

CORR PROCEDURE AND POWER OF GTL

Mirjana Stojanovic, Cancer Center, Duke University Medical Center, Durham, NC
Laura Archer, Cancer Center, Duke University Medical Center, Durham, NC
Donna Hollis, Cancer Center, Duke University Medical Center, Durham, NC

ABSTRACT

This paper will demonstrate a new feature in SAS® 9.1.3 and detail an experimental procedure in ODS graphics with SAS/BASE which will demonstrate the future of graphics in SAS®.

Graphics are an essential part of statistical presentation and until now customized graphs were cumbersome to create using SAS/GRAPH. In SAS® 9.1 an experimental extension of the Output Delivery System (ODS) is being used with a CORR procedure in SAS/BASE to automatically create graphics with the same ease ODS provides in creating tables. In order to create statistical graphs using ODS include the statement ODS GRAPHICS ON before your procedure statements, this will invoke the facility. Graphics are then created automatically. ODS statistical graphics do not create output for the LISTING destination. You must send your output to other ODS destination (RTF, HTML, DOCUMENT), ODS Graphics use Java technology, and you do not need to learn SAS Graph to use this.

The paper will give step-by-step instructions on how to create and use the Output Delivery System (ODS) Graphics by demonstrating the production of statistical graphics, customization of template procedure, changing variable name of plot to variable label. As an example, we are using the CORR procedure from SAS/BASE. We will detail using ODS graphics in conjunction with this SAS/BASE procedure. Then we will give an overview of graph customization using the Graphical Template Language. Next we will show how we customize the layout and the appearance of the CORR procedure using ODS template.

INTRODUCTION

ODS Statistical Graphics, or ODS Graphics for short, is an experimental extension of the Output Delivery System (ODS) first introduced in SAS® 9.1.3. ODS graphics can be used with several SAS procedures, and will allow you to create statistical graphics automatically. Once graphics are created modifications can be made using PROC TEMPLATE. This paper will give a brief overview of ODS graphics, and the use of PROC TEMPLATE to modify graphs.

ODS GRAPHICS AND PROC CORR

Beginning in SAS 8.0 ODS was introduced to create and manage statistical tables, as well as store and display them in several formats including HTML and RTF. In SAS® 9.1.3 ODS was expanded to include a graphical component which will allow you to create and manage statistical graphics in much the same way ODS creates and manages tables. Like ODS, ODS Graphics incorporates procedure options (and defaults) to specify what output will be created. Other similarities between ODS and ODS Graphics are that both allow for the specification of an ODS destination, and the use of the ODS select and ODS exclude statements to select or exclude graphs from the output. Many other similarities exist and users will find that using ODS Graphics can be done with the same ease and control as ODS.

To invoke ODS Graphics include the code ODS GRAPHICS ON before your PROC statement. ODS Graphics does not support the LISTING destinations so, prior to the ODS GRAPHICS ON statement, an ODS destination should be specified, for example:

```
ODS HTML;  
ODS GRAPHICS ON;
```

Here ODS HTML specifies that the output should be stored in the HTML destination, and ODS GRAPHICS ON request ODS graphics be added to the tabular output.

The following table lists the output object graph names which Base SAS procedures produce. The table provides the name of each graph, a description of what the graph contains, and the option, if any, that creates the output object graph.

ODS STATISTICAL GRAPH NAMES PRODUCED BY THE CORR PROCEDURE

For detail information about the CORR procedure, see Base SAS Procedures Guide, Volume 3.		
Graph Name	Description	Option
ScatterPlot	Scatter plot	PLOTS=SCATTER
SymMatrixPlot	Symmetric scatter plot matrix	PLOTS=MATRIX
ODS Graphs Created by the WITH Statement		
RecMatrixPlot	Rectangular scatter plot matrix	PLOTS=MATRIX

CUSTOMIZING ODS GRAPHICS

When modifications become necessary, you can customize a particular graph by changing its template definition. You can access ODS statistical graph definitions and modify them with the TEMPLATE procedure.

