When Simpler is Better – Visualizing Laboratory Data Using "SG Procedures"

Wei Cheng, Isis Pharmaceuticals, Inc., Carlsbad, CA

ABSTRACT

In SAS® 9.2, SAS/GRAPH introduces a family of new procedures to create stand-alone graphs that use the ODS Statistical Graphics infrastructure and are designed to use ODS styles. These new "SG procedures" include SGPLOT, SGPANEL, and SGSCATTER. With a simple and clear syntax, procedure-quality plots can be generated to assist your data exploration and presentation. This paper provides examples that illustrate how you can use these new procedures to examine and visualize laboratory data.

INTRODUCTION

One of the most important component s of clinical research data is laboratory data, because the results of laboratory tests contain indicators of drug safety or efficacy. Clinicians and statisticians closely monitor the laboratory data during the course of clinical trial to find the best diagnosis and treatment. Graphical representation of laboratory data is often used to provide visual information on the trend or comparison of laboratory tests among different treatment groups of subjects, as well as presenting the treatment safety or efficacy observed in a study.

Prior to the introduction of "SG procedures" in SAS 9.2, we had to generate statistics data from statistical procedures; configure the appropriate fonts and image types by GOPTION; use the traditional SAS/GRAPH procedures, statements, and annotation to generate the plots; and combine the plots into a panel by PROC GREPLAY. By using the features of the new procedures SGPLOT, SGPANEL, and SGSCATTER, the process of generating graphs to visualize laboratory data has been greatly simplified.

I will use laboratory data from a clinical study to generate commonly used statistical graphs which can assist clinicians to understand the data. There are sixteen subjects in the study, results from four laboratory tests creatinine, platelet, red blood cell count, and white blood cell count were collected for analysis. The same code can be applied to bigger clinical trials which contain more subjects and more laboratory tests. By using the new "SG procedures", just a few lines of code can generate much better plots than plots generated by traditional SAS/GRAPH language with hundreds of lines of code.

EXAMPLE 1: SPAGHETTI PLOTS

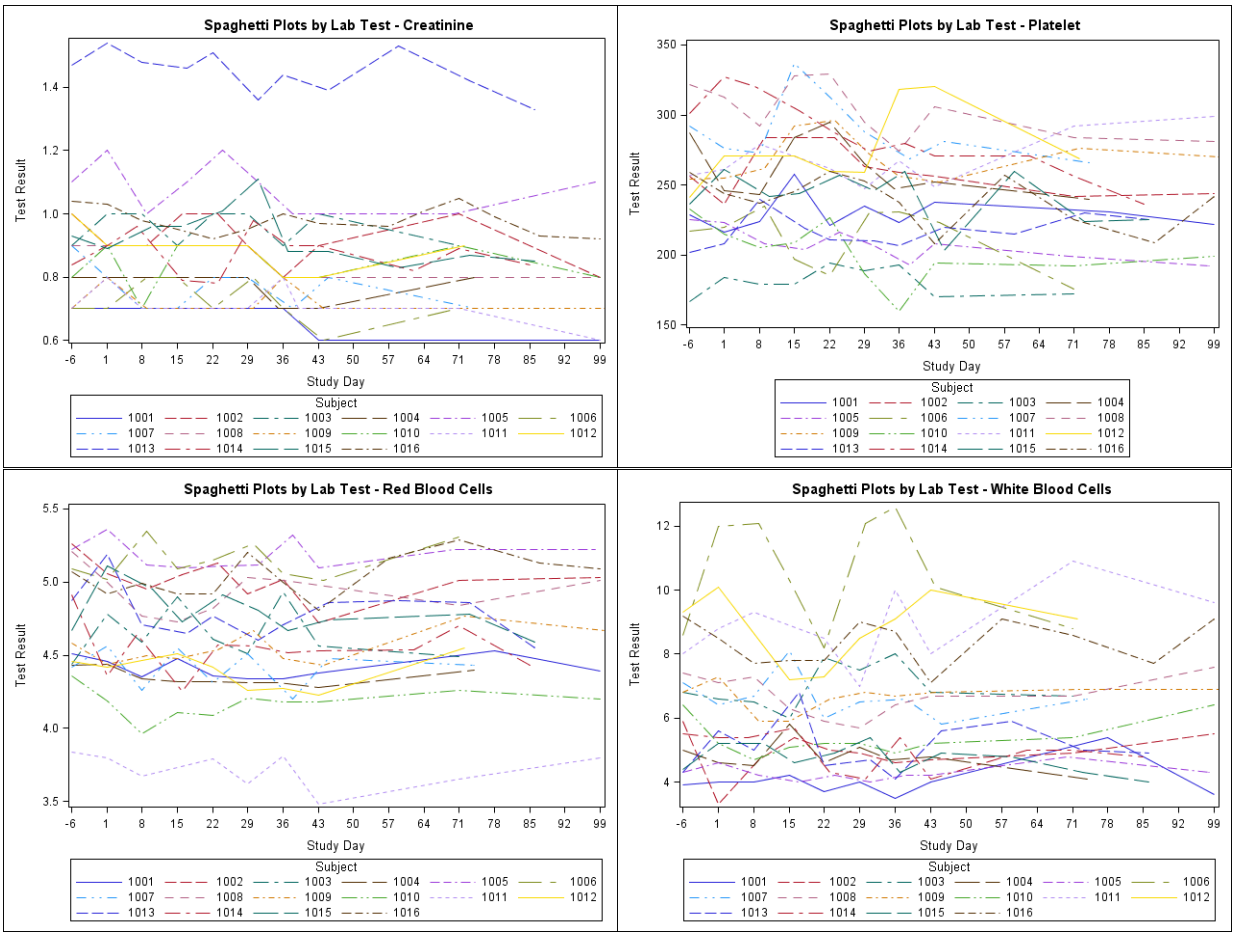
Spaghetti plots are several line graphs that have been overlaid in a single graph. Sometimes it is helpful to plot each subject's raw data at each time point to assess the trends or patterns. If the individual lines are not all in the same direction, then subjects are showing different responses to the treatment, which will lead to further examination.

