

Creating a Publication-Quality Multi-Panel Graph For Generalized Linear Model Analysis.

Neeta V Shenvi M.S. and Kirk A. Easley, MAppStat, Emory University, Department of Biostatistics and Bioinformatics, Emory University, Atlanta, Georgia

ABSTRACT

The SAS procedure GENMOD is often used to fit classical linear models to analyze longitudinal data such as patient-reported outcomes (PRO) measurements over time. We used the SAS procedure GENMOD to estimate responder rate defined as improvements in visual analog score knee pain by 25% and at least 20mm or points) . . The responder (yes/no) endpoint evaluated one-year post-baseline was analyzed as a binary outcome. The model included predictors for sex, age group, 4 treatment groups, two -way and three-way interactions. We show use of SAS template language to create a publication grade 4 panel plot to visualize model-based responder rates and 95% confidence intervals.

INTRODUCTION

Effective graphics are an integral part of publication in modern statistical analysis and add visual clarity and rich content to publication and presentation. They reveal patterns, differences, and uncertainty that are not readily apparent in tabular output. Statistical graphs are produced by ODS Graphics, which is an extension of ODS (the Output Delivery System). With SAS 9.2, users can also generate ODS graphs automatically from analytical procedures, which can produce the graphs. Users who need to go beyond the graphs created by these SAS procedures can use Graph Template Language (GTL) directly to design their graphs using the TEMPLATE and SGRENDER procedures.

GTL has a layout-centric architecture. Each graph comprises components such as plots, insets, and legends that can be combined in flexible ways inside layout containers to build complex graphs. GTL supports the use of transparency and anti-aliasing for creating modern graphs. The rendering technology for GTL is not based on device drivers. The graphics area is not partitioned in cell-based units. Graphical content gracefully scales up and down as the size of the output is changed. Markers, lines, and fonts are scaled using DPI. GTL produces all output in industry standard formats, such as PNG, GIF, JPEG, and so on.

The 4-panel plot shown in Figure 1 cannot be automatically generated by SAS GENMOD procedure. We utilized advanced capabilities of GTL to display GENMOD results to deliver presentation quality graphic that will have high resolution and clarity.

DATA

A multi-arm parallel-group, multicenter, single-blind, randomized, controlled clinical trial, was designed to recruit adult patients with a diagnosis of knee osteoarthritis (KOA) as determined by radiographs (Mautner et al 2023). Patients were randomized to three different arms with a 3:1 distribution and were each compared against the gold-standard corticosteroid injection (CSI). Arm 1: Autologous bone marrow concentrate (BMAC) (n=120): corticosteroid (CSI) (n=40); Arm 2: Stromal Vascular Fraction (SVF)(n=120): CSI (n=40); Arm 3: umbilical cord tissue (UCT) (n=120): CSI (n=40). Subjects were enrolled if they were between 40 and 70 years of age and carried a diagnosis of KOA as determined by radiographs within three months of their clinical visit regardless of radiographic severity. We hypothesized that cell therapies would be superior to corticosteroids for treatment of knee OA at 1 year. A responder was defined as improvement in visual analog scale (VAS) by 25% or at least 20 points at 12 months follow up. We analyzed the difference in responder (responder rate) based on cellular injection given as well as age, sex and severity of KOA. In this paper we illustrate the multi-panel plot to visualize differences in responder rate by age, sex, and treatment after adjusting for center.

To estimate the responder rate we used GENMOD procedure for yes/no endpoint. During treatment responder status represented by the variable `responder_yes` (coded here as 0=No, 1=YES), is determined at Month 12. The

variables **center**, **sex**, and **age** (≥ 60 and < 60) are classification variables with two levels and variable **treatment** has 4 levels.

We estimated model-based responder rates and 95% confidence intervals in each subgroup. The LSMEANS statement computes and compares least squares means (LS-means) of fixed effects. LS-means are *predicted population margins*—that is, they estimate the marginal means over a balanced population.

Step 1:

The programmer has to understand the input data set variables and make sure all variables required for the model are present in the data set. A dummy responder data set with each subject's responder status, treatment, age, and sex is shown below for demonstration purposes.

Partial input dataset is shown below:

responder_yes	trtp	Sex	Agegr
no	SVF	F	<60
yes	SVF	M	<60
yes	CSI	F	≥ 60
yes	BMAC	F	≥ 60
no	BMAC	M	≥ 60
yes	CSI	F	≥ 60
yes	CSI	F	<60
no	UCT	F	<60
yes	CSI	F	≥ 60
no	BMAC	M	<60
yes	BMAC	M	<60
no	CSI	M	<60
yes	UCT	M	≥ 60
no	UCt	F	≥ 60

Step 2 GENMOD Model Fitting:

Fit GENMOD model for responder outcome (yes/no) endpoint evaluated at one year post-baseline visit. The predictors used are sex (2-levels), age group (2 levels: < 60 vs ≥ 60), treatment groups (4 levels), two-way and three-way interactions and adjusted for center. Derive **fixed effect estimates** using the model below:

```

PROC GENMOD DATA=responder DESCENDING;
CLASS trtp sex agegr center;
MODEL responder_yes= trtp
    sex
    agegr
    agegr*trtp
    sex*trtp
    agegr*trtp*sex center/ LINK=log DIST=binomial TYPE3 ;
LSMEANS trtp
    sex
    agegr
    agegr*trtp
    agegr*trtp*sex/ CL EXP;
ODS OUTPUT lsmeans=lsmeans2024;
run;

```

lsmeans2024: lsmeans2024 is the lsmeans object from the ODS Output statement. This data set has a mean estimate (LS means or responder rate), stdErr (standard error), Probt (p-value), Lower (lower confidence interval), and Upper (upper confidence interval) for all effects listed in the lsmeans statement.

LSMEANS trtp sex agegr agegr2*trtp sex*trtp agegr*trtp*sex/ **CL EXP**;

LS-means can be computed for any effect in the MODEL statement that involves CLASS variables. Option EXP requests exponentiation of the LS-means for link=log function.

The lsmeans statement with TRTP gives marginal (overall) means (responder rate) for 4 treatments, agegr gives two means (responder rate) for 2 age group levels, 2-way interaction “agegr*trtp” gives eight means for four treatment levels by two age groups , 3-way interaction term “agegr*trtp*sex” term gives 16 means (responder rate) for 4 treatments, 2 sex and 2 age group levels.

