Assignment #4

Scott M. Morgan

Introduction:

The purpose of this assignment is to continue building regression models to predict home sale price. To accomplish this objective, I use data from the Ames housing data set provided by the course instructor and SAS Studio to perform the analysis. In a world where data is becoming increasingly abundant, analysis is often conducted and conclusions are drawn naively without the appropriate formation, statistical knowledge and validation. The accessibility of free open source tools that leverage R and Python puts analytical power in the hands of anyone who can download software. This has precarious implications for business decision makers who rely on predictive analytics to make informed, forward-looking decisions. In this exercise, we use functionality within SAS to complete our end-to-end predictive modeling process by using an assortment of observed and categorical characteristics to predict the sales price of a home. Specifically, we will dummy code categorical variables, explore automated variable selection procedures and validate our models using data splitting as well as predictive grading. I believe this exercise will reinforce the notion that modeling is an iterative process where there is seldom a singular metric that points to the "best" model, but rather a confluence of factors which includes statistical analysis, business acumen and logical reasoning.

Results:

In the subsequent sections, we dummy code categorical variables, perform automated variable selection procedures and present a framework for model validation.

PART A: Dummy Coding Categorical Variables. In this initial section, we select 2 character variables that have between 3 to 10 categories that are analytically appealing in terms of predicting SalePrice (Y).

Model 1: Categorical Variables (No Dummy Coding). There are numerous categorical variables in the Ames housing data; many are subjective assessments of the quality of the home. To avoid possible bias and create the most robust model possible, we select a categorical variable that is based on factual features of the home. The BldgType, in particular, represents the type of dwelling and has 5 different categories which are given in Table 1 below:

Table 1: BldgType Categories

Category	Description
1Fam	Single-family Detached
2fmCon	Two-family Conversion; originally built as one-family dwelling
Duplex	Duplex
Twnhs	Townhouse End Unit
TwnhsE	Townhouse Inside Unit

It makes sense intuitively that different building types would be related to home price in some capacity. We might expect townhouses (whether end unit or inside unit) to be generally more expensive than duplexes but less expensive than two family conversions and single-family detached homes. Using PROC SORT and PROC MEANS BY, the means, standard deviations, maximums and minimums of the categories are provided in Table 2 below:

Table 2: Summary of BldgType

BldgType	N	Mean	Std Dev	Minimum	Maximum
1Fam	2425	184,812	82,822	12,789	755,000
2fmCon	62	125,582	31,089	55,000	228,950
Duplex	109	139,809	39,499	61,500	269,500
Twnhs	101	135,934	41,939	73,000	280,750
TwnhsE	233	192,312	66,192	71,000	392,500

The majority of homes are single-family detached units and there does appears to be the most variation in SalePrice in this category as well. Townhouse Inside Units are the most expensive homes on average while the two family conversion homes are the cheapest. Both of these outcomes were the opposite of the original hypothesis above. Before fitting our initial model, we perform a simple transformation of the categorical variables into numerical variables. This new variable is denoted as buildtype. Using the SAS regression procedure, we generate the following parameter estimates for Model 1 (Table 3):

Table 3: Parameter Estimates for buildtype to Predict SalePrice

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	186102	2353.59880	79.07	<.0001
buildtype	1	-3497.27215	1209.49086	-2.89	0.0039

This results in the following model in equation form:

SalePrice = 186102+ -3497.27215 x buildtype

Within the context of this model, if buildtype were equal to 0 then we expect SalePrice to be $186,102. However, the interpretation of the coefficient does not make sense as a 1 unit change in builtype is not consistent across every possible value of the variable. There is no “0” building type and the numerical variables are nominal data, which are unmeasured and simple distinct categories. Additionally, when dealing with categorical variables it is preferred that the predicted group means pass through the means of the actual groups. Table 4 below illustrates the consistent over and under prediction of the group means.

Table 4: Predict Group Means

Category	Model Means	Group Means	Over/Under Prediction
1Fam (1)	182,608	184,812	Under
2fmCon (2)	179,110	125,582	Over
Duplex (3)	175,613	139,809	Over
Twnhs (4)	172,116	135,934	Over
TwnhsE (5)	168,619	192,312	Under

We next shift our attention to the automatically generated ODS output from SAS to assess the goodness-of-fit for this model. Of the 2930 observations in the sample there were no missing values. The F Value is small but still statistically significant, so we can reject the null hypothesis that there is no linear relationship between SalePrice and buildtype. R-Squared and Adjusted R-squared are both less than 1%, which is obviously far lower than we would like. The scatterplot for residual values appears to be somewhat concentrated in the single-family home category; however given the nature of the variable interpreting this plot is not particularly helpful. The Q-Q plot of the residuals suggests a non-normal distribution as the residuals trail off in both the lower left and upper right corner of the chart indicating positive skewness. The presence of 2 possible outliers is also apparent. The histogram of the residuals describes a leptokurtic distribution with positive skewness. Cook's D highlights the presence of several potential outliers as well numerous values that are above the threshold and potentially influential. The left pane of the Fit-Mean Residual plot is much lower than the right, indicating that there is a large amount of variation not explained by the model.

