CREATE COMPREHENSIVE PATIENT PROFILES WITH SAS

Daniel O’Connor and Sanjay Matange, SAS® Institute Inc., Cary, NC

ABSTRACT

Patient profile reports visually display a patient’s data along a common time axis, providing medical reviewers with an at-a-glance view of a patient’s clinical assessments over time. SAS provides a rich toolset to efficiently create these important graphical reports. This approach can easily be used for multiple studies, and may be customized for a specific study’s needs. By efficiently producing these reports, mountains of clinical data can be transformed into meaningful information to assist researchers in fully understanding how patients respond in clinical trials.

INTRODUCTION

A comprehensive visualization of key patient data can aid researchers in efficiently discovering potential safety and efficacy trends. Entire industries have devoted years of research into improving the visualization of clinical data. With the introduction the new “ODS Graphics” framework in SAS 9.2, coupled with new reporting enhancements in BASE SAS, creating a comprehensive patient profile display for large or small studies is as simple as calling a macro.

PATIENT PROFILE 101

Comprehensive patient profile displays go far beyond previous attempts at creating disparate adverse events, and pharmaceutical dosing tabular displays. A comprehensive patient profile encompasses all facets of a patient’s medical information in a single display, and shows the relationship amongst all the various data points. It is not simply enough to create a single patient’s display, but more importantly to have the ability to quickly display all the medical information in a sub-set of patients in the study. For example, show me all the patients that exhibited moderate or severe adverse events. Furthermore, what was the relationship of the adverse event, and the medication dose prescribed, and the patient’s vital signs and lab work? A comprehensive patient profile is a holistic view of all relevant medical information. SAS 9.2 provides the basis for creating a macro to allow you to quickly create a holistic view of your patient information. The “patient_profile” macro typically creates a separate 8 page PDF file for each patient that meets the where clause criteria.

The following sample code is the macro call that will produce Figure 1, 3, 4, 5, 6 for each patient that meets the where criteria.

%patient_profile( libref=cdisc, where="usubjid='01-701-1097'");
The Demographic information at the top of the page is generated with the ODS Report Writing Interface which is designed exclusively for custom report writing purposes. The ODS Report Writing Interface is intended to fully embrace ODS features such as proportional fonts, traffic lighting, colors, images, Unicode characters, while at the same time providing pixel perfect placement capabilities. This interface is not only fully integrated with all the capabilities of the ODS System, but also takes advantage of the rich programming features that the DATA step offers such as conditional logic, formatting capabilities, by-group processing, arrays, and a wealth of other features. The ODS Report Writing Interface is an object-oriented language that provides you with flexibility and control so that even the most rigid reporting requirements can be met with ease.

The Adverse Event Timeline display is created using a new SAS/GRAPH procedure called SGPLOT which is part of the new ODS Statistical Graphics architecture. It is designed to create complex statistical graphics using advanced
design techniques. There are currently three other SG Procedures available in SAS/GRAPH in release 9.2: SGPANEL, which creates classification paneled graphs using one or more classification variables, SGSCATTER, which creates paneled graphs with multiple scatter plots in a variety of different displays, and SGRENDER, a utility procedure that produces graphs using the underlying Graph Template Language (GTL). The ODS Statistical Graphics architecture is also fully integrated with SAS/STAT, SAS/ETS, SAS/HPF, and BASE SAS. Most procedures take advantage of this enhancement to automatically produce graphical displays while performing the requested analysis. For example, PROC LIFETEST will automatically produce Survival Estimate Plots.

The following sample code will create Figure 2.

```sas
ods graphics on / imagename='Survival' width=8in height=6in;
ods select SurvivalPlot;
proc lifetest data=bygroup plots(only)=(survival(hw));
  time survival*d(0);
  Survival out=out1 confband=all;
  strata group;
  by bygroup;
run;
```

![Figure 2: PROC LIFETEST default output](image-url)
Creating a concomitant Medication display can easily be done by using the same PROC SGPLOT approach used in the previous Adverse Event display.

