

Why Should You Be Using the New SG (Statistical Graphics) Procedures in SAS 9.2?

Philip R Holland
Holland Numerics Ltd



Agenda

- Introduction
- Creating the data
- Using PROC GPLOT
- Using PROC SGPLOT
- Using PROC SGPANEL
- Using PROC SGSCATTER
- Using PROC SGRENDER
- Conclusions



Introduction

There is always a hope that a new software feature will improve on what already exists, but fear of the unknown stops you from trying it out, just in case it is a waste of time. We are going to try out the new SG (Statistical Graphics) procedures in SAS 9.2 for you, and demonstrate some of the new and useful features of PROC SGPLOT, PROC SGPANEL, PROC SGSCATTER and PROC SGRENDER.

For all the various SG Procedures the comparison necessarily has to be with PROC GPLOT. Nearly every SAS programmer who creates graphic reports will have used PROC GPLOT, so it is reasonable to use it as the basic output. The following image and code use PROC GPLOT, and similar data will be used with the other SG Procedures, so that a reasonable comparison can be made.



Creating the data - 1

```
PROC SQL;
  CREATE TABLE plotdata AS
    SELECT INTCK('QTR','01jan1994'd,monyr)
           AS visitnum
           ,product
           ,mean(predict) AS price
           ,stderr(predict) AS price_se
           ,count(*) AS count
    FROM   sashelp.prdsal2 (WHERE=(
           product IN ('BED','CHAIR','DESK')
           AND predict>400))
    GROUP BY
           visitnum
           ,product
;
QUIT;
```



Creating the data - 2

```
DATA plotdata_classic;
  SET plotdata (RENAME=(price=val));
  SELECT (product);
    WHEN ('BED') visitnum=visitnum-0.1;
    WHEN ('DESK') visitnum=visitnum+0.1;
    OTHERWISE;
  END;
  price=val+price_se;
  OUTPUT;
  price=val-price_se;
  OUTPUT;
  LABEL price='Sales ($)'
        visitnum='Quarter'
        ;
RUN;
```



Creating the data - 3

```
DATA plotdata_ods;
  SET plotdata;
  LENGTH ccount $4;
  SELECT (product);
    WHEN ('BED') visitnum=visitnum-0.1;
    WHEN ('DESK') visitnum=visitnum+0.1;
    OTHERWISE;
  END;
  price_upper=price+price_se;
  price_lower=price-price_se;
  ccount=STRIP(PUT(count,4.));
  LABEL price='Sales ($)'
        visitnum='Quarter'
        ;
RUN;
```