ODS statistical graph definitions are written in an experimental graph template language, which has been added to PROC TEMPLATE in SAS® 9.1.3.

A *template definition* is a set of SAS statements that can be run with the TEMPLATE procedure to create a compiled template. Two common types of template definitions are *table definitions* and *style definitions*. A table definition describes how to display the output for an output object that is to be rendered as a table, and a style definition provides formatting information for specific visual aspects of your SAS output. A third type of template definition is a *graph template definition* (or *graph definition* for short), which controls the layout and details of graphs produced with ODS Graphics. Graph definitions begin with a DEFINE STATGRAPH statement and end with an END statement.

In common applications of ODS Graphics, it should not be necessary to modify the default template for each graph, which is supplied by SAS. However, when customization is necessary, you can modify the default template with the graph template language in the TEMPLATE procedure

The following statements invoke the CORR procedure and request a correlation analysis:

```
ods html;  
ods graphics on;  
proc corr data=score_test plots;  
run;  
ods graphics off;  
ods html close;
```

The experimental PLOTS option display a symmetric matrix plot for the analysis variables.

LOCATING TEMPLATES

The first step in customizing a graph is to determine which template was used to create the original graph. The simple way to do this is to specify the ODS TRACE ON statement prior to the procedure statements that created the graph. The qualified template name is displayed in the SAS log, or it can be displayed in text file specified in the FILE=option.

```

ods html;
ods trace on / label;
ods graphics on;
proc corr data=db.score_test plots=scatter;
run;
ods trace off;
ods graphics off;
ods html close;

NOTE: Writing HTML Body file: sashtml.htm
54 ods trace on / label;
55 ods graphics on;
NOTE: ODS Statistical Graphics will require a SAS/GRAPH license when it is declared production.
56 proc corr data=db.score_test plots=scatter;
57 run;

```

Output Added:

```

-----
Name:      VarInformation
Label:     Variables Information
Template:  base.corr.VarInfo
Path:      Corr.VarInformation
Label Path: 'The Corr Procedure'. 'Variables Information'
-----

```

Output Added:

```

-----
Name:      SimpleStats
Label:     Simple Statistics
Template:  base.corr.UniStat
Path:      Corr.SimpleStats
Label Path: 'The Corr Procedure'. 'Simple Statistics'
-----

```

Output Added:

```

-----
Name:      PearsonCorr
Label:     Pearson Correlations
Template:  base.corr.StackedMatrix
Path:      Corr.PearsonCorr
Label Path: 'The Corr Procedure'. 'Pearson Correlations'
-----

```

WARNING: Statistical graphics displays created with ODS are experimental in this release.

