MWSUG 2016 - Paper AA23

A Demonstration of Various Models Used in a Key Driver Analysis

Steven LaLonde, Rochester Institute of Technology, Rochester, New York

ABSTRACT

A key driver analysis investigates the relationship between predictor variables and a response, such as customer satisfaction or net promoter score. The response is often measured on a five, seven, or ten-point scale, and collected using a survey. The predictors are generally other scaled questions, asked on the same survey, or demographics. Analyses often use multiple linear regression to fit the response as a function of the predictors, and some function of the regression coefficients as a measure of importance of individual predictors. This approach suffers from two major criticisms. The scaled response, especially if each point on the scale is individually labeled, may not be an interval scale, which would make the linear regression model invalid. Secondly, the predictors are generally correlated with one another, which can lead to counter-intuitive regression coefficients, even coefficients with the wrong sign! The first criticism can be alleviated by fitting an ordinal logistic model to the response, rather than the multiple linear regression. The second criticism is often addressed by fitting a more parsimonious model, using some form of variable selection. In this paper various approaches to the key driver analysis will be demonstrated using SAS® statistical procedures, and the advantages and disadvantages of each approach will be summarized.

INTRODUCTION

In this paper a number of different issues pertinent in a key driver analysis will be examined. In a key driver analysis the analyst first seeks to identify those variables that have the largest effect on the target variable (the importance). The current state of those variables is then measured in order to see how much room for improvement can be made (the performance). Finally the importance and performance is combined in a quadrant plot to identify those variables that are both importance, and have room for improvement (the key drivers). In this paper SAS will be used to illustrate the calculation of typical measures of importance and performance, and to display a quadrant plot.

SAMPLE DATA

The data used in this paper come from an “Order to Invoice” telephone survey conducted some time ago and has been used before in other research. Some of the scales of the data have been modified to make the data easier to use in this demonstration.

The demographic variables retained from the study include a respondent specific id, a business unit number, and the week in which the survey was conducted (see Table 1). This survey is a transactional survey conducted continuously over time. Nineteen weeks of data have been selected for this study (1,084 respondents).

SAS Variable Name	Description
ID	Respondent ID (unique identifier)
Unit	Business Unit (1 or 2)
Week	Week Survey Conducted (1 – 19)

Table 1. Demographic Variables

Questions from the original survey specific to “Order Placement” have been selected for the key driver analysis and are shown in Table 2. The overall satisfaction with order placement will be used as the response variable, and the other questions will be considered potential key drivers. The scales were end-anchored only, with a 1 representing “completely dissatisfied” and a 5 representing “completely satisfied”.

SAS Variable Name	**“Satisfaction with…” (1 – 5)**
OPlaceSat	the overall service on this order with respect to ORDER PLACEMENT.
EaseOfContact	the overall ease of contacting us to place the order.
TimeToConnect	the time it took to be connected to a customer service rep.
TimeToPlace	the time it took to actually place the order.
ProdAvailable	the availability of product desired.
RepKnowledge	the knowledge of the customer service rep.
RepCourteous	the courtesy, or professionalism, of the service rep.
RepResponsive	the responsiveness, of follow-through, of the service rep.
RepAbilitySolve	the ability of the service rep. to address my needs.

Table 2. Scaled Response Variables

As is typically the case, the distribution of responses on the 5-point scaled question is not at all normally distributed, or even symmetric. One would expect, in any high performing process, for there to be many more high ratings than low ratings. This is illustrated in Figure 1 with the histogram plots from PROC SGPANEL.

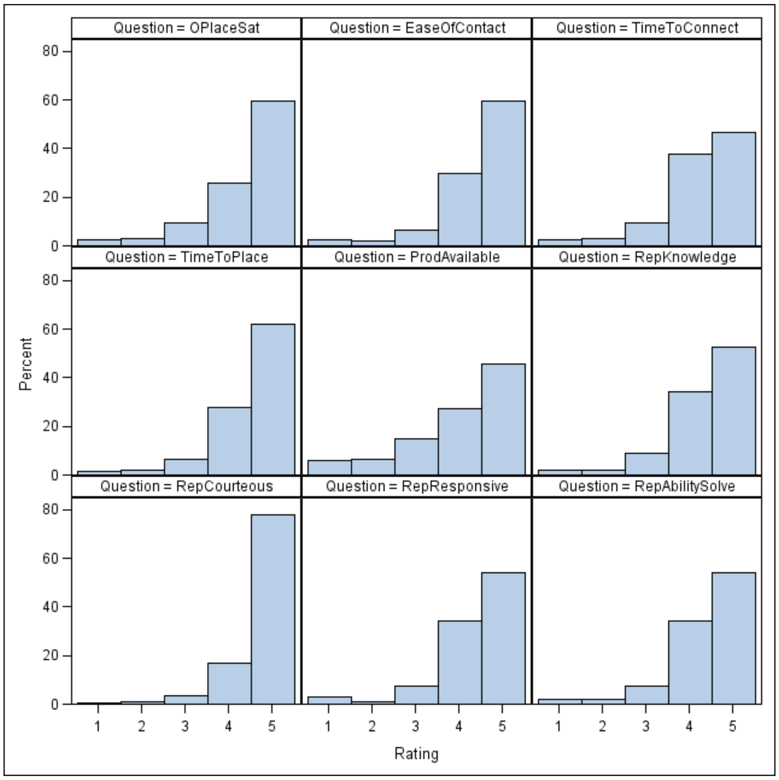

Figure 1. Histograms of Data

This plot was created by first transposing the data as illustrated in this SAS code:

```
proc transpose data=sortdata out=longdata
    (rename=(Col1=Rating)) Name = Question;
     var OPlaceSat EaseOfContact TimeToConnect TimeToPlace ProdAvailable
         RepKnowledge RepCourteous RepResponsive RepAbilitySolve;
     by Id;
run;

proc sgpanel data=longdata;
     panelby Question/columns=3 rows=3 sort=data;
     histogram  Rating/ nbins=5;
     label Question='Question' Rating='Rating';
run;
```

The fact that the data is not normally distributed, or even symmetric, does not in itself pose a problem for modeling. Since both the response, OplaceSat, and each of the potential key drivers follow the same shaped distribution, the residuals from modeling may well be okay. Transformations may be used to try to “normalize” the raw variables, but these transformations complicate the interpretation of the models and may not help to normalize the residuals from models (LaLonde, 2012).

