Analysis for Century 21 Ames

Vo Nguyen | Santiago Gutierrez

Santiago's Github Page: https://santigtz95.github.io
Vo Nguyen's Github Page: https://vochannguyen.github.io/

Introduction

As requested by Century 21 Ames, we have analyzed nearly 3,000 homes with about 80 categorical variables that describe residential homes in Ames, Iowa in order to build and test predictive models that allows us to estimate home prices.

In this report, we begin by studying and discussing the effects that Gross Living Area has on Sales Price for three specific Neighborhoods in Ames. We do so by finding linear relationships between variables, building a model, and then tuning our model to be more accurate.

Additionally, using a multitude of variables, we build predictive models for sales price and then analyze the accuracy of each to determine the best fitting model. We use four different variable selection methods to decide which variables were of importance and used those to build the four models.

Data Description

The data used in this analysis has been provided by Dean De Cock through Kaggle. The data describes the sale of individual homes in Ames, Iowa between 2006 and 2010. The data is divided between two sets – a training set for building our models and a test set used to test the models. Our predictive models are built using 1,460 observations and specific variables (of the 79 provided) from the training data set. We test our models using additional information from 1,458 homes in the Ames area to predict the sales prices and examine the accuracy.

Question 1 Analysis

The Problem
Century 21 Ames only sells houses in the NAmes, Edwards, and BrkSide neighborhoods and would like to determine how the sales price of homes is related to square footage per 100 sq ft. of living area (GrLivArea) and if the relationship depends on which neighborhood the house is located in.

Build and Fit the Model
Assuming independent observations, we begin by investigating to see if there exists a linear relationship between our primary explanatory variable, GrLivArea, and Sales Price. By creating a simple scatter plot, we are able to see what appears to be a pretty positive, linear relationship - see Plot 1.1 [Appendix]. There is also some evidence of outliers, which is supported by plotting the probability distributions and boxplot of GrLivArea, as seen in Plot 1.2. We also added an interaction between GrLivArea and Neighborhood because we found slight evidence of the Neighborhood having a possible effect on GrLivArea (Plot 1.1), which makes sense logically - a wealthier neighborhood will probably have larger homes, and thus, a higher SalePrice. Given the basic assumptions being met, we can run our model below, analyze, and adjust to meet the necessary assumptions.

The Model:

$$\mu(\text{SalePrice}) = b0 + b1(\text{GrLivArea}) + b2{*}\text{BrkSide} + B3{*}\text{Edwards} + b4(\text{GrLivArea}{*}\text{BrkSide}) + b5(\text{GrLivArea}{*}\text{Edwards})$$

Checking Assumptions
Our initial model depicted by Plot 1.3 [Appendix] confirms the existence of outliers. With the model as-is, we have obtained an adjusted R-Square of 0.44. Overall, the model looks decent but will likely improve with the removal of said outliers. Based on the plots, there does not appear to be any major trends or any evidence against a normal distribution with constant variance. We already assume that the observations are independent.

Comparing Competing Models
By creating table 1.4 [Appendix], we are able to easily identify the outliers based on residual and Cook's D. We chose a Cook's D threshold of 0.02.
and found eleven outliers, see Table 1.4 [Appendix]. Given that the majority of Cook's D were below 0.01 with a few just above the threshold, so we decided to increase the threshold to 0.02. Now, the model appears to be much better fit - see Plots 1.5 - 1.8. We ran the model without those outliers and obtained an R-Squared of 0.5244, which can be seen in Table 1.5.

Parameters
Our resulting parameter estimates from our model can be seen in Table 1.6 [Appendix].

Fitted Models:

$$\mu(\text{SalePrice}|\text{NAmes}) = 80157.51 + 49.66(\text{GrLivArea})$$

$$\mu(\text{SalePrice}|\text{BrkSide}) = 22396.27 + 84.17(\text{GrLivArea})$$

$$\mu(\text{SalePrice}|\text{Edwards}) = 63756.64 + 46.25(\text{GrLivArea})$$

Given our results, we can estimate that:

- For a home in NAmes, holding BrkSide & Edwards constant, a 100 sqft. increase in gross living area is associated with a mean increase of $4,966 in sale price. We are 95% confident that the increase in sale price is between $4,197 and $5,734.
- For a home in BrkSide, holding NAmes & Edwards constant, a 100 sqft. increase in gross living area is associated with a mean increase of $8,417 in sale price. We are 95% confident that the increase in sale price is between $6,861 and $9,972.
- For a home in Edwards, holding NAmes & BrkSide constant, a 100 sqft. increase in gross living area is associated with a mean increase of $4,625 in sale price. We are 95% confident that the increase in sale price is between $3,121 and $6,129.

Conclusion
The model is quite good given our p-value < 0.0001 for the F-test. Our adjusted R^2 tells us that 52.44% of the variability of Sale Price can be explained by Gross Living Area. Given the $ increase per 100 sqft., one can infer that a home in BrkSide will likely cost/increase more than a home in NAmes or Edwards; however, the mean home sale price appears to be greatest in Edwards - Table 1.3 [Appendix]. Given that this is an observational study, causal inference cannot be attributed to any parameter with relation to sale price.

R Shiny App: Sale Price vs. Gross Living Area

https://santigtz95.shinyapps.io/STAT_RShiny/

https://vochannguyen.shinyapps.io/Stats1Shiny/

Question 2 Analysis

The Problem

Build the most predictive model for sales prices of homes in all of Ames, Iowa - including all neighborhoods. Produce four models using: forward selection, backward selection, stepwise selection, and a custom-built model. Generate an adjusted R^2, CV Press, and Kaggle Score for each of the models. Clearly describe which model is the best fit in terms of being able to predict future sale prices of homes in Ames.

