081-2012

Customizing ODS Graphical Output for SAS/STAT® Procedures

Yang Xiao, University of Cincinnati, Cincinnati, OH
Xiangxiang Meng, SAS Institute Inc., Cary, NC

ABSTRACT

Many SAS/STAT® procedures can generate graphs based on ODS Graphics. However, in these statistical procedures there are few options available for the image configuration of the ODS graphs. In this paper, we discuss three ways to customize the ODS graphical outputs from a statistical procedure including 1) using the built-in PLOTS= option in the procedure statement; 2) modifying the Graphical Template Language (GTL) template used by the SAS/STAT procedures; and 3) using ODS style template. The odds ratio plots and the ROC curves produced by the LOGISTIC procedure are used throughout the examples to introduce these three methods step by step.

INTRODUCTION

Many SAS/STAT procedures can generate graphs based on ODS Graphics. However, in these statistical procedures there are few options available for the image configuration of the ODS graphs. In this paper, we discuss three ways to customize the ODS graphical outputs from a statistical procedure including 1) using the built-in PLOTS= option in the procedure statement; 2) modifying the Graphical Template Language (GTL) template used by the SAS/STAT procedures; and 3) using ODS style template. The odds ratio plots and the ROC curves produced by the LOGISTIC procedure are used throughout the examples to introduce these three methods step by step.

We use the data set *Neuralgia* extracted from the manual of PROC LOGISTICS. The data set contains five variables: treatment, sex, age, duration and pain. The binary variable *Pain* (YES/NO) is used as the response variable in the logistic regressions in this paper. The variable *Treatment* is a categorical variable with three levels: A, B, P representing two new drugs and the placebo. The variable *Age* is recorded in years. The variable *Duration* (duration of pain) is not in use.

```
Data Neuralgia;
   input Treatment $ Sex $ Age Duration Pain $ @@;
   datalines;
     P  F  68   1  No   B  M  74  16  No  P  F  67  30  No
     P  M  66  26  Yes  B  F  67  28  No  B  F  77  16  No
     A  F  71  12  No   B  F  72  50  No  B  F  76   9  Yes
     A  M  71  17  Yes  A  F  63  27  No  A  F  69  18  Yes
     B  F  66  12  No   A  M  62  42  No  P  F  64   1  Yes
     A  F  64  17  No   P  M  74   4  No  A  F  72  25  No
     P  M  70   1  Yes  B  M  66  19  No  B  M  59  29  No
     A  F  64  30  No   A  M  70  28  No  A  M  69   1  No
     B  F  78   1  No   P  M  83   1  Yes B  F  69  42  No
     B  M  75  30  Yes  P  M  77  29  Yes P  F  79  20  Yes
     A  M  70  12  No   A  F  69  12  No  B  F  65  14  No
     B  M  70   1  No   B  M  67  23  No  A  M  76  25  Yes
     P  M  78  12  Yes  B  M  77   1  Yes B  F  69  24  No
     P  M  66   4  Yes  P  F  65  29  No  P  M  60  26  Yes
     A  M  78  15  Yes  B  M  75  21  Yes A  F  67  11  No
     P  F  72  27  No   P  F  70  13  Yes A  M  75   6  Yes
     B  F  65   7  No   P  F  68  27  Yes P  M  68  11  Yes
     P  M  67  17  Yes  B  M  70  22  No  A  M  65  15  No
     P  F  67   1  Yes  A  M  67  10  No  P  F  72  11  Yes
     A  F  74   1  No   B  M  80  21  Yes A  F  69   3  No
  ;
run;
```