MEASURES OF IMPORTANCE

First order of business in a key driver analysis is to come up with a measure of importance for each of the potential drivers. An important variable, in this context, would be a variable that has a quantifiable relationship with the response variable. In this case, when satisfaction with the driver variable is improved, a corresponding improvement in the satisfaction of the response variable would be expected.

TREATING THE RESPONSE AS AN INTERVAL SCALE

As mentioned before, the response, OPlaceSat, is an end-anchored scale, with a 1 representing “completely dissatisfied” and a 5 representing “completely satisfied”. It is common to assume with these types of scales that the scale is perceived by the respondent as an interval scale; that the perceptual distance between a 1 and a 2, and a 2 and a 3, and so on, are all the same. This is a more difficult assumption to make if each point on the scale is labeled, although it is still often done in practice.

The Basic Measure of Importance - Multiple Regression Coefficients

If we are willing to assume that the response, and in this case, the predictors, are measured on an interval scale, then multiple regression is the natural choice as a modeling procedure to determine the quantitative effect of each of the independent variables on the response. PROC REG can be used to produce the necessary parameter estimates shown in Table 3.

Obs	*Model*	*Dependent*	*Variable*	*DF*	*Estimate*	*StdErr*	*tValue*	*Probt*	*VarianceInflation*
1	MODEL1	OPlaceSat	EaseOfContact	1	0.11856	0.03793	3.13	0.0018	2.21222
2	MODEL1	OPlaceSat	TimeToConnect	1	-0.00893	0.03469	-0.26	0.7969	2.07182
3	MODEL1	OPlaceSat	TimeToPlace	1	0.34193	0.03612	9.47	<.0001	1.77065
4	MODEL1	OPlaceSat	ProdAvailable	1	0.12627	0.02239	5.64	<.0001	1.38758
5	MODEL1	OPlaceSat	RepKnowledge	1	-0.05860	0.03917	-1.50	0.1349	2.26428
6	MODEL1	OPlaceSat	RepCourteous	1	0.19879	0.04655	4.27	<.0001	1.82437
7	MODEL1	OPlaceSat	RepResponsive	1	0.12365	0.03766	3.28	0.0011	2.24154
8	MODEL1	OPlaceSat	RepAbilitySolve	1	0.07880	0.04114	1.92	0.0557	2.49545

Table 3. Multiple Regression Results

The Table shown is actually the results of running PROC PRINT on a dataset that was output from PROC REG. Creating a SAS dataset makes it easier to use the results later should we want to make a quadrant plot. Note also that the intercept has been removed from the dataset. The following SAS code was used to create the dataset.

```
ODS OUTPUT ParameterEstimates=mylib.MyParmEst1;

PROC REG data=mylib.rawdata Plots(UnPack);
   model OPlaceSat = EaseOfContact TimeToConnect TimeToPlace ProdAvailable
                     RepKnowledge RepCourteous RepResponsive RepAbilitySolve
                     /tol;
run;

data mylib.MyParmEst1;
set mylib.MyParmEst1;
where Variable ne 'Intercept';
run;
```

Since we are dealing with highly discretized data, the usual diagnostic plots are very unusual looking, and are often not shown to clients in practice. The plot of Observed by Predicted is especially interesting as it shows how the model tends to under predict the high values on the scale, and over predict the low values on the scale (See Figure 2).

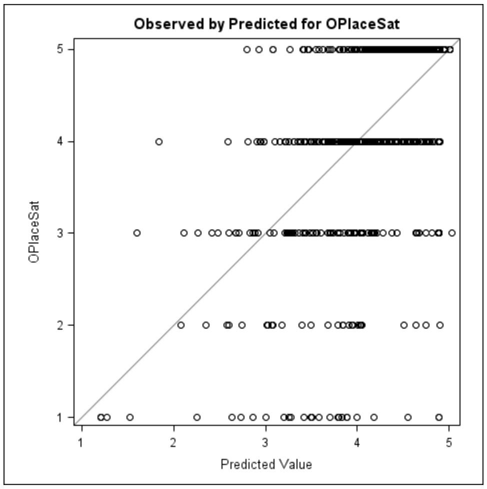

Figure 2. Plot of Observed by Predicted

It is not unusual for this type of behavior to be observed in a key driver analysis. Switching to a model that accounts for the discrete nature of the response will not compensate for this inability to predict the extremes well, as we will see later. It should be noted that this regression model has an r-square of only 40%, which is not atypical, but a little on the low side for my liking. ODS is especially useful to help create the dataset, and to break apart individual plots (see LaLonde, 2008).

Dealing with Multicollinearity

You will notice from the Table 3 that not all potential key drivers are statistically significant, and some even appear to have the wrong sign! No one would expect that to make people less satisfied on one of these drivers would result in higher overall satisfaction with order placement. It is possible that there is no relationship between these problematic independent variables and the response, and that the small negative coefficient happened by chance. It is also possible that multicollinearity between the independent variables is also creating a problem the estimation of the coefficients, as evidence by the variance inflation factors. Nevertheless, we probably don't want to show a negative relationship to our clients, as we are likely to get questions that will be difficult to answer, and distract the clients from making their decisions about where to put their improvement efforts.

