

MSDS 6372

Applied Statistics: Inference and Modeling

Section 403

Regression Data Analysis Project

Ames Iowa Housing Data

June 11, 2017

Dr. Jack K. Rasmus-Vorrath

Laurie Harris

## **Multiple Linear Regression of Residential Housing Sales Data**

Selecting an appropriate sales price is a critical step in preparing a residential property to be listed on the real estate market. Most buyers begin their property search with specific attributes in mind: location, living area, lot size and numerous other features, as well as how much money they wish to spend on a home, in the form of lower and upper limits of sales price. This price range will qualify which homes a buyer will consider in their search. If a home's sales price is set too high, it will preclude eligible buyers from considering the property and if it is set too low, the home's perceived value could be negatively impacted. Although, the real estate market will often drive corrections to improperly priced properties, dependence on the market can be an expensive approach, costing sellers and agents time and money in the form of excess days on the market and increased selling expenses.

### **Problem Statement**

Develop a regression model based on observations of sold residential properties that can be used to predict future home sales prices based on forecasted values.

### **Constraints and Limitations**

The analysis was completed on home sales data for residential properties in Ames, Iowa from 2006 through 2010. Because these data are observational, no causal inferences can be made. It also would not be appropriate to draw inferences to other area-specific real estate markets outside of Ames.

It should be noted that the data file contains attributes that are specific to each property sold. We recognize that home sales prices can be influenced by economic and employment conditions in the local market and variation in buyer demographics. This data set does not contain any external economic or buyer-specific information that we can consider for this analysis.

### **Data Set Description**

The data set contains 1460 observations representing residential property sales in Ames, Iowa during the years 2006 through 2010 and contains 75 explanatory variables identifying attributes which may have some influence in determining housing sales prices. The response variable in the data set is the home's sales price. We have classified the explanatory variables into three

primary categories known to affect property valuations: location, land attributes, and building improvements.

Location:

- Neighborhood classification
- Access to property and proximity to railroads, feeder roads, etc.
- Utility presence

Land Attributes:

- Lot size
- Zoning conditions

Building Improvements:

- Type of dwelling, condition and quality
- Total living area (in square feet)
- Numbers of bedrooms and bathrooms
- Basement presence, and features of the space (finished, bathrooms, etc.)
- Other considerations including fireplace, swimming pool, porches, and exterior materials and conditions

### **Exploratory Data Analysis**

During the exploratory data analysis, we notice some fields have missing values indicated by (-1). We have resolved these to be shown as true missing values. We observe that the missing values seem appropriate in relation to the other attributes for each particular observation. For example, data describing garage quality are missing when there happens to be no garage for that observation. Likewise, data for pool quality would be missing if the property did not have a pool.

Because of the number of variables, our analysis included many visual examinations of the explanatory variables related to the response variable. One challenge of this analysis is that many of the explanatory variables are factors with discrete levels and subjective evaluation. For example, we know that quality and condition can influence home prices; however, these attributes are usually defined as “good”, “excellent”, “poor”, etc. and we cannot be certain of the criteria used to evaluate the subject properties.

We also analyzed the frequency of identifiers for many of these attributes to eliminate variables that would have little effect on the response. For example, of the 1460 observations, only six

were on gravel road compared to paved roads. Therefore, street description can be eliminated as a potential predictor as the identifiers are homogeneous.

Preliminary examinations of matrix scatterplots indicate it would be appropriate to transform some variables, specifically the sales price (response variable), living area, and lot area. It is a real estate industry practice to describe homes in price per square foot (of improved dwelling space) as well as properties as price per acre, therefore it is reasonable to expect a relationship between these two variables and the response. However, the linearity seems to improve with log-transformation of these variables (Figures 1 and 2).

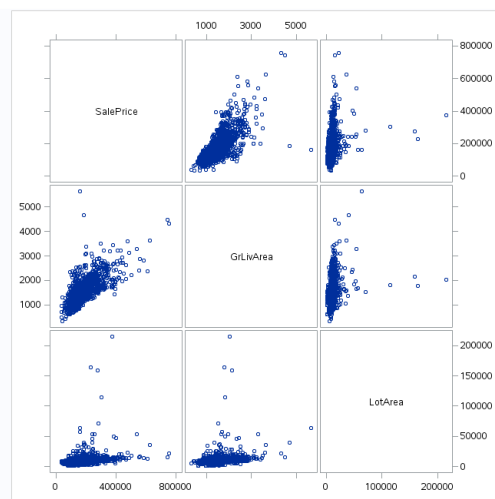


Figure 1: Sales Price, Living Area and Lot Area (Raw)

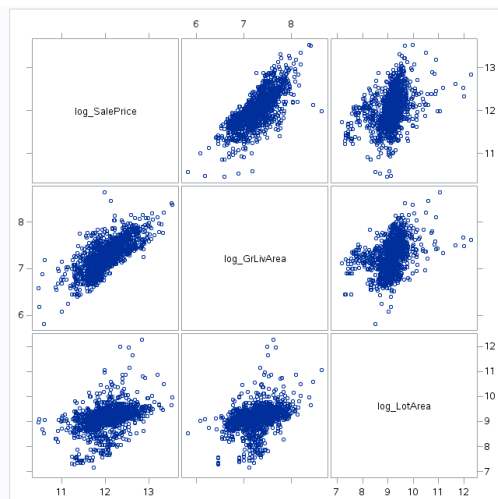


Figure 2: Sales Price, Living Area and Lot Area (Log Transformed)

## **Variable Screening**

We perform several iterative statistical tests to help us identify the most relevant predictive variables. During the analysis, we look for different ways to identify relationships among and between many of the categorical variables, including: Neighborhood, Building Type, Location/Access, Sale Condition and Kitchen Quality. We notice two observations with extremely high leverage. Because of the size of this sample and the leverage of these observations, we have chosen to remove them from our analysis.

At this point we consider the applicability of our analysis. Although we can produce a very good predictive model using automatic variable selection techniques, we consider the nature of our

original problem. Because real estate is a subjective practice, we have chosen to reassess our variable selection process and focus on some of the attributes that we believe, through inductive reasoning, should have an impact on sales price.

We focus on the following attributes to fit an intuitively-selected model. We retain the log-transformed sales price as the response variable, and the log-transformed living and lot areas, as explanatory variables. We add to those Overall Condition, Overall Quality and Year Built, since these are general terms used to describe home values. We infer that there are some other meaningful characteristics for homes in this area including Garage Car Size, Fireplace, Above Ground Kitchen, Total Basement living area and presence of a bathroom in the basement. The output of the regression model using this variables is shown in Figure 3.

The REG Procedure

Model: MODEL1

Dependent Variable: log\_SalePrice

Number of Observations Read	1458
Number of Observations Used	1458

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	10	209.03569	20.90357	1273.43	<.0001
Error	1447	23.75272	0.01642		
Corrected Total	1457	232.78842			

Root MSE	0.12812	R-Square	0.8980
Dependent Mean	12.02401	Adj R-Sq	0.8973
Coeff Var	1.06555		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr >  t	Variance Inflation
Intercept	1	0.16825	0.35724	0.47	0.6377	0
log_GrLivArea	1	0.42450	0.01538	27.61	<.0001	2.29109
log_LotArea	1	0.09128	0.00758	12.04	<.0001	1.34731
OverallQual	1	0.07866	0.00418	18.80	<.0001	2.94406
OverallCond	1	0.06074	0.00337	18.02	<.0001	1.25082
YearBuilt	1	0.00351	0.00016256	21.61	<.0001	2.13843
GarageCars	1	0.05660	0.00625	9.06	<.0001	1.93421
Fireplaces	1	0.03351	0.00622	5.39	<.0001	1.41550
Kitche-1bvGr	1	-0.09301	0.01658	-5.61	<.0001	1.18585
TotalBsmtSF	1	0.00013958	0.00001065	13.11	<.0001	1.73370
BsmtFullBath	1	0.06516	0.00695	9.37	<.0001	1.14802

Figure 3: Regression Model using Inductive Reasoning

This inductive approach performs quite well, and we decide to move forward with this model.

## **Model Selection**

We have selected the following model to be used for the prediction of home sales prices.

$$\log Y_{SalePrice} = \beta_0 + \log \beta_{LivArea} + \log \beta_{LotArea} + \beta_{Quality} + \beta_{Condition} + \beta_{YrBuilt} + \beta_{GarageCars} \\ + \beta_{Fireplaces} + \beta_{KitchAbove} + \beta_{BsmtSqFt} + \beta_{BsmtFullBath}$$

As we have identified an appropriate model, it is necessary to review residual plots to determine that the regression assumptions are met. (Figures 4-6)

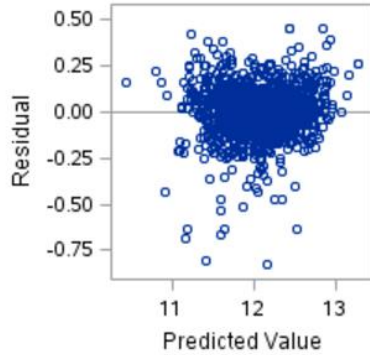


Figure 4: Residual Scatterplot

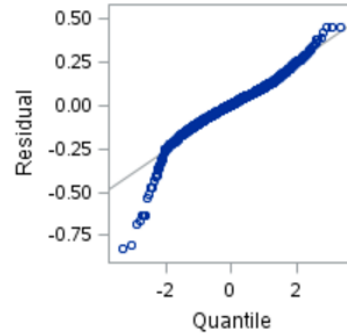


Figure 5: Residual QQ Plot

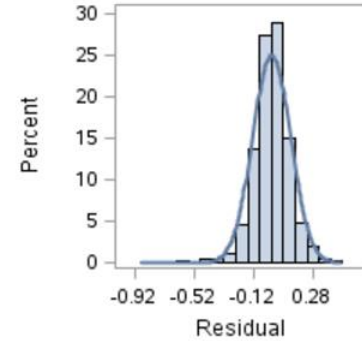


Figure 6: Residual Histogram

We find that the regression assumptions are nicely met as we see no departures from normality in the residual histogram or QQplot. We also examine the scatter plot of residuals and note no concerns regarding the constant variance assumption. In addition, we have no evidence to believe the observations are not independent, as each one represents a single home sales transaction.

