Case Project – House Prices

Coursera/Erasmus U., Econometric Methods and Applications

Anthony Nguyen

Notes:

- See website for how to submit your answers and how feedback is organized.
- This exercise uses the datafile `CaseProject-HousePrices` and requires a computer.
- The dataset `CaseProject-HousePrices` is available on the website.
- Perform all tests at 5% significance level.

Goals and skills being used:

- Experience the process of variable transformation and model selection.
- Apply tests to evaluate models, including effects of endogeneity.
- Study the predictive ability of a model.

Background

This project is of an applied nature and uses data that are available in the data file Capstone-HousePrices. The source of these data is Anglin and Gencay, "Semiparametric Estimation of a Hedonic Price Function" (Journal of Applied Econometrics 11, 1996, pages 633-648). We consider the modeling and prediction of house prices. Data are available for 546 observations of the following variables:

- `sell`: Sale price of the house
- `lot`: Lot size of the property in square feet
- `bdms`: Number of bedrooms
- `fb`: Number of full bathroom
- `sty`: Number of stories excluding basement
- `drv`: Dummy that is 1 if the house has a driveway and 0 otherwise
- `rec`: Dummy that is 1 if the house has a recreational room and 0 otherwise
- `ffin`: Dummy that is 1 if the house has a full finished basement and 0 otherwise
- `ghw`: Dummy that is 1 if the house uses gas for hot water heating and 0 otherwise

- ca: Dummy that is 1 if there is central air conditioning and 0 otherwise
- gar: Number of covered garage places
- reg: Dummy that is 1 if the house if located in a preferred neighborhood of the city and 0 otherwise
- obs: Observation number, needed in part (h)

Questions

(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?

(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?

(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 of its logarithm?

(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?

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

(f) Now perform model specification on the interaction variables using the general-to-specific approach (Only eliminate the interaction effects).

(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.)

(h) 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)?

Answers

(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?

First we regress `sell` by all of the other variables in the dataset (except for `obs`). The results are as follows:

```
mod1 <- lm(sell~lot+bdms+fb+sty+drv+rec+ffin+ghw+ca+gar+reg,
           data = dat)

summary(mod1)$coefficients
```

```
##                 Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) -4038.350425  3409.4713 -1.184451 2.367616e-01
## lot             3.546303     0.3503 10.123618 3.732442e-22
## bdms         1832.003466  1047.0002  1.749764 8.073341e-02
## fb          14335.558468  1489.9209  9.621691 2.570369e-20
## sty          6556.945711   925.2899  7.086369 4.374046e-12
## drv          6687.778890  2045.2458  3.269914 1.145151e-03
## rec          4511.283826  1899.9577  2.374413 1.792936e-02
## ffin         5452.385539  1588.0239  3.433440 6.422381e-04
## ghw         12831.406266  3217.5971  3.987885 7.595575e-05
## ca          12632.890405  1555.0211  8.123935 3.150681e-15
## gar          4244.829004   840.5442  5.050096 6.069790e-07
## reg          9369.513239  1669.0907  5.613544 3.189602e-08
```

Next, to determine whether the model is better specified in linear form vs. a non-linear/polynomial form, we can perform a RESET test:

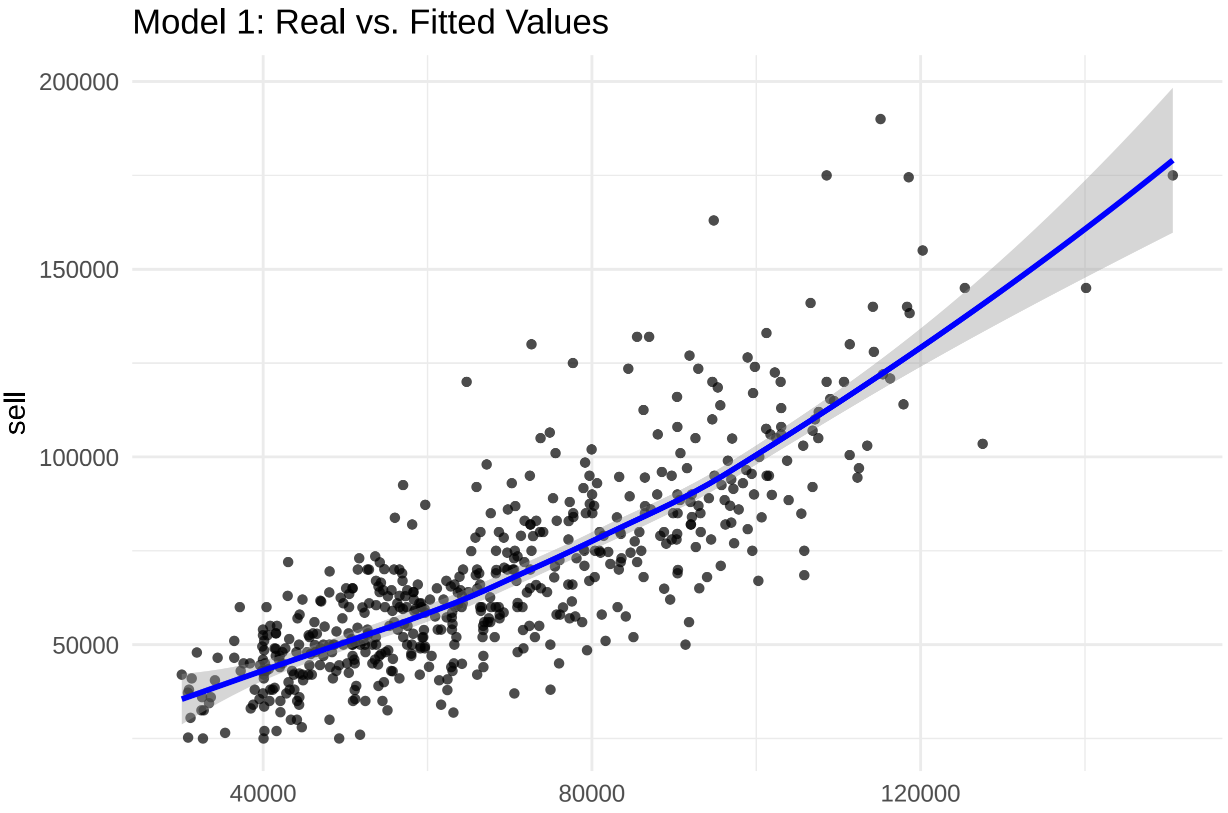
```
# call `resettest()` from `lmtest` package
resettest(mod1, power = 2:3, type = "fitted", data = dat)
```

```
##
##  RESET test
##
## data:  mod1
## RESET = 13.481, df1 = 2, df2 = 532, p-value = 1.944e-06
```