Given the coefficient interpretation problem, inconsistent prediction of the groups means and poor overall fit, the recommendation is to not use categorical variables with 3 or more categories as a predictor in the regression model. The alternative option is to use dummy coding, which is discussed next.

Model 2: Categorical Variables (Dummy Coding). In this section, we generate a regression model using dummy coding, report the model in equation form, interpret each coefficient, report the hypothesis tests for each beta and, lastly, prepare another variable for automated variable selection (Part B).

Table 5 below provides the frequency of the BldgType. For Model 2, the basis of interpretation will be 2fmCon (Two-family Conversion) as it is the variable with the smallest frequency.

Table 5: Frequency of BldgType Variable

BldgType	Frequency	Percent	Cumulative Frequency	Cumulative Percent
1Fam	2,425	83	2,425	83
2fmCon	62	2	2,487	85
Duplex	109	4	2,596	89
Twnhs	101	3	2,697	92
TwnhsE	233	8	2,930	100

Before fitting our second model, we perform another transformation of the categorical variables into dummy variables. Using the SAS regression procedure, we generate the following parameter estimates for Model 2 (Table 6):

Table 6: Parameter Estimates for BldgType dummy codes to Predict SalePrice

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	125,582	9,975.74285	12.59	<.0001
buildtype_1Fam	1	59,230	10,102	5.86	<.0001
buildtype_Duplex	1	14,227	12,495	1.14	0.2549
buildtype_Twnhs	1	10,352	12,673	0.82	0.4141
buildtype_TwnhsE	1	66,730	11,225	5.94	<.000

This results in the following model in equation form. Note 2fmCon is the basis for interpretation:

SalePrice = 125,582 + 59,230 x buildtype_1Fam + 14,227 x buildtype_Duplex + 10,352 x buildtype_Twnhs + 66,730 x buildtype_TwnhsE

We now interpret each of the model coefficients. If BldgType is 1Fam, then the resulting model is:

SalePrice = 125,582 + 59,230 x buildtype_1Fam

Within the context of this model, if BldgType is 1Fam, then we expect SalePrice to be $184,812. This is compared to the group mean of $184,812.

If BldgType is Duplex, then the resulting model is:

SalePrice = 125,582 + 14,227 x buildtype_Duplex

Within the context of this model, if BldgType is Duplex, then we expect SalePrice to be $139,809. This is compared to the group mean of $ 139,809.

If BldgType is Twnhs, then the resulting model is:

SalePrice = 125,582 + 10,352 x buildtype_Twnhs

Within the context of this model, if BldgType is Twnhs, then we expect SalePrice to be $135,934. This is compared to the group mean of $135,934.

If BldgType is TwnhsE, then the resulting model is:

SalePrice = 125,582 + 66,730 x buildtype_TwnhsE

Within the context of this model, if BldgType is Twnhs, then we expect SalePrice to be $192,312. This is compared to the group mean of $192,312.

Lastly, if BldgType is 2fmCon, then the resulting model is:

SalePrice = 125,582

Within the context of this model, if BldgType is 2fmCon, then we expect SalePrice to be $125,582. This is compared to the group mean of $125,582.

We next shift our attention to the automatically generated ODS output from SAS to assess the goodness-of-fit for this model. Of the 2930 observations in the sample there were no missing values. The F Value is larger than in Model 1 though still small but statistically significant, so we can reject the null hypothesis that there is no linear relationship between SalePrice and buildtype. R-Squared and Adjusted R-squared are larger than the previous model at about 3% which is still lower than we would like. The Root Mean Square (Root MSE) is about the same between the models as well. The various fit visualizations are largely the same as Model 1. While the predicted model did go through the mean of Y in each category, it's difficult to conclude beyond this that Model 2 is an improvement to Model 1.

The hypothesis test for each of the betas presented below.

Null

H_0: $\beta_1 = \beta_2 = \ldots \beta_4 = 0$

Alternative

H_1: $\beta_{1\ldots4} \neq 0$

As displayed in Table 6 earlier, the t Value is a test statistic with a student's t distribution and the Pr > |t| is the p-value associated with that test statistic. Buildtype_1Fam and buildtype_TwnhsE are statistically significant while buildtype_Duplex and buildtype_Twnhs are not. In the cases of buildtype_Duplex and buildtype_Twnhs, we cannot reject the null hypothesis that the coefficient is equal to zero or, phrased differently, there appears to be no significant linear correlation between these predictors and the response.

Model 3: Additional Categorical Variable and Preparation for Automated Selection. Lastly, we will create another dummy coded variable that has at least 3 categories. We are still interested in an

objective categorical variable with between 3 to 10 categories. As such, we will use SaleCondition as our next categorical variable to dummy code. Table 7 below provides a description of these categories. Note that in a professional environment, we would be creating dummy coded variables for all categorical and character variables of interest.