Squared Bivariate Correlations

One way to deal with the multicollinearity problem is to assess the relationship between the response and each of the potential key drivers one at a time. This is often done using a squared bivariate correlation as shown in Table 4.

Obs	*Variable*	*OPlaceSat*	*POPlaceSat*	*OPlaceSat2*
1	EaseOfContact	0.47217	<.0001	0.22294
2	TimeToConnect	0.40310	<.0001	0.16249
3	TimeToPlace	0.53604	<.0001	0.28734
4	ProdAvailable	0.40411	<.0001	0.16331
5	RepKnowledge	0.40039	<.0001	0.16031
6	RepCourteous	0.43570	<.0001	0.18984
7	RepResponsive	0.46760	<.0001	0.21865
8	RepAbilitySolve	0.46041	<.0001	0.21198

Table 4. Squared Bivariate Correlation Results

In Table 4, OPlaceSat2 is the squared bivariate correlation created with the following SAS code:

```
ODS OUTPUT PearsonCorr=mylib.MyParmEst2;
PROC CORR data=mylib.rawdata;
     with EaseOfContact TimeToConnect TimeToPlace ProdAvailable
          RepKnowledge RepCourteous RepResponsive RepAbilitySolve;
     var  OPlaceSat;
run;

data mylib.MyParmEst2;
set mylib.MyParmEst2;
OPlaceSat2 = OPlaceSat*OPlaceSat;
```

Notice that in Table 4 all the potential key drivers show a positive correlation, and all are statistically significant (p < .05)! Using the correlations, or the squared correlations, eliminates the problem caused by the multicollinearity of the variables. However, it is not all sunshine and rainbows. Recall that the r-square in the multiple regression was around .40 or 40%. Each of these variables is showing a squared bivariate correlation, or r-square, around 20%. Collectively this implies that they would explain around 160% of the variability in the response! This method better reflects the individual contribution to improvement of the response, but drastically overestimates the collective effect of improving the satisfaction of the independent variables.

Variable Selection

Another way to deal with the apparent multicollinearity in the multiple regression model is to eliminate the redundancy with some sort of variable selection method. There are lots of methods for doing variable selection; in this paper stepwise selection will be illustrated.

In Table 5 the results of last step using stepwise selection using PROC REG is shown. The stepwise method results in a model that has five, rather than the original eight, independent variables. All of the variables show a positive regression coefficient, and all are statistically significant (p < .05). Note that this table has also had the intercept removed from it to facilitate use in the construction of the quadrant plot later.

Obs	*Model*	*Dependent*	*Step*	*Variable*	*Estimate*	*StdErr*	*TypeIISS*	*FValue*	*ProbF*
1	MODEL1	OPlaceSat	7	EaseOfContact	0.11388	0.03445	5.95220	10.93	0.0010
2	MODEL1	OPlaceSat	7	TimeToPlace	0.34240	0.03538	51.03639	93.68	<.0001
3	MODEL1	OPlaceSat	7	ProdAvailable	0.12949	0.02177	19.28052	35.39	<.0001
4	MODEL1	OPlaceSat	7	RepCourteous	0.20894	0.04340	12.62636	23.18	<.0001
5	MODEL1	OPlaceSat	7	RepResponsive	0.12959	0.03444	7.71242	14.16	0.0002

Table 5. Stepwise Selection Results

A slight modification of the previously run PROC REG produces the stepwise results:

```
ODS OUTPUT SelParmEst=mylib.MyParmEst3;
PROC REG data=mylib.rawdata;
  Model OPlaceSat = EaseOfContact TimeToConnect TimeToPlace ProdAvailable
                    RepKnowledge RepCourteous RepResponsive RepAbilitySolve
                    /selection=stepwise sls=.10;
run;

proc sql;
create table mylib.MyParmEst3 as
  select *
  from mylib.MyParmEst3
  having Step=max(Step) and Variable ne 'Intercept';
```

It should be noted that the sls option had to be changed from the default of .15 to .10 in order to get the model to eliminate the variable with the negative coefficient. Using the regression coefficients from Table 5 as a measure of importance, rather than the original ones from the full model in Table 3, eliminates the need to explain the negative coefficients, which as we saw from the bivariate correlations are clearly not a good indication of the importance of the deselected variables. The absence of the deselected variables, however, would seem to imply that those variables are not important, which may not be the case. It cannot be concluded that since the variables that did not enter the model that they are not important.

TREATING THE RESPONSE AS AN ORDINAL SCALE

The highly discrete nature of the five point scale on which the response is measured is a cause for concern for many modeling this type of data. Others are not comfortable making the assumption that the perceptual separation between points on the scale is equal-distant. Furthermore, in scales where every point on the scale is labeled, the points on the scale may not even be ordinal in nature! Since our data was collected with end-anchored scales, the scale will be assumed to be ordinal, that is, a 2 is greater than a 1, a 3 is greater than 2, and so on…

The Basic Measure of Importance – Beta Weights

An ordinal logistic regression is fit using PROC LOGISTIC in SAS, resulting in estimates of beta for each variable in the model (see Table 6). The model would also contain four intercepts (since the response has five levels), but they were eliminated as this table was created. You will also notice that the beta estimates generally have negative signs. These signs are normally reversed in the quadrant plot to show the higher negative effect of scoring poorly on the independent variables as higher importance.