With the assumptions being met, we evaluate the significance of the model and predictor variables. With a F-statistic of 1273.43 and p-value of <0.0001 the overall model is statistically significant. In addition, each of the explanatory variables show significance with p-values <0.0001 and Variance Inflation Factors are also noted to be low and within tolerances, indicating low multicollinearity. The adjusted r-squared for the model is noted as .8973, indicating almost 90% of the variability in the sales price can be explained by the model below.

$$\begin{aligned} & \text{Predicted Log Sales Price} \\ &= 0.168 + 0.425 * \text{Log\_LivArea} + 0.091 * \text{Log\_LotArea} + 0.079 * \text{OverallQual} \\ &+ 0.061 * \text{OverallCond} + 0.0035 * \text{YearBuilt} + 0.057 \text{GarageCars} + 0.034 \\ &* \text{Fireplaces} - 0.091 * \text{KitchenAbvGr} + 0.0001 * \text{TotalBsmtSF} + 0.065 \\ &* \text{BasementFullBath} \end{aligned}$$

## Improved Predictive Model

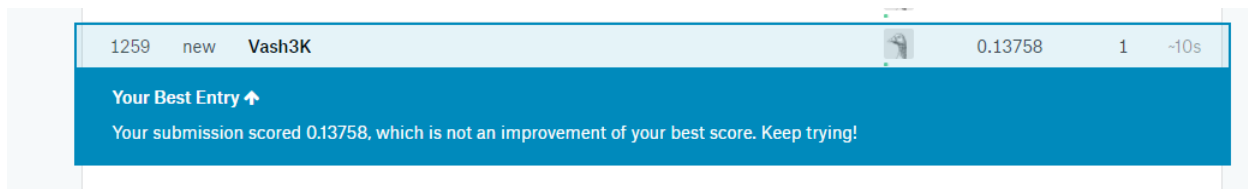
Although the above model performs quite well, we believe it can be refined to result in increased accuracy in home sales price predictions. For this analysis, we retain the variables considered in the parsimonious model and run LASSO automated selection to help identify additional categorical variables that will drive increased prediction precision. (Figure 7)

The REG Procedure						
Model: MODEL1						
Dependent Variable: log_SalePrice						
Number of Observations Read			1457			
Number of Observations Used			1457			
Analysis of Variance						
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F	
Model	18	211.60859	11.75603	883.19	<.0001	
Error	1438	19.14104	0.01331			
Corrected Total	1456	230.74962				
Root MSE		0.11537	R-Square	0.9170		
Dependent Mean		12.02499	Adj R-Sq	0.9160		
Coeff Var		0.95944				
Parameter Estimates						
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr >  t	Variance Inflation
Intercept	1	1.98538	0.37003	5.37	<.0001	0
log_GrLivArea	1	0.42904	0.01396	30.72	<.0001	2.32972
log_LotArea	1	0.08044	0.00765	10.52	<.0001	1.68903
OverallQual	1	0.06660	0.00398	16.74	<.0001	3.27590
OverallCond	1	0.05352	0.00310	17.26	<.0001	1.30306
YearBuilt	1	0.00262	0.00016931	15.49	<.0001	2.85487
GarageCars	1	0.05559	0.00567	9.80	<.0001	1.96414
Fireplaces	1	0.02742	0.00569	4.82	<.0001	1.45729
Kitchen-1bvGr	1	-0.08399	0.01506	-5.58	<.0001	1.20746
TotalBsmtSF	1	0.00010648	0.00001022	10.42	<.0001	1.96758
BsmtFullBath	1	0.02799	0.00787	3.56	0.0004	1.81560
dKQ_3	1	0.09043	0.01384	6.53	<.0001	1.31590
dSC_1	1	0.06534	0.01070	6.11	<.0001	1.83710
dSC_3	1	0.13180	0.01579	8.35	<.0001	2.10816
dMS_2	1	-0.21467	0.04022	-5.34	<.0001	1.08691
dMS_3	1	0.09654	0.01790	5.39	<.0001	1.49538
dMS_6	1	0.04210	0.01033	4.07	<.0001	1.94894
dHQ_1	1	0.03841	0.00724	5.31	<.0001	1.43390
BsmtFinSF1	1	0.00008327	0.00001030	8.08	<.0001	2.17583

Figure 7: Regression Model- Improved Predictive Model

We find that the following categorical variables have significance in our predictive model and can positively impact sales price: Excellent Kitchen Quality, Excellent Heating Quality, Residential Zoning as Low Density or Floating Village, and whether the home is considered Normal or New Construction. Conversely, if a property is zoned for Commercial, the sales price can be negatively impacted.

The Kaggle score for this model was .13758, as indicated below:



To maximize use of the available data for the purposes of prediction, a second process of model enrichment was then undertaken.

All remaining categorical variable levels were dummy coded. Forcing the predictors already identified as significant into the model, the dummy variables were run through three sets of non-penalizing automated selection procedures: stepwise, forward, and backward.

To avoid overfitting, and to simulate a real-world validation process where the answers are not available as a means for repeatedly correcting one's analysis, a single Kaggle submission was made for the improved model and for the final enriched model described below. Similarly, the statistical significance of the predictors was prioritized over any single exceptional model performance during the analytic process. Predictors identified as non-significant at the  $\alpha = .05$  level were manually removed from the models suggested by non-penalizing automated selection procedures before regression analysis was applied to each.

As a final step, a custom model was built on the basis of the results from each automated selection procedure, only including predictors that were identified as significant at the  $\alpha = .05$  level, and present in at least two of the three procedures.

The final enriched model included the original 10 predictors from the parsimonious model, and four of the six predictors from the improved model, before adding an additional 12 dummy-coded predictors from categorical factor levels identified as the most statistically significant.

The model and figures are indicated below:



The REG Procedure						
Model: MODEL1						
Dependent Variable: log_SalePrice						
Number of Observations Read		1455				
Number of Observations Used		1455				
Analysis of Variance						
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F	
Model	26	211.95779	8.15222	675.31	<.0001	
Error	1428	17.23864	0.01207			
Corrected Total	1454	229.19643				
Root MSE		0.10987	R-Square	0.9248		
Dependent Mean		12.02618	Adj R-Sq	0.9234		
Coeff Var		0.91361				
Parameter Estimates						
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr >  t	Variance Inflation
Intercept	1	1.74380	0.38199	4.56	<.0001	0
log_GrLivArea	1	0.42439	0.01355	31.31	<.0001	2.41680
log_LotArea	1	0.08283	0.00744	11.11	<.0001	1.76243
OverallQual	1	0.05411	0.00399	13.55	<.0001	3.62717
OverallCond	1	0.06135	0.00298	20.56	<.0001	1.32039
YearBuilt	1	0.00280	0.00017730	15.78	<.0001	3.45048
GarageCars	1	0.05362	0.00542	9.90	<.0001	1.97466
Fireplaces	1	0.02564	0.00563	4.55	<.0001	1.57654
Kitchen1bvGr	1	-0.08596	0.01435	-5.99	<.0001	1.20885
TotalBsmtSF	1	0.00009957	0.00000981	10.15	<.0001	1.99065
BsmtFullBath	1	0.03107	0.00751	4.14	<.0001	1.82079
dKQ_3	1	0.06637	0.01432	4.63	<.0001	1.55233
dMS_3	1	0.14222	0.01774	8.02	<.0001	1.61941
dMS_6	1	0.04686	0.00996	4.70	<.0001	1.99615
BsmtFinSF1	1	0.00005478	0.00000994	5.51	<.0001	2.23212
d5	1	0.07685	0.01621	4.74	<.0001	1.58664
d10	1	0.14123	0.02370	5.96	<.0001	1.14365
d14	1	0.08660	0.01918	4.52	<.0001	1.21379
d18	1	0.03367	0.01099	3.06	0.0022	1.34603
d19	1	0.11665	0.01686	6.92	<.0001	1.15869
dE1_3	1	0.08913	0.01639	5.44	<.0001	1.05400
dF_3	1	0.04416	0.00863	5.12	<.0001	2.21283
dBQ_1	1	0.06669	0.01386	4.81	<.0001	1.73970
dBE_1	1	0.05502	0.01105	4.98	<.0001	1.21304
dGQ_1	1	0.24278	0.06436	3.77	0.0002	1.02729
dC1_2	1	-0.07382	0.01682	-4.39	<.0001	1.08728
dC1_7	1	-0.12182	0.03363	-3.62	0.0003	1.02262

Figure 8: Regression Model- Final Enriched Model

$$\begin{aligned}
\text{Predicted Log Sales Price} = & 1.7436 + 0.42439 * \text{Log}_{GrLivArea} + 0.08263 * \text{Log}_{LotArea} + \\
& 0.05411 * \text{OverallQual} + 0.06135 * \text{OverallCond} + 0.0028 * \text{YearBuilt} + 0.05362 * \\
& \text{GarageCars} + 0.02564 * \text{Fireplaces} - 0.08596 * \text{KitchenAbvGr} + 0.00009957 * \text{TotalBsmtSF} + \\
& 0.03107 * \text{BasementFullBath} + .06637 * \text{KitchenQual}_{Ex} + .14222 * \text{MSZoning}_{FV} + .04686 * \\
& \text{MSZoning}_{ResLowDense} + .00005478 * \text{BsmtFinSF1} + .07685 * \text{Neighborhood}_{NridgHt} + .14123 * \\
& \text{Neighborhood}_{StoneBr} + .08660 * \text{Neighborhood}_{NoRidge} + .03367 * \text{Neighborhood}_{CollgCr} + \\
& .11665 * \text{Neighborhood}_{Crawfor} + .08913 * \text{BrkFace} + .04416 * \text{Foundation}_{pouredConc} + \\
& .06669 * \text{BsmtQual}_{Ex} + .05502 * \text{BsmtExposure}_{Gd} + .24278 * \text{GarageQual}_{Ex} - .07382 * \\
& \text{Condition1}_{Artery} - .12182 * \text{Condition1}_{RRAdjacentEW}
\end{aligned}$$

Twenty-three out of 26 predictors are significant at the  $\alpha < .0001$  level. Regression diagnostics indicate that the assumptions of multiple linear regression are well satisfied. (Figure 9)