The null hypothesis (H_0) is that the model is linear. Here, we have tested the model for both quadratic and cubic influence of the fitted response, returning a RESET test statistic of 13.481 and a p-value of approximately 0, which means we can reject the null at the 1% significance level.

We can verify this visually, by plotting the real vs. fitted value for this model, where we can see the regression line appears to curve upwards.

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



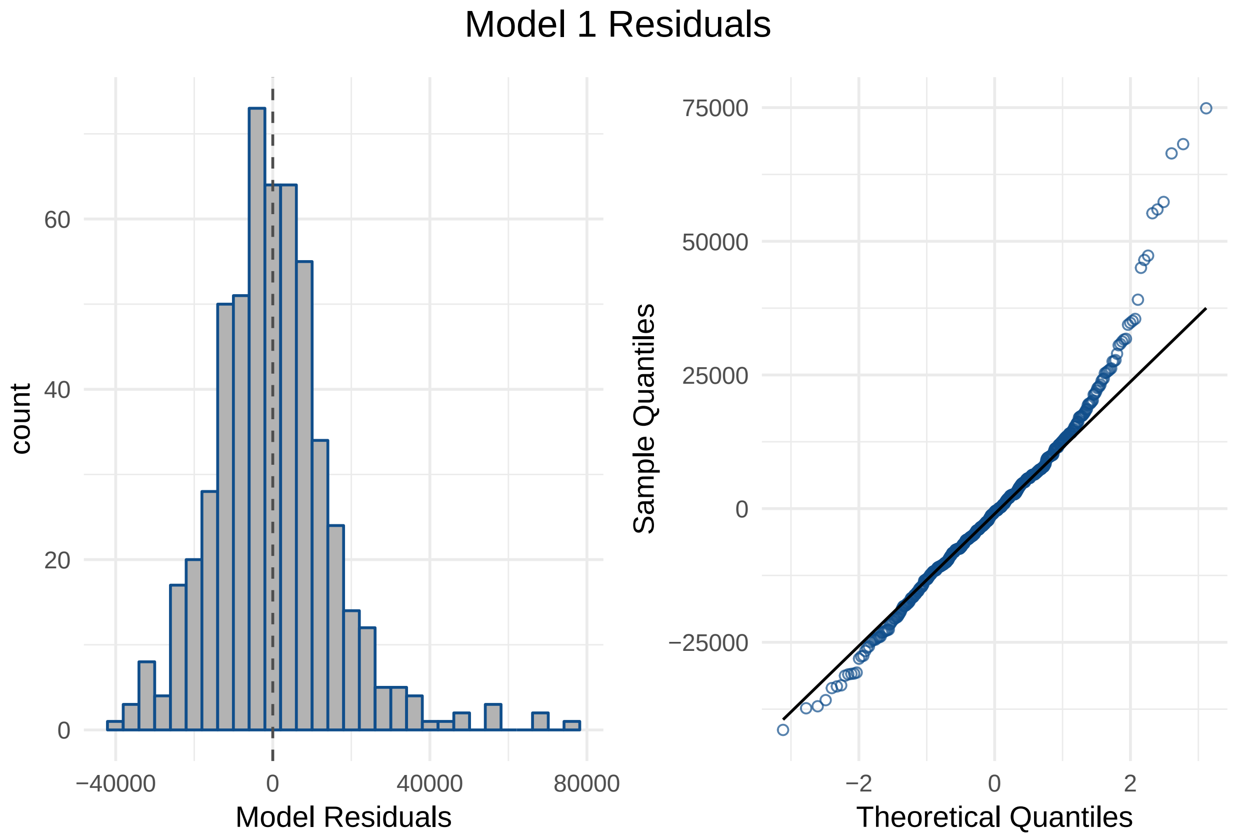
Next, we can perform a Jarque-Bera test to see if the residuals from our regression are normally distributed:

```
jarque.bera.test(mod1$residuals)
```

```
##
##  Jarque Bera Test
##
## data:  mod1$residuals
## X-squared = 247.62, df = 2, p-value < 2.2e-16
```

Here, we can see that the p-value of our test statistic is approximately 0, which implies that we can reject the null hypothesis that our residuals are normally distributed.

Graphically, we can see that the residuals do not appear to be normally distributed when we look at the histogram of the residuals and the normal probability plot (Q-Q plot):



(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?