SAMPLE ODS GRAPHICS FROM SAS/STAT PROCEDURES

Many procedures in SAS/STAT support ODS graphics. With ODS graphics turned on, the statistical procedures can automatically generate statistical graphs and deliver them together with the output tables to an ODS (Output Delivery System) destination such as PDF and HTML. Many procedures also support the PLOTS= procedure option for users to select the type of graphs to produce. The following two examples show how to generate odds ratio plots and ROC curves in the LOGISTIC procedure using the neuralgia data set.

```
ods html style=htmlblue;
ods graphics on;
* [Example 1] ODDS ratio plots generated by the LOGISTIC procedure;
* Fit a model with interaction effects between treatment, age, and sex;
* Plot the odds ratio for age at each level of treatment and sex;
proc logistic data=neuralgia plots(only)=oddsratio;
  class treatment sex;
  model pain= treatment age sex treatment*age sex*age treatment*sex;
  oddsratio age / at(treatment='A' 'B' 'P' sex='M' 'F');
run;

* [Example 2] ROC curves generated by the LOGISTIC procedures;
* The NOFIT option suppresses fitting the model in the MODEL statement;
* Instead, fit three different models in the ROC statements;
proc logistic data=neuralgia plots(only)=roc rocoptions(nodetails out=dat_roc);
  class treatment sex;
  model pain(event='Yes') =treatment age sex treatment*age sex*age treatment*sex
       / nofit;
  roc 'Treatment'     treatment;
  roc 'Treatment|Age' treatment sex treatment*sex;
  roc 'Treatment|Age' treatment age treatment*age;
run; quit;
```

In both examples, the ONLY sub-options for PLOTS= suppress the output of the other default ODS graphs and only generate the user- specified graphs (odds ratios or ROC curves). The ROCOPTIONS in the second example further suppress the output of the ROC tables (NODETAILS) but output a table (OUT=DAT_ROC) which contains the sensitivity and specificity values computed for each ROC curve. The ODS graphical outputs are displayed in Figure 1 and Figure 2.

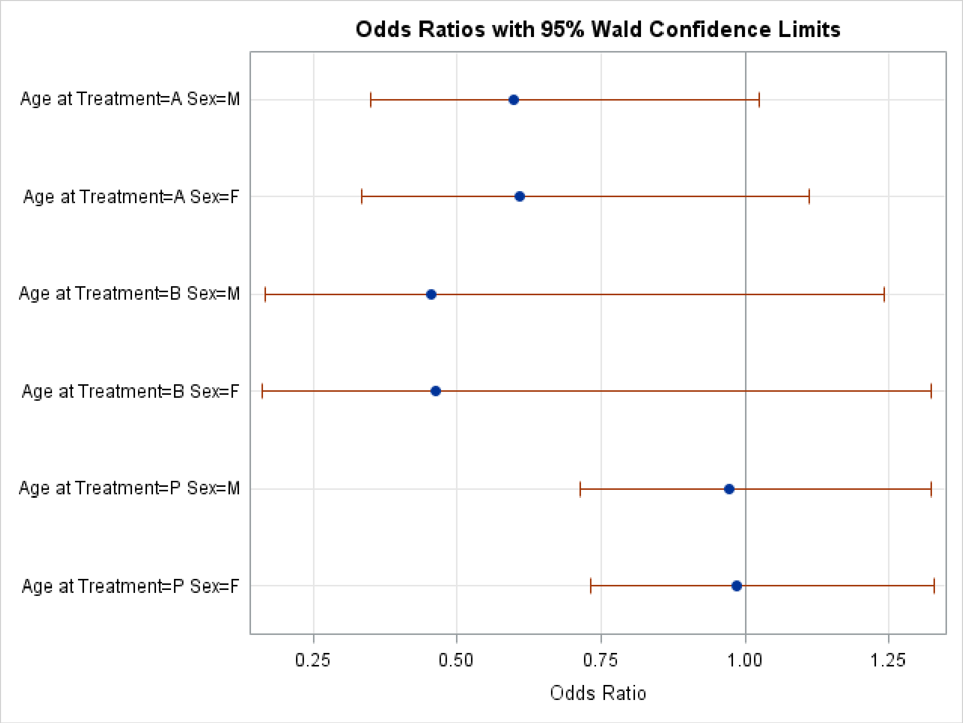

Figure 1. The default odds ratio plot produced by the logistic procedure (Example 1)