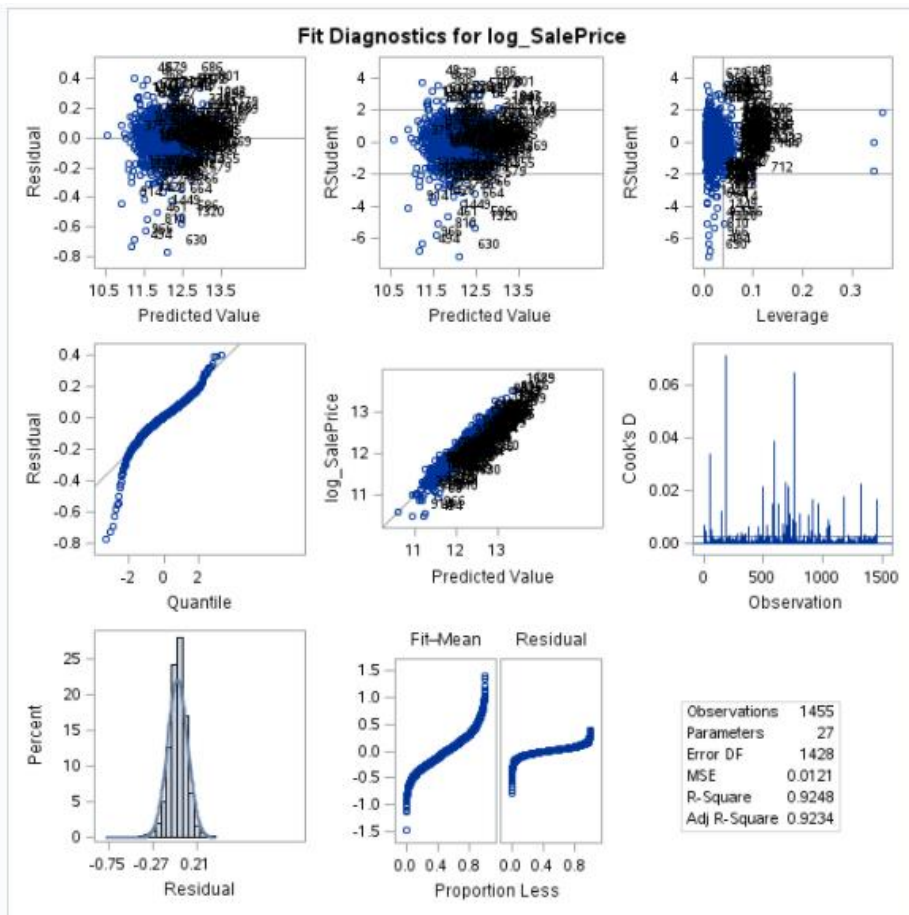
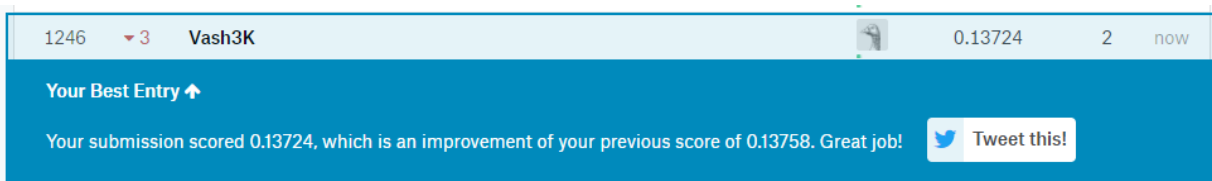


Figure 9: Residual Plots- Final Enriched Model

Enriching the model as explained above improved the Kaggle score slightly, to .13724:



A screenshot of a Kaggle submission leaderboard. The top bar shows a rank of 1246, a change of 3 (down), the username 'Vash3K', a profile icon, a score of 0.13724, 2 other submissions, and the time 'now'. Below this, a blue banner reads 'Your Best Entry ↑' and 'Your submission scored 0.13724, which is an improvement of your previous score of 0.13758. Great job!'. A 'Tweet this!' button is on the right.

1246	▼ 3	Vash3K		0.13724	2	now
------	-----	--------	--	---------	---	-----

**Your Best Entry ↑**

Your submission scored 0.13724, which is an improvement of your previous score of 0.13758. Great job!

[Tweet this!](#)

## **Conclusions**

In building the parsimonious, improved, and fully enriched models, broad applicability and predictive longevity were prioritized, and cross-validation procedures were used throughout the variable-screening process. Consequently, even with the improved and enriched models, results were easily interpretable, and consistent with many common-sense expectations when dealing with the question of housing sales price. It's logical that certain neighborhoods would have an especially strong influence on prediction; a casual look at the distribution of current sales prices in certain neighborhoods in Ames on Zillow.com confirmed the findings. Several other generalizations regarding desirable housing characteristics appeared to hold true: 1) people are willing to pay more for brick-faced houses; 2) likewise for basements with good exposure and of excellent quality; 3) similarly for garages of excellent quality; 4) as far as foundations are concerned, poured concrete is perceived as especially sturdy; and 5) taking traffic and noise disturbance factors into consideration, living on an arterial road or adjacent to an East-West running railway may have a negative impact on sale price. While the models constructed did not account for all the variance that other advanced algorithms might capture, the simplicity and statistically founded methodology of their construction ensures a high level of reliability when applied to future housing sales price prediction projects.

## APPENDIX

### **SAS Code – PARSIMONIOUS & IMPROVED MODELS**

```
FILENAME REFFILE '/home/jrasmusvorrath0/HousingTrain.csv';
```

```
PROC IMPORT DATAFILE=REFFILE  
              DBMS=CSV  
              OUT=Ames_train;  
              GETNAMES=YES;
```

```
RUN;
```

```
PROC CONTENTS DATA=Ames_train; RUN;  
*proc print data = Ames_train;* run;
```

```
/*  
data Ames_train_sub; set Ames_train;  
where id >= 1 AND id <= 1000; run;
```

\*Partition training set for presubmission cross-

validation, if desired;

```
*proc print data = Ames_train_sub;* run;
```

```
data Ames_train_sub2; set Ames_train;  
where id > 1000; run;
```

```
*proc print data = Ames_train_sub2;* run;
```

```
data Ames_train_val; set Ames_train_sub2;
```

\*Using second subset to create empty SalePrice column;

```
SalePrice = .;
```

```
*proc print data = Ames_train_sub;* run;
```

```
*proc print data = Ames_train_val;* run;
```

```
data Ames_train_test; set Ames_train_sub Ames_train_val; run;
```

\*Concatenating subsets to create alternative test set;

```
proc contents data = Ames_train_test; run;
```

```
*proc print data = Ames_train_test;* run;
```

```
*/
```

```
data Ames_train_missing1; set Ames_train;
```

\*Fixing character missing value codes;

```
array change _character_;  
do over change;  
  if change=-1 then change = .;  
end;
```

```

run;

*proc contents data = Ames_train_missing1;* run;

/*proc export data= Ames_train_missing1
                                *Verifying file contents
    outfile="/home/jrasmusvorrath0/Missingtest.csv"
    dbms=csv
    replace;
run;*/

data Ames_train_missing2; set Ames_train_missing1;
                                *Fixing numeric missing value codes;
array change _numeric_;
    do over change;
        if change=-1 then change = .;
    end;
run;

*proc contents data = Ames_train_missing2;* run;

/*proc export data= Ames_train_missing2
                                *Verifying file contents;
    outfile="/home/jrasmusvorrath0/Missingtest3.csv"
    dbms=csv
    replace;
run;*/

data Ames_train_missing3; set Ames_train_missing2;
                                *Fixing unusual 'Neighborhood' factor variable value
codes;
array change Neighborhood;
    do over change;
        if change="-1mes" then change = .;
    end;
run;

*proc contents data = Ames_train_missing3;* run;
*proc print data = Ames_train_missing3;* run;

proc means data = Ames_train_missing3 nmiss n; run;
                                *Verifying missing value counts;

proc mi data=Ames_train_missing3 seed=999 nimpute=1 out=Ames_train_missing_imputed;
                                *Imputing missing continuous variable values-- NB: Check of categorical missing
values;
fcs nbiter=10 reg(/details);
                                *indicated no logical need for imputation;

var LotFrontage GarageYrBlt MasVnrArea

```

```

MSSubClass OverallQual OverallCond LotArea GrLivArea
BsmtFinSF1 BsmtUnfSF TotalBsmtSF BsmtFullBath BsmtHalfBath
FullBath HalfBath BedroomAbvGr TotRmsAbvGrd
Fireplaces GarageCars GarageArea
WoodDeckSF OpenPorchSF
MoSold YrSold SalePrice;

run;

proc means data = Ames_train_missing_imputed nmiss n; run;
      *Verifying missing value counts;

*proc print data = Ames_train_missing_imputed;* run;

proc sgscatter data = Ames_train_missing_imputed;
      *EDA: Scatterplot matrix, anticipated continuous variables
of interest;
matrix SalePrice GrLivArea OverallQual OverallCond YearBuilt YearRemodAdd;
run;

proc sgscatter data = Ames_train_missing_imputed;
      *EDA: Scatterplot matrix, anticipated continuous variables
of interest;
matrix LotFrontage LotArea
      BsmtFinSF1 BsmtUnfSF TotalBsmtSF BsmtFullBath BsmtHalfBath;
run;

proc sgscatter data = Ames_train_missing_imputed;
      *EDA: Scatterplot matrix, anticipated continuous variables
of interest;
matrix FullBath HalfBath BedroomAbvGr TotRmsAbvGrd
      _1stFlrSF _2ndFlrSF "Kitchn-1bvGr";
run;

proc sgscatter data = Ames_train_missing_imputed;
      *EDA: Scatterplot matrix, anticipated continuous variables
of interest;
matrix Fireplaces GarageCars GarageArea YrSold;
run;

proc sgscatter data = Ames_train_missing_imputed;
      *EDA: Scatterplot matrix, unlikely continuous variables of
interest;
matrix _3SsnPorch ScreenPorch MiscVal MoSold;
run;

proc freq data = Ames_train_missing_imputed;
      *EDA: Frequency counts, possible categorical variables of
interest;

