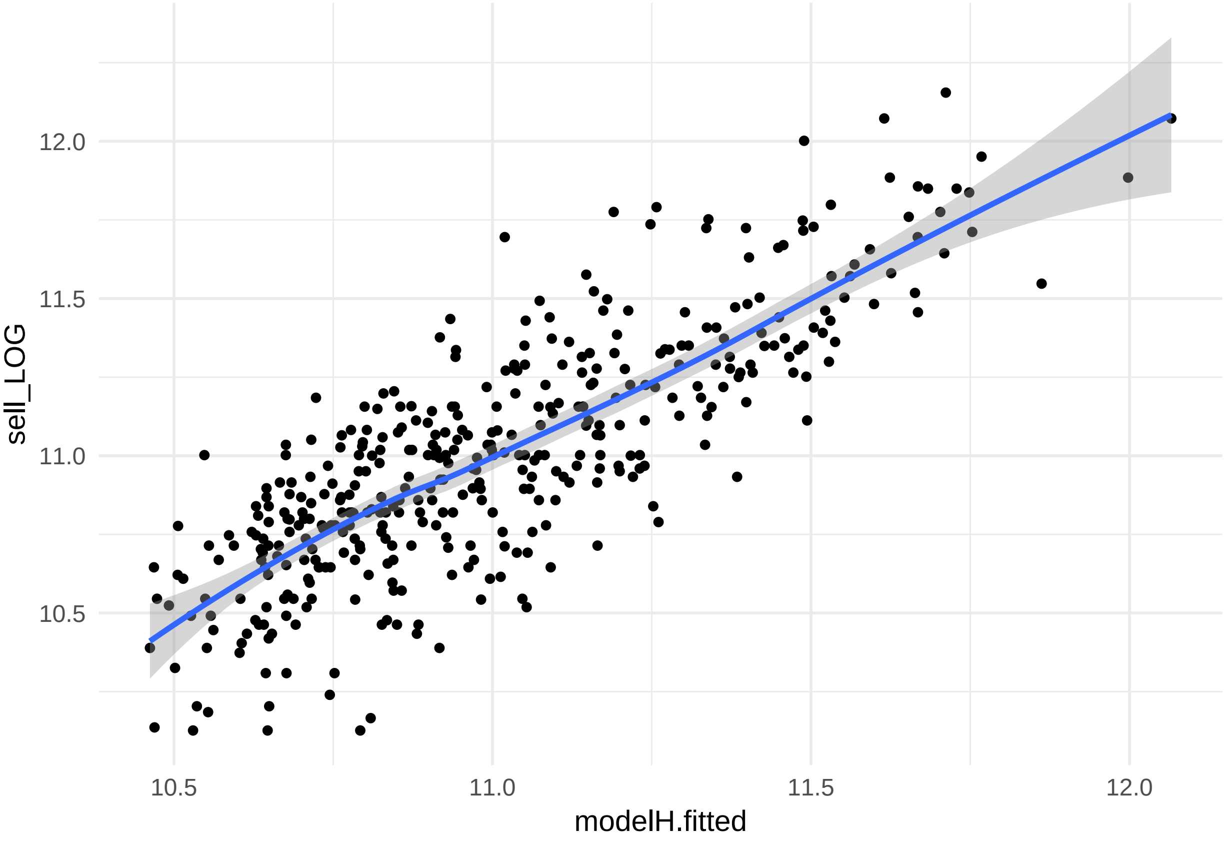
Both Ramsey's RESET and Jarque-Bera tests suggest that the seventh model might be correctly specified.