The following sample code will create Figure 3.

```sas
ods graphics / noborder width=7in height=3in imagename="Concomitant_Meds_&usubjid";
proc sgplot data=meds(where=(usubjid="&usubjid")) noautolegend nocycleattrs;
  reline 0 / axis=x lineattrs=(thickness=2);
  vector x=STARTDAY y=y / xorigin=ENDDAY yorigin=y group=cmtrt noarrowheads
    datalabel=cmtrt
    lineattrs=(thickness=5px pattern=solid color=blue);
  scatter x=STARTDAY y=y / group=cmtrt
    markerattrs=graphdatadefault(size=13px symbol=trianglefilled);
  scatter x=ENDDAY y=y / group=cmtrt
    markerattrs=graphdatadefault(size=13px symbol=trianglefilled);
  scatter x=STARTDATE y=y / markerattrs=(size=0) x2axis;
  yaxis display=(nolabel noticks novalues) min=0;
  xaxis grid label='Study Days' offsetmin=0.02 offsetmax=0.02
    values=(&minday to &maxday by 2);
  x2axis notimesplit display=(nolabel) tickvalueformat=date7. offsetmin=0.02
    offsetmax=0.02 values=(&mindate to &maxdate);
run;
```

Figure 3: Page 2 of 8 Concomitant Medications
One of the important items to consider when looking at the above graph is that it is generated using the SGPLOT procedure which means that this graph is dynamically generated based on the number of medications. No customization to the code is necessary when the number of medications vary from patient to patient, or study to study.

CLASSIFICATION GRAPHS (SGPANEL)

Classification or paneled graphs show the relationship inherit in the data, but also allows the ability to overlay related information. This provides us with the ability to show relationships between disparate data. A perfect example of this would be to show adverse event information color coded by severity overlaid with vital signs data.

Figure 4: Page 3 of 8 Show Color coded Adverse Event data overlaid with Vital Signs data
MULTI-PAGED CLASSIFICATION GRAPHS (SGPANEL)

A common problem when creating classification graphs is the sheer number of classifications to consider. When dealing with a physical piece of paper in the case of PDF output your real-estate is limited. One approach would be to shrink the visible dimension so that all classifications would fit on a page which might be sufficient depending on the number of classifications, but another alternative would be to create n pages of paneled output so that all data has sufficient space to accurately be displayed. A perfect example of this would be to show adverse event information color coded by severity overlaid with cardiology lab data. Notice that the reference range data is also displayed in the background for each classification so that out of tolerance values can easily be detected.

![Diagram showing CHF Cardiology Labs](image)

Figure 5: Page 5-7 of 8 Show Color coded Adverse Event data overlaid with Cardiology Lab data
MEDICAL EVENTS ( REPORT )

Creating a simple tabular display of medical events can easily be done by using PROC Report.

The following sample code will produce the Medical Events table on Page 8 of 8.

```sas
proc report data=ae(where=(usubjid="&usubjid");
    style (report)=[frame=void rules=none cellspacing=0
    width=100pct];
    columns AEDECOD AESTDT AESTDY AEENDY AESEV AEACN;
    define AESTDT / format=$10.;
run;
```

MEDICATIONS BY FREQUENCY ( SGRENDER )

The Graphical Template Language (GTL) is an extension to the Output Delivery System (ODS) that enables you to create sophisticated analytical graphics that are not available from traditional SAS/GRAF procedures statements. GTL is a comprehensive graphical language that allows you to create custom layouts and graphs that go well beyond the SGPLOT, SGSCATTER, and SGRENDER procedures. It is important to note that GTL is a reusable template that only describes the layout of the graph and its attributes, but actually does not create any output.

GTL has a rich set of features to provide for dynamic variable resolution so that the templates can be adapted to different data set variables without having to redefine the graphical display. In this particular example I used the direct variable reference. SGRENDER is the procedure that produces graphical output using the GTL defined template. It associates the data to the template and any potential dynamic variable substitutions.

The following sample code sample creates the Medication by frequency Graph on Page 8 or 8.

```sas
proc template;
    define statgraph meds;
        begingraph;
            layout lattice / columns=3
                columnweights= (.2 .2 .6) rowdatarange=union;
            rowaxes;
                rowaxis / display=(line ticks tickvalues);
            endrowaxes;
            layout overlay / xaxisopts=(label='Start Date' display=(line label));
                scatterplot x=x y=cmtrt / markercharacter=startdate;
            endlayout;
            layout overlay / xaxisopts=(label='End Date' display=(line label));
                scatterplot x=x y=cmtrt / markercharacter=enddate;
            endlayout;
            layout overlay;
                barchart x=cmtrt y=count / orient=horizontal;
            endlayout;
        endgraph;
    end;
    run;
proc sgrender data=cm3(where=(usubjid="&usubjid") template=meds;
run;
```
Figure 6: Page 8 of 8 Medical events and Medications by frequency
PATIENT PROFILE MACRO PROCESS

New File

Got AE?
- Create AE Graph for Patient
- Got CM?
- Create CM Graph for Patient
- New Page
- Got VS?
- Create VS Graph for Patient
- Got LB?
- Create LB Graph for Patient
- Got AE?
- Create AE Table for Patient
- Got CM?
- Create CM FREQ Graph for Patient
- Create MSG to indicate no AE Data
- Create MSG to indicate no CM Data
- Create MSG to indicate no VS Data
- Create MSG to indicate no LB Data

New File?

