Automatic Variable Selection and Predictive Accuracy

Saranyan Vasudevan

Regression & Multivariate Analysis (PREDICT-410, Winter 2016)

Introduction

Our Objective

Predicting home prices has never been easy, it is a tradeoff among a set of housing parameters. Ames, Iowa dataset provides an exhaustive source of elements related to house buying/selling. The objective of our data analysis is to build a linear regression model to be able to predict the Sale Price of a housing property. For this, we are using a medium sized dataset comprising of 82 different housing variables. The data set is rich with facts and dimensions - and can be classified as Continuous, Ordinal, Nominal and discrete. The data dictionary is available for each of these variables (link provided in reference section), and it greatly helps with the data profiling.

The objective of this report is to perform variable selection to be included in model, using automated techniques and evaluate them. The evaluation is based on a variety of metrics and error values. For this exercise, we will split the Ames Iowa dataset into 2 – one of training and another for testing. The model selected will be trained on the training set, and tested for prediction accuracy on the testing set.

For this analysis, we will be using 6 different automatic variable selection techniques based on Adjusted R-squared value, Maximized R-squared value, Mallow's Cp value, forward (addition) , backward (elimination) and stepwise selection (addition, elimination and reconciliation) methods.

Defining the Sample Population

Choosing candidate observations

As a part of defining our sample population, excluding observations that are not part of our intended analysis is an important step. Since, our broad objective is model for typical homes, we begin with identifying homes that are not typical, followed by excluding them from our working data set. From our preliminary analysis, we identify 4 types of exclusion criteria,

- Planned Unit Development Homes
- Housing Properties in Non-Housing zones
- Not a single family dwelling based on Building type

- Extremes in various housing parameters (Given we are modeling for single family homes, having extremes might impact our model's prediction ability)

The observations not falling in any of these screening criteria will form part of our working dataset. Table A contains the frequency of occurrence based on the discussed criteria.

Table A

Frequency of occurrence of various Drop Conditions identified

The FREQ Procedure

drop_condition	Frequency	Percent	Cumulative Frequency	Cumulative Percent
01: Planned Unit Development Homes	339	11.57	339	11.57
02: Properties in Non-Housing zones	106	3.62	445	15.19
03: Not a single family dwelling	1	0.03	446	15.22
04: Trimming extremes in TotalSF	6	0.2	452	15.43
05: Trimming extremes in GrLivArea	20	0.68	472	16.11
06: Trimming extremes in GarageArea	234	7.99	706	24.1
07: Trimming extremes in TotalBsmtSF	129	4.4	835	28.5
08: Trimming extremes in FirstFlrSF	31	1.06	866	29.56
09: Trimming extremes in MasVnrArea	17	0.58	883	30.14
10: Trimming extremes in BsmtFinSF1	15	0.51	898	30.65
11: Trimming extremes in FullBath	11	0.38	909	31.02
12: Trimming extremes in YearBuilt	103	3.52	1012	34.54
13: Trimming extremes in OverallQual	3	0.1	1015	34.64
14: Sample Population	1915	65.36	2930	100

The resultant dataset contains 1915 observations, with computed variable for total square footage.

Train / Test data using Cross Validation

Our objective is to select models based on automatic variable selection and determine the prediction accuracy. For this, we will logically partition our sample dataset and train the models on a training set and test them on a testing set. So, the automatic variable selection will happen on the training set (which is approximately 70% of the total sample population), and the prediction accuracy part will happen on the testing set (which is the remaining 30% of sample population). In order to retain the split, we will use SAS procedure that can be referred in the Appendix of this report. Below is the split:

Table B

Distribution of Training and Testing datasets

The FREQ Procedure

train	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	607	31.70	607	31.70
1	1308	68.30	1915	100.00

Train value of 1 indicates ‘training’ set, and 0 indicates ‘testing’ set.

Inputs for Automatic Variable Selection

Although, the model selection is automatic, we still have to supply the procedures with variables we think are reasonable and will benefit the model. This can be done using the PROC CORR procedure on the sample dataset, and will work with the variables that have the high correlation coefficients. Table C contains the output from the procedure.

Table C – PROC CORR Procedure Output

Variable	Pearson Correlation Coefficients	p Value	No. of Observations
OverallQual	0.82	<.0001	1915
TotalSF	0.82	<.0001	1915
GrLivArea	0.76	<.0001	1915
GarageCars	0.68	<.0001	1915
YearBuilt	0.63	<.0001	1915
FullBath	0.61	<.0001	1915
GarageArea	0.61	<.0001	1915
TotRmsAbvGrd	0.61	<.0001	1915
GarageYrBlt	0.57	<.0001	1915
YearRemodel	0.56	<.0001	1915
TotalBsmtSF	0.55	<.0001	1915
FirstFlrSF	0.54	<.0001	1915
MasVnrArea	0.5	<.0001	1915
Fireplaces	0.42	<.0001	1915
BsmtFinSF1	0.29	<.0001	1915

Automated Variable Selection

With the list of Predictor variables identified using Pearson's coefficient in table B, we will use them in the below automated variable selection techniques to identify and analyze models. We will use the following techniques and corresponding model names throughout this report for easier reference.

Table D – AVS Techniques and Corresponding Model names

Criteria / Metric	Model Name
Adjusted R-Squared	Model_AdjR2
Max R (Maximum R2 Improvement)	Model_MaxR
Mallow's Cp	Model_MCp
Forward Variable Selection	Model_F
Backward Variable Selection	Model_B
Stepwise Variable Selection	Model_S

Below will be our input to each of the techniques mentioned above.