This is also intuitively demonstrated by the seventh model real to fitted-values diagram shown at the next page (looks about the same or more like a linear relationship).

```
modelH.fitted <- fitted.values(modelH)

ggplot(data1, aes(x=modelH.fitted, y=sell_LOG)) +
    geom_point(shape=16) +
    geom_smooth()
```

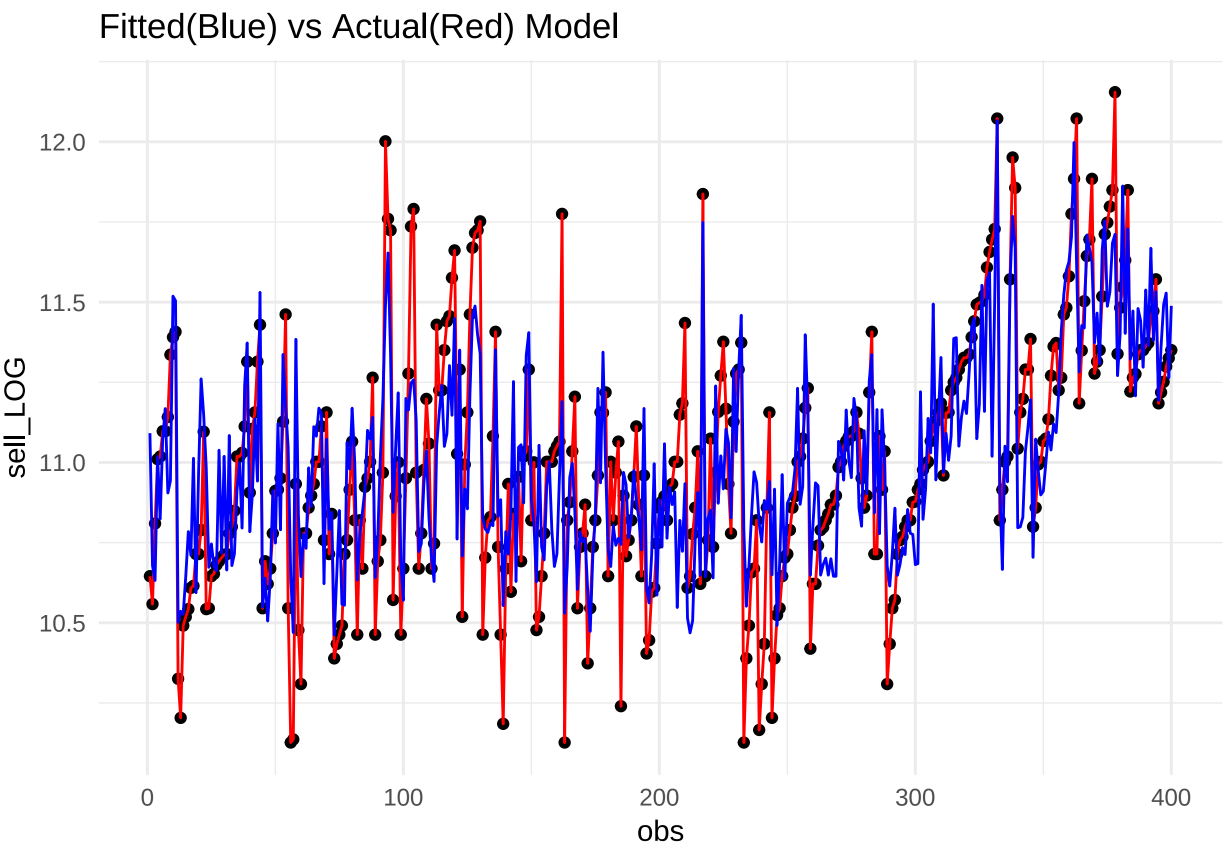
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



