Alison Au
8/8/21

A7: Modeling Assignment #3: Validation, Automated Variable Selection, and Finalizing the Model

Introduction:

In the first two modeling assignments, we gained familiarity of the Ames, Iowa housing data and what may be a good linear regression model for predicting home sales price by manually fitting different continuous variables. This assignment builds on this with the goal of finalizing the regression model. This entails carving out training and test data sets, considering continuous and categorical explanatory variables together, leveraging automated variable selection, and validating the model on the test data set. Throughout this process, we will also examine various statistics as done before such as goodness-of-fit, hypothesis testing, diagnostic plots, and outliers and influential points. In the end, we will arrive at a suitable model for predicting residential home sale prices.

Results

(1) Preparing the Categorical Variables

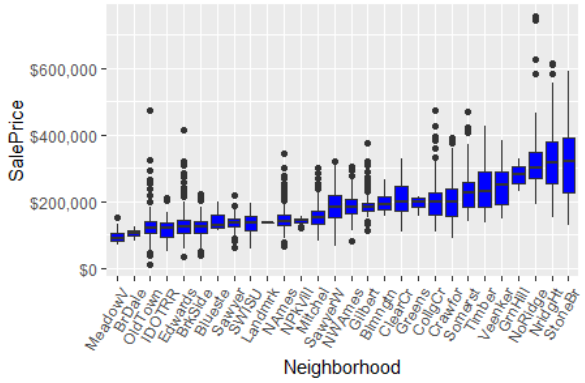
Before diving into the modeling, we will clean the data similar to the process in prior assignments. The sample population will focus on residential homes. Thus, we have dropped agriculture, commercial, and industrial properties (1% of data). A summary is provided below.

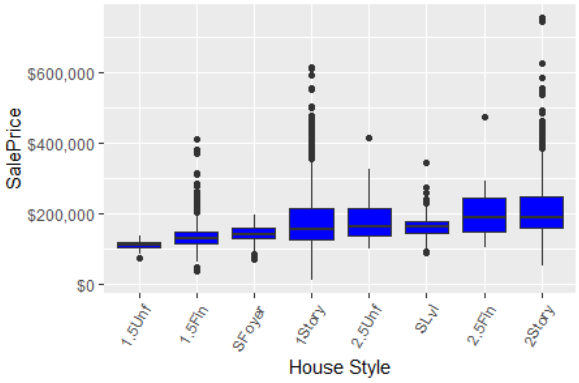
drop_conditions	drop_cnt	cume_drop_cnt	final_row_cnt
Before dropping data	0	0	2930
Non-residential zoning	29	29	2901

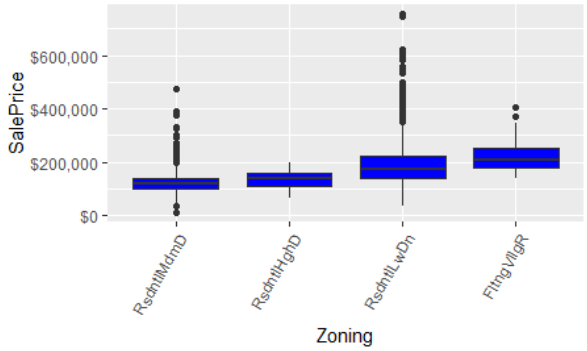
Missing numeric data has primarily been imputed with the variables' means. The exceptions to this are Garage Year Built and Masonry Veneer Area, which have been dropped due to a large amount of missing data and suspected lack of importance. Lot Frontage "NA"s have also been replaced with zeroes, as this seems to be the equivalent rather than the data being missing, and it could be a potential predictor. "NA"s are also recorded for categorical/ordinal variables. They appear to be used when the feature is not in the home (e.g. basement, pool) rather than missing data, so these have been replaced with 0's. The other ordinal labels from Excellent to Poor have also been relabeled to ordinal numbers from 5 (Excellent) to 1 (Poor).

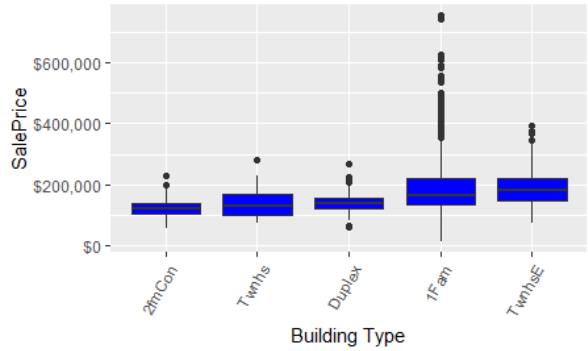
Extreme outliers that are very unusual for the population of interest have also been adjusted down to the next highest value. Outlier identification was focused on variables that will potentially be in the model. This impacts a handful of data points across TotRmsAbvGrd, and GarageArea (<1% of total data). Other outliers have been retained as these seem to be legitimate data points representing the population. We may find out later that these variables should instead be transformed.

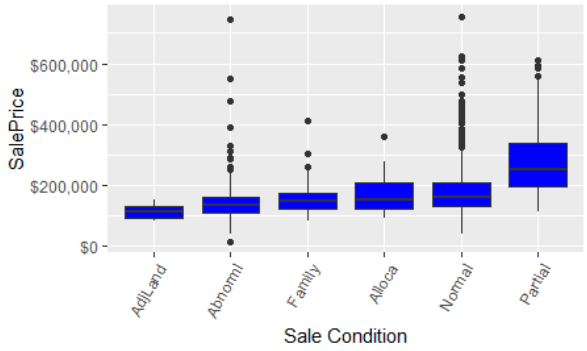
After learnings from EDA, we will now examine which categorical variables seem predictive of sale price using boxplots, focusing on ones that seem reasonable.







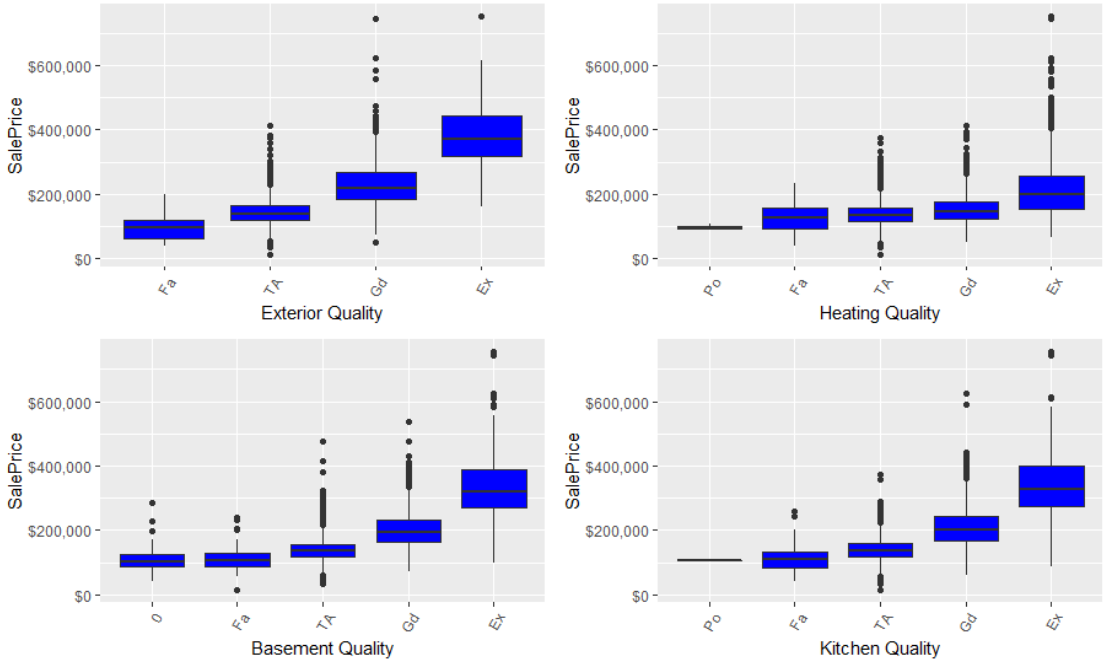




House Style has numerous groups with limited variation in sales price compared to these other variables. Grouping one-story vs. two-story homes and fitting a regression model produced an R-squared of just 4%. Zoning and Building Type also do not vary much by group. All three of these will be dropped from our consideration set.

On the other hand, sale price can vastly differ depending on the Neighborhood. This makes sense since a well-known and important factor in buying a house is “location, location, location.” However, there are several neighborhoods which adds model complexity, and some may have limited sample. We can group them later rather than throwing out the variable for ease. Sale Condition also shows clear variation. We will keep these two variables in consideration.

Next, we will examine ordinal variables. While these can be treated as numeric, this assumes going from Poor to Fair is the same as Good to Excellent. The relationship with sale price may not be linear.



These variables depict differentiation in sale price depending on quality. Heating Quality is the only one with more muted differences, so this will be dropped from consideration. For the other three variables, excellent and good ratings lead to higher prices, so perhaps separate dummy variables should be created for each of these. Typical, fair, and poor ratings could be grouped together.

Let us first view the summary statistics of sale price segmented by the levels of these five categorical variables.

