Paper 135-31

Combining the Power of ODS, Data Set Concatenation, and DDE to Output Customized Statistical Results from SAS® to Microsoft Excel

Navneet K. Dhand, The University of Sydney, Camden, NSW, Australia
Jenny-Ann LML. Toribio, The University of Sydney, Camden, NSW, Australia

ABSTRACT

SAS is a widely used software package for statistical analyses, however presentation of customized results for reports and publications is cumbersome and time consuming if done manually and there is a high likelihood of incurring errors during the process. The method reported here makes the presentation of results easier and more efficient especially when a large number of variables are being tested and the procedure needs to be repeated many times. A three step procedure is suggested for outputting results from SAS to Excel: firstly, creation of data sets during model building employing the ODS OUTPUT statement, followed by concatenation of the data sets produced and finally exporting of customized results to specific cells of an Excel worksheet employing dynamic data exchange (DDE). The application of this approach employing the LOGISTIC procedure is illustrated in this paper for univariable and multivariable logistic regression modeling and for creating contingency tables for the outcome variable with explanatory variables. Modifications of the code for other linear and mixed modeling procedures like REG, GLM, MIXED and GLIMMIX are also briefly discussed.

INTRODUCTION

Statisticians, public health scientists, epidemiologists and social scientists-all use SAS for data analysis especially for statistical modeling. But MS Excel is frequently the preferred spreadsheet software for comprehension and presentation of the results and for tabulating data for publication in papers and reports. Transferring the results from SAS to Excel for this purpose is very time consuming and can also introduce errors. This is especially true for statistical modeling as the number of variables being tested is often very large and secondly, the process of model fitting is interactive and generally has to be repeated a number of times until the best model is attained. In this paper a simple technique is described to output the desired results to Excel and perform interactive analyses.

The technique consists of three simple steps: beginning with creating data sets during model fitting using the ODS OUTPUT statement, followed by concatenating data sets and finally outputting desired fields to Excel employing dynamic data exchange (DDE) between SAS and Excel. This paper illustrates use of this technique for outputting results of univariable and multivariable logistic regression to Excel and for performing interactive likelihood ratio (LR) tests. Modifications to the code for other procedures such as REG, GLM, MIXED and GLIMMIX are also presented in the end, though our aim is to provide guidelines so that you may create your own code according to your own requirements. The text is supplemented with tips and notes to resolve practical issues and to help you better implement this technique in your work. We assume a basic understanding of statistical methods and these SAS procedures, although a brief description is provided wherever necessary.

All codes described here are written in SAS version 9.1; however, they are equally suitable for version 8.2 (except PROC GLIMMIX) and brief information is provided wherever the code is different in the earlier version. Please note that all example analyses were done using a hypothetical data set and the values are shown just for pedagogical purposes.

To better comprehend the technique we will follow through the example of logistic regression analyses starting with univariable logistic regression.

UNIVARIABLE LOGISTIC REGRESSION ANALYSES

Suppose we have a data set (health) with 20 explanatory variables (var1 to var20) and an outcome variable (outcome). We will assume that the data set is stored in a library with LIBREF ***d*** while all temporary data sets (to be created in the process) are stored in the library with LIBREF ***t***. We will use three steps to output the results to Excel:

STEP 1: CREATING SAS DATA SETS EMPLOYING ODS OUTPUT STATEMENT

ODS OUTPUT is a commonly used statement for creating data sets from SAS output objects. The general form of the statement is as follows:

```
ODS OUTPUT <ods-table-name> = <data-set-name>;
```

where <ods-table-name> is the name assigned by SAS to the output table and <data-set-name> is any user-assigned name of the data set created in the process. We will use this statement to output results of logistic regression analyses to various data sets, for example in the following code, results of odds ratios are outputted to t.odds1 while those of 'type III tests of effects' to t.type1 and of 'fit statistics' to t.fit1. Similarly data sets for all the 20 variables are created as shown below:

```
    PROC LOGISTIC data=d.health;
    CLASS var1 / PARAM=ref REF=first;
    MODEL outcome = var1;
ODS OUTPUT oddsratios=t.odds1 type3=t.type1 fitstatistics = t.fit1;
RUN;
.                 .                   .                   .
.                 .                   .                   .
PROC LOGISTIC data=d.health;
    CLASS var20 / PARAM=ref REF=first;
    MODEL outcome = var20;
ODS OUTPUT oddsratios=t.odds20 type3=t.type20 fitstatistics = t.fit20;
RUN;
```