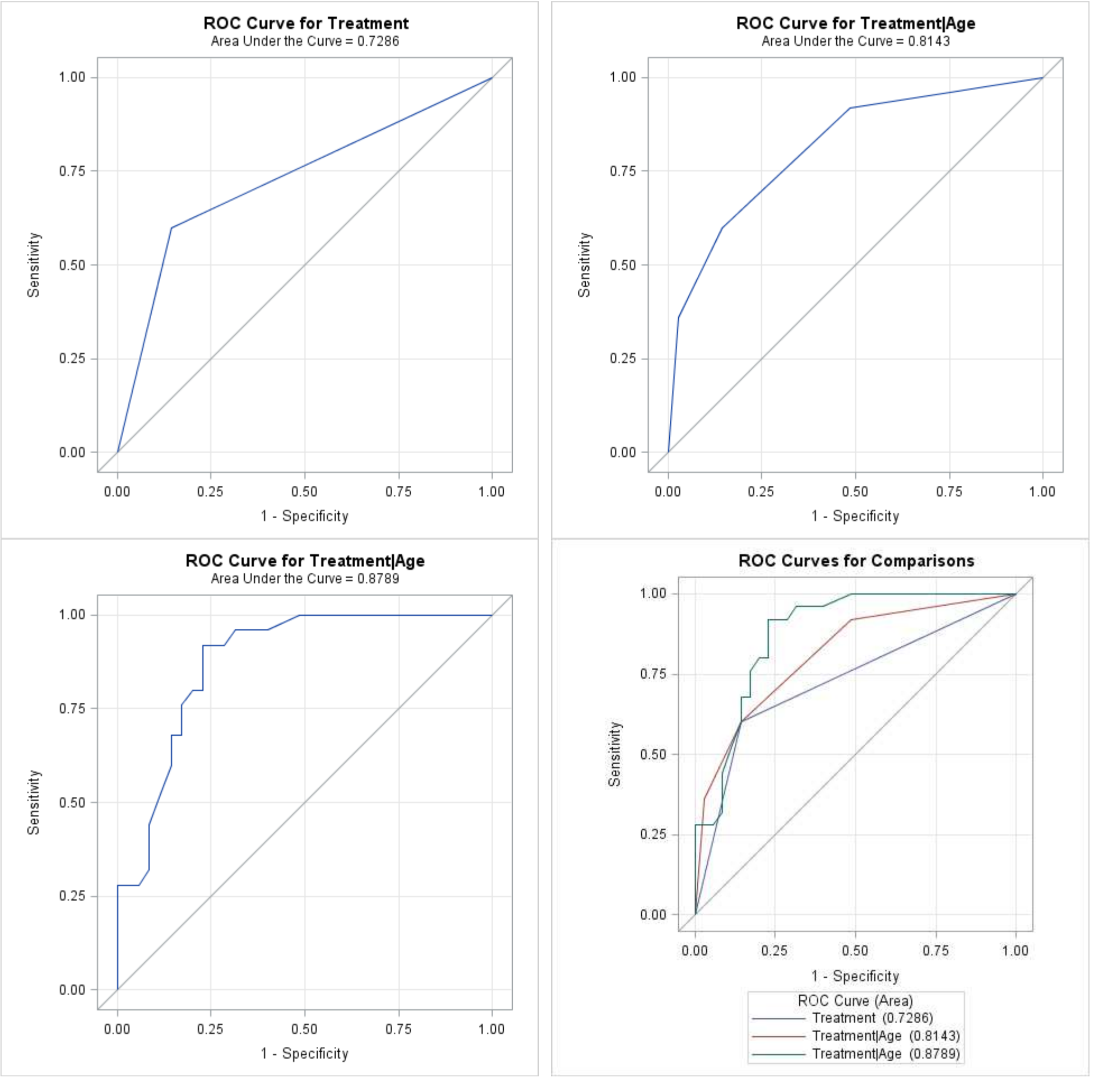

Figure 2. The default ROC curves produced by the logistic procedure (Example 2)