Build and Fit the Models

Before building the model, we remove the outliers we found in question 1. We also inspected the data to identify and remove variables with large amounts of NA/missing values, as well as those with unnecessary application. Once we had our "good" variables, we built our initial model.

```
/* Remove Known Outliers */
data Train2;
set TrainData;
keep Id Neighborhood GrLivArea SalePrice;
where Id ~= 176 AND Id ~= 524 AND Id ~= 608 AND Id~= 643 AND
Id ~= 667 AND Id ~= 725 AND Id ~= 808 AND Id ~= 889 AND
Id ~= 1169 AND Id ~= 1299 AND Id ~= 1424;
run;

/* Inspect for NA Values */
proc means data=Train2 NMISS N;
run;

/* Create New Train Data Set with Good Variables*/
data Train2;
set Train2;
keep Id MSSubClass MSZoning LotArea LotShape LandContour FirstFlrSF SecondFlrSF
LotConfig   LandSlope   Neighborhood    Condition1  Condition2  BldgType    HouseStyle
OverallQual OverallCond YearBuilt   YearRemodAdd    RoofStyle   RoofMatl    Exterior1st
Exterior2nd MasVnrType  ExterQual   ExterCond   Foundation  BsmtQual    BsmtCond
BsmtExposure    BsmtFinType1    BsmtFinSF1  BsmtFinType2    BsmtFinSF2  BsmtUnfSF
TotalBsmtSF Heating HeatingQC   CentralAir  Electrical  LowQualFinSF    GrLivArea
BsmtFullBath    BsmtHalfBath    FullBath    HalfBath    BedroomAbvGr    KitchenAbvGr
KitchenQual TotRmsAbvGrd    Functional  Fireplaces GarageType   GarageFinish
GarageCars  GarageArea  GarageQual  GarageCond  PavedDrive  WoodDeckSF  OpenPorchSF EnclosedPorch
ScreenPorch PoolArea Fence  MiscFeature MiscVal MoSold  YrSold  SaleType    SaleCondition   SalePrice;
run;
```

We begin every model by checking for linearity, primarily between numerical variables. From there, we select those that appear to have some linear relation. Some relations appear to be better with sale price logged, so we perform the transformation and select the linear relation variables. We remove outliers using Cook's D / residual data and proceed with our analysis of the models.

Forward Model

Build and Run Forward Selection Model. Using cross validation, we found 19x potential variables with very good results in terms of our R^2, AIC, SBC, CV Press, and RMSE. See table 2.1 [Appendix].

- $R^2 = 91.91\%$
- CV Press = 20.7020

Backward Model

Build and Run Backward Selection Model. Using cross validation, the model eliminated 5x variables and produced good results in terms of our R^2, AIC, SBC, CV Press, and RMSE. However, the slightly better results (in comparison to forward selection model) required a lot more parameters. See table 2.2 [Appendix].

- $R^2 = 92.32\%$
- CV Press = 19.8670

Stepwise Model

Build and Run Stepwise Selection Model. Using cross validation, we found 14x potential variables with very good results in terms of our R^2, AIC, SBC, CV Press, and RMSE. See table 2.3 [Appendix].

- $R^2 = 91.62\%$
- CV Press = 20.7020

Custom-Backward Model

Given the results from the backward selection model, we build and run a custom-backward selection model with 5x variables eliminated. Using the custom-backward model, we have maintained a good R^2 and slightly reduced the CV Press, as well as a very good Kaggle score. See table 2.4 [Appendix].

- $R^2 = 92.32\%$
- CV Press = 19.3796

Comparing Competing Models

Model	Adjusted R^2	CV Press	Kaggle Score
Forward	0.9191	20.7020	0.14174
Backward	0.9232	19.8670	0.13954
Stepwise	0.9162	20.6177	0.13954
Custom	0.9232	19.3796	0.13954

Conclusion

After running our models, it appears that for this data and model, the backward selection method is best for selecting variables. This model provided us with a slightly higher adjusted R^2 to the other two selection models and slightly lower CV Press. Our original custom model did not perform as well as we had hoped, so we incorporated the backward model and ultimately built a custom-backward model, which provided the highest adjusted R^2, lowest CV Press, and a low kaggle score. With our custom-backward model, we are able to determine that 92.32% of the variance for the dependent variable (sale price) is explained by the independent/explanatory variables in the model. We highly recommend using our custom-backward model for predicting home prices in Ames, Iowa.

Appendix

Question 1 Analysis (Code, Tables, and Plots)

- ➔ Inspect data and filter necessary variables (Id, Neighborhood, GrLivArea, SalePrice). Filter the three neighborhoods we are analyzing.

```
/* Print First Twenty Lines - Inspect */
proc print data=TrainData (obs=20);
run;

/* Select Needed Variables and Filter Neighborhoods */
data Train1;
set TrainData;
keep Id Neighborhood GrLivArea SalePrice;
where Neighborhood = 'NAmes' OR Neighborhood = 'Edwards' OR Neighborhood = 'BrkSide';
run;

proc print data=Train1 (obs=20);
run;
```

Table 1.1

Obs	Id	Neighborhood	GrLivArea	SalePrice
1	10	BrkSide	1077	118000
2	15	NAmes	1253	157000
3	16	BrkSide	854	132000
4	17	NAmes	1004	149000
5	20	NAmes	1339	139000
6	27	NAmes	900	134800
7	29	NAmes	1600	207500
8	30	BrkSide	520	68500
9	34	NAmes	1700	165500
10	38	NAmes	1297	153000
11	39	NAmes	1057	109000
12	40	Edwards	1152	82000
13	41	NAmes	1324	160000
14	45	NAmes	1150	141000
15	52	BrkSide	1176	114500
16	55	NAmes	1360	130000
17	56	NAmes	1425	180500
18	67	NAmes	2207	180000
19	71	NAmes	2223	244000
20	74	NAmes	1086	144900

→ Check for missing NA values - none present with these variables.

```
/* Inspect for NA Values - No NA Values */
proc means data=Train1 NMISS N;
run;
```

Table 1.2

Variable	N Miss	N
Id	0	383
GrLivArea	0	383
SalePrice	0	383

→ Review data summary.

```
/* Data Summary */
proc means data=Train1 alpha=0.05;
class Neighborhood;
```