The contents of these data sets can be checked by using the PRINT procedure. The output created by printing table t.odds1 is shown below:

Obs	Effect	Unit	Odds RatioEst	LowerCL	UpperCL
1	Var1 1 vs 0	1.0000	2.886	1.081	8.054
2	Var1 2 vs 0	1.0000	10.481	3.884	30.198
3	Var1 3 vs 0	1.0000	6.902	3.951	9.627

TIPS AND NOTES

- In version 8.2 of SAS, the ODS table name for 'type III tests of effects' is TYPEIII instead of TYPE3. You may check the ODS table names in SAS documentation or can quickly verify these by issuing ODS TRACE ON statement prior to PROC statement. Information about all ODS table names for the current procedure will be shown in the SAS LOG. If the information is no longer required, you may issue ODS TRACE OFF statement, after which SAS will stop displaying such information.
- You would be better off by just increasing the suffix of the data set name with each variable (see code above). It is often easier to copy and paste the codes for various variables and then change the variable names and data set suffix numbers. Macro can also be written to loop through this code.

STEP 2: CONCATENATING DATA SETS

In this step we concatenate (merge vertically) the data sets produced in the first step. The general code is as under:

```
DATA <data-set-name>;
    SET <data-set1> <data-set2>;
RUN;
```

The observations from the data sets mentioned in SET statement are copied on to a new data set referred to in the DATA statement and are arranged in the order in which the data set names appear in the SET statement, that is, observations from the <data-set2> will follow those from the <data-set1>. For accurate merger, the different data sets should have the same variables. In the following example we will concatenate all the 20 data sets of odds ratios results created for 20 different variables in step 1, to create one data set (t.odds) having results of all variables. The contents of the data set t.odds can be checked by PROC PRINT (output will be similar to that shown earlier, but for all variables)

```
DATA t.odds;
    SET t.odds1 t.odds2 t.odds3 t.odds4 t.odds5 t.odds6 t.odds7 t.odds8 t.odds9
    t.odds10 t.odds11 t.odds12 t.odds13 t.odds14 t.odds15 t.odds16 t.odds17 t.odds18
    t.odds19 t.odds20;
RUN;
```

Similarly we can create one data set for 'type III tests of effects' and the other for 'fit statistics'. However, we can bypass this step by merging it with the 3rd step to produce same results as shown later.

STEP 3: OUTPUTTING THE REQUIRED VARIABLES TO EXCEL

This is the important step in which we will export the required variables from data sets to Excel sheet using DDE. There are many uses and options with DDE but we will restrict ourselves only to simplest possible form, sufficient for our purposes. Please refer to SAS documentation (TS 325, 1999) or detailed papers for further information (DelGobbo, 2003; Rhodes, 2005 and Vyverman, 2002). The general form of FILENAME statement is as under and is followed by a DATA step to finalize export of the results to Excel:

```
FILENAME <fileref> DDE <DDE-triplet> <DDE-options>
DATA <data-set-name>;
    SET <data-set-name>;
    FILE <fileref>;
    PUT <variable1-name> <variable2-name> <variable3-name>;
```

FILENAME statement establishes a DDE link with MS Excel and assigns the reference <fileref> to the Excel file. DDE triplet is practically the name of the Excel file and has three parts - application, topic and item (please see the example code below for better comprehension). EXCEL is the application for the purposes of this document and is always followed by a bar (|). Topic consists the Excel file name enclosed in square brackets ([ ]) and the spreadsheet name followed by an exclamation mark (!). Item specifies the row and column numbers of the spreadsheet to which the data is to be exported, for example R2C1:R100C4 in the code below will output data from 1^{st} column of the 2^{nd} row (A2) up to the 4^{th} column of the 100^{th} row (D100) in the Excel sheet 'odds' of file 'univariable'.
DDE-options are used to customize output. NOTAB is the only DDE-option that we will use in this document. It instructs SAS to ignore tab characters between variables and export each variable into a separate cell. Without this option each word of a character string is exported to separate cell which creates a lot of problems. This option is not required if the field name does not have blank spaces, however it is better to use it, if unsure. If we give a NOTAB option in FILENAME, we will have to force tabs ('09'x) between various variables specified for export in the PUT statement (see example below). FILE refers to <fileref> of FILENAME statement.