Sale Price by Neighborhood

Category	count	min	q25	mean	median	q75	max	sd
NoRidge	71	190000	270395	330319	302000	349000	755000	101445
StoneBr	51	130000	225000	324229	319000	393216	591587	119273
NridgHt	166	154000	253470	322018	317750	378750	615000	95932
GrnHill	2	230000	255000	280000	280000	305000	330000	70711
Veenker	24	150000	190750	248315	250250	291000	385000	65475
Timber	72	137500	186025	246600	232106	288519	425000	69326
Somerst	182	139000	185000	229707	225500	259375	468000	57437
ClearCr	44	107500	171000	208662	197500	244550	328000	51280
Crawfor	103	90350	154950	207551	200624	238500	392500	65230
CollgCr	267	110000	160875	201803	200000	228250	475000	54188
Blmngtn	28	156820	175425	196662	191500	216248	264561	29318
Greens	8	155000	183500	193531	198000	212062	214000	21999
Gilbert	165	115000	173000	190647	183000	195500	377500	33050
NWAmes	131	82500	163500	188407	181000	206750	306000	37688
SawyerW	125	67500	148325	184070	180000	220000	320000	48996
Mitchel	113	80000	135000	162941	153500	173000	300000	40828
NAmes	443	68000	127000	145097	140000	157500	345000	31883
Blueste	10	115000	121725	143590	130500	159625	200000	30159
NPkVill	23	120000	137700	140711	143750	147000	155000	9340
Landmrk	1	137000	137000	137000	137000	137000	137000	NA
Sawyer	151	62383	125000	136751	135000	150000	219000	23130
SWISU	47	60000	114950	135959	136500	159717	200000	30411
Edwards	191	35000	102750	130131	125000	144250	415000	48045
BrkSide	108	39300	106425	124756	126750	138775	223500	35741
OldTown	236	37900	103875	124699	119950	140000	475000	43797
IDOTRR	68	50000	89875	114209	117450	135000	212300	32385
BrDale	30	83000	96250	105608	106000	113650	125500	12145
MeadowV	37	71000	81000	95756	88250	105000	151400	20131

Sale Price by Sale Condition

Category	count	min	q25	mean	median	q75	max	sd
Partial	242	113000	198111	274579	250290	336850	611657	100021
Normal	2397	35000	130000	176172	159000	207000	755000	70797
Alloca	22	89471	118884	171733	152123	204881	359100	66942
Family	46	79275	121025	157489	144400	174000	409900	63377
Abnorml	178	37900	106625	145387	131500	161425	745000	80051
AdjLand	12	81000	89500	108917	110000	126375	150000	21988

Sale Price by Exterior Quality

Category	count	min	q25	mean	median	q75	max	sd
Ex	104	176500	318578	383737	372198	447196	755000	102745
Gd	988	52000	183150	230864	219500	267454	745000	70366
TA	1778	35000	119500	144159	139500	163500	415000	40953
Fa	27	39300	62750	97221	94550	120500	200000	37537

Sale Price by Basement Quality

Category	count	min	q25	mean	median	q75	max	sd
Ex	255	100000	272000	335218	320000	388500	755000	104280
Gd	1219	70000	163995	201727	192000	232649	538000	56611
TA	1263	35000	118500	140724	136000	157000	475000	40087
Fa	84	55000	88438	112478	108450	130050	240000	36478
0	76	39300	86850	110162	102950	125250	284700	37988

Sale Price by Kitchen Quality

Category	count	min	q25	mean	median	q75	max	sd
Ex	202	86000	275000	339732	330950	402440	755000	113483
Gd	1159	59000	168082	210910	201000	245000	625000	63562
TA	1470	35000	118425	140567	137000	160000	375000	37691
Fa	65	39300	82000	109691	111500	131000	260000	40641
Po	1	107500	107500	107500	107500	107500	107500	NA

Exterior Quality has the largest mean difference between the top and bottom categories ($286,516) followed by Neighborhood ($234,563). The spreads for Kitchen Quality and Exterior Quality are also close to this magnitude. This variation in sale price indicates these categories are helpful differentiators, thus likely stronger predictors. We will focus on these variables and drop Sale Condition from consideration. The summary tables also confirm that sample sizes get thin for some categories. Grouping them makes sense to combat this and will also make the categorical variables easier to work with.

I will combine the neighborhoods into four groups of equal size based on the mean sale price. The summary statistics are below. Each group contains seven neighborhoods with group 1 having the highest average sale prices and group 4 having the lowest.

Category	count	min	q25	mean	median	q75	max	sd
1	568	130000	210300	280854	263492	326250	755000	95111
2	746	82500	168000	197900	190000	221725	475000	48577
3	866	62383	129212	151453	144000	165000	345000	37647
4	717	35000	100000	123605	120000	139500	475000	41242

This shows clear delineation in sales price between the groups.

I then tested fitting a regression model with the dummy coded variables to ensure it is still a strong predictor. In the output below, neigh1 corresponds to the group of neighborhoods with the highest

average sale price, neigh2 is the next most expensive group, etc. Group 4 (most affordable neighborhood) is used as the basis of interpretation.

```
subdat4$neighg <- ifelse(subdat4$Neighborhood %in% c("NoRidge", "StoneBr", "NridgHt",
"GrnHill", "Veenker", "Timber", "Somerst"), 1,
       ifelse(subdat4$Neighborhood %in% c("Crawfor", "ClearCr", "CollgCr", "Blmngtn",
"Greens", "Gilbert", "NWAmes"), 2,
              ifelse(subdat4$Neighborhood %in% c("SawyerW", "Mitchel", "NAmes",
       "Blueste", "NPkVill", "Sawyer", "Landmrk"), 3,
                     ifelse(subdat4$Neighborhood %in% c("SWISU", "Edwards", "BrkSide",
              "OldTown", "IDOTRR", "BrDale", "MeadowV"), 4, 0))))

subdat4$neigh1 <- ifelse(subdat4$neighg == 1,1,0)
subdat4$neigh2 <- ifelse(subdat4$neighg == 2,1,0)
subdat4$neigh3 <- ifelse(subdat4$neighg == 3,1,0)
subdat4$neigh4 <- ifelse(subdat4$neighg == 4,1,0)
```

```
Call:
lm(formula = logSalePrice ~ neigh1 + neigh2 + neigh3, data = subdat4)

Residuals:
    Min      1Q  Median      3Q     Max
-1.2142 -0.1545 -0.0121  0.1550  1.3937

Coefficients:
            Estimate Std. Error t value            Pr(>|t|)
(Intercept) 11.67734    0.01008 1158.79 <0.0000000000000002
neigh1       0.81675    0.01516   53.89 <0.0000000000000002
neigh2       0.49057    0.01411   34.76 <0.0000000000000002
neigh3       0.22295    0.01362   16.36 <0.0000000000000002

Residual standard error: 0.2698 on 2893 degrees of freedom
Multiple R-squared:  0.5329,    Adjusted R-squared:  0.5324
F-statistic:  1100 on 3 and 2893 DF,  p-value: < 0.00000000000000022
```

The model has a strong R^2 at 53% and high statistical significance in predicting sale price, so I will keep this as a potential predictor. Note that log(SalePrice) is used as the response variable given the discovery in the previous assignment that this better follows normality assumptions. Models in this report will continue using the log transformation of Sale Price.

For Exterior, Basement, and Kitchen Quality, excellent-rated homes have a much higher sale price than the other groups, followed by good-rated. This makes sense as buyers commonly place high value on these aspects. Average, fair, and poor quality all have similar prices. I will collapse these last three ratings into one group.

The dummy variables have been defined below. Low (typical + fair + average) is used as basis of interpretation for all of these Quality variables.

Exterior Quality

```
subdat4$ExterEx_dum <- ifelse(subdat4$ExterQual == "Ex", 1, 0)
subdat4$ExterGd_dum <- ifelse(subdat4$ExterQual == "Gd", 1, 0)
```

Basement Quality

```
subdat4$BsmtEx_dum <- ifelse(subdat4$BsmtQual == "Ex", 1, 0)
subdat4$BsmtGd_dum <- ifelse(subdat4$BsmtQual == "Gd", 1, 0)
```

Kitchen Quality

```
subdat4$KitEx_dum <- ifelse(subdat4$KitchenQual == "Ex", 1, 0)
```

```
subdat4$KitGd_dum <- ifelse(subdat4$KitchenQual == "Gd", 1, 0)
```

(2) The Predictive Modeling Framework

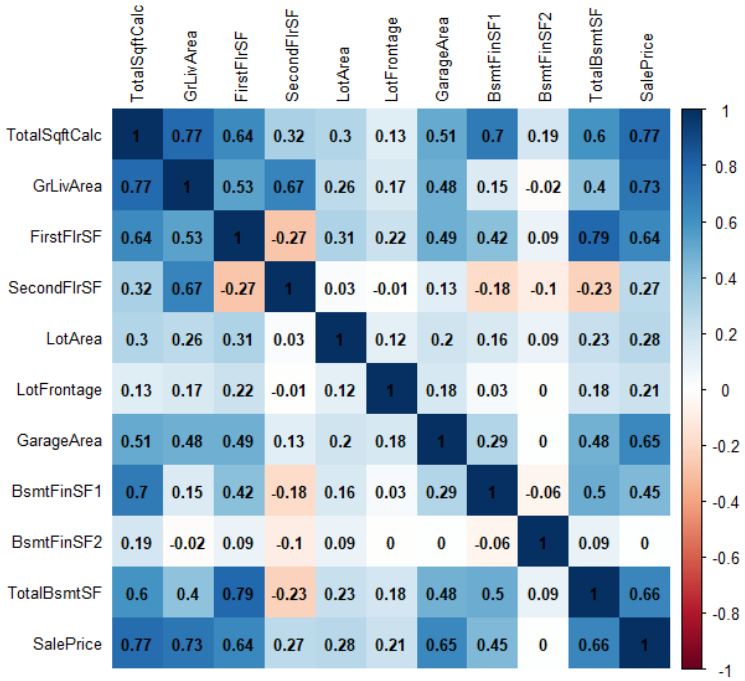
The Ames data has been split into 70/30 training/test data sets. A summary of the observation counts for the original total and partitioned data is below.

	# Observations	% Observations
Total	2897	100.000
Training	2029	70.038
Test	868	29.962

This confirms the training and test observations have a 70/30 ratio and sum to the original total.