Table 1.3

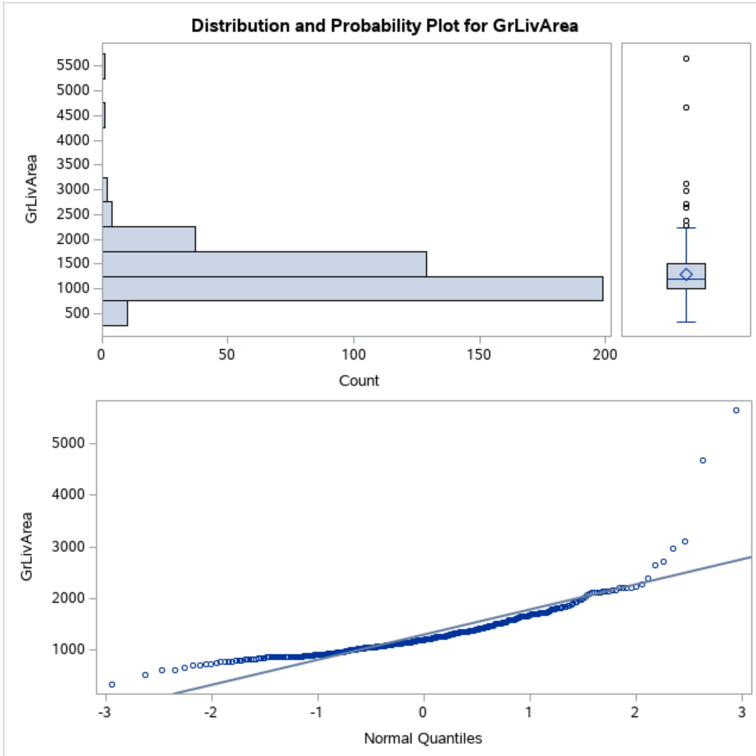
Neighborhood	N Obs	Variable	N	Mean	Std Dev	Minimum	Maximum
BrkSide	58	Id	58	734.7241379	435.8508836	10.0000000	1444.00
		GrLivArea	58	1203.07	386.6142219	334.0000000	2134.00
		SalePrice	58	124834.05	40348.69	39300.00	223500.00
Edwards	100	Id	100	762.9300000	413.2906199	40.0000000	1460.00
		GrLivArea	100	1340.04	655.2099196	605.0000000	5642.00
		SalePrice	100	128219.70	43208.62	58500.00	320000.00
NAmes	225	Id	225	737.9955556	431.3032834	15.0000000	1459.00
		GrLivArea	225	1310.31	413.4982522	767.0000000	3112.00
		SalePrice	225	145847.08	33075.35	87500.00	345000.00

➔ Inspect scatter plot of Sales Price vs. GrLivArea (Plot 1.1), as well as distribution and boxplot (Plot 1.2). Scatter plot appears to depict a positive linear relation with a few possible outliers for both variables. The outliers become evident after inspecting the additional plots. The distribution of GrLivSpace appears to be slightly skewed, most likely due to the outliers. No log transformation needed. There is also slight evidence that two of the Neighborhoods appear to frequently have higher GrLivArea, thus our interaction.

```
/* Visually Inspect Variables for Normality and Outliers */
proc sgplot data = Train1;
    title 'Sales Price vs Gross Living Area';
    scatter x = GrLivArea y = SalePrice /
        markerattrs= (color=blue symbol=circlefilled);
    xaxis label = 'Gross Living Area';
    yaxis label = 'Sales Price';
run;

proc univariate data=Train1 alpha=0.05 plot;
var GrLivArea SalePrice;
run;
```