PROGRAM 1

```
options nobyline;

title "Spaghetti Plots by #byvar1 - #byval1";

proc sgplot data = labdata;
    by labtest;
    series x = visit y = result / group = subjid;
    xaxis values = (-6 to 105 by 7);
run;
```

Figure 1 displays four spaghetti plots for four laboratory tests generated from the proceeding program:

FIGURE 1



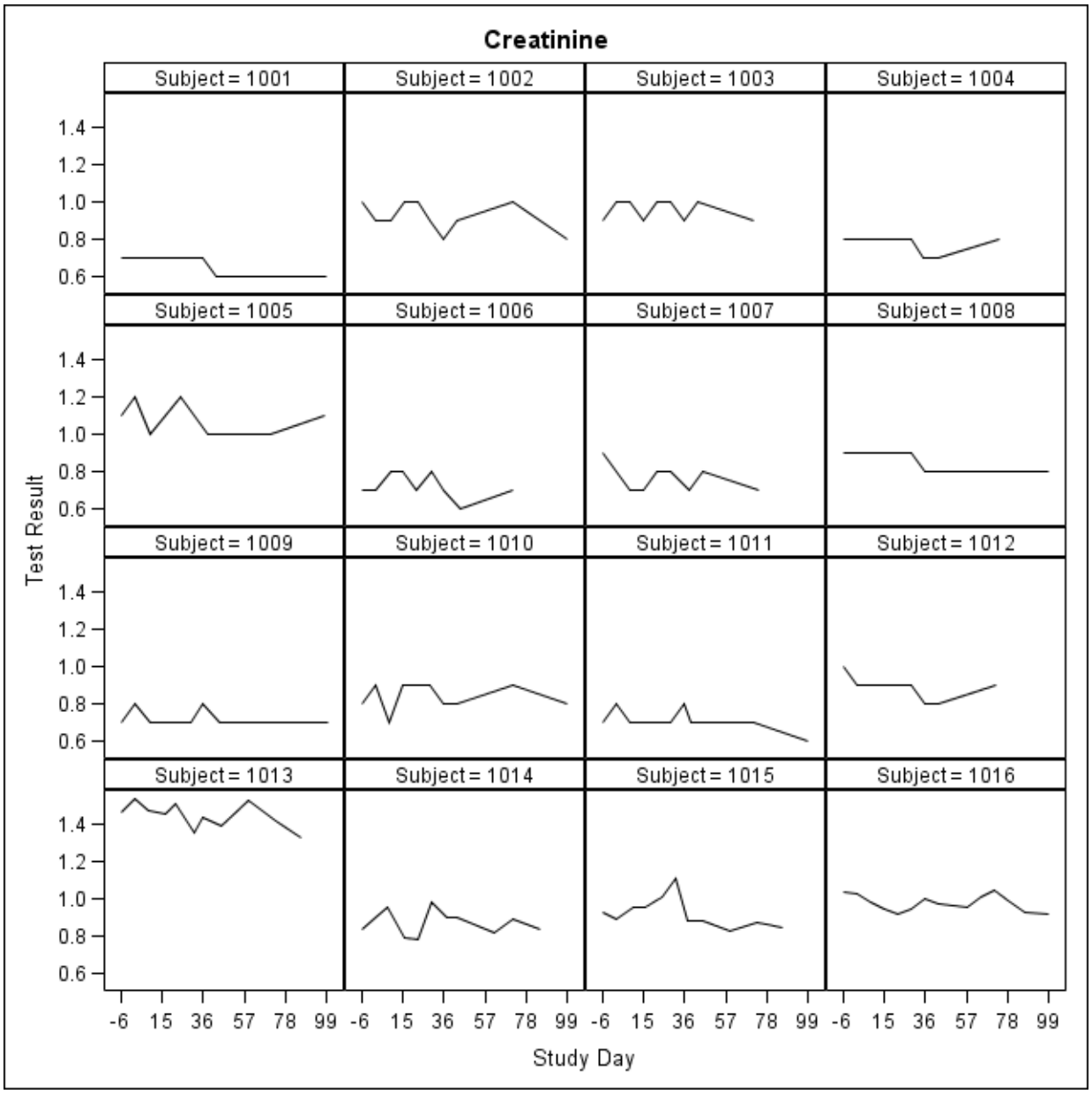
EXAMPLE 2: A PANEL OF LINE PLOTS FOR ONE LABORATORY TEST

If you feel the spaghetti plots are too crowded, you can separate them to a panel of individual line plots. Program 2 will generate a panel of plots for creatinine, as shown in Figure 2. The plots for all the subjects are sharing a same axis scale for comparison.