EXPORTING ODDS RATIOS

We will illustrate use of DDE step for outputting the contents of t.odds data set (containing results of odds ratios of all the variables) to Excel. First we will create an Excel file (say 'univariable') and then rename the sheets (say sheet1 as 'odds' and sheet2 as 'chi'). Code for exporting odds ratio results is shown below:

```
FILENAME nd DDE 'Excel| [univariable.xls] odds! R2C1:R100C4' notab;
DATA _null_;
    SET t.odds;
    FILE nd;
    PUT effect '09'x oddsratioest '09'x lowercl '09'x uppercl;
RUN;
```

where 'nd' is file reference for the Excel file 'univariable' and sheet 'odds' This file reference is again referred to in FILE statement. You may customize the code according to your requirements but the basic steps are as follows:

- Check the fields of the data set created in Step 2 by PROC PRINT.
- Use these field names in the PUT statement. You may omit the unnecessary variables from the PUT statement. Also, you can use the WHERE statement to subset the results (see examples below).
- Specify the number of rows and columns in the FILENAME statement according to the number of fields in PUT statement

In this example data set t.odds is read which consist of 6 fields, namely, OBS, EFFECT, UNIT, ODDSRATIOEST, LOWERCL and UPPERCL. But we have not included OBS and UNIT fields in the PUT statement as we do not want these to be outputted to Excel. We have specified four columns (C1 to C4) in FILENAME statement corresponding to 4 fields in PUT statement. Also, we have forced tabs ('09'x) between variables as we are using the NOTAB option. The first few rows of results of odds ratios exported in Excel are shown in Table 1.

	A	B	C	D
1	Effect	Odds ratios	Odds LCL	Odds UCL
2	Var1 1 vs 0	2.886	1.081	8.054
3	Var1 2 vs 0	10.481	3.884	30.198
4	Var1 3 vs 0	6.902	3.951	9.627
5	Var2 1 vs 0	3.954	1.513	10.868
6	Var2 2 vs 0	9.371	3.587	26.067

Table 1: Partial output of odds ratio results exported to Excel

TIPS AND NOTES

- To understand the utility of NOTAB option, temporarily omit it from the FILENAME statement as well as tabs ('09'x) from the PUT statement. All words of field effect will now be outputted to separate cells.
- You can check the number of rows in the data set by printing and use that number to specify rows in the FILENAME statement. Otherwise, you can allocate a higher number to the last row, for example, although we have only 20 variables but we have set R100 in the above example as odds ratios for some variables will occupy multiple rows in Excel sheet based on the number of categories. Thus it is always safer to specify a larger last row number unless it is exactly known.
- The computer resources can be used efficiently by specifying the data set name as _NULL_. SAS performs all functions but does not produce the data set which we do not need, anyway.
- It is often better to output results to the 2nd row of Excel sheet rather than the 1st, as then we can use the 1st row to write column headings as shown in Table 1. Although we can output the column headings directly from SAS to Excel in the same DATA step, the procedure is not being shown here.
- The Excel file can be created from scratch using DDE, and worksheet cells can even be formatted, however we have not given these codes for simplicity. Please refer to the specific documents for such information (Brown, 2005; Parker, 2003).
- Please note that Excel must be running and the requisite file should be open to submit these commands. Also, the sheet name should be the same as referenced in the FILENAME statement. If any of the above is not true, SAS will give the following error message:

```
ERROR: Physical file does not exist, Excel|[univariable.xls]odds!R2C1:R100C4.
NOTE: The SAS System stopped processing this step because of errors.
```

EXPORTING CHI SQUARE STATISTICS:

Procedure for outputting results of 'type III tests of effects' and 'fit statistics' is similar to that of 'odds ratios'. However, as we had not concatenated these data sets in Step 2, we will set up all 20 data sets in the SET statement instead of reading a single concatenated data set as done earlier. Thus the work of 2nd step can actually be performed in step 3 directly, and we can omit step 2 altogether.

```
*Type III tests of effects;
FILENAME nd DDE 'Excel| [univariable.xls] chi! R2C1:R100C4' notab;
DATA _null_;
    LENGTH effect $50;
    SET t.type1 t.type2 t.type3 t.type4 t.type5 t.type6 t.type7 t.type8 t.type9
    t.type10 t.type11 t.type12 t.type13 t.type14 t.type15 t.type16 t.type17 t.type18
    t.type19 t.type20
    FILE nd;
    PUT effect '09'x df '09'x waldchisq '09'x probchisq;
*Fit statistics;
FILENAME nd DDE 'Excel| [univariable.xls] chi! R2C5:R100C6' notab;
DATA _null_;
    SET t.fit1 t.fit2 t.fit3 t.fit4 t.fit5 t.fit6 t.fit7 t.fit8 t.fit9 t.fit10
    t.fit11 t.fit12 t.fit13 t.fit14 t.fit15 t.fit16 t.fit17 t.fit18 t.fit19 t.fit20;
    FILE nd;
    PUT interceptonly '09'x interceptandcovariates;
    WHERE criterion ='-2 Log L';
RUN;
```

Like the usual DATA step, the WHERE statement can be used after the PUT statement to select a particular result. In this case we have selected only '-2 log likelihood' results to be exported to Excel. We can also use the LENGTH statement after the DATA statement to specify the length of any variable being exported (see code above).

	A	B	C	D	E	F	G	H
1	Effect	DF	Wald chi	Wald P	Intercept	Covariate	LR Chi	LR P
2	Var1	3	26.2799	<.0001	248.697	219.888	28.809	0.0000
3	Var2	2	20.4527	<.0001	248.697	226.79	21.907	0.0000

Table 2: Partial printout of results of chi-square statistics exported to Excel and calculations performed.

Please note that we have outputted results from 'type III tests of effects' in columns A to D and results from 'fit statistic' in column E-F of the same sheet (Table 2). In this way the results from a particular variable are exported side-by-side. Once in Excel, we can calculate likelihood ratio chi square statistics by subtracting 'intercept and covariate' value of from the 'intercept' and then calculate P values using the CHIDIST function of Excel (Table 2, column G-H). We can use the degrees of freedom of the Wald statistics for calculating P value (Table 2, column B). Further, we can sort these results in Excel by P value and select the subset of variables for inclusion in multivariable analyses.

TIPS AND NOTES:

- Only calculate the chi square and P value for the first variable and then drag both the columns until the end of columns to fill values for remaining variables
- For univariable analysis we can also use the likelihood ratio chi square and P values from the ODS table GLOBALTESTS (Test for global null hypothesis), however, as we will be using 'fit statistics' results during multivariable analysis it is easier to use the same procedure in univariable analyses.
- Please note that we have assumed all variables to be categorical. If some variable is continuous, 'type III tests of effects' for that variable will not be produced which will affect order of variables in the Excel sheet. In such cases, it would be safer to put data sets of continuous variables at the end during concatenation.
- The term 'variable' is used instead of 'effect' in SAS version 8.2. Please verify the names by PROC PRINT before writing PUT statement.

CREATION OF CONTINGENCY TABLES

During logistic regression analysis we need contingency tables for cross-tabulation of the outcome variable with all explanatory variables for presentation of results in papers and reports and for checking cell frequencies so as to identify any cell with zero or low frequency. SAS usually outputs results for an outcome variable with each explanatory variable separately and it is a very arduous job to put together all the results. However, while performing the univariable analysis as above, we can easily create and export contingency tables to Excel (Table 3). Like other data sets created during univariable analysis, a data set for results of contingency tables can also be produced for each variable using ODS table CLASSFREQ and the option SIMPLE in PROC LOGISTIC statement. Code for outputting the results into a sheet (say frequency) of an Excel file (say univariable) is shown below (assuming that the outcome variable is binary with two categories 0 and 1 and data sets names range from freq1 to freq20 for the 20 different variables). Again, it is useful to print any one of the data sets to check the field names for specifying in PUT statement

```
FILENAME nd DDE 'Excel| [univariable.xls] frequency!R2C1:R200C5' notab;
DATA _null_;
    SET t.freq1 t.freq2 t.freq3 t.freq4 t.freq5 t.freq6 t.freq7 t.freq8 t.freq9
    t.freq10 t.freq11 t.freq12 t.freq13 t.freq14 t.freq15 t.freq16 t.freq17 t.freq18
    t.freq19 t.freq20
    FILE nd;
    PUT class '09'x value '09'x _0 '09'x _1 '09'x Total;
RUN;
```