(3) Model Identification by Automated Variable Selection

After setting up the test/training data and examining the categorical variables, I will now select a pool of candidate predictors. One consideration to keep in mind is the high amount of collinearity that likely exists between variables in the data set. Highly correlated predictors should not be included in the model nor in the variable selection, as the algorithm will automatically select them together. Multi-collinearity is especially apparent with the metrics related to size, so I will take a look at their correlation to determine which make sense to retain. My hunch is that TotalSqftCalc would make the most sense as it sums the above ground and basement areas.



	TotalSqftCalc	GrLivArea	FirstFlrSF	SecondFlrSF	LotArea	LotFrontage	GarageArea	BsmtFinSF1	BsmtFinSF2	TotalBsmtSF	SalePrice
TotalSqftCalc	1	0.77	0.64	0.32	0.3	0.13	0.51	0.7	0.19	0.6	0.77
GrLivArea	0.77	1	0.53	0.67	0.26	0.17	0.48	0.15	-0.02	0.4	0.73
FirstFlrSF	0.64	0.53	1	-0.27	0.31	0.22	0.49	0.42	0.09	0.79	0.64
SecondFlrSF	0.32	0.67	-0.27	1	0.03	-0.01	0.13	-0.18	-0.1	-0.23	0.27
LotArea	0.3	0.26	0.31	0.03	1	0.12	0.2	0.16	0.09	0.23	0.28
LotFrontage	0.13	0.17	0.22	-0.01	0.12	1	0.18	0.03	0	0.18	0.21
GarageArea	0.51	0.48	0.49	0.13	0.2	0.18	1	0.29	0	0.48	0.65
BsmtFinSF1	0.7	0.15	0.42	-0.18	0.16	0.03	0.29	1	-0.06	0.5	0.45
BsmtFinSF2	0.19	-0.02	0.09	-0.1	0.09	0	0	-0.06	1	0.09	0
TotalBsmtSF	0.6	0.4	0.79	-0.23	0.23	0.18	0.48	0.5	0.09	1	0.66
SalePrice	0.77	0.73	0.64	0.27	0.28	0.21	0.65	0.45	0	0.66	1

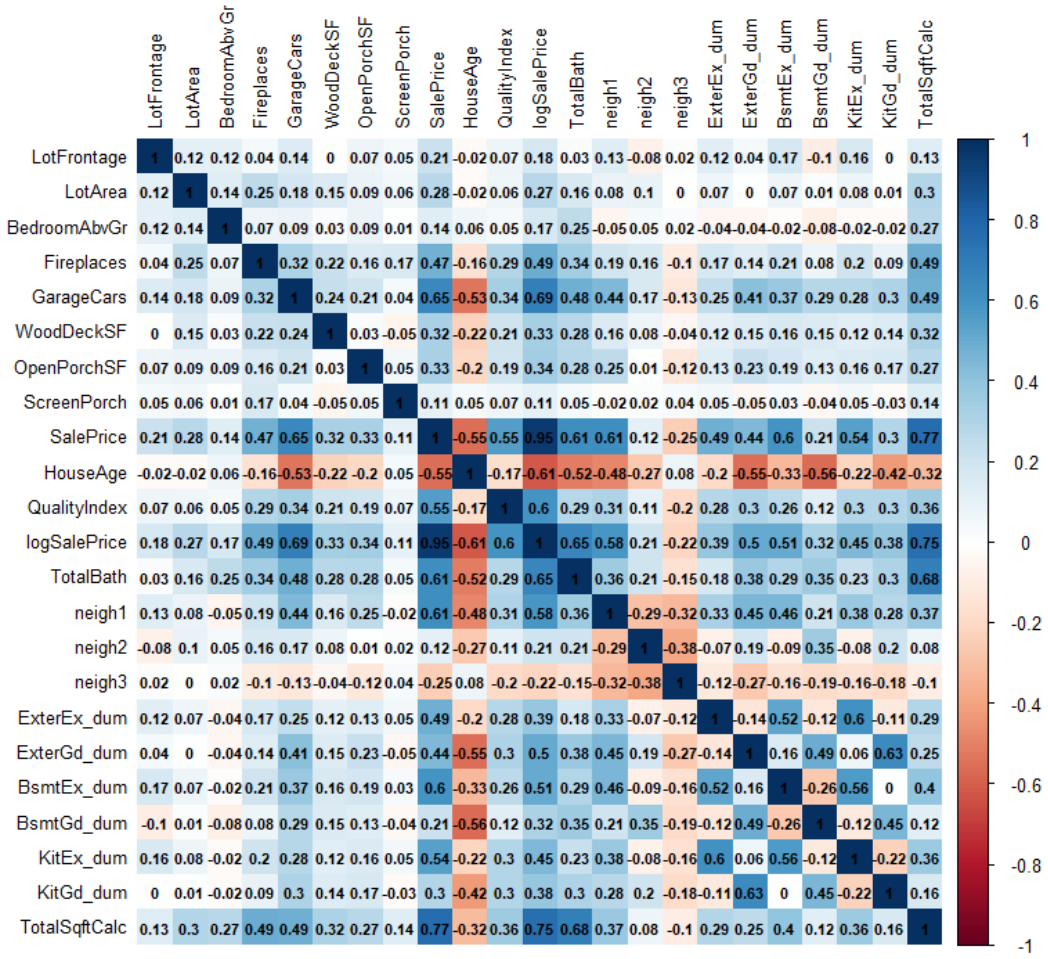
Indeed, TotalSqftCalc has the highest correlation with SalePrice, while being highly correlated with other variables. We will drop those that are accounted for in TotalSqftCalc (e.g. GrLivArea) and other variables with a high correlation coefficient. LotArea, LotFrontage, and GarageArea look fine to retain for now.

Similarly, variables used to calculate QualityIndex and YearBuilt will be dropped. The only categorical factors that will be kept are dummy coded Neighborhood and Kitchen based on the earlier investigation. The final whittling down is based on variables that have almost no correlation with sale price or high correlation with others. An example of the latter, GarageArea and GarageCars strongly correlate.

GarageCars correlates slightly better with SalePrice, so only this variable is kept. The pool of potential predictors that will be used in the automated variable selection are below.

#	Variable	Description
1	LotFrontage	Linear feet of street connected to home
2	LotArea	Lot size (sq. ft.)
3	BedroomAbvGr	# of bedrooms above ground (excludes basement)
4	Fireplaces	# of fireplaces
5	GarageCars	Garage size based on car capacity
6	WoodDeckSF	Size of wood deck (sq. ft.)
7	OpenPorchSF	Size of open porch (sq. ft.)
8	ScreenPorch	Size of porch area (sq. ft.)
9	HouseAge	Age of House; calculated as YrSold - YearBuilt
10	QualityIndex	Overall quality and condition of house; calculated as OverallQual*OverallCond
11	TotalBath	# of total bathrooms (including basement); calculated as FullBath + HalfBath + BsmtFullBath + BsmtHalfBath
12	Neigh1	Neighborhoods in quartile of highest average sale price (dummy var)
13	Neigh2	Neighborhoods in quartile of 2nd highest average sale price (dummy var)
14	Neigh3	Neighborhoods in quartile of 3rd highest average sale price (dummy var) (Neighborhoods in quartile with lowest average sale price is used as the basis of interpretation)
15	ExterEx_dum	Excellent quality exterior material
16	ExterGd_dum	Good quality exterior material (Low exterior quality (typical + fair + poor) is used as the basis of interpretation)
17	BsmtEx_dum	Excellent basement height
18	BsmtGd_dum	Good basement height (Low basement (typical + fair + poor) is used as the basis of interpretation)
19	KitEx_dum	Excellent quality kitchen
20	KitGd_dum	Good quality kitchen (Low kitchen quality (typical + fair + poor) is used as the basis of interpretation)
21	TotalSqftCalc	Total sq. ft. of house across above ground living area and basement; calculated as GrLivArea + BsmtFinSF1 + BsmtFinSF2

A correlation plot of these variables is below. Their linear relationship with SalePrice ranges from an absolute value correlation coefficient of 0.1-0.7. The correlation among the potential predictors is less than 0.6, helping to avoid multi-collinearity issues. The only exception is between TotalBath and TotalSqftCalc at 0.68. Since bathrooms are usually important factors for home buyers, this has been left in. We will evaluate in the fitted models whether it should be removed.



We will now use three automated variable selection methods, forward, backwards, and stepwise, to fit models predicting log(SalePrice). The summary outputs including their VIF values are below. Note that all these methods produced the same regression model, which is not always the case.

Forward Selection

```
Call:
lm(formula = logSalePrice ~ TotalSqftCalc + HouseAge + QualityIndex +
    GarageCars + neigh1 + neigh2 + BsmtEx_dum + LotArea + BedroomAbvGr +
    Fireplaces + LotFrontage + OpenPorchSF + ScreenPorch + ExterGd_dum +
    ExterEx_dum + neigh3 + KitEx_dum + KitGd_dum + BsmtGd_dum,
    data = train.clean)

Residuals:
     Min       1Q   Median       3Q      Max
-0.79036 -0.07038  0.00481  0.07493  0.46225

Coefficients:
                Estimate    Std. Error t value             Pr(>|t|)
(Intercept)   10.9813590841  0.0190705705 575.828 < 0.0000000000000002
TotalSqftCalc  0.0001633989  0.0000055243  29.578 < 0.0000000000000002
HouseAge      -0.0021939233  0.0001803425 -12.165 < 0.0000000000000002
QualityIndex   0.0105561112  0.0003841593  27.478 < 0.0000000000000002
GarageCars     0.0627179640  0.0051086044  12.277 < 0.0000000000000002
neigh1         0.1725438576  0.0141325569  12.209 < 0.0000000000000002
neigh2         0.1157315729  0.0114074717  10.145 < 0.0000000000000002
BsmtEx_dum     0.1079825055  0.0159325800   6.777 0.000000000016018377
LotArea        0.0000028320  0.0000003564   7.947 0.000000000000003163
BedroomAbvGr   0.0345961821  0.0036596607   9.453 < 0.0000000000000002
Fireplaces     0.0432237837  0.0052309922   8.263 0.000000000000000255
LotFrontage    0.0005123465  0.0000883299   5.800 0.000000007668320653
OpenPorchSF    0.0002149790  0.0000456072   4.714 0.000002599516342545
ScreenPorch    0.0002757768  0.0000516116   5.343 0.000000101618954839
ExterGd_dum    0.0507072301  0.0098699935   5.138 0.000000305337245635
ExterEx_dum    0.0975436949  0.0219795225   4.438 0.000009574544037426
neigh3         0.0468013280  0.0092388583   5.066 0.000000444101535112
KitEx_dum      0.0857276277  0.0165038233   5.194 0.000000226145738414
KitGd_dum      0.0366004540  0.0085076925   4.302 0.000017735041213210
BsmtGd_dum     0.0166462266  0.0093877396   1.773               0.0763

Residual standard error: 0.1267 on 2009 degrees of freedom
Multiple R-squared:  0.9004,    Adjusted R-squared:  0.8995
F-statistic:   956 on 19 and 2009 DF,  p-value: < 0.00000000000000022
```