CUSTOMIZING THE ODS GRAPHS WITH THE PLOTS OPTION

A common PLOTS= option is avaialbe in the procedure statement of each SAS/STAT procedure to support ODS Graphics. This option can control the types of graphs to produce (*Plot Request*); assign global plot attributes to all plot requests, and assign individual plot attributes to each plot request. Example 3 uses four plot attributes to customize the odds ratio plot in example 1.

```
* [Example 3] Customized odds ratio plot with the PLOTS= option;
proc logistic data=neuralgia
  plots(only)=oddsratio(cLDisplay = line
                        order     = ascending
                        type      = horizontalstat
                        range     = (0,2)           );
  class treatment sex;
  model pain= treatment age sex treatment*age sex*age treatment*sex;
  oddsratio age / at(treatment='A' 'B' 'P' sex='M' 'F');
run;
```

In this example, four plot attributes are modified for the “ODDSRATIO” plot request: the style of the confidence limit error bars (CLDisplay=line), the sorting order of the odds ratios (order=ascending), the look of the graph (type=horizontalstat), and the range of the ratios (range=(0,2)). The graph is displayed in Figure 3.

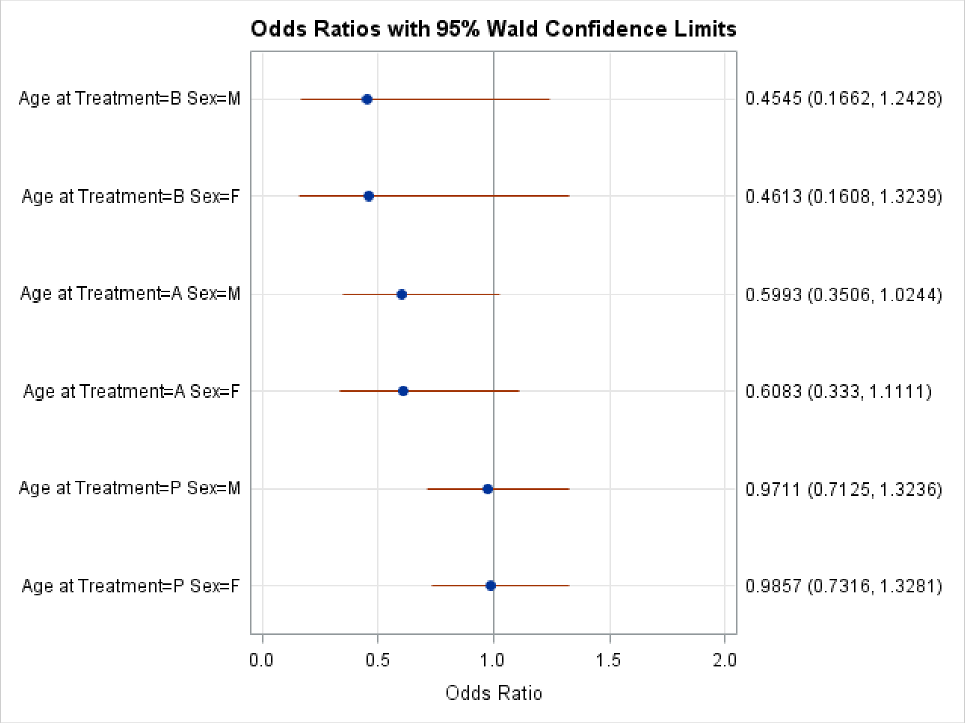

Figure 3. Using PLOTS= option to customize odds ratio plot (Example 3)