Plot 1.1

Plot 1.2

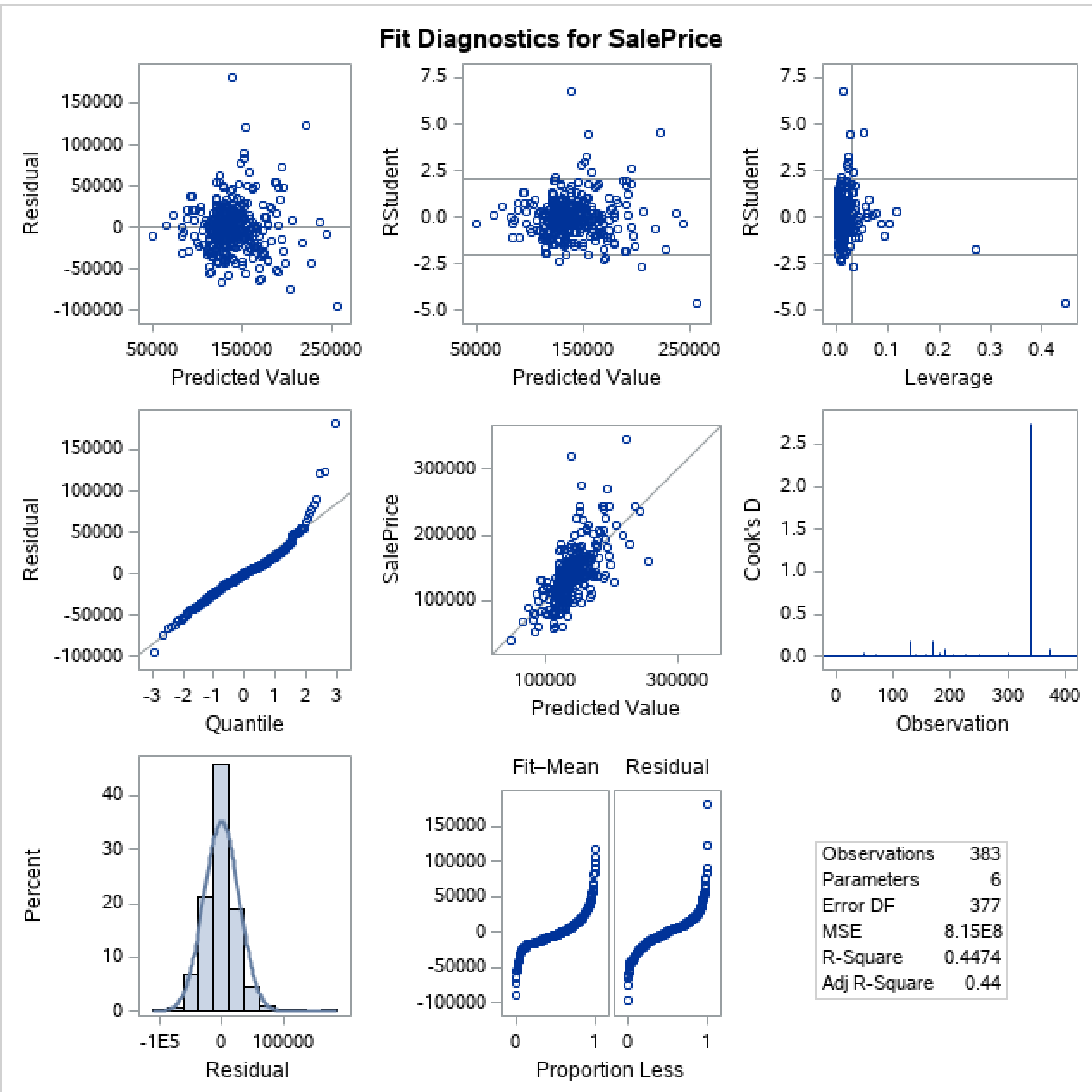


Distribution and Probability Plot for GrLivArea
GrLivArea
5500
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
50
100
150
200
Count
GrLivArea
5000
4000
3000
2000
1000
-3
-2
-1
0
1
2
3
Normal Quantiles

→ Build the model and run it. Inspect the plots to determine which outliers to possibly remove.

```
/* Run Model: SalePrice-GrLivArea by Class=Neighborhood*/
proc glm data=Train1 alpha=0.05 plots = All;
class Neighborhood;
model SalePrice = GrLivArea|Neighborhood / solution clparm;
run;
```

Plot 1.3



➔ Based on the results from our model, we have determined that a few outliers need to be removed. We run a modified model to allow us to extract and remove the desired outliers - those with Cook's D greater than 0.02 (ie., ~ 4/n). Although the threshold should be 0.01, we chose 0.02 because there were a few observations that were ever so slightly above the threshold. Keeping those observations felt just.

```
/* Identify the Outliers */
proc glm data=Train1 alpha=0.05;
class Neighborhood;
model SalePrice = GrLivArea|Neighborhood / solution clparm;
output out=outliers1 P=Fitted PRESS=PRESS H=HAT
RSTUDENT=EXTST R=RESID DFFITS=DFFITS COOKD=COOKD;
run ;
proc print data=outliers1;

data outliers1;
set outliers1;
where COOKD > (0.02);
run;
proc print data=outliers1;

/* Remove Outliers */
data Train1;
set Train1;
keep Id Neighborhood GrLivArea SalePrice;
where Id ~= 176 AND Id ~= 524 AND Id ~= 608 AND Id~= 643 AND
Id ~= 667 AND Id ~= 725 AND Id ~= 808 AND Id ~= 889 AND
Id ~= 1169 AND Id ~= 1299 AND Id ~= 1424;
run;
```

Table 1.4

Obs	Id	Neighborhood	GrLivArea	SalePrice	Fitted	PRESS	HAT	EXTST	RESID	DFFITS	COOKD
1	176	Edwards	2158	243000	152554.26	92835.54	0.02574	3.24974	90445.74	0.52824	0.04536
2	524	Edwards	4676	184750	227465.52	-58662.73	0.27185	-1.75806	-42715.52	-1.07420	0.19126
3	608	Edwards	2008	225000	148091.71	78517.74	0.02050	2.74512	76908.29	0.39711	0.02584
4	643	NAmes	2704	345000	221546.49	130660.67	0.05516	4.56366	123453.51	1.10267	0.19252
5	667	NAmes	2380	129000	203948.15	-77611.81	0.03432	-2.69324	-74948.15	-0.50773	0.04226
6	725	Edwards	1698	320000	138869.12	183519.37	0.01301	6.75266	181130.88	0.77543	0.08961
7	808	BrkSide	1576	223500	157339.67	68458.16	0.03357	2.37147	66160.33	0.44195	0.03216
8	889	NAmes	2217	268000	195094.67	74844.47	0.02591	2.60694	72905.33	0.42516	0.02967
9	1169	Edwards	2108	235000	151066.74	85986.31	0.02388	3.00693	83933.26	0.47028	0.03609
10	1299	Edwards	5642	160000	256204.31	-173481.23	0.44545	-4.64654	-96204.31	-4.16445	2.74075
11	1424	Edwards	2201	274970	153833.52	124554.37	0.02744	4.40585	121136.48	0.74007	0.08703

→ We reinspect our scatter plot of Sales Price vs. GrLivArea without the outliers. A positive linear relationship is much more evident without the outliers, as well as a normal distribution.