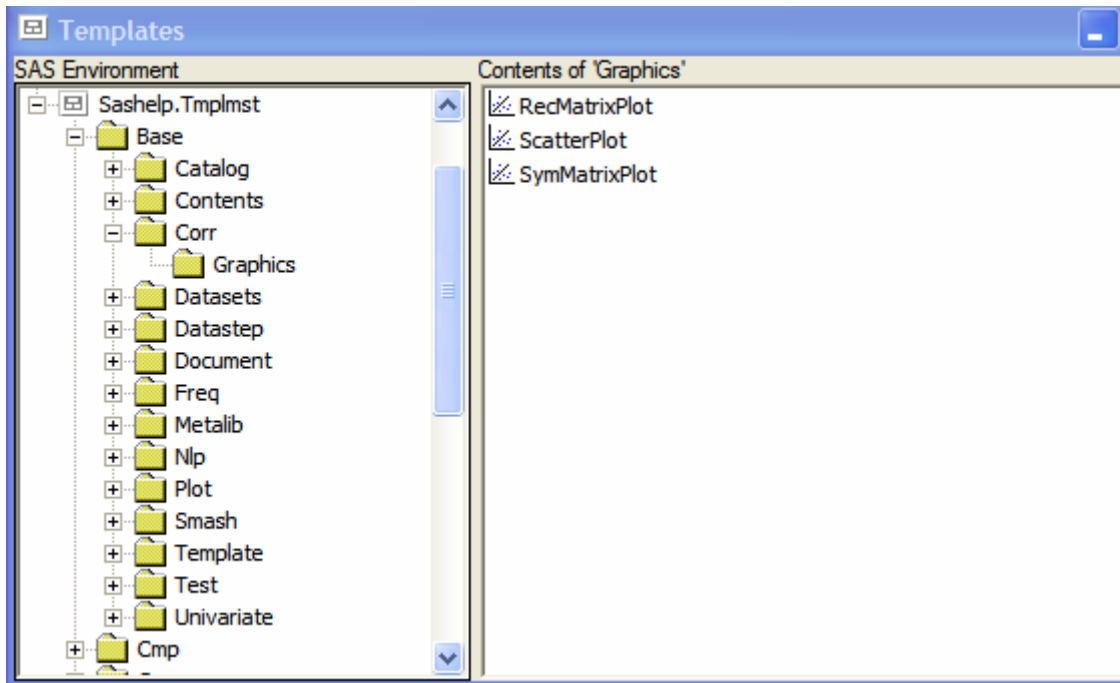
Output Added:

```

-----
Name:      ScatterPlot
Template:  Base.Corr.Graphics.ScatterPlot
Path:      Corr.ScatterPlot
Label Path: 'The Corr Procedure'. 'Corr'
-----

```

From this output we see that the graphic we are interested in is called ScatterPlot and the template, ScatterPlot are stored in the path Base.Corr.Graphics.ScatterPlot. Now that we know which template is being used need to find the code for that template. Source code can be found in three ways. The first way is to right click on the result viewer (on the far left) then scroll down to templates, then select the location of the template (if you are customizing a SAS defined default this will be in the location SASHELP.TMPLST). Next follow the drop down menus to find the procedure you are using. To find CORR you would select BASE, then CORR, then Graphics. Finally double click on the template of interest.



The second way to get to the directory of templates is to type ODSTEMPLATES in the SAS command window. This will bring up an explorer window and you will need to follow the steps above to find the template of interest. The third way is to use the source statement in the TEMPLATE procedure.

Once a template has been written, or modified a new compiled template will need to be saved, before it can be applied to your graphics. ODS automatically saves the compiled template in the first template store that it can update, according to the currently defined ODS path. If you have not changed the ODS path, then the modified template is saved in the Sasuser.Templat template store.