First, transform the dependent variable, and then re-run the regression:

```
dat$log_sell <- log(dat$sell)
```

```
mod2 <- lm(log_sell~lot+bdms+fb+sty+drv+rec+ffin+ghw+ca+gar+reg,
           data = dat)

summary(mod2)$coefficients
```

```
##                 Estimate   Std. Error    t value     Pr(>|t|)
## (Intercept) 1.002556e+01 4.724349e-02 212.210317 0.000000e+00
## lot         5.057053e-05 4.853947e-06  10.418435 2.908737e-23
## bdms        3.402048e-02 1.450780e-02   2.344978 1.939345e-02
## fb          1.677687e-01 2.064515e-02   8.126299 3.096505e-15
## sty         9.227447e-02 1.282132e-02   7.196956 2.098439e-12
## drv         1.306513e-01 2.834004e-02   4.610130 5.040563e-06
## rec         7.351654e-02 2.632685e-02   2.792455 5.418509e-03
## ffin        9.939967e-02 2.200452e-02   4.517238 7.717764e-06
## ghw         1.783545e-01 4.458477e-02   4.000347 7.217517e-05
## ca          1.780197e-01 2.154722e-02   8.261841 1.138204e-15
## gar         5.075683e-02 1.164704e-02   4.357918 1.575321e-05
## reg         1.271134e-01 2.312783e-02   5.496122 6.021383e-08
```

Again, we can perform a RESET test to see if the linear specification is correct using this transformed dependent variable:

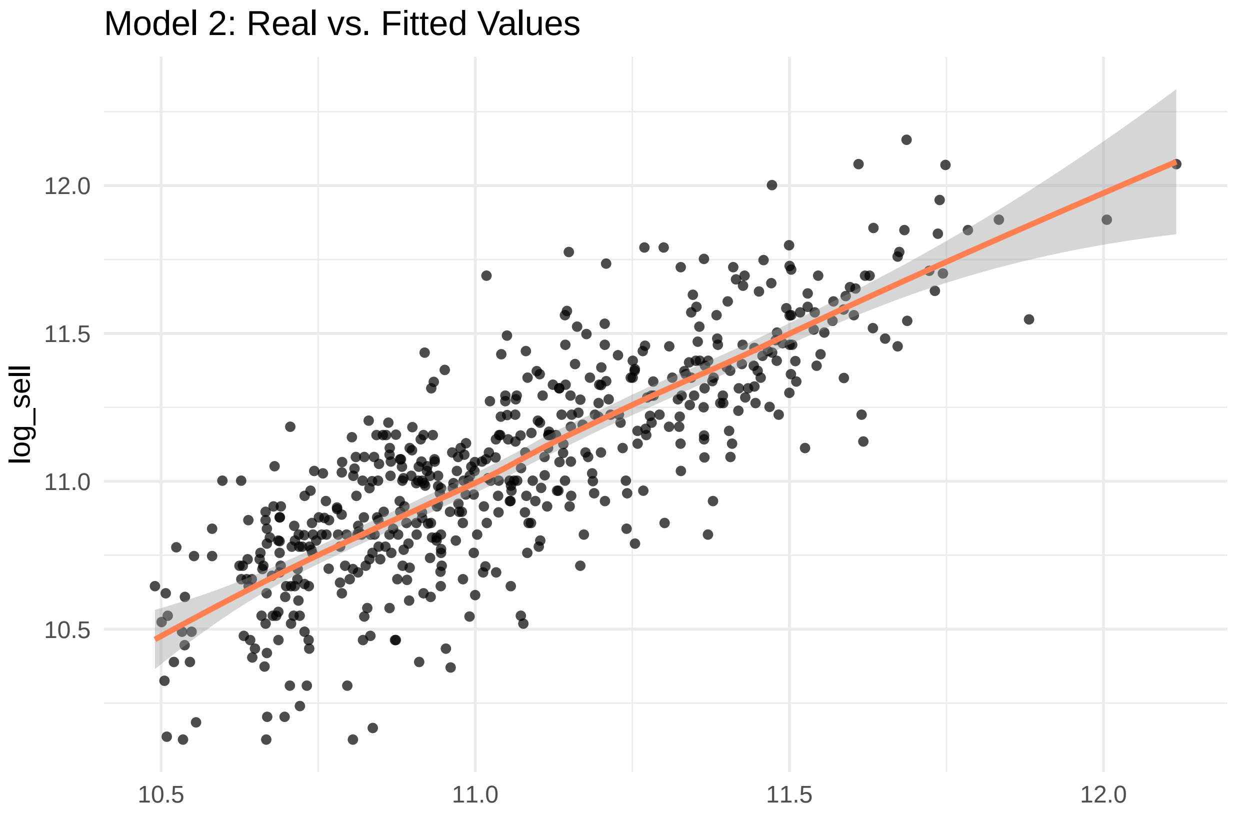
```
# call `resettest()` from `lmtest` package
resettest(mod2, power = 2:3, type = "fitted", data = dat)
```

```
## 
##  RESET test
## 
## data:  mod2
## RESET = 0.13767, df1 = 2, df2 = 532, p-value = 0.8714
```

Here, we can see that the p-value of our RESET test statistic is too large to reject the null, which implies that this model **is** correctly specified in this linear form.

When we plot the real vs. fitted value for this new model, we can indeed see that the regression line has a much straighter, more linear shape.