Obs	*Variable*	*ClassVal0*	*DF*	*Estimate*	*StdErr*	*WaldChiSq*	*ProbChiSq*	*_ESTTYPE_*
1	EaseOfContact		1	-0.2250	0.1009	4.9706	0.0258	MLE
2	TimeToConnect		1	-0.0938	0.0947	0.9819	0.3217	MLE
3	TimeToPlace		1	-0.9455	0.0981	92.9430	<.0001	MLE
4	ProdAvailable		1	-0.3114	0.0619	25.3153	<.0001	MLE
5	RepKnowledge		1	0.0404	0.1074	0.1411	0.7072	MLE
6	RepCourteous		1	-0.4580	0.1245	13.5273	0.0002	MLE
7	RepResponsive		1	-0.2407	0.1004	5.7528	0.0165	MLE
8	RepAbilitySolve		1	-0.1801	0.1100	2.6813	0.1015	MLE

Table 6. Ordinal Logistic Regression Results

The syntax used to call PROC LOGISTIC is very similar to the PROC REG syntax:

```
ODS OUTPUT ParameterEstimates=mylib.MyParmEst4;
PROC LOGISTIC data=mylib.rawdata;
  Model OPlaceSat = EaseOfContact TimeToConnect TimeToPlace ProdAvailable
                    RepKnowledge RepCourteous RepResponsive RepAbilitySolve;
run;

data mylib.MyParmEst4;
set mylib.MyParmEst4;
where Variable ne 'Intercept';
run;
```

For a good general discussion of PROC LOGISTIC and logistic regression, see Flom (2010). For a more in depth illustration of the use of ordinal logistic regression in a key driver analysis, see Jeske, Callanan and Guo (2011).

Dealing with Multicollinearity

As we saw before with the multiple regression model, not all independent variables in Table 6 are statistically significant, and one variable even has the “wrong” sign! Fitting the ordinal logistic model has not gotten rid of the problem of multicollinearity among the independent variables.

Bivariate Measures of Association

The categorical analog to PROC CORR in SAS would be PROC FREQ. There are many choices for a measure of association; here I will illustrate the use of Gamma, which assumes that both variables are ordinal by nature.

Table 7 shows the bivariate measure of association from PROC FREQ.

Obs	*Statistic*	*Value*	*Variable*
1	Gamma	0.6751	EaseOfContact
2	Gamma	0.5863	TimeToConnect
3	Gamma	0.7637	TimeToPlace
4	Gamma	0.5426	ProdAvailable
5	Gamma	0.6088	RepKnowledge
6	Gamma	0.7084	RepCourteous
7	Gamma	0.6561	RepResponsive
8	Gamma	0.6384	RepAbilitySolve

Table 7. Bivariate Measures of Association Results

The SAS code used to produce the table is as follows:

```
ODS OUTPUT Measures=mylib.MyParmEst5;
PROC FREQ data=mylib.rawdata;
  Table  OPlaceSat *(EaseOfContact TimeToConnect TimeToPlace ProdAvailable
          RepKnowledge RepCourteous RepResponsive RepAbilitySolve)/measures;

run;

data mylib.MyParmEst5;
set mylib.MyParmEst5;
where Statistic='Gamma';
Variable = left(scan(Table,2,'*'));
keep Statistic Value Variable;
run;
```

PROC FREQ outputs many different measures of association and this SAS code could be modified easily to choose a different measure.   You will notice that these measures are all as would be expected, with the same two variables appearing as the key drivers.

Variable Selection

The Stepwise method can also be implemented in PROC LOGISTIC to eliminate redundancy in the ordinal logistic model. This reduces the model from eight variables to five variables, all of which are statistically significant (p<<.05).

Obs	*Variable*	*ClassVal0*	*DF*	*Estimate*	*StdErr*	*WaldChiSq*	*ProbChiSq*	*_ESTTYPE_*
1	EaseOfContact		1	-0.2768	0.0910	9.2519	0.0024	MLE
2	TimeToPlace		1	-0.9661	0.0962	100.7627	<.0001	MLE
3	ProdAvailable		1	-0.3356	0.0601	31.1380	<.0001	MLE
4	RepCourteous		1	-0.5184	0.1155	20.1320	<.0001	MLE
5	RepResponsive		1	-0.3014	0.0921	10.7135	0.0011	MLE

Table 8. Stepwise Ordinal Logistic Regression Results

The SAS code used to accomplish this task in PROC LOGISTIC is very similar to that which was used in PROC REG:

```
ODS OUTPUT ParameterEstimates=mylib.MyParmEst6;
PROC LOGISTIC data=mylib.rawdata;
  Model OPlaceSat = EaseOfContact TimeToConnect TimeToPlace ProdAvailable
                    RepKnowledge RepCourteous RepResponsive RepAbilitySolve
                    /selection=stepwise sls=.10;
run;

data mylib.MyParmEst6;
set mylib.MyParmEst6;
where Variable ne 'Intercept';
run;
```

As before, the same two variables appear to be most important. It should be noted that going to an ordinal logistic model does not entirely get rid of the problem we observed with regression, where we under predict the number of 1's and 5's observed. Recall here that the main objective in a key driver analysis is to determine which variables to "work on", not to accurately predict the answer a particular customer will give to the overall satisfaction with order placement question.

BUT WHAT ABOUT THE SCALE OF THE INDEPENDENT VARIABLES?

The scale of the independent variables has largely been ignored to this point. The independent variables are measure on the same scale as the response variable. In the multiple regression, bivariate correlations, and ordinal logistic methods we have treated the independent variables as if there were all measured on an interval scale. As we said before, since they are end-anchored, this is probably not an unreasonable assumption. But we chose to use a ordinal logistic model to allow for the response variable to be treated as ordinal, rather than interval. How would we make a similar adjustment to allow for the independent variables to be treated as ordinal or even nominal (categorical)?