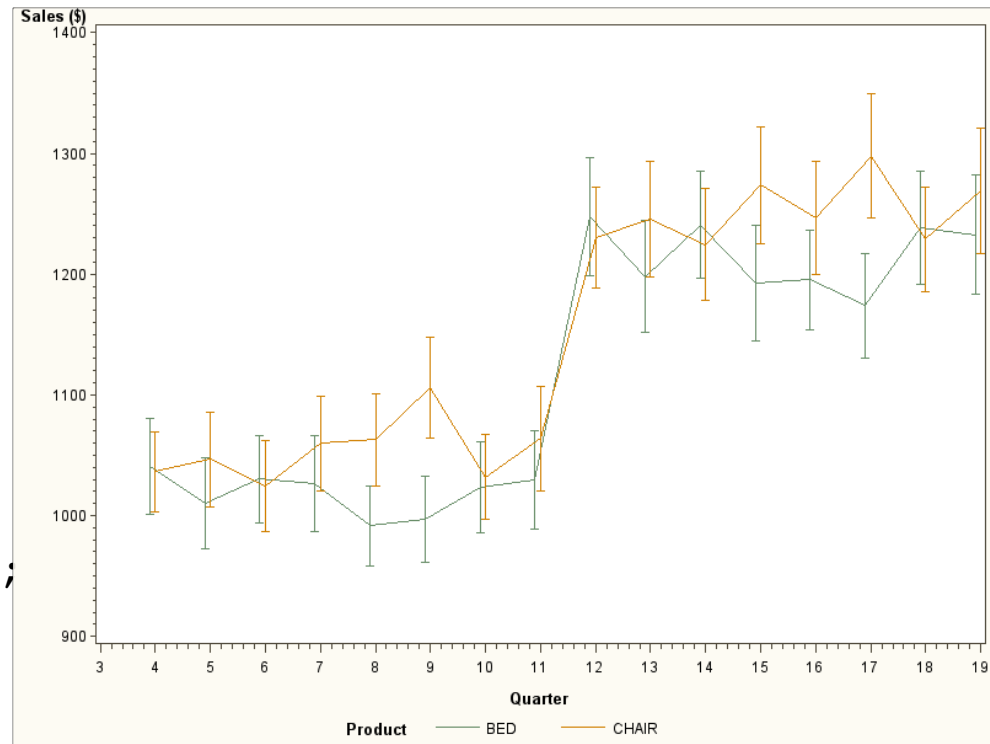


Using PROC GPLOT

```

ODS RTF
  FILE="gplot.rtf"
  STYLE=serifprinter;
PROC GPLOT
  DATA=plotdata_classic
    (WHERE=(product IN
      ('CHAIR', 'BED')));
  SYMBOL1 I=HILOTJ V=NONE;
  SYMBOL2 I=HILOTJ V=NONE;
  PLOT
    price*visitnum=product;
RUN;
QUIT;
ODS RTF CLOSE;

```

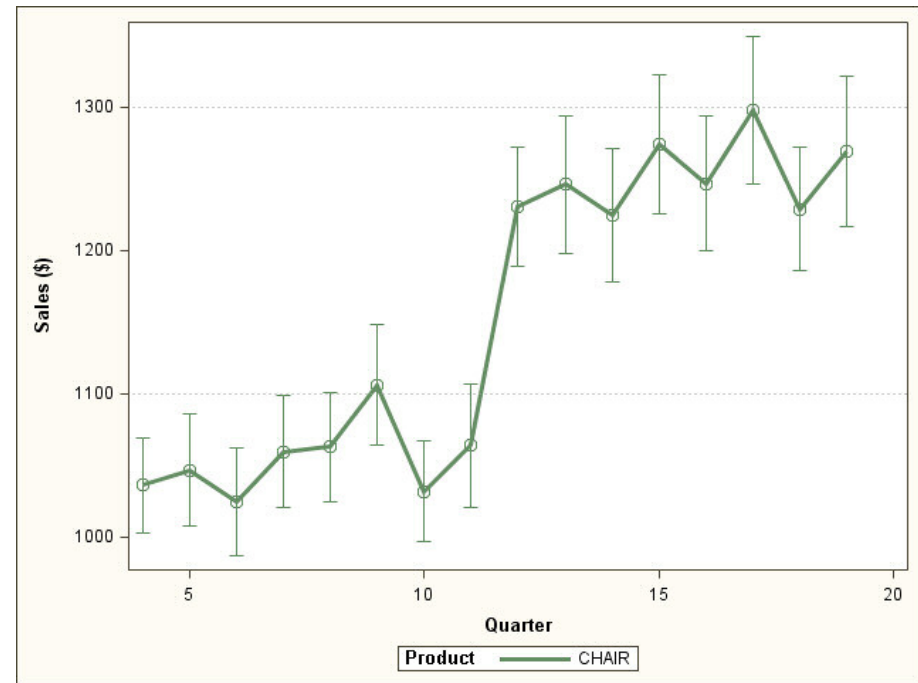


Using PROC SGPLOT

```

ODS RTF FILE="sgplot.rtf"
  STYLE=serifprinter;
ODS GRAPHICS ON;
PROC SGPLOT DATA=plotdata_ods
  (WHERE=(product='CHAIR'))
  TMPLOUT="sgplot.sas";
  SERIES X=visitnum Y=price /
    MARKERATTRS=(SIZE=10PX)
    LINEATTRS=(THICKNESS=3PX)
    GROUP=product;
  SCATTER X=visitnum Y=price /
    YERRORUPPER=price_upper
    YERRORLOWER=price_lower
    MARKERATTRS=(SIZE=10PX)
    GROUP=product;
  REFLINE 1100 / AXIS=Y
    LINEATTRS=(PATTERN=DOT);
  REFLINE 1300 / AXIS=Y
    LINEATTRS=(PATTERN=DOT);
RUN;
ODS GRAPHICS OFF;
ODS RTF CLOSE;