```

```

tables BsmtQual BsmtCond BsmtExposure BsmtFinType1 BsmtFinType2
      Electrical Utilities Heating HeatingQC CentralAir KitchenQual
      GarageType GarageYrBlt GarageFinish GarageQual GarageCond
      MsZoning Neighborhood Condition1 Condition2
      Street PavedDrive LotShape LandContour LotConfig LandSlope
      BldgType HouseStyle Roofstyle RoofMatl Foundation
      Exterior1st Exterior2nd MasVnrArea ExterQual ExterCond
      SaleType SaleCondition;

run;

data reg1; set Ames_train_missing_imputed;

log_SalePrice = log(SalePrice);
*EDA: log of SalePrice, given scatterplot

diagnosis;
log_GrLivArea = log(GrLivArea); run;
*EDA: log of GrLivArea, given scatterplot diagnosis;

*proc print data = reg1;* run;

data reg2; set reg1;
*Dummy-coding Neighborhoods & possible
interaction, for EDA & PROC REG;
if Neighborhood = "BrkSide" then d1 = 1; else d1 = 0;
if Neighborhood = "Edwards" then d2 = 1; else d2 = 0;
if Neighborhood = "NAMES" then d3 = 1; else d3 = 0;
if Neighborhood = "BrDale" then d4 = 1; else d4 = 0;
if Neighborhood = "NridgHt" then d5 = 1; else d5 = 0;
if Neighborhood = "OldTown" then d6 = 1; else d6 = 0;
if Neighborhood = "Sawyer" then d7 = 1; else d7 = 0;
if Neighborhood = "SawyerW" then d8 = 1; else d8 = 0;
if Neighborhood = "Somerst" then d9 = 1; else d9 = 0;
if Neighborhood = "StoneBr" then d10 = 1; else d10 = 0;
if Neighborhood = "Gilbert" then d11 = 1; else d11 = 0;
if Neighborhood = "Mitchel" then d12 = 1; else d12 = 0;
if Neighborhood = "NWAmes" then d13 = 1; else d13 = 0;
if Neighborhood = "NoRidge" then d14 = 1; else d14 = 0;
if Neighborhood = "ClearCr" then d15 = 1; else d15 = 0;
intA = d1*log_GrLivArea; intB = d2*log_GrLivArea; intC = d3*log_GrLivArea;
intD = d4*log_GrLivArea; intE = d5*log_GrLivArea; intF = d6*log_GrLivArea;
intG = d7*log_GrLivArea; intH = d8*log_GrLivArea; intI = d9*log_GrLivArea;
intJ = d10*log_GrLivArea; intK = d11*log_GrLivArea; intL = d12*log_GrLivArea;
intM = d13*log_GrLivArea; run;

*proc print data = reg2;* run;

data reg3; set reg2;
*Dummy-coding potential categorical
variable levels of interest, for PROC REG;

```

```

if BldgType = "1Fam" then dBT_1 = 1; else dBT_1 = 0;
if BldgType = "2fmCon" then dBT_2 = 1; else dBT_2 = 0;
if BldgType = "Duplex" then dBT_3 = 1; else dBT_3 = 0;
if BldgType = "TwnhsE" then dBT_4 = 1; else dBT_4 = 0;
if BldgType = "Twnhs" then dBT_5 = 1; else dBT_5 = 0;
if Condition2 = "Norm" then dC2_1 = 1; else dC2_1 = 0;
if Condition2 = "Artery" then dC2_2 = 1; else dC2_2 = 0;
if Condition2 = "RRNn" then dC2_3 = 1; else dC2_3 = 0;
if Condition2 = "Feedr" then dC2_4 = 1; else dC2_4 = 0;
if Condition2 = "PosN" then dC2_5 = 1; else dC2_5 = 0;
if Condition2 = "PosA" then dC2_6 = 1; else dC2_6 = 0;
if Condition2 = "RRAn" then dC2_7 = 1; else dC2_7 = 0;
if Condition2 = "RR Ae" then dC2_8 = 1; else dC2_8 = 0;
if SaleCondition = "Normal" then dSC_1 = 1; else dSC_1 = 0;
if SaleCondition = "Abnormal" then dSC_2 = 1; else dSC_2 = 0;
if SaleCondition = "Partial" then dSC_3 = 1; else dSC_3 = 0;
if SaleCondition = "AdjLand" then dSC_4 = 1; else dSC_4 = 0;
if SaleCondition = "Alloca" then dSC_5 = 1; else dSC_5 = 0;
if SaleCondition = "Family" then dSC_6 = 1; else dSC_6 = 0;
if KitchenQual = "Gd" then dKQ_1 = 1; else dKQ_1 = 0;
if KitchenQual = "TA" then dKQ_2 = 1; else dKQ_2 = 0;
if KitchenQual = "Ex" then dKQ_3 = 1; else dKQ_3 = 0;
if KitchenQual = "Fa" then dKQ_4 = 1; else dKQ_4 = 0;
if KitchenQual = "Po" then dKQ_5 = 1; else dKQ_5 = 0;
run;

```

```

proc glmselect data = reg3 plots = (CriterionPanel ASE ASEPlot) seed = 99;
    *1st-Pass Hypothesis, Automated Selection and CV, using Neighborhoods best
meeting LR Assumptions;
partition fraction(validate=0.3 test=0.2);

```

\*NB: exploring potential categorical variable levels

```

of interest;
class Neighborhood MSZoning Condition1 Condition2 Street
    Utilities Heating HeatingQC CentralAir Electrical KitchenQual FireplaceQu
    LotFrontage LotShape LotConfig LandSlope LandContour
    BldgType HouseStyle RoofStyle RoofMatl Foundation
    Exterior1st Exterior2nd ExterQual ExterCond MasVnrType
    BsmtQual BsmtCond BsmtExposure BsmtFinType1 BsmtFinType2
    GarageType GarageFinish GarageQual GarageCond PavedDrive
    SaleType SaleCondition / param = glm;
where Neighborhood in ("BrkSide", "Edwards", "NAmes", "BrDale", "NridgHt", "OldTown",
    "Sawyer",
    "SawyerW", "Somerst", "StoneBr", "Gilbert", "Mitchel", "NWAmes");
model log_SalePrice = log_GrLivArea LotArea OverallCond OverallQual YearBuilt BsmtFullBath
    BsmtFinSF1
    d1 d2 d3 d4 d6 d7 d8 d9 d11 d12 d13 dBT_1 dBT_5 dC2_5 dC2_3 dKQ_2
    dKQ_3 dSC_1 dSC_2 dSC_3 dSC_6
/ selection = lasso(choose = AIC stop = AICC) details = steps showpvalues;
run;

```



```
proc reg data = reg3 plots(label)=(CooksD RStudentByLeverage) outest= reg_est1 edf;
    *1st-Pass Regression Analysis, exploring automated selection output;
model log_SalePrice = log_GrLivArea BsmtFullBath OverallQual YearBuilt
    dBT_5 dKQ_2 dKQ_3 dSC_3 / VIF influence adjrsq;
run; quit;
```

```
data reg3a; set reg3;
                                                    *Removing unusually high leverage
observations;
if _n_ = 1299 then delete;
if _n_ = 524 then delete;
run;
```

```
proc reg data = reg3a plots(label) outest= reg_est2 edf;
    *Registering effect of high leverage removal;
model log_SalePrice = log_GrLivArea BsmtFullBath OverallQual YearBuilt
    dBT_5 dKQ_2 dKQ_3 dSC_3 / VIF;
run; quit;
```

```
proc glm data = reg3a plots = all outstat = reg_est3;
    *Registering PROC GLM output statistics;
model log_SalePrice = log_GrLivArea BsmtFullBath OverallQual YearBuilt
    dBT_5 dKQ_2 dKQ_3 dSC_3 / cli solution;
output out = results p = predict;
run;
```

```
proc sgscatter data = reg3a;
                                                    *EDA: Scatterplot matrix of automatically selected
continuous variables;
matrix log_SalePrice log_GrLivArea OverallQual YearBuilt;
run;
```

```
proc reg data = reg3a plots(label) outest= reg_est4 edf;
    *2nd-Pass Hypothesis, including additional continuous variables by
inductive reasoning;
model log_SalePrice = log_GrLivArea OverallQual OverallCond YearBuilt
    LotFrontage LotArea GarageCars Fireplaces
    BsmtFullBath TotalBsmtSF BsmtUnfSF
    dBT_5 dKQ_2 dKQ_3 dSC_3 / VIF;
run; quit;
```

```

proc reg data = reg3a plots(label) outest= reg_est5 edf;
    *Refining list of continuous variables, paring down categorical levels not
broadly applicable;
model log_SalePrice = log_GrLivArea OverallQual OverallCond YearBuilt
    LotArea GarageCars Fireplaces "Kitche-1bvGr"n
    TotalBsmtSF BsmtFullBath
    dBT_5 dKQ_3 dSC_3 / VIF;
run; quit;

*proc contents data = reg3a;* run;

data reg4; set reg3a;
    *log of LotArea, given regression diagnostic

plots;
log_LotArea = log(LotArea); run;

proc reg data = reg4 plots(label) outest= reg_est6 edf;
    *Registering effect of log transform of LotArea, further paring
down of dummy categorical levels;
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
    GarageCars Fireplaces "Kitche-1bvGr"n
    TotalBsmtSF BsmtFullBath
    dKQ_3 dSC_3 / VIF;
run; quit;

proc glm data = reg4 plots = all outstat = reg_est7;
    *Registering PROC GLM output statistics;
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
    GarageCars Fireplaces "Kitche-1bvGr"n
    TotalBsmtSF BsmtFullBath
    dKQ_3 dSC_3 / cli solution;
output out = results2 p = predict2;
run;

proc sgscatter data = reg4;
    *EDA: Scatterplot matrix of additional
continuous variables of interest;
matrix log_SalePrice log_GrLivArea log_LotArea OverallQual YearBuilt;
run;

proc reg data = reg4 plots(label) outest= reg_est8 edf;
    *Parsimonious preferred model, paring down all dummy categorial
levels not broadly applicable;
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
    GarageCars Fireplaces "Kitche-1bvGr"n
    TotalBsmtSF BsmtFullBath / VIF;
run; quit;

```

```
proc glm data = reg4 plots = all outstat = reg_est9;
    *Registering PROC GLM output statistics;
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
    GarageCars Fireplaces "Kitche-1bvGr"n
    TotalBsmtSF BsmtFullBath / cli solution;
output out = results3 p = predict3;
run;
```

```
proc means data = reg4;
    *Registering distributions of
parsimonious preferred predictors;
var log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
    GarageCars Fireplaces "Kitche-1bvGr"n
    TotalBsmtSF BsmtFullBath;
run;
```

```
data reg5; set reg4;
    *3rd-Pass Analysis, dummy-coding
potentially applicable categorical variables, selected by inductive reasoning;
if MSZoning = "A" then dMS_1 = 1; else dMS_1 = 0;
if MSZoning = "C" then dMS_2 = 1; else dMS_2 = 0;
if MSZoning = "FV" then dMS_3 = 1; else dMS_3 = 0;
if MSZoning = "I" then dMS_4 = 1; else dMS_4 = 0;
if MSZoning = "RH" then dMS_5 = 1; else dMS_5 = 0;
if MSZoning = "RL" then dMS_6 = 1; else dMS_6 = 0;
if MSZoning = "RP" then dMS_7 = 1; else dMS_7 = 0;
if MSZoning = "RM" then dMS_8 = 1; else dMS_8 = 0;

if LotConfig = "Inside" then dLC_1 = 1; else dLC_1 = 0;
if LotConfig = "Corner" then dLC_2 = 1; else dLC_2 = 0;
if LotConfig = "CulDSac" then dLC_3 = 1; else dLC_3 = 0;
if LotConfig = "FR2" then dLC_4 = 1; else dLC_4 = 0;
if LotConfig = "FR3" then dLC_5 = 1; else dLC_5 = 0;

if ExterQual = "Ex" then dEQ_1 = 1; else dEQ_1 = 0;
if ExterQual = "Gd" then dEQ_2 = 1; else dEQ_2 = 0;
if ExterQual = "TA" then dEQ_3 = 1; else dEQ_3 = 0;
if ExterQual = "Fa" then dEQ_4 = 1; else dEQ_4 = 0;
if ExterQual = "Po" then dEQ_5 = 1; else dEQ_5 = 0;

if ExterCond = "Ex" then dEC_1 = 1; else dEC_1 = 0;
if ExterCond = "Gd" then dEC_2 = 1; else dEC_2 = 0;
if ExterCond = "TA" then dEC_3 = 1; else dEC_3 = 0;
if ExterCond = "Fa" then dEC_4 = 1; else dEC_4 = 0;
if ExterCond = "Po" then dEC_5 = 1; else dEC_5 = 0;

if HeatingQC = "Ex" then dHQ_1 = 1; else dHQ_1 = 0;
```

```

if HeatingQC = "Gd" then dHQ_2 = 1; else dHQ_2 = 0;
if HeatingQC = "TA" then dHQ_3 = 1; else dHQ_3 = 0;
if HeatingQC = "Fa" then dHQ_4 = 1; else dHQ_4 = 0;
if HeatingQC = "Po" then dHQ_5 = 1; else dHQ_5 = 0;