Partial output of lsmeans2024 data set for sex and agegr is given below.

Sex	Estimate	Standard Error	Lower	Upper	Exponentiated	Exponentiated Lower	Exponentiated Upper
F	-0.4722	0.05438	-0.5788	-0.3656	0.6236	0.5606	0.6938
M	-0.6903	0.08237	-0.8517	-0.5288	0.5014	0.4267	0.5893

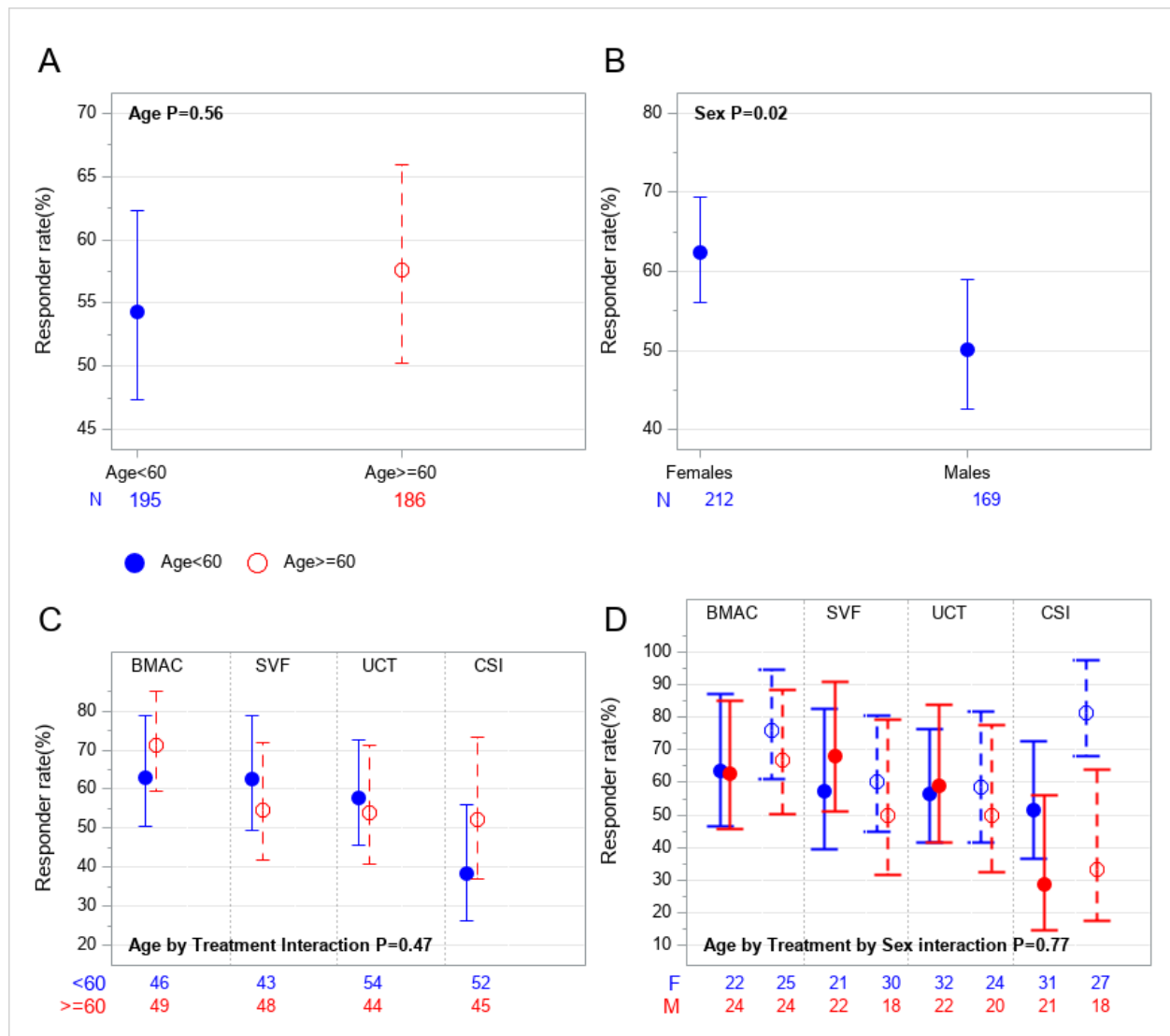
agegr	Estimate	Standard Error	Lower	Upper	Exponentiated	Exponentiated Lower	Exponentiated Upper
<60	-0.6100	0.06993	-0.7471	-0.4730	0.5433	0.4737	0.6231
>=60	-0.5524	0.06966	-0.6890	-0.4159	0.5756	0.5021	0.6597

The responder rate estimates for the two sex levels are provided by the EXP option in the LSMEANS statement (Exponentiated column). The responder rate for females is 62% (95% CI: 56-69%) and for males is 50%(95% CI: 43-59%).

We then add sample size variables for each effect in lsmeans2024 dataset (code not shown).

To visualize treatment differences in responder rates among groups we created a 4-panel graph (Figure 1) for main effect age group (panel A), sex (panel B), age group by treatment interaction (panel C), and 3-way treatment by sex by age interaction (panel D).

Figure 1 Adjusted Responder Rates with 95% Confidence Intervals based on VAS Knee Pain Scores at 12 Months.



Step 3:

LSMEANS data manipulation for plotting: Here we simply create a numeric treatment variable for trtpn. This numeric variable will enable correct sorting of 4 treatment groups in our preferred order for panel C and D. Then we create responder rate (mean, lower and upper CI) by multiplying exponentiated variables by 100.

```

*Step 3;
DATA lsmeans2024;
SET lsmeans2024;

*Making numeric variable for character treatment , sex, age group variables;
IF trtp ="BMAC" THEN trtpn=1;
IF trtp ="SVF" THEN trtpn=2;
IF trtp ="UCT" THEN trtpn=3;
IF trtp ="CSI" THEN trtpn=4;

*Responder mean , lower and upper ;
responder_mean=ExpEstimate*100;
responder_lower=LowerExp*100;
responder_upper=upperExp*100;

RUN;

```

We now create panel variable for 4 panels for 4 effects used in the model.

```

*Step 3;

DATA lsmeans2024;
SET lsmeans2024;
IF effect="agegr" THEN panel="A";
IF effect="SEX" THEN panel="B";
IF effect="TRTP*agegr" THEN panel="C";
IF effect="TRTP*SEX*agegr" THEN panel="D";
RUN;