```



Using PROC SGPLOT: sgplot.sas

This is the SAS code saved to **sgplot.sas**:

```
proc template;
  define statgraph sgplot;
    begingraph;
      EntryTitle "Sales" /;
      EntryTitle "Bed and Chair" /
        textattrs=(size=GraphLabelText:fontsize);
      layout overlay;
      SeriesPlot X='visitnum'n Y='price'n / Group='PRODUCT'n
        Markerattrs=( Size=10px)
        Lineattrs=( Thickness=3px) LegendLabel="Sales ($)"
        NAME="series";
      ScatterPlot X='visitnum'n Y='price'n / primary=true
        Group='PRODUCT'n Markerattrs=( Size=10px)
        YErrorUpper='price_upper'n YErrorLower='price_lower'n
        LegendLabel="Sales ($)" NAME="SCATTER";
      DiscreteLegend "series"/ title="Product";
      ReferenceLine y=1000 / clip=true Lineattrs=( Pattern=34);
      ReferenceLine y=1300 / clip=true Lineattrs=( Pattern=34);
    endlayout;
    EntryFootnote "Program: v92_sgplot_lineplot.sas" /;
  endgraph;
end;
run;
```

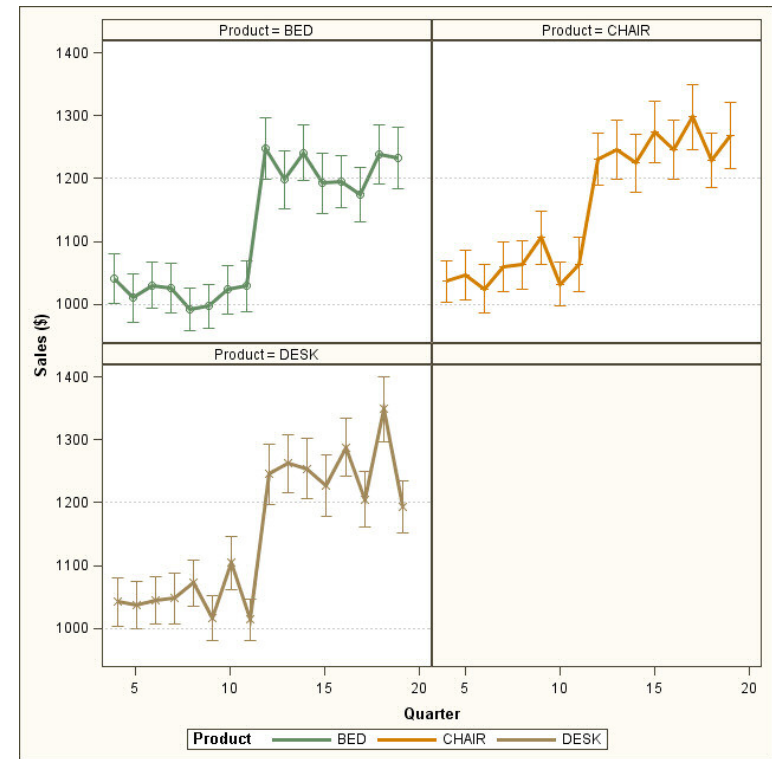


Using PROC SGPANEL

```

ODS RTF FILE="sgpanel.rtf"
  STYLE=serifprinter;
ODS GRAPHICS ON;
PROC SGPANEL DATA=plotdata_ods
  TMPLOUT="sgpanel.sas";
  PANELBY product / LAYOUT=PANEL;
  SERIES X=visitnum Y=price /
    MARKERATTRS=(SIZE=10PX)
    LINEATTRS=(THICKNESS=3PX
      PATTERN=SOLID) GROUP=product;
  SCATTER X=visitnum Y=price /
    YERRORUPPER=price_upper
    YERRORLOWER=price_lower
    MARKERATTRS=(SIZE=10PX)
    GROUP=product;
  REFLINE 1000 / AXIS=Y
    LINEATTRS=(PATTERN=DOT) ;
  REFLINE 1200 / AXIS=Y
    LINEATTRS=(PATTERN=DOT) ;
RUN;
ODS GRAPHICS OFF;
ODS RTF CLOSE;

