Assignment #5

Andrea Bruckner

Introduction

In this report we are using the Ames Housing data set to create a predictive modeling framework, identify a "best" model using various automated variable selection techniques, examine our models for the presence of multicollinearity, assess the predictive accuracy of our models by cross-validating the training set statistics with the testing set statistics, and look at how the statistical validation of a model does or does not translate to the business application of the model. We chose 15 predictor variables to explore with six automated variable selection techniques, and we ended up seeing the same "best" 12-variable model selected by each technique. Although there are no obvious signs of strong multicollinearity in the model, we identify several predictor variables that warrant closer examination, especially since a couple of them show an illogically inverse relationship with the response variable. Comparing the MAE and MSE statistics between the training set and testing set indicated that our "best" model performs better in-sample that out-of-sample. This report explores the many challenges of building an accurate predictive model, and it suggests that that is always room for improvement.

Data

The Ames Housing data set is from the Ames Assessor's Office. It contains 2930 observations with 82 variables concerning residential properties sold between 2006 and 2010 in Ames, Iowa. The variables consist of 23 nominal, 23 ordinal, 14 discrete, and 20 continuous variables, as well as 2 additional observation variables[1]. A broad overview of the data in Assignment #1 revealed that we have enough data to predict home prices. The data set contains a variety of categorical and numeric variables from which we can draw conclusions, and the data set contains very few missing values. In the next section we explain how we created a sample population from this data set in order to build predictive models for home sale prices.

Sample Population

Because not all of the properties in the Ames Housing data set are representative of a typical home, we created a sample population to focus on just the relevant data for our model. For instance, we did not want to include data about apartment sales or commercial properties in our sample. **Table 1** below illustrates the criteria that caused observations to drop from our sample population. The seventh drop condition, 07: Pricing Error, does not appear in the table because all of the SalePrice data met the condition that the house sale price must be greater than $0. We decided to include this drop condition to eliminate any obvious errors. As shown in the table below, the sample population totals 1831 observations and represents 62.5 percent of the original data set.

[1] Source: http://www.amstat.org/publications/jse/v19n3/decock/DataDocumentation.txt

Drop Condition	Frequency	Percent	Cumulative Frequency	Cumulative Percent
01: Not a Single Family Home	505	17.24	505	17.24
02: Non-Residential Zone or RV Park	103	3.52	608	20.75
03: Not Normal Sale Condition	379	12.94	987	33.69
04: Atypically Large House	1	0.03	988	33.72
05: No Basement	44	1.50	1032	35.22
06: No Garage	66	2.25	1098	37.47
08: Not All Public Utilities Available	1	0.03	1099	37.51
09: Sample Population	1831	62.49	2930	100.00

Table 1: Drop Conditions for the Sample Population

We decided on these drop conditions based on what a typical homebuyer would likely consider when purchasing a home. Most homebuyers are seeking single-family, non-mobile homes; are purchasing the home under normal conditions (i.e., not buying the home from a family member or through a short sale); are buying a normal sized home (i.e., not a mansion); and are looking for homes that include a basement, garage, and all public utilities.

Partitioning the Sample Population for Cross-Validation

In order to assess the predictive accuracy of the models we identified using automated variable selection, we split the sample population into a training set that contained approximately 70 percent of the sample population, and a testing set that contained the rest of the sample population. **Table 2** shows the number of observations in each set.

train	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	579	31.62	579	31.62
1	1252	68.38	1831	100.00

Table 2: Observation Counts for Test/Train Data Set Partition

Once we trained the models, we tested the predictive accuracy of the models on the remaining 30 percent of the sample population. In the discussion that follows, we will review the models identified from each selection technique and assess their quantitative measures of goodness-of-fit, multicollinearity, and operational validation.

Model Identification by Automated Variable Selection

Prior to performing any automated variable selection techniques, we selected which predictor variables to model against the response variable, train_response. The Ames housing data set contains many categorical variables, which we decided to leave out of the model rather than convert to numeric variables. Instead we focused on just the continuous and discrete variables available in the data set. The 15 predictor variables we chose to run through the various automated variable selection techniques were: GrLivArea, TotalBsmtSF, LotFrontage, LotArea, BsmtFinSF1, BsmtFinSF2, FirstFlrSF, GarageArea, OpenPorchSF, OverallQual, OverallCond, YearBuilt, BedroomAbvGr, TotRmsAbvGrd, and TotalBath_Calc. The last variable in that list was calculated from the total number of full and half bathrooms throughout the house, including

the basement. We chose these predictor variables because we determined that they applied to most of the typical houses in Ames, Iowa, unlike some other numeric variables in the data set, such as the number of fireplaces or pool size

In order to assess the predictive potential and effect these predictors had on the response variable, train_response, we used several techniques to automatically select the variables for our models: Adjusted R-Squared Selection, Maximum R-Squared Improvement Selection, Mallows' Cp Selection, Forward Selection, Backward Selection, and Stepwise Selection. We named the models Model_AdjR2, Model_MaxR, Model_MCp, Model_F, Model_B, and Model_S to represent each of the automated variable selection techniques. We discovered that all of the automated variable selection techniques resulted in the same "best" 12-variable model[2].