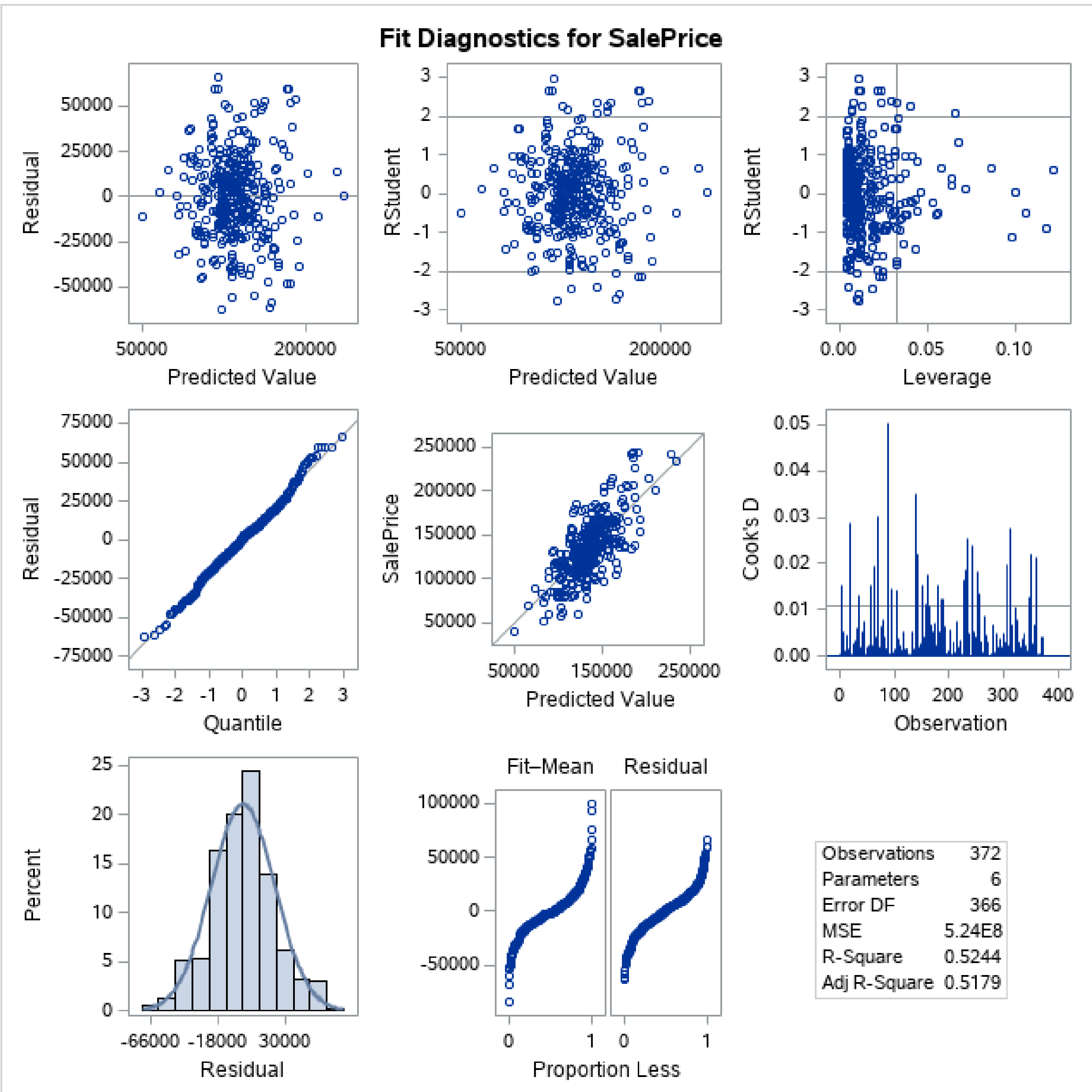
```
/* Visually Inspect Without Outliers */
proc sgplot data = Train1;
    title 'Sales Price vs Gross Living Area';
    scatter x = GrLivArea y = SalePrice /
        markerattrs= (color=blue symbol=circlefilled);
    xaxis label = 'Gross Living Area';
    yaxis label = 'Sales Price';
run;

proc univariate data=Train1 alpha=0.05 plot;
var GrLivArea SalePrice;
run;
```

Plot 1.4

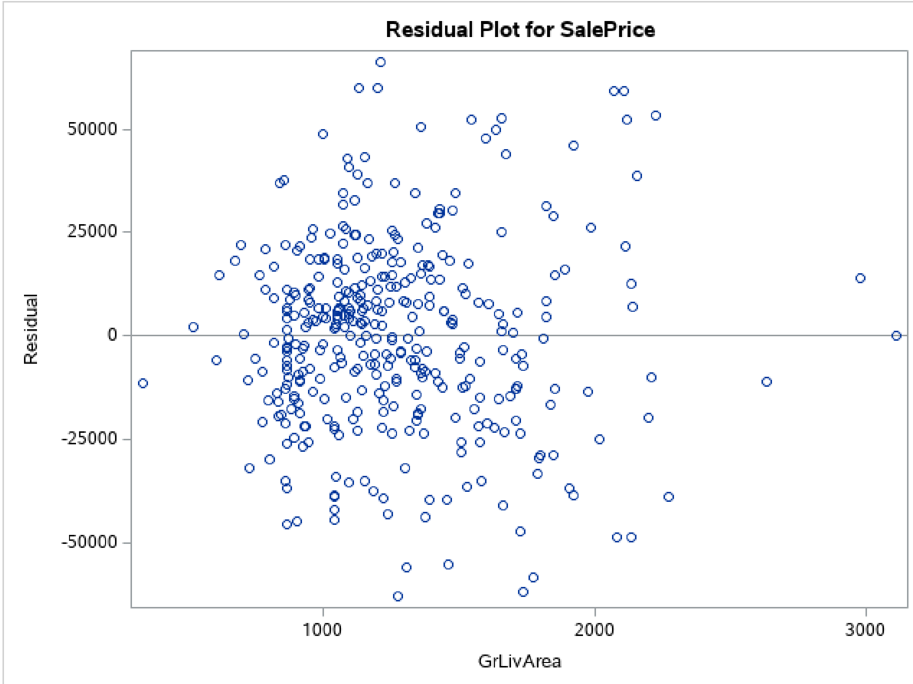
→ Run the model without outliers and inspect. From our results, we can see a slightly improved model with an increased adjusted R^2 =~ 0.52

```
/* Run Model Without Outliers - Better Model */
proc glm data=Train1 alpha=0.05 plots = All;
class Neighborhood;
model SalePrice = GrLivArea|Neighborhood / solution clparm;
run;
```