```



Using PROC SGPANEL: sgpanel.sas

This is the SAS code saved to **sgpanel.sas**:

```
proc template;
  define statgraph sgpanel;
    dynamic _xviewmin_ _xviewmax_ _yviewmin_ _yviewmax_;
    dynamic _panelnumber_ _byline_;
    beginnograph / designwidth=640 designheight=640;
      EntryTitle "Sales" /;
      EntryTitle "Bed, Chair and Desk" / textattrs=(size=GraphLabelText:fontsize);
      layout gridded / rowgutter=5;
        layout datapanel classvars=( 'PRODUCT'n) / sparse=false includeMissingClass=false
          rowDataRange=unionall columnDataRange=unionall panelNumber=_panelnumber_ cellHeightMin=50px
          cellWidthMin=50px start=TopLeft columns=2 rows=2 rowAxisOpts=( display=all altdisplay=all
          linearOpts=( viewmin=_yviewmin_ viewmax=_yviewmax_))
          columnAxisOpts=( display=all altdisplay=all
          linearOpts=( viewmin=_xviewmin_ viewmax=_xviewmax_));
        layout prototype / __SGPROC;
          SeriesPlot X='visitnum'n Y='price'n / Group='PRODUCT'n Markerattrs=( Size=10px)
            Lineattrs=( Pattern=1 Thickness=3px) LegendLabel="Sales ($)" NAME="SERIES";
          ScatterPlot X='visitnum'n Y='price'n / primary=true Group='PRODUCT'n Markerattrs=( Size=10px)
            YErrorUpper='price_upper'n YErrorLower='price_lower'n LegendLabel="Sales ($)"
            NAME="SCATTER";
          ReferenceLine y=1000 / clip=true Lineattrs=( Pattern=34);
          ReferenceLine y=1200 / clip=true Lineattrs=( Pattern=34);
        endlayout;
      endlayout;
      DiscreteLegend "SERIES"/ title="Product";
    endlayout;
    EntryFootnote "Program: v92_sgpanel_lineplot.sas" /;
  endgraph;
end;
run;
```

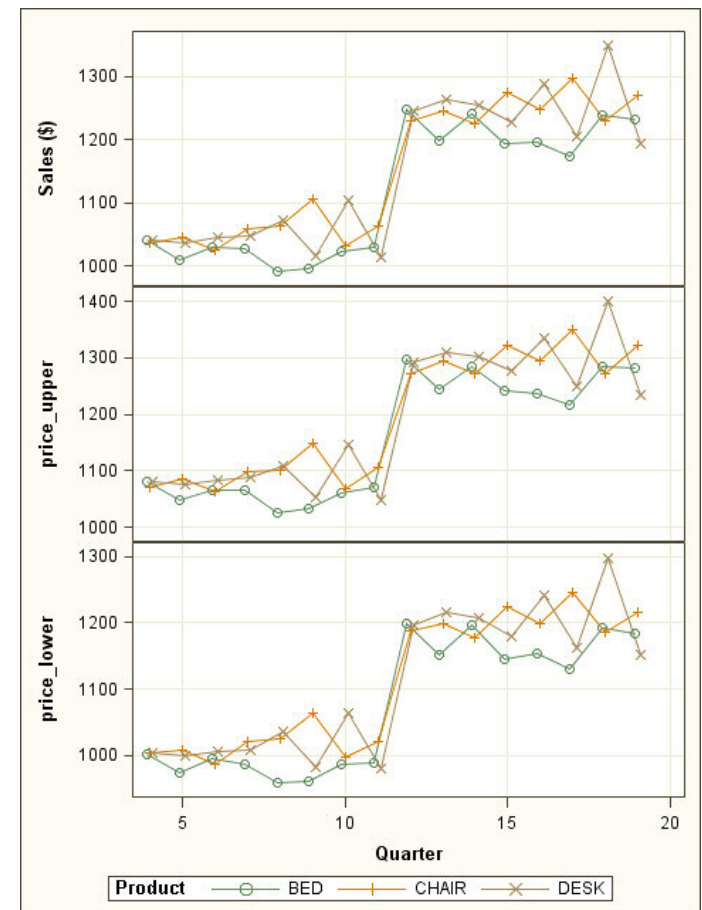


Using PROC SGSCATTER

```

ODS RTF FILE="sgscatter.rtf"
  STYLE=serifprinter;
ODS GRAPHICS ON;
PROC SGSCATTER DATA=plotdata_ods
  TMPLOUT="sgscatter.sas";
  COMPARE Y=(price price_upper
             price_lower)
           X=visitnum /
  GROUP=product
  MARKERATTRS=(SIZE=10)
  JOIN=(LINEATTRS=(PATTERN=SOLID))
  GRID;
RUN;
ODS GRAPHICS OFF;
ODS RTF CLOSE;