Per **Table 3**, the fitted equation for all the models mentioned above (Model_AdjR2, Model_MaxR, Model_MCp, Model_F, Model_B, and Model_S) is **train_response = -906548 + 65.49*GrLivArea + 34.73*TotalBsmtSF + 102.97*LotFrontage + 2.06*LotArea + 21.02*BsmtFinSF1 – 11.52*FirstFlrSF + 36.50*GarageArea + 16178*OverallQual + 7761.25*OverallCond + 397.88*YearBuilt – 9269.92*BedroomAbvGr + 1823.33*TotRmsAbvGrd.**

Parameter Estimates						
Variable	**DF**	**Parameter Estimate**	**Standard Error**	**t Value**	**Pr > |t|**	**Variance Inflation**
Intercept	1	-906548	67106	-13.51	<.0001	0
GrLivArea	1	65.49045	3.26646	20.05	<.0001	4.84545
TotalBsmtSF	1	34.72988	4.06903	8.54	<.0001	4.35435
LotFrontage	1	102.96549	46.67642	2.21	0.0276	1.38700
LotArea	1	2.06187	0.20048	10.28	<.0001	1.35114
BsmtFinSF1	1	21.02109	2.10742	9.97	<.0001	1.47357
FirstFlrSF	1	-11.51894	4.31078	-2.67	0.0077	4.45458
GarageArea	1	36.50327	5.21911	6.99	<.0001	1.82734
OverallQual	1	16178	912.82115	17.72	<.0001	2.59058
OverallCond	1	7761.24864	763.30773	10.17	<.0001	1.36072
YearBuilt	1	397.87825	34.36038	11.58	<.0001	2.05451
BedroomAbvGr	1	-9269.91594	1438.21338	-6.45	<.0001	1.97256
TotRmsAbvGrd	1	1823.32997	1010.75147	1.80	0.0715	3.81318

Table 3: Parameter Estimates for Each Model Fitted from the Training Set

The fitted equation, in general, makes sense. Since train_response represents the sale price for the training set, it logically increases as the living space, quality, condition, etc. increase. However, as the fitted equation shows, both FirstFlrSF and BedroomAbvGr cause the response variable to decrease if either predictor variable increases by a unit while the other predictor variables are held constant. This is not logical since more first floor living space and more

[2] It is important to note that the Maximum R-Squared Improvement Selection technique results in 15 "best" models—a "best" 1-variable model, a "best" 2-variable model, etc. until it finds a "best" model using all the available predictor variables. We selected the "best" 12-variable model as Model_MaxR since it was consistent with the "best" models for the other automated variable selection techniques.

bedrooms should cause the sale price to increase. However, both of these predictor variables have strong ties to other predictor variables. For instance, FirstFlrSF represents a portion of GrLivArea, and BedroomAbvGr factors into the TotRmsAbvGrd variable. Further investigation into these variables will be necessary to determine if multicollinearity is present in the model. While all of the automated variable selection techniques resulted in the same parameter estimates depicted in **Table 3** above, Model_F, Model_B, and Model_S produced some additional summary tables. Both Model_F and Model_S resulted in the same summary table, shown in **Table 4**, while Model_B resulted in the summary table shown in **Table 5**.

Summary of Forward Selection/Summary of Stepwise Selection							
Step	Variable Entered	Number Vars In	Partial R-Square	Model R-Square	C(p)	F Value	Pr > F
1	OverallQual	1	0.6550	0.6550	2486.77	1923.61	<.0001
2	GrLivArea	2	0.1237	0.7788	1234.11	566.05	<.0001
3	BsmtFinSF1	3	0.0590	0.8377	638.206	367.42	<.0001
4	LotArea	4	0.0197	0.8574	440.454	139.58	<.0001
5	TotalBsmtSF	5	0.0142	0.8717	298.351	111.73	<.0001
6	YearBuilt	6	0.0100	0.8817	198.700	85.41	<.0001
7	OverallCond	7	0.0090	0.8906	109.894	82.46	<.0001
8	GarageArea	8	0.0060	0.8966	50.9906	58.46	<.0001
9	BedroomAbvGr	9	0.0033	0.8999	20.0222	32.64	<.0001
10	FirstFlrSF	10	0.0006	0.9005	16.0853	5.91	0.0153
11	LotFrontage	11	0.0005	0.9010	12.8046	5.28	0.0218
12	TotRmsAbvGrd	12	0.0003	0.9013	11.5551	3.25	0.0715