```

Step 4: Four-panel TEMPLATE SHELL CODE

To implement this in GTL, you can use the lattice layout with two cells arranged in two rows. Each cell contains an overlay layout for panels A, B, C, D with a series plot for population-specific responder rates and 95% confidence intervals. Here is the template program shell. We will illustrate each of the 4 panel code subsequently.

```

PROC TEMPLATE;
  DEFINE STATGRAPH Cl;
    BEGINGRAPH / designwidth=9in designheight=5in ;
    ENTRYTITLE halign=left title/textattrs=(size=14pt);

```

```
LAYOUT lattice / columns=2 rows=2 columngutter=5 rowgutter=5;
```

```
***** Panel A code goes here;
```

```
***** Panel B code goes here;
```

```
***** Panel C code goes here;
```

```
***** Panel D code goes here;
```

```
ENDLAYOUT;
```

```
ENDGRAPH;
```

```
END;
```

```
RUN;
```

The four panels in figure 1 are clustered plots. We have kept code very simple for clustered display. We avoided code for attribute maps to control styling attributes (such as line and symbol) for each group value within the clusters. We instead achieved cluster effect by creating discrete x- coordinate variables for each group within the cluster and styled it by customizing line and maker attributes. This method is intuitive and scalable when you want your graphs to have consistent visual properties for multiple panels.

Step 5: Panel A Plotting responder rate means and 95% confidence intervals for two age groups.

Here we want to display two members of age group cluster. We use lsmeans2024.sas7bdat dataset (where effect="agegr") to plot mean and 95% CI and sample size. The data is shown below.

Effect	N	agegr	responder_mean	responder_lower	responder_upper
agegr	195	<60	54.3	47.4	62.3
agegr	186	>=60	57.6	50.2	66.0

We create two sample size variables for each age group (**A_n_L60, A_n_G60**), 2 variables for X-coordinate (**A_X1_plot, A_X2_plot**). The sequence variable A_X was created as x-coordinate for sample size shown on the graph. The prefix "A" for each variable is used for convenience to recognize that these variables belong to panel A.

Effect	N	agegr	Panel	A_X	A_X1_plot	A_n_L60	A_x2_plot	A_x2	A_n_G60
agegr	195	<60	A	1	1	195	.	.	.
agegr	186	>=60	A	2	.	.	1.5	1.5	186

Data step to create new variables for Panel A:

```
***** Panel A new variables creation;
```

```
DATA lsmeans2024;
```

```
SET lsmeans2024;
```

```
IF effect="agegr" THEN
```

```
DO;
```

```
IF agegr="<60" THEN DO; A_x=1; A_x1_plot=1;A_n_L60=n;END;
```

```

IF agegr2=">=60" THEN DO; A_x=2;a_x2_plot=1.5;A_x2=1.5;A_n_G60=n; END;
END;

RUN;

```

Panel A template code:

We encapsulate panel A code within `layout` lattice with `rowweights=(0.95 0.05)`. The rowweight 0.95 is used for the main plot and rowweight 0.05 space is used to show the sample size.

Next, the main plot is rendered within first `layout overlay block` and sample size is rendered within second `layout overlay block`. We used `entry` statements to designate panel identifiers and P values. In the first `layout overlay block`, we include two `scatterplot` statements to display two means and lines for confidence intervals.

In the second `layout overlay block` we use two `textplot statements` to show two sample size. The two `textplot statements` allow us flexibility to color code them same color as that we used for the scatterplot.

The discrete X and Y coordinate variables for each line enables to assign the preferred attributes to the line object such as line color and marker color using `markerattrs` and `errorbarattrs` options (for example here we coded two line colors blue/red, dashed/solid line and filled/open marker symbols).

```

***** Panel A;
layout lattice / columndatarange=union rowweights=(0.95 0.05 );
    layout overlay / cycleattrs=true
        yaxisopts=(Label="Responder rate(%)" griddisplay=on
            linearopts=( tickvaluelist=( 45 50 55 60 65 70 )viewmin=45 viewmax=70
                minorticks=yes ) )
        xaxisopts=(Label="Age" offsetmin=0.05 offsetmax=0.05 griddisplay=off
            display=( line ticks tickvalues)
            linearopts=(tickvaluelist=(1 1.5 )
                viewmin=1.0 viewmax=1.8 tickdisplaylist=('Age<60' 'Age>=60' )));

entry halign=left "A" / valign=top location=outside textattrs=(size=16pt color=Black);
entry halign=left "Age P=0.56" / valign=top location=inside textattrs=(size=9pt color=Black weight=bold);

scatterplot x=a_x1_plot y=responder_mean / yerrorlower=responder_lower yerrorupper=responder_upper
markerattrs=(color=blue size=10pt symbol=circlefilled ) errorbarattrs=(color=blue );

scatterplot x=a_x2_plot y=responder_mean / yerrorlower=responder_lower yerrorupper=responder_upper
markerattrs=(color=red size=10pt symbol=circle ) errorbarattrs=(color=red pattern=dash );

endlayout;

***** sample size;
layout overlay / walldisplay=none
    yaxisopts=(display=(tickvalues) linearopts=(tickvaluelist=(0 )

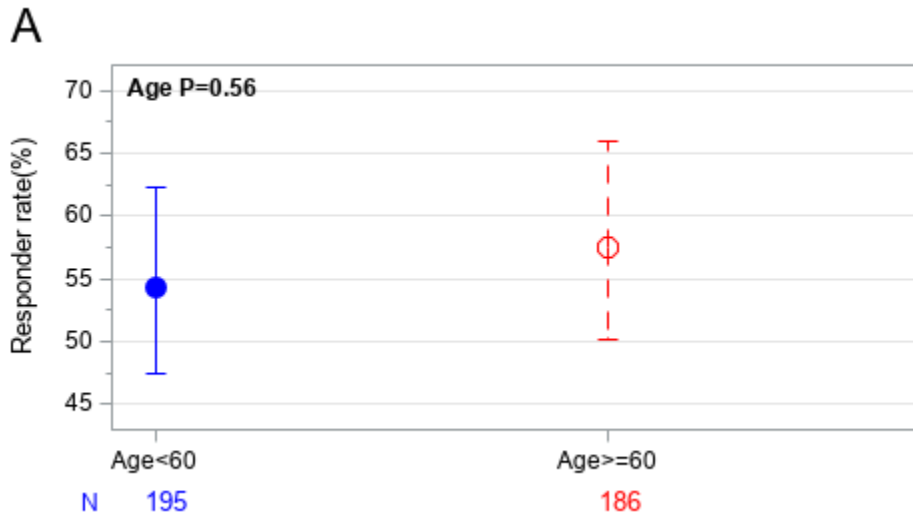
```

```

TICKdisplayList=("N" )
offsetmin=0 offsetmax=0 TICKVALUEATTRS=(color=blue))
    xaxisopts=(offsetmin=0.05 offsetmax=0.05 display=none
    linearopts=(tickvaluelist=(1 1.5) viewmin=1.0 viewmax=1.8));
textplot x=a_x1_plot y=eval(a_x*0) text=A_n_L60/textattrs=(color=blue size=10pt);
textplot x=a_x2_plot y=eval(a_x*0) text=A_n_G60/textattrs=(color=red size=10pt);
endlayout;
endlayout; ***** end plot A lattice;

