

## How to Create a Journal Quality Forest Plot with SAS® 9.4

John R. O’Leary, Department of Veteran Affairs; Warren F. Kuhfeld, SAS Institute Inc.

### ABSTRACT

This paper offers a solution for creating a high quality forest plot that is befitting even the most prestigious scientific journals. Building on code developed by Sanjay Matange and Prashant Hebbar, this paper provides intermediate SAS® programmers further versatility through a few simple enhancements. In particular, SG annotation is used to display text and handle special characters through its Unicode support (see Heath 2011). Additionally, error bar markers are sized based on subgroup population sizes to show treatment effectiveness, and hazard ratios (HR) are properly displayed on a log scale.

These enhancements became the basis for a forest plot figure that was published in the New England Journal of Medicine (NEJ, see Kernan et al., 2016) for the Insulin Resistance Intervention after Stroke (IRIS) study, a double-blind randomized control trial (RCT) in 3,876 participants designed to prevent secondary stroke or myocardial infarction in patients with insulin resistance. Although our plot sought to visually demonstrate the effectiveness of the treatment (prescription drug Pioglitazone) within subgroup populations compared to placebo, forest plots have many applications including displaying meta-analyses of related research studies. The simplicity of the Statistical Graphics (SG) procedures in SAS® 9.4 augmented by the power of both the Graph Template Language (GTL) and SG annotation will provide you with a powerful tool for forest plot construction.

### INTRODUCTION

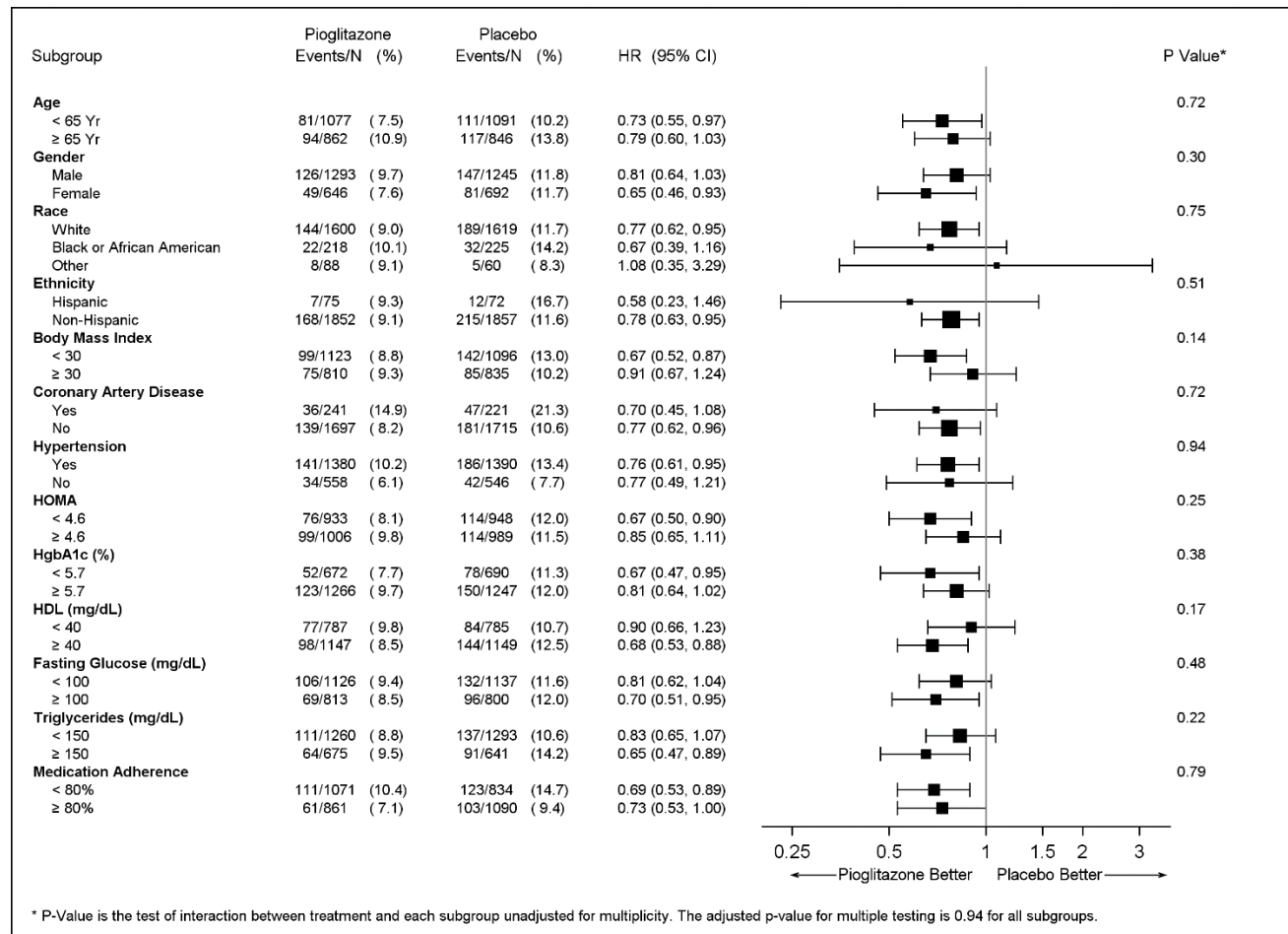
When the first author was tasked with creating a forest plot for the NEJ, he was initially relieved to learn of all the enhancements of the SG procedures, particularly the XAXISTABLE and YAXISTABLE statements. Axis tables display tabular information aligned with graphical information. Given the progression of improvements first in SAS® 9.2, then 9.3, and now 9.4 that have been well chronicled (see Matange 2012 and 2016), he anticipated a straightforward, simple solution. After first trying to use the YAXISTABLE in PROC SGPLOT, it became apparent that this problem had requirements that required the full power of the GTL (see Matange 2015).