```

```

if SaleType = "WD" then dST_1 = 1; else dST_1 = 0;
if SaleType = "CWD" then dST_2 = 1; else dST_2 = 0;
if SaleType = "VWD" then dST_3 = 1; else dST_3 = 0;
if SaleType = "New" then dST_4 = 1; else dST_4 = 0;
if SaleType = "COD" then dST_5 = 1; else dST_5 = 0;
if SaleType = "Con" then dST_6 = 1; else dST_6 = 0;
if SaleType = "ConLw" then dST_7 = 1; else dST_7 = 0;
if SaleType = "ConLI" then dST_8 = 1; else dST_8 = 0;
if SaleType = "ConLD" then dST_9 = 1; else dST_9 = 0;
if SaleType = "Oth" then dST_10 = 1; else dST_10 = 0;

```

```

if Exterior2nd = "AsbShng" then dEX2_1 = 1; else dEX2_1 = 0;
if Exterior2nd = "AsphShn" then dEX2_2 = 1; else dEX2_2 = 0;
if Exterior2nd = "BrkComm" then dEX2_3 = 1; else dEX2_3 = 0;
if Exterior2nd = "BrkFace" then dEX2_4 = 1; else dEX2_4 = 0;
if Exterior2nd = "CBlock" then dEX2_5 = 1; else dEX2_5 = 0;
if Exterior2nd = "CemntBd" then dEX2_6 = 1; else dEX2_6 = 0;
if Exterior2nd = "HdBoard" then dEX2_7 = 1; else dEX2_7 = 0;
if Exterior2nd = "ImStucc" then dEX2_8 = 1; else dEX2_8 = 0;
if Exterior2nd = "MetalSd" then dEX2_9 = 1; else dEX2_9 = 0;
if Exterior2nd = "Other" then dEX2_10 = 1; else dEX2_10 = 0;
if Exterior2nd = "PlyWood" then dEX2_11 = 1; else dEX2_11 = 0;
if Exterior2nd = "PreCast" then dEX2_12 = 1; else dEX2_12 = 0;
if Exterior2nd = "Stone" then dEX2_13 = 1; else dEX2_13 = 0;
if Exterior2nd = "Stucco" then dEX2_14 = 1; else dEX2_14 = 0;
if Exterior2nd = "VinylSd" then dEX2_15 = 1; else dEX2_15 = 0;
if Exterior2nd = "Wd Sdng" then dEX2_16 = 1; else dEX2_16 = 0;
if Exterior2nd = "WdShng" then dEX2_17 = 1; else dEX2_17 = 0;
run;

```

```

proc glmselect data = reg5 plots = (CriterionPanel ASE ASEPlot) seed = 444;
    *3rd-Pass Analysis, Automated Selection and CV, using additional
categorical levels;
partition fraction(validate=0.3 test=0.2);
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
    GarageCars Fireplaces "Kitch-1bvGr"n
    TotalBsmtSF BsmtFullBath
    dMS_1 dMS_2 dMS_3 dMS_4 dMS_5 dMS_6 dMS_7 dMS_8
    dSC_1 dSC_2 dSC_3 dSC_4 dSC_5 dSC_6
    dLC_1 dLC_2 dLC_3 dLC_4 dLC_5
    dEQ_1 dEQ_2 dEQ_3 dEQ_4 dEQ_5
    dEC_1 dEC_2 dEC_3 dEC_4 dEC_5
    dHQ_1 dHQ_2 dHQ_3 dHQ_4 dHQ_5
    dKQ_1 dKQ_2 dKQ_3 dKQ_4 dKQ_5

```

```

dST_1 dST_2 dST_3 dST_4 dST_5
      dST_6 dST_7 dST_8 dST_9 dST_10
dEX2_1 dEX2_2 dEX2_3 dEX2_4 dEX2_5
      dEX2_6 dEX2_7 dEX2_8 dEX2_9 dEX2_10
      dEX2_11 dEX2_12 dEX2_13 dEX2_14 dEX2_15 dEX2_16 dEX2_17
/ selection = lasso(choose = AIC stop = AICC) details = steps showpvalues;
run;

proc reg data = reg5 plots(label)=(CooksD) outest= reg_est10 edf;
      *3rd-Pass Hypothesis, enriching parsimonious model, consulting
automated selection and chosen by inductive reasoning;
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
      GarageCars Fireplaces "Kitche-1bvGr"n
      TotalBsmtSF BsmtFullBath
      dKQ_3 dSC_1 dSC_3 dMS_2 dMS_3 dMS_6 / VIF;
run; quit;

data reg6; set reg5;
                                                    *Removing unusually high leverage
observation;
if _n_ = 31 then delete;
run;

proc reg data = reg6 plots(label)=(CooksD) outest= reg_est11 edf;
      *Registering effect of high leverage removal;
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
      GarageCars Fireplaces "Kitche-1bvGr"n
      TotalBsmtSF BsmtFullBath
      dKQ_3 dSC_1 dSC_3 dMS_2 dMS_3 dMS_6 / VIF;
run; quit;

proc glm data = reg6 plots = all outstat = reg_est12;
      *Registering PROC GLM output statistics;
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
      GarageCars Fireplaces "Kitche-1bvGr"n
      TotalBsmtSF BsmtFullBath
      dKQ_3 dSC_1 dSC_3 dMS_2 dMS_3 dMS_6 / cli solution;
output out = results4 p = predict4;
run;

proc glmselect data = reg6 plots = (CriterionPanel ASE ASEPlot) seed = 777;
      *4th-Pass Analysis, Automated Selection and CV, enriching model with
remaining variables of interest;
partition fraction(validate=0.3 test=0.2);

class Neighborhood MSZoning Street LotFrontage LotShape LandContour Utilities LotConfig

```

```

LandSlope Condition1 Condition2 BldgType HouseStyle RoofStyle RoofMatl Exterior1st
Exterior2nd
MasVnrType ExterQual ExterCond Foundation BsmtQual BsmtCond BsmtExposure
BsmtFinType1 BsmtFinType2
Heating HeatingQC CentralAir Electrical KitchenQual FireplaceQu GarageType GarageFinish
GarageQual GarageCond PavedDrive SaleType SaleCondition / param = glm;

```

```

model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
GarageCars Fireplaces "Kitche-1bvGr"n
TotalBsmtSF BsmtFullBath
dKQ_3 dSC_1 dSC_3 dMS_2 dMS_3 dMS_6

```

```

LotFrontage MasVnrArea MSSubClass BsmtFinSF1 BsmtUnfSF BsmtHalfBath FullBath
HalfBath BedroomAbvGr
TotRmsAbvGrd GarageArea WoodDeckSF OpenPorchSF MoSold YrSold BsmtQual BsmtCond
BsmtExposure BsmtFinType1 BsmtFinType2
Electrical FireplaceQu GarageType GarageYrBlt GarageFinish GarageQual GarageCond Street
LotShape LandContour
Utilities LotConfig LandSlope Neighborhood Condition1 Condition2 BldgType HouseStyle
Roofstyle RoofMatl
Exterior1st Exterior2nd ExterQual ExterCond Foundation Heating HeatingQC CentralAir
PavedDrive SaleType

```

```

/ selection = stepwise(choose = AIC stop = AICC include = 16) details = steps showpvalues;
run;

```

```

proc reg data = reg6 plots(label)=(CooksD) outest= reg_est13 edf;
*Rich Preferred Model, including 2 additional variables;
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
GarageCars Fireplaces "Kitche-1bvGr"n
TotalBsmtSF BsmtFullBath
dKQ_3 dSC_1 dSC_3 dMS_2 dMS_3 dMS_6
dHQ_1 BsmtFinSF1 / VIF;
run; quit;

```

```

proc glm data = reg6 plots = all outstat = reg_est14;
*Registering PROC GLM output statistics for rich preferred model;
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
GarageCars Fireplaces "Kitche-1bvGr"n
TotalBsmtSF BsmtFullBath
dKQ_3 dSC_1 dSC_3 dMS_2 dMS_3 dMS_6
dHQ_1 BsmtFinSF1 / cli solution;
output out = results5 p = predict5;
run;