```

Panel A plot:



Step 6: Panel B plotting responder rate means and 95% confidence interval for males and females.

Here we want to display two members of sex cluster.

We use lsmeans2024.sas7bdat dataset (where effect="SEX") to plot mean and 95% CI and sample size. The data is shown below.

Effect	SEX	N	responder_mean	responder_lower	responder_upper
SEX	F	212	62.4	56.1	69.4
SEX	M	169	50.1	42.7	58.9

Added new variables in data step (similar to panel A shown in step 5 above)

Here we create two sample size variables for males and females (**B_n_F**, **B_n_M**), 2 variables for X-coordinate (**B_X1_plot**, **B_X2_plot**). The sequence variable B_X was created as x-coordinate for sample size shown on the graph. The prefix "B" for each variable here is used for convenience to recognize that these variables belong to panel B.

Effect	SEX	Panel	B_X	B_X1_plot	B_n_F	B_x2_plot	Bx2	B_n_M
SEX	F	B	1	1	195	.	.	.
SEX	M	B	2	.	.	1.5	1.5	186

Data step to create new variables for Panel B:

```

***** Panel B new variables;

DATA lsmeans2024;
SET lsmeans2024;
IF effect="SEX" THEN
  DO;
  IF SEX="F" THEN DO; B_x=1; b_x1_plot=1;B_n_F=n; END;
  IF SEX="M" THEN DO; B_x=2;b_x1_plot=1.5;B_n_M=n; END;
  END;

RUN;

```

Panel B template code is similar to Panel A. Here we used `blockplot` statement to render the sample size. We did not want to color code sample size values differently so we could use `blockplot` statement.

```

layout lattice / columndatarange=union rowweights=(0.95 0.05 );
layout overlay / cycleattrs=true

  yaxisopts=(Label="Responder rate(%)" griddisplay=on      linearopts=( tickvaluelist=(20 30 40 50 60 70
80 )viewmin=40 viewmax=80 minorticks=yes ) )
  xaxisopts=(Label="sex" offsetmin=0.05 offsetmax=0.05 griddisplay=off
display=( line ticks tickvalues)
linearopts=(tickvaluelist=(1 1.5 ) viewmin=1.0 viewmax=1.8
tickdisplaylist=('Females' 'Males' )));

entry halign=left "B" / valign=top location=outside textattrs=(size=16pt color=Black);
entry halign=left "Sex P=0.02" / valign=top location=inside textattrs=(size=9pt color=Black weight=bold);

scatterplot x=B_x1_plot y=responder_mean / yerrorlower=responder_lower yerrorupper=responder_upper
name="B1" legendlabel="Females"
markerattrs=(color=blue size=10pt symbol=circlefilled )errorbarattrs=(color=blue );

scatterplot x=B_x2_plot y=responder_mean / yerrorlower=responder_lower yerrorupper=responder_upper
name="B2" legendlabel="Males"

```

```

markerattrs=(color=red size=10pt symbol=circle ) errorbarattrs=(color=red pattern=dash );
endlayout;

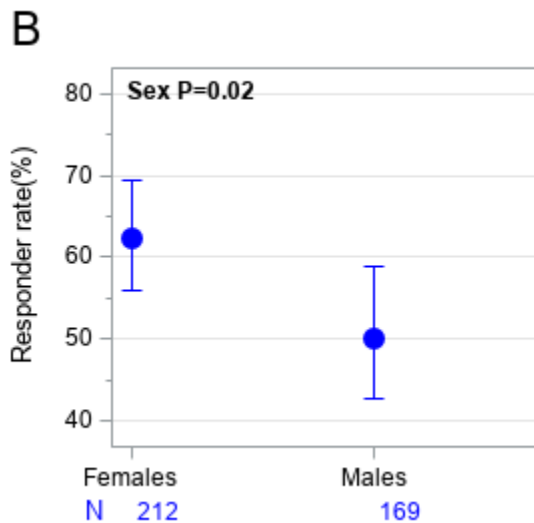
blockplot x=B_x1_plot block=n / display=(values label) valuehalign=center label="N" repeatedvalues=true
valueattrs=(color=blue) labelattrs=(color=blue);

endlayout;

```

A block plot contains information about X locations and text values to be associated with corresponding intervals along the X axis. There is no Y-axis information in this plot. In the **BLOCKPLOT** statement, the **BLOCK=** argument is used to reference text values from a column. The resulting graph displays those values within the plot wall. This example shows how a block plot can be specified with a scatter plot within an **OVERLAY** layout.

Panel B Plot



Step 7: Panel C Plotting 8 responder rate means and 95% confidence interval by 4 treatments and 2 age groups.

Here we want to display two members of age group clusters for each of the 4 treatments.

We use `lsmeans2024.sas7bdat` dataset (where effect=" TRTP*agegr") to plot mean and 95% CI and sample size. The data is shown below.

Effect	TRTP	agegr	N	responder_mean	responder_lower	responder_upper
TRTP*agegr2	BMAC	<60	46	63.1	50.5	78.7
TRTP*agegr2	CSI	<60	52	38.4	26.3	56.1
TRTP*agegr2	SVF	<60	43	62.4	49.4	78.9

TRTP*agegr2	UCT	<60	54	57.7	45.7	72.7
TRTP*agegr2	BMAC	>=60	49	71.2	59.5	85.2
TRTP*agegr2	CSI	>=60	45	52.1	37.1	73.1
TRTP*agegr2	SVF	>=60	48	54.8	41.7	72.0
TRTP*agegr2	UCT	>=60	44	54.0	40.9	71.2

Added new variables in data step: Here we want to display 8 means. We want to display two age groups within a treatment next to each other. We achieved this by creating two X-coordinate variables.