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

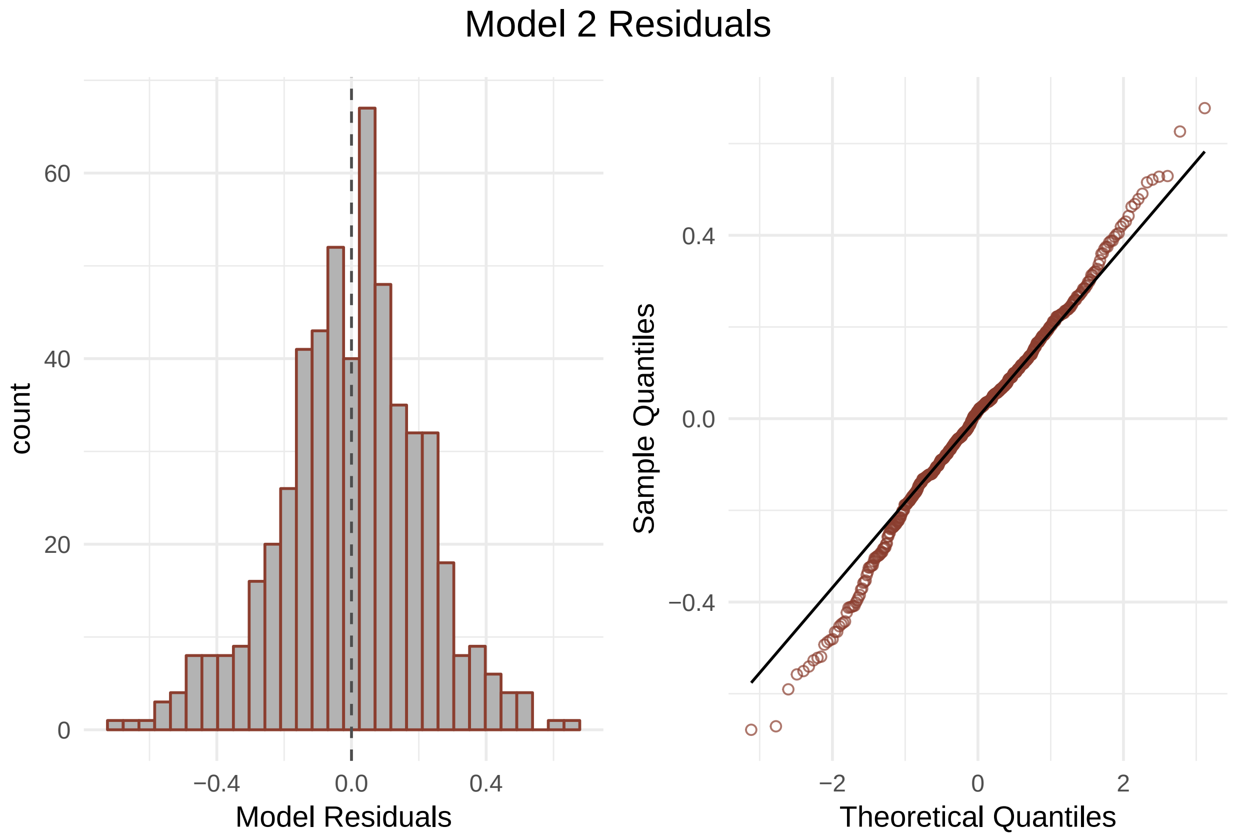


Again, we can check the distribution of our model residuals with a JB test:

```r
jarque.bera.test(mod2$residuals)
```

```
##
##  Jarque Bera Test
##
## data:  mod2$residuals
## X-squared = 8.4432, df = 2, p-value = 0.01467
```

Here, we can see that while the p-value of our JB test statistic is much bigger than that of the first model, it is smaller than the 5% significance level, so we would again reject H_0 here, which implies that for our second model, our residuals are also **not** normally distributed. It is interesting to note that when looking at the plots of the residuals for this model, they do appear to be normally distributed, and at the 1% significance level, the p-value of the JB test would be just big enough to reject the null and assume normality.



(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 of its logarithm?

For this third model, we add the transformed `log_lot` variable to our model and re-run the regression:

```r
dat$log_lot <- log(dat$lot)
```

```
mod3 <- lm(log_sell~lot+log_lot+bdms+fb+sty+drv+rec+ffin+ghw+ca+gar+reg,
           data = dat)

summary(mod3)$coefficients
```

```
##                  Estimate   Std. Error    t value     Pr(>|t|)
## (Intercept)  7.150471e+00 6.829837e-01 10.4694603 1.876854e-23
## lot         -1.490377e-05 1.623681e-05 -0.9179004 3.590862e-01
## log_lot      3.826891e-01 9.069768e-02  4.2193921 2.879096e-05
## bdms         3.489243e-02 1.428629e-02  2.4423716 1.491493e-02
## fb           1.659385e-01 2.033243e-02  8.1612735 2.402834e-15
## sty          9.121302e-02 1.262674e-02  7.2237965 1.757274e-12
## drv          1.068137e-01 2.847057e-02  3.7517232 1.948833e-04
## rec          5.466914e-02 2.630421e-02  2.0783422 3.815573e-02
## ffin         1.052473e-01 2.171056e-02  4.8477460 1.640526e-06
## ghw          1.790867e-01 4.389978e-02  4.0794431 5.204821e-05
## ca           1.643382e-01 2.146236e-02  7.6570449 9.007543e-14
## gar          4.826474e-02 1.148320e-02  4.2030731 3.087586e-05
## reg          1.343625e-01 2.283703e-02  5.8835350 7.100650e-09
```