THE CORR PROCEDURE

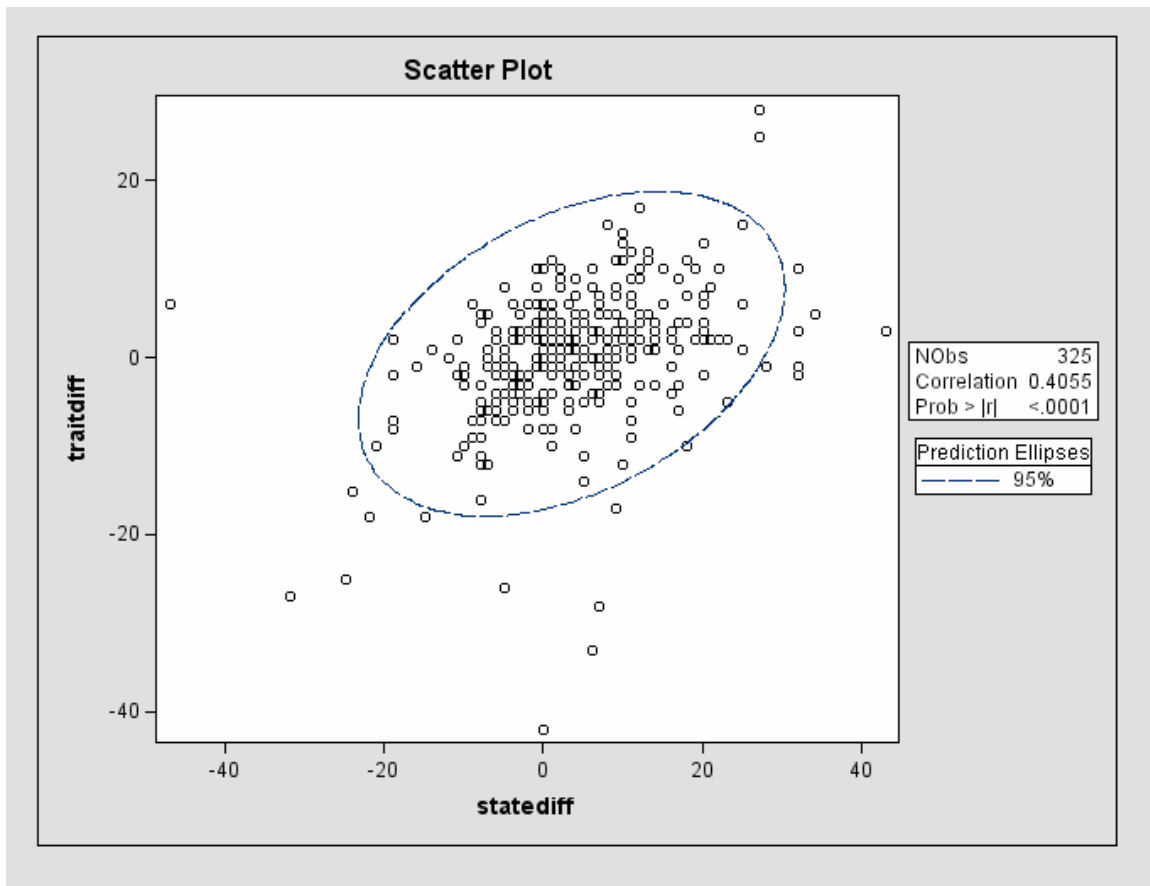
Before we proceed with our example, let's examine some of the new features of proc CORR. The CORR procedure has the following new features in SAS® 9.1.3

- The FISHER option in the PROC CORR statement requests confidence limits and **p**-values for Pearson and Spearman correlation coefficients based on Fisher's **z** transformation. Using the FISHER option, you can specify an alpha value and a null hypothesis value. You can also specify the type of confidence limit (upper, lower, or two-sided) and whether the bias adjustment should be used for the confidence limits.
- The PLOTS=MATRIX option in the PROC CORR statement uses ODS graphics to produce either a rectangular matrix plot (if you also specify a WITH statement) or a symmetric matrix plot (if you do not specify a WITH statement) for variables.
- The PLOTS=SCATTER option in the PROC CORR statement uses ODS graphics to produce scatter plots for variables. By default, the scatter plot also includes a 95% prediction ellipse. You can use the ELLIPSE= option with the PLOTS=SCATTER option to include prediction ellipses for new observations, confidence ellipses for the mean, or no ellipses.

Here we are particularly interested in the third new feature.

We used graph template language syntax to create a modified template.

