ABSTRACT

To obtain a parsimonious set of variables that efficiently predicts the response of interest, many people delve straight in and start performing an automated model selection method such as forward selection. This paper covers the fundamentals of what the first step really should be – getting to know your data – and how SAS® can help you do this! This knowledge is then utilized to build multiple linear regression models. Direct comparisons between PROC REG and PROC GLMSELECT are made.

INTRODUCTION

In this paper we guide you in how you can get to know your data before proceeding to build a multiple linear regression model and in doing so we give a few examples of procedures that are useful to use. We compare and highlight the differences between the two SAS procedures, PROC REG and PROC GLMSELECT, which can be used to build a multiple linear regression model.

This paper does not cover multiple linear regression model assumptions or how to assess the adequacy of the model and considerations that are needed when the model does not fit well. It assumes some knowledge of multiple linear regression (MLR) and does not cover the statistical theory behind the different procedures and options that are overviewed in this paper. The information on all procedures is based on SAS 9.2. Where examples of SAS code are given, uppercase indicates SAS specified syntax and lowercase italics indicates user supplied code.

STEP 1: GET TO KNOW YOUR DATA!

It is extremely important and good practice before building a multiple linear regression model, or any type of model for that matter, you know your data. It enables you to anticipate the important features that you may need to include in your model and will give you an indication of what variables you expect to find important. It also helps you know whether there are non-linear patterns, interactions or correlations among the independent predictors - this could suggest collinearity. It helps you understand what the relationship is between the dependent variable and each of the independent variables and therefore gives you the ability to assess the accuracy of the model. At the end of the day, you need to know your data, do not leave it up to the computer.

- Talk to the experts, researchers, physicians, clients. Find out what their knowledge and expectation of how the variables are related and affect each other. Use their knowledge and prior research or studies to help guide the analysis.
- Consider the number of variables you have in relationship to the number of observations.
- Identify the proportion of missing values for each of the variables to know if you have enough data and whether the data is representative or there are inherent sampling biases.
- Compute descriptive statistics for the dependent and independent variables. These should include, but is not exclusive to, the mean, median, standard deviation, skewness of the distribution and the range of the values for continuous variable; frequency counts for categorical variables.
- Examine possible interactions, confounders or collinearity by:
  - Carrying out cross tabulations on the independent categorical variables
  - Look at the correlation between the independent continuous variables.
  - Examine summary statistics of the independent continuous variables by each category of an independent categorical variable to see if the distribution varies by category.
- Graphical displays such as scatter plots, histograms, and box plots are extremely helpful to look at the distribution of both dependent and independent variables. They allow you to look for skewness, clumping of
observations around particular values, possible outliers, extreme values, unexpected values etc. It is also useful to highlight different categorical levels within these plots. Remember though, even if the distribution of the variables are skewed you can still at this point go ahead and fit the model as it is the residuals after the model fitting that need to be normally distributed.

Since there are multiple procedures with many options that can help you with all of the above, this paper only briefly highlights a few commonly used procedures and options within them. Together they will enable the users to gain a basic understanding of their data.

**PROC FREQ**

PROC FREQ is useful for investigating relationships between categorical variables for possible interactions and the distribution of the data, including missing values.

An example of code:

```plaintext
ODS GRAPHICS ON;
PROC FREQ DATA = dset;
   TABLE cat1 cat2 cat1*cat2
      /NOPERCENT NOROW NOCOL
      MISSING
      PLOTS=FREQPLOT (TYPE=DOT);
RUN;
ODS GRAPHICS OFF;
```

PROC FREQ lists the frequency tables for categorical variables `cat1`, `cat2` and cross-tabulation of `cat1` and `cat2` in 3 different tables. By default the percentage of all observations, row percentages and column percentages are displayed in the frequency table; NOPERCENT, NOROW and NOCOL, respectively, can be used to suppress these. The MISSING option displays missing values within each table and they are treated as a valid category level within the frequency table. In PROC FREQ, you must enable ODS graphics using the ‘ODS GRAPHICS ON’ statement before requesting plots. The PLOTS option, in the above example, produces a frequency ‘DOT’ plot for each table. By default, a bar chart is produced if TYPE= option is missing. If you do not specify the PLOTS= option but have enabled ODS graphics, then PROC FREQ produces all plots associated with the analyses you request in the current TABLE statement.