Table 7: SaleCondition Categories

Category	Description
Abnormal	Abnormal Sale - trade, foreclosure, short sale
AdjLand	Adjoining Land Purchase
Alloca	Allocation - two linked properties with separate deeds, typically condo with a garage unit
Family	Sale between family members
Normal	Normal Sale
Partial	Home was not completed when last assessed (associated with New Homes)

It makes sense that the type of transaction could be indicative of SalePrice. We might expect Adnormal and Family sales, for example, to be less than Normal and Partial, or new home, purchases. Using PROC SORT and PROC MEANS BY, the means, standard deviations, maximums and minimums of the categories are provided in Table 8 below:

Table 8: Summary of SaleCondition

SaleCondition	N	Mean	Std Dev	Minimum	Maximum
Abnormal	190	140,396	80,443	12,789	745,000
AdjLand	12	108,917	21,988	81,000	150,000
Alloca	24	161,844	72,214	50,138	359,100
Family	46	157,489	63,377	79,275	409,900
Normal	2,413	175,568	70,980	35,000	755,000
Partial	245	273,374	100,001	113,000	611,657

Our hypothesis that Abnormal and Family transactions are cheaper than Normal and Partial transactions was accurate. In fact, the price of new homes (Partial) in the area is substantially higher than the other categories. It would be interesting to investigate the relationship between year constructed or remodeled and SalePrice. While those variables are considered categorical and very well could provide interesting insight, their inclusion in these models is beyond the scope of this exercise as they have greater than 10 categories. AdjLand will be the basis of interpretation as it has the smallest frequency.

PART B: Automated Variable Selection. Automated variable selection methods are useful when there are many variables to evaluate and manual selection becomes cumbersome. While there is debate on whether variable selection should be automated, for this exercise we use six common computational techniques for generating subset regression models to arrive at Model 3; which will then be used in the subsequent cross-validation. According Montgomery, Peck, and Vining, (2012), we should expect that

the different selection algorithms will not necessarily lead to the same choice of final model. The main criteria we will be evaluating on are the F-value, Root MSE, Adjusted R-squared, Mallow's Cp, AIC and BIC. Our guiding rules are to maximize F-value and Adjusted R-squared and minimize Root MSE, Mallow's Cp, AIC and BIC. We will be using all possible continuous predictor variables from the data set as well as both dummy coded variables (BldgType and SaleCondition). The two bases for interpretation variables, 2fmCon and AdjLand, were excluded from the model as they were already taken into consideration when the dummy variables were created. Using the SAS functionality for PROC REG and SELECTION, the following output (Table 9) was generated:

Table 9: Automated Variable Selection Summary

Model	Adjusted R-Squared	Mallow's Cp	AIC	Forward	Backward	Stepwise
Ind. Variables	17	16	16	21	17	16
Missing Values	672	214	214	672	214	214
F-Value	407.58	510.09	510.09	329.82	407.3	432.83
Root MSE	40930	39518	39518	40944	39518	39518
R-Square	0.7557	0.7555	0.7555	0.7560	0.7556	0.7515
Adj R-Sq	0.7538	0.7538	0.7538	0.7537	0.7537	0.7500
Cp	13.6971	13.3012	13.3012	19.3129	14.8343	13.3012
AIC	47976.0811	47975.7006	47975.7006	47981.6718	47977.2294	47975.7006
BIC	47978.44	47978.015	47978.015	47984.1579	47979.5698	47978.015

Overall, there was dramatic improvement by the inclusion of the continuous variables. As expected, the different variables selection procedures selected several distinct models. Note that the Mallow's Cp and AIC techniques identified the same model. The different techniques selected models with largely the same number predictor variables, except for Forward selection which has 21 x variables. The Backward and Stepwise techniques also selected models that were very similar to Mallow's Cp/AIC. It is important to highlight that the Adjusted R-squared and Forward models are both missing about 23% of the values (672 out of a possible 2930). As this represents almost a quarter of the sample size, we eliminate these two from contention as they are not marginally better than the other option. The Mallow's Cp/AIC model has the largest F-value and lowest Mallow's Cp measurement. The remaining metrics are largely the same between the models. At this point, the recommendation is to use the model selected by the Mallow's Cp and AIC techniques (i.e. Model 3).

Each of the models selected contained at least one dummy coded variable. As most predictive modelers agree that if a dummy coded variable is selected by an automated procedure, all of the dummy coded variables for that categorical variable need to be included in the final model, whether they are statistically significant or not, for interpretive reasons. The final model should also contain only those continuous variables that are statistically significant. To conform with these standards, we include all dummy coded variables for both BldgType and SaleCondition, except for the base of interpretations, and remove the continuous variable LotArea as it was not statistically significant.