PROGRAM 2

```
title "Creatinine";

proc sgpanel data = labdata (where = (labtest = "Creatinine"));
    panelby subjid / rows=4 columns=4;
    series x = visit y = result;
    colaxis values = (-6 to 105 by 7);
run;
```

FIGURE 2



EXAMPLE 3: A PANEL OF LINE PLOTS FOR ONE SUBJECT

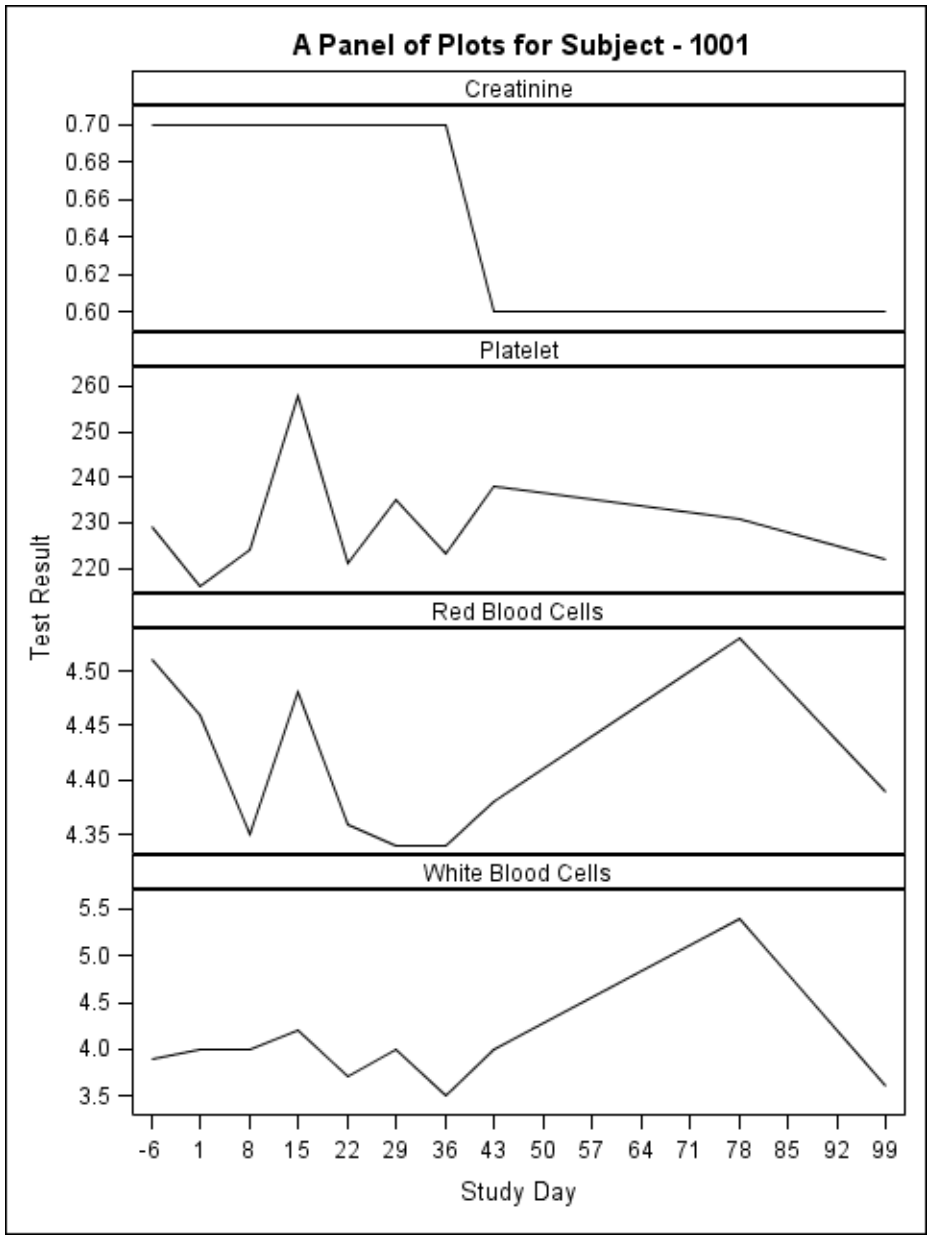
On the other hand, if you want to examine all the laboratory test results together for one subject, you can use Program 3 to generate a panel of plots for each subject. Figure 3 shows one of the sixteen panels generated by Program 3. The plots for all the test results are sharing a same x-axis scale as study day from first treatment dose.

PROGRAM 3

```
options nobyline;

title "A Panel of Plots for #byvar1 - #byval1";

proc sgpanel data = labdata;
```

```
    panelby labtest / columns = 1 rows = 4 uniscale = column novarname;
    series x = visit y = result;
    by subjid;
    colaxis values = (-6 to 105 by 7);
run;
```