```



Using PROC SGSCATTER: sgscatter.sas

This is the SAS code saved to **sgscatter.sas**:

```
proc template;
  define statgraph sgscatter;
    begingraph / designwidth=480 designheight=640;
      EntryTitle "Sales" /;
      EntryTitle "Bed, Chair and Desk" / textattrs=(size=GraphLabelText:fontsize);
      layout gridded;
      layout lattice / pad=(top=5 bottom=5) columnDataRange=union;
      ColumnAxes;
      ColumnAxis / griddisplay=on;
      EndColumnAxes;
      layout overlay / xaxisopts=( griddisplay=on) yaxisopts=( griddisplay=on);
      SeriesPlot X='visitnum'n Y='price'n / primary=true display=(markers)
        CONNECTORDER=XAXIS Lineattrs=( Pattern=1) Group='PRODUCT'n
        Markerattrs=( Size=10) NAME="COMPARE";
      endlayout;
      layout overlay / xaxisopts=( griddisplay=on) yaxisopts=( griddisplay=on);
      SeriesPlot X='visitnum'n Y='price_upper'n / primary=true
        display=(markers) CONNECTORDER=XAXIS Lineattrs=( Pattern=1)
        Group='PRODUCT'n Markerattrs=( Size=10);
      endlayout;
      layout overlay / xaxisopts=( griddisplay=on) yaxisopts=( griddisplay=on);
      SeriesPlot X='visitnum'n Y='price_lower'n / primary=true
        display=(markers) CONNECTORDER=XAXIS Lineattrs=( Pattern=1)
        Group='PRODUCT'n Markerattrs=( Size=10);
      endlayout;
      endlayout;
      DiscreteLegend "COMPARE" / order=rowmajor title="Product";
      endlayout;
      EntryFootnote "Program: v92_sgscatter_lineplot.sas" /;
    endgraph;
  end;
run;
```



Using PROC SGRENDER: Template

This is a graph template for PROC SGRENDER:

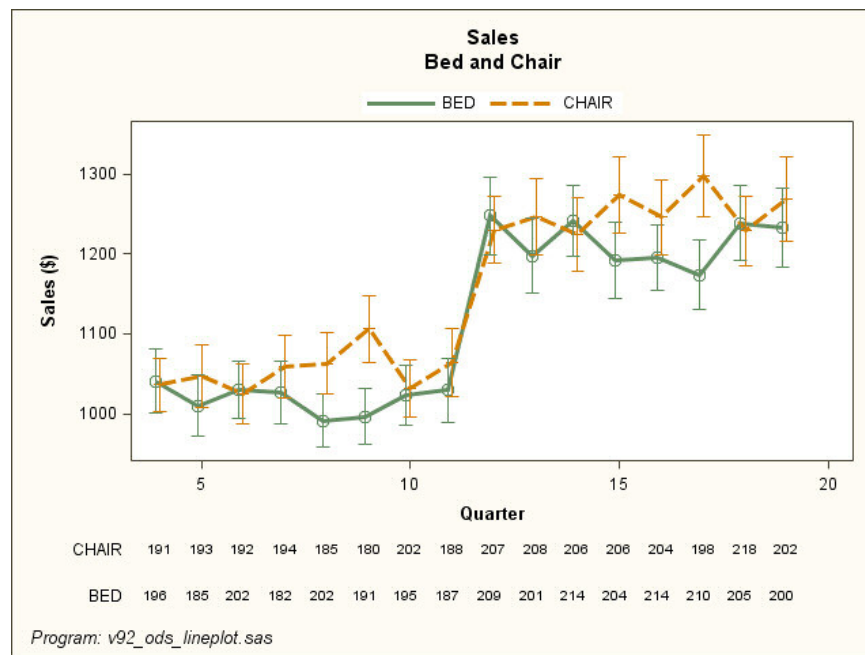
```
ODS PATH work.mypath(UPDATE) sashelp.tmplmst(READ);
PROC TEMPLATE;
  DEFINE STATGRAPH Graphics.SECountPlot2;
    DYNAMIC _title _title2 _title3 _footnote _footnote2 _footnote3 _xvar _xlabel
      _ylabel _yintercepta _yinterceptb _yvar1 _yupper1 _ylower1 _nvar1 _group;
    BEGINGRAPH;
      ENTRYTITLE _title;
      ENTRYTITLE _title2;
      ENTRYTITLE _title3;
      LAYOUT LATTICE / COLUMNS=1 ROWS=2 ROWWEIGHTS=(.85 .15) COLUMNDATARANGE=UNIONALL;
      LAYOUT OVERLAY / PAD=(TOP=2% BOTTOM=2% LEFT=2% RIGHT=2%) XAXISOPTS=(LABEL=_xlabel)
        YAXISOPTS=(LABEL=_ylabel) OPAQUE=FALSE;
      SERIESPLOT X=_xvar Y=_yvar1 / MARKERATTRS=(SIZE=10PX) LINEATTRS=(THICKNESS=3PX)
        NAME='series' GROUP=_group;
      SCATTERPLOT X=_xvar Y=_yvar1 / YERRORUPPER=_yupper1 YERRORLOWER=_ylower1
        MARKERATTRS=(SIZE=10PX) GROUP=_group;
      IF (_yintercepta) LINEPARM X=0 Y=_yintercepta SLOPE=0 / LINEATTRS=(PATTERN=DOT);
      ENDIF;
      IF (_YINTERCEPTB) LINEPARM X=0 Y=_yinterceptb SLOPE=0 / LINEATTRS=(PATTERN=DOT);
      ENDIF;
      DISCRETELEGEND 'series' / ACROSS=4 BORDER=FALSE VALIGN=TOP;
    ENDLAYOUT;
    LAYOUT OVERLAY / PAD=(BOTTOM=2% LEFT=2% RIGHT=2%) BORDER=FALSE WALLDISPLAY=NONE
      XAXISOPTS=(DISPLAY=NONE) X2AXISOPTS=(DISPLAY=NONE) Y2AXISOPTS=(DISPLAY=none)
      YAXISOPTS=(DISPLAY=(TICKVALUES));
      SCATTERPLOT X=_xvar Y=_group / MARKERCHARACTERATTRS=(COLOR=BLACK) MARKERCHARACTER=_nvar1;
    ENDLAYOUT;
  ENDLAYOUT; /* lattice*/
  ENTRYFOOTNOTE HALIGN=LEFT _footnote;
  ENTRYFOOTNOTE HALIGN=LEFT _footnote2;
  ENTRYFOOTNOTE HALIGN=LEFT _footnote3;
ENDGRAPH;
END;
RUN;
```

Using PROC SGRENDER

```

ODS RTF FILE="sgrender.rtf" STYLE=serifprinter;
ODS GRAPHICS ON;
PROC SGRENDER DATA=plotdata_ods
  (WHERE=(product IN ('BED','CHAIR')))
  TEMPLATE='Graphics.SECountPlot2';
  DYNAMIC _title="Sales"
    _title2="Bed and Chair" _footnote="Program: template.sas"
    _xvar="visitnum"
    _xlabel="Quarter"
    _ylabel="Sales ($)"
    _yvar1="price"
    _yupper1="price_upper"
    _ylower1="price_lower"
    _nvar1="CCOUNT"
    _group="product";
RUN;
ODS GRAPHICS OFF;
ODS RTF CLOSE;

```



Conclusions

While PROC GPLOT, and also PROC GCHART, can create acceptable graphs, their default settings are generally not what is required. The SG procedures employ default settings that are much closer to what is likely to be required, and adjustments are therefore easier to code and document.

