Paper 377-2008

ODS for the Professional Statistician

Steven M. LaLonde, Rochester Institute of Technology, Rochester, NY

ABSTRACT

ODS, or the Output Delivery System, is used to control the output from various SAS® procedures. Being a statistician first, and SAS programmer a very distant second, I wasn't very interested in ODS when I first heard about it. I figured ODS was for people concerned more with appearance than substance. However, after playing a bit, I'm convinced that there are some really useful features there, even for a stubborn, stuffy, old statistician like me.   This paper will demonstrate some ODS features, and tricks, that I've picked up that I think will be useful to any statisticians using SAS, and may even encourage non-SAS statisticians to try SAS...

INTRODUCTION

Without even knowing it, I've been using ODS whenever I used SAS to produce procedural output. ODS has been displaying the results from my programs without any help from me! I had played a little with the choices under the Results tab under Tools/Options/Preferences, but hadn't really thought much about the corresponding SAS commands that were being executed. It turns out that the "LISTING" and "HTML" check boxes, along with the "Style" pull-down menu correspond to fairly simple ODS commands that could just as easily be included in your program.

Selecting the checkbox for "Create listing" is equivalent to the following program:

```
ODS LISTING;
*** Put SAS Statements here ***;
ODS LISTING Close;
```

Selecting the checkbox for "Create HTML" is equivalent to the following program:

```
ODS HTML;
*** Put SAS Statements here ***;
ODS HTML Close;
```

Of course, both checkboxes may be selected, and is equivalent to the following program:

```
ODS HTML;
ODS LISTING;
*** Put SAS Statements here ***;
ODS HTML Close;
ODS LISTING Close;
```

The default HTML display was not all that attractive to me. It was a bit more interesting than the generic SAS listing, but it wasn't really something that I would cut and paste into a technical report, or show to a class on a PowerPoint slide. Virtually all of my technical reports to clients and handouts to students are printed in black and white. In general, I don't tend to upload my results to a Web page, so the fancy shading and colors just aren't useful to me.

I started playing with the "Style:" pull down menu to see how I could change the appearance of the HTML output. There are lots of choices. Some had names that I thought would appeal to the serious academic that I am: Analysis, Education, Normal, Printer, Science, Statdoc, and Statistics. Others had names that I didn't anticipate would appeal to me: Astronomy, Curve, Festival, Gears, Meadow, and Watercolor. You pretty much have to look at the output created from the various Style choices to see what appeals to you, or fits your particular need.

For me, to get fairly predictable results, with the professional (boring) look that I'm going for, I chose the Journal style. You can make this selection with the pull down menu, or add the Style= option to your ODS HTML statement in your SAS code:

```
ODS HTML Style=Journal;
*** Put SAS commands here ***;
ODS HTML Close;
```

At this point I had created some decent looking output in the Results Viewer, and had exhausted the extent to which I could use ODS with pull down menus on the programming interface. I still had some issues with using the ODS HTML output. When I copied a table from the Results Viewer and pasted it into Microsoft Word, the table had a black background. This could be fixed by Right-Clicking on the table in Word, choosing Borders and Shading, and changing the Fill to No Fill. Still other problems remained, the cell borders were also not as they had appeared in the HTML output and would also have to be changed in Word to create the same appearance. It turns out that in order for ODS to be useful to me, I was going to have to learn to do more than could be done with the pull down menus.

CREATING WORD FRIENDLY ODS OUTPUT

The options “listing” and “HTML” are just two of the many possible destinations for your ODS output. Other somewhat familiar destinations might include: PCL, PDF, PRINTER, PS, AND RTF. In particular, the RTF destination creates a file in the “Rich Text Format”, designed specifically to be edited by Microsoft Word. If you cut and paste output from the Results Viewer of an ODS RTF file, then you get much more predictable results.