THE TEMPLATES BEHIND THE GRAPHS

In many situations such as presentations and publications, using the PLOTS= option are far less than enough to meet the on-demand graphical specifications. To make further modification to the graph structure and elements, we need to know how the ODS graphs are being produced behind the procedure. Different from traditional SAS/GRAPH outputs from the statistical procedures, each ODS graph is based on an ODS Graphics template written using the Graphical Template Language (GTL). To identify which templates are being used for the graphs, ODS TRACE needs to be turn on:

```
ods trace on;
  <codes from example 2>
ods trace off;
```

The templates in use are shown in the LOG window:

```
Output Added:
-------------
Name:       ROCCurve
Label:      ROC Curve
Template:   Stat.Logistic.Graphics.ROC
Path:       Logistic.ROC3.ROCCurve
-------------

Output Added:
-------------
Name:       ROCOverlay
Label:      ROC Curves
Template:   Stat.Logistic.Graphics.ROCOverlay
Path:       Logistic.ROCComparisons.ROCOverlay
-------------
```

Using the TEMPLATE procedure to show the contents of templates:

```
proc template;
  source Stat.Logistic.Graphics.ROC;
  source Stat.Logistic.Graphics.ROCOverlay;
run; quit;
```

The source codes of these two templates are shown in the Appendix .

CREATING NEW GTL TEMPLATE FOR SAS/STAT PROCEDURES

Since the ODS graphs are produced based on templates, we can create a new template upon the default template and then let the procedure use the new template to generate a customized graph. For instance, running the following template procedure will create a simplified version of the Stat.Logistic.Graphics.ROC template (Appendix A) with titles and tooltips being turned off.

```
* [Example 4] Running Proc Logistic with modified graph template;
proc template;
  define statgraph Stat.Logistic.Graphics.Roc;
    dynamic _cValue _TITLE _ROCID;
    BeginGraph / designwidth=defaultDesignHeight;
      layout overlayequated
        / equatetype=square
          yaxisopts=(gridDisplay=auto_on offsetmin=0.05 offsetmax=0.05
                     label="Sensitivity" shortlabel="TPR")
          xaxisopts=(gridDisplay=auto_on offsetmin=0.05 offsetmax=0.05
                     label="1 - Specificity" shortlabel="FPR")
          commonaxisopts=(tickvaluelist=(0 .25 .5 .75 1) viewmin=0 viewmax=1)
      ;
        lineparm   x=0 y=0 slope=1       / extend=true lineattrs=GRAPHREFERENCE;
        seriesplot y=_SENSIT_ x=_1MSPEC_ / primary=true connectorder=xaxis;
      endlayout;
    EndGraph;
  end;
run;

proc logistic data=neuralgia plots(only)=roc rocoptions(nodetails out=dat_roc);
  class treatment sex;
  model pain(event='Yes') =treatment age sex treatment*age sex*age treatment*sex
       / nofit;
  roc 'Treatment'     treatment;
  roc 'Treatment|Age' treatment sex treatment*sex;
  roc 'Treatment|Age' treatment age treatment*age;
run; quit;
```

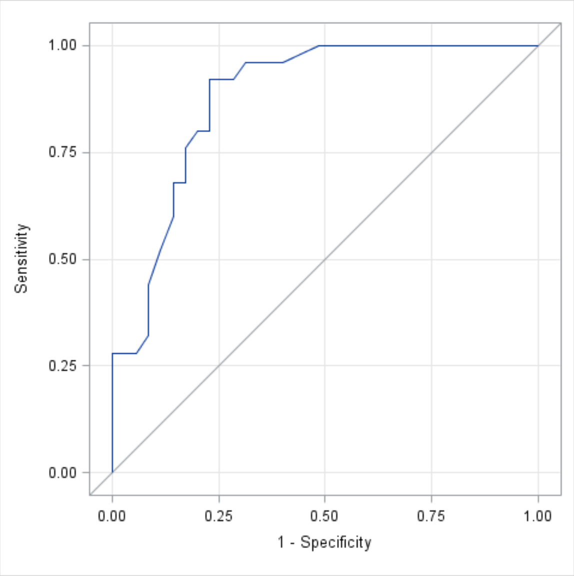

Figure 4. The ROC curve (Treatment|Age) produced by the new GTL template (Example 4)

