PharmaSUG 2019 - Paper HT-063

Developing Custom SAS® Studio Tasks for Clinical Trial Graphs

Olivia Wright, SAS Institute Inc., and Sanjay Matange

ABSTRACT

SAS Studio provides point-and-click tasks for basic statistics, biostatistics models, and statistical graphs. SAS Studio users can create their own Custom Tasks by modifying existing tasks or writing new ones using simple text commands from Apache Velocity Template Language. Custom Tasks can simplify sharing of code by turning complex SAS graphics code into point-and-click tasks that can be shared. The same functionality can be integrated into a more complete analytic modeling workflow by adding additional graphing options to an existing built-in task. In this workshop, we will work hands-on to create a graphical user interface for code-heavy SGPLOT clinical trial graphs.

INTRODUCTION

Often, programmers have code they want to run repeatedly with different parameters. This happens especially when generating graphs. They want to recreate the same graph in a new style, or with a different data set, or different variables. SAS Studio Custom Tasks allow programmers to create point-and-click interfaces for running SAS code, making altering the options and re-running to create a new graph as easy as a click of a button.

This paper will introduce several SAS Studio Custom Tasks for creating customized versions of popular clinical trials graphs created using the SGPLOT procedure. These tasks allow users to re-create these clinical trials graphs using their own data and style choices without ever having to manually edit any SGPLOT code.

There will be a brief introduction of the Custom Task code sections, followed by a listing of common graph parameters and their corresponding task controls. Finally, the paper will end with four examples of tasks for creating clinical graphs, the first of which will include an in-depth description of each of its code sections. All of the original plots and SGPLOT code are from past Graphically Speaking blog posts by Sanjay Matange. All of the task examples in this paper are available for download on the Custom Task Tuesday GitHub.

CUSTOM TASK BASICS

SAS Studio tasks are written in Apache Velocity Template Language. Six different sections make up a task, as shown in Table 1.

Task Section	Description
Registration	Includes elements of the task, such as Name, Description, and Procedures used, as well as links to helpful information.
Metadata	Defines the data source, role objects, and controls wanted in the task.
UI	Defines the layout of the user interface and order of the objects listed in the Metadata section.
Dependencies (not required)	Specifies how certain objects (or controls) rely on one another in order for the task to work properly.
Requirements (not required)	Specifies conditions for the task to run. If the condition is true, the SAS code can be generated.
Code Template	Written in Apache Velocity Template Language. The task triggers the SAS code, filled in with Velocity Macro Variables from the corresponding controls.

Table 1. Task Sections

In SAS Studio, under Tasks and Utilities, you can click the icon to create a “New Task” or view a “Sample Task” or “Advanced Task.”

COMMON GRAPH PARAMETERS

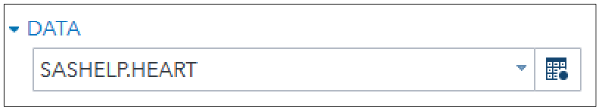
There are many common changes users want to make to graphs, all of which we can build in as parameters for Custom Tasks. Examples of these include:

- Choosing different data
- Changing the color scheme
- Creating a new title
- Changing legend attributes

Some of these changes will apply to each of the example plots for this paper. The following sections will describe each Custom Task control that will be used.

CHOOSING DIFFERENT DATA

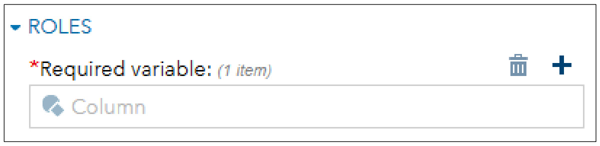
The most common need is to recreate the same graph exactly using a different data set or different variables. To allow users to supply their data set and variables in a SAS Studio task, a data set selector is necessary, as well as role (variable) selectors.



SASHELP.HEART

Figure 1. Custom Task Data Source Selector

With the data source selector, the task author has options to specify a library engine of that data set.



*Required variable: *(1 item)*

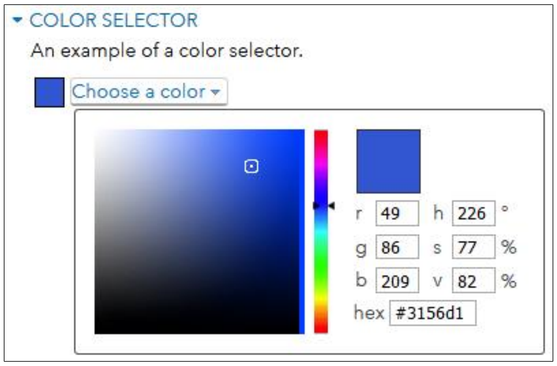
Column

Figure 2. Custom Task Role (Variable) Selector

With the role selector, the task author can restrict what type of variable can be selected (numeric, character, or either), as well as set the minimum and maximum number of variables that can be selected.

CHANGING THE COLOR SCHEME

Another common alteration is to recreate the same graph using a different color scheme. The necessary Custom Task controls for doing this are the color selector or the combo box (drop-down list) with color scheme options.

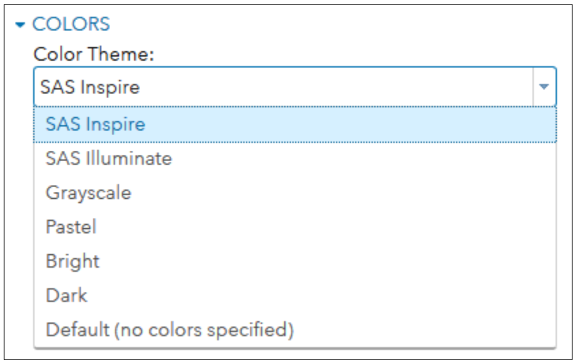


An example of a color selector.

r 49 h 226 °
g 86 s 77 %
b 209 v 82 %
hex #3156d1

Figure 3. Custom Task Role Color Selector

The color selector allows the task user to select one custom color by clicking a color or providing an RGB, HSV, or hex value.



Color Theme: SAS Inspire

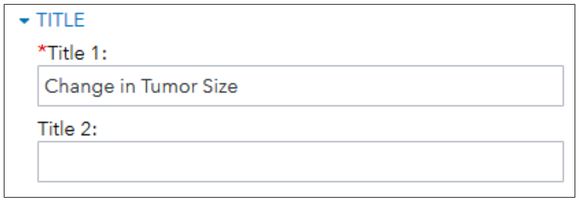
Figure 4. Custom Task Color Theme Drop-Down

The Custom Task Color Theme selector is a combo box or drop-down field that provides different color theme options. The themes in this list were individually selected. A task author can choose any list of themes they are interested in to populate the drop-down list. Each theme has a corresponding list of hex codes that are used in the SAS code when that particular theme is chosen. Note that this theme has nothing to do with ODS style. ODS style (HTML, Listing, etc.) can be changed in SAS Studio preferences.

Themes are beneficial because often the color that needs to be selected is for a group variable in the graph where different values of the variable are assigned different colors. In this situation, the task author would not know how many individual colors are needed and therefore would not know how many color selectors to provide. Additionally, providing a curated list of cohesive color schemes makes things much easier for task users who may not want to spend time selecting their own individual colors.