If you are going to copy/paste your results into an existing Word document, then you may find a few other options useful. I like to use the NoDate, NoCenter, and NoNumber SAS System options to simplify the formatting. Try surrounding your SAS Statements with the following ODS statements:

```
   OPTIONS NoCenter NoDate NoNumber;
   ODS LISTING;
   ODS RTF Style=Journal startpage=no;
   ODS NoPtitle;
   *** Put SAS Statements here;
   ODS LISTING Close;
   ODS RTF Close;
   RUN;
```

The option StartPage=No on the ODS RTF statement prevents SAS from creating a page separation before each PROC. Adding the NoPtitle ODS statement prevents the usual labeling of the output with the procedural title such as “The REG Procedure”. I don’t really need a reminder of which PROC I ran, and it really only clutters up things. At this point I have some very nice looking output that I find very suitable for cutting and pasting into a report.

The main reason that I am still cutting and pasting into another document is that there is an awful lot of extraneous output that I don’t want in my final document. I often times want only one table out of the whole procedural output, so it is easier to copy the one table than it is to delete everything I don’t want. But, of course, there is a fix for that too…

CHOOSING WHICH OUTPUT TO OUTPUT

If you only want to output a portion of the procedural output to the ODS destination, you can use a Select or Exclude option to specify the tables of interest. The trick is to know what the names of the elements that you want to keep are. This is generally accomplished in two steps: 1) use the ODS Trace statement to list out the names of all the objects produced by a given procedure, then, 2) re-run the procedure preceding it by and ODS Select or ODS Exclude statement so that only parts of the output are displayed.

Suppose, for instance, that you want to use PROC CONTENTS to display the list of variables and their attributes from a given SAS data set, but that you don’t want all that other stuff that comes along with PROC CONTENTS such as the Engine/Host Information. First we use the ODS Trace statement to find out the names of all the objects that PROC CONTENTS produces:

```
   LIBNAME sgf “C:\sgf\”;;
   OPTIONS NoCenter NoDate NoNumber LS=64;
   ODS LISTING;
   ODS RTF Style=Journal StartPage=no;
   ODS NoPtitle;
   ODS Trace On;

   PROC CONTENTS data=sgf._all_;
   RUN;

   ODS Trace Off;
   ODS LISTING Close;
   ODS RTF Close;
   RUN;
```

Of course, you don’t really need the ODS RTF statements or the ODS NoPtitle statement in order to get the trace, but I’ve included them here just to be consistent. The list of ODS tables appears in the SAS log. Here is a partial listing of what you will see:

```
Output Added:
-------------
Name:       Directory
Label:      Directory Information
Template:   Base.Datasets.Directory
Path:       Contents.Directory
-------------
```

```
Output Added:
-------------
Name:       EngineHost
Label:      Engine/Host Information
Template:   Base.Contents.EngineHost
Path:       Contents.DataSet.EngineHost
-------------

Output Added:
-------------
Name:       Variables
Label:      Variables
Template:   Base.Contents.Variables
Path:       Contents.DataSet.Variables
-------------
```

By matching up the list of elements in the SAS log, with the tables that you see in the listing, you can usually guess which element matches with which table.

In this case we are looking for the “Variables”, which has a path of: “Contents.DataSet.Variables”. Now we are ready to Select the specific table we want to include in the RTF file. Let’s include a more appropriate title for the top of the page, and while we are at it, let’s specify a specific place, and name for our RTF file:

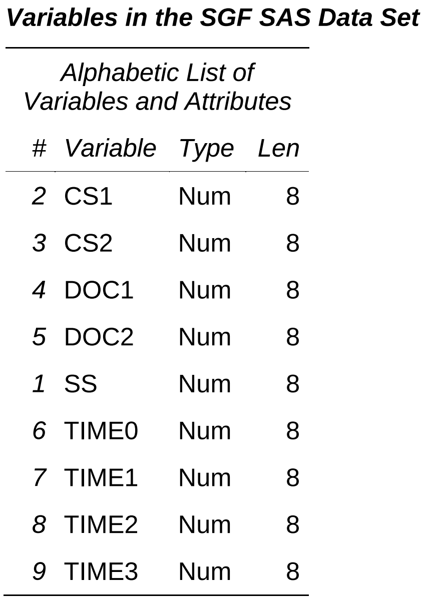
```
   LIBNAME sgf “C:\sgf\”;;
   OPTIONS NoCenter NoDate NoNumber LS=64;
   ODS LISTING;
   ODS RTF Style=Journal StartPage=no
           File=”C:\sgf\sample.RTF”;
   ODS NoPtitle;
   ODS Trace On;
   ODS Select Contents.DataSet.Variables;

   PROC CONTENTS data=sgf._all_;
   TITLE “Variables in the SGF SAS Data Set”;
   RUN;

   ODS Trace Off;
   ODS LISTING Close;
   ODS RTF Close;
   RUN;
```