For example treatment BMAC <60 age is assigned X co-ordinate variable **C_x_L60=1.1** and >=60 age gets **C_X_G60=1.2**. Similarly for SVF treatment <60 age X co-ordinate variable C_x_L60=2.1 and >=60 age gets C_X_G60=2.2. The sequence variable C_X was created as x-coordinate for sample size shown on the graph.

Effect	TRTP	agegr	N	Panel	c_x	C_x_L60	C_X_G60
TRTP*agegr2	BMAC	<60	46	C	1.1	1.1	.
TRTP*agegr2	CSI	<60	52	C	4.1	4.1	.
TRTP*agegr2	SVF	<60	43	C	2.1	2.1	.
TRTP*agegr2	UCT	<60	54	C	3.1	3.1	.
TRTP*agegr2	BMAC	>=60	49	C	1.2	.	1.2
TRTP*agegr2	CSI	>=60	45	C	4.2	.	4.2
TRTP*agegr2	SVF	>=60	48	C	2.2	.	2.2
TRTP*agegr2	UCT	>=60	44	C	3.2	.	3.2

Panel C Plot Template Code:

This code is similar to Panel B. We used 2 scatterplot statements to display 8 means and 8 lines. The clustered effect was achieved using x-coordinate values close to each other within the group. We created 4 blocks for clarity by using four **REFERENCELINE** statements. We then labelled 4 blocks on the top edge to display the treatment using additional four **REFERENCELINE** statements with **CURVELABEL** option. We use **DISCRETELEGEND** statement to show legend. We used 2 **BLOCKPLOT** statements to render the sample size. We styled scatterplot and blockplot statements for age group <60 by blue color and age group >=60 by red color.

```

layout lattice / columns=1 rows=1 columngutter=5
               rowgutter=15 columnweights=(0.5 0.5 );

               *****;

**** PLOT 3 ;
layout lattice / columndatarange=union rowweights=(0.90 0.05 0.05 );
layout overlay / cycleattrs=true
               yaxisopts=(Label="Responder rate(%)" griddisplay=on
display=( line ticks tickvalues)
               linearopts=( tickvaluelist=(20 30 40 50 60 70 80 ) viewmin=20 viewmax=90 minorticks=yes ) )
               xaxisopts=(Label="Treatment" offsetmin=0.05 offsetmax=0.05 griddisplay=off
display=( line )
               linearopts=(tickvaluelist=(1.1 2.1 3.1 4.1)
               viewmin=1 viewmax=5
               tickdisplaylist=('BMAC' 'SVF' 'UCT' 'CSI' ));

```

```

entry halign=left "C" / valign=top location=outside textattrs=(size=16pt color=Black);
entry halign=left "Age by Treatment Interaction P=0.47 " / valign=bottom location=inside textattrs=(size=9pt
color=Black weight=bold);

scatterplot x=C_x_L60 y=responder_mean / yerrorlower=responder_lower yerrorupper=responder_upper
name='C1'

    legendlabel="Age<60"
    markerattrs=(color=blue size=10pt symbol=circlefilled )
    errorbarattrs=(color=blue pattern=solid);

scatterplot x=C_x_G60 y=responder_mean / yerrorlower=responder_lower yerrorupper=responder_upper
name='C2'

    legendlabel="Age>=60"
    markerattrs=(color=red size=10pt symbol=circle )
    errorbarattrs=(color=red pattern=dash );

referenceline x=1.9 / lineattrs=(color=grey pattern=dot);
referenceline x=2.9 / lineattrs=(color=grey pattern=dot);
referenceline x=3.9 / lineattrs=(color=grey pattern=dot);

referenceline x=1.5/ CURVELABEL="BMAC" lineattrs=(color=white) CURVELABELattrs=(size=9pt);
referenceline x=2.5/ CURVELABEL="SVF" lineattrs=(color=white) CURVELABELattrs=(size=9pt);
referenceline x=3.5/ CURVELABEL="UCT" lineattrs=(color=white) CURVELABELattrs=(size=9pt);
referenceline x=4.5/ CURVELABEL="CSI" lineattrs=(color=white) CURVELABELattrs=(size=9pt);

discretelegend 'C1' 'C2' / title="" across=2 border=false location=outside halign=left valign=top;

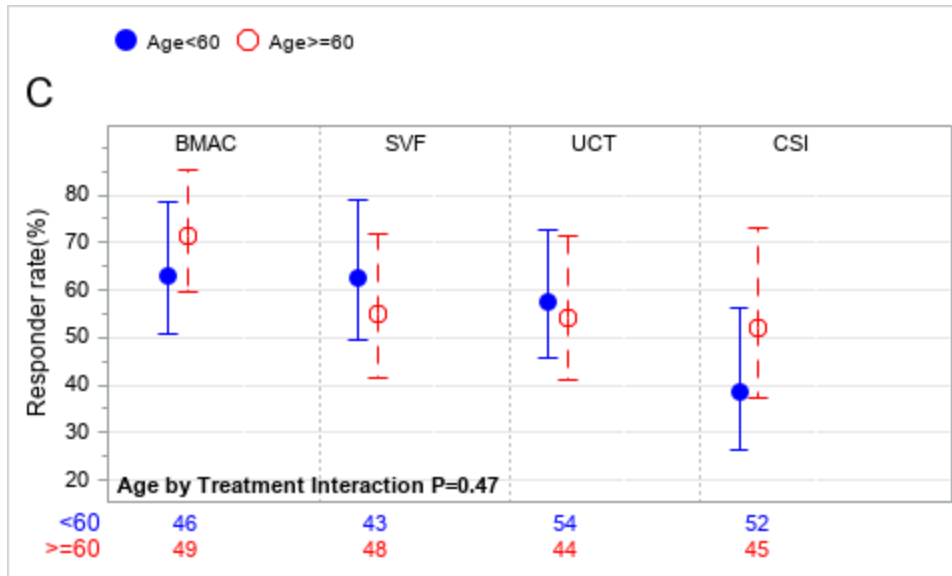
    endlayout;

    blockplot x=eval(C_x_L60-0.0) block=n / display=(values label) valuehalign=center label="<60"
repeatedvalues=true valueattrs=(color=blue) labelattrs=(color=blue);

    blockplot x=eval(C_x_G60-0.1) block=n / display=(values label) valuehalign=center label=">=60"
repeatedvalues=true valueattrs=(color=red) labelattrs=(color=red);

endlayout;

```



Step 7: Panel D Plotting 16 responder rate means and 95% confidence interval by 4 treatments and 2 age groups for males and females.