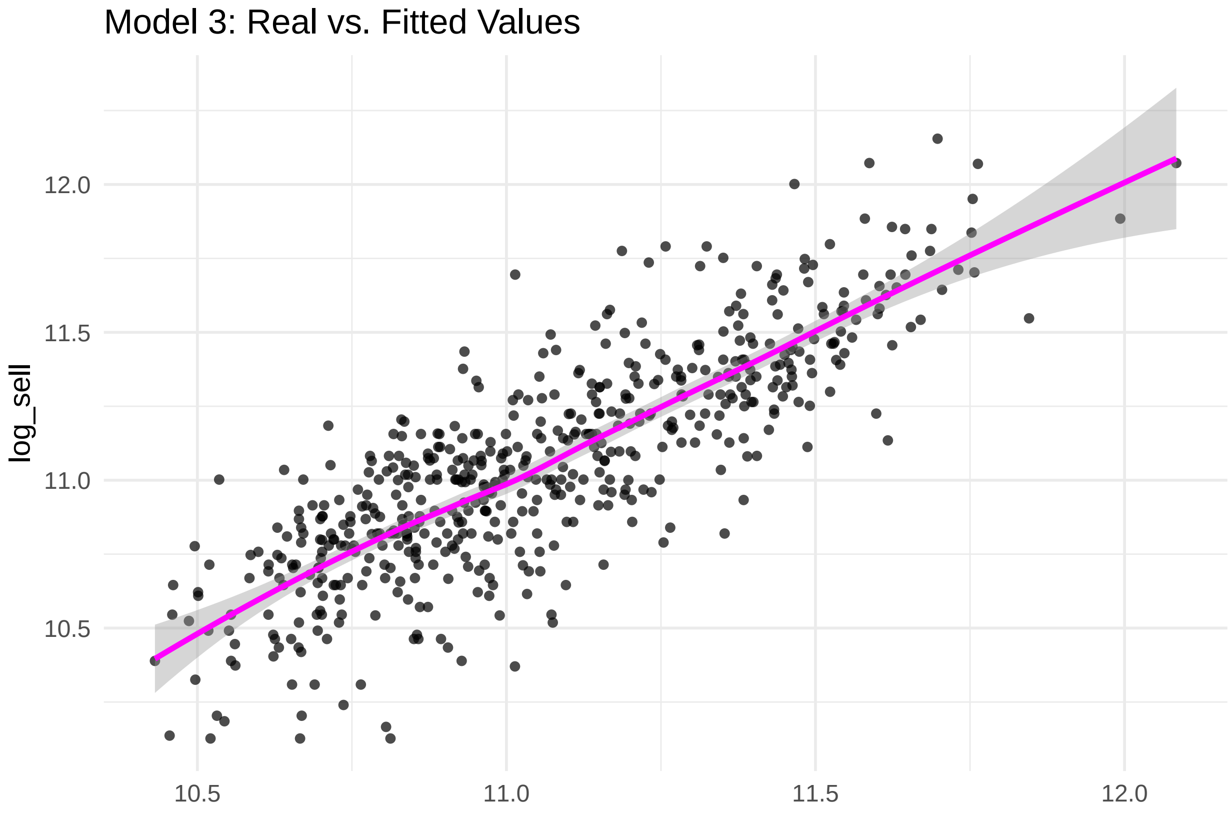
Again, we check the linear specification of this model with the RESET test:

```
# call `resettest()` from `lmtest` package
resettest(mod3, power = 2:3, type = "fitted", data = dat)
```

```
##
##  RESET test
##
## data:  mod3
## RESET = 0.05263, df1 = 2, df2 = 531, p-value = 0.9487
```

Similar to the previous model, we do **not** reject H_0 here, which means this third model is also correctly specified. With the lowest JB test statistic and highest p-value so far, this model outperforms the other two.

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



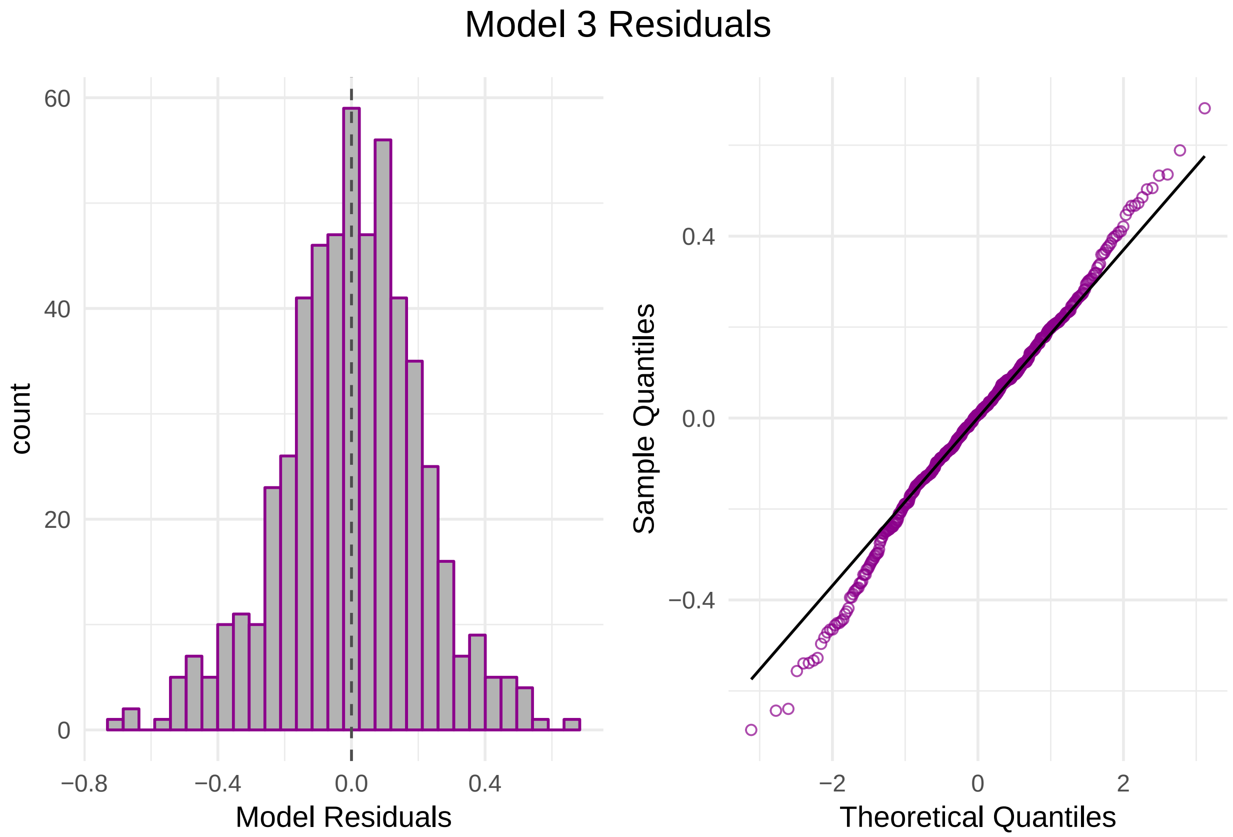
When we inspect the residuals for this third model we find:

```
jarque.bera.test(mod3$residuals)
```

```
##
##  Jarque Bera Test
##
## data:  mod3$residuals
## X-squared = 9.3643, df = 2, p-value = 0.009259
```