CHANGING THE TITLE

When users change any aspect of a graph, they will likely want to adjust the title of the graph as well. The control that is needed to change the title is the input text control.



*Title 1: Change in Tumor Size

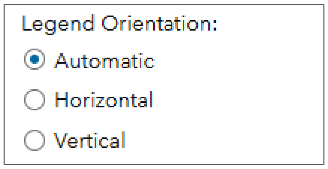
Title 2:

Figure 5. Custom Task Input Text Box for Title

The example shown in Figure 5 provides two text boxes for first and second graph titles. The box for title 1 is required, while the box for title 2 is not. This allows the user to title their graph as they see fit.

CHANGING LEGEND ATTRIBUTES

A final aspect of the graph that users will want to be able to change is the legend.



Legend Orientation:

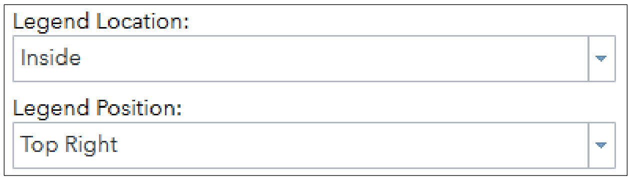
● Automatic

○ Horizontal

○ Vertical

Figure 6. Custom Task Radio Button for Legend Orientation

Desired legend orientations will change depending on how many items are in the legend and the length of each string. This radio button gives users the opportunity to let PROC SGPLOT decide the best orientation, or to force the orientation to be horizontal or vertical.



Legend Location:
Inside

Legend Position:
Top Right

Figure 7. Custom Task Drop Downs for Legend Location and Position

Desired legend locations and positions often depend on what the data look like on the graph. For example, if a histogram begins with high values and steadily descends, there is probably plenty of room for the legend to be placed in the top-right corner inside the graph. It is helpful to have drop-down lists that allow users to run the task and see how the data looks on the graph, then change the legend location and re-run the task.

EXAMPLE TASKS FOR CLINICAL GRAPHS

The following example plots and corresponding SGPLOT code are from past Graphically Speaking blog posts by Sanjay Matange. Each plot was converted into a task to allow for easy graph customization using a point-and-click interface. The examples are listed in order of task complexity starting with the simplest. Provided with the first example is an in-depth explanation of how the code was converted into a task. The subsequent examples show additional modifications that can be included to further improve the task.

INJECTION SITE REACTION GRAPH

The first example task creates an injection site reaction graph. This graph is from the 2014 blog titled "Clinical Graphs" on Graphically Speaking. Figure 8 displays the cluster-grouped bar chart that shows incidence of reaction by time and cohort.

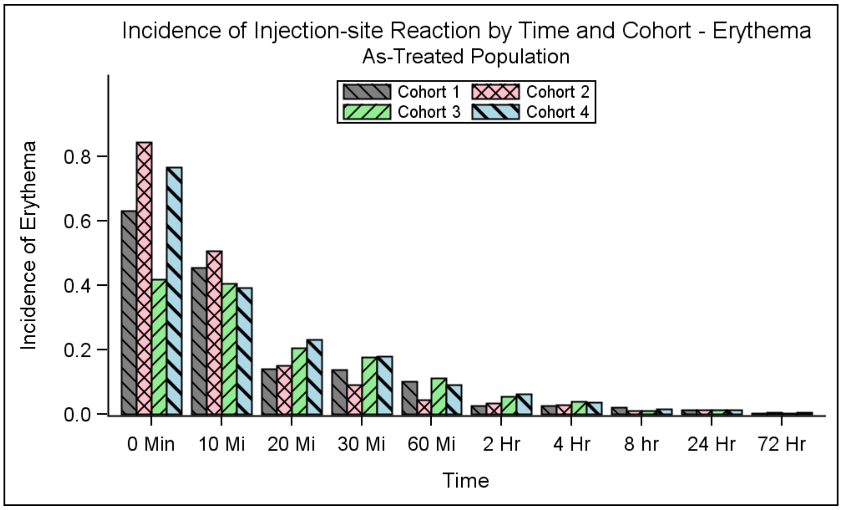
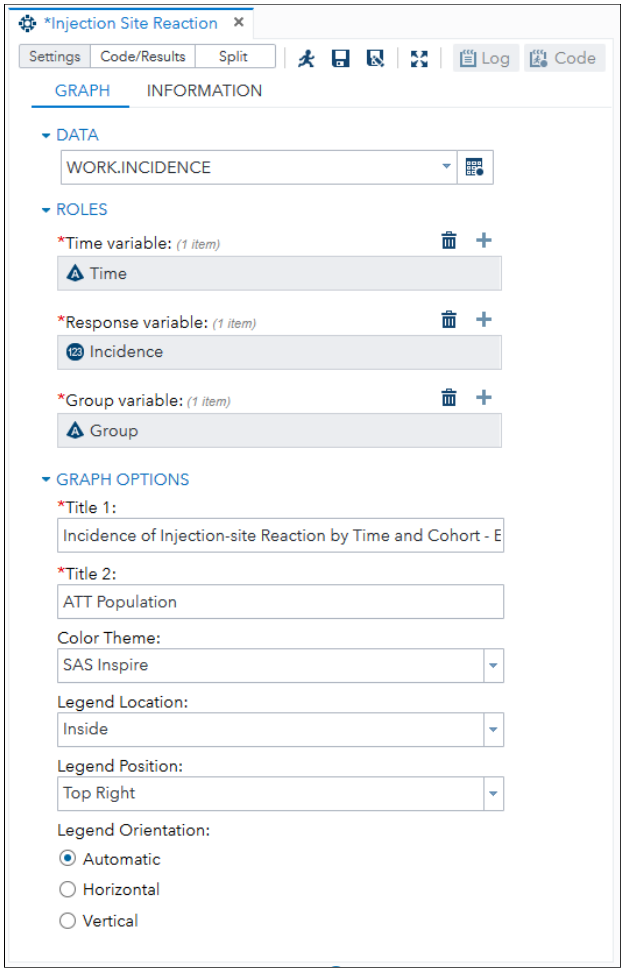


Figure 8. Original Injection Site Reaction Graph

The original SGPLOT code to create this chart is shown below. Boxes appear around the pieces of code that will be filled in by task controls:

```
title 'Incidence of Injection-site Reaction by Time and Cohort - Erythema';
title2 'ATT Population';
proc sgplot data=Incidence nowall noborder;
  styleattrs datacolors=(gray pink lightgreen lightblue)
  datacontrastcolors=(black);
  vbar time / response=incidence group=group groupdisplay=cluster;
  xaxis discreteorder=data;
  yaxis offsetmax=0.2;
  keylegend / title='' location=inside position=top across=2 border
  autoitemsize valueattrs=(size=8);
run;
```

The final task that will replace this SGPLOT code is shown in Figure 9.