FIGURE 3



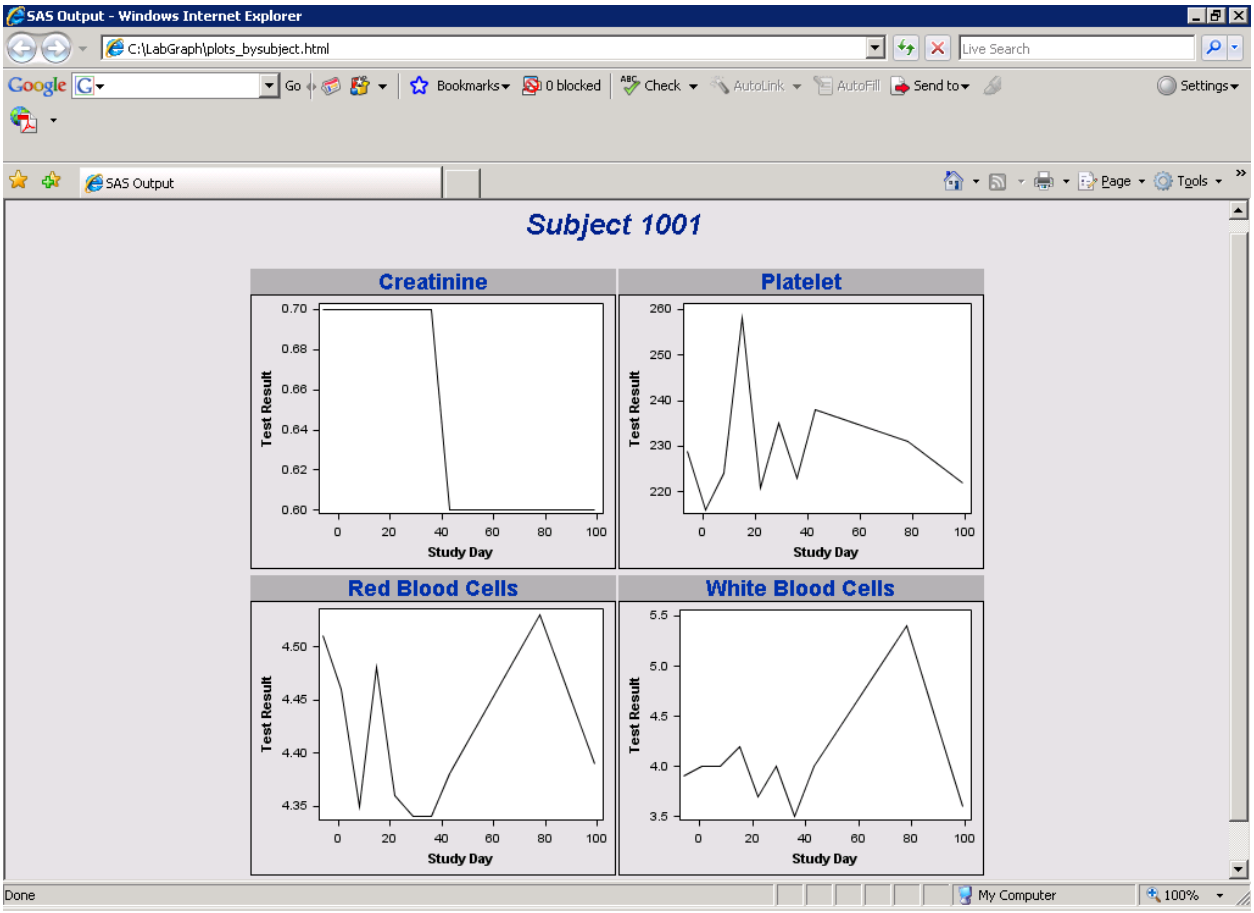
EXAMPLE 4: A 2X2 PANEL OF LINE PLOTS FOR ONE SUBJECT

Figure 3 shows a panel which contains one column and four rows, if you want to have a 2x2 panel which has two columns and two rows to show the four laboratory tests, you can combine PROC SGPLOT with TAGSET HTMLPANEL, so each laboratory test has their own Y-axis scale. The resulting plots, after submitting Program 4, are shown in Figure 4.

PROGRAM 4

```
title "Subject 1001";

ods tagsets.htmlpanel nogtitle file = "plots_bysubject.html";

ods tagsets.htmlpanel event = panel(start) options(panelcolumns = "2" bylabels =
"no");

proc sgplot data = labdata (where = (subjid = "1001"));
    by labtest;
    series x = visit y = result;
run;

ods tagsets.htmlpanel event = panel(finish);

ods tagsets.htmlpanel close;
```