Here again, according to the JB test statistic, we reject the assumption of normality in the distribution of the residuals of our model. We can see from the lower p-value and from the shape of the plotted distributions below, that the distribution of the error terms here are less normal shaped than in the previous model.

Finally, with regard to the `lot` and `log_lot` variables in the three models, after accounting for their respective regression coefficients and model test statistics, it would appear that the best way forward would be to omit the `lot` variable while keeping `log_lot`. Our third model, where we included `log_lot` had the lowest JB test statistic (and corresponding highest p-value), which implies it was the best linear fit. Simultaneously, in model 3, the regression coefficient for `log` is approximately zero, so it cancels out of our final regression equation in any case. Conversely, when we look at model 2, where we only had `log` but did not include `log_lot`, the JB statistic was less favorable while the regression coefficient for `log` also came out to 0 and cancelled out of the model, meaning there is no real advantage to only using `lot` without including `log_lot`.



(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?

Here, we repeat the regression from model 3, but we drop `lot` and add interaction variables between `log_lot` with all of the remaining independent variables.

```
##
## Call:
## lm(formula = log_sell ~ log_lot + bdms + fb + sty + drv + rec +
##     ffin + ghw + ca + gar + reg + log_lot * bdms + log_lot *
##     fb + log_lot * sty + log_lot * drv + log_lot * rec + log_lot *
##     ffin + log_lot * ghw + log_lot * ca + log_lot * gar + log_lot *
##     reg, data = dat)
##
## 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 ***
## log_lot       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  
## log_lot:bdms  0.002070   0.038654   0.054   0.9573  
## log_lot:fb    0.062037   0.050145   1.237   0.2166  
## log_lot:sty  -0.046361   0.035942  -1.290   0.1977  
## log_lot:drv   0.191542   0.087361   2.193   0.0288 *
## log_lot:rec  -0.188462   0.076373  -2.468   0.0139 *
## log_lot:ffin  0.015913   0.052851   0.301   0.7635  
## log_lot:ghw   0.081135   0.106929   0.759   0.4483  
## log_lot:ca    0.059549   0.058024   1.026   0.3052  
## log_lot:gar  -0.041359   0.030142  -1.372   0.1706  
## log_lot: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
```

Looking at the regression results, we can see that two of the interaction variables, `log_lot*drv` and `log_lot*rec` are individually significant at the 5% level.

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

First, we need to run the regression for our new model which only uses the two interaction variables (`log_lot*drv` and `log_lot*rec`) that we found significant in the previous part:

```
mod5 <- lm(log_sell~log_lot+bdms+fb+sty+drv+rec+ffin+ghw+ca+gar+reg+
             log_lot*drv+log_lot*rec,
           data = dat)

summary(mod5)$coefficients
```

```
##                Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)  8.74188916 0.62862956 13.906265 1.011640e-37
## log_lot      0.17905742 0.07706633  2.323419 2.053264e-02
## bdms         0.03881271 0.01430118  2.713951 6.864273e-03
## fb           0.16145102 0.02025353  7.971500 9.624711e-15
## sty          0.09082775 0.01254156  7.242141 1.557277e-12
## drv         -1.18996148 0.66461748 -1.790446 7.395079e-02
## rec          1.50253115 0.62552752  2.402022 1.664652e-02
## ffin         0.10276169 0.02157347  4.763336 2.459225e-06
## ghw          0.18448135 0.04368200  4.223281 2.832274e-05
## ca           0.16525994 0.02120846  7.792172 3.480136e-14
## gar          0.04690188 0.01142088  4.106678 4.645737e-05
## reg          0.13260282 0.02254997  5.880398 7.235580e-09
## log_lot:drv  0.15943009 0.08124263  1.962395 5.023736e-02
## log_lot:rec -0.16825882 0.07270422 -2.314292 2.103167e-02
```

Looking at the p-values for our regression coefficients, we note that the t-scores for both interaction variables have gone down compared to the previous model, so that `log_lot*rec` is still significant at the 5% threshold, but `log_lot*drv` is just barely above the threshold and would not be considered significant at 5%.