Using the SAS regression procedure, we generate the following parameter estimates for the ADJUSTED model (Table 10). For reference, the upcoming derivatives of Model 3 have been renamed BASE and ADJUSTED for the remainder of the exercise:

Table 10: Parameter Estimates for ADJUSTED Model to Predict SalePrice

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-35914	13030	-2.76	0.0059
MasVnrArea	1	48.83789	4.94304	9.88	<.0001
BsmtFinSF1	1	19.92054	2.04096	9.76	<.0001
TotalBsmtSF	1	30.42898	2.37631	12.81	<.0001
LowQualFinSF	1	-68.35790	17.78625	-3.84	0.0001
GrLivArea	1	68.28775	1.94378	35.13	<.0001
GarageArea	1	72.57830	4.55670	15.93	<.0001
WoodDeckSF	1	46.61637	6.43111	7.25	<.0001
EnclosedPorch	1	-48.48119	12.55833	-3.86	0.0001
ScreenPorch	1	54.55141	13.83156	3.94	<.0001
PoolArea	1	-118.07267	22.68900	-5.20	<.0001
buildtype_1Fam	1	35008	5512.33981	6.35	<.0001
buildtype_Duplex	1	-7165.82904	6886.39809	-1.04	0.2982
buildtype_Twnhs	1	24314	6852.63807	3.55	0.0004
buildtype_TwnhsE	1	43602	6113.83588	7.13	<.0001
salecond_Abnormal	0	0	.	.	.
salecond_Alloca	1	13369	14308	0.93	0.3502
salecond_Family	1	-18741	12967	-1.45	0.1485
salecond_Normal	1	-4411.81885	11629	-0.38	0.7044
salecond_Partial	1	33995	12027	2.83	0.0047

The interpretation of the model is difficult given the combination of continuous and indicator variables. Also note that buildtype_Duplex was set to 0 automatically by SAS since the variable is a linear combination of other variables shown.

We shift our attention to Table 11 below which is a summary of goodness of fit statistics comparing the BASE and ADJUSTED models. There were 5 more predictor variables in the ADJUSTED model. Of the 2930 observations in the sample there were 214 missing values in both models which is acceptable. The F Value for the ADJUSTED model is smaller than the BASE model but still statistically significant. R-Squared and Adjusted R-squared are slightly lower in the ADJUSTED model than the BASE model while the Mallow's Cp, BIC and AIC are all lower in the BASE model. It appears that the ADJUSTED model is slightly inferior to the BASE model despite attempting to conform to best modeling practices. Overall it is surprising the ADJUSTED model is not a better fit. In PART C, we take the ADJUSTED model and create a train/split of the data to cross validate the model.

Table 11: BASE and ADJUSTED Models

Model	BASE	ADJUSTED
Independent Variables	16	19
Missing Values	214	214
F-Value	510.09	453.32
Root MSE	39518	39524
Dependent Mean	183188	183188
Coeff Var	21.5721	21.57571
R-Square	0.7555	0.7516
Adj R-Sq	0.7538	0.7499
Cp	13.3012	19.0000
AIC	47975.7006	57514.8500
BIC	47978.015	57517.1100

PART C: Validation Framework

Cross Validation. As discussed in the assignment instructions, the trouble with using the entire dataset to fit a model is that we will not know if the entire model is transportable beyond the current dataset or is only idiosyncratic to the current data set. Comparing residuals for training and validation components of the data set is an important step in verifying the error performance of a system or model. In this section, we assess the predictive accuracy of our model using cross-validation and then contrast the difference between a statistical model and business model validation.

Using the ADJUSTED model from Part B, we create a train/test split of the data set for cross-validation; 70% of the data is used for training and the remaining 30% is used for testing. In order to "stress" the model, we randomly divide the data set using SAS to generate a uniform random variable with seed set to 123. This might be considered a superior approach to data splitting relative to more arbitrary methods.

Table 12: Summary of Parameter Estimates and Statistical Significance

	Parameter Estimates			Statistical Significance		
Variable	**Original Model**	**Training Model**	**Validation Model**	**Original Model**	**Training Model**	**Validation Model**
Intercept	-35914	-55834	-9735.730	0.0059	<.0001	0.7269
MasVnrArea	48.83789	44.77885	63.09736	<.0001	<.0001	<.0001
BsmtFinSF1	19.92054	26.87846	8.6767	<.0001	<.0001	0.0571
TotalBsmtSF	30.42898	37.21585	20.9133	<.0001	<.0001	<.0001
LowQualFinSF	-68.3579	-63.30566	-96.32659	0.0001	0.0002	0.0543
GrLivArea	68.28775	73.28098	59.63052	<.0001	<.0001	<.0001
GarageArea	72.5783	67.99165	80.31339	<.0001	<.0001	<.0001
WoodDeckSF	46.61637	40.24035	47.23145	<.0001	<.0001	0.0022
EnclosedPorch	-48.4812	-54.65973	-38.65568	0.0001	<.0001	0.1391
ScreenPorch	54.55141	46.17872	77.7921	<.0001	0.0014	0.0102
PoolArea	-118.073	-3.97425	-164.2115	<.0001	0.9011	<.0001
buildtype_1Fam	35008	35115	36115	<.0001	<.0001	0.0032
buildtype_Duplex	-7165.83	-11402	7817.7801	0.2982	0.1065	0.6271
buildtype_Twnhs	24314	28631	16081	0.0004	<.0001	0.3234
buildtype_TwnhsE	43602	42607	45358	<.0001	<.0001	0.001
salecond_Abnormal	0	0	0	.	.	.
salecond_Alloca	13369	17703	-1265.358	0.3502	0.2331	0.9691
salecond_Family	-18741	-5604.507	-37517	0.1485	0.6834	0.1715
salecond_Normal	-4411.82	1141.0415	-8975.617	0.7044	0.9261	0.717
salecond_Partial	33995	40823	25692	0.0047	0.0013	0.3184