	VIF
neigh1	4.030658
HouseAge	3.808547
neigh2	3.068828
BsmtEx_dum	2.823704
ExterGd_dum	2.741877
BsmtGd_dum	2.697846
KitEx_dum	2.369928
neigh3	2.245502
ExterEx_dum	2.227925
KitGd_dum	2.189165
TotalSqftCalc	2.043826
GarageCars	1.893798
QualityIndex	1.532299
Fireplaces	1.462224
BedroomAbvGr	1.181609
LotArea	1.164693
OpenPorchSF	1.147714
LotFrontage	1.101956
ScreenPorch	1.059989

Backward Selection

```
Call:
lm(formula = logSalePrice ~ LotFrontage + LotArea + BedroomAbvGr +
    Fireplaces + GarageCars + OpenPorchSF + ScreenPorch + HouseAge +
    QualityIndex + neigh1 + neigh2 + neigh3 + ExterEx_dum + ExterGd_dum
    BsmtEx_dum + BsmtGd_dum + KitEx_dum + KitGd_dum + TotalSqftCalc,
    data = train.clean)

Residuals:
     Min       1Q   Median       3Q      Max
-0.79036 -0.07038  0.00481  0.07493  0.46225

Coefficients:
                 Estimate    Std. Error t value            Pr(>|t|)
(Intercept)   10.9813590841  0.0190705705 575.828 < 0.0000000000000002
LotFrontage    0.0005123465  0.0000883299   5.800 0.000000007668320653
LotArea        0.0000028320  0.0000003564   7.947 0.000000000000003163
BedroomAbvGr   0.0345961821  0.0036596607   9.453 < 0.0000000000000002
Fireplaces     0.0432237837  0.0052309922   8.263 0.000000000000000255
GarageCars     0.0627179640  0.0051086044  12.277 < 0.0000000000000002
OpenPorchSF    0.0002149790  0.0000456072   4.714 0.000002599516342547
ScreenPorch    0.0002757768  0.0000516116   5.343 0.000000101618954839
HouseAge      -0.0021939233  0.0001803425 -12.165 < 0.0000000000000002
QualityIndex   0.0105561112  0.0003841593  27.478 < 0.0000000000000002
neigh1         0.1725438576  0.0141325569  12.209 < 0.0000000000000002
neigh2         0.1157315729  0.0114074717  10.145 < 0.0000000000000002
neigh3         0.0468013280  0.0092388583   5.066 0.000000444101535112
ExterEx_dum    0.0975436949  0.0219795225   4.438 0.000009574544037424
ExterGd_dum    0.0507072301  0.0098699935   5.138 0.000000305337245635
BsmtEx_dum     0.1079825055  0.0159325800   6.777 0.000000000016018377
BsmtGd_dum     0.0166462266  0.0093877396   1.773                0.0763
KitEx_dum      0.0857276277  0.0165038233   5.194 0.000000226145738414
KitGd_dum      0.0366004540  0.0085076925   4.302 0.000017735041213211
TotalSqftCalc  0.0001633989  0.0000055243  29.578 < 0.0000000000000002

Residual standard error: 0.1267 on 2009 degrees of freedom
Multiple R-squared:  0.9004,    Adjusted R-squared:  0.8995
F-statistic:   956 on 19 and 2009 DF,  p-value: < 0.00000000000000022
```

	VIF
neigh1	4.030658
HouseAge	3.808547
neigh2	3.068828
BsmtEx_dum	2.823704
ExterGd_dum	2.741877
BsmtGd_dum	2.697846
KitEx_dum	2.369928
neigh3	2.245502
ExterEx_dum	2.227925
KitGd_dum	2.189165
TotalSqftCalc	2.043826
GarageCars	1.893798
QualityIndex	1.532299
Fireplaces	1.462224
BedroomAbvGr	1.181609
LotArea	1.164693
OpenPorchSF	1.147714
LotFrontage	1.101956
ScreenPorch	1.059989

Stepwise Selection

```
Call:
lm(formula = logSalePrice ~ TotalSqftCalc + HouseAge + QualityIndex +
    GarageCars + neigh1 + neigh2 + BsmtEx_dum + LotArea + BedroomAbvGr +
    Fireplaces + LotFrontage + OpenPorchSF + ScreenPorch + ExterGd_dum +
    ExterEx_dum + neigh3 + KitEx_dum + KitGd_dum + BsmtGd_dum,
    data = train.clean)

Residuals:
     Min       1Q   Median       3Q      Max
-0.79036 -0.07038  0.00481  0.07493  0.46225

Coefficients:
                Estimate    Std. Error t value             Pr(>|t|)
(Intercept)   10.9813590841  0.0190705705 575.828 < 0.0000000000000002
TotalSqftCalc  0.0001633989  0.0000055243  29.578 < 0.0000000000000002
HouseAge      -0.0021939233  0.0001803425 -12.165 < 0.0000000000000002
QualityIndex   0.0105561112  0.0003841593  27.478 < 0.0000000000000002
GarageCars     0.0627179640  0.0051086044  12.277 < 0.0000000000000002
neigh1         0.1725438576  0.0141325569  12.209 < 0.0000000000000002
neigh2         0.1157315729  0.0114074717  10.145 < 0.0000000000000002
BsmtEx_dum     0.1079825055  0.0159325800   6.777 0.000000000016018377
LotArea        0.0000028320  0.0000003564   7.947 0.000000000000003163
BedroomAbvGr   0.0345961821  0.0036596607   9.453 < 0.0000000000000002
Fireplaces     0.0432237837  0.0052309922   8.263 0.000000000000000255
LotFrontage    0.0005123465  0.0000883299   5.800 0.000000007668320653
OpenPorchSF    0.0002149790  0.0000456072   4.714 0.000002599516342545
ScreenPorch    0.0002757768  0.0000516116   5.343 0.000000101618954839
ExterGd_dum    0.0507072301  0.0098699935   5.138 0.000000305337245635
ExterEx_dum    0.0975436949  0.0219795225   4.438 0.000009574544037426
neigh3         0.0468013280  0.0092388583   5.066 0.000000444101535112
KitEx_dum      0.0857276277  0.0165038233   5.194 0.000000226145738414
KitGd_dum      0.0366004540  0.0085076925   4.302 0.000017735041213210
BsmtGd_dum     0.0166462266  0.0093877396   1.773               0.0763

Residual standard error: 0.1267 on 2009 degrees of freedom
Multiple R-squared:  0.9004,    Adjusted R-squared:  0.8995
F-statistic:   956 on 19 and 2009 DF,  p-value: < 0.00000000000000022
```

	VIF
neigh1	4.030658
HouseAge	3.808547
neigh2	3.068828
BsmtEx_dum	2.823704
ExterGd_dum	2.741877
BsmtGd_dum	2.697846
KitEx_dum	2.369928
neigh3	2.245502
ExterEx_dum	2.227925
KitGd_dum	2.189165
TotalSqftCalc	2.043826
GarageCars	1.893798
QualityIndex	1.532299
Fireplaces	1.462224
BedroomAbvGr	1.181609
LotArea	1.164693
OpenPorchSF	1.147714
LotFrontage	1.101956
ScreenPorch	1.059989

The three methods produced the same model with the equation:
Log(SalePrice) = 10.981 + 0.0002*TotalSqftCalc – 0.002*HouseAge + 0.011*QualityIndex + 0.063*GarageCars + 0.173*neigh1 + 0.116*neigh2 + 0.108*BsmtEx_dum + 0.000003*LotArea + 0.035*BedroomAbvGr + 0.043*Fireplaces + 0.0005*LotFrontage + 0.0002*OpenPorchSF + 0.0002*ScreenPorch + 0.051*ExterGd_dum + 0.098*ExterEx_dum + 0.047*neigh3 + 0.086*KitEx_dum + 0.037*KitGd_dum + 0.017*BsmtGd_dum

WoodDeckSF and TotalBath are not included in the model as the algorithm found them not valuable. To interpret the coefficients, they need to be exponentiated and signify a percentage change in sale price. For example, the coefficient for QualityIndex is 0.011. Taking the exponent of this and subtracting 1 tells us for every unit increase in the quality index, sale price increases 1%. All the coefficients are positive except for HouseAge, which aligns logically. As a house becomes older, its sale price is expected to decrease. Many coefficients are noticeably small in magnitude, even after being exponentialized. Given this, we can likely drop some variables without significantly sacrificing model fit and accuracy. This will also better achieve parsimony. We will investigate this later.

This selected model has a strong R-squared, explaining 90% of the variation in log(SalePrice). The omnibus F-test produces a p-value close to 0, so we can reject the null hypothesis that all Beta coefficients are equal to 0. The alternative hypothesis is that at least one Beta coefficient is non-zero, meaning this model significantly predicts sale price.