Fortunately, when help was sought from the second author, it was gladly extended. This paper shares the solution that we jointly developed. Our hope is that more SAS programmers will consider using GTL when creating highly-customized forest plots and other figures. While it might be initially challenging for a beginning SAS programmer to delve into GTL, an intermediate or more experienced user who is willing to experiment will find this method to be highly customizable, offering the ability to precisely position and format text and labels.

The solution we offer borrows heavily from previous work (Matange 2014, 2016; Hebbar 2015) and includes four basic steps:

1. Create an input data set with subgroup statistics including counts, HRs and P-values used in columns 2 – 8 of the figure. Create the variable SquareSize that determines the marker size based on the subgroup population size used by the SIZERESPONSE= option for the scatter plot.
2. Create an SG annotation data set with two distinct purposes. First, use Unicode to replace '>=' with the desired operator '≥', to properly display the subgroup labels in column one. Next, set up the location of text and arrows that will go below the X axis of the scatter plot in column seven.
3. Create a StatGraph template named Forest by using PROC TEMPLATE and the power of GTL to lay out the eight columns of the figure. Use a scatter plot in column seven to display the 95% confidence intervals around the HRs of the subgroups examined in the IRIS RCT.
4. Pull the components together from the first three steps using the ODS Graphics procedure SGRENDER along with SG annotation to produce the desired output. (In our case we create PDF and EPS formatted output but there are many others to choose from including JPEG, PNG etc.)

Display 1 is a reduced version of the nine-inch-wide by six and one half inch high (or whatever size you choose) forest plot figure that you can produce by using these steps which are explained in more detail to follow.



Display 1. PharmaSUG2017\_ForestPlot.png produced by Appendix SAS® program code.