Table 12 provides a summary of the model components in terms of parameter estimates and statistical significance. The magnitude of parameter estimates between the Original and Training models suggests they are relatively similar while the Validation Model coefficients are more extreme. The coefficient that differs notably between the Original and Training Models in statistical significance is PoolArea; it is significant in Original Model but not the Training Model. The majority of coefficients in both models are otherwise mostly statistically significant. Conversely, many of the coefficients in the Validation Model were not significant. This is potentially caused by the data splitting reducing the precision with which the coefficients are being estimated (Montgomery, Peck, and Vining, 2012). Overall, the Original and Training Models are relatively similar while the Validation Model could potentially produce notably different predictions. If this were the end of the analysis, I would recommend we reject the model and begin re-engineering which variables to include. This is beyond the scope of this exercise but worth noting. In general, if models are highly similar it would be a situation called overfitting, where a model too closely fits a limited set of data points. Conversely, if they were substantively different it would be a case of underfitting which is when a statistical model does not fit the data well enough.

Table 13: Fit Statistics

Statistic	Original Model	Training Model	Validation Model
F-Value	453.32	437.59	89.42
Root MSE	39524	34198	48028
MSE	1.56E+09	1.17E+09	2.31E+09
MAE	25010.55	24063.30	28138.92
R-Square	0.7516	0.807	0.6697
Adj R-Squared	0.7499	0.8051	0.6622
Predicted R-Square	0.73436	0.79794	0.58547

Table 13 provides summary statistics of the models in terms of fit. The Training Model's F-value is slightly lower than the Original Model. The R-Square and Adjusted R-squared of the Training Model are both superior to the Original. The Predicted R-squared, which indicates how well a regression model predicts responses for new observations, is highest in the Training Model. The Root MSE and MSE is lowest in the Training Model as well. The Validation Model is inferior by every metric. For reference, the MSE (Mean Squared Error) is a measure of how close a fitted line is to data points. It is the average of the squares of the difference between the actual observations and predicted data points. The lower the MSE, the better the model is fit to the data. Due to the square, however, potential outliers can have greater impact on the MSE. The RMSE (Root Mean Square Error) is simply the square root of the MSE. The advantage of the RMSE is that it is easier to interpret than the MSE as the units are standardized. The MAE (Mean Absolute Error) measures the average magnitude of the errors in a group of predictions. It is the average of the absolute difference between predicted and observed points where all have equal weight. The lower the MAE, the better. Again, the Training Model trumps the Original and Validation Models in terms of MSE and MAE.

Operational Validation. In this final portion of Part C, we use the MSE and MAE to determine predictive grades of each model. The prediction grade will tell us if the given model is within 10%, 15%, or greater than 15% of the actual value. The lower the difference (i.e. the lower the grade) the better. Table 14 below provides the grades by model and percent of the data set. The categorization of the metric is much more translatable into business policy. Surprisingly, the differences in residuals across the 3 models is similar which is encouraging. However, greater than 30% of each of the models' differences in residuals is greater than 15%. While this is not terrible it is less than ideal and the recommendation would be to continue refining and refitting the model to improve predictive accuracy.

Table 14: Distribution of Prediction Grades

Statistic	Original Model	Training Model	Validation
Grade 1 (<10%)	51.09%	49.83%	50.95%
Grade 2 (10% - 15%)	16.18%	17.85%	12.68%
Grade 3 (>15%)	32.73%	32.32%	36.36%

Reflections / Conclusions:

Building and validating regression models is a cornerstone of predictive analytics. For this exercise, the Ames, Iowa data set was used to construct regression models using dummy coded variables, select a subset of models using automated variable selection procedures and validate their predictive accuracy. If I were charged with analyzing this entire data set with the purpose of developing a model to present to the AMES REAL ESTATE ASSOCIATION, I would stress simplicity. While there is intellectual appeal in understanding as many factors as possible in what might be predictive of home prices, it is best to isolate those few variables that are easily interpretable and perhaps less statistically robust.

While several prospective variables were identified, certain elements of the data set are problematic. First, there are certain data integrity issues when dealing with publicly available data sets. The original proprietors of the data could have altered and scaled the data differently than what was initially measured. Additionally, having not collected these data ourselves we are forced to rely on the original proprietors in terms of collection techniques and accuracy. In terms of the data itself, several variables are potential candidates for transformation to achieve linearity. Specially, GarageArea, PoolArea and TotalBsmtSF could be transformed given the frequencies of zeros in the data. Also, it is unclear what timeframe the sample is taken across, forcing us to assume that these measurements are consistent across time.

Assignments #1 and #2 were valuable introductions to data manipulation using SAS, interpreting statistical outputs to identify potential predictor variables and initial model building. Assignment #3 expounded upon this by introducing variable transformations and outlier removal, two key tools in predictive modeling. This final assignment working with the Ames, Iowa data set introduced some of the more powerful automated tools available to modelers while also reinforcing the need for practitioners to continually scrutinize and reevaluate their processes and conclusions. Having interacted with many “data scientists” outside of the Northwestern MSPA program that don’t go beyond statistical summaries to validate models, these exercises overall also sheds light on the shortcomings of the current state of data driven decision-making in professional settings which are ripe for improvement.