```

## SAS Code – KAGGLE SUBMISSION 1

```
FILENAME REFFILE '/home/jrasmusvorrath0/HousingTest.csv';
```

```
PROC IMPORT DATAFILE=REFFILE  
              DBMS=CSV  
              OUT=Ames_test;  
              GETNAMES=YES;
```

```
RUN;
```

```
PROC CONTENTS DATA=Ames_test; RUN;
```

```
data Ames_test2 (drop=old); set Ames_test (rename=(LotFrontage=old));  
  LotFrontage = input(old, 8.0);  
run;
```

```
proc contents data = Ames_test2; run;
```

```
data Ames_test3(drop= Alley PoolQC Fence MiscFeature); set Ames_test2;  
run;
```

```
proc contents data = Ames_test3; run;
```

```
data Ames_test4; set Ames_test3(rename=(Functional="Function-1" KitchenAbvGr="Kitchen-  
1bvGr"));  
run;
```

```
proc contents data = Ames_test4; run;
```

```
data Ames_test5; set Ames_test4;  
if SaleCondition = "Normal" then dSC_1 = 1; else dSC_1 = 0;  
if SaleCondition = "Partial" then dSC_3 = 1; else dSC_3 = 0;  
if KitchenQual = "Ex" then dKQ_3 = 1; else dKQ_3 = 0;  
if MSZoning = "C" then dMS_2 = 1; else dMS_2 = 0;  
if MSZoning = "FV" then dMS_3 = 1; else dMS_3 = 0;  
if MSZoning = "RL" then dMS_6 = 1; else dMS_6 = 0;  
if HeatingQC = "Ex" then dHQ_1 = 1; else dHQ_1 = 0;  
log_SalePrice = log(SalePrice);  
log_GrLivArea = log(GrLivArea);  
log_LotArea = log(LotArea);  
run;
```

```
data Ames_train2; set Ames_train;  
if _n_ = 31 then delete;  
if _n_ = 524 then delete;  
if SaleCondition = "Normal" then dSC_1 = 1; else dSC_1 = 0;  
if SaleCondition = "Partial" then dSC_3 = 1; else dSC_3 = 0;  
if KitchenQual = "Ex" then dKQ_3 = 1; else dKQ_3 = 0;  
if MSZoning = "C" then dMS_2 = 1; else dMS_2 = 0;  
if MSZoning = "FV" then dMS_3 = 1; else dMS_3 = 0;
```

```

if MSZoning = "RL" then dMS_6 = 1; else dMS_6 = 0;
if HeatingQC = "Ex" then dHQ_1 = 1; else dHQ_1 = 0;
log_SalePrice = log(SalePrice);
log_GrLivArea = log(GrLivArea);
log_LotArea = log(LotArea);
run;

data Ames_train3; set Ames_train2;
if _n_ = 1299 then delete;
run;

data Ames_full; set Ames_train3 Ames_test5;
run;

proc contents data = Ames_full; run;

proc print data = Ames_full;
run;

proc glm data = Ames_full plots = all outstat = reg_est15;

model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
      GarageCars Fireplaces "Kitch-1bvGr"n
      TotalBsmtSF BsmtFullBath
      dKQ_3 dSC_1 dSC_3 dMS_2 dMS_3 dMS_6
      dHQ_1 BsmtFinSF1 / cli solution;
output out = result6 p = predict6;
run;

proc print data = result6;
run;

proc contents data = result6; run;

data finalp; set result6 (where=(Id > 1460));
if exp(predict6) <= 30000 then Sale_P = 40000;
else Sale_P = exp(predict6);
keep Id Sale_P;
rename Sale_P = SalePrice;
run;

proc print data = finalp;
run;

proc export data= finalp
  outfile="/home/jrasmusvorrath0/Ames_Kaggle_Submission.csv"
  dbms=csv
  replace;
run;

```



## SAS Code – ENRICHED FULL MODEL (continued from above)

```
data reg7; set reg6;
```

\*5th-Pass Analysis, dummy coding

```
additional categorical variable levels;
```

```
if Neighborhood = "Blmngtn" then d16 = 1; else d16 = 0;
if Neighborhood = "Blueste" then d17 = 1; else d17 = 0;
if Neighborhood = "CollgCr" then d18 = 1; else d18 = 0;
if Neighborhood = "Crawfor" then d19 = 1; else d19 = 0;
if Neighborhood = "IDOTRR" then d20 = 1; else d20 = 0;
if Neighborhood = "MeadowV" then d21 = 1; else d21 = 0;
if Neighborhood = "NPkVill" then d22 = 1; else d22 = 0;
if Neighborhood = "SWISU" then d23 = 1; else d23 = 0;
if Neighborhood = "Timber" then d24 = 1; else d24 = 0;
if Street = "Gravel" then dS_1 = 1; else dS_1 = 0;
if LotShape = "IR1" then dLS_1 = 1; else dLS_1 = 0;
if LotShape = "IR2" then dLS_2 = 1; else dLS_2 = 0;
if LotShape = "IR3" then dLS_3 = 1; else dLS_3 = 0;
if LandContour = "Bnk" then dLaC_1 = 1; else dLaC_1 = 0;
if LandContour = "HLS" then dLaC_2 = 1; else dLaC_2 = 0;
if LandContour = "Low" then dLaC_3 = 1; else dLaC_3 = 0;
if LandSlope = "Gtl" then dLaS_1 = 1; else dLaS_1 = 0;
if LandSlope = "Mod" then dLaS_2 = 1; else dLaS_2 = 0;
if Exterior1st = "AsbShng" then dE1_1 = 1; else dE1_1 = 0;
if Exterior1st = "BrkComm" then dE1_2 = 1; else dE1_2 = 0;
if Exterior1st = "BrkFace" then dE1_3 = 1; else dE1_3 = 0;
if Exterior1st = "CemntBd" then dE1_4 = 1; else dE1_4 = 0;
if Exterior1st = "HdBoard" then dE1_5 = 1; else dE1_5 = 0;
if Exterior1st = "MetalSd" then dE1_6 = 1; else dE1_6 = 0;
if Exterior1st = "Other" then dE1_7 = 1; else dE1_7 = 0;
if Exterior1st = "Plywood" then dE1_8 = 1; else dE1_8 = 0;
if Exterior1st = "PreCast" then dE1_9 = 1; else dE1_9 = 0;
if Exterior1st = "Stone" then dE1_10 = 1; else dE1_10 = 0;
if Exterior1st = "VinylSd" then dE1_11 = 1; else dE1_11 = 0;
if Exterior1st = "Wd Sdng" then dE1_12 = 1; else dE1_12 = 0;
if Foundation = "BrkTil" then dF_1 = 1; else dF_1 = 0;
if Foundation = "CBlock" then dF_2 = 1; else dF_2 = 0;
if Foundation = "PConc" then dF_3 = 1; else dF_3 = 0;
if Foundation = "Slab" then dF_4 = 1; else dF_4 = 0;
if Heating = "Floor" then dH_1 = 1; else dH_1 = 0;
if Heating = "GasA" then dH_2 = 1; else dH_2 = 0;
if Heating = "GasW" then dH_3 = 1; else dH_3 = 0;
if Heating = "Grav" then dH_4 = 1; else dH_4 = 0;
if Heating = "OthW" then dH_5 = 1; else dH_5 = 0;
if CentralAir = "N" then dCA_1 = 1; else dCA_1 = 0;
if PavedDrive = "P" then dPD_1 = 1; else dPD_1 = 0;
if PavedDrive = "N" then dPD_2 = 1; else dPD_2 = 0;
if BsmtQual = "Ex" then dBQ_1 = 1; else dBQ_1 = 0;
if BsmtQual = "Gd" then dBQ_2 = 1; else dBQ_2 = 0;
if BsmtQual = "Fa" then dBQ_3 = 1; else dBQ_3 = 0;
if BsmtQual = "Po" then dBQ_4 = 1; else dBQ_4 = 0;
```

```

if BsmtQual = "NA" then dBQ_5 = 1; else dBQ_5 = 0;
if BsmtCond = "Ex" then dBC_1 = 1; else dBC_1 = 0;
if BsmtCond = "Gd" then dBC_2 = 1; else dBC_2 = 0;
if BsmtCond = "Fa" then dBC_3 = 1; else dBC_3 = 0;
if BsmtCond = "Po" then dBC_4 = 1; else dBC_4 = 0;
if BsmtCond = "NA" then dBC_5 = 1; else dBC_5 = 0;
if BsmtExposure = "Gd" then dBE_1 = 1; else dBE_1 = 0;
if BsmtExposure = "Av" then dBE_2 = 1; else dBE_2 = 0;
if BsmtExposure = "Mn" then dBE_3 = 1; else dBE_3 = 0;
if BsmtExposure = "NA" then dBE_4 = 1; else dBE_4 = 0;
if BsmtFinType1 = "GLQ" then dBFT_1 = 1; else dBFT_1 = 0;
if BsmtFinType1 = "ALQ" then dBFT_2 = 1; else dBFT_2 = 0;
if BsmtFinType1 = "BLQ" then dBFT_3 = 1; else dBFT_3 = 0;
if BsmtFinType1 = "Rec" then dBFT_4 = 1; else dBFT_4 = 0;
if BsmtFinType1 = "LwQ" then dBFT_5 = 1; else dBFT_5 = 0;
if BsmtFinType1 = "NA" then dBFT_6 = 1; else dBFT_6 = 0;
if BsmtFinType2 = "GLQ" then dBFT2_1 = 1; else dBFT2_1 = 0;
if BsmtFinType2 = "ALQ" then dBFT2_2 = 1; else dBFT2_2 = 0;
if BsmtFinType2 = "BLQ" then dBFT2_3 = 1; else dBFT2_3 = 0;
if BsmtFinType2 = "Rec" then dBFT2_4 = 1; else dBFT2_4 = 0;
if BsmtFinType2 = "LwQ" then dBFT2_5 = 1; else dBFT2_5 = 0;
if BsmtFinType2 = "NA" then dBFT2_6 = 1; else dBFT2_6 = 0;
if GarageType = "2Types" then dGT_1 = 1; else dGT_1 = 0;
if GarageType = "Attchd" then dGT_2 = 1; else dGT_2 = 0;
if GarageType = "Basment" then dGT_3 = 1; else dGT_3 = 0;
if GarageType = "BuiltIn" then dGT_4 = 1; else dGT_4 = 0;
if GarageType = "CarPort" then dGT_5 = 1; else dGT_5 = 0;
if GarageType = "NA" then dGT_6 = 1; else dGT_6 = 0;
if GarageFinish = "Fin" then dGF_1 = 1; else dGF_1 = 0;
if GarageFinish = "RFn" then dGF_2 = 1; else dGF_2 = 0;
if GarageFinish = "NA" then dGF_3 = 1; else dGF_3 = 0;
if GarageQual = "Ex" then dGQ_1 = 1; else dGQ_1 = 0;
if GarageQual = "Gd" then dGQ_2 = 1; else dGQ_2 = 0;
if GarageQual = "Fa" then dGQ_3 = 1; else dGQ_3 = 0;
if GarageQual = "Po" then dGQ_4 = 1; else dGQ_4 = 0;
if GarageQual = "NA" then dGQ_5 = 1; else dGQ_5 = 0;
if GarageCond = "Gd" then dGC_1 = 1; else dGC_1 = 0;
if GarageCond = "Fa" then dGC_2 = 1; else dGC_2 = 0;
if GarageCond = "Po" then dGC_3 = 1; else dGC_3 = 0;
if GarageCond = "NA" then dGC_4 = 1; else dGC_4 = 0;
if Electrical = "FuseA" then dE_1 = 1; else dE_1 = 0;
if Electrical = "FuseF" then dE_2 = 1; else dE_2 = 0;
if Electrical = "FuseP" then dE_3 = 1; else dE_3 = 0;
if HouseStyle = "1Story" then dHS_1 = 1; else dHS_1 = 0;
if HouseStyle = "1.5Fin" then dHS_2 = 1; else dHS_2 = 0;
if HouseStyle = "1.5Unf" then dHS_3 = 1; else dHS_3 = 0;
if HouseStyle = "2Story" then dHS_4 = 1; else dHS_4 = 0;
if HouseStyle = "2.5Fin" then dHS_5 = 1; else dHS_5 = 0;
if HouseStyle = "2.5Unf" then dHS_6 = 1; else dHS_6 = 0;
if HouseStyle = "SFoyer" then dHS_7 = 1; else dHS_7 = 0;
if FireplaceQu = "Ex" then dFQ_1 = 1; else dFQ_1 = 0;