DATA

WORK.INCIDENCE

ROLES

*Time variable: (1 item)

Time

*Response variable: (1 item)

Incidence

*Group variable: (1 item)

Group

GRAPH OPTIONS

*Title 1:

Incidence of Injection-site Reaction by Time and Cohort - E

*Title 2:

ATT Population

Color Theme:

SAS Inspire

Legend Location:

Inside

Legend Position:

Top Right

Legend Orientation:

● Automatic

○ Horizontal

○ Vertical

Figure 9. Custom Task for creating the Injection Site Reaction Graph

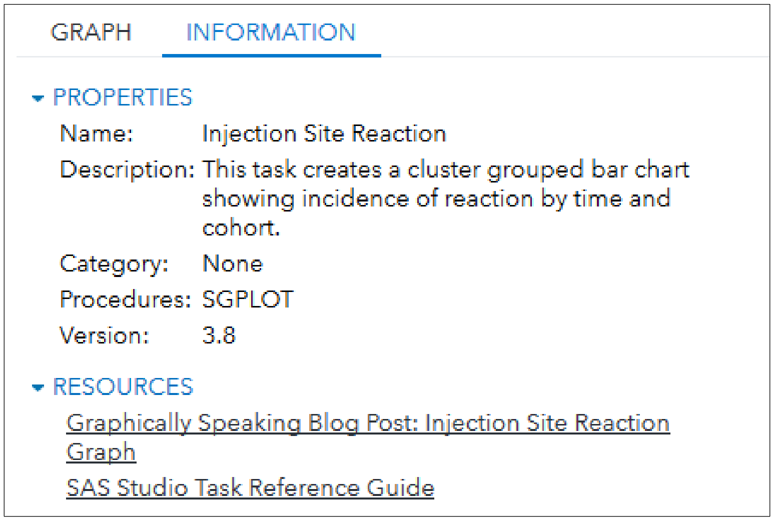
The following sections will describe the creation of the task for this example. All code for each of the four necessary sections (Registration, Metadata, UI, and CodeTemplate) is shown.

Registration

The Registration section defines a Name, Description, and the Procedures that were used. The links included here are to the corresponding Graphically Speaking blog post, and general help with tasks is from the SAS Studio Task Reference guide. The registration section for this task is shown below:

```
<Registration>
<Name>Injection Site Reaction</Name>
<Description>This task creates a cluster grouped bar chart showing incidence of
reaction by time and cohort. </Description>
<GUID>726B2B12-4542-4200-96F4-960C645D767F</GUID>
<Procedures>SGPLOT</Procedures>
<Version>3.8</Version>
<Links>
  <Link
href="https://blogs.sas.com/content/graphicallyspeaking/2014/11/15/clinical-
graphs/">Graphically Speaking Blog Post: Injection Site Reaction
Graph</Link>
  <Link
href="http://documentation.sas.com/?softwareId=STUDIOMID&amp;softwareVersion=3.71&amp;
softwareContextId=tasks&amp;requestor=inapp">SAS Studio Task Reference Guide</Link>
</Links>
</Registration>
```

The information from the filled-in Registration section will appear in the automatically created INFORMATION tab for the task. The information tab for this task is shown below:



PROPERTIES

Name: Injection Site Reaction

Description: This task creates a cluster grouped bar chart showing incidence of reaction by time and cohort.

Category: None

Procedures: SGPLOT

Version: 3.8

RESOURCES

Graphically Speaking Blog Post: Injection Site Reaction Graph

SAS Studio Task Reference Guide

Figure 10. Information Tab of Injection Site Reaction Task

Metadata

The Metadata portion defines all objects and controls to appear in the task. This task includes options for changing the data source, variables, color scheme, legend position/location, and titles. The full metadata section is shown below:

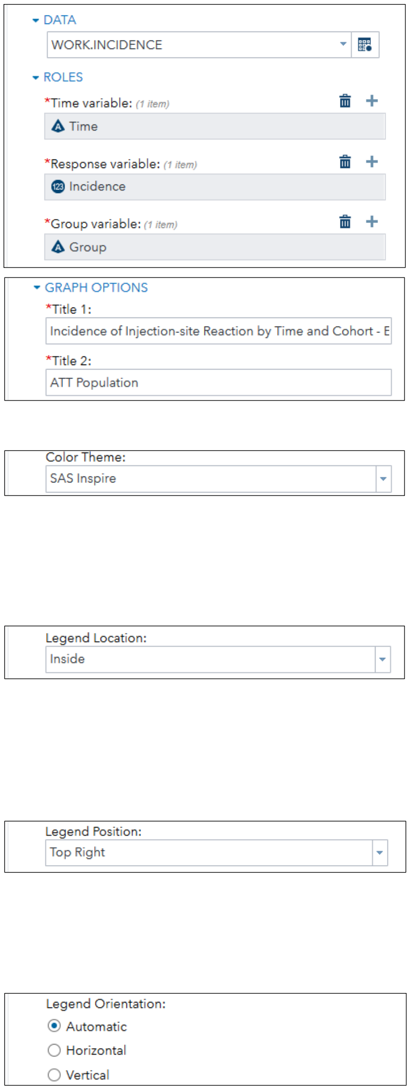
```
<Metadata>
  <DataSources>
    <DataSource name="DATASOURCE">
      <Roles>
        <Role type="C" maxVars="1" order="true" minVars="1" name="TIMEVAR">Time
variable:</Role>
        <Role type="N" maxVars="1" order="true" minVars="1" name="RESPVAR">Response
```

```
variable:</Role>
        <Role type="C" maxVars="1" order="true" minVars="1" name="GROUPVAR"
exclude="TIMEVAR">Group variable:</Role>
      </Roles>
    </DataSource>
  </DataSources>
  <Options>
    <Option name="GRAPHTAB" inputType="string">GRAPH</Option>
    <Option name="DATAGROUP" inputType="string">DATA</Option>
    <Option name="ROLESGROUP" inputType="string">ROLES</Option>
    <Option name="GROUPGRAPH" inputType="string">GRAPH OPTIONS</Option>
    <Option name="textTITLE1" defaultValue="Incidence of Injection-site Reaction by
Time and Cohort - Erythema" inputType="inputtext" required="true" promptMessage="Enter
a title" missingMessage="Missing text.">Title 1:</Option>
    <Option name="textTITLE2" defaultValue="ATT Population"
inputType="inputtext" required="true" promptMessage="Enter a second title"
missingMessage="Missing text.">Title 2:</Option>
    <Option name="comboTHEME" defaultValue="inspire" inputType="combobox">Color
Theme:</Option>
    <Option name="inspire" inputType="string">SAS Inspire</Option>
    <Option name="illuminate" inputType="string">SAS Illuminate</Option>
    <Option name="grayscale" inputType="string">Grayscale</Option>
    <Option name="pastel" inputType="string">Pastel</Option>
    <Option name="bright" inputType="string">Bright</Option>
    <Option name="dark" inputType="string">Dark</Option>
    <Option name="default" inputType="string">Default (no colors specified)</Option>
    <Option name="comboLEGENDPOS" defaultValue="topright" inputType="combobox">Legend
Position:</Option>
    <Option name="bottom" inputType="string">Bottom</Option>
    <Option name="bottomleft" inputType="string">Bottom Left</Option>
    <Option name="bottomright" inputType="string">Bottom Right</Option>
    <Option name="left" inputType="string">Left</Option>
    <Option name="right" inputType="string">Right</Option>
    <Option name="top" inputType="string">Top</Option>
    <Option name="topleft" inputType="string">Top Left</Option>
    <Option name="topright" inputType="string">Top Right</Option>
    <Option name="comboLEGENDLOC" defaultValue="inside" inputType="combobox">Legend
Location:</Option>
    <Option name="inside" inputType="string">Inside</Option>
    <Option name="outside" inputType="string">Outside</Option>
    <Option name="labelORIENTATION" inputType="string">Legend Orientation:</Option>
    <Option name="radioAuto" variable="radioORIENTATION" defaultValue="1"
inputType="radio">Automatic</Option>
    <Option name="radioHorizontal" variable="radioORIENTATION"
inputType="radio">Horizontal</Option>
    <Option name="radioVertical" variable="radioORIENTATION"
inputType="radio">Vertical</Option>
  </Options>
</Metadata>
```