Conceptually the adjustment is relatively easy, just fit indicator variables for each level of each independent variable, and fit models similar to those already described. This would get complicated pretty fast though, since our eight variables with five levels each would be replaced with 8*4=32 indicator variables. Interpreting those 32 measures of importance would be especially challenging.

A compromise would be to take the five point scale, and collapse it to a two point scale. For instance, you could represent scores from 1 to 3 as a zero, and 4 and 5 as a one, for each independent variable. This would get you back to eight independent variables, which do not require the interval assumption to be used in the models. Some market researchers start with a two point scale in the survey, rather than collapsing the scale later. If you do decide to collapse the scales of your independent variables, it is recommended that you try it a couple different ways, and see how robust your solution is to the choice you make.

MEASURES OF PERFORMANCE

In order to identify the key drivers we need to know which variables have the most room for improvement, that is, which ones are we currently performing lower on? Several measures of performance can be considered depending, again, on what we are willing to assume about the scale of measurement of the independent variables.

THE INDEPENDENT VARIABLES AS INTERVAL SCALE

If the independent variables are assumed to be measured on an interval scale, then the mean, or average, would be a logical choice for a measure of current performance.

Table 9 shows the averages for each of the independent variables based on the five point scale.

Obs	*Variable*	*Average*
1	EaseOfContact	4.422
2	TimeToConnect	4.234
3	TimeToPlace	4.473
4	ProdAvailable	4.004
5	RepKnowledge	4.344
6	RepCourteous	4.704
7	RepResponsive	4.360
8	RepAbilitySolve	4.367

Table 9. Means as a Measure of Performance

The averages are easily calculated using PROC MEANS, but a transposition of the output dataset is required to get the table in the proper format for the quadrant plot. Here is the code used to produce the output in Table 9:

```
PROC MEANS data=mylib.rawdata mean;
  Vars EaseOfContact TimeToConnect TimeToPlace ProdAvailable
       RepKnowledge RepCourteous RepResponsive RepAbilitySolve;
  output out=MyPerformEst1a(drop=_type_ _freq_)
         mean=EaseOfContact TimeToConnect TimeToPlace ProdAvailable
              RepKnowledge RepCourteous RepResponsive RepAbilitySolve;
run;

proc transpose data=MyPerformEst1a
               out=mylib.MyPerformEst1 (rename=(col1=Average))
               name=Variable;
run;
```

THE INDEPENDENT VARIABLES AS ORDINAL SCALE

If you are not willing to assume that the five point scale is an interval scale, then the averages would not be a proper summary statistic to use. In the section on measuring importance, it was suggested that you could treat each separate level of each independent variable as a potential key driver (using indicator variables). The analog here would be to calculate the percentage of times each scale value for each independent variable occurs. As before, this quickly becomes unwieldy, and is not done in practice.

The previously mentioned compromise, that is, to reduce the scale of each independent variable to two levels is often done in practice as a precursor to a measure of performance. This is generally referred to as Top Box (or Top 2 Box) performance. The performance measure is just the proportion of responses that fall into the top box, or highest scale value.

The results for our data are shown in Table 10.

Obs	*Variable*	*TopBox*
1	EaseOfContact	0.595018450
2	TimeToConnect	0.468634686
3	TimeToPlace	0.621771217
4	ProdAvailable	0.456642066
5	RepKnowledge	0.528597786
6	RepCourteous	0.779520295
7	RepResponsive	0.543357933
8	RepAbilitySolve	0.542435424

Table 10. Top Box as a Measure of Performance

The SAS code used to reduce the scale to a 0-1 scale in a DATA STEP is shown here:

```
data temp1;
  set mylib.rawdata;
  array bv (ii) EaseOfContact TimeToConnect TimeToPlace ProdAvailable
                RepKnowledge RepCourteous RepResponsive RepAbilitySolve;
  do over bv; if bv eq 5 then bv = 1; else bv = 0; end;
run;
```

A simple change to the IF-THEN statement could be made to produce a TOP 2 Box summary.

The SAS code used to calculate the proportions (using PROC MEANS) and reformat the data into the proper table (using PROC TRANSPOSE) is shown here:

```
PROC MEANS data=temp1 mean;
  Vars EaseOfContact TimeToConnect TimeToPlace ProdAvailable
       RepKnowledge RepCourteous RepResponsive RepAbilitySolve;
  output out=MyPerformEst2a(drop=_type_ _freq_)
         mean=EaseOfContact TimeToConnect TimeToPlace ProdAvailable
              RepKnowledge RepCourteous RepResponsive RepAbilitySolve;
run;

proc transpose data=MyPerformEst2a
               out=mylib.MyPerformEst2 (rename=(col1=TopBox))
               name=Variable;
run;
```

The Top Box summary is commonly preferred by managers to the average because of its simplicity of interpretation.