Table E – Variable Inputs to AVS Techniques

Response Variable Supplied	train_response
Predictor Variables Supplied	BsmtFinSF1 FullBath GarageArea GrLivArea MasVnrArea OverallQual TotalSF YearBuilt GarageCars TotRmsAbvGrd YearRemodel GarageYrBlt Fireplaces

TotalBsmtSF and FirstFlrSF from Table B, are already taken into account in TotalSF variable, so are ignored from this list.

Model: Model_AdjR2

Variable Selection based on Adjusted R-squared

Based on our training dataset, the below model was selected with 11 of the 13 predictor variables supplied. The ignored predictor variables are TotRmsAbvGrd and GarageYrBlt.

train_response =	*- 1191549*	*+ 23.86380*	*	*BsmtFinSF1*
		- 9028.98009	*	*FullBath*
		+ 19.42821	*	*GarageArea*
		+ 32.16722	*	*GrLivArea*
		+ 16.03908	*	*MasVnrArea*
		+ 17998	*	*OverallQual*
		+ 22.56238	*	*TotalSF*
		+ 181.05927	*	*YearBuilt*
		+ 7449.33746	*	*GarageCars*
		+ 390.43871	*	*YearRemodel*
		+ 5732.18231	*	*Fireplaces*

The model has an adjusted R-squared of 0.8652, which is complemented with the significant F value of 763.55. This indicates much of the variation in the data is explained by the model. However, Standard Error (RMSE) value of 23922 can be used to compare relative accuracy of various models produced.

Table F – Proc Reg Summary

Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	11	4.806502E12	4.369547E11	763.55	<.0001
Error	1296	7.41663E11	572270799		
Corrected Total	1307	5.548165E12			

Root MSE	23922	**R-Square**	0.8663
Dependent Mean	181316	**Adj R-Sq**	0.8652
Coeff Var	13.19364		

Below table contains other metrics that are of our interest to analyze this model further.

Table G – Model_Adjr2 - Metrics Summary

Metric	Value
Mallow's Cp	10.9710
AIC	26387.9253
BIC	26390.1666
MSE	572270799
MAE	17585.45

The above table shows the Mean Absolute Error and Mean Squared Error from our training dataset, these are 2 critical variables for accuracy measurement. The MSE value can indicate the bias and variance of the predictor variable. Mean Absolute Error (Average Magnitude of Error) and Mean Squared Error values in our testing dataset are as follows.

The average magnitude of forecast errors in the testing dataset has increased by approximately 7.9% in the test dataset (as seen in Tables G and H).

Table H – Model_Adjr2 – MAE and MSE – Testing Set

The MEANS Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
ae	607	19097.57	19485.18	0.8941263	214028.25
se	607	743763926	2436179358	0.7994618	45808091564

The Variance Inflation Factor (VIF) for the predictors are under the benchmark of 10, indicating no multicollinearity. We will discuss about this in a separate section of this report.

Note the negative coefficient for FullBath (Full Bathrooms above Grade) variable, which is functionally important for price prediction. We will discuss more about this in the conclusion section.

Model: Model_MaxR

Variable Selection based on Maximized R-squared

The selection criteria in this case is based on the model that has the best Maximized R-squared value. The automatic variable selection procedure returned a model that used all the 13 variables, with the maximum possible R-square value of 0.8664. Below is the model:

*train_response = -1195337 + 23.74840 * BsmtFinSF1*
*- 8857.94980 * FullBath*
*+ 18.35870 * GarageArea*
*+ 34.05654 * GrLivArea*
*+ 16.37906 * MasVnrArea*
*+ 17973 * OverallQual*
*+ 22.65850 * TotalSF*
*+ 169.58143 * YearBuilt*
*+ 7597.97526 * GarageCars*
*- 787.75622 * TotRmsAbvGrd*
*+ 387.16316 * YearRemodel*
*+ 17.79292 * GarageYrBlt*
*+ 5700.02630 * Fireplaces*

Note the negative coefficient for FullBath (Full Bathrooms above Grade) and TotRmsAbvGrd (Total Rooms above Grade) variables, which are functionally important for price prediction. We will discuss more about this in the conclusion section.

Below table indicates the model has a statistically significant F value of 645.64, and the adjusted R squared value (0.8651) very close to the previously seen model (0.8652), although this model uses all the predictor variables.

Table I – Model_MaxR – Proc Reg Summary

Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	13	4.807058E12	3.697737E11	645.64	<.0001
Error	1294	7.411068E11	572725523		
Corrected Total	1307	5.548165E12			

Root MSE	23932	**R-Square**	0.8664
Dependent Mean	181316	**Adj R-Sq**	0.8651
Coeff Var	13.19888		

Below table contains other metrics that are of our interest to analyze this model further.

Table J – Model_MaxR – Metrics Summary

Metric	Value
Mallow's Cp	14
AIC	26390.94
BIC	26393.25
MSE	572725523
MAE	17573.34

The Mallow's Cp value (14) is slightly larger when compared to previous model (10.97), this is because this model has 2 additional predictors compared to the previous one, and the difference is a reflection of the relative lack of fit and bias. The AIC and BIC values appear approximately closer to the previous model.

The below table shows the Mean Absolute Error and Mean Squared Error from our testing dataset, these are 2 critical variables for accuracy measurement. Comparing the MAE values between training and testing datasets, we can say that the predictions in the training dataset is more close to the actuals. The MSE in the testing dataset indicates a potential risk associated with the model.