Table 4: Summary Table for Forward and Stepwise Selection

Summary of Backward Elimination							
Step	Variable Removed	Number Vars In	Partial R-Square	Model R-Square	C(p)	F Value	Pr > F
1	TotalBath_Calc	14	0.0000	0.9015	14.2304	0.23	0.6314
2	OpenPorchSF	13	0.0000	0.9014	12.6323	0.40	0.5261
3	BsmtFinSF2	12	0.0001	0.9013	11.5551	0.92	0.3366

Table 5: Summary Table for Backward Selection

For Model_F, Model_B, and Model_S, we set all the algorithm parameter values to the 0.10 significance level as the criteria for including or eliminating predictor variables in the model. Had we chosen different parameter values or gone with the default values for each selection technique, we would have ended up with different “best” models. As **Table 5** depicts, the eliminated variables have very low F-statistics and high p-values, which indicates that these variables did not contribute enough to the model to warrant staying in the model. In comparison, the predictor variables in **Table 4** reveal high F-statistics and low p-values.

Quantitative Measures of Goodness-of-Fit

Because all of the automated variable selection techniques resulted in the same model, Model_AdjR2, Model_MaxR, Model_MCp, Model_F, Model_B, and Model_S all had the same statistics. **Table 6** shows a summary of the statistics.

Number in Model	Adjusted R-Square	AIC	BIC	Variables in Model
12	0.9001	20412.9194	20415.2563	GrLivArea TotalBsmtSF LotFrontage LotArea BsmtFinSF1 FirstFlrSF GarageArea OverallQual OverallCond YearBuilt BedroomAbvGr TotRmsAbvGrd

Table 6: Summary Statistics Table for Each Model Fitted from the Training Set

Per the above table, Model_AdjR2, Model_MaxR, Model_MCp, Model_F, Model_B, and Model_S all had an adjusted R-Squared value of 0.9001, which means that the "best" model accounted for approximately 90 percent of the variation in the response variable, train_response. The AIC and BIC values allow us to compare models containing different number of predictor variables, and the models with the smallest AIC and BIC values are considered the "best." To put the AIC and BIC values in **Table 6** in context, **Table 7** depicts the statistics for the top 5 models identified using the Adjusted R-Squared selection technique. The first model in **Table 7** is the same model in **Table 6**. As **Table 7** shows, the AIC and BIC values increase in size the further the models are from the top of the list, thus indicating worse models even if the Adjusted R-Squared values are all the same.

Number in Model	Adjusted R-Square	AIC	BIC	Variables in Model
12	0.9001	20412.9194	20415.2943	GrLivArea TotalBsmtSF LotFrontage LotArea BsmtFinSF1 FirstFlrSF GarageArea OverallQual OverallCond YearBuilt BedroomAbvGr TotRmsAbvGrd
13	0.9001	20413.9828	20416.4128	GrLivArea TotalBsmtSF LotFrontage LotArea BsmtFinSF1 BsmtFinSF2 FirstFlrSF GarageArea OverallQual OverallCond YearBuilt BedroomAbvGr TotRmsAbvGrd
13	0.9001	20414.5744	20416.9879	GrLivArea TotalBsmtSF LotFrontage LotArea BsmtFinSF1 FirstFlrSF GarageArea OpenPorchSF OverallQual OverallCond YearBuilt BedroomAbvGr TotRmsAbvGrd
14	0.9001	20415.5746	20418.0475	GrLivArea TotalBsmtSF LotFrontage LotArea BsmtFinSF1 BsmtFinSF2 FirstFlrSF GarageArea OpenPorchSF OverallQual OverallCond YearBuilt BedroomAbvGr TotRmsAbvGrd
14	0.9001	20415.6987	20418.1679	GrLivArea TotalBsmtSF LotFrontage LotArea BsmtFinSF1 BsmtFinSF2 FirstFlrSF GarageArea OverallQual OverallCond YearBuilt BedroomAbvGr TotRmsAbvGrd TotalBath_Calc

Table 7: Automated Variable Selection Statistics Table for Model_AdjR2

In **Table 8** below, we computed the MSE and MAE for both the testing and the training data sets. The first row of the table shows the default computations for the entire sample population, but the second row shows the statistics for the testing set, while the third row shows the statistics for the training set. Both the MSE and MAE are considerably higher for the testing data set, which indicates that the models performed worse in the test sample than the training sample. Ideally, the MSE and MAE for the testing set would be similar in value to the MSE and MAE for the training set.