Almost all coefficients are also significant at the 95% level based on the individual t-statistic and p-values. In these cases, we can reject the null hypothesis that their Beta coefficients equal 0. The alternative hypothesis for each test is the Beta coefficient is non-zero, indicating they are predictors of sale price. The exception to this is BsmtGd_dum, but this is still significant at the 90% level.

The VIF values are all less than 10, indicating no concerns of multi-collinearity. This model looks valid so far. Note that Neigh1 has the highest VIF, but VIF values for indicator variables are generally not concerning as they are for classification. There is technically a separate model for each level, even though there is just one regression equation.

I also created a model appropriately called “junk” as a comparison of a poorly fit model.

Junk Model

```
Call:
lm(formula = logSalePrice ~ OverallQual + OverallCond + QualityIndex
    GrLivArea + TotalSqftCalc, data = train.df)

Residuals:
     Min       1Q   Median       3Q      Max
-1.08501 -0.09027  0.01468  0.10055  0.51330

Coefficients:
                Estimate   Std. Error t value             Pr(>|t|)
(Intercept)    9.884444453  0.075470165 130.972 < 0.0000000000000002
OverallQual    0.255575714  0.012886911  19.832 < 0.0000000000000002
OverallCond    0.115909173  0.014013167   8.271 0.000000000000000237
QualityIndex  -0.017750832  0.002437082  -7.284 0.000000000000463255
GrLivArea      0.000116714  0.000012099   9.647 < 0.0000000000000002
TotalSqftCalc  0.000180893  0.000008014  22.571 < 0.0000000000000002

Residual standard error: 0.1633 on 2023 degrees of freedom
Multiple R-squared:  0.8336,    Adjusted R-squared:  0.8331
F-statistic:  2026 on 5 and 2023 DF,  p-value: < 0.00000000000000022
```

	VIF
QualityIndex	37.153576
OverallQual	25.070836
OverallCond	18.031618
GrLivArea	2.699843
TotalSqftCalc	2.591600

Highly correlated variables are purposely included in this as a bad example. QualityIndex is the product of OverallQual and OverallCond, so they obviously have a strong relationship. Similarly, GrLivArea is a component of TotalSqftCalc, so both should not be included together. Highly correlated variables provide overlapping information thus should be dropped except for one.

The omnibus F-test shows the model still significantly predicts sale price, and the individual t-tests indicate each variable is a significant predictor. The R-squared is also substantial at 83%, though not as high as the prior better fitting models. However, the VIF values for overall quality and condition are very large at 18+. Another red flag is the negative coefficient of QualityIndex. This suggests sale price goes down as quality increases, which logically does not make sense. This model is not reliable, especially for inferences, hence "junk."

Below is a comparison summary of the models based on in-sample fit.

model	adj_r_squared	AIC	BIC	MSE	MAE
Forward	0.90	-2602.34	-2484.41	0.016	0.095
Backward	0.90	-2602.34	-2484.41	0.016	0.095
Stepwise	0.90	-2602.34	-2484.41	0.016	0.095
Junk	0.83	-1588.14	-1548.83	0.027	0.122

Since the three methods selected the same model, there is not a comparison or choice to be made among the models. These criteria do confirm this model is a better fit than the junk model. It explains 7% more variability in sale price (adjusted for number of predictors). It also has lower information criteria, MSE, and MAE, all indicators of lower error in the model's fitted values. Note that if the selection methods produced different models, a model may not always have the same ranking across all metrics. AIC and BIC impose heavier penalties on more complex models (i.e. more predictors) than adjusted R-squared, MSE, and MAE.

(4) Predictive Accuracy

We will now evaluate the models' prediction accuracy using the test data set, key for utilizing models for predictive purposes.

model	MSE	MAE
Forward	0.017	0.094
Backward	0.017	0.094
Stepwise	0.017	0.094
Junk	0.025	0.120

Again, the model built with automatic variable selection fit better than the junk model, depicted by the lower predictive errors. In terms of whether to use MSE or MAE as the criterion, there is not necessarily one that is better than the other. MSE is more sensitive to large errors since they are squared, though less so in the log space. MAE is more robust when there are outliers and a bit easier to understand. Typically, both metrics show consistent findings—a model with the best fit based on MSE will also be the best on MAE.

Compared to the training data, the test sample's MAE is slightly lower, i.e. has smaller residuals. This indicates the out-of-sample sale prices are closer to the average predicted by the model and have less variance. It also suggests the model generalizes well or could be by chance, though is evidence that overfitting is not occurring.

MAPE (mean absolute percentage error) is another helpful measurement that gives a sense of the error's magnitude. The automatic selected model has a MAPE of 0.8% on the test and training data. In other words, the error of the prediction is just 0.8% of the actual value. This very low error further supports the model's strong fit and accuracy. For comparison, the junk model has a MAPE of 1%.

(5) Operational Validation

To validate the automatically selected model from a business perspective, we will look at how far off the predicted sale price is from the actual sale price for every observation in the training and test set. This is similar to MAPE above but viewed as a distribution instead of a single mean value. The output for the forward selection model is below, which is also the same as the backward and stepwise models.

Training Data – log scale

forward.PredictionGrade	Freq
Grade 1: [0,0.10]	1

Test Data – log scale

forward.testPredictionGrade	Freq
Grade 1: [0,0.10]	1

Based on the log-transformed space, the model has very high accuracy, confirming findings form the MAPE. For both the training and test data, all home sale prices (100%) were predicted by the model within 10% of the actual value.

However, we should transform the data back to normal scale since in reality, business managers are interested in sale price rather than log(SalePrice).

Training Data – original scale

fwdexp.PredictionGrade	Freq
Grade 1: [0,0.10]	0.6456382
Grade 2: (0.10,0.15]	0.1596846
Grade 3: (0.15,0.25]	0.1409561
Grade 4: (0.25+]	0.0537210

Test Data – original scale

fwdexp.PredictionGrade	Freq
Grade 1: [0,0.10]	0.6612903
Grade 2: (0.10,0.15]	0.1532258
Grade 3: (0.15,0.25]	0.1301843
Grade 4: (0.25+]	0.0552995

The prediction accuracy does not seem as favorable in the actual, non-log transformed scale, but it is still a useful model that business managers can trust. 64% of the training data is predicted within 10% of the actual value (Grade 1), and similarly, 66% of the test data is predicted with Grade 1 accuracy. This would be considered "underwriting quality" based on the GSEs definition of being accurate within 10% more than half the time.

For a comparison, I also validated the junk data set. Interestingly, it also predicts 100% of sale prices within 10% of the actual value for both the training and test data. After taking the exponent to view the more accurate non-log scale, the training and test data have 52% and 54% Grade 1, respectively. This is substantially lower by ~10%, expectedly given the high collinear variables. Technically, it is still underwriting quality.

Training Data – log scale

junk.PredictionGrade	Freq
Grade 1: [0,0.10]	1

Test Data – log scale

junk.testPredictionGrade	Freq
Grade 1: [0,0.10]	1

Training Data – original scale

junkexp.PredictionGrade	Freq
Grade 1: [0,0.10]	0.5234105
Grade 2: (0.10,0.15]	0.1936915
Grade 3: (0.15,0.25]	0.1769345
Grade 4: (0.25+]	0.1059635

Test Data – original scale

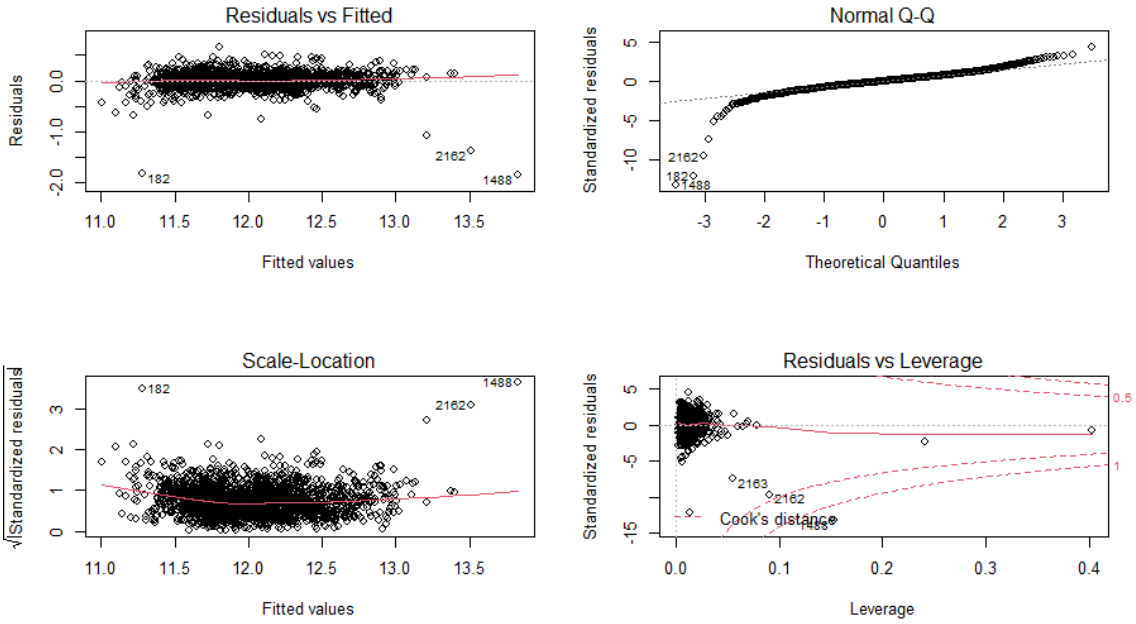
junkexp.PredictionGrade	Freq
Grade 1: [0,0.10]	0.5403226
Grade 2: (0.10,0.15]	0.1820276
Grade 3: (0.15,0.25]	0.1808756
junktest.testPCT 4: (0.25+]	0.0967742