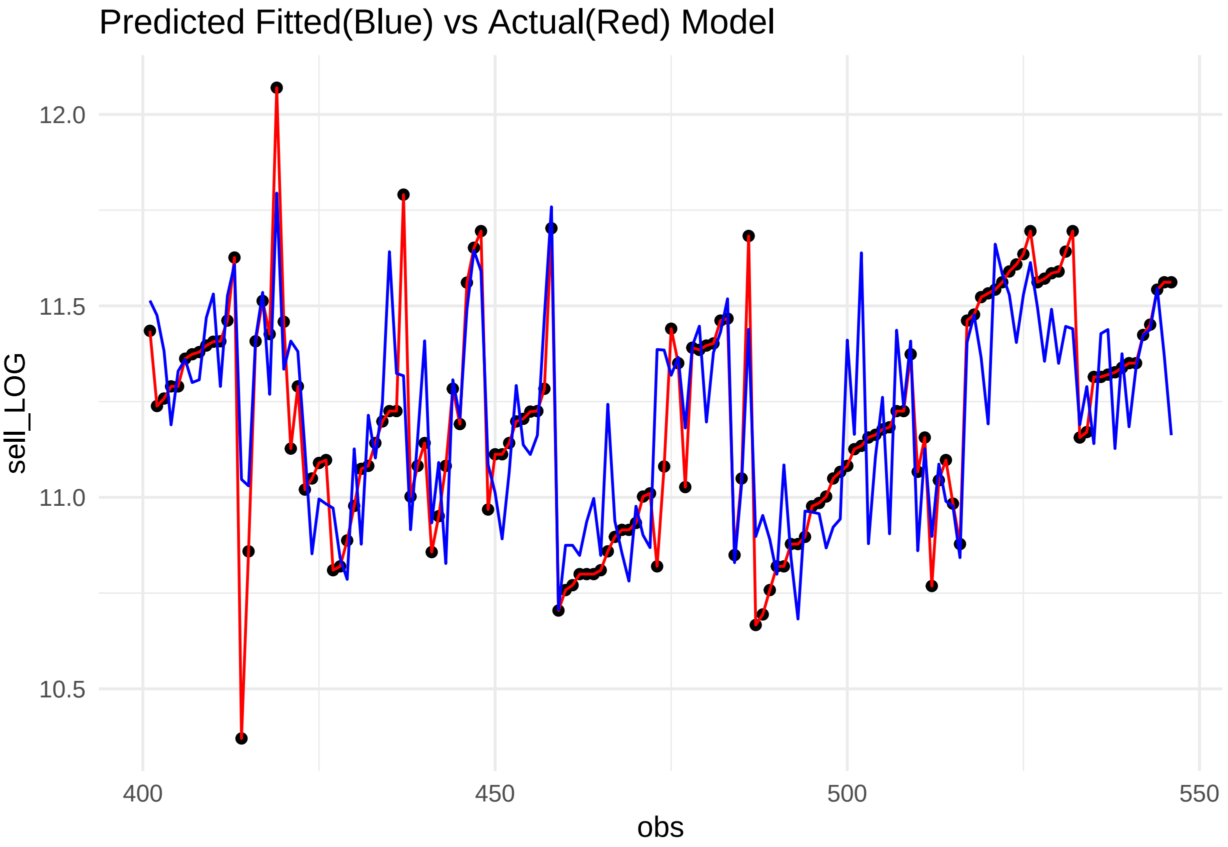
Model predictive ability

The seventh model, as estimated using the first data group, produced the following LOG(sell) values on the second data group:

```
fiited_value <- modelH$fitted.values
data1$Fitted <- fiited_value
data1%>% ggplot(aes(obs, sell_LOG)) + geom_point()+ geom_line(color="red") +
  geom_line(y= data1$Fitted, color= "blue") +ggtitle("Fitted(Blue) vs Actual(Red) Model")
```



```
data2$Fitted <- predict(modelH, data2)
data2%>% ggplot(aes(obs, sell_LOG)) + geom_point()+ geom_line(color="red") +
  geom_line(y= data2$Fitted, color= "blue") +ggtitle("Predicted Fitted(Blue) vs Actual(Red) Model")
```



```
sell_LOG_mean <- mean(data2$sell_LOG)
sell_LOG_sd   <- sd(data2$sell_LOG)

cat(sell_LOG_mean, sell_LOG_sd)
```

```
## 11.20665 0.2887723
```

```
digits <- 3

n <- nrow(data2)
resids_SUM <- sum(abs(data2$Fitted-data2$sell_LOG))
cat("resids_SUM: ", resids_SUM, "\n")
```

```
## resids_SUM:  18.66487
```

```
MAE <- resids_SUM/n
cat("MAE: ", round(MAE, digits))
```

```
## MAE:  0.128
```

The Mean Absolute Error (MAE) value of 0.128 is less than the dependent variable standard deviation itself, which leads to the conclusion that the model has some predictive ability.

Our final model is the only one whose Jarque-Bera test does not reject the null hypothesis of normality of the residuals.

Therefore it is the only model correctly specified.