UI

The UI portion of the task is what defines the layout of all the options/controls that were created in the Metadata section. The layout is top-down, so options are listed in the order they are to appear in the task from top to bottom. The full UI section is shown below:

```
<UI>
  <Container option="GRAPHTAB">
    <Group option="DATAGROUP" open="true">
      <DataItem data="DATASOURCE"/>
    </Group>
    <Group option="ROLESGROUP" open="true">
      <RoleItem role="TIMEVAR"/>
      <RoleItem role="RESPVAR"/>
      <RoleItem role="GROUPVAR"/>
    </Group>
    <Group option="GROUPGRAPH" open="true">
      <OptionItem option="textTITLE1"/>
      <OptionItem option="textTITLE2"/>
      <OptionChoice option="comboTHEME">
        <OptionItem option="inspire"/>
        <OptionItem option="illuminate"/>
        <OptionItem option="grayscale"/>
        <OptionItem option="pastel"/>
        <OptionItem option="bright"/>
        <OptionItem option="dark"/>
        <OptionItem option="default"/>
      </OptionChoice>
      <OptionChoice option="comboLEGENDLOC">
        <OptionItem option="inside"/>
        <OptionItem option="outside"/>
      </OptionChoice>
      <OptionChoice option="comboLEGENDPOS">
        <OptionItem option="bottom"/>
        <OptionItem option="bottomleft"/>
        <OptionItem option="bottomright"/>
        <OptionItem option="left"/>
        <OptionItem option="right"/>
        <OptionItem option="top"/>
        <OptionItem option="topleft"/>
        <OptionItem option="topright"/>
      </OptionChoice>
      <OptionItem option="labelORIENTATION"/>
      <OptionItem option="radioAuto"/>
      <OptionItem option="radioHorizontal"/>
      <OptionItem option="radioVertical"/>
    </Group>
  </Container>
</UI>
```



WORK.INCIDENCE

*Time variable: (1 item)
Time

*Response variable: (1 item)
Incidence

*Group variable: (1 item)
Group

*Title 1:
Incidence of Injection-site Reaction by Time and Cohort - E

*Title 2:
ATT Population

Color Theme:
SAS Inspire

Legend Location:
Inside

Legend Position:
Top Right

Legend Orientation:
● Automatic
○ Horizontal
○ Vertical

CodeTemplate

The CodeTemplate portion of the task is what makes everything work. It contains the SAS code necessary for creating the graph, filled in with velocity macro variables (accessed using the $ symbol) that correspond with the UI controls. There are let statements with hard-coded hex values for color schemes, followed by the PROC SGPLOT for a grouped bar chart. The full CodeTemplate section is below:

```
#if ($comboTHEME == 'inspire')
%let fillcolors= cx21b9b7 cx4141e0 cx7db71a cx8e2f8a cxd38506 cx0abf85 cx2f90ec;
#end
#if ($comboTHEME == 'illuminate')
%let fillcolors= cx00929f cxf08000 cx90b328 cx3d5aae cxffca39 cxa6427c cx9c2910
cx736519;
#end
#if ($comboTHEME == 'grayscale')
%let fillcolors= cx585858 cxa2a2a2 cx1e1e1e cx707070 cxbbbbbb cx3b3b3b cxd0d0d0;
#end
#if ($comboTHEME == 'pastel')
```

```
%let fillcolors= cxb3e2cd cxfdcdac cxcbd5e8 cxf4cae4 cxe6f5c9 cxfff2ae cxf1e2cc
cxcccccc;
#end
#if ($comboTHEME == 'bright')
%let fillcolors= cx3190D0 cxFF991A cx3EBB3B cxF23E39 cxAF7ED9 cxA77063 cxFF8EDD
cxD6D91A;
#end
#if ($comboTHEME == 'dark')
%let fillcolors= cx1b9e77 cxd95f02 cx7570b3 cxe7298a cx66a61e cxe6ab02 cxa6761d
cx666666;
#end

title '$textTITLE1';
title2 '$textTITLE2';
proc sgplot data=$DATASOURCE nowall noborder;
  #if ($comboTHEME != 'default')
  styleattrs datacolors=(&fillcolors) datacontrastcolors=(black);
  #end
  vbar #foreach ($item in $TIMEVAR) $item #end / #foreach ($item in $RESPVAR)
response=$item #end #foreach ($item in $GROUPVAR) group=$item #end
groupdisplay=cluster baselineattrs=(thickness=0);
  xaxis discreteorder=data;
  yaxis offsetmax=0.2;
  keylegend / title='' location=$comboLEGENDLOC position=$comboLEGENDPOS border
autoitemsize valueattrs=(size=8) #if($radioORIENTATION == 'radioHorizontal') down=1
#end #if($radioORIENTATION == 'radioVertical') across=1 #end;
run;
title;
```

LIPID PROFILE GRAPH

The second example task creates a lipid profile graph. This graph is from the 2011 blog titled "It Pays to Be Discrete" on Graphically Speaking. Figure 11 displays the chart that shows medians and 95% confidence limits of lipid values by visit and treatment.