With the default template storage paths in SAS,

1. SASUSER.TEMPLAT(UPDATE)
2. SASHELP.TMPLMST(READ)

The template created by PROC TEMPLATE is saved in the path SASUSER.TEMPLAT while the default template (as named Stat.Logistic.Graphics.ROC) is saved in the path SASHELP.TMPLMST. The first path has higher priority and SAS will search the first path first and find the modified Stat.Logistic.Graphics.ROC template to create the ROC charts. Thus, ROC charts without graph titles and tooltips are created in this example (Figure 4). To retrieve the default SAS setting, simply delete the template in SASUSER.TEMPLAT:

```
proc template;
    delete Stat.Logistic.Graphics.ROC;
run; quit;
```

Due to the path priority, the template in SASUSER.TEMPLAT will be deleted.

USING GTL AND ODS STYLE TEMPLATES TOGETHER

Besides the GTL template, the ODS style template affect the ODS Graphics outputs in SAS/STAT procedures. The GTL template defines the basic graphical structure of a ODS graph, and then the ODS style masks the graphical output with further specifications on the attributes of the elements within the graph, such as fonts, lines, markers, fills, and axes. The following example illustrates how to use GTL template and ODS style template together to produce presentation- or publication-ready images directly from running the PROC LOGISTICS.

Based on the template in example 4, this example first creates a GTL template with further modifications on the ROC curves produced by PROC LOGISTIC, including

1. adding a scatter plot of the sensitivity and specificity values;
2. using the number of Correctly Predicted Events as data labels;
3. filling the area under curve (AUC);
4. changing the reference line into a dash line; and
5. changing the axis options (grids, offsets, and ticks).

And a new style template is also defined based on the HTMLBLUE template:

```
* [Example 5] Using GTL and ODS style templates;
proc template;
  /* the GTL template */
  define statgraph Stat.Logistic.Graphics.Roc;
    dynamic _cValue _TITLE _ROCID;
    BeginGraph / designwidth=defaultDesignHeight;
      layout overlayequated
        / equatetype=square
          yaxisopts=(label="Sensitivity" shortlabel="TPR")
          xaxisopts=(label="1 - Specificity" shortlabel="FPR")
          commonaxisopts=(tickvaluelist=(0 0.2 0.4 0.6 0.8 1) viewmin=0 viewmax=1)
      ;
        bandplot    x=_1MSPEC_ limitlower=0 limitupper=_SENSIT_
                  / fillattrs=(color=yellow);
        lineparm    x=0 y=0 slope=1
                  / extend=true lineattrs=(pattern=shortdash thickness=0.5pct);
        seriesplot  y=_SENSIT_ x=_1MSPEC_
                  / primary=true connectorder=xaxis lineattrs=(thickness=0.5pct);
        scatterplot y=_SENSIT_ x=_1MSPEC_
                  / datalabel=_pos_ ;
      endlayout;
    EndGraph;
  end;

  /* the ODS style template */
  define style styles.OrlandoBlue;
    parent = styles.HTMLBlue;
    style GraphValueText from GraphValueText / font=('Times New Roman', 12pt, Bold);
    style GraphLabelText from GraphLabelText / font=('Times New Roman', 12pt, Bold);
    style GraphBorderLines from GraphBorderLines / LineThickness=0;
    style GraphWalls from GraphWalls / color= CXC0C0C0;
    style GraphBackground / transparency=1;
  end;
run;

ods html style=OrlandoBlue image_dpi=300;

proc logistic data=neuralgia plots(only)=roc rocoptions(nodetails out=dat_roc);
  class treatment sex;
  model pain(event='Yes') =treatment age sex treatment*age sex*age treatment*sex
       / nofit;
  roc 'Treatment'     treatment;
  roc 'Treatment|Age' treatment sex treatment*sex;
  roc 'Treatment|Age' treatment age treatment*age;
run; quit;
```