Table K – Model_MaxR – MAE and MSE –Testing Set

The MEANS Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
ae	607	19078.12	19477.85	51.9239920	214529.14
se	607	742736416	2443975474	2696.10	46022753678

Model: Model_MCp

Variable Selection based on Maximized R-squared

The automatic variable selection is based on the model that has the best Mallow's Cp value. The procedure returned a model using 11 of the 13 variables, with the maximum possible adjusted R-square value of 0.8652 and Mallow's Cp value of 10.97. Below is the model:

Train_response = -1191549
*+ 23.86380 * BsmtFinSF1*
*- 9028.98009 * FullBath*
*+ 19.42821 * GarageArea*
*+ 32.16722 * GrLivArea*
*+ 16.03908 * MasVnrArea*
*+ 17998 * OverallQual*
*+ 22.56238 * TotalSF*
*+ 181.05927 * YearBuilt*
*+ 7449.33746 * GarageCars*
*+ 390.43871 * YearRemodel*
*+ 5732.18231 * Fireplaces*

The variables ignored are TotRmsAbvGrd and GarageYrBlt. These 2 variables were ignored in the variable selection by Adjusted R-squared method as well, the model selected is exactly the same. Below is the analysis of variance table for this model:

Table L – Model_MCp – Proc Reg Summary

Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	12	4.809398E12	4.007832E11	702.54	<.0001
Error	1295	7.387667E11	570476194		
Corrected Total	1307	5.548165E12			

Root MSE	23922	**R-Square**	0.8663
Dependent Mean	181316	**Adj R-Sq**	0.8652
Coeff Var	13.19364		

Below table contains other metrics that are of our interest to analyze this model further.

Table M – Model_MCp – Metrics Summary

Metric	Value
Mallow's Cp	10.9710
AIC	26387.9253
BIC	26390.1666
MSE	572270799
MAE	17585.45

Mean Absolute Error (Average Magnitude of Error) and Mean Squared Error values in our testing dataset are as follows. The model performance as seen from these 2 metrics is approximately the same, when compared to the MaxR model.

Table N – Model_MCp – MAE and MSE –Testing Set

The MEANS Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
ae	607	19097.57	19485.18	0.8941263	214028.25
se	607	743763926	2436179358	0.7994618	45808091564

As a next step, we check out the forward variable selection procedure in the following section.

Model: Model_F

Forward Variable Selection

In this model building process, we add variables one at a time, and the procedure automatically checks the variable for inclusion eligibility. Predictors are added as long as their p value is below 0.1. The procedure returned a model using 11 of the 13 variables, with the maximum possible adjusted R-square value of 0.8652 and Mallow's Cp value of 10.97. The variables ignored are TotRmsAbvGrd and GarageYrBlt – which indicates that they are not significant. The model selected is the same as the one generated in the automatic variable selection using adjusted R-squared and Mallow's Cp. This makes us think if this is the best model possible with the set of 13 predictors that we chose.

Train_response = -1191549
*+ 23.86380 * BsmtFinSF1*
*-9028.98009 * FullBath*
*+ 19.42821 * GarageArea*
*+ 32.16722 * GrLivArea*
*+ 16.03908 * MasVnrArea*
*+ 17998 * OverallQual*
*+ 22.56238 * TotalSF*
*+ 181.05927 * YearBuilt*
*+ 7449.33746 * GarageCars*
*+ 390.43871 * YearRemodel*
*+ 5732.18231 * Fireplaces*

Below table contains other metrics that are of our interest to analyze this model further.

Table O – Model_F – Metrics Summary

Metric	Value
Mallow's Cp	10.9710
AIC	26387.9253
BIC	26390.1666
MSE	572270799
MAE	17585.45

Note that we could change the SLENTRY option in Forward selection to allow variables that are less significant to be added to the model, but the p value of 0.1 is optimal and widely used. As a next step, we check out the backward variable selection procedure in the following section.

Mean Absolute Error (Average Magnitude of Error) and Mean Squared Error values in our testing dataset are as follows. The model performance as seen from these 2 metrics is approximately the same, when compared to the previous model.

Table P – Model_F – MAE and MSE – Testing Set

The MEANS Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
ae	607	19097.57	19485.18	0.8941263	214028.25
se	607	743763926	2436179358	0.7994618	45808091564

Model: Model_B

Backward Variable Selection

This is essentially the opposite of forward selection process, in the sense that we begin with all the variables supplied. Then, each variable which is not statistically significant is discarded from the model and evaluated. This continues until we have all statistically significant variables in our model. The level of statistical significance set is 0.1 in this case, which is set using the SLSTAY option.

The procedure returned a model using 11 of the 13 variables, with the maximum possible adjusted R-square value of 0.8652 and Mallow's Cp value of 10.97. The variables ignored are TotRmsAbvGrd and GarageYrBlt – which indicates that they are not significant. The model selected is the same as the one generated in the automatic variable selection using adjusted R-squared, Mallow's Cp and Forward Selection.

Train_response = -1191549
*+ 23.86380 * BsmtFinSF1*
*- 9028.98009 * FullBath*
*+ 19.42821 * GarageArea*
*+ 32.16722 * GrLivArea*
*+ 16.03908 * MasVnrArea*
*+ 17998 * OverallQual*
*+ 22.56238 * TotalSF*
*+ 181.05927 * YearBuilt*
*+ 7449.33746 * GarageCars*
*+ 390.43871 * YearRemodel*
*+ 5732.18231 * Fireplaces*

Below table contains other metrics that are of our interest to analyze this model further.

Table Q – Model_B – Metrics Summary

Metric	Value
Mallow's Cp	10.9710
AIC	26387.9253
BIC	26390.1666
MSE	572270799
MAE	17585.45

It becomes more assuring that this might be the best possible model, given the predictor variables are in their original state. However, we might need to check for multicollinearity and out of sample accuracy.

Although, it does not seem to apply in this particular case, since we start the model selection will all variables, the risk of predictor variables that co-exist and impact one another is reduced here. As a next step, we check out the stepwise variable selection procedure in the following section.

Mean Absolute Error (Average Magnitude of Error) and Mean Squared Error values in our testing dataset are as follows. The model performance as seen from these 2 metrics is approximately the same, when compared to the previous model.