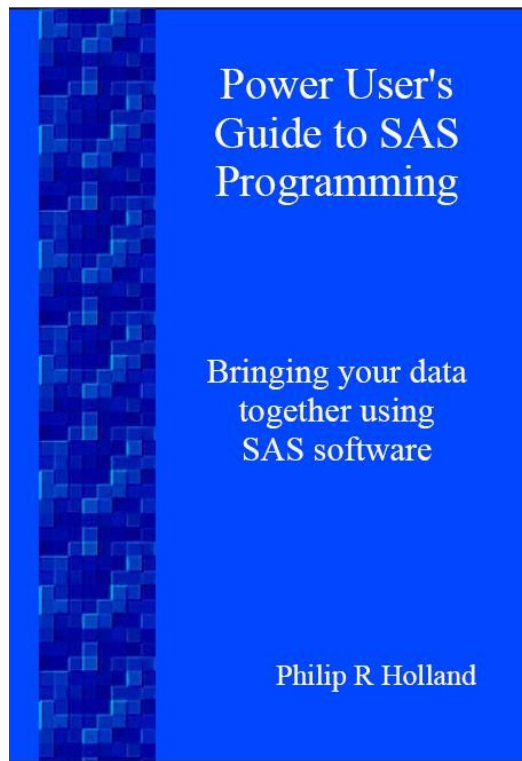
PROC SGPLOT and PROC SGPANEL allow the production of combined graphs which are, at best, difficult to create using PROC GPLOT or PROC GCHART and Annotate. PROC SGPANEL and PROC SGSCATTER have the ability to create multiple graphs with common axes, which previously required many SAS/GRAPH procedure calls, careful template design, followed by the use of PROC GREPLAY to render the completed graph. For the ultimate control over any graph, ODS Graphics templates, which can include user-selected features, can be rendered using PROC SGRENDER to allow the simple reuse of tested graphics code.



Contact details

Philip R Holland, Holland Numerics Ltd
94 Green Drift, Royston, Herts SG8 5BT, UK
phil@hollandnumerics.com
www.hollandnumerics.com

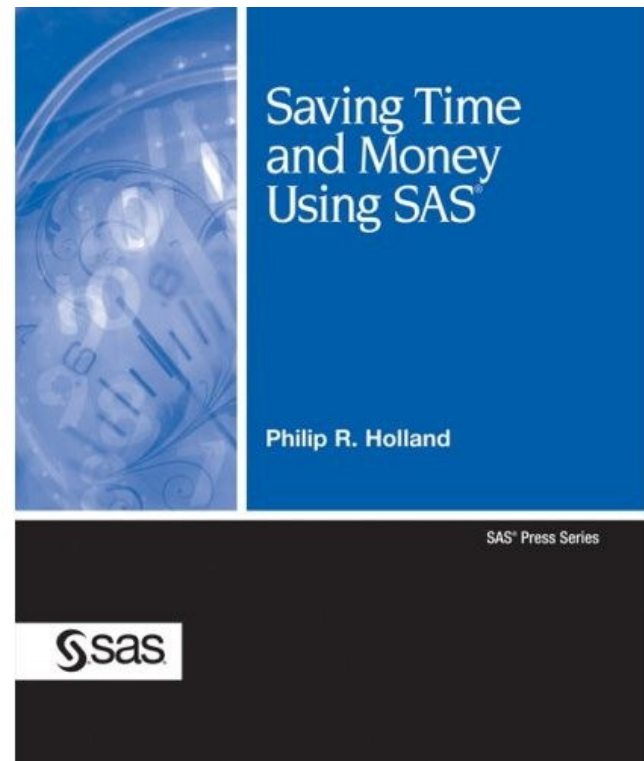
This paper, presentation and the SAS code
used can all be downloaded from:
www.hollandnumerics.com/SASPAPER.HTM



Published by Lulu.com in December 2008.
A5 format, 90 pages.

This book is available for purchase through:

- **www.hollandnumerics.com** (including links to Amazon.co.uk, Amazon.com, Blackwell Books and Lulu.com)
- ISBN: 978-1-4092-5088-3
- Available as printed book or download



Published by SAS Press in July 2007.
9 x 7 format, 248 pages.

This book is available for purchase through:

- **www.hollandnumerics.com** (including links to Amazon.co.uk, Amazon.com, Blackwell Books and SAS Press)
- ISBN: 978-1-59047-574-4
- Available as printed book or for Kindle