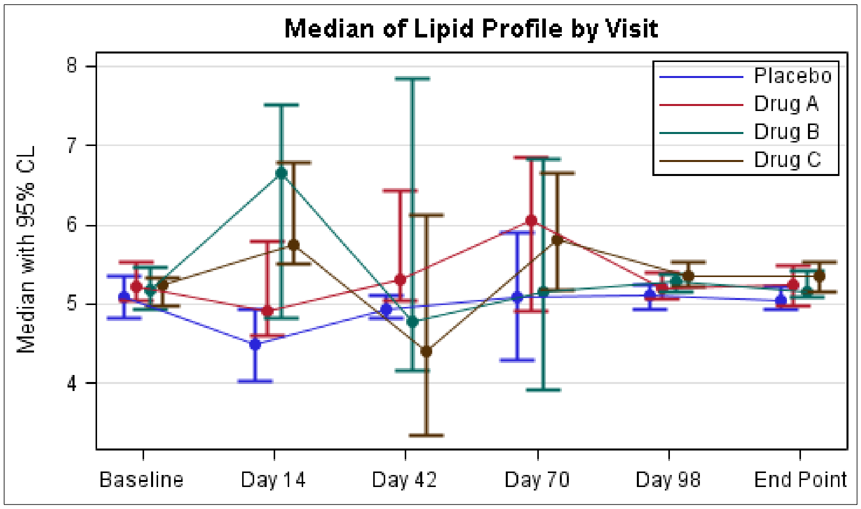
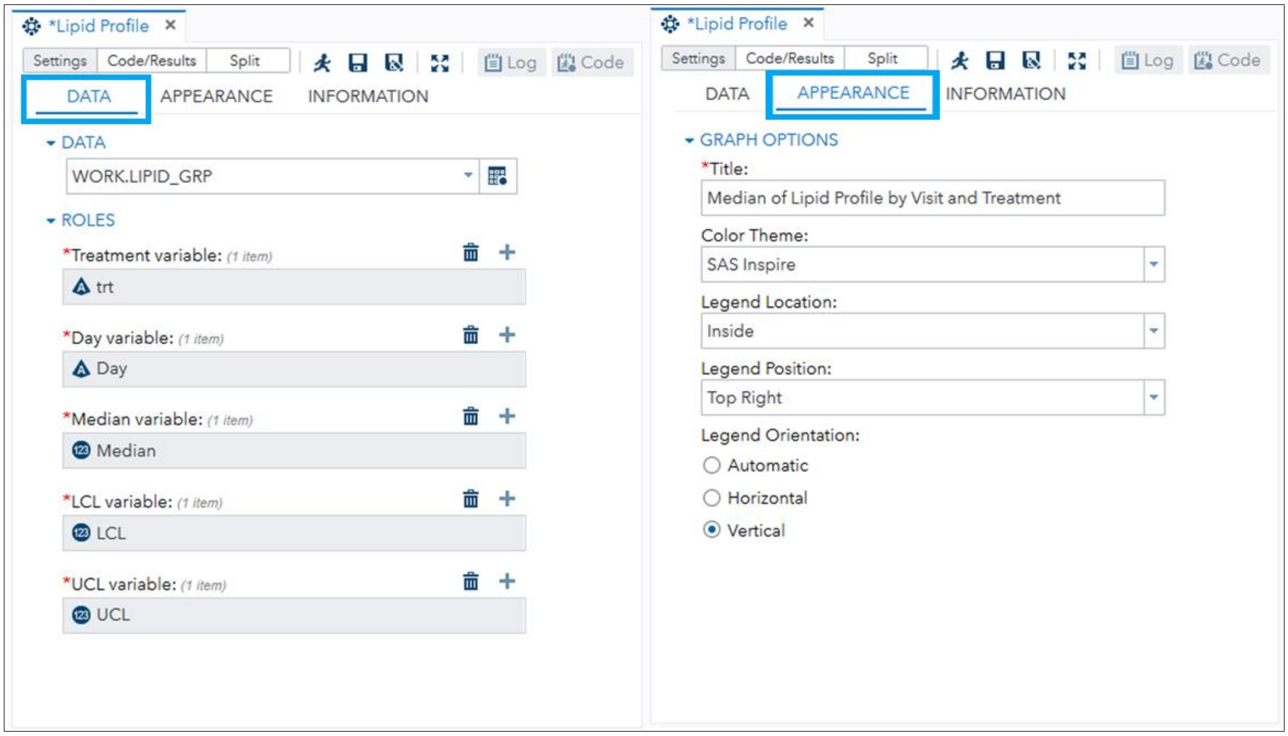


Figure 11. Original Lipid Profile Graph

The task includes options for changing the data source, variables, color scheme, legend position/location, and title. Unlike the first task, the controls here are separated into two tabs: DATA and APPEARANCE (Figure 12).



DATA

WORK.LIPID_GRP

ROLES

*Treatment variable: *(1 item)* trt

*Day variable: *(1 item)* Day

*Median variable: *(1 item)* Median

*LCL variable: *(1 item)* LCL

*UCL variable: *(1 item)* UCL

GRAPH OPTIONS

*Title: Median of Lipid Profile by Visit and Treatment

Color Theme: SAS Inspire

Legend Location: Inside

Legend Position: Top Right

Legend Orientation:

○ Automatic
○ Horizontal
● Vertical

Figure 12. Custom Task for Creating the Lipid Profile Graph

CHANGE IN TUMOR SIZE GRAPH

The third example task creates a change in tumor size graph. This graph is from the 2014 blog titled "Clinical Graphs" on Graphically Speaking. The data show range of change in tumor size using a waterfall graph grouped by treatment.

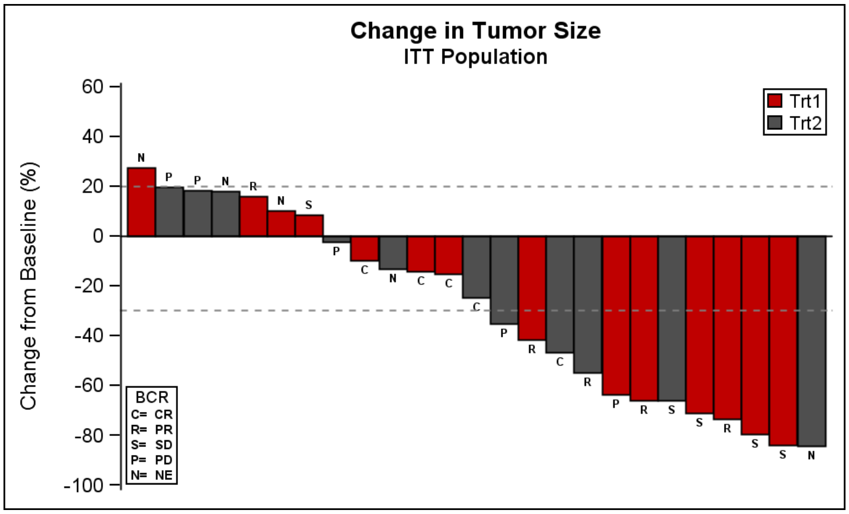
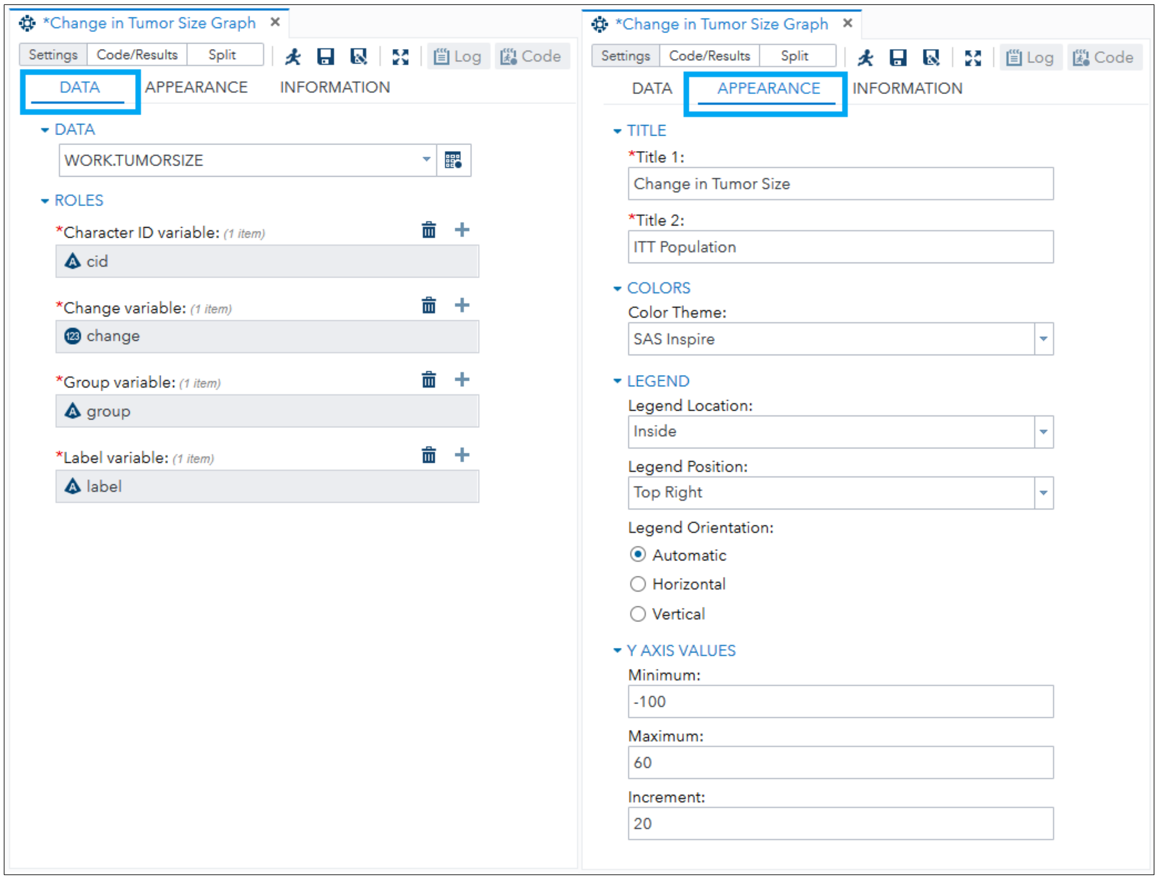