Plot 1.5



Plot 1.7



Plot 1.8

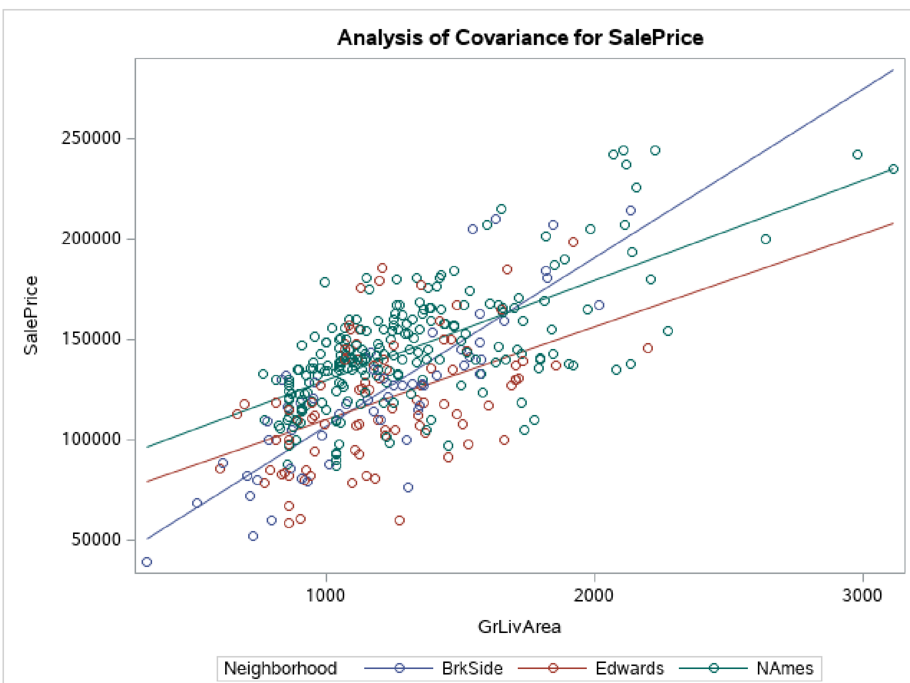


Table 1.5

The GLM Procedure

Dependent Variable: SalePrice

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	5	211560334729	42312066946	80.72	<.0001
Error	366	191839359847	524151256.41		
Corrected Total	371	403399694577			

R-Square	Coeff Var	Root MSE	SalePrice Mean
0.524443	16.94201	22894.35	135133.7

Table 1.6

Parameter	Estimate		Standard Error	t Value	Pr > |t|	95% Confidence Limits	
Intercept	80157.51293	B	5288.07308	15.16	<.0001	69758.69320	90556.33267
GrLivArea	49.66150	B	3.90687	12.71	<.0001	41.97877	57.34423
Neighborhood BrkSide	-57761.24177	B	11257.25240	-5.13	<.0001	-79898.25400	-35624.22953
Neighborhood Edwards	-16400.87051	B	10987.15831	-1.49	0.1364	-38006.75170	5205.01067
Neighborhood NAmes	0.00000	B	.	.	.	.	.
GrLivArea*Neighborho BrkSide	34.50447	B	8.82180	3.91	0.0001	17.15669	51.85225
GrLivArea*Neighborho Edwards	-3.41233	B	8.58721	-0.40	0.6913	-20.29880	13.47414
GrLivArea*Neighborho NAmes	0.00000	B	.	.	.	.	.

Question 2 Analysis (Code, Tables, and Plots)

➔ Build Initial model with “good” variables and remove known outliers from Question 1.

```
/* Remove Known Outliers */
data Train2;
set TrainData;
keep Id Neighborhood GrLivArea SalePrice;
where Id ~= 176 AND Id ~= 524 AND Id ~= 608 AND Id~= 643 AND
Id ~= 667 AND Id ~= 725 AND Id ~= 808 AND Id ~= 889 AND
Id ~= 1169 AND Id ~= 1299 AND Id ~= 1424;
run;

/* Inspect for NA Values */
proc means data=Train2 NMISS N;
run;

/* Create New Train Data Set with Good Variables*/
data Train2;
set Train2;
keep Id MSSubClass MSZoning LotArea LotShape LandContour FirstFlrSF SecondFlrSF
LotConfig   LandSlope   Neighborhood    Condition1  Condition2  BldgType    HouseStyle
OverallQual OverallCond YearBuilt   YearRemodAdd    RoofStyle   RoofMatl    Exterior1st
Exterior2nd MasVnrType  ExterQual   ExterCond   Foundation  BsmtQual    BsmtCond
BsmtExposure    BsmtFinType1    BsmtFinSF1  BsmtFinType2    BsmtFinSF2  BsmtUnfSF
TotalBsmtSF Heating HeatingQC   CentralAir  Electrical  LowQualFinSF    GrLivArea
BsmtFullBath    BsmtHalfBath    FullBath    HalfBath    BedroomAbvGr    KitchenAbvGr
KitchenQual TotRmsAbvGrd    Functional  Fireplaces GarageType   GarageFinish
GarageCars  GarageArea  GarageQual  GarageCond  PavedDrive  WoodDeckSF  OpenPorchSF EnclosedPorch
ScreenPorch PoolArea Fence  MiscFeature MiscVal MoSold  YrSold  SaleType    SaleCondition   SalePrice;
run;
```

➔ Although some values still contained NA values, we decided to move forward with our linear relation analysis and variables selection, and then later on would convert any necessary NA values. We looked for linear relationships and found a decent amount.

```
/* Check for Linear Relationships for All Numerical Variables */
PROC sgscatter DATA=Train2;
matrix SalePrice MSSubClass LotArea OverallQual OverallCond YearBuilt YearRemodAdd;
run;
PROC sgscatter DATA=Train2;
matrix SalePrice BsmtFinSF1 BsmtFinSF2 BsmtUnfSF FirstFlrSF SecondFlrSF;
run;
PROC sgscatter DATA=Train2;
matrix SalePrice LowQualFinSF GrLivArea BsmtFullBath BsmtHalfBath FullBath HalfBath;
run;
PROC sgscatter DATA=Train2;
matrix SalePrice BedroomAbvGr KitchenAbvGr TotRmsAbvGrd Fireplaces GarageYrBlt GarageCars GarageArea;
run;
PROC sgscatter DATA=Train2;
matrix SalePrice WoodDeckSF OpenPorchSF EnclosedPorch ScreenPorch PoolArea;
run;
PROC sgscatter DATA=Train2;
matrix SalePrice MiscVal MoSold YrSold;
run;
```