Table R – Model_F – MAE and MSE – Testing Set

The MEANS Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
ae	607	19097.57	19485.18	0.8941263	214028.25
se	607	743763926	2436179358	0.7994618	45808091564

Model: Model_S

Stepwise Variable Selection

This method combines the forward and backward selection methods, and can add or drop a variable based on its statistical significance value. So, we provide 2 threshold values, one for entry criteria and another for retention. The method drops statistically insignificant variables in steps, and then reconsiders them for being added to the model in a stepwise fashion based on their p value. In our case, we provide the entry and retention criteria to be 0.1.

The stepwise procedure returned a model using 11 of the 13 variables, with the maximum possible adjusted R-square value of 0.8652 and Mallow's Cp value of 10.97. The variables ignored are TotRmsAbvGrd and GarageYrBlt – which indicates that they are not significant. The model selected is the same as the one generated in the automatic variable selection using adjusted R-squared, Mallow's Cp, Forward and backward Selection methods. Below is the model:

*Train_response = -1191549 + 23.86380 * BsmtFinSF1*
*- 9028.98009 * FullBath*
*+ 19.42821 * GarageArea*
*+ 32.16722 * GrLivArea*
*+ 16.03908 * MasVnrArea*
*+ 17998 * OverallQual*
*+ 22.56238 * TotalSF*

*+ 181.05927 * YearBuilt*
*+ 7449.33746 * GarageCars*
*+ 390.43871 * YearRemodel*
*+ 5732.18231 * Fireplaces*

Below table contains other metrics that are of our interest to analyze this model further.

Table S – Model_S – Metrics Summary

Metric	Value
Mallow's Cp	10.9710
AIC	26387.9253
BIC	26390.1666
MSE	572270799
MAE	17585.45

Mean Absolute Error (Average Magnitude of Error) and Mean Squared Error values in our testing dataset are as follows. The model performance as seen from these 2 metrics is approximately the same, when compared to the previous model.

Table T – Model_S – MAE and MSE – Testing Set

The MEANS Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
ae	607	19097.57	19485.18	0.8941263	214028.25
se	607	743763926	2436179358	0.7994618	45808091564

This being our last model selection technique in this analysis, it becomes imperative to analyze this model more and to why 5 of the 6 techniques chose this model.

Summary of Models from Automatic Variable Selection

The criteria based techniques such as adjusted R-squared, Mallow's Cp, Forward, Backward and Stepwise all pointed to the same model with 11 predictors and excluded 2 variables from the initial set, as they were statistically insignificant to make it to the model. The maximized R2 model shows the best model for each

possible size, however, thing to note here is that the plastic steps in MaxR model did not pick the other model to be the best 11 variable model.

Table U – Model Comparison

Technique	Model Name	Model	Excluded Variables
Adjusted R-Squared / Mallow's Cp / Forward Selection / Backward Selection / Stepwise Selection	Model_AdjR2 / Model_MCp / Model_F / Model_B / Model_S	train_response = -1191549 + 23.8638 * BsmtFinSF1 - 9028.98009 * FullBath + 19.42821 * GarageArea + 32.16722 * GrLivArea + 16.03908 * MasVnrArea + 17998 * OverallQual + 22.56238 * TotalSF + 181.05927 * YearBuilt + 7449.33746 * GarageCars + 390.43871 * YearRemodel + 5732.18231 * Fireplaces	TotRmsAbvGrd,GarageYrBlt
Maximized R^2	Model_MaxR	train_response = -1195337 + 23.74840 * BsmtFinSF1 - 8857.94980 * FullBath + 18.35870 * GarageArea + 34.05654 * GrLivArea + 16.37906 * MasVnrArea + 17973 * OverallQual + 22.65850 * TotalSF + 169.58143 * YearBuilt + 7597.97526 * GarageCars - 787.75622 * TotRmsAbvGrd + 387.16316 * YearRemodel + 17.79292 * GarageYrBlt + 5700.02630 * Fireplaces	None

Metrics Summary

The below table shows the comparison of metrics between the 2 models selected, this includes Adjusted R-squared, AIC, BIC, Cp, MSE and MAE across train and test methods.

Table V – Comparison of Metrics between the two models

Model Name	No. of Predictors	Step	Adj R^2	AIC	BIC	Cp	MAE	MSE
Model_AdjR2, Model_MCp, Model_F, Model_B, Model_S	11	Train	0.8652	26387.93	26390.17	10.97	17585.45	572270799
		Test					19097.57	743763926
Model_MaxR	13	Train	0.8651	26390.94	26393.25	14	17573.34	572725523
		Test					19078.12	742736416

The adjusted R2 values are tight in both the models, with a marginal improvement seen in the model chosen by most of the techniques. A similar marginal improvement is observed in AIC and BIC metrics as well, which is lower the better. This could be because of less number of predictors used in the model, thereby reducing model complexity and relieving penalty to some extent.

Mallow's Cp value for the model with 11 predictors is 10.97 (i.e. close to 11, the number of predictors in the models). However, the Model_MaxR has a relatively higher value for Mallow's Cp indicating lack of fit and bias.

The MAE for the training datasets look better than the values for testing datasets in both the models. We will discuss more about this in the Operational Validation section of this report.

Given these points, we can conclude that in this case, the Model with 11 predictors is better than the model with 13 predictors.

Multicollinearity

A model is likely to exhibit multicollinearity, if more than 1 predictor variables tend to do the same job, i.e. they have relationships, and impact the overall model. For the purposes of this analysis, we can identify this using Variance Inflation Factor (VIF), which is listed below for both the models in discussion. A VIF value of >10 raises some serious concerns of multicollinearity.