DEFAULT PLOT AND TEMPLATE



```
proc template;
  define statgraph Base.Corr.Graphics.ScatterPlot;
    notes "Scatter Plot";
    dynamic _Title _Inset _Legend _LegendTitle _Nobs _Corr _PValue _Slope
      _XMean _YMean _Major1 _Minor1 _Level1 _Major2 _Minor2 _Level2 _Major3
      _Minor3 _Level3 _Major4 _Minor4 _Level4 _Major5 _Minor5 _Level5;
    layout lattice / rows=1 columns=1;
    sidebar / align=top;
      layout overlay / padbottom=5;
        entrytitle _TITLE;
      endlayout;
    endsidebar;
    layout overlay;
      ScatterPlot y=WVAR x=VVAR / markersymbol=
        GraphDataDefault:markersymbol markercolor=
        GraphDataDefault:contrastcolor markersize=
        GraphDataDefault:markersize tip=( OBSERVATION WVAR VVAR );
      EllipseParm semimajor=_MAJOR1 semiminor=_MINOR1 slope=_SLOPE /
        origins=_XMEAN yorigin=_YMEAN name="ellipse1" LegendLabel=
        _LEVEL1 linecolor=StatGraphConfidence:contrastcolor linepattern
        =dash;
      EllipseParm semimajor=_MAJOR2 semiminor=_MINOR2 slope=_SLOPE /
        xorigin=_XMEAN yorigin=_YMEAN name="ellipse2" LegendLabel=
        _LEVEL2 linecolor=StatGraphConfidence:contrastcolor linepattern
        =dashdot;
      EllipseParm semimajor=_MAJOR3 semiminor=_MINOR3 slope=_SLOPE /
        xorigin=_XMEAN yorigin=_YMEAN name="ellipse3" LegendLabel=
```

```

        _LEVEL3 linecolor=StatGraphConfidence:contrastcolor linepattern
        =dashshort;
    EllipseParm semimajor=_MAJOR4 semiminor=_MINOR4 slope=_SLOPE /
    xorigin=_XMEAN yorigin=_YMEAN name="ellipse4" LegendLabel=
    _LEVEL4 linecolor=StatGraphConfidence:contrastcolor linepattern
    =dotdash;
    EllipseParm semimajor=_MAJOR5 semiminor=_MINOR5 slope=_SLOPE /
    xorigin=_XMEAN yorigin=_YMEAN name="ellipse5" LegendLabel=
    _LEVEL5 linecolor=StatGraphConfidence:contrastcolor linepattern
    =dot;
endlayout;
row2header;
    layout Gridded / padleft=5;
        if (_INSET)
            layout gridded / columns=2 valign=center border=true
            background=GraphWalls:background;
            entry _NOBS / halign=right valign=top format=best6.;
            entry "Correlation" / halign=left valign=top;
            entry _CORR / halign=right valign=top format=best6.;
            entry "Prob > |r|" / halign=left valign=top;
            entry _PVALUE / halign=right valign=top format=pvalue6.4;
        endlayout;
        endif;
        if (_LEGEND)
            layout Gridded / padtop=10;
            DiscreteLegend "ellipse1" "ellipse2" "ellipse3" "ellipse4"
            "ellipse5" / down=5 border=true titleborder=true title=
            _LEGENDTITLE background=GraphWalls:background;
        EndLayout;
        endif;
    EndLayout;
endrow2header;
endlayout;
end;
run;

```

MODIFIED PLOT AND TEMPLATE

To make STATGRAPH templates more flexible you can introduce dynamics or macro variables.

We can add our own macro variables with an MVAR statement in the template. Declared MVARs are resolved when the template is executed:

```
%macro better_plot ;
```

```

proc template;
    define statgraph Base.Corr.Graphics.ScatterPlot;
        notes "Scatter Plot";

        /* add macro variables for axis labels */
        mvar XLABEL YLABEL;

dynamic _Title _Inset _Legend _LegendTitle _Nobs _Corr _PValue _Slope
_XMean _YMean _Major1 _Minor1 _Level1 _Major2 _Minor2 _Level2 _Major3
_Minor3 _Level3 _Major4 _Minor4 _Level4 _Major5 _Minor5 _Level5;
        layout lattice / rows=1 columns=1;
            sidebar / align=top;
                layout overlay / padbottom=5;
                    entrytitle "&mytitle." ;
                endlayout;
            endsidebar;
    enddefine;