→ Although we found linear relationships amongst variables, we decided to check the relationships with a log transformation of Sale Price. In terms of linear relationships, we obtained much better results with the log-transformed data.

```
/* Log SalePrice and ReCheck Relationships for All Numerical Variables */
data Train2;
set Train2;
logSalePrice = log(SalePrice);
run;

PROC sgscatter DATA=Train2;
matrix logSalePrice MSSubClass LotArea OverallQual OverallCond YearBuilt YearRemodAdd;
run;
PROC sgscatter DATA=Train2;
matrix logSalePrice BsmtFinSF1 BsmtFinSF2 BsmtUnfSF FirstFlrSF SecondFlrSF;
run;
PROC sgscatter DATA=Train2;
matrix logSalePrice LowQualFinSF GrLivArea BsmtFullBath BsmtHalfBath FullBath HalfBath;
run;
PROC sgscatter DATA=Train2;
matrix logSalePrice BedroomAbvGr KitchenAbvGr TotRmsAbvGrd Fireplaces GarageYrBlt GarageCars GarageArea;
run;
PROC sgscatter DATA=Train2;
matrix logSalePrice WoodDeckSF OpenPorchSF EnclosedPorch ScreenPorch PoolArea;
run;
PROC sgscatter DATA=Train2;
matrix logSalePrice MiscVal MoSold YrSold;
run;
```

→ Build and Run Forward Selection Model. Using cross validation, we found 19x potential variables with very good results in terms of our R^2, AIC, SBC, CV Press, and RMSE.

```
/* Forward Selection Model*/
proc glmselect data = Train2;
class Neighborhood MSZoning LotShape LotConfig Condition1 Condition2
BldgType HouseStyle RoofStyle Exterior1st Exterior2nd Foundation
BsmtFinType1 HeatingQc CentralAir Electrical KitchenQual GarageType
GarageFinish SaleType;
model logSalePrice = OverallQual OverallCond YearBuilt YearRemodAdd
BsmtFinSF1 BsmtFinSF2 BsmtUnfSF FirstFlrSF SecondFlrSF FullBath HalfBath
BedroomAbvGr TotRmsAbvGrd Fireplaces GarageCars GarageArea WoodDeckSF
OpenPorchSF EnclosedPorch ScreenPorch PoolArea YrSold GrLivArea
Neighborhood MSZoning LotShape LotConfig Condition1 Condition2 BldgType
HouseStyle RoofStyle Exterior1st Exterior2nd Foundation BsmtFinType1
HeatingQc CentralAir Electrical KitchenQual GarageType GarageFinish SaleType
/selection = Forward(stop = cv) cvmethod = random(5) CVDETAILS stats = adjrsq;
run;
```

Table 2.1

Forward Selection Summary						
Step	Effect Entered	Number Effects In	Number Parms In	Adjusted R-Square	SBC	CV PRESS
0	Intercept	1	1	0.0000	-2654.3390	231.0492
1	OverallQual	2	2	0.6775	-4287.7495	74.6395
2	GrLivArea	3	3	0.7648	-4738.8921	54.4237
3	Neighborhood	4	27	0.8365	-5115.5683	39.1490
4	BsmtFinSF1	5	28	0.8609	-5343.5957	33.5614
5	OverallCond	6	29	0.8724	-5461.9570	30.7789
6	YearBuilt	7	30	0.8860	-5618.8752	27.6347
7	GarageArea	8	31	0.8945	-5725.3395	25.7864
8	BsmtUnfSF	9	32	0.8994	-5788.2476	24.6765
9	BsmtFinSF2	10	33	0.9043	-5853.6077	23.5027
10	MSZoning	11	37	0.9093	-5905.9068	22.7153
11	Fireplaces	12	38	0.9117	-5938.9022	22.1422
12	BldgType	13	42	0.9142	-5955.1940	21.8758
13	YearRemodAdd	14	43	0.9153	-5968.3304	21.5523
14	GarageCars	15	44	0.9162	-5977.8733	21.3963
15	CentralAir	16	45	0.9172	-5988.2589	21.1821
16	ScreenPorch	17	46	0.9179	-5995.5084	20.9562
17	WoodDeckSF	18	47	0.9184	-5997.9005	20.8602
18	OpenPorchSF	19	48	0.9188	-5998.4087*	20.7605
19	EnclosedPorch	20	49	0.9191*	-5997.8532	20.7020*
* Optimal Value of Criterion						

Root MSE	0.11355
Dependent Mean	12.02165
R-Square	0.9218
Adj R-Sq	0.9191
AIC	-4805.50605
AICC	-4801.85798
SBC	-5997.85323
CV PRESS	20.70205

→ Build and Run Backward Selection Model. Using cross validation, the model eliminated 5x variables and produced good results in terms of our R^2, AIC, SBC, CV Press, and RMSE. However, the slightly better results (in comparison to forward selection model) required a lot more parameters.

```
/* Backward Selection Model*/
proc glmselect data = Train2;
class Neighborhood MSZoning LotShape LotConfig Condition1 Condition2
BldgType HouseStyle RoofStyle Exterior1st Exterior2nd Foundation
BsmtFinType1 HeatingQc CentralAir Electrical KitchenQual GarageType
GarageFinish SaleType;
model logSalePrice = OverallQual OverallCond YearBuilt YearRemodAdd
BsmtFinSF1 BsmtFinSF2 BsmtUnfSF FirstFlrSF SecondFlrSF FullBath HalfBath
BedroomAbvGr TotRmsAbvGrd Fireplaces GarageCars GarageArea WoodDeckSF
OpenPorchSF EnclosedPorch ScreenPorch PoolArea YrSold GrLivArea
Neighborhood MSZoning LotShape LotConfig Condition1 Condition2 BldgType
HouseStyle RoofStyle Exterior1st Exterior2nd Foundation BsmtFinType1
HeatingQc CentralAir Electrical KitchenQual GarageType GarageFinish SaleType
/ selection = Backward(stop = cv) cvmethod = random(5) CVDETAILS stats = adjrsq;
run;
```