Table W – Multicollinearity Analysis

Predictor Variable	VIF (Model_MaxR)	VIF (Model_AdjR2 / Model_MCp / Model_F / Model_B / Model_S)
BsmtFinSF1	1.21575	1.20179
FullBath	2.56628	2.53892
GarageArea	3.90504	3.71042
GrLivArea	7.56976	5.6221
MasVnrArea	1.43706	1.42875
OverallQual	2.71148	2.70147
TotalSF	6.115	6.02527
YearBuilt	5.26986	2.84162
GarageCars	4.49476	4.45007
TotRmsAbvGrd	3.27618	NA
YearRemodel	2.0068	1.95282
GarageYrBlt	4.94174	NA
Fireplaces	1.37582	1.36494

Overall, both the models do not show a sign of multicollinearity, but a closer look reveals that the model with 11 predictors has slightly lower values for VIF. This supplements our conclusion in previous section that the model chosen by most of the techniques is better than Model_MaxR.

Operational Validation

As the next step we look for some prediction accuracy of the models created. For the purposes of this report, we define the prediction accuracy to be the proximity

of the predicted value to the actual value. If the predicted value is within 10% of actual value, then we rank those as Grade 1 Predictions. Grade 2 will contain 10 to 15% of variance with respect to actual values. Anything above 15% will be considered Grade 3.

We create SAS format to tag observations based on the rubric mentioned above. We will use these formats on the 2 models that we have identified so far.

Model_AdjR2, Model_MCp, Model_F, Model_B and Model_S

Approximately 70% of the dataset used for analysis was used to train the model, and the remaining 30% to test the model for predictive accuracy. Below is the frequency table of Pred_Grade by Training vs. testing observations. Train value of 1 indicates Training metrics and 0 indicates Testing metrics. The metrics listed below are the counts and column percentages for better comparison. From the table, we see that 59.56% of observations in training dataset reported a Grade 1 prediction, which is comparable to 57.50% of the total testing dataset. Likewise, the Grade 2 figures are closely located i.e. 18.58% in training and 17.96% in testing. Grade 3 figures too don't vary much between the training and testing datasets.

Table X – Prediction Grade Split-up between training and testing sets

Table of pred_grade by train

pred_grade	train		
	0	1	Total
Grade 1	349 57.50	779 59.56	1128
Grade 2	109 17.96	243 18.58	352
Grade 3	149 24.55	286 21.87	435
Total	607	1308	1915

These results can be compared to the MAE values of the training and testing datasets, where the Mean Absolute error rose by 8.5% in the test dataset

Table Y – MAE and MSE between training and testing sets

The MEANS Procedure

train	N Obs	Variable	N	Mean	Std Dev	Minimum	Maximum
0	607	ae	607	19097.57	19485.18	0.8941263	214028.25
		se	607	743763926	2436179358	0.7994618	45808091564
1	1308	ae	1308	17585.45	16061.43	9.0502846	145869.58
		se	1308	567020608	1320185462	81.9076504	21277934508

From this, we can say that the model displayed a marginally better performance accuracy in the training set, when compared to the testing set.

Model_MaxR

Approximately 70% of the dataset used for analysis was used to train the model, and the remaining 30% to test the model for predictive accuracy. Below is the frequency table of Pred_Grade by Training vs. testing observations. Train value of 1 indicates Training metrics and 0 indicates Testing metrics. The metrics listed below are the counts and column percentages for better comparison. From the table, we see that 59.79% of observations in training dataset reported a Grade 1 prediction, which is comparable to 57.50% of the total testing dataset. Likewise, the Grade 2 figures are closely located i.e. 18.73% in training and 18.29% in testing. Grade 3 figures too don't vary much between the training and testing datasets.

Table Z – Prediction Grade Split-up between training and testing sets

Table of pred_grade by train

pred_grade	train		
	0	1	Total
Grade 1	349	782	1131
	57.50	59.79	

Table of pred_grade by train			
pred_grade	train		
	0	1	Total
Grade 2	111 18.29	245 18.73	356
Grade 3	147 24.22	281 21.48	428
Total	607	1308	1915

These results can be compared to the MAE values of the training and testing datasets, where the Mean Absolute error rose by 8.5% in the test dataset.

Table AA – MAE and MSE between training and testing sets

The MEANS Procedure

train	N Obs	Variable	N	Mean	Std Dev	Minimum	Maximum
0	607	ae	607	19078.12	19477.85	51.9239920	214529.14
		se	607	742736416	2443975474	2696.10	46022753678
1	1308	ae	1308	17573.34	16061.46	6.6388778	146865.71
		se	1308	566595434	1326428254	44.0746980	21569536646

From this, we can say that the model demonstrated a marginally better performance accuracy in the training set, when compared to the testing set.

Conclusion

We started our analysis with 13 predictors, and most of the models ended up using 11 and one of them using all 13. The results of the Automatic variable selection were quite interesting, with 5 of the 6 techniques pointing to the same model (below). Also, this raises a question, what would have been the case had we analyzed with more variables or included some computed categorical variables or some transformations. Such a test, would have revealed the possibility of overfitting with all the predictors, which did not happen in our analysis.

train_response = -1191549
*+ 23.8638 * BsmtFinSF1* ***- 9028.98009 * FullBath*** *+ 19.42821 * GarageArea*
*+ 32.16722 * GrLivArea + 16.03908 * MasVnrArea + 17998 * OverallQual*
*+ 22.56238 * TotalSF +* ***181.05927 * YearBuilt*** *+ 7449.33746 * GarageCars*
*+ 390.43871 * YearRemodel + 5732.18231 * Fireplaces*

Clearly, the better of the two, is the model with 11 predictors, and was selected by 5 of the 6 methods we ran. Collectively, this model displayed better adjusted R-squared, mallow's Cp (with smaller variance), AIC and BIC values. The metrics that SAS computed such as AIC, BIC, Mallow's Cp, MSE helped choosing this model over the other. Multicollinearity was not observed with the variables in the models we worked with. Without these metrics and the results from operational validation – it would have been a tough decision to make, given the marginality. Also, the performance in training set was slightly better compared to the testing dataset.