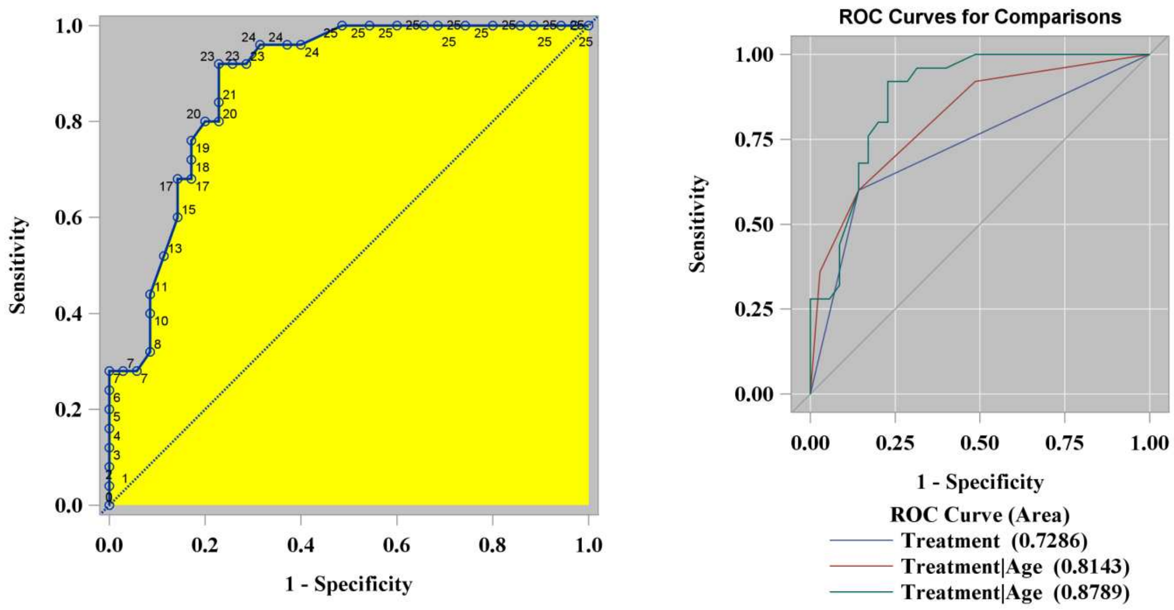

Figure 4. The ROC plot (Treatment|Age) and the ROC comparison plot produced by Proc Logistic with modified GTL and ODS style templates. Note: Both templates affect the first graph and only the ODS style template affects the second graph (Example 5)

CONCLUSION

The statistical procedures in SAS/STAT have been widely used by statisticians and analysts to produce reliable modeling results such as P-values and confidential intervals. With the methods introduced in this paper to customize the ODS graphical output using GTL and ODS style templates, these procedure now can directly produce high-quality ready-to-use statistical graphs according to the graph configurations in a specific application or project scenario. This paper introduces the template-based graph enhancement using examples from PROC LOGISTICS, but the methods can be extended to control the GTL template or ODS style template across different SAS/STAT procedures to regulate the graphical output with common features such as font size and company or project headers.

REFERENCES

- Xiangxiang Meng. The Power to Plot: Three Ways to Enhance SG Graphical Outputs. SAS Global Forum 2011, Paper 095-2011.
- Robert N. Rodriguez. An Introduction to ODS for Statistical Graphics in SAS 9.1. The Proceedings of the 29th SAS Users Group International, Page 204-29.