Figure 13. Original Change in Tumor Size Graph

Like the previous examples, this task includes options for changing the data source, variables, color scheme, legend position/location, and title. In addition, this task also includes an option for a second title, as well as y-axis minimum, maximum, and increment.



DATA

WORK.TUMORSIZE

ROLES

*Character ID variable: (1 item)
cid

*Change variable: (1 item)
change

*Group variable: (1 item)
group

*Label variable: (1 item)
label

TITLE

*Title 1:
Change in Tumor Size

*Title 2:
ITT Population

COLORS

Color Theme:
SAS Inspire

LEGEND

Legend Location:
Inside

Legend Position:
Top Right

Legend Orientation:
● Automatic
○ Horizontal
○ Vertical

Y AXIS VALUES

Minimum:
-100

Maximum:
60

Increment:
20

Figure 14. Custom Task for Creating the Change in Tumor Size Graph

SURVIVAL ANALYSIS GRAPH

The fourth and final example task creates a survival analysis plot. This graph is from the 2014 blog titled "Survival Plot" on Graphically Speaking. This is the same plot that is created as output of the LIFETEST procedure.

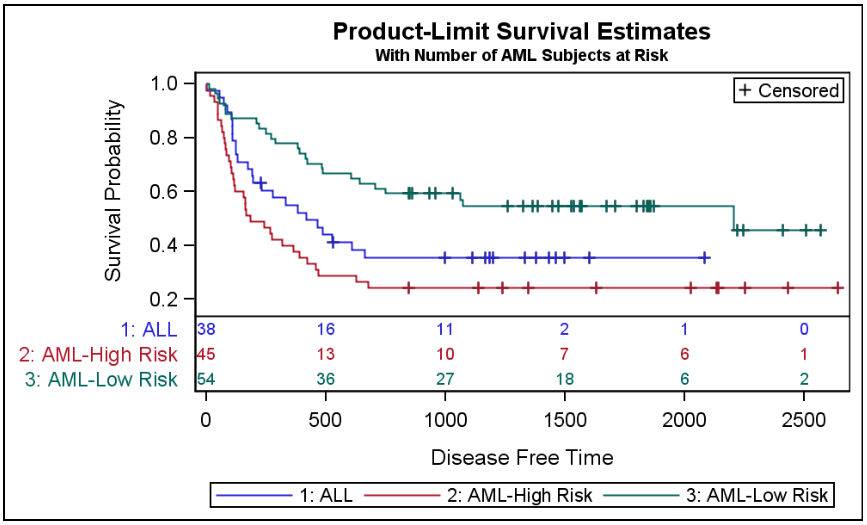
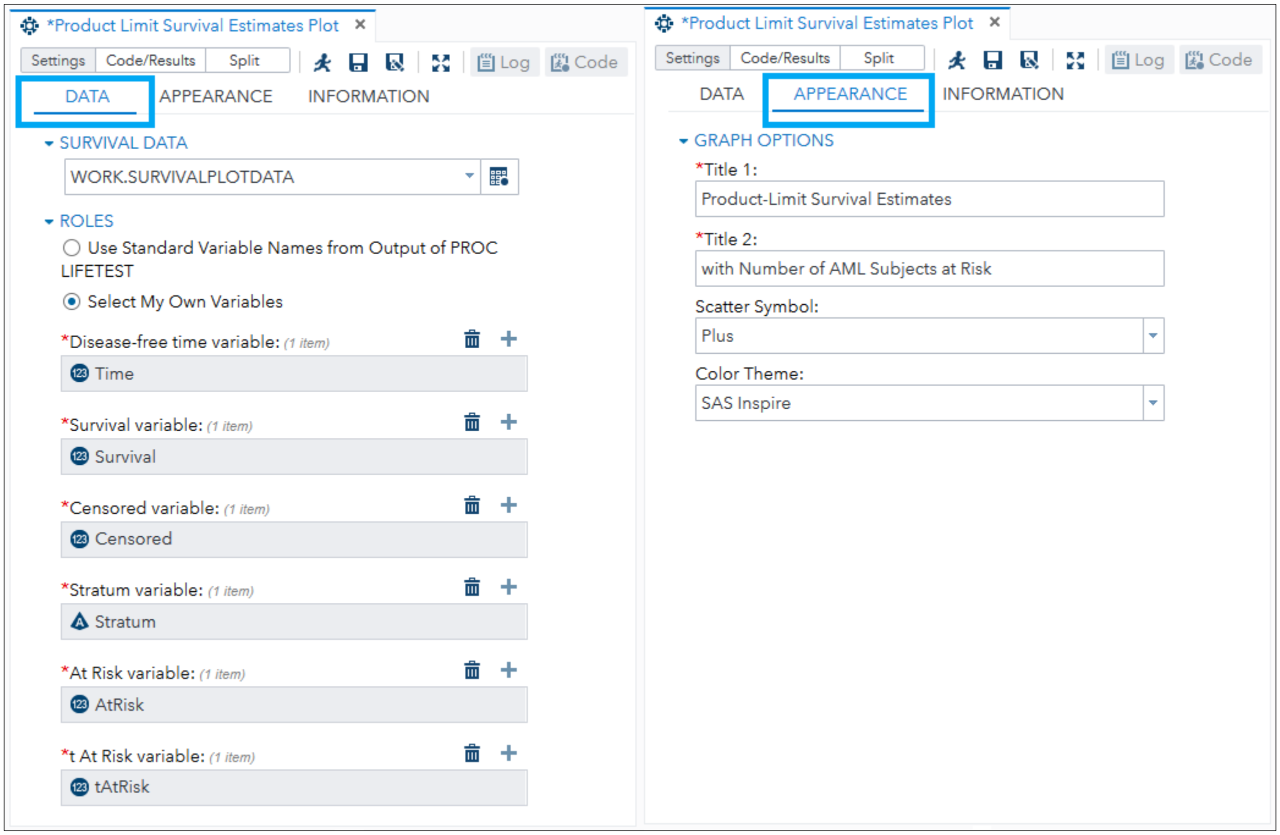


Figure 15. Original Survival Analysis Graph

This task includes options for changing the data source, variables, scatter symbol, color scheme, and title. Because this graph is made using a data set output from PROC LIFETEST, an option exists to use the standard SAS output variable names that does not require the user to select each of the variables individually.