### STEP 1: CREATE THE INPUT DATA SET OF SUBGROUP STATISTICS

The initial step formats how the data will be displayed in the columns of the plot. There are ten variables that are read into the SubgroupData data set by the INPUT statement. The variable Indent specifies whether the row containing the description variable Subgroup is a header which is not indented or whether it is a category that needs to be indented. The remaining eight variables are numeric statistics that have previously been calculated by the SAS® procedures PROC FREQ and PROC PHREG. Note there are two treatment groups that are compared. For simplicity of coding, Pioglitazone is equivalent to Treatment A, while the Placebo group is known as Treatment B. PROC FREQ obtained the counts (CountA and CountB) of participants and the number of primary study events (EventA and EventB) occurring for the respective treatment groups within the subgroup categories. PROC PHREG produced the HRs and P-values showing the treatment effect of preventing stroke or myocardial infarction within the chosen subgroups of interest.

In order to provide some desired formatting to the counts and percentages (for example placing the percentages within parentheses) some values are constructed by using concatenation. Frequently in forest plots, the marker squares displayed on the 95% confidence error bars are all the same size which is perfectly fine. However, with just a single line of code, the variable SquareSize can be created and

used with the SIZERESPONSE= option of the scatter plot in the template Forest. By varying the size of the marker square based on the count of participants in the subgroup categories, the journal reader is better able to visualize the effect of the treatment.

The abbreviated coding for the first step is as follows:

```
data SubgroupData;
  input  Indent Subgroup $3-27 CountA EventA CountB EventB
        HR CIL CIU PValue;
  format HR CI_Lower CI_Upper PValue 7.2 CountA EventA CountB EventB 5.
  PctA PctB 4.1 EventDisplayA CountDisplayA EventDisplayB CountDisplayB $4.;

  if CountA ne . then do ;
    PctA      = (EventA/CountA)*100;
    PctB      = (EventB/CountB)*100;
    EventDisplayA = right(put(EventA, 4.0));
    CountDisplayA = left(put(CountA, 4.0));
    EventCountA  = right(EventDisplayA || "/" || CountDisplayA);
    PctDisplayA  = "(" || put(PctA, 4.1) || "%";
    EventDisplayB = right(put(EventB, 4.0));
    CountDisplayB = left(put(CountB, 4.0));
    EventCountB  = right(EventDisplayB || "/" || CountDisplayB);
    PctDisplayB  = "(" || put(PctB, 4.1) || "%";
    HRCI        = put(HR, 4.2) || " (" || put(CIU, 4.2) || "%", " ||
                  put(CIU, 4.2) || "%)";

    /* Determine the marker size based on population size */
    SquareSize = ((CountA + CountB)/3876) * 12; /* Max pixels 12 */
  end;
  row = _n_;

datalines;
0 Age . . . . . . . . 0.72
1 < 65 Yr 1077 81 1091 111 0.73 0.55 0.97 .
1 >= 65 Yr 862 94 846 117 0.79 0.60 1.03 .
0 Gender . . . . . . . . 0.30
1 Male 1293 126 1245 147 0.81 0.64 1.03 .
1 Female 646 49 692 81 0.65 0.46 0.93 .
0 Race . . . . . . . . 0.75
1 White 1600 144 1619 189 0.77 0.62 0.95 .
1 Black or African American 218 22 225 32 0.67 0.39 1.16 .
1 Other 88 8 60 5 1.08 0.35 3.29 .
... (Note: see Appendix for remainder of subgroup input)
;
```

## STEP 2: CREATE SG ANNOTATION DATA SET

The second step creates an SG annotation data set that positions text under the X axis of the scatter plot and adds Unicode characters. In our case we needed to display the operators '<' and '≥', and SG annotation is up to the task. SG annotation also comes in handy for displaying subscript and superscript characters (see Heath 2011). While some might want to avoid creating an annotate data set all together (admittedly it can be a bit cryptic), SG annotation can offer you a precise way to add text to plots even when using GTL. While some SAS experts might not prefer this method, as is usually the case, SAS offers multiple solutions to a problem, and this way worked for our purposes. You might find a different way.