It is imperative to note the huge negative coefficient on FullBath variable. Which means that while holding rest of the variables constant, the Sale Price would go down with a unit increase in Full Bathrooms, which does not appear to be correct. This does not indicate that the variable FullBath is not applicable for SalePrice prediction, but could be because of the presence of other irrelevant variables supplied to the variable selection methods. This emphasizes the importance of choosing predictor variables based on their relevance, and the downside of choosing variables based on 'p' values. Likewise, to make more sense, the YearBuilt variable should have been converted to age at the time of sale and ranked before applying to the model.

Another takeaway is that, the Sale Price prediction will be based on a bunch of predictors and not just one or two. One thing to remember in our analysis, is that we chose the variables that reported a top correlation figure. This does not assure that one of our models will be the best candidate to predict housing prices in Ames, Iowa.

Appendix:

```
title 'Predict 410 Winter 2016 Sec 55 Assignment 5';
libname mydata '/scs/crb519/PREDICT_410/SAS_Data/' access=readonly;

data ames_stg0;
set mydata.ames_housing_data;
run;

* Create additional variables for computation and to qualify sample population;
data ames_stg1;
set ames_stg0;
format drop_condition $50.;
TotalSF = TotalBsmtSF + FirstFlrSF + SecondFlrSF;
if subclass >= 120 and subclass <= 180 then drop_condition='01: Planned Unit Development Homes';
else if Zoning ='A' or Zoning ='C' or Zoning ='FV' or Zoning ='I' then drop_condition='02: Properties in Non-
Housing zones';
else if BldgType = '2FmCon' or BldgType = 'Duplx' or BldgType = 'TwnhsE' or BldgType = 'TwnhsI' then
drop_condition='03: Not a single family dwelling';
else if TotalSF > 6000 then drop_condition='04: Trimming extremes in TotalSF';
else if GrLivArea < 400 OR GrLivArea > 3000 then drop_condition='05: Trimming extremes in GrLivArea';
else if GarageArea < 220 OR GarageArea > 1000 then drop_condition='06: Trimming extremes in
GarageArea';
else if TotalBsmtSF < 400 OR TotalBsmtSF > 2100 then drop_condition='07: Trimming extremes in
TotalBsmtSF';
else if FirstFlrSF < 400 OR FirstFlrSF > 2000 then drop_condition='08: Trimming extremes in FirstFlrSF';
else if MasVnrArea > 1000 or MasVnrArea=. then drop_condition='09: Trimming extremes in
MasVnrArea';
else if BsmtFinSF1 > 1500 then drop_condition='10: Trimming extremes in BsmtFinSF1';
else if FullBath < 1 OR FullBath > 3 then drop_condition='11: Trimming extremes in FullBath';
else if YearBuilt < 1920 then drop_condition='12: Trimming extremes in YearBuilt';
else if OverallQual < 3 then drop_condition='13: Trimming extremes in OverallQual';
else drop_condition='14: Sample Population';
run;

title 'Frequency of occurrence of various Drop Conditions identified';
proc freq data=ames_stg1;
tables drop_condition;
run; quit;

* Keep only qualifed sample population in our work dataset;
data ames_stg2;
set ames_stg1;
where drop_condition='14: Sample Population';
run;
```

```
data ames_training;
set ames_stg2;
* generate a uniform(0,1) random variable with seed set to 123;
u = uniform(123);
if (u < 0.70) then train = 1;
else train = 0;
if (train=1) then train_response=SalePrice;
else train_response=.;
run;

title 'Distribution of Training and Testing datasets';
proc freq data=ames_training;
table train;
run;quit;


proc corr data=ames_training nosimple rank;
var BsmtFinSF1 FirstFlrSF FullBath GarageArea GrLivArea MasVnrArea OverallQual TotalBsmtSF TotalSF
YearBuilt GarageCars TotRmsAbvGrd YearRemodel GarageYrBlt Fireplaces;
with train_response;

/* Model ADJRSQ */
proc reg data=ames_training outest=ADJRSQ_out1;
model train_response=BsmtFinSF1 FullBath GarageArea GrLivArea MasVnrArea OverallQual TotalSF
YearBuilt GarageCars TotRmsAbvGrd YearRemodel GarageYrBlt Fireplaces
/ selection=ADJRSQ AIC BIC mse rmse CP VIF best=10;
output out=ADJRSQ_out2
    PREDICTED=PREDICT_SalePricehat
    RESIDUAL=RESID_SalePriceresid;
run;


proc print data=ADJRSQ_out2 (obs=10);
run;quit;


data err_adjrsq_train;
set ADJRSQ_out2;
ae=abs(PREDICT_SalePricehat-train_response);
se=abs(PREDICT_SalePricehat-train_response) ** 2;
run;
quit;


proc means data=err_adjrsq_train;
where train=1;
var ae se;
```

```
run;


data err_adjrsq_test;
set ADJRSQ_out2;
*where train=0;
ae=abs(PREDICT_SalePricehat-SalePrice);