Obs	train	_TYPE_	_FREQ_	MSE	MAE
1	.	0	1831	683615896.53	17868.48
2	0	1	579	1029547617.51	19547.04
3	1	1	1252	528543056.09	17116.02

Table 8: MSE and MAE Computations for Training and Testing Sets

From **Table 8** we can conclude that the six automated variable selection techniques resulted in overfitting the "best" model for Model_AdjR2, Model_MaxR, Model_MCp, Model_F, Model_B, and Model_S since the "best" model fits better in-sample than it predicts out-of-sample. This is something that we should further investigate and try to correct before deploying the model in a business application.

Multicollinearity Assessment

Although Model_AdjR2, Model_MaxR, Model_MCp, Model_F, Model_B, and Model_S all resulted in the same "best" 12-variable model, it is important to examine the model for signs of multicollinearity. To do this, we calculated the variance inflation factors (VIFs) for each predictor variable present in the model. **Table 9** below is the same table as **Table 3**, but we reprinted it here for convenience during our discussion.

Parameter Estimates						
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Variance Inflation
Intercept	1	-906548	67106	-13.51	<.0001	0
GrLivArea	1	65.49045	3.26646	20.05	<.0001	4.84545
TotalBsmtSF	1	34.72988	4.06903	8.54	<.0001	4.35435
LotFrontage	1	102.96549	46.67642	2.21	0.0276	1.38700
LotArea	1	2.06187	0.20048	10.28	<.0001	1.35114
BsmtFinSF1	1	21.02109	2.10742	9.97	<.0001	1.47357
FirstFlrSF	1	-11.51894	4.31078	-2.67	0.0077	4.45458
GarageArea	1	36.50327	5.21911	6.99	<.0001	1.82734
OverallQual	1	16178	912.82115	17.72	<.0001	2.59058
OverallCond	1	7761.24864	763.30773	10.17	<.0001	1.36072
YearBuilt	1	397.87825	34.36038	11.58	<.0001	2.05451
BedroomAbvGr	1	-9269.91594	1438.21338	-6.45	<.0001	1.97256
TotRmsAbvGrd	1	1823.32997	1010.75147	1.80	0.0715	3.81318

Table 9: VIF Computations for Each Model Fitted from the Training Set

Typically a VIF greater than 10 is the threshold for determining if a model has a serious problem with multicollinearity.[3] In the table below, all of the predictor variables have a VIF less than 5. This suggests that multicollinearity is not a major issue for this model; however, a VIF greater

[3] Montgomery, D.C., Peck, E.A., and Vining, G.G. (2012). *Introduction to Linear Regression Analysis*. (5th Edition). New York, NY: Wiley

than 1 indicates that there is correlation among the predictor variables. Earlier we mentioned that the parameter estimates for FirstFlrSF and BedroomAbvGr are not as expected since they cause the sale price to decrease as they increase. Since FirstFlrSF factors into the total above ground living space represented by GrLivaArea, and since BedroomAbvGr factor into the count of total rooms above ground represented by TotRmsAbvGrd, it is important that we look closer at the relationships between these variables. In the table above, GrLivArea, TotalBsmtSF, FirstFlrSF, and TotRmsAbvGrd have the highest VIFs of the predictor variables in the model. Each of these variables either help make up a total square footage of some house feature or represent the total square footage of some house feature. Taking a closer look at all the variables that logically have a relationship, even if they do not have a high VIF, can help improve the "best" model.

Operational Validation Assessment

The purpose of model building is rarely to see if we can validate a model in just the statistical sense. Rather, the ultimate goal of model building is to build a model that can be deployed and produce accurate predictions for some practical or business application. Validating a model in the statistical sense does not always equate to the model being suitable for a business application. From **Table 8** we discovered that the "best" model we found using the six automated variable selection techniques performed much worse out-of-sample than it did in-sample by comparing the MSE and MAE values for the testing and training sets. In **Table 10** below, we defined the predictive accuracy of the model by assigning it Grade 1 if the predicted value was within 10 percent of the actual value, Grade 2 if the predicted value was within 15 percent of the actual value, and Grade 3 if the predicted value was greater than 15 percent of the actual value. As the table shows, approximately 67 percent of the observations had predicted values that were within 10 percent of the actual value, and approximately 16 percent of the observations had predicted values that were within 15 percent of the actual value. Only 17 percent of observations had worse than a 15 percent accuracy grade.