If you have been coding long enough to have worked with Annotate in SAS/GRAPH®, you will find that SG annotation in ODS Graphics is very similar. For an introduction to SG annotation, see Kuhfeld

(2016a). Notice that the Forest template has two ANNOTATE statements, each with a different ID= value ('id1' and 'id2'). These match the two values of the ID variable in the SGANNO=Anno data set and provide annotation to two different parts of the graph.

The coding used to create the data set Anno is as follows:

```

/*--Used for Subgroup labels in column 1--*/
data anno(drop=indent);
  set SubgroupData(keep=row subgroup indent rename=(row=y1));
  retain Function 'Text ' ID 'id1' X1Space 'DataPercent'
         Y1Space 'DataValue ' x1 x2 2 TextSize 7 Width 100 Anchor 'Left ';
  if indent;
  label = tranwrd(subgroup, '>=', '(*ESC*){Unicode ''2265''x}');
  run;
/*--Used for text under x axis of HR scatter plot in column 7--*/
data anno2;
  retain Function 'Arrow' ID 'id2' X1Space X2Space 'DataValue'
         FillTransparency 0 Y1Space Y2Space 'GraphPercent' Scale 1e-40
         LineThickness 1 y1 y2 7.0 Width 100 FillStyleElement 'GraphWalls'
         LineColor 'Black';
  x1 = 0.90; x2 = 0.25; output;
  x1 = 2.10; x2 = 3.6; output;
  function = 'Text'; y1 = 7.0; y2 = 7.0;
  x1 = 0.92; anchor = 'Right'; label = 'Pioglitazone Better';
         Textsize=8; output;
  x1 = 1.07; Anchor = 'Left '; label = 'Placebo Better'; Textsize=8;
  output;
  run;
data anno; set anno anno2; run;

```

### STEP 3: USE GTL TO CREATE TEMPLATE FOREST

As noted previously, you might be able to produce a forest plot that meets your needs without combining GTL and SG Annotation. Yet if you find yourself with specifications being modified close to a publication deadline, and you already have a workable solution, there may not be time to seek out an alternative, simpler approach. So given that a GTL solution was already available (Matange, 2016) and nearly capable of meeting the NEJ specifications, all we really needed was to add some additional columns to the layout lattice and do a bit of housekeeping.

When GTL was first released in SAS® 9.2, as the foundation for the ODS Graphics System, a powerful new tool became available to SAS® users. Admittedly, GTL requires some learning to be able to manipulate the syntax, but having the ability to customize existing STATGRAPH templates to produce even better graphs is well worth the investment. A considerable amount of literature (Kuhfeld 2010, 2016b; Matange 2013) has been developed since the first release of GTL, and we recommend that you at least develop a basic understanding of how GTL operates behind the scenes of the ODS.

The Forest template (see Appendix) has an eight column lattice layout that controls the overall layout of the graph. The column widths are specified in the COLUMNWEIGHTS= option. The column heading text is supplied next in an additional lattice layout. Beneath the column headings, the forty rows of data from the input data set SubgroupData are straightforwardly placed into the columns using the AXISTABLE statement, which is the engine at the heart of this plot. AXISTABLE statements are used to layout the text in columns one through six and column eight. In column seven, the SCATTERPLOT statement is used to graph the HR and associated 95% confidence limits. For a better understanding of this template and the AXISTABLE statement, an excellent paper is available for your reference (see Hebbar, 2015).

The first column is constructed from the values of the Subgroup variable in the DATA= data set and the AXISTABLE statement, the values of the Label variable in the ID='id1' observations in the SG annotation data set, and an attribute map. The Subgroup variable in the DATA= data set contains the group headings. The Label variable in the SG annotation data set contains the subgroup values. These values

are indented by the Anchor = 'Left' and x1=2 variables. The attribute map, which begins with the DISCRETEATTRMAP statement in the template, is used along with the Head variable in the DATA= data set to display bold group headers.