DISPLAYING FINAL RESULTS WITH A QUADRANT PLOT

The last step in the Key Driver Analysis is to create a Quadrant Plot. For this step we need to choose one among our measures of importance, and one among our measures of performance, and merge those into one dataset. For this illustration the bivariate correlations were chosen as the measure of importance, and the average performance chosen as the measure of performance. Table 11 shows the information merged into one SAS dataset:

Obs	*Importance*	*Performance*	*Variable*
1	22.3	4.4	EaseOfContact
2	16.2	4.2	TimeToConnect
3	28.7	4.5	TimeToPlace
4	16.3	4.0	ProdAvailable
5	16.0	4.3	RepKnowledge
6	19.0	4.7	RepCourteous
7	21.9	4.4	RepResponsive
8	21.2	4.4	RepAbilitySolve

Table 11. Selected Performance and Importance Measures

The bivariate correlations were multiplied by 100 for convenience. In addition to creating the required SAS dataset, PROC SQL was used to create two SAS MACRO variables, &avgrel and &avgper; to store the average importance and average performance over the range of independent variables. These macro variables will be used to place the crosshairs on the quadrant plot. The SAS code used to merge the data together and create the macro variables is shown here:

```
proc sql noprint;
select avg(100*a.OplaceSat2), avg(b.Average)
       into :avgrel, :avgper
 from mylib.MyParmEst2 as a, mylib.MyPerformEst1 as b
 where a.Variable = b.Variable;
create table quaddata as
select 100*a.OplaceSat2 as Importance, b.Average as Performance, a.Variable
 from mylib.MyParmEst2 as a, mylib.MyPerformEst1 as b
 where a.Variable = b.Variable;
quit;
run;
```

The quadrant plot is essentially just a scatterplot of the two variables, importance and performance, with one point for each potential key driver. In this paper I have chosen to put the importance on the vertical axis, and performance on the horizontal axis, but has been done both ways. Placing the crosshairs on the plot that divide the plot into four quadrants is somewhat arbitrary, and can be done many ways. Here I have chosen to place the crosshairs at the intersection of the average importance and average performance. Depending on the resources available for process improvement, these lines are often adjusted up or down, left or right…

The quadrants on the plot are generally labeled with the “appropriate” action to be taken. There are many variations on what these labels should be, in this case I have used: Critical, Leverage, Monitor, and Maintain.

The axis values are sometimes shown, sometimes not. In this case I have chosen to display the default axis values. Figure 3 shows the final quadrant plot.

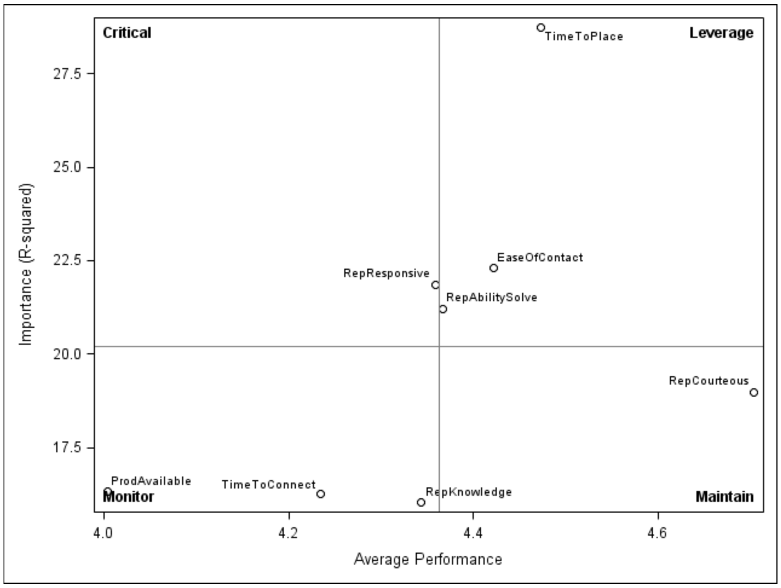

Figure 3. Quadrant Plot

The PROC SGPLOT SAS code is shown here for creating the scatterplot, labeling the points, adding the crosshairs as reference lines, an labeling each quadrant with an inset statement:

```
proc sgplot data=quaddata;
  scatter y=Importance x=Performance / datalabel= Variable;
  refline &avgrel/axis=y;
  refline &avgper/axis=x;
  inset "Critical" / position=topleft Textattrs=(weight=bold);
  inset "Leverage" / position=topright Textattrs=(weight=bold);
  inset "Monitor" / position=bottomleft Textattrs=(weight=bold);
  inset "Maintain" / position=bottomright Textattrs=(weight=bold);
  label Performance="Average Performance"
        Importance="Importance (R-squared)";
run;
```

If one of the techniques involving variable selection was used in calculating importance then there will be fewer variables on the plot than you started with. Care must be taken in explaining to the client why those variables are not shown, and to clarify that just because they are not there, that it does not mean they are not related to the response variable.

USE OF DEMOGRAPHIC VARIABLES

In the first section of this paper the data was described as contain three demographic variables: ID, Unit, and Week. The ID variable uniquely identifies each survey respondent, and was used in SAS code in PROC TRANSPOSE earlier.

The second demographic variable, Unit, identifies which, of two, business units the respondent place an order with. Up to this point the data has been analyzed all together, basically assuming that the measures of importance, and measures of performance, were not substantially different between the two units (or at least we were not interested in the differences and were happy with the amount of weight each unit was receiving in the analysis). These are testable hypotheses, but beyond the scope of this paper (and beyond the interest of most market research consultants and their clients). However, it is not uncommon to want to control for possible differences between the units by including an indicator variable in the regression and/or logistic models to account for intercept differences. You could also include interaction terms between the unit indicator variable and each of the potential key drivers, allowing for different importance levels for each driver in each unit. Interpreting the coefficients in these models get pretty tricky, very fast, so you are probably better off just running the previously described analyses separately for each unit if you are interested in each unit allowing for different importance and performance.

It is common to assume that respondents would assign the same level of importance to each independent variable, regardless of the business unit they were ordering product from. However, it would not be unusual for the representatives answering the phone to have different levels of knowledge about products from each business unit, or that different business units might have different policies for what product is immediately available, etc… Therefore, it might well be important to calculate measures of performance separately for each business unit. Table 12 shows the results of the calculations. Note also that a new SAS variable named Variable2 has been created with a number 1 or 2 has been appended to the original SAS variable named Variable identifying the independent variable to show which unit that mean represents. The original variable is retained for purposes of merging.