SURVIVAL DATA

WORK.SURVIVALPLOTDATA

ROLES

○ Use Standard Variable Names from Output of PROC LIFETEST

● Select My Own Variables

*Disease-free time variable: *(1 item)* Time

*Survival variable: *(1 item)* Survival

*Censored variable: *(1 item)* Censored

*Stratum variable: *(1 item)* Stratum

*At Risk variable: *(1 item)* AtRisk

*t At Risk variable: *(1 item)* tAtRisk

GRAPH OPTIONS

*Title 1: Product-Limit Survival Estimates

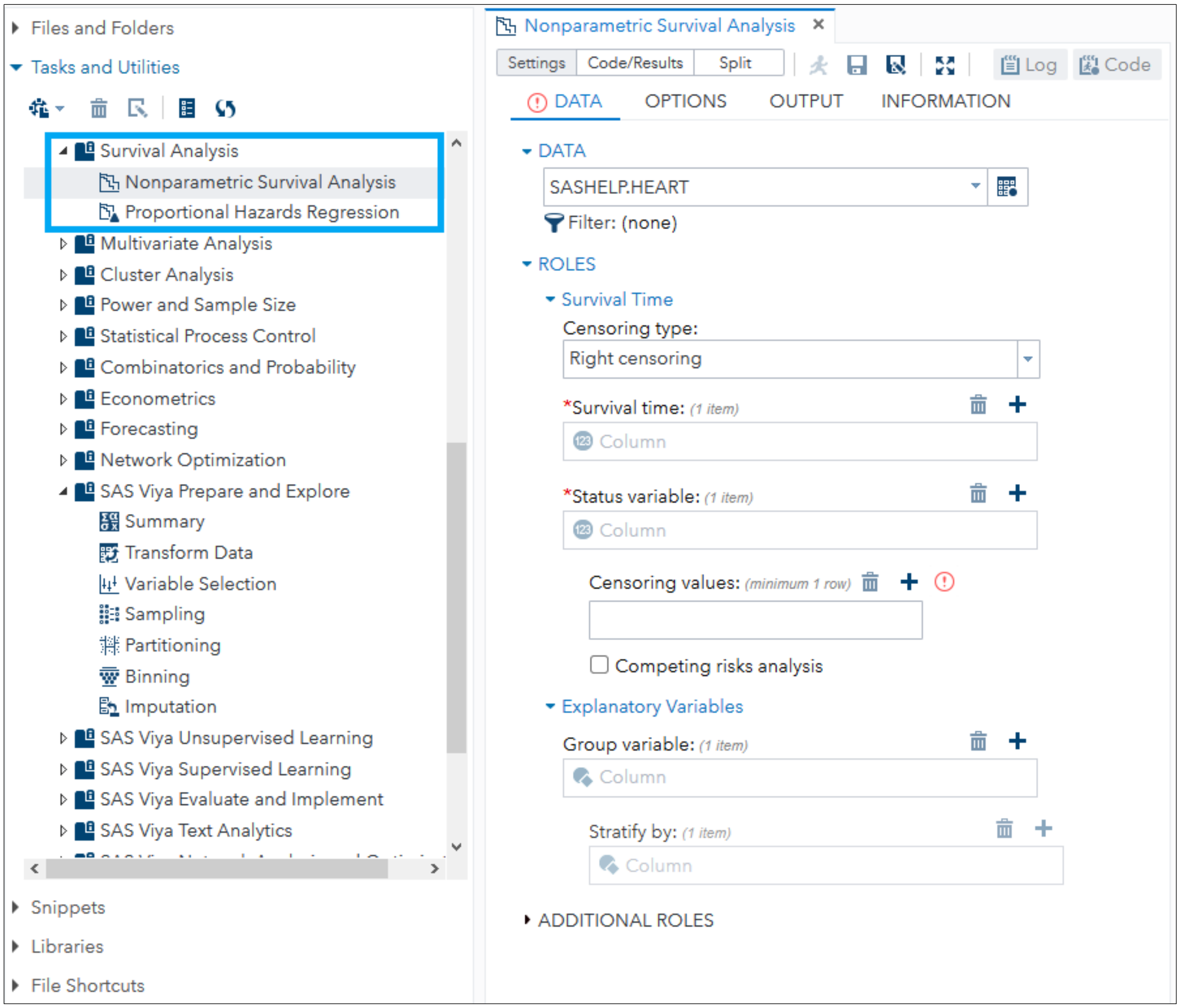
*Title 2: with Number of AML Subjects at Risk

Scatter Symbol: Plus

Color Theme: SAS Inspire

Figure 16. Custom Task for creating the Survival Analysis Graph

SAS Studio 3.8 offers new built-in Survival Analysis tasks. The built-in Nonparametric Survival Analysis task (Figure 17) can be edited to include the additional graphing customization options shown in this paper.



- Files and Folders
- Tasks and Utilities
  - Survival Analysis
    - Nonparametric Survival Analysis
    - Proportional Hazards Regression
  - Multivariate Analysis
  - Cluster Analysis
  - Power and Sample Size
  - Statistical Process Control
  - Combinatorics and Probability
  - Econometrics
  - Forecasting
  - Network Optimization
  - SAS Viya Prepare and Explore
    - Summary
    - Transform Data
    - Variable Selection
    - Sampling
    - Partitioning
    - Binning
    - Imputation
  - SAS Viya Unsupervised Learning
  - SAS Viya Supervised Learning
  - SAS Viya Evaluate and Implement
  - SAS Viya Text Analytics
- Snippets
- Libraries
- File Shortcuts

Nonparametric Survival Analysis ×

Settings | Code/Results | Split | Log | Code

DATA | OPTIONS | OUTPUT | INFORMATION

DATA

SASHELP.HEART

Filter: (none)

ROLES

Survival Time

Censoring type:

Right censoring

*Survival time: *(1 item)*

Column

*Status variable: *(1 item)*

Column

Censoring values: *(minimum 1 row)*

□ Competing risks analysis

Explanatory Variables

Group variable: *(1 item)*

Column

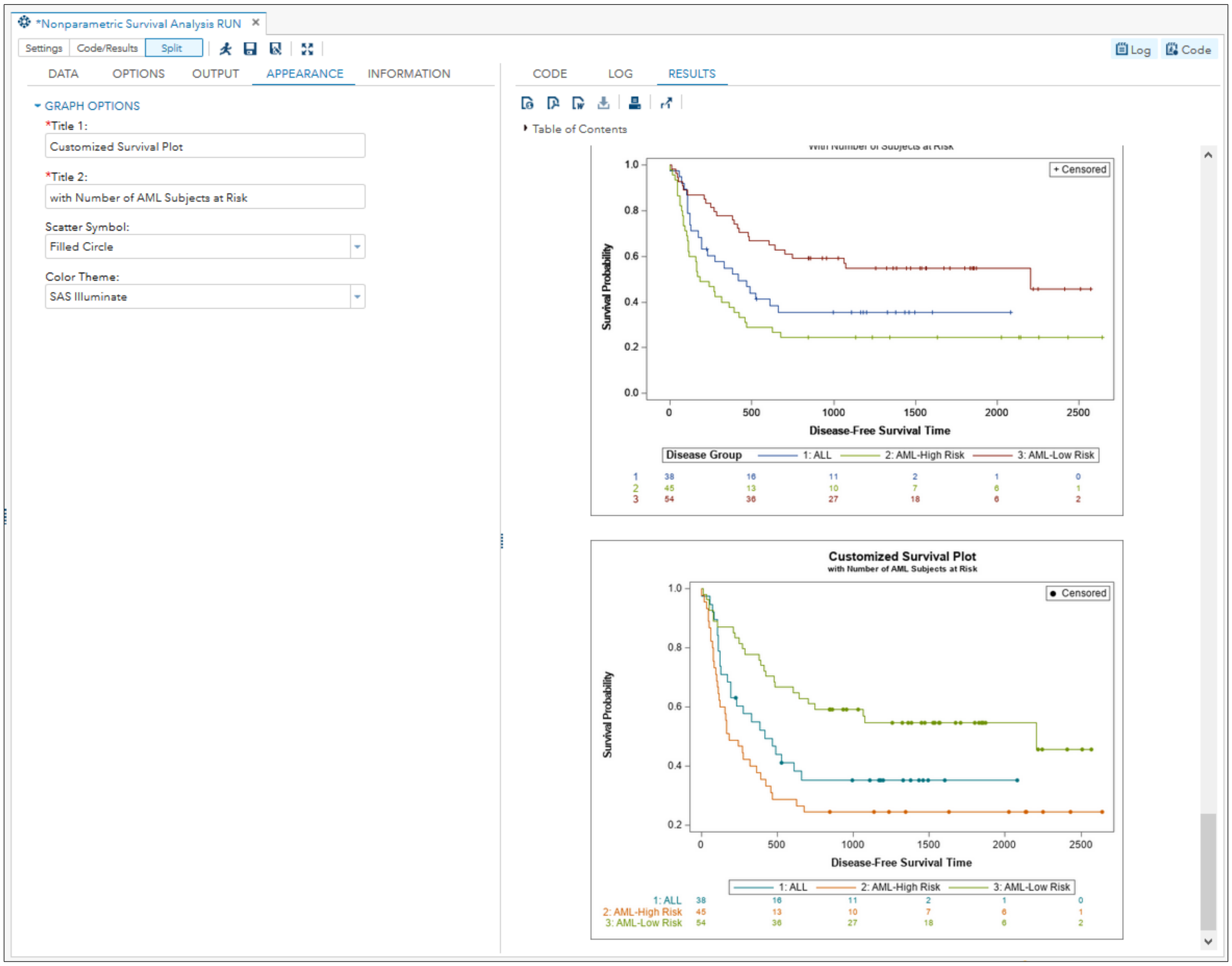
Stratify by: *(1 item)*

Column

ADDITIONAL ROLES

Figure 17. Built-In Survival Analysis Tasks (New in SAS Studio 3.8)

An edited version of the above built-in task can be seen in Figure 18. The task has the same APPEARANCE tab that was included in the Custom Task (Figure 16). Editing the built-in task is beneficial because it allows for everything to be done in one step, rather than having to run PROC LIFETEST and subsequently create the customized survival plot.



GRAPH OPTIONS

*Title 1: Customized Survival Plot

*Title 2: with Number of AML Subjects at Risk

Scatter Symbol: Filled Circle

Color Theme: SAS Illuminate

Figure 18. Added Appearance Tab in Built-in Task with Original and Customized Survival Plots

The output in the results window shows the original survival plot (created by default with PROC LIFETEST) followed by the customized graph (created using PROC SGPLOT).

CONCLUSION

SAS Studio Custom Tasks offer a way to streamline clinical graph creation. Using existing SGPLOT code for clinical graphs to create a Custom Task eliminates the need for memorization of SGPLOT syntax and options, which can be complex. Editing a built-in task to add graph customization can make the process even easier in some cases, creating the data necessary for the graph and generating the graph in one step. Custom tasks make updating, theming, or reusing graphs as easy as a click of a button.

REFERENCES

CUSTOM TASKS

Inman, Elliot, and Wright, Olivia. 2017. “Developing Your Own SAS Studio Custom Tasks for Advanced Analytics.” Proceedings of SAS Global Forum 2017, Orlando, FL. Available at https://support.sas.com/resources/papers/proceedings17/SAS0677-2017.pdf. Accessed April 29, 2019.

Wright, Olivia. 2019. “Custom Task Tuesday: Download the Ultimate Custom Task Cheat Sheet.” *SAS Support Communities*. Available at https://communities.sas.com/t5/SAS-Communities-Library/Custom-Task-Tuesday-Download-the-Ultimate-Custom-Task-Cheat/tac-p/553280#M3685. Accessed April 23, 2019.

Wright, Olivia. 2017. “Custom Task Tuesday Examples.” *SAS Software GitHub*. Available at https://github.com/sassoftware/sas-studio-tasks/tree/master/contributed/tasktuesdays. Accessed April 23, 2019.

CLINICAL GRAPHS

Matange, Sanjay. 2014. “Clinical Graphs.” *SAS Blogs*. Available at https://blogs.sas.com/content/graphicallyspeaking/2014/11/15/clinical-graphs/. Accessed April 23, 2019.

Matange, Sanjay. 2016. *Clinical Graphs Using SAS*. Cary, NC: SAS Institute Inc.

Matange, Sanjay. 2011. “It Pays to Be Discrete.” *SAS Blogs*. Available at https://blogs.sas.com/content/graphicallyspeaking/2011/11/10/it-pays-to-be-discrete/#prettyPhoto. Accessed April 23, 2019.

Matange, Sanjay. 2014. “Survival Plot.” *SAS Blogs*. Available at https://blogs.sas.com/content/graphicallyspeaking/2014/02/09/survival-plot/. Accessed April 23, 2019.

CONTACT INFORMATION

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

Olivia Wright
SAS Global Hosting and US Professional Services Division
(919) 531-3262
Olivia.Wright@sas.com
Twitter: @OliviaJWright
https://www.linkedin.com/in/oliviajwright/

Sanjay Matange
Individual Contributor
SanjayMatange@gmail.com