```



```
/* add axis options to relabel the X and Y axes */
layout overlay / xaxisopts=(label=XLABEL) yaxisopts=(label=YLABEL);
```

```
ScatterPlot y=WVAR x=WVAR / markersymbol=
GraphDataDefault:markersymbol markercolor=
GraphDataDefault:contrastcolor markersize=
GraphDataDefault:markersize tip=( OBSERVATION WVAR WVAR );
EllipseParm semimajor=_MAJOR1 semiminor=_MINOR1 slope=_SLOPE /
  xorigin=_XMEAN yorigin=_YMEAN name="ellipse1" LegendLabel=
  _LEVEL1 linecolor=StatGraphConfidence:contrastcolor linepattern
    =dash;
EllipseParm semimajor=_MAJOR2 semiminor=_MINOR2 slope=_SLOPE /
  xorigin=_XMEAN yorigin=_YMEAN name="ellipse2" LegendLabel=
  _LEVEL2 linecolor=StatGraphConfidence:contrastcolor linepattern
    =dashdot;
EllipseParm semimajor=_MAJOR3 semiminor=_MINOR3 slope=_SLOPE /
  xorigin=_XMEAN yorigin=_YMEAN name="ellipse3" LegendLabel=
  _LEVEL3 linecolor=StatGraphConfidence:contrastcolor linepattern
    =dashshort;
EllipseParm semimajor=_MAJOR4 semiminor=_MINOR4 slope=_SLOPE /
  xorigin=_XMEAN yorigin=_YMEAN name="ellipse4" LegendLabel=
  _LEVEL4 linecolor=StatGraphConfidence:contrastcolor linepattern
    =dotdash;
EllipseParm semimajor=_MAJOR5 semiminor=_MINOR5 slope=_SLOPE /
  xorigin=_XMEAN yorigin=_YMEAN name="ellipse5" LegendLabel=
  _LEVEL5 linecolor=StatGraphConfidence:contrastcolor linepattern
    =dot;
```

```

endlayout;
row2header;
layout Gridded / padleft=5;
  if (_INSET)
    layout gridded / columns=2 valign=center border=true
      background=GraphWalls:background;
      entry "NObs" / halign=left valign=top;
      entry _NOBS / halign=right valign=top format=best6.;
      entry "Correlation" / halign=left valign=top;
      entry _CORR / halign=right valign=top format=best6.;
      entry "Prob > |r|" / halign=left valign=top;
      entry _PVALUE / halign=right valign=top format=pvalue6.4;
    endlayout;
  endif;
  if (_LEGEND)
    layout Gridded / padtop=10;
    DiscreteLegend "ellipse1" "ellipse2" "ellipse3" "ellipse4"
      "ellipse5" / down=5 border=true titleborder=true title=
      _LEGENDTITLE background=GraphWalls:background;
  EndLayout;
  endif;
EndLayout;
endrow2header;
endlayout;
end;
run;

```

Now simply initialize the XLABEL and YLABEL macro variables before you run PROC CORR.

%mend better_plot ;

```

ods html;
ods trace on / label;
ods graphics on;
%let XLABEL="Difference in State Score";
%let YLABEL="Difference in Trait Score";
%let mytitle=Differences in state versus trait scores;

```

%better_plot

```

proc corr data=db.score_test plots=scatter;
var StateDiff TraitDiff ;
run;
ods trace off;
ods graphics off;
ods html close;

```

CONCLUSION:

Since often the default plotting output of common SAS procedures is not easily cut and pasted into reports to be shared with investigators, the advent of ODS Graphics has made plot sharing much easier and quicker. With simple template modifications and the creation of macro variables, producing graphics suitable for quick sharing has made using SAS a more complete communication product. It is a valuable addition which has increased efficiency for our statistical group.