As noted in Step 2, the SG annotation data set along with the two ANNOTATE statements are used to overlay text onto the graph in columns one and seven. By using the ID= option in the ANNOTATE statement, you can use multiple IDs to identify or subset, how the information in the annotation data set is added to your graph. Note that when ID equals 'id2', annotation is only added to column seven, which adds the arrows and text under the X axis. Being able to use multiple IDs in an annotate data set is a valuable tool which offers flexibility for handling a multitude of overlays.

The other aspect of the template that should be highlighted is the syntax for the scatter plot in the seventh column. Since hazard ratios and their associated confidence intervals or standard errors are exponential, they need to be displayed on a log scale. If they were displayed on a linear scale, they would be asymmetrical and misleading. Forest plots are sometimes incorrectly displayed on a linear scale in journal figures. Fortunately, this requirement was very easy to implement. By choosing the TYPE=LOG option and defining the appropriate tick values, the axis is statistically accurate.

## STEP 4: RUN PROC SGRENDER TO PULL IT ALL TOGETHER

The last step, PROC SGRENDER, requires very little code. That is because SAS software does all the work. After first defining what type of output you would like, along with the size and storage location for your output, you are ready to use PROC SGRENDER to render your output. The syntax in our case is pretty simple. We supply the required argument TEMPLATE= which specifies our STATGRAPH template, Forest; we use the optional SGANNO= option as described earlier; and then we supply two dynamic variables that are defined in our template. The dynamic variables enable horizontal reference lines. Set the color to light gray (for example, 'CXEEEEEE') instead of 'white' if you want alternate groups to be highlighted using colored bands. By running PROC SGRENDER twice with two different types of graphics output, we are able to satisfy requests for either a PNG file that is a compressed format which retains image quality, or an EPS raster-based file which allows the journal staff to easily move or change elements of your graph. EPS files are essential because even though you may think you have meticulously created the "perfect" graph, there is still a chance the editor may want to make a few tweaks to it (see Kernan et al., 2016, Figure 2).

## CONCLUSION

As with many things in life, we don't often develop new skills until we find them necessary. If and when you are faced with the task of developing a forest plot that is good enough to feature in a top-notch medical journal, you will discover that SAS® software is up to the challenge. While you may be reluctant to attempt using GTL and/or SG annotation, we hope this paper will provide you with the impetus to experiment with these valuable tools. The code we offer will likely be a good starting point, and when you add even better innovations, please share them with your time crunched SAS® community.

## REFERENCES

- Heath, Dan. *Now you can Annotate your Statistical Graphics Procedure Graphs*. Paper 277-2011, SAS® Global Forum 2011. Available at <https://support.sas.com/resources/papers/proceedings11/277-2011.pdf>
- Hebbar, Prashant (2015) *Lost in the Forest Plot? Follow the GTL AXISTABLE Road!*. Paper SAS1748-2015, SAS® Global Forum 2015. Available at <https://support.sas.com/resources/papers/proceedings15/SAS1748-2015.pdf>
- Kernan, W.N, Viscoli, C.N., Furie, K.L., Young, L.H., Inzucchi, S.E., ..., Windner, T.R. (2016) *Pioglitazone after Ischemic Stroke or Transient Ischemic Attack*. The New England Journal of Medicine, 374:1321-1331. <http://www.nejm.org/doi/pdf/10.1056/NEJMoa1506930>
- Kuhfeld, Warren, (2017) *Advanced ODS Graphics: Axis tables in PROC SGPlot and the GTL*. "Graphically Speaking Blog." Available at <http://blogs.sas.com/content/graphicallyspeaking/2017/01/11/advanced-ods-graphics-axis-tables-in-proc->

[sgplot-and-gtl/](#) Accessed on March 11, 2017

Kuhfeld, Warren, *Advanced ODS Graphics Examples*. Cary, NC: SAS Institute Inc. 2016a Available at <http://support.sas.com/documentation/prod-p/grstat/9.4/en/PDF/odsadvq.pdf>

Kuhfeld, Warren, *Basic ODS Graphics Examples*. Cary, NC: SAS Institute Inc. 2016b Available at <http://support.sas.com/documentation/prod-p/grstat/9.4/en/PDF/odsbasicg.pdf>

Kuhfeld, Warren. (2010) *Statistical Graphics in SAS® An Introduction to the Graph Template Language and the Statistical Graphics Procedures*. Cary, NC: SAS Institute Inc.