FIGURE 4



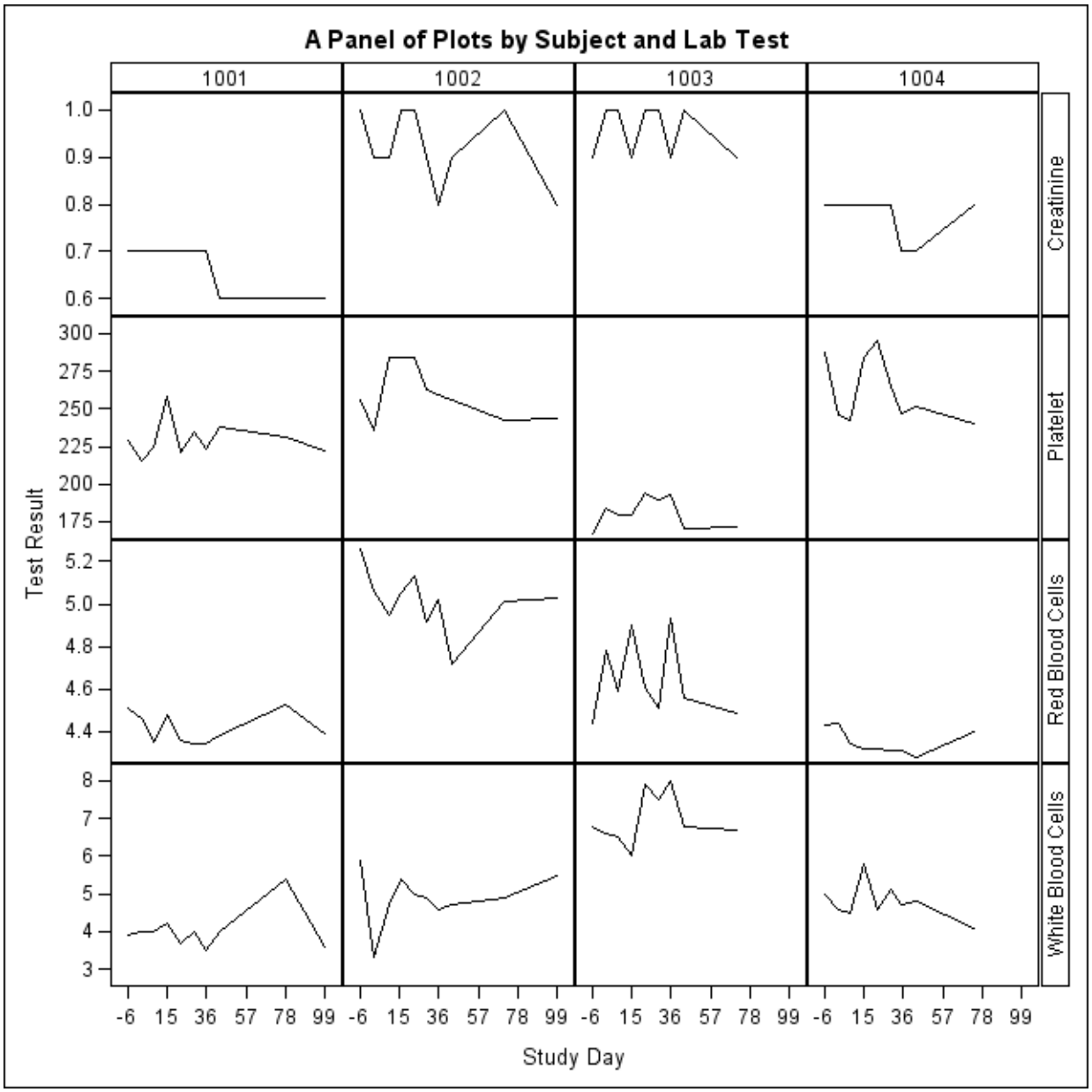
EXAMPLE 5: A PANEL OF LINE PLOTS BY SUBJECT AND LABORATORY TEST

By using the LATTICE layout, a panel of plots by subject and laboratory test can be generated by Program 5. The subject identifications are displayed above the columns, laboratory tests are displayed to the right side of the rows, as shown in Figure 5.

PROGRAM 5

```
title "A Panel of Plots by Subject and Lab Test";
proc sgpanel data = labdata;
    panelby subjid labtest / layout = lattice uniscale = column
                             columns = 4 rows = 4 novarname;
    series x = visit y = result;
    colaxis values = (-6 to 105 by 7);
run;
```

FIGURE 5



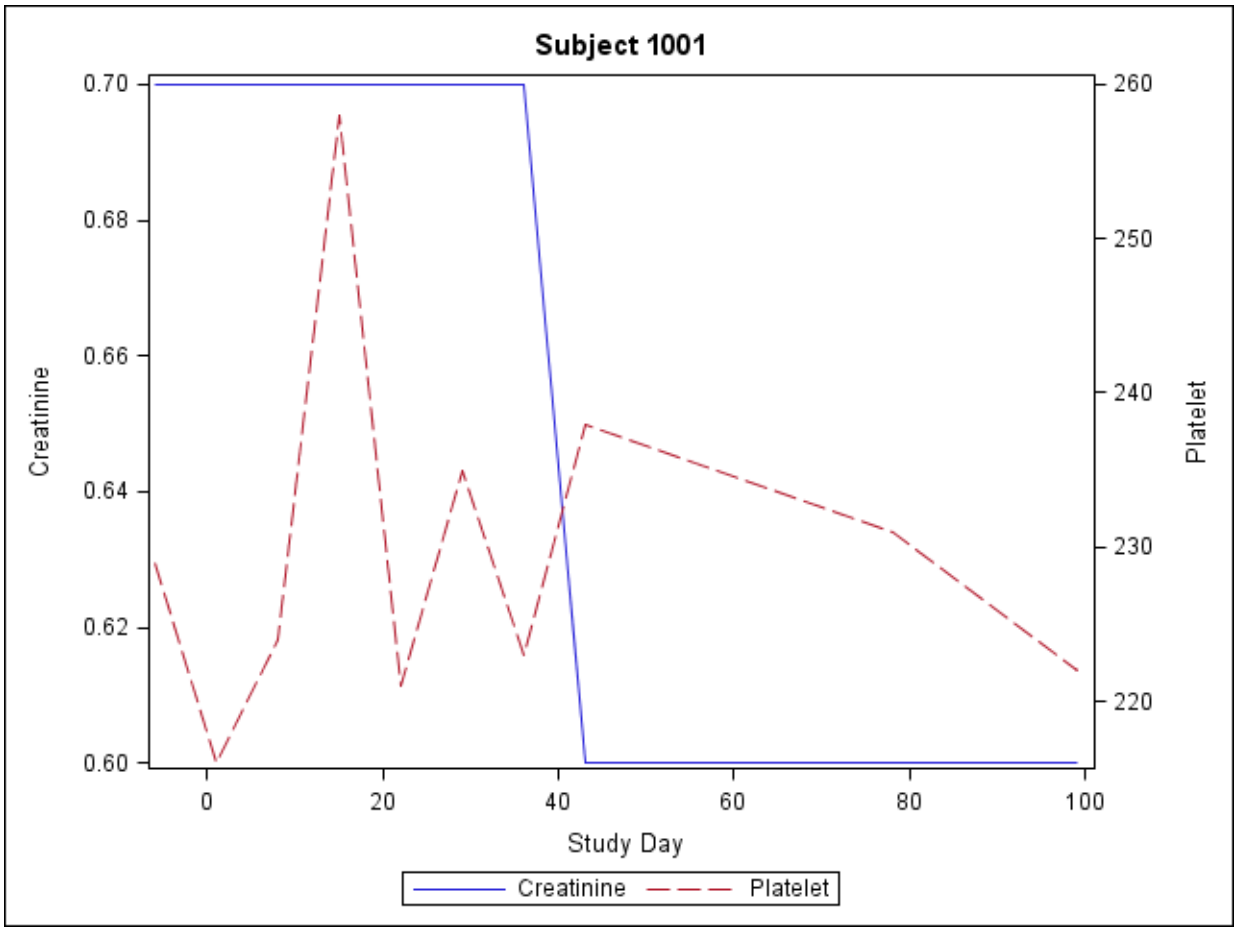
EXAMPLE 6: TWO PLOTS OVERLAID IN A SINGLE GRAPH