**PROC TABULATE**

PROC TABULATE is similar to PROC FREQ and computes many of the same descriptive statistics but displays them in a tabular format. However, this procedure provides flexibility in classifying the values of variables and establishing hierarchical relationships between variables.

An example of code:

```plaintext
PROC TABULATE DATA=dset;
   CLASS cat1 cat2 cat3 / EXCLUSIVE PRELOADFMT;
   TABLE cat1, cat2*cat3 / PRINTMISS;
   FORMAT cat1 cat1fmt. cat2 cat2fmt. cat3 cat3fmt.
RUN;
```

This example produces a cross-tabulated table where the category levels in `cat1` are represented as rows, and the category levels of `cat3` within each category level of `cat2` are represented as columns in a hierarchical way. PRELOADFMT specified in the CLASS statement preloads the class variable formats and then using the EXCLUSIVE option excludes the values of class variables that are not found in the preloaded range of user-defined formats from the output. In addition, specifying PRELOADFMT in the CLASS statement and then using PRINTMISS in the TABLE statement will display all possible combinations of the formatted class variable values even for zero frequency. The PRINTMISS option treats the missing values as a valid category level in the table.
PROC UNIVARIATE

PROC UNIVARIATE provides a large number of descriptive statistics to help you examine the continuous variables. Although several statements and many options are available, the knowledge of just a few will allow you to perform a thorough examination of your data, including graphics output.

An example of code:

```sas
ODS GRAPHICS ON;
PROC UNIVARIATE DATA = dset;
   CLASS categoryvble;
   HISTOGRAM contvble / NORMAL;
   PROBPLOT contvble / NORMAL (MU=EST SIGMA=EST);
RUN;
```

By default, PROC UNIVARIATE gives you: basic statistical measures such as the mean, median and variance; moments such as skewness (a measure of asymmetry), kurtosis (a measure of flatness or peakness) and the coefficient of variation; percentiles; extreme observations showing the 5 lowest and 5 highest values – unusually large gaps between some values may indicate an outlier.

In order to examine possible interactions, confounding or collinearity between continuous and categorical variables the CLASS statement can be used.

If you have a license for SAS/GRAPH then you can get high-resolution graphics. However, ODS graphics can be initiated using ODS GRAPHICS ON, although, it is experimental in SAS 9.2 for use with this procedure.

In the above example, the HISTOGRAM statement with the NORMAL option displays a fitted normal curve on the histogram and would produce a histogram for the data in each category of the category variable specified in the CLASS statement. It also requests a NORMAL probability plot using the PROBPLOT statement and asks for the mean (MU) and standard deviation (SIGMA) to be shown on the plot. You are able to specify other theoretical distributions. You can also request percentile/quantile plots and cumulative distribution function plots.

PROC CORR

PROC CORR is the most often used SAS procedure to examine the correlation between variables. This can help towards identifying collinearity. Collinearity can appear as a very high correlation among 2 variables, but it could also be when several variables add up to something that is very close to a constant value. Collinearity can cause a loss in power and make interpretation difficult since parameter estimates measure the effect of a change on the dependent variable for one unit of measure, assuming all other variables are held constant. If one variable changes depending on the value of another variable i.e. high correlation, it is difficult to assume it remains constant for all values of the other variable (a violation of the independence assumption for linear regression).

One thing you do need to remember is that correlations give an indication of how strong a linear relationship there is. There are conditions when a correlation coefficient may appear to show a strong linear relationship but when the two variables are plotted against each other the relationship is clearly not a straight line. To avoid misinterpretation and missing those non-linear relationships between variables, a good practice is to accompany the calculation with a graph.