Prediction Grade	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Grade 1	1229	67.12	1229	67.12
Grade 2	286	15.62	1515	82.74
Grade 3	316	17.26	1831	100.00

Table 10: Operational Accuracy Assessment

What the above table means in a statistical sense differs from what it means in a business sense. Statistically, having the majority, or 67 percent, of observations fall in Grade 1 is pretty good since it means that most predicted values were within 10 percent of the actual values. However, in a business sense, this might not be acceptable. For example, if a business is building a model to predict how well a new resource-intensive policy or an expensive product will perform, seeing the table above might be a red flag that the proposed policy or product is too much of a risk. **Table 10** shows that approximately a third of the predictions differ from the actual values by more than 10 percent. Not only is this a lot of inaccurate predictions, but a greater than 10 percent difference between predicted and actual values might have severe financial consequences for the company.

Whenever we try to validate a model in the business sense, we have to go beyond the statistical validation and consider how the model might perform with respect to the business's specific goals, resources, willingness to accept risk, and the potential payoff the model might have.

Conclusions

In this report we created a predictive modeling framework by partitioning our data, identified a "best" model using several automated variable selection techniques, assessed the multicollinearity presence in the model, used cross-validation to assess the predictive accuracy of the model, and assessed how validating a model in the statistical sense differs from validating it in the operational or business sense. From the six automated variable selection techniques we explored, we discovered that each technique resulted in the same 12-variable model from the 15 candidate variables. Although we theoretically selected enough predictor variables to use with the automated variable selection techniques, the fact that each technique produced the same "best" model indicates that we either should have included more predictor variables or changed some of the predictor variables we included. Oftentimes a couple selection techniques will suggest different "best" models than the rest of the selection techniques. Deciding which "best" model is actually best requires familiarity with the data or subject matter knowledge. Although we did not have to pick a "best" model since all the automated variable selection techniques chose the same one, we can see from the calculated MSE and MAE values, as well as the prediction grades, that our "best" model still has some room for improvement. In the future, we would like to perform the automated variable selection techniques explored in this report with more variables, including some key categorical variables that we can convert into discrete variables. We think this will enable us to create a better model for predicting single-family home sale prices in Ames, Iowa.