se=abs(PREDICT_SalePricehat-SalePrice) ** 2;
run;
quit;


proc means data=err_adjrsq_test;
where train=0;
var ae se;
run;


proc format;
value perf_sfmt
        0.15 - high = 'Grade 3'
        0.10 -< 0.15 = 'Grade 2'
        0 -< 0.10 = 'Grade 1'
        ;
run;

data ADJRSQ_op_valdtn_train;
set ADJRSQ_out2;
where train=1;
pred_grade_val = abs(abs(PREDICT_SalePricehat-train_response)/train_response);
pred_grade= put(pred_grade_val ,perf_sfmt.);
run; quit;

proc freq data=ADJRSQ_op_valdtn_train;
tables pred_grade;
run;


data ADJRSQ_op_valdtn_test;
set ADJRSQ_out2;
where train=0;
pred_grade_val = abs(abs(PREDICT_SalePricehat-SalePrice)/SalePrice);
pred_grade= put(pred_grade_val ,perf_sfmt.);
run; quit;

proc freq data=ADJRSQ_op_valdtn_test;
tables pred_grade;
```

```
run;


data ADJRSQ_op_valdtn_traintest;
set ADJRSQ_out2;
if train=1 then
do;
pred_grade_val = abs(abs(PREDICT_SalePricehat-train_response)/train_response);
pred_grade= put(pred_grade_val ,perf_sfmt.);
end;
else
do;
pred_grade_val = abs(abs(PREDICT_SalePricehat-SalePrice)/SalePrice);
pred_grade= put(pred_grade_val ,perf_sfmt.);
end;
run; quit;

proc freq data=ADJRSQ_op_valdtn_traintest;
tables pred_grade * train / norow nopercent;
run;

/* Model MAXR*/
proc reg data=ames_training outest=MAXR_out1;
model train_response=BsmtFinSF1 FullBath GarageArea GrLivArea MasVnrArea OverallQual TotalSF
YearBuilt GarageCars TotRmsAbvGrd YearRemodel GarageYrBlt Fireplaces
/ selection=MAXR adjrsq aic bic mse rmse cp vif;
output out=MAXR_out2
    PREDICTED=PREDICT_SalePricehat
    RESIDUAL=RESID_SalePriceresid;
run;


proc print data=MAXR_out1;
run;quit;


data err_maxr_train;
set MAXR_out2;
ae=abs(PREDICT_SalePricehat-train_response);
se=abs(PREDICT_SalePricehat-train_response) ** 2;
run;
quit;


proc means data=err_maxr_train;
where train=1;
var ae se;
```

```
run;


data err_maxr_test;
set MAXR_out2;
*where train=0;
ae=abs(PREDICT_SalePricehat-SalePrice);
se=abs(PREDICT_SalePricehat-SalePrice) ** 2;
run;
quit;


proc means data=err_maxr_test;
where train=0;
var ae se;
run;


data MaxR_op_valdtn_train;
set MAXR_out2;
where train=1;
pred_grade_val = abs(abs(PREDICT_SalePricehat-train_response)/train_response);
pred_grade= put(pred_grade_val ,perf_sfmt.);
run; quit;

proc freq data=MaxR_op_valdtn_train;
tables pred_grade;
run;


data MaxR_op_valdtn_test;
set MAXR_out2;
where train=0;
pred_grade_val = abs(abs(PREDICT_SalePricehat-SalePrice)/SalePrice);
pred_grade= put(pred_grade_val ,perf_sfmt.);
run; quit;

proc freq data=MaxR_op_valdtn_test;
tables pred_grade;
run;


data MaxR_op_valdtn_traintest;
set MaxR_out2;
if train=1 then
do;
pred_grade_val = abs(abs(PREDICT_SalePricehat-train_response)/train_response);
```

```
pred_grade= put(pred_grade_val ,perf_sfmt.);
end;
else
do;
pred_grade_val = abs(abs(PREDICT_SalePricehat-SalePrice)/SalePrice);
pred_grade= put(pred_grade_val ,perf_sfmt.);
end;
run; quit;

proc freq data=MaxR_op_valdtn_traintest;
tables pred_grade * train / norow nopercent;
run;

/* Model Mallow's CP*/
proc reg data=ames_training outest=Cp_out1;
model train_response=BsmtFinSF1 FullBath GarageArea GrLivArea MasVnrArea OverallQual TotalSF
YearBuilt GarageCars TotRmsAbvGrd YearRemodel GarageYrBlt Fireplaces
/ selection=cp adjrsq aic bic mse rmse vif best=10;
output out=Cp_out2
    PREDICTED=PREDICT_SalePricehat
    RESIDUAL=RESID_SalePriceresid;
run;


proc print data=Cp_out2(obs=10);
run;quit;


data err_Cp_train;
set Cp_out2;
ae=abs(PREDICT_SalePricehat-train_response);
se=abs(PREDICT_SalePricehat-train_response) ** 2;
run;
quit;


proc means data=err_Cp_train;
where train=1;
var ae se;
run;


data Cp_maxr_test;
set Cp_out2;
*where train=0;
ae=abs(PREDICT_SalePricehat-SalePrice);
se=abs(PREDICT_SalePricehat-SalePrice) ** 2;
```

```
run;
quit;


proc means data=Cp_maxr_test;
where train=0;
var ae se;
run;


data Cp_op_valdtn_train;
set Cp_out2;
where train=1;
pred_grade_val = abs(abs(PREDICT_SalePricehat-train_response)/train_response);
pred_grade= put(pred_grade_val ,perf_sfmt.);
run; quit;

proc freq data=Cp_op_valdtn_train;
tables pred_grade;
run;


data Cp_op_valdtn_test;
set Cp_out2;
where train=0;
pred_grade_val = abs(abs(PREDICT_SalePricehat-SalePrice)/SalePrice);
pred_grade= put(pred_grade_val ,perf_sfmt.);
run; quit;

proc freq data=Cp_op_valdtn_test;
tables pred_grade;
run;

/* Model Forward Selection*/
proc reg data=ames_training outest=forward_out1;