Reference(s):

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

Code:

```
libname mydata '/scs/wtm926/' access=readonly;


Data temp1;

set mydata.ames_housing_data;







/* PART A Dummy Coding of Categorical Varibles */







/** Question 1**/



/***Analysis Potential Categorical Variable***/


proc sort data=temp1 outobs=10;;

run;


proc sql number outobs=10;

create table jj as

select distinct BldgType from temp1;
```

```
quit;



Data Part1;
        set temp1;
                Keep saleprice BldgType buildtype;
        if BldgType='1Fam' then buildtype=1;
        if BldgType='2fmCon' then buildtype=2;
        if BldgType='Duplex' then buildtype=3;
        if BldgType='Twnhs' then buildtype=4;
        if BldgType='TwnhsE' then buildtype=5;



proc sort data=Part1;
        by buildtype;

proc means data=Part1;
        by buildtype;
        var saleprice;
run;



/** Question 2**/
```

```
proc reg data=part1;
model saleprice = buildtype;
run;

/***Check that is working***/

proc freq data=part1;
tables BldgType buildtype;
run;



/** Question 3**/



proc freq;
        tables BldgType;
        run;



Data Part2;
        set temp1;
```

```
        keep saleprice BldgType buildtype_1Fam buildtype_Duplex buildtype_Twnhs
buildtype_TwnhsE;


    if BldgType  in ('1Fam' '2fmCon' 'Duplex' 'Twnhs' 'TwnhsE') then do;
        buildtype_1Fam = (BldgType eq '1Fam');
        buildtype_Duplex = (BldgType eq 'Duplex');
        buildtype_Twnhs = (BldgType eq 'Twnhs');
        buildtype_TwnhsE = (BldgType eq 'TwnhsE');
    end;


***** Leave out smallest 2fmCon;
Proc freq data=part2;
tables BldgType buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE;
run;


proc reg data=part2;
model saleprice =  buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE;
run;




/** Question 5**/




proc sql number outobs=10;
```

```
create table gg as
select distinct SaleCondition from temp1;
quit;



/*** Table 8 ***/
Data Part5;
        set temp1;
                Keep saleprice SaleCondition saleCond;
        if SaleCondition ='Abnorml' then saleCond=1;
        if SaleCondition ='AdjLand' then saleCond=2;
        if SaleCondition ='Alloca' then saleCond=3;
        if SaleCondition ='Family' then saleCond=4;
        if SaleCondition ='Normal' then saleCond=5;
        if SaleCondition ='Partial' then saleCond=6;

proc sort data=Part5;
        by saleCond;

proc means data=Part5;
        by saleCond;
        var saleprice;
run;

/**** Check to make sure works ****/
```

```
proc freq data=part5;
tables SaleCondition saleCond;
run;

/* PART B Automated Variable Selection */

/** Question 6**/

Data Part6;
	set temp1;
		keep saleprice LotFrontage LotArea MasVnrArea BsmtFinSF1 BsmtFinSF2 BsmtUnfSF TotalBsmtSF FirstFlrSF SecondFlrSF LowQualFinSF GrLivArea GarageArea WoodDeckSF OpenPorchSF EnclosedPorch ThreeSsnPorch ScreenPorch PoolArea BldgType buildtype_1Fam buildtype_2fmCon buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE SaleCondition salecond_Abnormal salecond_AdjLand salecond_Alloca salecond_Family salecond_Normal salecond_Partial;

	if BldgType in ('1Fam' '2fmCon' 'Duplex' 'Twnhs' 'TwnhsE') then do;
		buildtype_1Fam = (BldgType eq '1Fam');
```

```
        buildtype_Duplex = (BldgType eq 'Duplex');
        buildtype_Twnhs = (BldgType eq 'Twnhs');
        buildtype_TwnhsE = (BldgType eq 'TwnhsE');
end;

if BldgType='1Fam' then buildtype=1;
if BldgType='2fmCon' then buildtype=2;
if BldgType='Duplex' then buildtype=3;
if BldgType='Twnhs' then buildtype=4;
if BldgType='TwnhsE' then buildtype=5;

if SaleCondition in ('Abnormal' 'AdjLand' 'Alloca' 'Family' 'Normal' 'Partial') then do;
        salecond_Abnormal = (SaleCondition eq 'Abnormal');
        salecond_Alloca = (SaleCondition eq 'Alloca');
        salecond_Family = (SaleCondition eq 'Family');
        salecond_Normal = (SaleCondition eq 'Normal');
        salecond_Partial = (SaleCondition eq 'Partial');
end;

if SaleCondition='Abnormal' then salecond=1;
if SaleCondition='AdjLand' then salecond=2;
if SaleCondition='Alloca' then salecond=3;
if SaleCondition='Family' then salecond=4;
if SaleCondition='Normal' then salecond=5;
if SaleCondition='Partial' then salecond=6;
```

```
/*** Adjusted R Squared***/

proc reg data=part6 outest=reg_rsq_out;

model saleprice = LotFrontage LotArea MasVnrArea BsmtFinSF1 BsmtFinSF2 BsmtUnfSF TotalBsmtSF FirstFlrSF SecondFlrSF LowQualFinSF GrLivArea GarageArea WoodDeckSF OpenPorchSF EnclosedPorch ThreeSsnPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/

selection=adjrsq aic bic cp;


proc print data=reg_rsq_out;