Table 2.2

Backward Selection Summary						
Step	Effect Removed	Number Effects In	Number Parms In	Adjusted R-Square	SBC	CV PRESS
0		40	131	0.9245	-5588.6090	21.0506
1	Exterior2nd	39	117	0.9247*	-5678.7884	20.6426
2	Exterior1st	38	103	0.9235	-5741.7021	20.5616
3	HouseStyle	37	96	0.9234	-5783.6818	20.2713
4	RoofStyle	36	91	0.9235	-5816.4287	19.8741
5	BsmtFinType1	35	85	0.9232	-5849.2810*	19.8669*
* Optimal Value of Criterion						

Root MSE	0.11062
Dependent Mean	12.02165
R-Square	0.9277
Adj R-Sq	0.9232
AIC	-4846.96442
AICC	-4835.97764
SBC	-5849.28096
CV PRESS	19.86695

→ Build and Run Stepwise Selection Model. Using cross validation, we found 14x potential variables with very good results in terms of our R^2, AIC, SBC, CV Press, and RMSE.

```
/* Stepwise Selection Model*/
proc glmselect data = Train2;
class Neighborhood MSZoning LotShape LotConfig Condition1 Condition2
BldgType HouseStyle RoofStyle Exterior1st Exterior2nd Foundation
BsmtFinType1 HeatingQc CentralAir Electrical KitchenQual GarageType
GarageFinish SaleType;
model logSalePrice = OverallQual OverallCond YearBuilt YearRemodAdd
BsmtFinSF1 BsmtFinSF2 BsmtUnfSF FirstFlrSF SecondFlrSF FullBath HalfBath
BedroomAbvGr TotRmsAbvGrd Fireplaces GarageCars GarageArea WoodDeckSF
OpenPorchSF EnclosedPorch ScreenPorch PoolArea YrSold GrLivArea
Neighborhood MSZoning LotShape LotConfig Condition1 Condition2 BldgType
HouseStyle RoofStyle Exterior1st Exterior2nd Foundation BsmtFinType1
HeatingQc CentralAir Electrical KitchenQual GarageType GarageFinish SaleType
/ selection = Stepwise(stop = cv) cvmethod = random(5) CVDETAILS stats = adjrsq;
run;
```

Table 2.3

The GLMSELECT Procedure

Stepwise Selection Summary

Step	Effect Entered	Effect Removed	Number Effects In	Number Parms In	Adjusted R-Square	SBC	CV PRESS
0	Intercept		1	1	0.0000	-2654.3390	231.0822
1	OverallQual		2	2	0.6775	-4287.7495	74.5838
2	GrLivArea		3	3	0.7648	-4738.8921	54.3698
3	Neighborhood		4	27	0.8365	-5115.5683	38.6142
4	BsmtFinSF1		5	28	0.8609	-5343.5957	32.8438
5	OverallCond		6	29	0.8724	-5461.9570	30.0525
6	YearBuilt		7	30	0.8860	-5618.8752	27.1182
7	GarageArea		8	31	0.8945	-5725.3395	25.3094
8	BsmtUnfSF		9	32	0.8994	-5788.2476	24.1205
9	BsmtFinSF2		10	33	0.9043	-5853.6077	22.8936
10	MSZoning		11	37	0.9093	-5905.9068	22.0452
11	Fireplaces		12	38	0.9117	-5938.9022	21.4539
12	BldgType		13	42	0.9142	-5955.1940	21.0197
13	YearRemodAdd		14	43	0.9153	-5968.3304	20.7554
14	GarageCars		15	44	0.9162*	-5977.8733*	20.6176*

* Optimal Value of Criterion

Root MSE	0.11557
Dependent Mean	12.02165
R-Square	0.9187
Adj R-Sq	0.9162
AIC	-4759.13295
AICC	-4756.18213
SBC	-5977.87328
CV PRESS	20.61765

➔ Given the results from the backward selection model, we build and Run Custom-Backward Selection Model. Using the custom-backward model, we have maintained a good R^2 and slightly reduced the CV Press, as well as a very good Kaggle score.

```
/* Custom-Backward Selection Model and Store as TrainModelCustom */
proc glmselect data = Train2;
class Neighborhood MSZoning LotShape
BldgType HouseStyle RoofStyle Exterior1st Exterior2nd Foundation
BsmtFinType1 HeatingQc CentralAir Electrical KitchenQual
GarageFinish SaleType;
model logSalePrice = OverallQual OverallCond YearBuilt YearRemodAdd
BsmtFinSF1 BsmtFinSF2 BsmtUnfSF FirstFlrSF SecondFlrSF FullBath HalfBath
BedroomAbvGr TotRmsAbvGrd Fireplaces GarageCars GarageArea WoodDeckSF
OpenPorchSF EnclosedPorch ScreenPorch PoolArea YrSold GrLivArea
Neighborhood MSZoning LotShape BldgType
HouseStyle RoofStyle Exterior1st Exterior2nd Foundation BsmtFinType1
HeatingQc CentralAir Electrical KitchenQual GarageFinish SaleType
/selection = Backward(stop = cv) cvmethod = random(5) CVDETAILS stats = adjrsq;
store TrainModelCustom;
run;
```

Table 2.4

Backward Selection Summary						
Step	Effect Removed	Number Effects In	Number Parms In	Adjusted R-Square	SBC	CV PRESS
0		40	131	0.9245	-5588.6090	20.8264
1	Exterior2nd	39	117	0.9247*	-5678.7884	20.4141
2	Exterior1st	38	103	0.9235	-5741.7021	19.9314
3	HouseStyle	37	96	0.9234	-5783.6818	19.8198
4	RoofStyle	36	91	0.9235	-5816.4287	19.4135
5	BsmtFinType1	35	85	0.9232	-5849.2810*	19.3796*

* Optimal Value of Criterion

Analysis of Variance				
Source	DF	Sum of Squares	Mean Square	F Value
Model	84	214.15773	2.54950	208.34
Error	1364	16.69126	0.01224	
Corrected Total	1448	230.84899		

Root MSE	0.11062
Dependent Mean	12.02165
R-Square	0.9277
Adj R-Sq	0.9232
AIC	-4846.96442
AICC	-4835.97764
SBC	-5849.28096
CV PRESS	19.37959