Code

```
*************************************************************************;
* Preliminary Steps;
*************************************************************************;

* Access library where Ames housing dataset is stored;
libname mydata '/scs/crb519/PREDICT_410/SAS_Data/' access=readonly;
proc datasets library=mydata; run; quit;

* Set Ames housing dataset to short name;
data ames;
set mydata.AMES_HOUSING_DATA;
run;

*************************************************************************;
* Define the Sample Population;
*************************************************************************;

* Create drop conditions;
data temp;
	set ames;
	format drop_condition $40.;
	if (BldgType ne '1Fam') then drop_condition='01: Not a Single Family Home';
	else if (Zoning = 'A' or zoning = 'C' or zoning = 'FV' or zoning = 'I' or zoning = 'RP')
		then drop_condition='02: Non-Residential Zone or RV Park';
	else if (SaleCondition ne 'Normal') then drop_condition='03: Not Normal Sale Condition';
	else if (GrLivArea >= 4000) then drop_condition='04: Atypically Large House';
	else if (TotalBsmtSF < 1) then drop_condition='05: No Basement';
	else if (GarageArea < 1) then drop_condition='06: No Garage';
	else if (SalePrice <= 0) then drop_condition='07: Pricing Error';
	else if (Utilities ne 'AllPub') then drop_condition='08: Not All Public Utilities Available';
else drop_condition='09: Sample Population';
run;

* View frequency of drop conditions;
proc freq data=temp;
tables drop_condition;
title 'Sample Population Waterfall';
run; quit; * (Reveals that all Sale Prices are positive);

* Subset data for just Sample Population and Define TotalBath_Calc;
data sample;
set temp;
if (drop_condition='09: Sample Population');
TotalBath = max(FullBath,0) + max(BsmtFullBath,0);
TotalHalfBath = max(HalfBath,0) + max(BsmtHalfBath,0);
TotalBath_Calc = TotalBath + TotalHalfBath;
title;
run;
```

```
*************************************************************************;
* Create a Train/Test Split of Data for Cross-Validation;
*************************************************************************;

data test_train;
set sample;
* 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;

proc contents data=test_train; run;

* Print first 10 observations as preview of data;
Title "Preview of Dataset";
options obs=10;
proc print data=test_train; run;
options obs=max;

* View frequency test/train data;
proc freq data=test_train;
tables train;
title 'Observation Counts for Test/Train Data Partition';
run; quit;

/*
If we wanted to create two separate data sets from the split,
this is how we would code it.

* Define Train Data Set;
data ames_train;
set test_train;
if (train = 1);
run;

* Define Test Data Set;
data ames_test;
set test_train;
if (train = 0);
run;

title "Preview of Training Dataset";
options obs=10;
proc print data=ames_train; run;
options obs=max;

title "Preview of Testing Dataset";
```

```
options obs=10;
proc print data=ames_test; run;
options obs=max;
*/

**********************************************************************;
* Model Identification;
**********************************************************************;

* (Orignally included BsmtUnfSF as a predictor but removed it since it was
showing bias/multicollinearity with BsmtFinSF1 BsmtFinSF2);

* Adjusted R-Squared Selection;
ods graphics on;
title "Automated Variable Selection Using Adjusted R-Squared Selection";
proc reg data=test_train outest=Model_AdjR2_out;
Model_AdjR2: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 BsmtFinSF2 FirstFlrSF
GarageArea OpenPorchSF OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd TotalBath_Calc / adjrsq aic bic mse cp vif selection =
adjrsq best=10;
run;
ods graphics off;

proc print data=Model_AdjR2_out; run; * This contains parameter estimates for all models using
the Adjusted R-Squared Selection;

* NB: Running proc reg with data=test_train and train_response, or data=ames_train and
SalePrice
all use the same number of observations, 1015, as the code above even if they read different
amounts of variables,
1831 v. 1252. They all result in the same variable selection.;

* Adjusted R-Squared Selection with Variables Removed/Multicollinearity Assessment Using
VIFs;
ods graphics on;
title "Adjusted R-Squared Selection with Variables Removed";
proc reg data=test_train outest=Model_AdjR2_out2;
Model_AdjR2: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 /*BsmtFinSF2*/ FirstFlrSF
GarageArea /*OpenPorchSF*/ OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd /*TotalBath_Calc*/ / selection=rsquare start=12
stop=12 adjrsq aic bic mse cp vif;
output out = AdjR2_fit pred=yhat;
run;
ods graphics off;

proc print data=Model_AdjR2_out2; run; * This contains parameter estimates for just the best
model from the Adjusted R-Squared Selection;

* Defining variables for MSE, MAE, and predictive accuracy computations;
```

```
data AdjR2_fit;
set AdjR2_fit;
title;
residual = SalePrice - yhat;
abs_res = abs(residual);
sq_res = residual*residual;
mse = mean(sq_res); * Not really necessary since, for a single observation, mse=sq_res and