The file “sample.RTF”, without any editing, is shown in Figure 1:

FIGURE 1:



Variables in the SGF SAS Data Set

Alphabetic List of Variables and Attributes

#	*Variable*	*Type*	*Len*
2	CS1	Num	8
3	CS2	Num	8
4	DOC1	Num	8
5	DOC2	Num	8
1	SS	Num	8
6	TIME0	Num	8
7	TIME1	Num	8
8	TIME2	Num	8
9	TIME3	Num	8

The title appears in the Header of the document and the rest of it appears as a Word table. Of course, if this look is too plain for your liking, you can experiment with the other choices of Style. This capability would have saved me hours of post editing on a project that I did recently. I had run dozens of logistic regressions for a client, but didn't want to include all of the typical output from PROC LOGISTIC in the final report. I had already learned how to output to RTF files, but I had not learned to use the Select option. So I copied, and pasted, and deleted my way to a final report. Being an Old Dog, it takes a very juicy bone to get me motivated enough to learn a new trick, and it was actually the features described in the next section that encouraged me to learn more about ODS.

COOL NEW (TO SAS) GRAPHICS

SUGI 29 was held in Montreal, Canada, which is only about a six-hour drive from where I now live. I also pass by the place where I grew up on the way there. That, along with the fact that I recently took over our statistical computing course prompted me to think about attending my first SUGI conference in many years. There was also the fact that SAS had recently released Version 9.1, which seemed to have some really interesting new features that I had been teased with at a local SAS User's Group meeting.

Interestingly enough, it was the scatterplot matrix that SAS Version 9.1 can produce from the PROC CORR procedure that got my attention. I had written SAS GRAPH® code to produce a scatterplot matrix more than 10 years ago, as many SAS User's have, but now SAS has figured out a way to include it in the package. Without getting into "too little, too late" argument with my colleagues that have switched to other packages, I have to admit that I'm pretty excited about the "Experimental" graphics that are included with Version 9.1. To produce a scatterplot matrix in SAS Version 9.1 you need an ODS Graphics statement, along with a PLOTS option on the PROC CORR statement:

```
LIBNAME sgf “C:\sgf\”;
OPTIONS NoCenter NoDate NoNumber LS=64;
ODS LISTING;
ODS RTF Style=Journal StartPage=no
        File="C:\sgf\sample2.RTF";
ODS NoPtitle;
ODS Trace On;
ODS Graphics On;

PROC CORR Data=sgf.rawdata Plots=(Matrix);
var time0-time3;
TITLE “Matrix Plot";
RUN;

ODS Trace Off;
ODS Graphics Off;
ODS LISTING Close;
ODS RTF Close;
RUN;
```

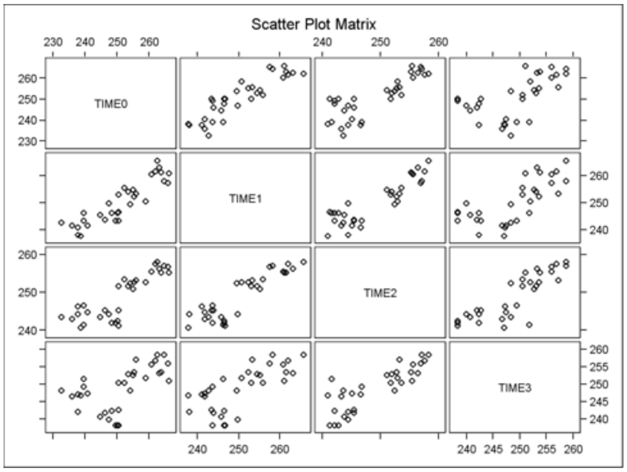
Of course, this program puts a whole bunch of output in addition to the matrix plot into the RTF file. Since I ran the program with ODS Trace ON, I can look in the SAS Log to see what actual ODS elements have been created. If I only want to show the correlation matrix, and the matrix scatterplot, for instance, I would need only to add a SELECT statement before PROC CORR:

```
ODS SELECT
PearsonCorr
    SymMatrixPlot;
```