An example of code:

```sas
ODS GRAPHICS ON;
PROC CORR DATA = dset
   PEARSON SPEARMAN Hoeffding
   PLOTS = MATRIX(HISTOGRAM)
   NOMISS;
   BY vble_list;
   VAR vble_list;
RUN;
ODS GRAPHICS OFF;
```
PROC CORR can produce some graphical plots. However, you do need to specify ‘ODS GRAPHICS ON’ to obtain them. It does not support a CLASS statement so you will have to sort your data and use the BY statement if you want to obtain correlations for different levels of a categorical variable.

The above example requests Pearson correlation coefficients (a measure of linear relationship between two variables), spearman rank-order correlations (uses the rank of the data values) and Hoeffding’s measure of dependence (detects general departures from independence) for all pair wise associations of the variables listed in the VAR statement.

When carrying out a multiple linear regression model, if you use SAS’s automated model selection methods such as forward, backward and stepwise, it only includes those observations that have completed data for all independent variables that are considered in the model. The NOMISS option in PROC CORR allows you to look at correlations when just these complete observations are included.

**PROC SGSCATTER**

New in 9.2, the SGSCATTER procedure, just as the name suggests, is very useful for obtaining scatter plots of your variables. This helps you identify non-linear relationships, outliers, whether there are any groupings of observations around particular values and the range of values the variables take; it gives an indication of whether you may need to transform your variables. If you need to consider the natural logarithm transformation, it will enable you to see if there are any values that are zero.

PROC SGSCATTER contains three statements that you can use to create a panelled graph of scatter plots. These are:

1. **PLOT** – produces a panelled graph of scatter plots where each graph cell has its own independent set of axes.
2. **COMPARE** – used to create a shared axis panel. The list of X and Y variables are crossed to create a cell. All cells in the row share the same row axis range and all cells in the column share the same column axis range.
3. **MATRIX** – creates a scatter matrix. Each variable specified is graphed against each other.

An example of code:

```
PROC SGSCATTER DATA = dset;
   COMPARE X=(age waittime) Y=(tumorsize exposuretime)
       / REG ELLIPSE=(ALPHA=0.05 TYPE=MEAN);
RUN;
```

The above code uses the COMPARE statement to produce a shared axis panel where age and wait time are each plotted against tumor size and exposure time using the same Y axis range. The REG option fits a linear regression line for each variable pair and calculates the 95% confidence ELLIPSEs for the MEAN (see Figure 1). If TYPE= and ALPHA= options are missing, by default, 95% confidence ellipse for the prediction of an individual observation.

The code below is another example to create a scatter plot matrix as shown in Figure 2. The GROUP= option specifies a classification variable to divide the values into groups. The ELLIPSE= option for the MATRIX statement works the same way as for the COMPARE statement.

```
PROC SGSCATTER DATA=dset;
   MATRIX age waittime tumorsize exposuretime / GROUP=stage;
RUN;
```
PROC SGPLOT

The SGPLOT procedure is great for quickly creating statistical graphics such as box plots and histograms. Box plots are useful to graphically display the location of a variable and the spread of the values at a glance. They provide useful information of the variable's symmetry/skewness and are excellent for seeing if there are outliers.

The example of SAS code below and the plot it produces, shown in Figure 3, highlights just how simple and little code is needed to provide you with a lot of information.

```
PROC SGPLOT DATA = dset;
   HBOX density / CATEGORY=severity;
RUN;
```

The CATEGORY option allows you to produce a box plot at each level of the category variable for each of the variables specified after HBOX.
STEP 2: MODEL SELECTION

Before carrying out your MLR model building, in addition to getting to know your data, you need to consider what the scope of your project is, what the question of interest is and the goals of the analysis. Generally, you can classify the reason for wanting to carry out multiple linear regression modeling into two aspects:

- You want to investigate a collection of factors for their potential association with the outcome of interest.
- You want to investigate a collection of known relevant factors for their ability to predict the outcome of interest.