proc reg data=part6;

model saleprice = LotFrontage LotArea MasVnrArea BsmtFinSF1 TotalBsmtSF FirstFlrSF SecondFlrSF GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Twnhs buildtype_TwnhsE salecond_Family salecond_Partial/

run;


/*** Mallow's Cp***/

proc reg data=part6 outest=reg_cp_out;

model saleprice = LotFrontage LotArea MasVnrArea BsmtFinSF1 BsmtFinSF2 BsmtUnfSF TotalBsmtSF FirstFlrSF SecondFlrSF LowQualFinSF GrLivArea GarageArea WoodDeckSF OpenPorchSF EnclosedPorch ThreeSsnPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/

selection= cp adjrsq aic bic cp best=10;


proc print data=reg_cp_out;
```

```
proc reg data=part6;

model saleprice = LotArea MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Twnhs buildtype_TwnhsE salecond_Family salecond_Partial/

run;


/*** AIC***/

proc reg data=part6;

model saleprice = LotFrontage LotArea MasVnrArea BsmtFinSF1 BsmtFinSF2 BsmtUnfSF TotalBsmtSF FirstFlrSF SecondFlrSF LowQualFinSF GrLivArea GarageArea WoodDeckSF OpenPorchSF EnclosedPorch ThreeSsnPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/

selection= cp adjrsq aic bic cp best=50;



proc reg data=part6;

model saleprice = LotArea MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Twnhs buildtype_TwnhsE salecond_Family salecond_Partial/

run;


/*** Forward***/

proc reg data=part6 outest=reg_forward_out;

model saleprice = LotFrontage LotArea MasVnrArea BsmtFinSF1 BsmtFinSF2 BsmtUnfSF TotalBsmtSF FirstFlrSF SecondFlrSF LowQualFinSF GrLivArea GarageArea WoodDeckSF OpenPorchSF EnclosedPorch ThreeSsnPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/

selection=forward adjrsq aic bic cp best=5;
```

```
outest=reg_forward_out


proc reg data=part6;

model saleprice = LotFrontage LotArea MasVnrArea BsmtFinSF1 BsmtUnfSF TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF OpenPorchSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Family salecond_Normal salecond_Partial/

run;


/***Backward***/

proc reg data=part6 outest=reg_backward_out;

model saleprice = LotFrontage LotArea MasVnrArea BsmtFinSF1 BsmtFinSF2 BsmtUnfSF TotalBsmtSF FirstFlrSF SecondFlrSF LowQualFinSF GrLivArea GarageArea WoodDeckSF OpenPorchSF EnclosedPorch ThreeSsnPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/

selection=backward adjrsq aic bic cp best=5;


outest=reg_backward_out


proc reg data=part6;

model saleprice = LotArea MasVnrArea BsmtFinSF1 BsmtFinSF2 BsmtUnfSF FirstFlrSF SecondFlrSF GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Twnhs buildtype_TwnhsE salecond_Family salecond_Partial /

run;


/***Stepwise***/

proc reg data=part6 outest=reg_stepwise_out;
```

```
model saleprice = LotFrontage LotArea MasVnrArea BsmtFinSF1 BsmtFinSF2 BsmtUnfSF  TotalBsmtSF FirstFlrSF SecondFlrSF LowQualFinSF GrLivArea GarageArea WoodDeckSF OpenPorchSF EnclosedPorch ThreeSsnPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/

selection= stepwise adjrsq aic bic cp best=5;


proc print data=reg_stepwise_out;


proc reg data=part6;

model saleprice = LotArea MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Twnhs buildtype_TwnhsE salecond_Family salecond_Partial /

run;




/****Final Model****/

/*Dummies added. LotArea removed*/


proc reg data=part6 outest=est;

model saleprice= MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/ cp bic aic;

run;

proc print data=est; run;
```

```
/* PART C Validation Framework */




/** Question 8**/



Data Part8_train;

set Part6;


* 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;




Data Part8_test;

        set part6;
```

```
* generate a uniform(0,1) random variable with seed set to 123;
        u = uniform(123);
        if (u > 0.70) then test= 0;
        else test = 1;
        if (test=0) then test_response=SalePrice;
        else test_response=.;
        run;



title 'Original';
proc reg data=part6;
model SalePrice = MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/  cp bic aic;
run;


title 'Training';
proc reg data=Part8_train;
model train_response = MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/  cp bic aic;
run;


title 'Validation';
proc reg data=part8_test;
```

```
model test_response = MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/  cp bic aic;

run;


/****Predicted R-squared****/;


title 'Original';

proc reg data=part6 outest=outest plots=none;

model SalePrice = MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/ rsquare press sse adjrsq;

run;

quit;


data outestPlus;

set outest;

_PRSQ_ = 1 - _PRESS_ * (1 - _RSQ_)/_SSE_;

label _PRSQ_ = "Predicted r-squared";