The two elements, the first a table element, and the second a graph element, are shown below:

Pearson Correlation Coefficients, N = 32
Prob > |r| under H0: Rho=0

	TIME0	*TIME1*	*TIME2*	*TIME3*
TIME0	1.00000	0.89620 <.0001	0.84285 <.0001	0.57085 0.0006
TIME1	0.89620 <.0001	1.00000	0.89508 <.0001	0.71853 <.0001
TIME2	0.84285 <.0001	0.89508 <.0001	1.00000	0.84178 <.0001
TIME3	0.57085 0.0006	0.71853 <.0001	0.84178 <.0001	1.00000



When you turn on ODS Graphics in Version 9.1, many procedures will display high-resolution graphics without any additional options specified. For a detailed list of graphics produced by default in Version 9.1, look at the SUGI 29 Paper 204-29 by Robert N. Rodriguez. For a comparison of ODS Graphics to SAS/GRAPH plots that statisticians have probably been programming all along, take a look at “Thumbs Up for ODS Graphics, But Don’t Throw Out All Your SAS/GRAPH Programs!” by Rick M. Mitchell, SUGI 29, Paper 083-29. To understand all the options available, you would need to read Chapter 15 for the SAS/STAT® 9.1 User’s Guide, “Statistical Graphics Using ODS (Experimental)”, and the individual procedural chapters in the SAS/STAT 9.1 User’s Guide.

Since ODS Graphics are experimental in Version 9.1, it is possible that the availability or format of individual graphics, as well as the syntax used to create the graphics, may change in future releases.

Let’s take a more detailed look at simple linear regression as an example. Here is a PROC REG sandwiched in the middle of our ODS code we’ve been using.

```
LIBNAME sgf “C:\sgf\”;;
OPTIONS NoCenter NoDate NoNumber LS=64;
ODS LISTING;
ODS RTF Style=Journal startpage=no
        file="C:\sgf\sample3.RTF";
ODS NoPtitle;
ODS Trace On;
ODS Graphics On;

PROC REG Data=sgf.rawdata;
MODEL time1 = time0;
TITLE “Simple Linear Regression";
QUIT;
RUN;

ODS Trace Off;
ODS Graphics Off;
ODS LISTING Close;
ODS RTF Close;
RUN;
```

The code in this form produces seven elements and four pages of output. The first page contains three table elements, and the next three pages include a panel of fit diagnostics, a residual plot, and a fit plot. The panel is made up of nine individual elements that can be separated by adding “Plots(UnPack)” as an option on the PROC REG statement:

```
   PROC REG Data=sgf.rawdata Plots(UnPack);
```

If the fit diagnostics panel is unpacked, then there are fourteen individual ODS elements and eleven pages of output. As more independent variables are added, the number of pages rises very quickly. We can examine the SAS Log for the names of the elements that we are interested in “keeping”, rerun the program using the ODS SELECT options as we did before, but this is going to get quite costly in terms of time if we have to run every program twice in order to get the output we want. It would be more convenient if we could run the procedure once, and then create the desired output from the results, in a fashion similar to PROC REPLAY in SAS/GRAPH. That is what the next section is about.