mae=abs_res;
mae = mean(abs_res);
accuracy = abs_res / SalePrice;
format Prediction_Grade $7.;
if accuracy <= 0.1 then Prediction_Grade = 'Grade 1';
else if accuracy <= 0.15 then Prediction_Grade = 'Grade 2';
else Prediction_Grade = 'Grade 3';
run;

proc print data=AdjR2_fit (obs=10);
run;

title "Model_AdjR2's MSE and MAE";
proc means data=AdjR2_fit n mean;
class train;
var sq_res abs_res;
output out = msemae_AdjR2
mean(sq_res abs_res) = MSE MAE;
proc print data = msemae_AdjR2;
run; quit;

title "Model_AdjR2's Operational Accuracy";
proc freq data = AdjR2_fit;
tables Prediction_Grade;
run;

**********************************************************************;

* Maximum R-Squared Selection;
ods graphics on;
title "Automated Variable Selection Using Maximum R-Squared Selection";
proc reg data=test_train outest=Model_MaxR_out;
Model_MaxR: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 BsmtFinSF2 FirstFlrSF
GarageArea OpenPorchSF OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd TotalBath_Calc / adjrsq aic bic mse cp vif selection =
maxr;
run;
ods graphics off;

proc print data=Model_MaxR_out; run;

* Maximum R-Squared Selection with Variables Removed/Multicollinearity Assessment Using
VIFs;
```

```
ods graphics on;
title "Maximum R-Squared Selection with Variables Removed";
proc reg data=test_train outest=Model_MaxR_out2;
Model_MaxR: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 /*BsmtFinSF2*/ FirstFlrSF
GarageArea /*OpenPorchSF*/ OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd /*TotalBath_Calc*/ / selection=rsquare start=12
stop=12 adjrsq aic bic mse cp vif;
output out = MaxR_fit pred=yhat;
run;
ods graphics off;

proc print data=Model_MaxR_out2; run;


* Defining variables for MSE, MAE, and predictive accuracy computations;
data MaxR_fit;
set MaxR_fit;
title;
residual = SalePrice - yhat;
abs_res = abs(residual);
sq_res = residual*residual;
accuracy = abs_res / SalePrice;
format Prediction_Grade $7.;
if accuracy <= 0.1 then Prediction_Grade = 'Grade 1';
else if accuracy <= 0.15 then Prediction_Grade = 'Grade 2';
else Prediction_Grade = 'Grade 3';
run;

proc print data=MaxR_fit (obs=10);
run;

title "Model_MaxR's MSE and MAE";
proc means data=MaxR_fit n mean;
class train;
var sq_res abs_res;
output out = msemae_MaxR
mean(sq_res abs_res) = MSE MAE;
proc print data = msemae_MaxR;
run; quit;

title "Model_MaxR's Operational Accuracy";
proc freq data = MaxR_fit;
tables Prediction_Grade;
run;

**********************************************************************;

* Mallow's Cp Selection;
ods graphics on;
title "Automated Variable Selection Using Mallow's Cp Selection";
```

```
proc reg data=test_train outest=Model_MCp_out;
Model_MCp: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 BsmtFinSF2 FirstFlrSF
GarageArea OpenPorchSF OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd TotalBath_Calc / adjrsq aic bic mse cp vif selection =
cp best=10;
run;
ods graphics off;

proc print data=Model_MCp_out; run;

* Mallow's Cp Selection with Variables Removed/Multicollinearity Assessment Using VIFs;
* (This selection resulted in the same model as Model_AdjR2.);
ods graphics on;
title "Mallow's Cp Selection with Variables Removed";
proc reg data=test_train outest=Model_MCp_out2;
Model_MCp: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 /*BsmtFinSF2*/ FirstFlrSF
GarageArea /*OpenPorchSF*/ OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd /*TotalBath_Calc*/ / selection=rsquare start=12
stop=12 adjrsq aic bic mse cp vif;
output out = MCp_fit pred=yhat;
run;
ods graphics off;

proc print data=Model_MCp_out2; run;

* Defining variables for MSE, MAE, and predictive accuracy computations;
data MCp_fit;
set MCp_fit;
title;
residual = SalePrice - yhat;
abs_res = abs(residual);
sq_res = residual*residual;
accuracy = abs_res / SalePrice;
format Prediction_Grade $7.;
if accuracy <= 0.1 then Prediction_Grade = 'Grade 1';
else if accuracy <= 0.15 then Prediction_Grade = 'Grade 2';
else Prediction_Grade = 'Grade 3';
run;

proc print data=MCp_fit (obs=10);
run;

title "Model_MCp's MSE and MAE";
proc means data=MCp_fit n mean;
class train;
var sq_res abs_res;
output out = msemae_MCp
mean(sq_res abs_res) = MSE MAE;
proc print data = msemae_MCp;
```

```
run; quit;

title "Model_MCp's Operational Accuracy";
proc freq data = MCp_fit;
tables Prediction_Grade;
run;


**********************************************************************;

* Forward Selection;
ods graphics on;
title "Automated Variable Selection Using Forward Selection";
proc reg data=test_train outest=Model_F_out;
Model_F: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 BsmtFinSF2 FirstFlrSF
GarageArea OpenPorchSF OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd TotalBath_Calc / adjrsq aic bic mse cp vif selection =
forward slentry=0.10;
run;
ods graphics off;

proc print data=Model_F_out (obs=10); run;

* Forward Selection with Variables Removed/Multicollinearity Assessment Using VIFs;
* (This selection resulted in the same model as Model_AdjR2 and Model_MCp.);
ods graphics on;
title "Forward Selection with Variables Removed";