A few people may wonder what the difference between the two is. The first one tends to be more exploratory in nature, you do not know what predicts the outcome and the aim is to identify quantities of potential importance for further investigation or to help with a more restricted design for prediction. Generally, it is desired to exclude factors that are not statistically significant and therefore concentrate only on potentially interesting ones. You do need to be careful when several factors are investigated as the chance of finding a spurious (false-positive rate) effect increases with each additional test. With the second one, you often already have a hypothesis of what factors are known to influence the outcome. For prediction, you need to make sure you have the correct information/factors in order for you to be confident in your predictions. The simplest strategy is to attempt to model all factors and see how well the model predicts. However, generally, you would want to simplify the model by removing factors, as long as this does not compromise the predictability of the model. One aspect to remember, statistical significance alone is not always sufficient to assess the extent to which a factor can predict the outcome of interest.

On the whole, the goal is to obtain a parsimonious set of variables that efficiently predicts the response variable. When thinking about multiple linear regression there are 3 main SAS procedures that come to mind; PROC REG, PROC GLM and PROC GLMSELECT. PROC GLM supports the CLASS statement but does not utilize any model selection methods. However, there are a variety of model selection methods that can be used in PROC REG but it does not support the CLASS statement. PROC GLMSELECT utilizes both the CLASS statement and model selection methods but it does not include regression diagnostics or post model selection options. It is a reasonably new SAS procedure and is available in SAS 9.2 but can be downloaded from [http://support.sas.com/md/app/da/glmselect.html](http://support.sas.com/md/app/da/glmselect.html) for use in SAS 9.1. It is not available for previous versions of SAS. The intention is to primarily use PROC GLMSELECT as a model selection tool and use the macro variables provided by the procedure to take the selected model and further investigate it in either PROC REG or PROC GLM.

Since the paper is primarily concentrating on building multiple linear regression models using SAS, only a comparison of PROC REG and PROC GLMSELECT is made.

MODEL SELECTION METHODS AVAILABLE

Table 1 shows the different model selection methods that are available in each of REG and GLMSELECT procedures.

<table>
<thead>
<tr>
<th>Selection Method</th>
<th>PROC REG</th>
<th>PROC GLMSELECT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forward Selection &lt;FORWARD&gt;</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Backward Elimination &lt;BACKWARD&gt;</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Stepwise Selection &lt;STEPWISE&gt;</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Maximum R² Improvement &lt;MAXR&gt;</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Minimum R² Improvement &lt;MINR&gt;</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>R² Selection &lt;RSQUARE&gt;</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Adjusted R² Selection &lt;ADJRSQ&gt;</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Mallow’s Cₜ Selection &lt;CP&gt;</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Least Angle Regression Selection &lt;LARS&gt;</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Lasso Selection &lt;LASSO&gt;</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>
Each of the procedures includes the most common automated selection methods; forward selection, backward elimination and stepwise selection. The key code needed to specify each of the methods are shown within the brackets < >. The statistical details of these methods are not covered in this paper; however, we would like to bring to your attention quite a recent selection method, the Least Angle Regression Selection method. This is developed by Bradley Efron et al and first presented in 2004 in The Annals of Statistics, Vol 32(2); 407-499. It is useful when the number of parameters is large compared to the number of observations and is quite often used for gene expression data analysis.

**MODEL SELECTION SAS CODE**

Below is the basic syntax needed to specify the method you want to use. You can see that both are very similar, except PROC GLMSELECT has extra method options that can be specified.