Here we want to display two members of age group and sex clusters for each of the 4 treatments.

We use lsmeans2024.sas7bdat dataset (where effect=" TRTP*SEX*agegr") to plot 16 mean and 95% CI and sample size. The data is shown below.

Effect	Trtp	Sex	agegr	N	responder_mean	responder_lower	responder_upper
TRTP*SEX*agegr2	BMAC	F	<60	22	63.6	46.4	87.3
TRTP*SEX*agegr2	BMAC	M	<60	24	62.5	45.8	85.2
TRTP*SEX*agegr2	BMAC	F	≥60	25	76.0	61.0	94.7
TRTP*SEX*agegr2	BMAC	M	≥60	24	66.7	50.2	88.5
TRTP*SEX*agegr2	SVF	F	<60	21	57.1	39.5	82.8
TRTP*SEX*agegr2	SVF	M	<60	22	68.2	51.3	90.7
TRTP*SEX*agegr2	SVF	F	≥60	30	60.0	44.8	80.4
TRTP*SEX*agegr2	SVF	M	≥60	18	50.0	31.5	79.4
TRTP*SEX*agegr2	UCT	F	<60	32	56.3	41.4	76.4
TRTP*SEX*agegr2	UCT	M	<60	22	59.1	41.7	83.7
TRTP*SEX*agegr2	UCT	F	≥60	24	58.3	41.6	81.8
TRTP*SEX*agegr2	UCT	M	≥60	20	50.0	32.3	77.5
TRTP*SEX*agegr2	CSI	F	<60	31	51.6	36.7	72.6
TRTP*SEX*agegr2	CSI	M	<60	21	28.6	14.5	56.2
TRTP*SEX*agegr2	CSI	F	≥60	27	81.5	68.1	97.5

TRTP*SEX*agegr2	CSI	M	>=60	18	33.3	17.3	64.1
-----------------	-----	---	------	----	------	------	------

Added new variables in data step: Here we want to display 16 means. We want to display four groups within a treatment next to each other. We achieved this by creating four X-coordinate variables. The x-coordinate values are assigned such that 4 vertical lines within a treatment can be clustered on the plot. Moreover, within a treatment we want to show Age<60 (M and F) as one cluster and Age ≥ 60 (M and F) as another cluster.

For example for two Age<60 lines for BMAC treatment, x-coordinates are 1.1 and 1.2 for females and males respectively and for two Age≥60 lines within BMAC treatment, x-coordinates are 1.6 and 1.7 for females and males respectively. This way Age<60 cluster (2 vertical male and female lines) and Age≥60 cluster (2 vertical male and female lines) are next to each other at the same time two age clusters are kept apart by 0.4 unit (1.2-1.6=0.4)

We create four x-coordinate variables for 4 lines: For females (d_x_f_L60, d_x_f_G60) and for males (d_x_m_L60, d_x_m_G60) as follows:

BMAC Females <60 age is assigned X co-ordinate variable d_x_f_L60=1.1.

BMAC Males <60 age is assigned X co-ordinate variable d_x_L60=1.2.

BMAC Females ≥ 60 age is assigned X co-ordinate variable d_x_f_G60=1.6

BMAC Males ≥ 60 age is assigned X co-ordinate variable d_x_G60=1.7.

Next, we illustrate our method for sample size x-coordinates.

We will show two sample size rows, one for females and another for males by using two [blockplot](#) statements below the main plot. [Blockplot](#) statement needs a x-coordinate variable and sample size variable (block=). We create these 2 variables (d_x_F, d_x_M) by using below logic.

If d_x_f_L60 ~= . then d_x = d_x_f_L60;

If d_x_f_G60 ~= . then d_x = d_x_f_G60;

If d_x_M_L60 ~= . then d_x = d_x_M_L60;

If d_x_M_G60 ~= . then d_x = d_x_M_G60;

if effect="TRTP*SEX*agegr2" and SEX ="F" then d_x_F=d_x;;

if effect="TRTP*SEX*agegr2" and SEX ="M" then d_x_M=d_x;;

We follow similar logic for the remaining 3 treatments. The resulting dataset with four x-coordinates for 16 lines and 2 coordinates for sample size lines is shown below.

Effect	Trtp	Sex	agegr	N	Panel	d_x	d_x_f_L60	d_x_f_G60	d_x_m_L60	d_x_G60	d_x_f	d_x_m
TRTP*SEX*agegr2	BMAC	F	<60	22	D	1.1	1.1	.	.	.	1.1	.
TRTP*SEX*agegr2	BMAC	M	<60	24	D	1.2	.	.	1.2	.	.	1.2
TRTP*SEX*agegr2	BMAC	F	>=60	25	D	1.6	.	1.6	.	.	1.6	.
TRTP*SEX*agegr2	BMAC	M	>=60	24	D	1.7	.	.	.	1.7	.	1.7
TRTP*SEX*agegr2	SVF	F	<60	21	D	2.1	2.1	.	.	.	2.1	.
TRTP*SEX*agegr2	SVF	M	<60	22	D	2.2	.	.	2.2	.	.	2.2
TRTP*SEX*agegr2	SVF	F	>=60	30	D	2.6	.	2.6	.	.	2.6	.
TRTP*SEX*agegr2	SVF	M	>=60	18	D	2.7	.	.	.	2.7	.	2.7
TRTP*SEX*agegr2	UCT	F	<60	32	D	3.1	3.1	.	.	.	3.1	.
TRTP*SEX*agegr2	UCT	M	<60	22	D	3.2	.	.	3.2	.	.	3.2
TRTP*SEX*agegr2	UCT	F	>=60	24	D	3.6	.	3.6	.	.	3.6	.