CONTACT INFORMATION

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

Yang Xiao
University of Cincinnati
xiaoyo@mail.uc.edu

Xiangxiang Meng
SAS Institute Inc.
Xiangxiang.Meng@sas.com

APPENDIX

The default source code of Stat.Logistic.Graphics.Roc:

```
define statgraph Stat.Logistic.Graphics.Roc;
  notes "Receiver Operating Characteristic Curve";
  dynamic _cValue _TITLE _ROCID;
  BeginGraph / designwidth=defaultDesignHeight;
    entrytitle _TITLE;
    entrytitle "Area Under the Curve = " eval (STRIP(PUT(_CVALUE,6.4)))
               / textattrs=GRAPHVALUETEXT;
    layout overlayequated
      / equatetype=square
        yaxisopts=(gridDisplay=auto_on label="Sensitivity"
                   shortlabel="TPR" offsetmin=0.05 offsetmax=0.05)
        xaxisopts=(gridDisplay=auto_on label="1 - Specificity"
                   shortlabel="FPR" offsetmin=0.05 offsetmax=0.05)
        commonaxisopts=(tickvaluelist=(0 .25 .5 .75 1) viewmin=0 viewmax=1);
      lineparm x=0 y=0 slope=1 / extend=true lineattrs=GRAPHREFERENCE;
      seriesplot y=_SENSIT_ x=_1MSPEC_
        / primary=true connectorder=xaxis rolename=(tip1=_ROC_ tip2=_FREQ_
          tip3=_WEIGHT_ tip4=_POSPRED_ tip5=_NEGPRED_ tip6=_PROB_)
          tip=(tip1 y x tip2 tip3 tip4 tip5 tip6);
      if (EXISTS(_ROCID))
          scatterplot y=_SENSIT_ x=_1MSPEC_
             / rolename=(tip1=_ROC_ tip2=_FREQ_ tip3=_WEIGHT_ tip4=_POSPRED_
               tip5=_NEGPRED_ tip6=_PROB_) tip=(tip1 y x tip2 tip3 tip4 tip5 tip6)
               freq= eval(_ROCID >= 0);
      endif;
      seriesplot y=_SENSIT_ x=_1MSPEC_
         / connectorder=xaxis datalabel=_ROCID datalabelattrs=GRAPHDATADEFAULT
           lineattrs=GRAPHDATADEFAULT(thickness=0)  markerattrs=GRAPHDATADEFAULT(size=0)
           rolename=(tip1=_ROC_ tip2=_FREQ_ tip3=_WEIGHT_ tip4=_POSPRED_
           tip5=_NEGPRED_ tip6 =_PROB_) tip=(tip1 y x tip2 tip3 tip4 tip5 tip6);
    endlayout;
  EndGraph;
end;
```

The default source code of Stat.Logistic.Graphics.Roc::

```
define statgraph Stat.Logistic.Graphics.ROCOverlay;
  notes "Receiver Operating Characteristic Overlaid Curves";
  dynamic _TITLE;
  BeginGraph / designwidth=defaultDesignHeight;
    entrytitle _TITLE;
    layout overlayequated
      / equatetype=square
        yaxisopts=(gridDisplay=auto_on label="Sensitivity"
                   shortlabel="TPR" offsetmin=0.05 offsetmax=0.05)
        xaxisopts=(gridDisplay=auto_on label="1 - Specificity"
                   shortlabel="FPR" offsetmin=0.05 offsetmax=0.05)
        commonaxisopts=(tickvaluelist=(0 .25 .5 .75 1) viewmin=0 viewmax=1);
      lineparm x=0 y=0 slope=1 / extend=true lineattrs=GRAPHREFERENCE;
      seriesplot y=_SENSIT_ x=_1MSPEC_ / connectorder=xaxis tip=(group y x)
               group=_ROC_ index=_GROUP_ name="Step" primary=true;
      discretelegend "Step" / title="ROC Curve (Area)";
    endlayout;
  EndGraph;
end;
```