<table>
<thead>
<tr>
<th>PROC REG</th>
<th>PROC GLMSELECT</th>
</tr>
</thead>
</table>
| PROC REG DATA = dset;  
MODEL y = vble_list / SELECTION = METHOD;  
RUN; | PROC GLMSELECT DATA = dset;  
MODEL y = vble_list / SELECTION = METHOD(<METHOD OPTIONS>);  
RUN; |

If the **SELECTION=** option is omitted from:
- PROC REG then no model selection is carried out and all terms in the model statement are included.
- PROC GLMSELECT then the default is **SELECTION=STEPWISE**

**Example 1 – specifying significance level criterion:**

Suppose that we have a dependent variable \(y\) and independent variables \(x_1, x_2\) and \(x_3\) and we want to use the well known forward selection method where variables are entered into the model until there are no remaining variables significant at the 5% significance level. The SAS code that seems intuitive to use for both PROC REG and PROC GLMSELECT is shown below:

**PROC REG:**

\[
\text{PROC REG:} \quad \text{MODEL } y = x_1 \times x_2 \times x_3 / \text{SELECTION = FORWARD; } \checkmark
\]

**PROC GLMSELECT:**

\[
\text{PROC GLMSELECT:} \quad \text{MODEL } y = x_1 \times x_2 \times x_3 / \text{SELECTION = FORWARD; } \times
\]

You may think the above two pieces of code are correct and would give you the same answer – **WRONG**!

In PROC GLMSELECT, although you have specified forward selection, you also need to specify that you want to use the significance level criterion for variables being entered into the model under the <METHOD OPTIONS>. This is also true if you specify the backward or stepwise methods. This is because there are multiple model selection criterions that can be specified in PROC GLMSELECT. However, when using these selection methods in PROC REG, the only selection criterion available is the significance level.

The selection criterions available in PROC GLMSELECT are:

- Adjusted R2 Statistic <ADJRSQ>
- Akaike Information Criteria <AIC>
- Corrected AIC <AICC>
- Mallow’s Cp Statistic <CP>
- Schwarz Bayesian Information Criteria <SBC> - this is the default criterion method used if no criterion is specified
- Significance Level <SL>
- Predicted Residual Sum of Squares Statistic <PRESS>
- Sawa Bayesian Information Criteria <BIC>
So, let us update the code:

**PROC REG:**  
MODEL \( y = x_1 \times x_2 \times x_3 \) / SELECTION = FORWARD; \( \times \)

**PROC GLMSELECT:**  
MODEL \( y = x_1 \times x_2 \times x_3 \) / SELECTION = FORWARD(SELECT = SL); \( \times \)

Are these now the same? – NO!

We need to consider the significance level at which variables are entered into the model. If the significance level is not specified then the SAS defaults are used. The problem here is that the defaults for PROC REG and PROC GLMSELECT differ. Table 2 shows the defaults that are used for the different selection methods in both PROC REG and PROC GLMSELECT. The key code needed for both procedures is the same. To enter variables into the model when using FORWARD and STEPWISE selection the key code needed is SLE=. The key code needed to keep variables in the model when using BACKWARD and STEPWISE selection methods is SLS=.

**Table 2: Significant Level Defaults Used in SAS for Variables Entering and Staying in the Model**

<table>
<thead>
<tr>
<th></th>
<th>PROC REG Defaults</th>
<th>PROC GLMSELECT Defaults</th>
</tr>
</thead>
<tbody>
<tr>
<td>Criterion for entering the model &lt;SLE=&gt;</td>
<td>FORWARD 0.50</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td>STEPWISE 0.15</td>
<td>0.15</td>
</tr>
<tr>
<td>Criterion for staying in the model &lt;SLS=&gt;</td>
<td>BACKWARD 0.10</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td>STEPWISE 0.15</td>
<td>0.15</td>
</tr>
</tbody>
</table>

Remember, when many significant tests are performed at a specified significance level, say 5%, the probability of falsely rejecting the null hypothesis becomes larger than 5%. Therefore, to guard against including variables that do not influence the response variable of interest, a small significance level should be specified. We would certainly caution against using the forward selection default for PROC REG.