run;


proc print data=outestPlus label;

var _RSQ_ _ADJRSQ_ _PRSQ_;

run;
```

```
title 'Train';

proc reg data=Part8_train outest=outest plots=none;

model train_response = MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/ rsquare press sse adjrsq;

run;

quit;


data outestPlus;

set outest;

_PRSQ_ = 1 - _PRESS_ * (1 - _RSQ_)/_SSE_;

label _PRSQ_ = "Predicted r-squared";

run;


proc print data=outestPlus label;

var _RSQ_ _ADJRSQ_ _PRSQ_;

run;


title 'Validation';

proc reg data=part8_test outest=outest plots=none;

model test_response = MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/ rsquare press sse adjrsq;

run;

quit;
```

```
data outestPlus;
set outest;
_PRSQ_ = 1 - _PRESS_ * (1 - _RSQ_)/_SSE_;
label _PRSQ_ = "Predicted r-squared";
run;

proc print data=outestPlus label;
var _RSQ_ _ADJRSQ_ _PRSQ_;
run;



/** Question 9**/



/***Train***/;
proc reg data=Part6;
        model SalePrice = MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/
                selection=forward;
output out=part9_original predicted=yhat;

proc print data=part9_original(obs=5);
```

```
Data Part9b_original;
        set part9_original;
         mae = abs(yhat - SalePrice);

proc print data=Part9b_original(obs=5);

proc means data=Part9b_original;
         var mae;
        title 'MAE calculation - Original';

/***Train***/;
proc reg data=Part8_train;
        model train_response = MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/
                selection=forward;
output out=part9_train predicted=yhat;

proc print data=part9_train(obs=5);

Data Part9b_train;
        set part9_Train;
         mae = abs(yhat - train_response);
```

```
proc print data=part9b_train(obs=5);


proc means data=part9b_train;

        var mae;

        title 'MAE calculation - Train';



/***Validation***/;



proc reg data=part8_test;

        model test_response = MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/

                selection=forward;

output out=part9_test predicted=yhat;


proc print data=part9_test(obs=5);



Data Part9b_test;

        set part9_test;

        mae = abs(yhat - test_response);


proc print data=part9b_test(obs=5);
```

```
proc means data=part9b_test;
            var mae;
           title 'MAE calculation - Test';



/** Question 10**/



/***Original***/
proc reg data=part6;
model SalePrice = MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/
selection=forward;
output out=part10_original predicted=yhat;

title 'Original';

proc print data=part10_original(obs=10);
run;

Data part10_original_b;
set part10_original ;
```

```
if SalePrice=. then delete;


Length Prediction_Grade $7.;


pct_diff = abs((yhat - SalePrice ) / SalePrice);


if pct_diff LE .10 then Prediction_Grade = 'Grade 1';

else if pct_diff GT .10 and pct_diff LE .15 then Prediction_Grade= 'Grade 2';

else Prediction_Grade = 'Grade 3';


proc print data=part10_original_b(obs=10);

run;


proc freq data=part10_original_b;

tables prediction_grade;

run;



/***Training***/

proc reg data=Part8_train;

model train_response = MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/

selection=forward;

output out=part10 predicted=yhat;
```

```
title 'Train';

proc print data=part10 (obs=10);
run;

Data Part10b;
set part10;

if train_response =. then delete;

Length Prediction_Grade $7.;

pct_diff = abs((yhat - train_response) / train_response);

if pct_diff LE .10 then Prediction_Grade = 'Grade 1';
else if pct_diff GT .10 and pct_diff LE .15 then Prediction_Grade= 'Grade 2';
else Prediction_Grade = 'Grade 3';

proc print data=part10b (obs=10);
run;

proc freq data=part10b;
tables prediction_grade;
run;
```

```
/***Validation***/

proc reg data=part8_test;

model test_response= MasVnrArea BsmtFinSF1 TotalBsmtSF LowQualFinSF GrLivArea GarageArea WoodDeckSF EnclosedPorch ScreenPorch PoolArea buildtype_1Fam buildtype_Duplex buildtype_Twnhs buildtype_TwnhsE salecond_Abnormal salecond_Alloca salecond_Family salecond_Normal salecond_Partial/

selection=forward;

output out=part10_test predicted=yhat;


title 'Validation';


proc print data=part10_test (obs=10);

run;


Data Part10b_test;

set part10_test;


if test_response =. then delete;


Length Prediction_Grade $7.;


pct_diff = abs((yhat - test_response) / test_response);


if pct_diff LE .10 then Prediction_Grade = 'Grade 1';

else if pct_diff GT .10 and pct_diff LE .15 then Prediction_Grade= 'Grade 2';
```

```
else Prediction_Grade = 'Grade 3';


proc print data=Part10b_test(obs=10);

run;


proc freq data=Part10b_test;

tables prediction_grade;

run;
```