	A	B	C	D	E
1	Effects	category	Outcome_0	Outcome_1	Total
2	Var1	0	11	7	18
3		1	6	12	18
4		2	2	14	16
5		3	0	8	8
6	Var2	0	11	7	18
7		1	6	13	19
8		2	2	21	23

Table 3: Contingency tables for outcome variable with explanatory variables

CORRELATION ANALYSES

Before commencing a multivariable analysis we often have to check the presence of highly correlated variables. This is usually done with Spearman rank (or Pearson) correlation coefficient and chi-square test. The FREQ procedure with MEASURES and CHISQ options is used for this purpose and SAS produces hundreds of pages of output which is very difficult to comprehend. Using the following procedure we can output the desired results into an Excel sheet which is more convenient for inspection (Table 4). ODS table 'measures' produced in the FREQ procedure is converted into a SAS data set (Step 1 below) followed by export of the requisite results to Excel (Step 2a). Similarly we can evaluate the significance of association using chi square by exporting the desired results to Excel (Step 2b).

```
*Step1: Creation of SAS data set containing results of table 'measures';
PROC FREQ data= d.health;
    TABLES (var1 var2 var3 var4 var5 var6 var7 var8 var9 var10 var11 var12 var13
    var14 var15 var16 var17 var18 var19 var20)*
    (var1 var2 var3 var4 var5 var6 var7 var8 var9 var10 var11 var12 var13 var14
    var15 var16 var17 var18 var19 var20)/NOPERCENT NOROW NOCOL MEASURES CHISQ;
    ODS OUTPUT measures=t.set chisq = t.chi;

*Step2a: Export of the Spearman correlation coefficient results;
FILENAME nd DDE 'Excel|[univariable.xls]correlation!R2C1:R5000C3'notab;
DATA _null_;
    LENGTH table $50;
    SET t.set;
    FILE nd;
    PUT table '09'x value '09'x ase;
    WHERE statistic ='Spearman Correlation';
/* can also specify 'Pearson Correlation' instead, according to requirements*/
RUN;

*Step2b: Export of chi-square results (partial code);
FILENAME nd DDE 'Excel|[univariable.xls] correlation!R2C4:R5000C6' notab;
    SET z.chi;
    PUT table '09'x value '09'x Prob;
    WHERE statistic ='Chi-Square';
```

	A	B	C	D	E	F
1	Variable tables	Spearman correlation	ASE	Variable tables	Chi-square	P value
2	Table Var1 * Var1	1	0	Table Var1 * Var1	1346	<.0001
3	Table Var2 * Var1	-0.0595	0.0369	Table Var2 * Var1	8.9699	0.0113
4	Table Var3 * Var1	-0.0656	0.0382	Table Var3 * Var1	3.8608	0.1451

Table 4: Results of association between various variables

TIPS AND NOTES:

- Results can be sorted by correlation value or P value (as per requirement) in SAS before outputting or later in Excel to identify variables with highest positive and negative associations.
- Instead of writing names of all variables in the above TABLES statement, we can use _all_*_all_ to specify correlation of all variables with each other.
- To avoid cluttering, you should delete all the temporary data sets in the library **t** after any procedure by running the following code. Take care not to delete any useful data set!

```
PROC DATASETS library =t KILL; RUN;
```

MULTIVARIABLE LOGISTIC REGRESSION ANALYSIS:

Use of the LR test to assess variable significance in SAS is unwieldy because you have to fit both a full and a reduced model and then check -2 log likelihood of both models, subtract one from the other manually to get chi-square value and then check P value from the statistical tables corresponding to this chi-square value. Further this procedure has to be repeated for each variable (may be multiple times) and for each interaction term. Many scientists end up using the less robust Wald chi square test instead whose results are printed automatically in SAS. The LR tests can be conveniently calculated using the present technique (see code below). The procedure is similar to univariable analysis, however, code for PROC LOGISTIC has to be written only once and all the variables can be tested for inclusion (or exclusion) in the same code. Once the results are in Excel, the chi-square statistic is calculated by subtracting 'intercept and covariates' of the model *with the variable* from the model *without the variable* (Table 5, column F). Again, we can get degrees of freedom from the Wald test for calculating P value using CHIDIST function in Excel. Next the model is fitted by including or excluding the variable being tested, as the case may be, and the process is repeated for all variables. Thus we can take decisions about the significance of a particular variable by performing modeling work in SAS and then calculating LR statistics in Excel. It also helps in tracking the model building process and why and how any variable was deleted or retained.

```
PROC LOGISTIC data = d.health;
    CLASS var1 var2 var3 /PARAM=ref REF=first;
    MODEL outcome = var1 var2 var3 / clodds=Pl clparm =Pl;
ODS OUTPUT type3=t.type fitstatistics = t.fit clparmpl=t.parm cloddspl=t.odds;

* Type III tests of effects (partial code);
FILENAME nd DDE 'Excel| [multivariable.xls] model! R4C1:R100C4' notab;
    PUT effect '09'x df '09'x waldchisq '09'x probchisq;
    WHERE effect = 'var3';

* fit-statistics table (partial code);
FILENAME nd DDE 'Excel| [multivariable.xls] model! R4C5' notab;
PUT interceptandcovariates;
WHERE criterion ='-2 Log L';
```

TIPS AND NOTES:

- For multivariable analysis we only need to export the 'interceptandcovariates' field, therefore, we have specified only one cell (R4C5) in FILENAME statement.
- During forward selection, we can restrict the export of results to the current variable being tested by WHERE statement (see code). Results of the next variable can then be exported by just changing row numbers in FILENAME statements and changing the nominated variable in the WHERE statement. Also, if you do not want to change row numbers every time you test a variable, you can set the row numbers to high at the outset, for example R20. When the results are outputted to that row, you can drag the results to the desired row. In this way you can test 20 variables without changing the row or column numbers: just add the variable and change the variable in the where statement!
- For backward selection, we require the code for 'type III tests of effects' to be run only once (for one step) while code for 'fit statistics' is run every time by excluding all variables one by one.
- It is better to leave two columns blank after the 'intercept and covariate' column to allow for calculation of LR chi-square and P values in Excel (Table 5, column F and G). Also one column can be left blank for writing comments during model building process (Table 5, column L).
- In this example, we want to calculate profile likelihood confidence limits for parameter estimates and odds ratios, therefore, the options and ODS table names are different from the example given for the univariable analysis. However, any type of confidence limits can be produced in both univariable and multivariable analyses based on your requirement.
- While performing a LR test of significance, due care should be taken so that analysis with and without the variable is done for the same set of observations. Thus when some variable has missing values, the model should be fitted first by including it only into the CLASS statement (of course, if it is categorical) and then by including it both into CLASS and MODEL statement. The value of 'intercept and covariate' obtained in these two models can then be used safely to calculate the LR chi-square and P value.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Variable	DF	Wald chi	Wald P	-2 LnL	LR-Chi	LR-P	AIC	NOBS	R^2	C	Comments
2	Var18	2	16.54	0.000	118.01			130.01	77	0.45	0.82	
3	Var9	1	11.55	0.001	103.01	15.00	0.000	117.01	77	0.57	0.87	
4	Var5	1	6.81	0.009	95.55	7.46	0.006	111.55	77	0.63	0.89	

Table 5: Multivariable logistic regression interactive analyses

Some research workers prefer using some other criteria for model selection such as Akaike's information criterion (AIC) instead of (or in addition to) LR chi-square test. These results can also be exported in the same row in Excel sheet (for example to column H in Table 5 using the following code).

```
FILENAME nd DDE 'Excel|[multivariable.xls]chi!R2C8' notab;
PUT interceptandcovariates;
WHERE criterion ='AIC';
```

This procedure can be extended by exporting some other results in the same row of the Excel sheet using statements shown in Table 6. Printout of some of these results exported to Excel is shown in Table 5.