COLLECTING AND REPLAYING ODS ELEMENTS

Rather than outputting your output directly to an RTF destination, it may be more convenient to collect all the desired ODS elements, and then create the RTF file. There are two obvious advantages to this approach: 1) you don’t have to run the program twice in order to figure out the name of the elements and then SELECT them on the second run, and 2) you can rearrange the order of the elements irregardless of the SAS PROC that was used to create the element, and then create the report. Creating the store of elements involves routing the ODS output the DOCUMENT destination as shown below:

```
   LIBNAME sgf “C:\sgf\”;;
   OPTIONS NoCenter NoDate NoNumber LS=64;
   ODS LISTING;
   ODS Trace On;
   ODS Graphics On;
   ODS NoPtitle;

   ODS DOCUMENT Name=SML2Doc(Write);

   PROC REG Data=sgf.rawdata Plots(UnPack);
   MODEL time1 = time0;
   TITLE “Simple Linear Regression";
   QUIT;
   RUN;

   ODS DOCUMENT Close;
   ODS Trace Off;
   ODS Graphics Off;
   ODS LISTING Close;
```

In order to use the elements stored in the ODS DOCUMENT; you must know the PATH of each element. To list the elements in the ODS DOCUMENT; you need to use PROC DOCUMENT with the LIST statement:

```
   ODS LISTING;

   PROC DOCUMENT Name=SML2Doc;
   LIST / Levels=All;
   RUN;
   QUIT;
   RUN;

   ODS LISTING Close;
```

Of course, the program to list the elements could easily be combined with the program to write the elements. The list of elements is printed to the SAS listing in this case. The PATH in the listing file created by PROC DOCUMENT is very similar to the PATH for element in the SAS log created by the TRACE statement.

Once you know the DOCUMENT PATH for the elements that you want to output, you can use PROC DOCUMENT to REPLAY the chosen elements in the desired order:

```
ODS RTF Style=Journal StartPage=no
          file="C:\sgf\sample4.RTF";

   PROC DOCUMENT Name=SML2Doc;
   REPLAY
   \Reg#1\MODEL1#1\Fit#1\TIME1#1\
   ParameterEstimates#1,

   \Reg#1\MODEL1#1\ObswiseStats#1\TIME1#1\
   DiagnosticPlots#1\ResidualHistogram#1,

   \Reg#1\MODEL1#1\ObswiseStats#1\TIME1#1\
   DiagnosticPlots#1\Fit#1;

   RUN;
   QUIT;
   RUN;

   ODS RTF Close;
   RUN;
```

Note that the PATH for each element spans two lines, which works fine in SAS, and makes for a better-formatted display here.

OTHER ODS TRICKS

There are a couple other ODS tricks that would be of interest to the typical statistician using SAS. I must have written a hundred programs over the years that used PROC PRINTTO to write out a SAS listing file, then a SAS DATA step to read the listing file, just to get some portion of the output into a SAS data set for further processing. The ability to do it separated me from other statisticians casually using SAS, but alas, even that skill has become obsolete.

OUTPUTING TABLES TO SAS DATA SETS

Suppose we want to output the table in PROC REG that has the parameter estimates and the corresponding t-tests to a SAS data set. This is easily accomplished with the ODS OUTPUT destination. You have to know the name that ODS is using for the table, which probable means running the program twice, first with the TRACE on to learn the name, and then with the ODS OUTPUT statement to create the data set. Of course, if you use the same SAS procedures from day to day, you will remember the names of the various tables associated with your favorite SAS PROC. Here is the SAS code to output the table of parameter estimates to a new SAS data set called MyParmEst:

```
   OPTIONS NoCenter NoDate NoNumber LS=88;
   ODS LISTING;
   ODS Trace On;

   ODS OUTPUT ParameterEstimates=MyParmEst;

   PROC REG Data=sgf.rawdata;
   MODEL time1 = time0;
   TITLE “Simple Linear Regression";
   QUIT;
   RUN;

   PROC PRINT data=MyParmEst;
   RUN;

   ODS Trace Off;
   ODS LISTING Close;
```

The SAS listing created from PROC PRINT looks like this:

```
   Simple Linear Regression

   Obs Model  Dependent  Variable  DF   Estimate    StdErr  tValue   Probt
    1  MODEL1   TIME1    Intercept  1   66.80414  16.60590    4.02  0.0004
    2  MODEL1   TIME1    TIME0      1    0.73146   0.06611   11.06  <.0001
```