Obs	*Variable*	*Unit*	*Average*	*Variable2*
1	EaseOfContact	1	4.495	EaseOfContact1
2	TimeToConnect	1	4.314	TimeToConnect1
3	TimeToPlace	1	4.504	TimeToPlace1
4	ProdAvailable	1	4.161	ProdAvailable1
5	RepKnowledge	1	4.445	RepKnowledge1
6	RepCourteous	1	4.684	RepCourteous1
7	RepResponsive	1	4.421	RepResponsive1
8	RepAbilitySolve	1	4.430	RepAbilitySolve1
9	EaseOfContact	2	4.344	EaseOfContact2
10	TimeToConnect	2	4.149	TimeToConnect2
11	TimeToPlace	2	4.441	TimeToPlace2
12	ProdAvailable	2	3.836	ProdAvailable2
13	RepKnowledge	2	4.237	RepKnowledge2
14	RepCourteous	2	4.725	RepCourteous2
15	RepResponsive	2	4.294	RepResponsive2
16	RepAbilitySolve	2	4.300	RepAbilitySolve2

Table 12. Performance Calculated Separately for each Unit

A slight modification of previously run SAS code accomplishes this task:

```
PROC MEANS data=mylib.rawdata mean;
  Vars EaseOfContact TimeToConnect TimeToPlace ProdAvailable
       RepKnowledge RepCourteous RepResponsive RepAbilitySolve;
  Class Unit;
  output out=MyPerformEst2a(drop=_type_ _freq_)
         mean=EaseOfContact TimeToConnect TimeToPlace ProdAvailable
              RepKnowledge RepCourteous RepResponsive RepAbilitySolve;
run;
proc transpose data=MyPerformEst2a
               out=mylib.MyPerformEst2 (rename=(col1=Average))
               name=Variable;
By Unit;
where Unit ne .;
run;
```

```
data mylib.MyPerformEst2;
length Variable $20.;
set mylib.MyPerformEst2;
Variable2 = compress(Variable||Unit);
run;
```

The quadrant plot is shown here, with two points for each variable, one for Unit 1 and one for Unit 2. Notice that they are on the same horizontal line since the same measure of importance is assigned, regardless of the Unit.

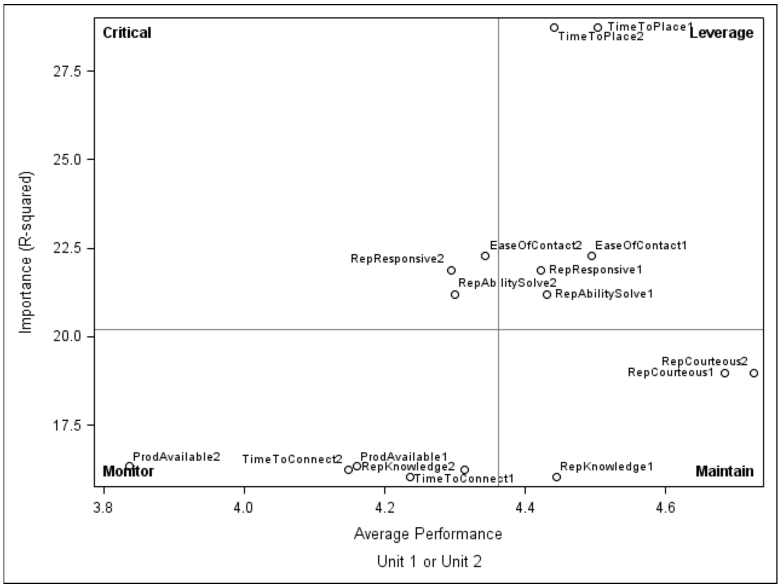

Figure 4. Quadrant Plot Showing Unit Differences in Performance

The SAS code need only be modified a little to accomplish this task:

```
proc sql noprint;
select avg(100*a.OplaceSat2), avg(b.Average)
       into :avgrel, :avgper
 from mylib.MyParmEst2 as a, mylib.MyPerformEst2 as b
 where a.Variable = b.Variable;
create table quaddata as
select 100*a.OplaceSat2 as Importance, b.Average as Performance, b.Variable2
 from mylib.MyParmEst2 as a, mylib.MyPerformEst2 as b
 where a.Variable = b.Variable;
quit; run;

proc sgplot data=quaddata;
  scatter y=Importance x=Performance / datalabel= Variable2;
  refline &avgrel/axis=y;
  refline &avgper/axis=x;
  inset "Critical" / position=topleft Textattrs=(weight=bold);
  inset "Leverage" / position=topright Textattrs=(weight=bold);
  inset "Monitor" / position=bottomleft Textattrs=(weight=bold);
  inset "Maintain" / position=bottomright Textattrs=(weight=bold);
  label Performance="Average Performance"
        Importance="Importance (R-squared)";
Footnote2 "Unit 1 or Unit 2";
run;
```

It appears that Unit 1 is performing at a higher average level than Unit 2 (with the exception of RepCourteous).

A second demographic variable was included in the dataset, that is, the week in which the survey was conducted. In this data the survey was conducted over a 19 week period. In practice we would use some sort of control chart to plot the data over time, but that is the topic of another paper! In the context of a Key Driver Analysis, and the associated quadrant plot, it would be useful to see whether current performance is better, or worse, than past performance. Are we improving or not? For purposes of illustration, we'll define current performance as the performance from Week 15 to Week 19, and the past performance as Week 1 through Week 14. Figure 5 shows a comparison of performance in Weeks 1-14 with Weeks 15 to 19. From this plot it is easier to see that we have made recent improvements on several of the variables (this is confirmed with the control charts).