Matange, Sanjay. (2016) *CTSPedia Clinical Graphs – Subgrouped Forest Plot*. “Graphically Speaking Blog.” Available at <http://blogs.sas.com/content/graphicallyspeaking/2016/05/30/ctspedia-clinical-graphs-subgrouped-forest-plot/>. Accessed on March 11, 2017.

Matange, Sanjay. (2012) *Forest Plot with Subgroups*. “Graphically Speaking Blog.” Available at <http://blogs.sas.com/content/graphicallyspeaking/2012/09/30/forest-plot-with-subgroups/>. Accessed on March 11, 2017.

Matange, Sanjay. (2013) *Getting Started with the Graph Template Language in SAS*. Cary, NC: SAS Institute Inc.

Matange, Sanjay. (2015) *Graphs are Easy with SAS® 9.4*. Paper SAS1780-2015, SAS® Global Forum 2015. Available at <https://support.sas.com/resources/papers/proceedings15/SAS1780-2015.pdf>

Matange, Sanjay. (2014) *Up Your Game with Graph Template Language Layouts*. Paper 2014-SS07, MWSUG 2014. Available at <http://www.mwsug.org/proceedings/2014/SS/MWSUG-2014-SS07.pdf>

Wicklin, Rick. *Odds Ratios with a Logarithmic Scale in SAS*. “The Do Loop Blog.” <http://blogs.sas.com/content/iml/2015/07/29/or-plots-log-scale.html>. Accessed on March 11, 2017.

## ACKNOWLEDGMENTS

Mr. O’Leary would like to thank the participants who volunteered for the IRIS study, my talented colleagues who are always willing to offer technical assistance and support, and all the SAS® authors who have shared their work and made my job a lot easier over the years. Thanks especially to Warren Kuhfeld and Sanjay Matange who patiently offered help and creative ideas and were the key ingredients that permitted me to meet our IRIS study deadlines.

The second author would like to thank Sanjay Matange, Dan Heath, Prashant Hebbar, and Lingxiao Li for all of their help with ODS Graphics over the years.

## CONTACT INFORMATION

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

John O’Leary  
Veterans Affairs Connecticut Healthcare  
950 Campbell Avenue, MZ 151B  
West Haven, CT 06419  
(203)-932-5711 ext. 2402  
[john.oleary@va.gov](mailto:john.oleary@va.gov)

Warren F. Kuhfeld  
SAS Institute Inc.  
Cary, NC, 27513 (919) 531-7922  
[Warren.Kuhfeld@sas.com](mailto:Warren.Kuhfeld@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