Results	ODS table name	PUT	WHERE
Number of observations used	NOBS	nobsused	-
Deviance goodness-of-fit test	GOODNESSOFFIT	chisq	criterion = 'Deviance'
Pearson goodness-of-fit test	GOODNESSOFFIT	chisq	criterion= 'Pearson'
Max-rescaled R-square	RSQUARE	nvalue2	-
c	ASSOCIATION	nvalue2	label2 ='c'
Test of cumulative model assumption (Only ordinal logistic regression)	CUMULATIVE MODELTEST	probchisq	-

Note: Options RSQUARE and SCALE = NONE AGGREGATE must be given in the model statement to calculate R-square and test of cumulative model assumption, respectively.

Table 6: The ODS table names and required statements for outputting some desired results to Excel

In addition to the above results, we also need to know parameter estimates and odds ratios during model building and for publication of the final model. Codes for these are similar to the univariable model and thus only FILENAME and PUT statements are being shown here. We can output both parameter estimates and adjusted odds ratios in the same Excel sheet in adjacent columns. As parameter estimates include one row for intercept (for binary outcome), it is better to increase row number of odds ratios results by one (R3 versus R2 in the code below) so as to export all the results of a particular variable to the same row(s) (Table 7).

```
*parameter estimates;
FILENAME nd DDE 'Excel|[multivariable.xls]odds!R2C1:R100C5' notab;
PUT parameter '09'x classval0 '09'x estimate '09'x lowercl '09'x uppercl;
*Odds ratios;
FILENAME nd DDE 'Excel|[ multivariable.xls]odds!R3C6:R100C9' notab;
PUT effect '09'x oddsratioest '09'x lowercl '09'x uppercl;
```

	A	B	C	D	E	F	G	H
1	Variables	Parameter estimates	LCL	UCL	Classes	Odds ratios	LCL (odds)	UCL (odds)
2	Intercept	-0.86	-4.12	2.41				
3	Var6	-1.00	-3.96	0.81	Var6 1 vs 0	0.37	0.02	2.25
4	Var6	-0.35	-3.31	1.47	Var6 2 vs 0	0.71	0.04	4.35
5	Var11	0.88	0.46	1.32	Var11 1 vs 0	2.41	1.58	3.76

Table 7: Results of parameter estimates and odds ratios exported to adjacent columns in Excel

CODES FOR OTHER MODELLING PROCEDURES

The logistic regression analysis example was used in this paper to illustrate the technique. Examples for application of this technique for other model building procedures like PROC REG, GLM, MIXED and GLIMMIX are given below. As most of the code is similar to the logistic regression example, therefore only PUT and/or SET statements are given, besides of course, the respective modeling procedure.

PROC REG

```
*Creation of data sets by ODS OUTPUT;
PROC REG data= d.health;
    MODEL continuous_outcome = variable1 dummy1 dummy2/ VIF CLB;
    DUMMYTEST: TEST dummy1, dummy2;
    ODS OUTPUT parameterestimates = z.parm testanova = z.anova;
* Parameter estimates;
    SET z.parm;
    PUT variable '09'x Estimate '09'x stderr '09'x tValue '09'x Probt '09'x
        varianceinflation '09'x lowercl '09'x uppercl;
* Test for dummy variables;
    SET z.anova;
    PUT test '09'x source '09'x df '09'x MS '09'x Fvalue '09'x Probf ;
```

PROC GLM

```
*Creation of data sets by ODS OUTPUT: we have used output Label "Type III Model
ANOVA" instead of table name ModelANOVA as we want to export only type III results;
PROC GLM data = d.health;
    CLASS var1;
    MODEL continuous_outcome =var2/solution clparm;
    ODS OUTPUT ParameterEstimates=z.odds2 "Type III Model ANOVA"=z.type2;
* Sum-of squares statistics;
    PUT source '09'x df '09'x ss '09'x ms '09'x fvalue '09'x probf;
* Parameter estimates;
    PUT parameter '09'x estimate '09'x lowercl '09'x uppercl '09'x stderr;
```

PROC MIXED

```
*Creation of data sets by ODS OUTPUT;
PROC MIXED data = d.health covtest cl;
    CLASS var1 var2 var3 var4 random_variable;
    MODEL continuous_outcome = var1 var2 var3 var4/ solution chisq cl;
    RANDOM random_variable;
    Parms;
    ODS output CovParms=t.parm tests3 = t.set solutionf =t.fixed;
* Covariance Parameter estimates;
    SET t.parm;
    PUT covparm '09'x estimate '09'x stderr '09'x lower '09'x upper;
*Fixed effect parameter estimates;
    SET t.fixed ;
    PUT effect '09'x estimate '09'x stderr '09'x lower '09'x upper '09'x probt;
*Type 3 tests of fixed effects;
SET z.set ;
PUT effect '09'x Chisq '09'x probchisq ;
```