To gain some intuition into whether or not we should keep both variables in the model, we can perform an F-test on their joint significance: $F(g, n-k)$, where $g =$ the number of parameter restrictions in H_0, n is the number of observations, and k is the number of variables in the unrestricted model. The null hypothesis (H_0) in this case states that the effect of our two interactions variables in the model are equal to zero (i.e. jointly insignificant).

```
# set H_0 restrictions
H_0 <- c("log_lot:drv=0", "log_lot:rec=0")

# run heteroskedasticity-robust version of the F-test
linearHypothesis(mod5, H_0, white.adjust = "hc1")
```

```
## Linear hypothesis test
##
## Hypothesis:
## log_lot:drv = 0
## log_lot:rec = 0
##
## Model 1: restricted model
## Model 2: log_sell ~ log_lot + bdms + fb + sty + drv + rec + ffin + ghw +
##     ca + gar + reg + log_lot * drv + log_lot * rec
##
## Note: Coefficient covariance matrix supplied.
##
##   Res.Df Df      F   Pr(>F)
## 1    534
## 2    532  2 5.1626 0.006017 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Here we can see that our F-test has degrees of freedom equal to: F(2, 532), and a value of 5.163, with a p-value of 0.006, which means that we would reject H_0, and our interaction variables are indeed jointly significant at the 1% significance level.

(f) Now perform model specification on the interaction variables using the general-to-specific approach (Only eliminate the interaction effects).

If we return to our fourth model from part (d), we can also perform model specification on all of the interaction variables by using the general-to-specific approach: that is, we regress everything, eliminate the variable with the lowest t-value, and repeat with as many rounds needed until all of the remaining interaction variables are significant.

Under the 5% significance threshold, we perform 10 rounds of general-to-specific regression until we arrive at a model where all of the interaction variables are significant. For our model 4 from part(d), the final model, it is only the `log_lot*rec` interaction variable that is still significant at the 5% level, which confirms what we saw in our model 5 regression results from part (e).

Table 1: General-to-Specific model specification: Model 4 interaction variables

	Dependent variable:									
	log_sell									
	Round									
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
log_lot:bdms	t = 0.054	t = 0.052								
log_lot:fb	t = 1.237	t = 1.238	t = 1.405	t = 1.417	t = 1.405	t = 1.418	t = 1.321			
log_lot:sty	t = −1.290	t = −1.312	t = −1.372	t = −1.403	t = −1.412	t = −1.344	t = −1.331	t = −1.058		
log_lot:drv	t = 2.193	t = 2.259	t = 2.277	t = 2.264	t = 2.241	t = 2.235	t = 2.018	t = 2.047	t = 1.962	
log_lot:rec	t = −2.468	t = −2.473	t = −2.479	t = −2.481	t = −2.496	t = −2.395	t = −2.327	t = −2.263	t = −2.314	t = −2.213
log_lot:ffin	t = 0.301	t = 0.319	t = 0.328							
log_lot:ghw	t = 0.759	t = 0.759	t = 0.758	t = 0.778						
log_lot:ca	t = 1.026	t = 1.027	t = 1.027	t = 1.066	t = 1.034					
log_lot:gar	t = −1.372	t = −1.377	t = −1.380	t = −1.383	t = −1.368	t = −1.164				
log_lot:reg	t = 0.027									
R^2	0.695	0.695	0.695	0.695	0.695	0.694	0.693	0.692	0.692	0.689

Note: This table shows the general-to-specific (GTS) regression results for our fourth model from part (d). Round 1 represents the regression of model 4 in its entirety, and for each subsequent round, only the interaction term with the lowest t-score was eliminated. This table only displays the t-scores of the interaction variables in each round.

(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.)

If the condition of the house is missing, the effect of the air conditioning variable (`ca`) on the log of the sale price (`log_sell`) will be overestimated, as houses with air conditioning also tend to be much newer than houses without air-conditioning, and how "new" a house is (i.e. its age) should, in theory, have a direct effect on home sales prices.

(h) 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)?

Here, our final model is similar model 3 from part (c), except that we remove the `lot` variable, leaving only `log_lot`, plus all of the other original explanatory variables (and no interaction variables).

$$log(sell) = \beta_0+\beta_1 log(lot)+\beta_2 bdms+\beta_3 fb+\beta_4 sty+\beta_5 drv+\beta_6 rec+\beta_7 ffin+\beta_8 ghw+\beta_9 ca+\beta_{10} gar+\beta_{11} reg+\varepsilon$$

After we split the dataset and run a regression on the new subset using this model, our model coefficients come out to be:

```r
mod6 <- lm(log_sell~log_lot+bdms+fb+sty+drv+rec+ffin+ghw+ca+gar+reg,
           data = dat_sub1)

summary(mod6)$coefficients
```

```
##                Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 7.67309380 0.29240366 26.241442 5.217086e-88
## log_lot     0.31377577 0.03614993  8.679843 1.109117e-16
## bdms        0.03787207 0.01743737  2.171891 3.046897e-02
## fb          0.15237512 0.02469448  6.170412 1.714913e-09
## sty         0.08823830 0.01819354  4.849979 1.790572e-06
## drv         0.08641375 0.03140936  2.751210 6.215867e-03
## rec         0.05465005 0.03392133  1.611082 1.079749e-01
## ffin        0.11471077 0.02673226  4.291099 2.247834e-05
## ghw         0.19869752 0.05301461  3.747977 2.053704e-04
## ca          0.17763419 0.02723870  6.521391 2.173975e-10
## gar         0.05301455 0.01479703  3.582784 3.832054e-04
## reg         0.15116021 0.04214854  3.586369 3.781456e-04
```

Next, we plug these coefficients back into our model as the β values and predict `log_sell` prices for the remaining 146 observations from our dataset. We can compare those predictions to the actual `log_sell` prices in the dataset and then calculate the MAE:

```r
# First we create a table to store the results of our respective models
nn <- 1:146

dat_pred <- tibble(obs = dat_sub2$obs, log_sell_actual = dat_sub2$log_sell[nn])
dat_pred$log_sell_predicted <- NA

# Next we can loop through all the rows in dat_sub2 and save
# the predicted values for log_sell to our table
for (i in 1:length(nn)) {
  temp <- (7.67309380 + 0.31377577*dat_sub2$log_lot[i] + 0.03787207*dat_sub2$bdms[i] +
             0.15237512*dat_sub2$fb[i] +  0.0882383*dat_sub2$sty[i] + 0.08641375*dat_sub2$drv[i] +
             0.05465005*dat_sub2$rec[i] + 0.11471077*dat_sub2$ffin[i] + 0.19869752*dat_sub2$ghw[i] +
             0.17763419*dat_sub2$ca[i] + 0.05301455*dat_sub2$gar[i] + 0.15116021*dat_sub2$reg[i])

  dat_pred[i, 3] <- temp

}
```

Just to check, we can have a look at the first few entries of our results table:

```r
knitr::kable(head(dat_pred, 10))
```

obs	log_sell_actual	log_sell_predicted
401	11.43496	11.51385
402	11.23849	11.47529
403	11.25803	11.38201
404	11.28978	11.18935
405	11.28978	11.32946
406	11.36210	11.36018
407	11.37366	11.29989
408	11.37939	11.30701
409	11.39639	11.46922
410	11.40645	11.53099

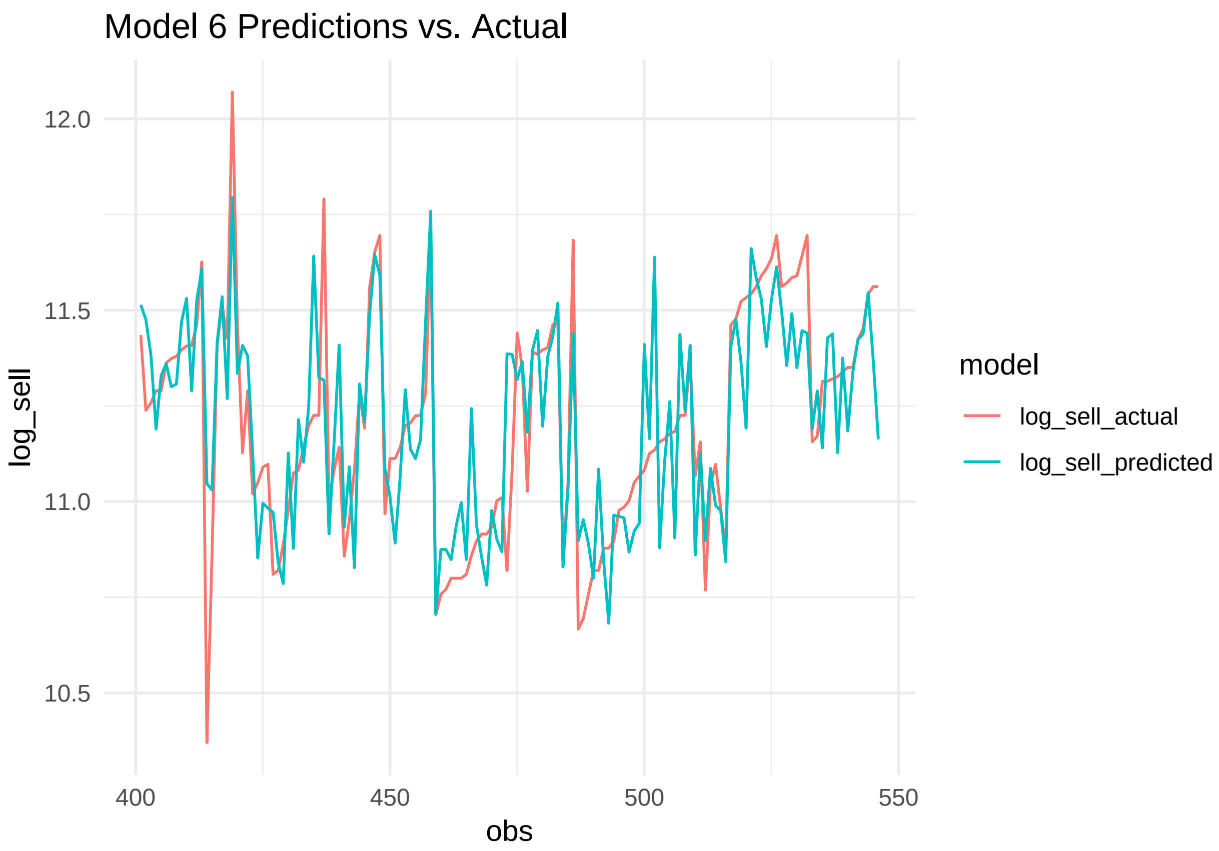
Finally, we can calculate the MAE of our model predictions:

```r
MAE_mod6 =  mean(abs(dat_pred$log_sell_actual - dat_pred$log_sell_predicted))

print(paste("Model 6 predictions MAE =", round(MAE_mod6, 3)))
```

```
## [1] "Model 6 predictions MAE = 0.128"
```

The actual variability (in standard deviations) of the log price `log_sell` in our original dataset is: 0.3719849. Comparing this to our MAE for model 6, we can see that our model has much smaller variability and, hence, it has good predictive power.


Model 6 Predictions vs. Actual
log_sell
12.0
11.5
11.0
10.5
400
450
500
550
obs
model
log_sell_actual
log_sell_predicted