```

```

if FireplaceQu = "Gd" then dFQ_2 = 1; else dFQ_2 = 0;
if Condition1 = "Norm" then dC1_1 = 1; else dC1_1 = 0;
if Condition1 = "Artery" then dC1_2 = 1; else dC1_2 = 0;
if Condition1 = "Feedr" then dC1_3 = 1; else dC1_3 = 0;
if Condition1 = "PosN" then dC1_4 = 1; else dC1_4 = 0;
if Condition1 = "PosA" then dC1_5 = 1; else dC1_5 = 0;
if Condition1 = "RRAn" then dC1_6 = 1; else dC1_6 = 0;
if Condition1 = "RRAe" then dC1_7 = 1; else dC1_7 = 0;
if Condition1 = "RRNe" then dC1_8 = 1; else dC1_8 = 0;
if RoofStyle = "Flat" then dRS_1 = 1; else dRS_1 = 0;
if RoofStyle = "Gable" then dRS_2 = 1; else dRS_2 = 0;
if RoofStyle = "Gambrel" then dRS_3 = 1; else dRS_3 = 0;
if RoofStyle = "Hip" then dRS_4 = 1; else dRS_4 = 0;
if RoofStyle = "Mansard" then dRS_5 = 1; else dRS_5 = 0;
if RoofMatl = "ClyTile" then dRM_1 = 1; else dRM_1 = 0;
if RoofMatl = "CompShg" then dRM_2 = 1; else dRM_2 = 0;
if RoofMatl = "Membran" then dRM_3 = 1; else dRM_3 = 0;
if RoofMatl = "Metal" then dRM_4 = 1; else dRM_4 = 0;
if RoofMatl = "Tar&Grv" then dRM_5 = 1; else dRM_5 = 0;
if RoofMatl = "WdShake" then dRM_6 = 1; else dRM_6 = 0;
run;

```

```
proc contents data = reg7; run;
```

\*Verifying dummy coding procedure;

```
proc freq data = reg7;
```

\*Verifying missing value contents of

```

unsuitable predictor;
tables LandContour*dLaC_1;
run;

```

```

proc glmselect data = reg7 plots = (CriterionPanel ASE ASEPlot) seed = 2222;
    *5th-Pass Analysis, Stepwise Automated Selection and CV, enriching model
with remaining variables of interest;
partition fraction(validate=0.3 test=0.2);

```

```

model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
    GarageCars Fireplaces "Kitch-1bvGr"n
    TotalBsmtSF BsmtFullBath
    dKQ_3 dSC_1 dSC_3 dMS_2 dMS_3 dMS_6
    dHQ_1 BsmtFinSF1

```

```

d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24
dS_1
dLS_1 dLS_2 dLS_3
/*
dLaC_1 dLaC_2 dLaC_3

```

\*NB: Missing values throw

```
error;
```

```

*/
dLaS_1 dLaS_2
dE1_1 dE1_2 dE1_3 dE1_4 dE1_5 dE1_6 dE1_7 dE1_8 dE1_9 dE1_10 dE1_11 dE1_12
dF_1 dF_2 dF_3 dF_4
dH_1 dH_2 dH_3 dH_4 dH_5
dCA_1
dPD_1 dPD_2
dBQ_1 dBQ_2 dBQ_3 dBQ_4 dBQ_5
dBC_1 dBC_2 dBC_3 dBC_4 dBC_5
dBE_1 dBE_2 dBE_3 dBE_4
dBFT_1 dBFT_2 dBFT_3 dBFT_4 dBFT_5 dBFT_6
dBFT2_1 dBFT2_2 dBFT2_3 dBFT2_4 dBFT2_5 dBFT2_6
dGT_1 dGT_2 dGT_3 dGT_4 dGT_5 dGT_6
dGF_1 dGF_2 dGF_3
dGQ_1 dGQ_2 dGQ_3 dGQ_4 dGQ_5
dGC_1 dGC_2 dGC_3 dGC_4
dE_1 dE_2 dE_3
dHS_1 dHS_2 dHS_3 dHS_4 dHS_5 dHS_6 dHS_7
dFQ_1 dFQ_2
dC1_1 dC1_2 dC1_3 dC1_4 dC1_5 dC1_6 dC1_7 dC1_8
dRS_1 dRS_2 dRS_3 dRS_4 dRS_5
dRM_1 dRM_2 dRM_3 dRM_4 dRM_5 dRM_6
dLC_2 dLC_3 dLC_4 dLC_5
dHQ_2 dHQ_4
dST_2 dST_3 dST_4 dST_5 dST_6 dST_7 dST_8 dST_9 dST_10
dEC_1 dEC_2 dEC_4
dEQ_1 dEQ_2 dEQ_4 dEQ_5
dBT_1 dBT_2 dBT_3 dBT_5
dC2_1 dC2_2 dC2_4 dC2_5 dC2_6

```

```

/ selection = stepwise(choose = AIC stop = AICC include = 18) details = steps showpvalues;
run;

```

```

data reg8; set reg7;

```

\*Removing unusually high leverage

```

observations, based on PROC REG below;
if _n_ = 410 then delete;
if _n_ = 999 then delete;
run;

```

```

data reg9; set reg8;

```

\*Removing additional high leverage

```

observation, based on PROC REG below;
if _n_ = 966 then delete;
run;

```

```

proc reg data = reg9 plots(label)=(CooksD) outest= new_reg_est edf;

```

\*Alternative rich model, based on Stepwise selection, manually

```

removing predictors not significant at alpha = .05;

```

```

model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
GarageCars Fireplaces "Kitche-1bvGr"n

```

```
TotalBsmtSF BsmtFullBath
dKQ_3 dSC_1 dSC_3 dMS_3 dMS_6
BsmtFinSF1
```

```
d2 d5 d10 d12 d14 d19
dE1_2 dE1_3
dF_3
dH_4
dBQ_1
dBE_1
dGQ_1
dC1_1 dC1_7
```

```
/ VIF;
```

```
output out = reg_results2 predicted = reg_predict2;
run; quit;
```

```
proc glmselect data = reg7 plots = (CriterionPanel ASE ASEPlot) seed = 22222;
    *5th-Pass Analysis, Forward Automated Selection and CV, enriching model with
    remaining variables of interest;
    partition fraction(validate=0.3 test=0.2);
```

```
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
    GarageCars Fireplaces "Kitch-1bvGr"n
    TotalBsmtSF BsmtFullBath
    dKQ_3 dSC_1 dSC_3 dMS_2 dMS_3 dMS_6
    dHQ_1 BsmtFinSF1
```

```
d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24
dS_1
dLS_1 dLS_2 dLS_3
/*
dLaC_1 dLaC_2 dLaC_3
```

\*NB: Missing values throw

```
error;
*/
dLaS_1 dLaS_2
dE1_1 dE1_2 dE1_3 dE1_4 dE1_5 dE1_6 dE1_7 dE1_8 dE1_9 dE1_10 dE1_11 dE1_12
dF_1 dF_2 dF_3 dF_4
dH_1 dH_2 dH_3 dH_4 dH_5
dCA_1
dPD_1 dPD_2
dBQ_1 dBQ_2 dBQ_3 dBQ_4 dBQ_5
dBC_1 dBC_2 dBC_3 dBC_4 dBC_5
dBE_1 dBE_2 dBE_3 dBE_4
dBFT_1 dBFT_2 dBFT_3 dBFT_4 dBFT_5 dBFT_6
dBFT2_1 dBFT2_2 dBFT2_3 dBFT2_4 dBFT2_5 dBFT2_6
dGT_1 dGT_2 dGT_3 dGT_4 dGT_5 dGT_6
dGF_1 dGF_2 dGF_3
```

```

dGQ_1 dGQ_2 dGQ_3 dGQ_4 dGQ_5
dGC_1 dGC_2 dGC_3 dGC_4
dE_1 dE_2 dE_3
dHS_1 dHS_2 dHS_3 dHS_4 dHS_5 dHS_6 dHS_7
dFQ_1 dFQ_2
dC1_1 dC1_2 dC1_3 dC1_4 dC1_5 dC1_6 dC1_7 dC1_8
dRS_1 dRS_2 dRS_3 dRS_4 dRS_5
dRM_1 dRM_2 dRM_3 dRM_4 dRM_5 dRM_6
dLC_2 dLC_3 dLC_4 dLC_5
dHQ_2 dHQ_4
dST_2 dST_3 dST_4 dST_5 dST_6 dST_7 dST_8 dST_9 dST_10
dEC_1 dEC_2 dEC_4
dEQ_1 dEQ_2 dEQ_4 dEQ_5
dBT_1 dBT_2 dBT_3 dBT_5
dC2_1 dC2_2 dC2_4 dC2_5 dC2_6

```

```

/ selection = forward(choose = AIC stop = AICC include = 18) details = steps showpvalues;
run;

```

```

data reg10; set reg7;

```

\*Removing unusually high leverage

```

observations, based on PROC REG below;
if _n_ = 410 then delete;
if _n_ = 587 then delete;
if _n_ = 999 then delete;
run;

```

```

proc reg data = reg10 plots(label)=(CooksD) outest= new_reg_est2 edf;
*Alternative rich model, based on Forward selection, manually removing

```

```

predictors not significant at alpha = .05;

```

```

model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
GarageCars Fireplaces "Kitch-1bvGr"n
TotalBsmtSF BsmtFullBath
dKQ_3 dSC_1 dMS_2 dMS_3 dMS_6
dHQ_1 BsmtFinSF1

```

```

d1 d5 d10 d14 d18 d19
dE1_2 dE1_3
dF_3
dBQ_1
dBE_1
dGQ_1
dC1_2 dC1_7
dST_4

```

```

/ VIF;

```

```

output out = reg_results3 predicted = reg_predict3;
run; quit;