PROC GLIMMIX

```
*Creation of data sets by ODS OUTPUT;
PROC GLIMMIX data = n.ojdrfs_mptb OR;
    CLASS var1 var2 var3 random_variable;
    MODEL outcome = var1 var2 var3 /solution cl chisq ;
    RANDOM intercept /subject =random_variable ;
    ODS OUTPUT tests3 = z.type covparms = z.set parameterestimates = z.param
    oddsratios = z.odds;
*Type III Tests of Fixed Effects;
    SET z.type;
    PUT effect '09'x chisq '09'x probchisq;
*Covariance Parameter Estimates;
    SET z.set;
    PUT covparm '09'x subject '09'x estimate '09'x stderr;
*Fixed Effects Parameter estimates;
    SET z.param;
    PUT effect '09'x estimate '09'x lower '09'x upper '09'x stderr '09'x probt ;
*Odds ratio estimates;
    SET z.odds;
    PUT effect '09'x estimate '09'x lower '09'x upper;
```

CONCLUSION

This technique outputs results from model building in SAS to specified cells in MS Excel which substantially aids comprehension and preparation of customized tables for reports and publications. It is especially useful when large numbers of explanatory variables need to be tested. We summarize and generalize the procedure in the list below so that it can be used and adapted for individual needs:

1. Create a new Excel file with appropriate name and sheet name.
2. Create data sets in normal modeling procedure by ODS OUTPUT statement.
3. Check the names of variables in these data sets by PROC PRINT and use them in the PUT statement.
4. Specify the row and column numbers of Excel sheet in the FILENAME statement corresponding to variables in the PUT statement.
5. SET the data set (or concatenate data sets in case of multiple data sets).

REFERENCES

Brown, David. “%sas2xl: A Flexible SAS® Macro That Uses Tagsets to Produce Complex, Multi-Tab Excel Spreadsheets with Custom Formatting” *Proceedings of 30th Annual SAS Users Group International Conference*. April, 2005. <http://www2.sas.com/proceedings/sugi30/092-30.pdf>

DelGobbo, Vincent. “A Beginner’s Guide to Incorporating SAS® Output in Microsoft® Office Applications” *Proceedings of the 28th Annual SAS Users Group International Conference.* March-April, 2003. <http://www2.sas.com/proceedings/sugi28/052-28.pdf>

Parker, Chevell. “Generating Custom Excel Spreadsheets using ODS” *Proceedings of the 28th Annual SAS Users Group International Conference.* March-April, 2003. <http://www2.sas.com/proceedings/sugi28/012-28.pdf>

Rhodes, Phil. “Automating the Drudgery Away: Using Macros and ODS to Produce (Almost) Complete Reports” *Proceedings of 30th Annual SAS Users Group International Conference.* April, 2005. <http://www2.sas.com/proceedings/sugi30/011-30.pdf>

TS 325. SAS Institute Technical Support Document number TS 325. 1999. “The SAS System and DDE.” <http://ftp.sas.com/techsup/download/technote/ts325.pdf>

Schreier, Howard. “Getting Started with Dynamic Data Exchange” *Proceedings of the Sixth Annual Conference of the SouthEast SAS Users Group*. September 1998. <http://www.dc-sug.org/>

Vyverman, Koen. “Creating Custom Excel Workbooks from Base SAS with Dynamic Data Exchange: A Complete Walkthrough” *Proceedings of the 27th Annual SAS Users Group International Conference*. April 2002. <http://www2.sas.com/proceedings/sugi27/p190-27.pdf>

ACKNOWLEDGMENTS

We are thankful to Professor Richard J Whittington, Chair Farm Animal Health and Dr Peter Thomson, Senior Lecturer in Veterinary Biometry, University of Sydney, Australia for their valuable advice.

CONTACT INFORMATION

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

Navneet Kumar Dhand
The University of Sydney
425 Werombi Road
Camden, NSW 2570. Australia
Phone: 02 93511610
Fax: 02 93511693
E-mail: navneetd@camden.usyd.edu.au