Okay, so now let us use the new information and specify 5% significance level and obtain the equivalent code needed for Example 1 using both procedures. The two sample codes below are now equivalent:

**PROC REG:**  
MODEL \( y = x_1 \times x_2 \times x_3 \) / SELECTION = FORWARD SLE=0.05; \( \checkmark \)

**PROC GLMSELECT:**  
MODEL \( y = x_1 \times x_2 \times x_3 \) / SELECTION = FORWARD(SELECT=SL SLE=0.05); \( \checkmark \)

This example highlights the very subtle differences that occur between these two procedures. They are not necessarily intuitive and you really do need to be careful when writing the code to the specifications that you want.

**Example 2 – the STOP option:**

```sas
PROC GLMSELECT DATA = dset;
   MODEL y / SELECTION=FORWARD (SELECT=SL SLS=0.1 STOP=PRESS);
RUN;
```

In this example, independent variables are added to the model based on a significance level of 0.1. Selection terminates if adding any effect increases the predicted residual sum of squares (PRESS). If the STOP option is not
specified then by default, the statistic used to terminate selection is the same statistic that is used to select the variables. This aspect of model selection criteria is not available using PROC REG.

**Example 3 – the INCLUDE option:**

```
PROC REG:    MODEL y = x4 x6 x1 x2 x3 x5 / SELECTION=FORWARD SLE=0.15 INCLUDE=2;
PROC GLMSELECT: MODEL y = x4 x6 x1 x2 x3 x5 /
            SELECTION = FORWARD(SELECT=SL INCLUDE=2);
```

If independent variables are known to be associated with the dependent variable and it is essential to include them in the multiple linear regression model, then the INCLUDE= option (used in both procedures) can be used to specify the number of variables you want to include. However, those variables do need to be listed first in the variable list. For example, above, the INCLUDE=2 option indicates that the first 2 variables listed are to be included in the model regardless of their significance level. In PROC GLMSELECT, INCLUDE= needs to be within the brackets.

**Example 4 – categorical variables:**

Suppose we have a dependent variable ‘time’ and we want to investigate how two categorical independent variables ‘skill’ with 3 categories (1, 2, 3) and ‘fieldgp’ with 4 categories (1, 2, 3, 4) are related to time.

Using PROC REG:

PROC REG does not support the CLASS statement and therefore dummy variables need to be created for each of the categorical variables before running PROC REG.

For example, below shows the values the dummy variables take depending on the level of the variable skill. Dummy3 does not necessarily need to be derived since Dummy3 equals 1 when both Dummy1 and Dummy2 equal 0. SAS is clever enough to determine this, so if you did include Dummy3 as a variable SAS automatically sets the coefficient of Dummy3 equal to 0 and shows this in the output. Hence, it is essential to have the number of dummy variables to be at least 1 less than there are categories, but you can also use the same number of dummy variables as there are categories.

<table>
<thead>
<tr>
<th>Skill level</th>
<th>Dummy1</th>
<th>Dummy2</th>
<th>Dummy3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

The following is the SAS code that is needed to create the dummy variables for the example above and then PROC REG proceeds to use these dummy variables:

```
DATA newdset;
  SET dset;
  ARRAY dummy_skill [*] skill1-skill2;
  ARRAY dummy_fieldgp [*] fieldgp1-fieldgp3;

  DO i=1 to dim(dummy_skill);
    DO j=1 to dim(dummy_fieldgp);
      dummy_skill(i) = 0;
      dummy_fieldgp(j) = 0;
    END;
  END;
  dummy_skill(skill) = 1;
  dummy_fieldgp(fieldgp) = 1;
RUN;

PROC REG DATA = dset;
  MODEL time = (skill1 skill2) (fieldgp1 fieldgp2 fieldgp3) /
            SELECTION=FORWARD SLE=0.05 GROUPNAMES='Skill' 'Fieldgp';
RUN;
```
When modeling a categorical variable, if you want all the new dummy variables related to the categorical variable to be entered or removed from the regression model together, then the appropriate variables need to be enclosed in braces \{\}. A name can then be given to this subset of variables by specifying GROUPNAMES=. This is available when either the FORWARD, BACKWARD or STEPWISE selection method is used. GROUPNAMES can be up to 32 characters. Variables not enclosed by braces are considered separately.