model train_response=BsmtFinSF1 FullBath GarageArea GrLivArea MasVnrArea OverallQual TotalSF
YearBuilt GarageCars TotRmsAbvGrd YearRemodel GarageYrBlt Fireplaces
/ selection=forward slentry=0.1 adjrsq aic bic mse rmse cp vif;
output out=forward_out2
    PREDICTED=PREDICT_SalePricehat
    RESIDUAL=RESID_SalePriceresid;
run;


proc print data=forward_out2(obs=10);
run;quit;
```

```
data err_forward_train;
set forward_out2;
ae=abs(PREDICT_SalePricehat-train_response);
se=abs(PREDICT_SalePricehat-train_response) ** 2;
run;
quit;


proc means data=err_forward_train;
where train=1;
var ae se;
run;


data err_forward_test;
set forward_out2;
*where train=0;
ae=abs(PREDICT_SalePricehat-SalePrice);
se=abs(PREDICT_SalePricehat-SalePrice) ** 2;
run;
quit;


proc means data=err_forward_test;
where train=0;
var ae se;
run;


data Forward_op_valdtn_train;
set forward_out2;
where train=1;
pred_grade_val = abs(abs(PREDICT_SalePricehat-train_response)/train_response);
pred_grade= put(pred_grade_val ,perf_sfmt.);
run; quit;

proc freq data=Forward_op_valdtn_train;
tables pred_grade;
run;


data Forward_op_valdtn_test;
set forward_out2;
where train=0;
pred_grade_val = abs(abs(PREDICT_SalePricehat-SalePrice)/SalePrice);
pred_grade= put(pred_grade_val ,perf_sfmt.);
```

```
run; quit;

proc freq data=Forward_op_valdtn_test;
tables pred_grade;
run;


/* Model backward Selection*/
proc reg data=ames_training outest=backward_out1;
model train_response=BsmtFinSF1 FullBath GarageArea GrLivArea MasVnrArea OverallQual TotalSF
YearBuilt GarageCars TotRmsAbvGrd YearRemodel GarageYrBlt Fireplaces
/ selection=backward slstay=0.1 adjrsq aic bic mse rmse cp vif;
output out=backward_out2
    PREDICTED=PREDICT_SalePricehat
    RESIDUAL=RESID_SalePriceresid;
run;


proc print data=backward_out2(obs=10);
run;quit;


data err_backward_train;
set backward_out2;
ae=abs(PREDICT_SalePricehat-train_response);
se=abs(PREDICT_SalePricehat-train_response) ** 2;
run;
quit;


proc means data=err_backward_train;
where train=1;
var ae se;
run;


data err_backward_test;
set backward_out2;
*where train=0;
ae=abs(PREDICT_SalePricehat-SalePrice);
se=abs(PREDICT_SalePricehat-SalePrice) ** 2;
run;
quit;


proc means data=err_backward_test;
where train=0;
```

```
var ae se;
run;


data backward_op_valdtn_train;
set backward_out2;
where train=1;
pred_grade_val = abs(abs(PREDICT_SalePricehat-train_response)/train_response);
pred_grade= put(pred_grade_val ,perf_sfmt.);
run; quit;

proc freq data=backward_op_valdtn_train;
tables pred_grade;
run;


data backward_op_valdtn_test;
set backward_out2;
where train=0;
pred_grade_val = abs(abs(PREDICT_SalePricehat-SalePrice)/SalePrice);
pred_grade= put(pred_grade_val ,perf_sfmt.);
run; quit;

proc freq data=backward_op_valdtn_test;
tables pred_grade;
run;

/* Model stepwise Selection*/
proc reg data=ames_training outest=stepwise_out1;
model train_response=BsmtFinSF1 FullBath GarageArea GrLivArea MasVnrArea OverallQual TotalSF
YearBuilt GarageCars TotRmsAbvGrd YearRemodel GarageYrBlt Fireplaces
/ selection=stepwise slentry=0.1 slstay=0.1 adjrsq aic bic mse rmse cp vif;
output out=stepwise_out2
    PREDICTED=PREDICT_SalePricehat
    RESIDUAL=RESID_SalePriceresid;
run;


proc print data=stepwise_out2(obs=10);
run;quit;


data err_stepwise_train;
set stepwise_out2;
ae=abs(PREDICT_SalePricehat-train_response);
se=abs(PREDICT_SalePricehat-train_response) ** 2;
run;
```

```
quit;


proc means data=err_stepwise_train;
where train=1;
var ae se;
run;


data err_stepwise_test;
set stepwise_out2;
*where train=0;
ae=abs(PREDICT_SalePricehat-SalePrice);
se=abs(PREDICT_SalePricehat-SalePrice) ** 2;
run;
quit;


proc means data=err_stepwise_test;
where train=0;
var ae se;
run;


data stepwise_op_valdtn_train;
set stepwise_out2;
where train=1;
pred_grade_val = abs(abs(PREDICT_SalePricehat-train_response)/train_response);
pred_grade= put(pred_grade_val ,perf_sfmt.);
run; quit;

proc freq data=stepwise_op_valdtn_train;
tables pred_grade;
run;


data backward_op_valdtn_test;
set backward_out2;
where train=0;
pred_grade_val = abs(abs(PREDICT_SalePricehat-SalePrice)/SalePrice);
pred_grade= put(pred_grade_val ,perf_sfmt.);
run; quit;

proc freq data=stepwise_op_valdtn_test;
tables pred_grade;
run;
```

References:

Montgomery, Douglas C., Elizabeth A. Peck, and G. Geoffrey. Vining. Introduction to Linear Regression Analysis. Hoboken, NJ: Wiley, Fifth Edition.

http://www.amstat.org/publications/jse/v19n3/decock/DataDocumentation.txt