New Page

Figure 7: Patient Profile macro flow chart
TOP ADVERSE EVENTS

Patient Profile displays on just one very focused aspect of visualizing clinical data. The ODS Statistical Graphics architecture provides a rich set of capabilities to create elaborate graphical displays. These features can be applied to creating new visual displays that address the study as a whole. For example, visualizing the top twenty adverse events detected in relation to the medication prescribed.

This code sample creates the Percent of Top Twenty Adverse Events.

```sas
proc template;
   define statgraph aedm;
   begingraph / designwidth=6in designheight=4in;
      entrytitle "Percent of Top Twenty Adverse Events";
      layout lattice / rowweights=(0.9 0.1);
      layout overlay / yaxisopts=(reverse=true display=(ticks tickvalues line))
                     xaxisopts=(display=(ticks tickvalues line));
      barchartparm x= aedecod y=barht / orient=horizontal
data=(fill) datatransparency=0.7;
      scatterplot y=aedecod x=percent / group=arm name='scatter'
                   markerattrs=(size=9 weight=bold);
      discretelegend 'scatter' / down=1;
   endlayout;
   endlayout;
   endgraph;
end;
runc;
```

Figure 8: Top twenty adverse events by treatment group for all the subjects in the study
CONCLUSION

Creating comprehensive patient profiles with SAS 9.2 is easy. The best part is that the patient_profile macro is just SAS code so it can be extended and enhanced to meet your requirements. The SAS/GRAPH ODS Graphics architecture in conjunction with BASE SAS reporting enhancements can fulfill even the most challenging clinical data visual need.

REFERENCES


ACKNOWLEDGMENTS

The author would like to thank Melisa Turner, Dan Heath, David Handelsman, Geoffrey Mann, Chevell Parker, Allison Crutchfield, Randy Poindexter, and Matt Gross for their contributions to this paper.

CONTACT INFORMATION

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

    Daniel O'Connor
    Building R, SAS Campus Drive
    SAS Institute Inc.
    Cary, NC 27513
    E-mail: Dan.OConnor@sas.com

    Sanjay Matange
    Building R, SAS Campus Drive
    SAS Institute Inc.
    Cary, NC 27513
    E-mail: Sanjay.Matange@sas.com

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.
## APPENDIX

### PATIENT PROFILE MACRO

<table>
<thead>
<tr>
<th><strong>Patient_profile</strong></th>
<th><strong>Syntax</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Patient_profile ( &lt; optional argument &gt;, ..., &lt; optional argument &gt; );</td>
</tr>
</tbody>
</table>

**Description**

Produces a multi-page comprehensive visualization of key patient data, and shows the relationship amongst all the various data points.

**Optional arguments**

<table>
<thead>
<tr>
<th><strong>libref</strong></th>
<th>&lt; libref &gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A valid libref that points to a directory of CDISC SAS data sets. If one is not assigned SAS will look in the current working directory for CDISC SAS data sets.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>where</strong></th>
<th>&lt; string &gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The where clause to be applied to the demographics data set.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>destination</strong></th>
<th>&lt; string &gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The ODS Destination to be used to publish the results. Currently only supports PDF, and HTML. Default: PDF</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>style</strong></th>
<th>&lt; string &gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The style to be used for the selected destination.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>path</strong></th>
<th>&lt; string &gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The physical path for the output files generated by the selected destination.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>filename</strong></th>
<th>&lt; string &gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The base filename to be used with the selected destination.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>extension</strong></th>
<th>&lt; string &gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The filename extension to be used for the filename.</td>
</tr>
</tbody>
</table>

| **newfile** | < PATIENT | NONE > |
|-------------|------------|
|             | Controls the how the output will be organized. Default: PATIENT |

- **PATIENT** Create a new PDF file for each Patient
- **NONE** Create the entire Patient Profiles in a single file.