CREATING EXCEL DATA FILES FROM TABLES

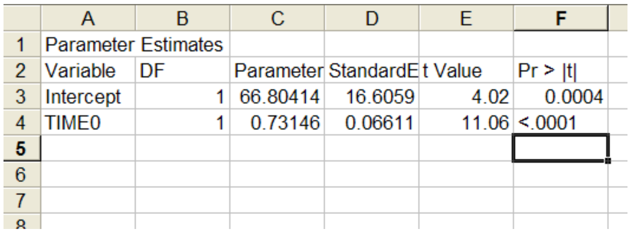
We can also create a Microsoft Excel spreadsheet directly from SAS tables using the ODS CSV statement. You could output the complete results of a PROC, but normally this is used in conjunction with the ODS SELECT statement to output a particular table for additional processing in Excel. Here is an example of how to output my favorite table, the ParameterEstimates table from PROC REG:

```
OPTIONS NoCenter NoDate NoNumber LS=64;
ODS LISTING;
ODS Trace On;
ODS CSV file="C:\sgf\sample5.csv";

ODS Select ParameterEstimates;
PROC REG Data=sgf.rawdata;
MODEL time1 = time0;
TITLE “Simple Linear Regression";
QUIT;
RUN;

ODS Trace Off;
ODS CSV Close;
ODS LISTING Close;
```

This is what the corresponding Excel spreadsheet looks like:



	A	B	C	D	E	F
1	Parameter Estimates					
2	Variable	DF	Parameter	StandardE	t Value	Pr > |t|
3	Intercept	1	66.80414	16.6059	4.02	0.0004
4	TIME0	1	0.73146	0.06611	11.06	<.0001
5						
6						
7						
8						

If you are interested in specifying cell formats, row heights or column widths, borders, alignments, or patterns, take a look at “Generating Custom Excel Spreadsheets using ODS” by Chevell Parker.

CONCLUSION

ODS is far more important, and useful, to the typical statistician than I imagined when I first heard of it. I thought ODS was about making colorful HTML output, most of which looked quite unbecoming to a stuffy old statistician such as myself. When I had a chance to attend SUGI 29, and was able to devote a little time to understanding ODS, I found that there were lots of things of interest to me. I have now made attending the SAS Global Forum, along with the NESUG Conference, regular stops on my conference tour each year.

Of course, I’ve only scratched the surface of what can be done with ODS. My attempt here was to gather the features of ODS that I’ve learned about that I think would be of particular interest to statisticians. I’ve intentionally stayed away from talking about modifying ODS templates for tables and graphs, which adds a great deal of flexibility to how elements are displayed, but are a whole other level of complexity.

I would contend that while those details may be the interest of almost every SAS Programmer, they are not the primary focus of most Statisticians using SAS. Although, there is that annoyance about how the residual graph was labeled that I could change by modifying the ODS template…. Just kidding, maybe I’ll tackle that next year…

REFERENCES:

Rodriguez, Robert N. 2004. “An Introduction to ODS for Statistical Graphics in SAS® 9.1”, *Proceedings of the Twenty-Ninth Annual SAS® Users Group International Conference,* SAS Institute, Cary, North Carolina, Paper 204-29.

Mitchell, Rick. 2004. “Thumbs Up for ODS Graphics, But Don’t Throw Out All Your SAS/GRAPH® Programs, *Proceedings of the Twenty-Ninth Annual SAS® Users Group International Conference,* SAS Institute, Cary, North Carolina, Paper 083-29.

Parker, Chevell. 2003. “Generating Custom Excel Spreadsheets using ODS”, *Proceedings of the Twenty-Eight Annual SAS® Users Group International Conference,* SAS Institute, Cary, North Carolina, Paper 12-28.

SAS Institute. 2004. SAS/STAT® 9.1 User’s Guide, SAS Institute, Cary, North Carolina.

CONTACT INFORMATION

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

Steven M. LaLonde, Ph.D.
John D. Hromi Center for Quality and Applied Statistics
Rochester Institute of Technology
98 Lomb Memorial Drive
Rochester, New York 14623-5604

Work Phone: (585) 475-5854
Fax: (585) 475-5959
Email: smleqa@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.