When line plots of two laboratory tests are overlaid in a single graph, it's easier to explore the relationship of changes over time. Each line plot is assigned to a different vertical axis, as shown in Figure 6. In Program 6, PROC TRANSPOSE was used to prepare the data set for PROC SGPLOT.

PROGRAM 6

```
proc sort data = labdata (keep = subjid labtest visit result);
    by visit;
    where subjid = "1001" and labtest in ("Creatinine" "Platelet");
run;

proc transpose data = labdata out = lab4plot;
    by visit;
    var result;
    id labtest;
run;

title "Subject 1001";

proc sgplot data = lab4plot;
    series x = visit y = creatinine;
    series x = visit y = platelet / y2axis;
run;
```

FIGURE 6



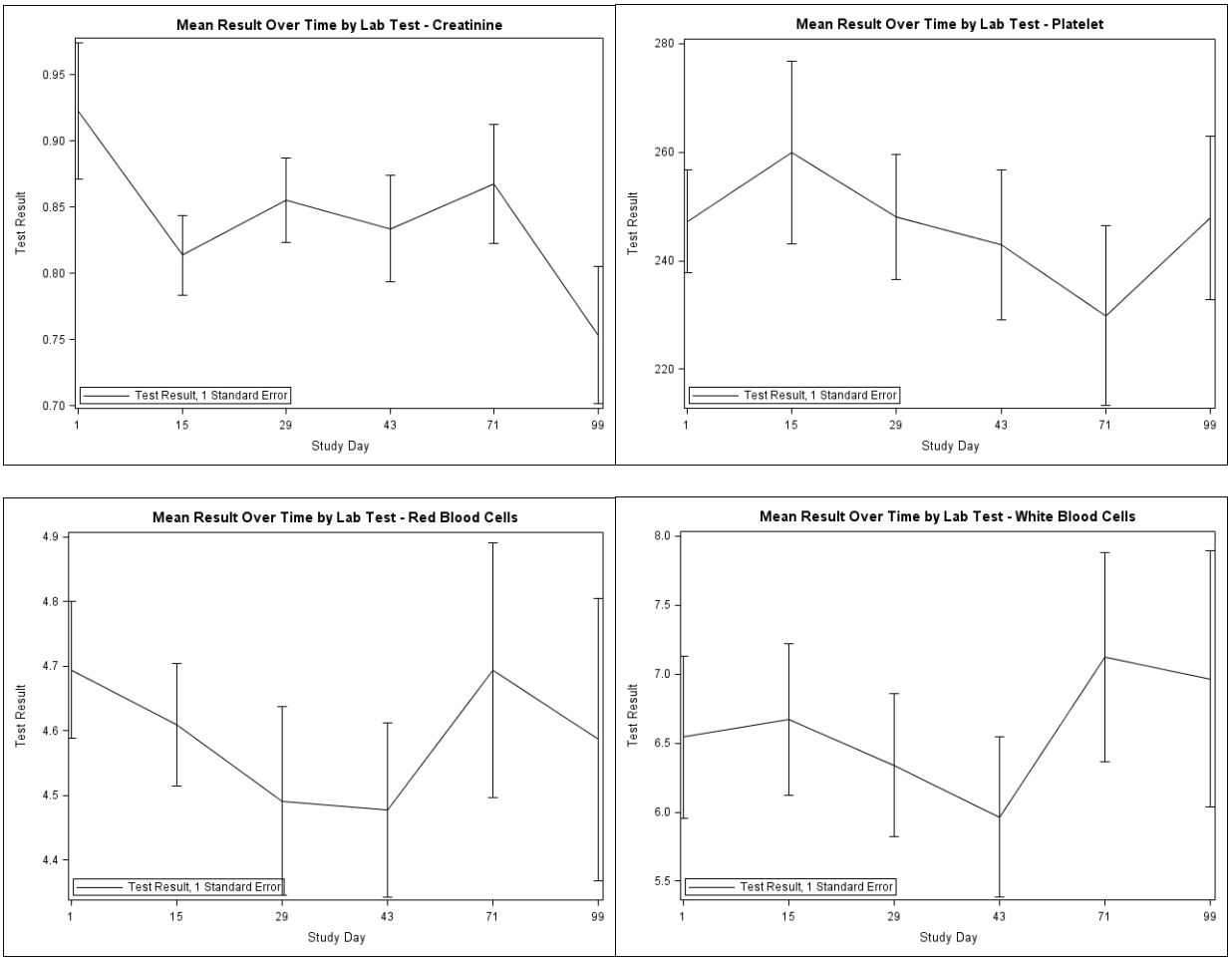
EXAMPLE 7: GROUP MEAN OVER TIME