REFERENCES:

<http://support.sas.com/rnd/base/topics/statgraph/>
<http://support.sas.com/rnd/base/topics/statgraph/proctemplate/a002774699.htm>
<http://support.sas.com/rnd/base/topics/statgraph/proctemplate/a002787384.htm>
<http://support.sas.com/rnd/base/topics/statgraph/proctemplate/a002807821.htm>
<http://support.sas.com/rnd/base/topics/statgraph/v91StatGraphStyles.htm>
<http://www2.sas.com/proceedings/sugi30/239-30.pdf>
<http://www2.sas.com/proceedings/sugi31/112-31.pdf>
http://www.stattutorials.com/SAS/TUT_PROCCORR.pdf
<http://www2.sas.com/proceedings/sugi26/p172-26.pdf>
<http://support.sas.com/rnd/datavisualization/papers/sugi30/GraphStyles.pdf>
<http://support.sas.com/rnd/datavisualization/papers/sugi30/GTL.pdf>
<http://www2.sas.com/proceedings/sugi29/204-29.pdf>
<http://www2.sas.com/proceedings/sugi30/094-30.pdf>
<http://support.sas.com/rnd/base/topics/statgraph/>
<http://support.sas.com/rnd/app/da/stat/odsgraph/index.html>
<http://support.sas.com/rnd/base/topics/statgraph/proctemplate/a002787405.htm>

More references on SAS Support web site.

[Examples](#)

[SAS/STAT User's Guide](#)

[Displaying Templates](#)

[SAS/STAT User's Guide](#)

[Example 15.9: Swapping the Axes in a Graph](#)

[SAS/STAT User's Guide](#)

[Example 15.10: Modifying Tick Marks and Superscripts](#)

[SAS/STAT User's Guide](#)

[Reverting to Default Templates](#)

[SAS/STAT User's Guide](#)

[Saving Customized Templates](#)

[SAS/STAT User's Guide](#)

[Editing Templates](#)

[SAS/STAT User's Guide](#)

[Example 15.8: Modifying Colors, Line Styles, and Symbols](#)

[SAS/STAT User's Guide](#)

[Example 15.7: Customizing Graph Titles and Axis Labels](#)

[SAS/STAT User's Guide](#)

[Summary of Options](#)

[SAS/QC User's Guide](#)

[Getting Started](#)

Example 15.7: Customizing Graph Titles and Axes Labels

This example shows how to use PROC TEMPLATE to customize the appearance and content of an ODS graph. It illustrates the discussion in the section "[Customizing Graphics with Templates](#)" in the context of changing the default title and y-axis label for a Q-Q plot created with the ROBUSTREG procedure.

The following statements request a Q-Q plot for robust residuals using PROC ROBUSTREG with the stack data from [Example 15.4](#).

```
ods trace on;
ods html;
ods graphics on;

ods select ResidualQQPlot;

proc robustreg plot=resqqplot data=stack;
  model y = x1 x2 x3;
run;
```

<http://post.queensu.ca:8080/SASDoc/getDoc/en/odsug.hlp/odsgraphicsstmt.htm>
<http://www2.sas.com/proceedings/sugi30/183-30.pdf>
<http://support.sas.com/faq/045/FAQ04553.html>
<http://support.sas.com/rnd/base/topics/statgraph/proctemplate/a002774501.htm>

ACKNOWLEDGMENTS:

We would like to thank you Mr. Jeff Cartier, SAS Institute, Cary, NC for his advice, support and efforts to help in the creation of this paper.

CONTACT INFORMATION:

Your comments are greatly appreciated and encouraged.

Contact the author at:

Mirjana Stojanovic
Duke University Medical Center
(919)668-9337(phone) (919)
E-mail: mirjana.stojanovic@duke.edu.

Laura Archer
Duke University Medical Center
E-mail laura.archer@duke.edu

Donna Hollis
Duke University Medical Center
(919) 681-5027(phone)
e-mail: donna.hollis@duke.edu

TRADEMARK INFORMATION

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.