proc reg data=test_train outest=Model_F_out2;
Model_F: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 /*BsmtFinSF2*/ FirstFlrSF
GarageArea /*OpenPorchSF*/ OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd /*TotalBath_Calc*/ / selection=rsquare start=12
stop=12 adjrsq aic bic mse cp vif;
output out = F_fit pred=yhat;
run;
ods graphics off;

proc print data=Model_F_out2; run;

* Defining variables for MSE, MAE, and predictive accuracy computations;
data F_fit;
set F_fit;
title;
residual = SalePrice - yhat;
abs_res = abs(residual);
sq_res = residual*residual;
accuracy = abs_res / SalePrice;
format Prediction_Grade $7.;
if accuracy <= 0.1 then Prediction_Grade = 'Grade 1';
else if accuracy <= 0.15 then Prediction_Grade = 'Grade 2';
```

```
else Prediction_Grade = 'Grade 3';
run;

proc print data=F_fit (obs=10);
run;

title "Model_F's MSE and MAE";
proc means data=F_fit n mean;
class train;
var sq_res abs_res;
output out = msemae_F
mean(sq_res abs_res) = MSE MAE;
proc print data = msemae_F;
run; quit;

title "Model_F's Operational Accuracy";
proc freq data = F_fit;
tables Prediction_Grade;
run;

**********************************************************************;

* Backward Selection;
ods graphics on;
title "Automated Variable Selection Using Backward Selection";
proc reg data=test_train outest=Model_B_out;
Model_B: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 BsmtFinSF2 FirstFlrSF
GarageArea OpenPorchSF OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd TotalBath_Calc / adjrsq aic bic mse cp vif selection =
backward slstay=0.10;
run;
ods graphics off;

proc print data=Model_B_out (obs=10); run;

* Backward Selection with Variables Removed/Multicollinearity Assessment Using VIFs;
* (This selection resulted in the same model as Model_AdjR2, Model_MCp, and Model_F.);
ods graphics on;
title "Backward Selection with Variables Removed";
proc reg data=test_train outest=Model_B_out2;
Model_B: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 /*BsmtFinSF2*/ FirstFlrSF
GarageArea /*OpenPorchSF*/ OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd /*TotalBath_Calc*/ / selection=rsquare start=12
stop=12 adjrsq aic bic mse cp vif;
output out = B_fit pred=yhat;
run;
ods graphics off;

proc print data=Model_B_out2; run;
```

```
* Defining variables for MSE, MAE, and predictive accuracy computations;
data B_fit;
set B_fit;
title;
residual = SalePrice - yhat;
abs_res = abs(residual);
sq_res = residual*residual;
accuracy = abs_res / SalePrice;
format Prediction_Grade $7.;
if accuracy <= 0.1 then Prediction_Grade = 'Grade 1';
else if accuracy <= 0.15 then Prediction_Grade = 'Grade 2';
else Prediction_Grade = 'Grade 3';
run;

proc print data=B_fit (obs=10);
run;

title "Model_B's MSE and MAE";
proc means data=B_fit n mean;
class train;
var sq_res abs_res;
output out = msemae_B
mean(sq_res abs_res) = MSE MAE;
proc print data = msemae_B;
run; quit;

title "Model_B's Operational Accuracy";
proc freq data = B_fit;
tables Prediction_Grade;
run;

**********************************************************************;

* Stepwise Selection;
ods graphics on;
title "Automated Variable Selection Using Stepwise Selection";
proc reg data=test_train outest=Model_S_out;
Model_S: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 BsmtFinSF2 FirstFlrSF
GarageArea OpenPorchSF OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd TotalBath_Calc / adjrsq aic bic mse cp vif selection =
stepwise slentry =0.10 slstay=0.10;
run;
ods graphics off;

proc print data=Model_S_out (obs=10); run;

* Stepwise Selection with Variables Removed/Multicollinearity Assessment Using VIFs;
* (This selection resulted in the same model as Model_AdjR2, Model_MCp, Model_F, and
Model_B.);
```

```
ods graphics on;
title "Stepwise Selection with Variables Removed";
proc reg data=test_train outest=Model_S_out2;
Model_S: model train_response = GrLivArea TotalBsmtSF LotFrontage
LotArea BsmtFinSF1 /*BsmtFinSF2*/ FirstFlrSF
GarageArea /*OpenPorchSF*/ OverallQual OverallCond
YearBuilt BedroomAbvGr TotRmsAbvGrd /*TotalBath_Calc*/ / selection=rsquare start=12
stop=12 adjrsq aic bic mse cp vif;
output out = S_fit pred=yhat;
run;
ods graphics off;

proc print data=Model_S_out2; run;

* Defining variables for MSE, MAE, and predictive accuracy computations;
data S_fit;
set S_fit;
title;
residual = SalePrice - yhat;
abs_res = abs(residual);
sq_res = residual*residual;
accuracy = abs_res / SalePrice;
format Prediction_Grade $7.;
if accuracy <= 0.1 then Prediction_Grade = 'Grade 1';
else if accuracy <= 0.15 then Prediction_Grade = 'Grade 2';
else Prediction_Grade = 'Grade 3';
run;

proc print data=S_fit (obs=10);
run;

title "Model_S's MSE and MAE";
proc means data=S_fit n mean;
class train;
var sq_res abs_res;
output out = msemae_S
mean(sq_res abs_res) = MSE MAE;
proc print data = msemae_S;
run; quit;

title "Model_S's Operational Accuracy";
proc freq data = S_fit;
tables Prediction_Grade;
run;

**********************************************************************;
* END;
**********************************************************************;
```