Figure 7 shows mean laboratory test results of repeated measures over time. The bars on the graphs show the standard errors of the mean values.

PROGRAM 7

```
proc sort data = labdata; by labtest visit; run;

options nobyline; title "Mean Result Over Time by #byvar1 - #byval1";

proc sgplot data = labdata;

    vline visit / response = result stat = mean limitstat = stderr;

    keylegend / location = inside position = bottomleft;

    by labtest; where visit in (1 15 29 43 71 99);

run;
```

FIGURE 7



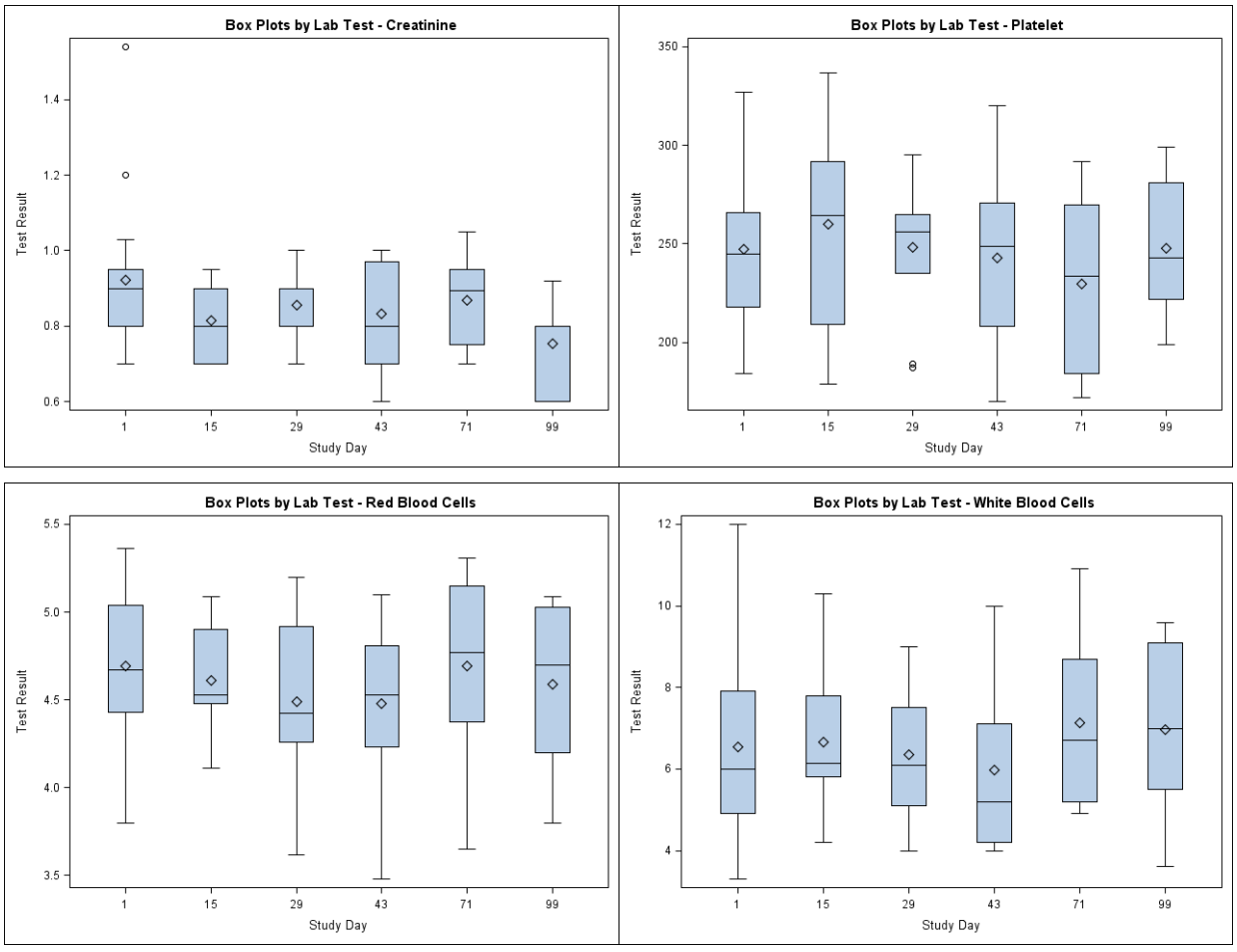
EXAMPLE 8: BOX PLOTS

Box and whisker plot will show central location and outliers of data, it's good for exploring the skewness of the data. Figure 8 shows the median, IQR, and range of laboratory test results of repeated measures over time.