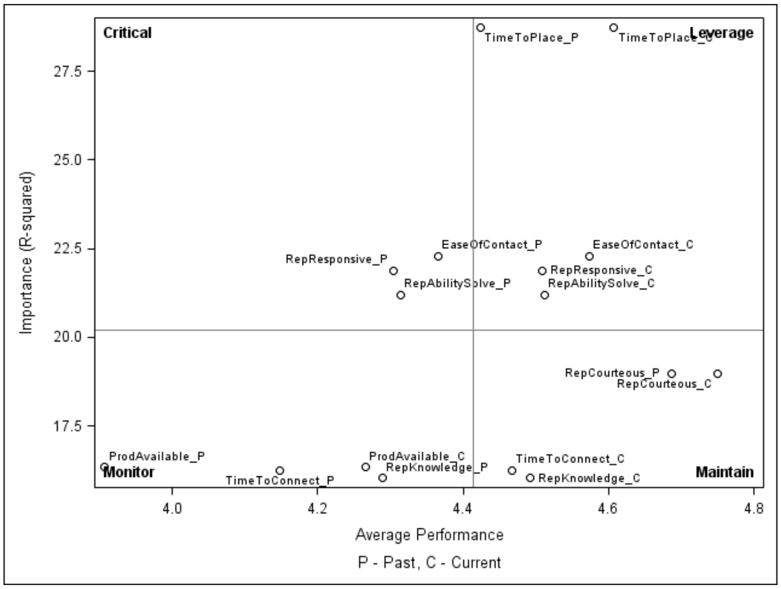

Figure 5. Quadrant Plot Showing Time Dimension of Performance

The SAS code for creating the required data and producing the quadrant plot is very similar to the code used for the unit quadplot and will not be reproduced here (left as an exercise to the reader!).

OTHER MEASURES OF IMPORTANCE FROM THE LITERATURE

As we have seen, a measure of importance of a variable in a regression model (or ordinal logistic regression model) is a relatively complicated idea when multicollinearity exists. The approaches that I have taken here are relatively simple, and very common in practice. However, others have taken these ideas to a much higher level. Kruskal (1987) describes a method for 'averaging' partial r-squares from multiple models to arrive a 'better' measure of the importance of a variable in a multiple regression model. Budescu (1993) extends these ideas to other measures of importance in multiple regression. Menard (2004) applied Kruskal's ideas in the context of logistic regression.

The difficulty with these methods is that they are not generally available in software, at least not SAS software. I have been able to reproduce Kruskal's measure of importance using SAS, but that SAS program is not ready to be distributed. I would like to write code to reproduce some of Menard's ideas, but I'm still looking for the time to do it…

Alternatively, although I have not used it, there is a package in R called relaimpo that appears to produce measures suggested by Budescu (Groemping, 2006).

CONCLUSION

This paper has demonstrated various calculations of measures of importance in multiple regression and ordinal logistic regression models. Issues regarding the scale of the response, and the multicollinearity typically present in the independent variables, have been discussed. Two alternatives for calculating measures of performance were also demonstrated. Quadrant plots were demonstrated using selected measures of importance and performance. Lastly, variations on the quadrant plots to illustrate different levels of categorical variables, and time dimensionality were illustrated.

REFERENCES

Budescu, David V. 1993. Dominance Analysis: A New Approach to the Problem of Relative Importance of Predictors in Multiple Regression. *Psychological Bulletin*, Vol. 114, No. 3, 542-551.

Jeske, Daniel R., Callanan, Terrance P., and Guo, Li (2011). Identification of Key Drivers of Net Promoter Score Using a Statistical Classification Model, Efficient Decision Support Systems - Practice and Challenges From Current to Future, Prof. Chiang Jao (Ed.), ISBN: 978-953-307-326-2, InTech, Available from: http://www.intechopen.com/books/efficient-decision-support-systems-practice-and-challenges-from-current-tofuture/identification-of-key-drivers-of-net-promoter-score-using-a-statistical-classification-model.

Flom, Peter L. 2010. “Multinomial and ordinal logistic regression using PROC LOGISTIC” *Proceeding of the Northeast SAS User’s Group (NESUG)*, Available from: http://www.lexjansen.com.

Groemping, Ulrike. 2006. Relative Importance for Linear Regression in R: The Package relaimpo, *Journal of Statistical Software*, 17, 1, pp. 1-27.

Kruskal, William. 1987. Relative Importance by Averaging Over Orderings, *The American Statistician*, Vol. 41, No. 1 (Feb., 1987), pp. 6-10.

LaLonde, Steven M. 2008. ODS for the Professional Statistician, *Proceedings of the SAS Global Foru*m, San Antonio, Texas, Available from: http://www.lexjansen.com.

LaLonde, Steven M. 2012. Transforming Variables for Normality and Linearity - When, How, Why and Why Not's, *Proceedings of the SAS Global Forum*, Orlando, Florida, Available from: http://www.lexjansen.com.

Menard, Scott. 2004. Six Approaches to Calculating Standardized Logistic Regression Coefficients, *The American Statistician*, 58:3, 218-223.

ACKNOWLEDGMENTS

I would like acknowledge Carestream Health, Inc. for giving me space and so many other resources during my sabbatical from Rochester Institute of Technology to explore some of these ideas. I would specifically like to thank Terrance Callanan and Brian Cashimere for their insightful observations and discussions during my time at Carestream Health, Inc.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Steven LaLonde
Carey Hall, 14-2215
School of Mathematical Sciences
Rochester Institute of Technology
Rochester, NY 14623
(585) 475-5854
Steven.LaLonde@rit.edu

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.