```

```

proc glmselect data = reg7 plots = (CriterionPanel ASE ASEPlot) seed = 222222;
    *5th-Pass Analysis, Backward Automated Selection and CV, enriching model
    with remaining variables of interest;
    partition fraction(validate=0.3 test=0.2);

    model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
        GarageCars Fireplaces "Kitche-1bvGr"n
        TotalBsmtSF BsmtFullBath
        dKQ_3 dSC_1 dSC_3 dMS_2 dMS_3 dMS_6
        dHQ_1 BsmtFinSF1

    d1 d2 d3 d4 d5 d6 d7 d8 d9 d10 d11 d12 d13 d14 d15 d16 d17 d18 d19 d20 d21 d22 d23 d24
    dS_1
    dLS_1 dLS_2 dLS_3
    /*
    dLaC_1 dLaC_2 dLaC_3

    error;
    */
    dLaS_1 dLaS_2
    dE1_1 dE1_2 dE1_3 dE1_4 dE1_5 dE1_6 dE1_7 dE1_8 dE1_9 dE1_10 dE1_11 dE1_12
    dF_1 dF_2 dF_3 dF_4
    dH_1 dH_2 dH_3 dH_4 dH_5
    dCA_1
    dPD_1 dPD_2
    dBQ_1 dBQ_2 dBQ_3 dBQ_4 dBQ_5
    dBC_1 dBC_2 dBC_3 dBC_4 dBC_5
    dBE_1 dBE_2 dBE_3 dBE_4
    dBFT_1 dBFT_2 dBFT_3 dBFT_4 dBFT_5 dBFT_6
    dBFT2_1 dBFT2_2 dBFT2_3 dBFT2_4 dBFT2_5 dBFT2_6
    dGT_1 dGT_2 dGT_3 dGT_4 dGT_5 dGT_6
    dGF_1 dGF_2 dGF_3
    dGQ_1 dGQ_2 dGQ_3 dGQ_4 dGQ_5
    dGC_1 dGC_2 dGC_3 dGC_4
    dE_1 dE_2 dE_3
    dHS_1 dHS_2 dHS_3 dHS_4 dHS_5 dHS_6 dHS_7
    dFQ_1 dFQ_2
    dC1_1 dC1_2 dC1_3 dC1_4 dC1_5 dC1_6 dC1_7 dC1_8
    dRS_1 dRS_2 dRS_3 dRS_4 dRS_5
    dRM_1 dRM_2 dRM_3 dRM_4 dRM_5 dRM_6
    dLC_2 dLC_3 dLC_4 dLC_5
    dHQ_2 dHQ_4
    dST_2 dST_3 dST_4 dST_5 dST_6 dST_7 dST_8 dST_9 dST_10
    dEC_1 dEC_2 dEC_4
    dEQ_1 dEQ_2 dEQ_4 dEQ_5
    dBT_1 dBT_2 dBT_3 dBT_5
    dC2_1 dC2_2 dC2_4 dC2_5 dC2_6

    / selection = backward(choose = AIC stop = AICC include = 18) details = steps showpvalues;
run;

```

\*NB: Missing values throw

```
data reg11; set reg7;
```

\*Removing unusually high leverage

```
observation, based on PROC REG below;  
if _n_ = 88 then delete;  
run;
```

```
data reg12; set reg11;
```

\*Removing additional high leverage

```
observation, based on PROC REG below;  
if _n_ = 586 then delete;  
run;
```

```
proc reg data = reg12 plots(label)=(CooksD) outest= new_reg_est3 edf;
```

\*Alternative rich model, based on Backward selection, manually removing

predictors not significant at alpha = .05;

```
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt  
GarageCars Fireplaces  
TotalBsmtSF BsmtFullBath  
dKQ_3 dMS_2 dMS_3  
BsmtFinSF1
```

```
d5 d10 d14 d16 d18 d19
```

```
dE1_5 dE1_8
```

```
dBE_1
```

```
dGC_2
```

```
dFQ_2
```

```
dC1_2
```

```
dBT_3
```

```
/ VIF;
```

```
output out = reg_results4 predicted = reg_predict4;
```

```
run; quit;
```

```
data reg13; set reg7;
```

\*Removing unusually high leverage

```
observations, based on PROC REG below;
```

```
if _n_ = 410 then delete;
```

```
if _n_ = 999 then delete;
```

```
run;
```

```
proc reg data = reg13 plots(label)=(CooksD) outest= new_reg_est4 edf;
```

\*Custom alternative rich model, based on Stepwise Forward and

Backward selection, choosing predictors selected by at least 2 of 3 procedures, manually removing predictors not significant at alpha = .05;

```
model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt  
GarageCars Fireplaces "Kitche-1bvGr"n  
TotalBsmtSF BsmtFullBath
```



```
dKQ_3 dMS_3 dMS_6  
BsmtFinSF1
```

```
d5 d10 d14 d18 d19  
dE1_3  
dF_3  
dBQ_1  
dBE_1  
dGQ_1  
dC1_2  
dC1_7
```

```
      / VIF;  
output out = reg_results5 predicted = reg_predict5;  
run; quit;
```

## SAS Code – KAGGLE SUBMISSION 2

```
FILENAME REFFILE '/home/jrasmusvorrath0/HousingTest.csv';
```

```
PROC IMPORT DATAFILE=REFFILE  
            DBMS=CSV  
            OUT=Ames_newtest;  
            GETNAMES=YES;
```

```
RUN;
```

```
PROC CONTENTS DATA=Ames_newtest; RUN;
```

```
data Ames_newtest2 (drop=old); set Ames_newtest (rename=(LotFrontage=old));  
    LotFrontage = input(old, 8.0);  
run;
```

```
proc contents data = Ames_newtest2; run;
```

```
data Ames_newtest3(drop= Alley PoolQC Fence MiscFeature); set Ames_newtest2;  
run;
```

```
proc contents data = Ames_newtest3; run;
```

```
data Ames_newtest4; set Ames_newtest3(rename=(Functional="Function-1"n  
KitchenAbvGr="Kitch-1bvGr"n));  
run;
```

```
proc contents data = Ames_newtest4; run;
```

```
data Ames_newtest5; set Ames_newtest4;  
if KitchenQual = "Ex" then dKQ_3 = 1; else dKQ_3 = 0;  
if MSZoning = "FV" then dMS_3 = 1; else dMS_3 = 0;  
if MSZoning = "RL" then dMS_6 = 1; else dMS_6 = 0;  
if Neighborhood = "NridgHt" then d5 = 1; else d5 = 0;  
if Neighborhood = "StoneBr" then d10 = 1; else d10 = 0;  
if Neighborhood = "NoRidge" then d14 = 1; else d14 = 0;  
if Neighborhood = "CollgCr" then d18 = 1; else d18 = 0;  
if Neighborhood = "Crawfor" then d19 = 1; else d19 = 0;  
if Exterior1st = "BrkFace" then dE1_3 = 1; else dE1_3 = 0;  
if Foundation = "PConc" then dF_3 = 1; else dF_3 = 0;  
if BsmtQual = "Ex" then dBQ_1 = 1; else dBQ_1 = 0;  
if BsmtExposure = "Gd" then dBE_1 = 1; else dBE_1 = 0;  
if GarageQual = "Ex" then dGQ_1 = 1; else dGQ_1 = 0;  
if Condition1 = "Artery" then dC1_2 = 1; else dC1_2 = 0;  
if Condition1 = "RR Ae" then dC1_7 = 1; else dC1_7 = 0;  
log_SalePrice = log(SalePrice);  
log_GrLivArea = log(GrLivArea);  
log_LotArea = log(LotArea);  
run;
```

```

data Ames_newtrain2; set Ames_train;
if _n_ = 524 then delete;
if _n_ = 1299 then delete;
if KitchenQual = "Ex" then dKQ_3 = 1; else dKQ_3 = 0;
if MSZoning = "FV" then dMS_3 = 1; else dMS_3 = 0;
if MSZoning = "RL" then dMS_6 = 1; else dMS_6 = 0;
if Neighborhood = "NridgHt" then d5 = 1; else d5 = 0;
if Neighborhood = "StoneBr" then d10 = 1; else d10 = 0;
if Neighborhood = "NoRidge" then d14 = 1; else d14 = 0;
if Neighborhood = "CollgCr" then d18 = 1; else d18 = 0;
if Neighborhood = "Crawfor" then d19 = 1; else d19 = 0;
if Exterior1st = "BrkFace" then dE1_3 = 1; else dE1_3 = 0;
if Foundation = "PConc" then dF_3 = 1; else dF_3 = 0;
if BsmtQual = "Ex" then dBQ_1 = 1; else dBQ_1 = 0;
if BsmtExposure = "Gd" then dBE_1 = 1; else dBE_1 = 0;
if GarageQual = "Ex" then dGQ_1 = 1; else dGQ_1 = 0;
if Condition1 = "Artery" then dC1_2 = 1; else dC1_2 = 0;
if Condition1 = "RR Ae" then dC1_7 = 1; else dC1_7 = 0;
log_SalePrice = log(SalePrice);
log_GrLivArea = log(GrLivArea);
log_LotArea = log(LotArea);
run;

```

```

data Ames_newtrain3; set Ames_newtrain2;
if _n_ = 31 then delete;
run;

```

```

data Ames_newtrain4; set Ames_newtrain3;
if _n_ = 410 then delete;
if _n_ = 999 then delete;
run;

```

```

data Ames_newfull; set Ames_newtrain4 Ames_newtest5;
run;

```

```

proc contents data = Ames_newfull; run;

```

```

proc print data = Ames_newfull;
run;

```

```

proc glm data = Ames_newfull plots = all outstat = new_reg_est5;

```

```

model log_SalePrice = log_GrLivArea log_LotArea OverallQual OverallCond YearBuilt
      GarageCars Fireplaces "Kitch1stFlrSF"
      TotalBsmtSF BsmtFullBath
      dKQ_3 dMS_3 dMS_6
      BsmtFinSF1

```

```

d5 d10 d14 d18 d19

```

```

dE1_3
dF_3
dBQ_1
dBE_1
dGQ_1
dC1_2
dC1_7 / cli solution;
output out = result7 p = predict7;
run;

proc print data = result7;
run;

proc contents data = result7; run;

data new_finalp; set result7 (where=(Id > 1460));
if exp(predict7) <= 30000 then Sale_P = 32000;
else Sale_P = exp(predict7);
keep Id Sale_P;
rename Sale_P = SalePrice;
run;

proc print data = new_finalp;
run;

proc export data= new_finalp
  outfile="/home/jrasmusvorrath0/Ames_Kaggle_Submission2.csv"
  dbms=csv
  replace;
run;

```