PROGRAM 8

```
options nobyline;

title "Box Plots by #byvar1 - #byval1";

proc sgplot data = labdata;
    by labtest;
    vbox result / category = visit;
run;
```

FIGURE 8



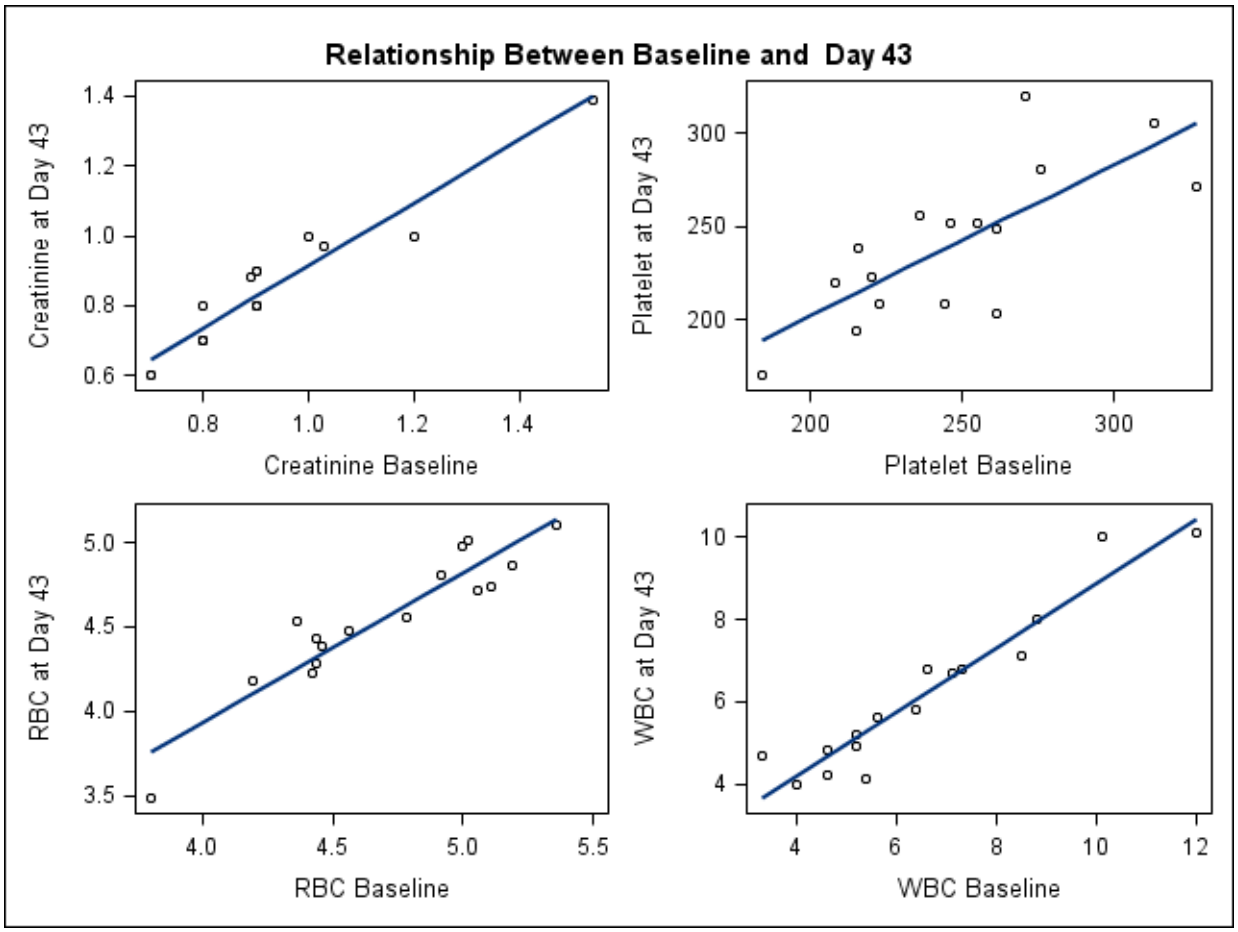
EXAMPLE 9: SCATTER PLOTS

A scatter plot can be used to display the relationship between two continuous variables. The relationship can be expressed with the use of regression line. As expected, Figure 9 shows a highly significant relationship between laboratory test results at baseline and study day 43 from baseline.

PROGRAM 9

```
title "Relationship Between Baseline and  Day 43";

proc sgscatter data = lab4plot;
    plot creatinine43*creatinine1 platelet43*platelet1
    red_blood_cells43*red_blood_cells1 white_blood_cells43*white_blood_cells1/reg;
run;
```

FIGURE 9



CONCLUSION

SAS/GRAPH SG procedures are exciting new functionalities provided by SAS 9.2. Simpler and minimal coding is required to generate high quality statistical graphs which are fully integrated with the Output Delivery System (ODS). By spending less time in coding to improve the visual appearance of graphs for laboratory data, we can focus more on the statistical analysis.

REFERENCES

SAS Institute, Inc. 2008. SAS/GRAPH® 9.2: *Statistical Graphics Procedures Guide*. Cary, NC: SAS Institute, Inc.

CONTACT INFORMATION

I welcome and appreciate your comments and questions. Contact the author at:

Wei Cheng, Isis Pharmaceuticals, Inc., 1896 Rutherford Rd., Carlsbad, CA 92008
(760) 603-3807 Email: wcheng@isisph.com