(6) Cleaning Up the Final Model

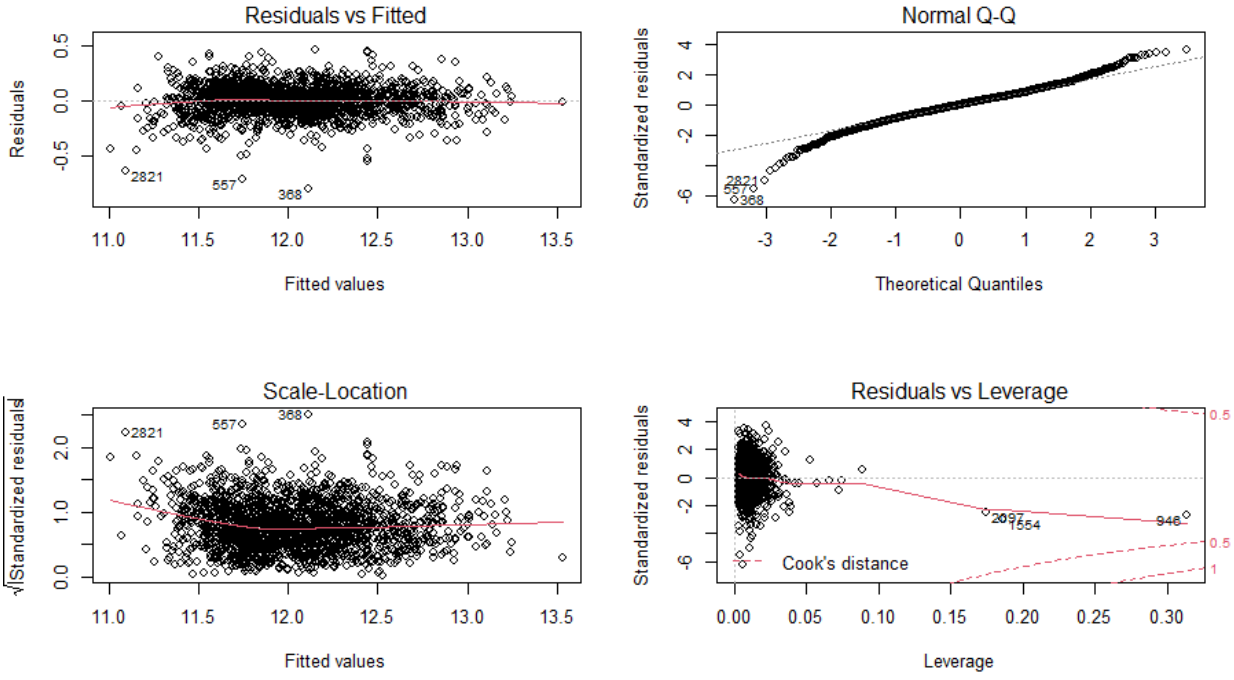
From what we have assessed so far, the model developed from the automated selection process seems to be a good fit. However, some additional checks and clean ups are needed. First, let us examine the diagnostic plots and underlying model assumptions.

In an initial fit of the model, the residual and leverage plots indicated a few influential points. An examination showed these are homes with abnormal and partial sale conditions. Given their atypical nature, these observations have been excluded. The process from above, including the automatic variable selection, was repeated on the data set with these points removed. This slightly improved the model fit and predictive accuracy, as R-squared increased by 4% and MAPE on the training data by 0.5%.

All outputs shown earlier are based off this cleaned data set. The outputs for the original model are not displayed except for the diagnostics below depicting the influential points.



We will now check the diagnostic plots for this latest model that has the influential observations cleaned.



There are a few outliers and points with slight leverage but nothing alarming that requires adjustment. All points now fall within the Cook's distance threshold. The residuals are close to a random scatter. Their variance does slightly decrease for larger fitted values, but there are no concerning violations of homoscedasticity. The Q-Q plot mainly follows normality except in the tails. While the model overall has

high predictive accuracy, it is less reliable dealing with homes with extreme price range. It tends to over predict the lowest priced homes and under predict homes in the highest quantile of price.

Next, let us take a closer look at the coefficients. None of their signs are reversed from theoretical expectations. All of them are expectedly positive except for HouseAge, the only factor that would decrease sale price as it increases. This plus the VIF values viewed earlier indicate multicollinearity is not an issue.

However, we can likely drop some variables that do not have much explanatory power. While all coefficients are statistically significant, some are very small in magnitude. I tested dropping variables, first focusing on those with the smallest coefficients, and compared how R-squared and adjusted R-squared changes.

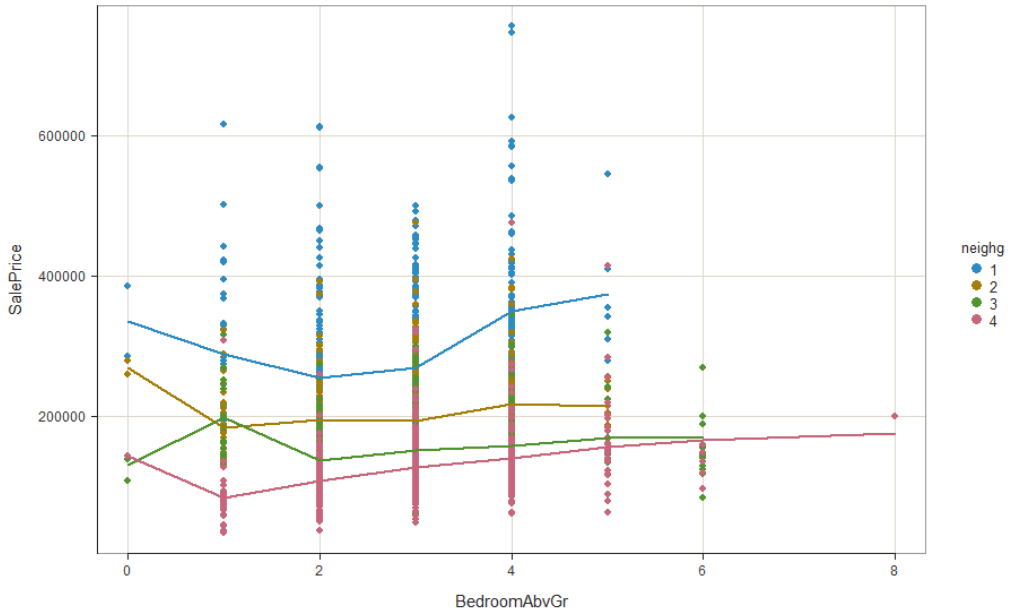
In the end, I dropped LotArea, LotFrontage, OpenPorchSF, ScreenPorch, and the dummy variables for basement and exterior quality, BsmtEx_dum, BsmtGd_dum, ExterEx_dum, ExterGd_dum. These had minimal contribution to the adjusted R-squared (<0.004). Kitchen quality was also recoded to be binary for excellent ratings vs. not excellent, as parsing out good quality did not add much value. The basis of interpretation are homes with non-excellent kitchens. The output for this model is below.

```
Call:
lm(formula = logSalePrice ~ TotalSqftCalc + HouseAge + QualityIndex +
    GarageCars + neigh1 + neigh2 + neigh3 + BedroomAbvGr + Fireplaces +
    KitEx_dum, data = train.clean)

Residuals:
     Min       1Q   Median       3Q      Max
-0.84901 -0.08215  0.00383  0.07979  0.59529

Coefficients:
                Estimate   Std. Error t value             Pr(>|t|)
(Intercept)   11.000627702  0.019201744 572.897 < 0.0000000000000002
TotalSqftCalc  0.000185221  0.000005736  32.292 < 0.0000000000000002
HouseAge      -0.002729032  0.000157837 -17.290 < 0.0000000000000002
QualityIndex   0.011750496  0.000386381  30.412 < 0.0000000000000002
GarageCars     0.078460806  0.005392471  14.550 < 0.0000000000000002
neigh1         0.221803397  0.014769278  15.018 < 0.0000000000000002
neigh2         0.123135710  0.012013853  10.249 < 0.0000000000000002
neigh3         0.038209654  0.009611064   3.976            0.0000727
BedroomAbvGr   0.034309433  0.003861938   8.884 < 0.0000000000000002
Fireplaces     0.048407839  0.005550221   8.722 < 0.0000000000000002
KitEx_dum      0.114836333  0.013250865   8.666 < 0.0000000000000002

Residual standard error: 0.1364 on 2018 degrees of freedom
Multiple R-squared:  0.8841,    Adjusted R-squared:  0.8835
F-statistic:  1539 on 10 and 2018 DF,  p-value: < 0.00000000000000022
```

While the model was reduced by 9 predictors, almost half of that in the automatically selected model, R-squared and adjusted R-squared decreased by only 1.6% to 88%, still strong explanatory power. The residual standard error increased by just 0.0097. All variables are statistically significant predictors at 95% significance, and those with very small coefficients and small contribution to R-squared have been removed. Parsimony is now better achieved without substantial sacrifice, allowing the model to focus on the most important variables, while making it easier to understand and manage.

Next, I checked for interaction between the quantitative and categorical variables, neighborhood and kitchen quality. As a reminder, Neighborhood is dummy coded into 4 groups/quartiles based on mean sale price in the given data, where the most affordable group is the baseline. Kitchen is dummy coded based on those rated excellent vs. other ratings, where the latter is the basis of interpretation.

Neighborhood shows significant interaction with many predictors in the model. For example, sales price varies differently depending on both the neighborhood and the number bedrooms. Neighborhood 1 prices dip for 2-3 bedroom homes then see a huge jump for 4-5 bedrooms. Other neighborhoods see a more steady increase in price as bedrooms increase.



To formally test this, interaction terms between Neighborhood and Bedrooms were then created. These were added to the model as the "full model" and compared to the model without, "reduced model", with a nested F-test.