TRTP*SEX*agegr2	UCT	M	>=60	20	D	3.7	.	.	.	3.7	.	3.7
TRTP*SEX*agegr2	CSI	F	<60	31	D	4.1	4.1	.	.	.	4.1	.
TRTP*SEX*agegr2	CSI	M	<60	21	D	4.2	.	.	4.2	.	.	4.2
TRTP*SEX*agegr2	CSI	F	>=60	27	D	4.6	.	4.6	.	.	4.6	.
TRTP*SEX*agegr2	CSI	M	>=60	18	D	4.7	.	.	.	4.7	.	4.7

Panel D Template code:

This code is similar to Panel C. We used 2 scatterplot statements to display 16 means and 16 lines. The clustered effect was achieved using x-coordinate values close to each other within the group. We created 4 blocks for clarity by using four **REFERENCELINE** statements. We then labelled 4 blocks on the top edge to display the treatment using additional four **REFERENCELINE** statements with **CURVELABEL** option. We use **DISCRETELEGEND** statement to show legend. We used two **BLOCKPLOT** statements to render the sample size. We styled scatterplot and blockplot statements for females by blue color and males by red color. We used closed circle for age <60 and open circle for age ≥ 60.

***** Panel D template;

```

layout lattice / columndatarange=union rowweights=(0.90 0.05 0.05 );

layout overlay / cycleattrs=true
    yaxisopts=(Label="Responder rate(%)" griddisplay=on
                display=( label line ticks tickvalues)
                linearopts=(tickvaluelist=(10 20 30 40 50 60 70 80 90 100 ) viewmin=10 viewmax=110
minorticks=yes ) )
    xaxisopts=(Label="Treatment" offsetmin=0.05 offsetmax=0.05 griddisplay=off
                display=( line )
                linearopts=(tickvaluelist=(1.1 2.1 3.1 4.1 5.1 )
                viewmin=1 viewmax=5
                tickdisplaylist=('BMAC' 'SVF' 'UCT' 'CSI' ));

entry halign=left "D" / valign=top location=outside textattrs=(size=16pt color=Black);
entry halign=left "Age by Treatment by Sex interaction P=0.77 " /
valign=bottom location=inside textattrs=(size=9pt color=Black weight=bold);

scatterplot x=d_x_f_L60 y=responder_mean / yerrorlower=responder_lower yerrorupper=responder_upper
markerattrs=(size=10pt symbol=circlefilled color=blue) errorbarattrs=(thickness=2pt color=blue)
name="F_x" legendlabel="F(<60)";
scatterplot x=d_x_m_L60 y=responder_mean / yerrorlower=responder_lower yerrorupper=responder_upper
markerattrs=(size=10pt symbol=circlefilled color=red) errorbarattrs=(thickness=2pt color=red)
name="M_x" legendlabel="M(<60)";

```

```

scatterplot x=d_x_f_G60 y=responder_mean / yerrorlower=responder_lower yerrorupper=responder_upper
markerattrs=(size=10pt symbol=circle color=blue) errorbarattrs=(thickness=2pt color=blue pattern=dash)
name="F2_x" legendlabel="(>=60)";
scatterplot x=d_x_m_G60 y=responder_mean / yerrorlower=responder_lower yerrorupper=responder_upper
markerattrs=(size=10pt symbol=circle color=red) errorbarattrs=(thickness=2pt color=red pattern=dash)
name="M2_x" legendlabel="(>=60)";

***** vertical dotted line;
referenceline x=1.9 / lineattrs=(color=grey pattern=dot);
referenceline x=2.9 / lineattrs=(color=grey pattern=dot);
referenceline x=3.9 / lineattrs=(color=grey pattern=dot);

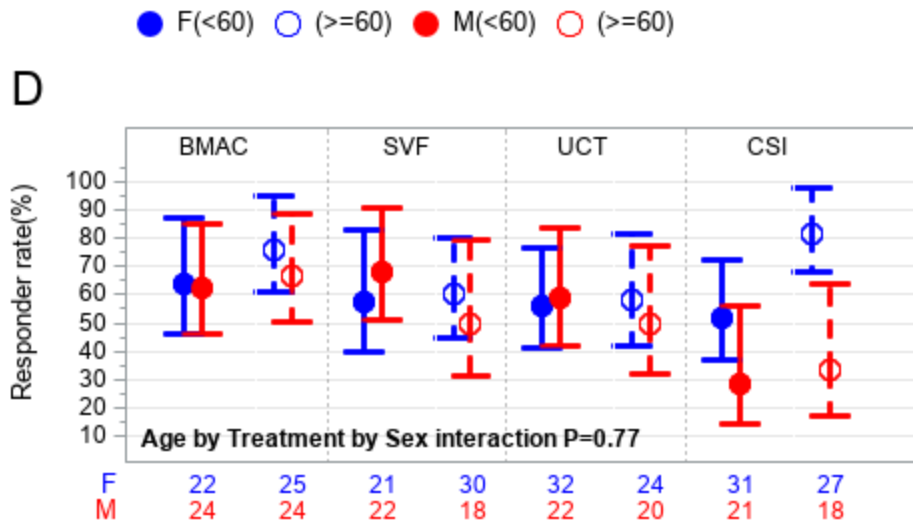
***** treatment header on top;
referenceline x=1.5 / CURVELABEL="BMAC" lineattrs=(color=white) CURVELABELattrs=(size=9pt);
referenceline x=2.5 / CURVELABEL="SVF" lineattrs=(color=white) CURVELABELattrs=(size=9pt);
referenceline x=3.5 / CURVELABEL="UCT" lineattrs=(color=white) CURVELABELattrs=(size=9pt);
referenceline x=4.5 / CURVELABEL="CSI" lineattrs=(color=white) CURVELABELattrs=(size=9pt);

***** Legend;
discretelegend "F_x" "F2_x" "M_x" "M2_x" / title=" across=4 border=false
location=outside halign=left valign=top DISPLAYCLIPPED=FALSE valueattrs=(size=10pt);
endlayout;
***** Female sample size;
blockplot x=d_x_f block=n / display=(values label) valuehalign=center label="F" repeatedvalues=true
valueattrs=(color=blue) labelattrs=(color=blue);
***** Male sample size;
blockplot x=eval(d_x_m-0.1) block=n / display=(values label) valuehalign=center label="M" repeatedvalues=true
valueattrs=(color=red) labelattrs=(color=Red);

endlayout;

```

Panel D Plot is shown below:



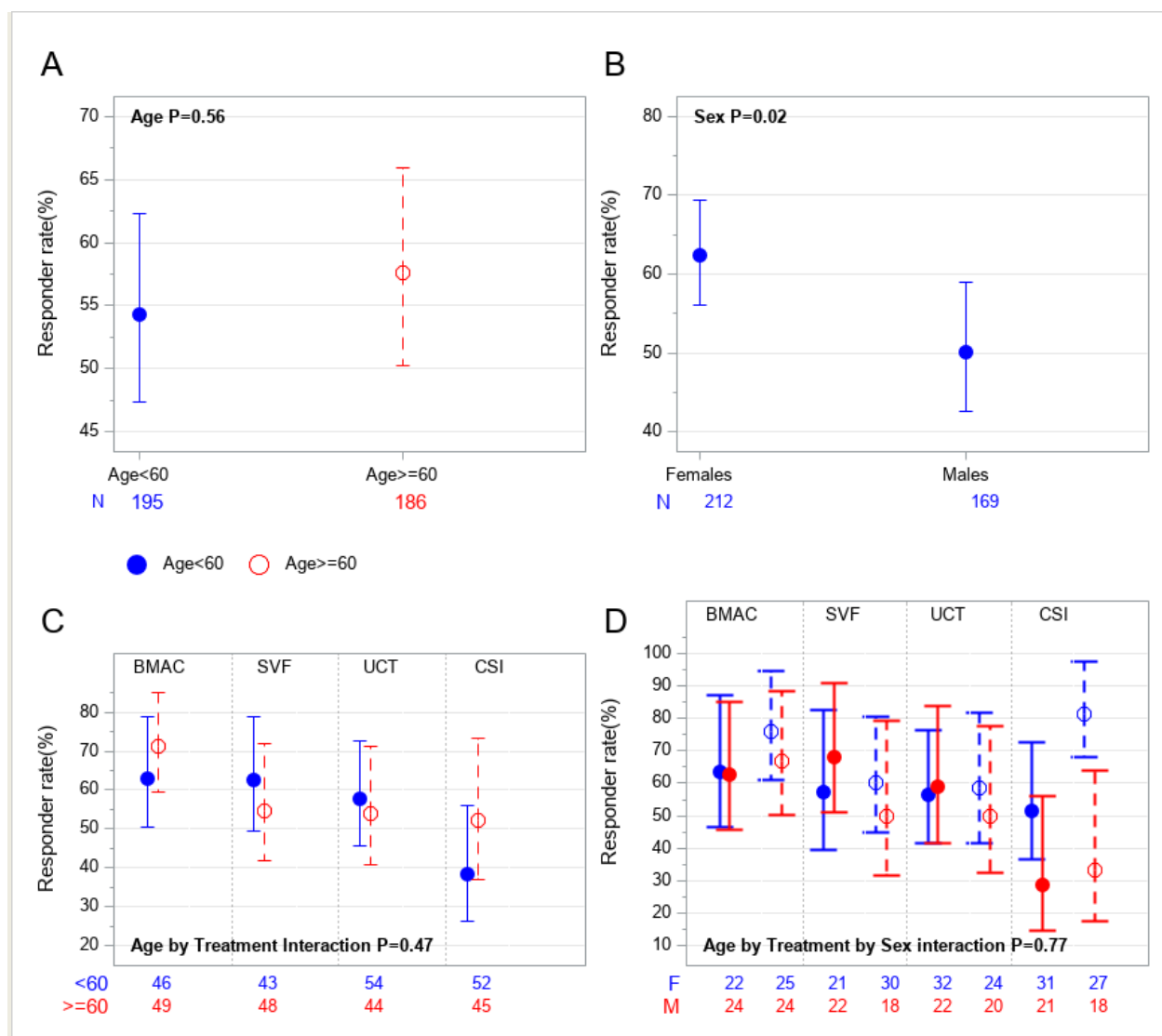
Step 8: If needed sort the data to render all objects in the correct order.

```
PROC SQL;
CREATE TABLE lsmeans2024_sorted AS
SELECT * FROM lsmeans2024
ORDER BY panel, trtpn, agegr2, sex;
QUIT
```

Step 9 Executing the Template to Produce the Graph: To produce a graph, use the SGRENDER procedure.

The **SGRENDER** procedure takes two required arguments: **DATA=** for the input data set and **TEMPLATE=** for the **STATGRAPH** template to be used.

```
ODS GRAPHICS /RESET WIDTH=6in HEIGHT=5in;
PROC SGRENDER DATA=lsmeans2024_sorted TEMPLATE=CL;
RUN;
```



CONCLUSION

The graphs shown above are commonly used in the clinical studies to display results and get a visual message across to the audience. We demonstrated the usage of GENMOD procedure and GTL to display graphs of mean patient outcome using multivariate model. Other procedures such as PROC MIXED, PROC GLIMIX can be used similarly for cross-sectional or longitudinal repeated measure data structure for categorical or continuous outcomes. An example using the longitudinal data is available upon request from the author.

The scientific journals often have very precise requirements for graphs that are submitted with the manuscripts. These graphs can be easily created using GTL. Even where the program is a few lines long, the structure of the graph is very clear and easy to understand. Most of the work is done by the statements supported in the syntax, and no special coding or annotation is required. We kept code very simple. We avoided writing code for attribute maps to control styling line and symbol for cluster display. We achieved cluster visuals by creating discrete x and y-coordinate variables for each group within the cluster and styled it. ODS Graphics gives you incredible power to customize graphs. These examples have many steps, yet each one is quite simple. Furthermore, you can program every step in SAS, ensuring reproducible results.

While authors often find themselves using tools other than SAS® to create these graphs, the combination of the GTL procedure and the Output Delivery System enables authors to create what they need in the same place as they conducted their analysis.

REFERENCES

Mautner, K., Gottschalk, M., Boden, S.D. *et al.* Cell-based versus corticosteroid injections for knee pain in osteoarthritis: a randomized phase 3 trial. *Nat Med* **29**, 3120–3126 (2023).

ACKNOWLEDGMENTS

The authors are grateful to SAS Technical Support for their help in polishing some code snippets shown in this paper.

RECOMMENDED READING

- For complete documentation about ODS and ODS Graphics, see SAS Output Delivery System: User's Guide, SAS Graph Template Language: Reference, SAS ODS Graphics: Procedures Guide, and SAS Graph Template Language: User's Guide.
- For more examples and more documentation, see Chapter 20, "Using the Output Delivery System"; Chapter 21, "Statistical Graphics Using ODS"; Chapter 22, "ODS Graphics Template Modification".

CONTACT INFORMATION

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

Neeta V Shenvi
nshenvi@emory.edu