To carry out an equivalent model selection process using PROC GLMSELECT the code needed is:

```
PROC GLMSELECT DATA = dset OUTDESIGN=dset2;
   CLASS skill fieldgp;
   MODEL time =skill fieldgp / SELECTION=FORWARD(SELECT=SL SLE=0.05);
RUN;
```

PROC GLMSELECT utilizes the CLASS statement and automatically creates the dummy variables for the categorical variables included in the CLASS statement. PROC GLMSELECT creates the same number of dummy variables as there are categories and in the output shows the coefficient for the last dummy variable as zero. These can be seen by requesting the dataset that contains the design matrix using OUTDESIGN=.

**Example 5 – including interactions in the model:**

If you suspected that a dependent variable depends both on main effects and two-way interactions and you want to check the significance of all two way interactions, using PROC REG you will need to create all the interaction terms in a prior data step and then include them in the model. Using the example used in Example 4, the PROC REG code is shown below. Similarly to Example 4, the interaction variable `skill2_fieldgp3` does not necessarily have to be derived and included in the model as SAS automatically sets its coefficient to zero.

```
PROC REG DATA = dset;
   MODEL lntm = {skill1 skill2} {fieldgp1 fieldgp2 fieldgp3}
    {skill1_fieldgp1 skill1_fieldgp2 skill1_fieldgp3
     skill2_fieldgp1 skill2_fieldgp2 skill2_fieldgp3}
   / SELECTION=FORWARD SLE=0.05
   GROUPNAMES='Skill' 'Fieldgp' 'skill*fieldgp_int';
RUN;
```

The code is much simpler using PROC GLMSELECT, as the pipeline between variables and “@2” at the end of the variable list can be used to specify all 2-way interactions:

```
PROC GLMSELECT DATA = dset;
   CLASS skill fieldgp;
   MODEL lntm = skill|fieldgp @2
   / SELECTION=FORWARD (SELECT=SL SLE=0.05);
RUN;
```

**CONCLUSION**

Multiple procedures may produce similar descriptive statistics but each procedure offers different unique features that can be utilized for better understanding of your data. It is always good practice to explore and understand your data before selecting a regression model and whilst in this paper we concentrated on examples using automated model selection methods, it is extremely important you use your knowledge and understanding of your data to drive your model selection.

Although in SAS there are multiple statistical procedures that can be used to perform the same analysis, this paper clearly indicates that transitioning between the different procedures needs to be done with care. If starting to use a new procedure, take the time to learn the differences between it and the procedure you are used to using. Transitioning between them is not always intuitive, different default parameters can be used, extra statements and options can be incorporated and extra data manipulation might be needed in one compared to another. However, once a new SAS procedure has be learned it can help improve your efficiency, be more flexible and open up new options and techniques to you.
In the comparison of PROC REG to PROC GLMSELECT, transitioning between the two is not always intuitive. PROC GLMSELECT provides greater flexibility and efficiencies with more options and statements made available. However, PROC REG has the advantage of “one stop shopping”; you are able to select the model and stay within the same procedure to carry out post-selection analyses, such as hypothesis testing, contrasts testing and least-square means analyses.

REFERENCES

- http://support.sas.com/
- SAS Help and Documentation
- Paper 231-2007: DATA: Learning What You Didn’t Know About Your Data. Larry Douglass, University of Maryland, College Park, Maryland

CONTACT INFORMATION

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

Colleen McGahan
Surveillance & Outcomes Unit, BC Cancer Agency
801 – 686 West Broadway
Vancouver, BC, V5Z 4C1

Mei Chong
Epidemiology Services, BC Centre for Disease Control
655 West 12th Avenue
Vancouver, BC V5Z 4R4

SAS and all other SAS Institute Inc. products 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.