Null hypothesis: $\beta_{23} = \beta_{24} = \beta_{25} = 0$

Alternative hypothesis: At least one $\beta \neq 0$

The Betas are the coefficients for the interaction between Neighborhood and BedroomAbvGr.

```
Analysis of Variance Table

Model 1: logSalePrice ~ TotalSqftCalc + HouseAge + QualityIndex + GarageCars +
    neigh1 + neigh2 + neigh3 + BedroomAbvGr + Fireplaces + KitEx_dum +
    neigh1sqft + neigh2sqft + neigh3sqft + neigh1age + neigh2age +
    neigh3age + neigh1qi + neigh2qi + neigh3qi + neigh1cars +
    neigh2cars + neigh3cars + neigh1bd + neigh2bd + neigh3bd
Model 2: logSalePrice ~ TotalSqftCalc + HouseAge + QualityIndex + GarageCars +
    neigh1 + neigh2 + neigh3 + BedroomAbvGr + Fireplaces + KitEx_dum +
    neigh1sqft + neigh2sqft + neigh3sqft + neigh1age + neigh2age +
    neigh3age + neigh1qi + neigh2qi + neigh3qi + neigh1cars +
    neigh2cars + neigh3cars
  Res.Df    RSS Df Sum of Sq     F Pr(>F)
1   2003 35.044
2   2006 35.211 -3  -0.16712 3.184  0.023
```

The p-value is 2.3%, so we can reject the null hypothesis under a 5% alpha level. This means there is significant interaction between Neighborhood and Bedrooms. This exercise was done for the various combinations of categorical and quantitative variables.

In the end, interaction was found between Neighborhood and TotalSqftCalc, HouseAge, QualityIndex, GarageCars, and BedroomsAbvGr. Additionally, Kitchen Quality and Bedrooms generated significant interaction. Inclusion of all these interactions resulted in this model.

```
Call:
lm(formula = logSalePrice ~ TotalSqftCalc + HouseAge + QualityIndex +
    GarageCars + neigh1 + neigh2 + neigh3 + BedroomAbvGr + Fireplaces +
    KitEx_dum + neigh1sqft + neigh2sqft + neigh3sqft + neigh1age +
    neigh2age + neigh3age + neigh1qi + neigh2qi + neigh3qi +
    neigh1cars + neigh2cars + neigh3cars + neigh1bd + neigh2bd +
    neigh3bd + kitbd, data = train.df)

Residuals:
     Min       1Q   Median       3Q      Max
-0.84820 -0.08242  0.00071  0.07524  0.67891

Coefficients:
               Estimate  Std. Error t value             Pr(>|t|)
(Intercept)   10.85387911  0.02605546 416.568 < 0.0000000000000002
TotalSqftCalc  0.00020603  0.00001364  15.109 < 0.0000000000000002
HouseAge      -0.00186778  0.00023254  -8.032  0.00000000000000162
QualityIndex   0.01218350  0.00055844  21.817 < 0.0000000000000002
GarageCars     0.08409835  0.00847446   9.924 < 0.0000000000000002
neigh1         0.30245295  0.05791032   5.223  0.00000019454827056
neigh2         0.33296981  0.05307694   6.273  0.00000000043185137
neigh3         0.41101060  0.04539205   9.055 < 0.0000000000000002
BedroomAbvGr   0.04573729  0.00744080   6.147  0.00000000095143005
Fireplaces     0.05523960  0.00546695  10.104 < 0.0000000000000002
KitEx_dum      0.20025925  0.04150287   4.825  0.00000150440798881
neigh1sqft    -0.00002083  0.00001646  -1.266              0.20574
neigh2sqft    -0.00002337  0.00001693  -1.381              0.16750
neigh3sqft    -0.00001986  0.00001747  -1.137              0.25586
neigh1age     -0.00633746  0.00089308  -7.096  0.00000000000177236
neigh2age     -0.00106039  0.00038528  -2.752              0.00597
neigh3age     -0.00209957  0.00040369  -5.201  0.00000021848598107
neigh1qi      -0.00046975  0.00137778  -0.341              0.73318
neigh2qi      -0.00063368  0.00104740  -0.605              0.54524
neigh3qi      -0.00273649  0.00089820  -3.047              0.00234
neigh1cars     0.02789124  0.01910692   1.460              0.14452
neigh2cars    -0.01496489  0.01765479  -0.848              0.39674
neigh3cars    -0.03900827  0.01210782  -3.222              0.00129
neigh1bd      -0.00313028  0.01237828  -0.253              0.80038
neigh2bd      -0.02241102  0.01114958  -2.010              0.04456
neigh3bd      -0.03074657  0.01031252  -2.981              0.00290
kitbd         -0.04262799  0.01411336  -3.020              0.00256

Residual standard error: 0.132 on 2002 degrees of freedom
Multiple R-squared:  0.8923,    Adjusted R-squared:  0.891
F-statistic: 638.3 on 26 and 2002 DF,  p-value: < 0.00000000000000022
```

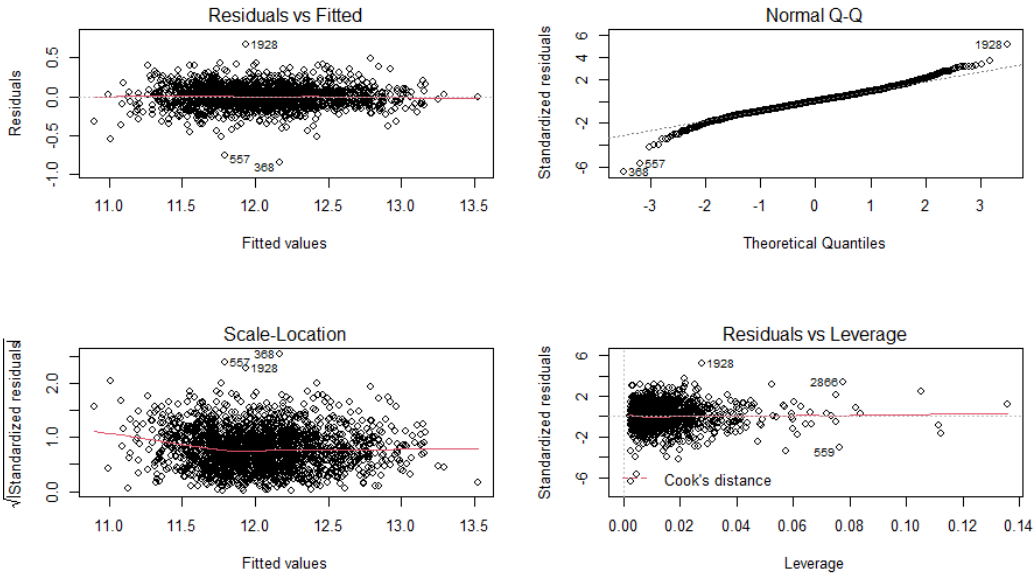
Interestingly, all three interaction terms for Neighborhood and Total Square Feet are not significant predictors despite being significant interaction terms based on the F-test. These will be removed resulting in the model below. Other interactions show lack of significance with certain neighborhoods (e.g. neighborhood 1 and 2 with garage cars), but since the other neighborhoods are significant with these same predictors (e.g. neighborhood 3 with garage cars), these have been left in. Overall, R-squared and the residual standard error are slightly improved with the addition of the interaction terms.

Final Model

```
Call:
lm(formula = logSalePrice ~ TotalSqftCalc + HouseAge + QualityIndex +
    GarageCars + neigh1 + neigh2 + neigh3 + BedroomAbvGr + Fireplaces +
    KitEx_dum + neigh1age + neigh2age + neigh3age + neigh1qi +
    neigh2qi + neigh3qi + neigh1cars + neigh2cars + neigh3cars +
    neigh1bd + neigh2bd + neigh3bd + kitbd, data = train.df)

Residuals:
     Min       1Q   Median       3Q      Max
-0.84860 -0.08183  0.00048  0.07545  0.67719

Coefficients:
               Estimate  Std. Error t value             Pr(>|t|)
(Intercept)  10.86117519  0.02556478 424.849 < 0.0000000000000002
TotalSqftCalc  0.00018807  0.00000575  32.707 < 0.0000000000000002
HouseAge      -0.00188783  0.00023208  -8.134 0.000000000000000718
QualityIndex   0.01231176  0.00055122  22.335 < 0.0000000000000002
GarageCars     0.08667599  0.00828203  10.466 < 0.0000000000000002
neigh1         0.29694599  0.05768189   5.148 0.000000289028536858
neigh2         0.32438644  0.05273794   6.151 0.000000000927414069
neigh3         0.40232812  0.04438847   9.064 < 0.0000000000000002
BedroomAbvGr   0.05137595  0.00634439   8.098 0.000000000000000961
Fireplaces     0.05507306  0.00542725  10.147 < 0.0000000000000002
KitEx_dum      0.19706518  0.04138365   4.762 0.000002055038429032
neigh1age     -0.00640146  0.00085997  -7.444 0.000000000000144412
neigh2age     -0.00105671  0.00038404  -2.752              0.005984
neigh3age     -0.00207379  0.00040222  -5.156 0.000000277389268896
neigh1qi      -0.00067138  0.00135252  -0.496              0.619674
neigh2qi      -0.00086601  0.00102421  -0.846              0.397910
neigh3qi      -0.00288194  0.00088610  -3.252              0.001163
neigh1cars     0.02356516  0.01821619   1.294              0.195939
neigh2cars    -0.01946319  0.01718423  -1.133              0.257510
neigh3cars    -0.04201913  0.01164926  -3.607              0.000317
neigh1bd      -0.00913651  0.01166587  -0.783              0.433612
neigh2bd      -0.02878902  0.01026355  -2.805              0.005081
neigh3bd      -0.03676348  0.00916218  -4.013 0.000062279049752104
kitbd         -0.04171951  0.01409628  -2.960              0.003116

Residual standard error: 0.132 on 2005 degrees of freedom
Multiple R-squared:  0.8922,    Adjusted R-squared:  0.891
F-statistic: 721.7 on 23 and 2005 DF,  p-value: < 0.00000000000000022
```

This is potentially the final model. The R-squared is strong, explaining 89% of the variability in sale price. The omnibus F-test is significant with a p-value close to null. This means we can reject the null hypothesis that all Betas coefficients are equal to 0. The alternative is that at least one Beta is non-zero. All coefficients are also significant based on the individual t-test (null hypothesis: Beta for individual coefficient is 0, alternative hypothesis: Beta is not equal to 0). The exception to this is some interaction terms. These do not need to be used if desired but have been kept since some neighborhoods with those variables do have significant interactions.

We next need to conduct the model diagnostics and goodness-of-fit checks to confirm its usability.



These plots support the model follows homoscedasticity, linearity, and normality assumptions. Only the tail ends diverge a bit from normal distribution, indicating predictions for very low or high-priced homes have higher error. There are no concerning outliers or influential points that need to be removed.

Let us also view how this model compares to the prior ones tested.

Training Data

model	adj_r_squared	AIC	BIC	MSE	MAE
Forward	0.90	-2602.34	-2484.41	0.016	0.095
Backward	0.90	-2602.34	-2484.41	0.016	0.095
Stepwise	0.90	-2602.34	-2484.41	0.016	0.095
Junk	0.83	-1588.14	-1548.83	0.027	0.122
Final Model	0.89	-2434.09	-2293.71	0.017	0.100

Test Data

model	MSE	MAE
Forward	0.017	0.094
Backward	0.017	0.094
Stepwise	0.017	0.094
Junk	0.025	0.120
Final Model	0.018	0.099

As expected, the adjusted R-squared is slightly lower by 1% than the automatic variable selection model since that contained many more variables. The error of the final model is slightly higher but still minimal. Both metrics are much stronger than the junk model.

In terms of operational validation, the final model predicts 60% of training and 61% of test sale prices within 10% of the actual price. This is a few percentage points lower than the automatic variable selected models (65%, 66% for training and test models, respectively). However, this trade-off has achieved better parsimony in the final model.

Training Data – original scale (not log scale)

m2exp.PredictionGrade	Freq
Grade 1: [0,0.10]	0.5953672
Grade 2: (0.10,0.15]	0.1946772
Grade 3: (0.15,0.25]	0.1493346
Grade 4: (0.25+]	0.0606210

Test Data – original scale (not log scale)

m2exp.PredictionGrade	Freq
Grade 1: [0,0.10]	0.6129032
Grade 2: (0.10,0.15]	0.1774194
Grade 3: (0.15,0.25]	0.1497696
Grade 4: (0.25+]	0.0599078

Another metric helpful for business evaluation is the root mean squared error (RMSE). After transforming this back to original scale, the RMSE is $25,916 on the training data and $25,301 on the test data. This means the average predictive error within one standard deviation is $25K. Dividing this by the mean sale price of $181,892 gives a coefficient of variation of 14%. This is the average error for a predicted sale price. We can discuss with partners whether this is appropriate or too large. From the information we have so far, this model follows all assumptions and can be brought forward to the business for discussion and approval for use.

Conclusion

In summary, the model arrived upon to predict residential home sale prices is:
log(SalePrice) = 10.861 + 0.0002*TotalSqftCalc – 0.002*HouseAge + 0.012*QualityIndex + 0.087*GarageCars + 0.051*BedroomAbvGr + 0.055*Fireplaces + 0.297*neigh1 + 0.324*neigh2 + 0.402*neigh3 + 0.197*KitEx_dum – 0.006*neigh1age – 0.001*neigh2age – 0.002*neigh3age – 0.0007*neigh1qi – 0.0009*neigh2qi – 0.003*neigh3qi + 0.024*neigh1cars – 0.019*neigh2cars – 0.042*neigh3cars – 0.009*neigh1bd – 0.029*neigh2bd – 0.037*neigh3bd – 0.042*kitbd

This model is log-linear where the response variable is natural log of SalePrice, so we should take the exponent of the coefficients for inferential purposes. Exponentializing the coefficients and subtracting 1 tells us the percent change in price for every unit increase in that predictor. Below is a table summarizing this value in the “% of price change” column. The column “1-unit price change” takes this percentage and multiplies is by the average home price as a reference comparison. Essentially, this table shows the impact of a one-unit change in each explanatory variable on the average home sale price.

Variable	% of price change (%)	1-unit price change ($)
TotalSqftCalc	0.02	34.21
HouseAge	-0.19	-343.06
QualityIndex	1.24	2253.25
GarageCars	9.05	16469.09
neigh1	34.57	62887.80
neigh2	38.32	69697.67
neigh3	49.53	90091.42
BedroomAbvGr	5.27	9589.09
Fireplaces	5.66	10298.32
KitEx_dum	21.78	39620.32
neigh1age	-0.64	-1160.66
neigh2age	-0.11	-192.11
neigh3age	-0.21	-376.81
neigh1qi	-0.07	-122.08
neigh2qi	-0.09	-157.45
neigh3qi	-0.29	-523.45
neigh1cars	2.38	4337.22
neigh2cars	-1.93	-3505.97
neigh3cars	-4.11	-7484.59
neigh1bd	-0.91	-1654.29
neigh2bd	-2.84	-5161.83
neigh3bd	-3.61	-6565.56
kitbd	-4.09	-7432.33

As an example, every year the house ages, average sale price decreases $343, all other factors held constant. The largest differentiator is the neighborhood the home is in. This confirms the well-known importance of location. A home in neighborhood group 1 has a $62,888 higher value than the baseline, neighborhood group 4, all else being equal. Kitchens that are excellent quality can increase the price by $39,620 compared to kitchens with other ratings. Fireplaces can also add value averaging $10,298 per fireplace for a home in Ames, Iowa. Given this, it may be worth investing in kitchen remodels and adding a fireplace.

The thirteen interaction terms at the end indicate the change in slope compared to the basis of interpretation. For instance, neighborhood group 1's slope for HouseAge is $1,161 lower than the baseline neighborhood 4, which is -$343. In other words, the slope for neighborhood 1 is -$1,504 so is much more impacted by aging.

This model check outs in terms of model assumptions (normality, homoscedasticity, linearity), omnibus F-test, and individual coefficient t-tests. In terms of goodness-of-fit, this model explains 89% of the variation in sale price. Its average predictive error is +/-$25,935 for in-sample and +/-$25,271 for out of sample. This is a 14% average error. Another way to look at its prediction accuracy, 60% of the training set's sale prices and 61% of the test holdout were predicted within 10% of the actual value. If business partners deem this sufficient for prediction, we can move forward with this model.

(7) For Reflection / Conclusions

Working on this model over multiple weeks was valuable to gain deep knowledge of the data, test and compare various prediction models of sale price, and most importantly go through the iterative, thorough process of model building.

I came across multiple challenges while building this model. A main one was deciding which predictors to include given the large number of variables in this Ames data set. Many of them strongly correlate, so I had to be cautious about multi-collinearity. It also contained a number of categorical predictors. These required careful consideration for model inclusion since they can lead to multiplicatively more potential interactions. I also pondered whether to treat ordinal variables as numeric or categorical, but I ultimately decided on the latter given their nonlinear nature.

Fortunately, there are many tools to help select predictors throughout the process including EDA for an initial idea, automatic variable selection, comparing goodness-of-fit metrics, and VIF values. However, deciding which variables to keep in the end is somewhat subjective. For instance, the change in R-squared from a single variable may seem important to me but may not to another modeler. This is also why it is critical to keep in mind specific business objectives (not given for this project e.g. a maximum predictive error).

Another challenge of this data set is the skew in sale price that required a log transformation. This made various outputs such as residual standard errors, MSE, and coefficients less straightforward to interpret. I had to become acquainted with working in the log space and doing a couple extra steps to convert it back. This also changed the operational validation. In log space, it appeared that 100% of sale prices were predicted within 10% of the actual value, due in part to the "squished" scale. However, in the normal scale, this dropped down to the 60% range. It is important to keep in mind ultimately the business objective is to predict sale price, not the log of it. The difficulty in interpretating the models were compounded by the number of variables, including a mix of numeric, categorical, and interaction terms.

Some other challenges related to subjective aspects, due largely to my limited experience modeling. This included dealing with outliers and deciding when the EDA and model chosen were good enough. More outliers or influential points could have been removed, which would result in a model with a better fit, but these observations may reflect reality. Similarly, is an R-squared of 88% sufficient, or should we aim for 90%? Again, business objectives are important to consider.

If the final model's predictive accuracy is not sufficient, it can be improved by further defining the population of interest for the business. While I had only removed non-residential properties from the data set, perhaps the audience wants to focus on just single-family homes or those sold under normal conditions. A new model can then better fit this narrower set of data. More data points can also be gathered to better train and inform the model. Prediction accuracy can also be improved by adding more predictors. The full model had lower errors than the final model. However, I had trimmed variables that had minimal contribution in explaining sale price to balance parsimony. If the business managers are fine with adding more predictors to gain predictive accuracy, those can be added back in. We do have to be careful about adding too many variables, which could lead to overfitting and actually lower predictive accuracy on new data.

Parsimony is important to consider. The final model should not be too simple nor too complex. The former can be less accurate while the latter can be less precise leading to more predictive error. Parsimony is especially important if the model is used for inferential purposes. The coefficients are

essential to determine how changing one factor affects the response variable. A simpler model has lower variation on the estimates. Additionally, there comes a point where adding more variables contributes little value. For prediction, a more complex model can be passable as long as it meets the business threshold for predictive error. Echoing my earlier statements, business objectives are critical to finding a useful model.