Below is the entire code for the program PharmaSUG2017\_ForestPlot.sas that produces Display 1.

```

/*
program:  PharmaSUG2017_ForestPlot
function: Forest Plot of Subgroups used for IRIS RCT
Author:   Sanjay Matange
          (https://www.ctspedia.org/do/view/CTSpedia/ClinAEGraph001)
Modified: John O'Leary & Warren Kuhfeld January 2016 &
          March 2017 for PharmaSUG conference
*/

/* Dimensions and resolution for output graph used by SGRender */
%let dpi=300;
%let w=9in;
%let h=6.5in;

data SubgroupData;
  input  Indent Subgroup $3-27 CountA EventA CountB EventB HR CIL CIU PValue;
  format HR CIL CIU PValue 7.2 CountA EventA CountB EventB 5. PctA PctB 4.1;
  format EventDisplayA CountDisplayA EventDisplayB CountDisplayB $4. ;

  if CountA ne . then do;

      PctA = (EventA/CountA)*100;
      PctB = (EventB/CountB)*100;

      EventDisplayA = right(put(EventA, 4.0));
      CountDisplayA = left(put(CountA, 4.0));
      EventCountA = right(EventDisplayA || "/" || CountDisplayA);
      PctDisplayA = "(" || put(PctA, 4.1) || "%";

      EventDisplayB = right(put(EventB, 4.0));
      CountDisplayB = left(put(CountB, 4.0));
      EventCountB = right(EventDisplayB || "/" || CountDisplayB);
      PctDisplayB = "(" || put(PctB, 4.1) || "%";

      HRCI = put(HR, 4.2) || " (" || put(CIL, 4.2) || ", "
                || put(CIU, 4.2) || "%)";

      /* Determine the marker size based on population size */
      SquareSize = ((CountA + CountB)/3876) * 12 ;

  end;
  row = _n_;

datalines;
0 Age . . . . . 0.72
1 < 65 Yr 1077 81 1091 111 0.73 0.55 0.97 .
1 >= 65 Yr 862 94 846 117 0.79 0.60 1.03 .
0 Gender . . . . . 0.30
1 Male 1293 126 1245 147 0.81 0.64 1.03 .
1 Female 646 49 692 81 0.65 0.46 0.93 .

```

0 Race	.	.	.	.	.	.	.	.	0.75
1 White	1600	144	1619	189	0.77	0.62	0.95	.	.
1 Black or African American	218	22	225	32	0.67	0.39	1.16	.	.
1 Other	88	8	60	5	1.08	0.35	3.29	.	.
0 Ethnicity	.	.	.	.	.	.	.	.	0.51
1 Hispanic	75	7	72	12	0.58	0.23	1.46	.	.
1 Non-Hispanic	1852	168	1857	215	0.78	0.63	0.95	.	.
0 Body Mass Index	.	.	.	.	.	.	.	.	0.14
1 < 30	1123	99	1096	142	0.67	0.52	0.87	.	.
1 >= 30	810	75	835	85	0.91	0.67	1.24	.	.
0 Coronary Artery Disease	.	.	.	.	.	.	.	.	0.72
1 Yes	241	36	221	47	0.70	0.45	1.08	.	.
1 No	1697	139	1715	181	0.77	0.62	0.96	.	.
0 Hypertension	.	.	.	.	.	.	.	.	0.94
1 Yes	1380	141	1390	186	0.76	0.61	0.95	.	.
1 No	558	34	546	42	0.77	0.49	1.21	.	.
0 HOMA	.	.	.	.	.	.	.	.	0.25
1 < 4.6	933	76	948	114	0.67	0.50	0.90	.	.
1 >= 4.6	1006	99	989	114	0.85	0.65	1.11	.	.
0 HgbA1c (%)	.	.	.	.	.	.	.	.	0.38
1 < 5.7	672	52	690	78	0.67	0.47	0.95	.	.
1 >= 5.7	1266	123	1247	150	0.81	0.64	1.02	.	.
0 HDL (mg/dL)	.	.	.	.	.	.	.	.	0.17
1 < 40	787	77	785	84	0.90	0.66	1.23	.	.
1 >= 40	1147	98	1149	144	0.68	0.53	0.88	.	.
0 Fasting Glucose (mg/dL)	.	.	.	.	.	.	.	.	0.48
1 < 100	1126	106	1137	132	0.81	0.62	1.04	.	.
1 >= 100	813	69	800	96	0.70	0.51	0.95	.	.
0 Triglycerides (mg/dL)	.	.	.	.	.	.	.	.	0.22
1 < 150	1260	111	1293	137	0.83	0.65	1.07	.	.
1 >= 150	675	64	641	91	0.65	0.47	0.89	.	.
0 Medication Adherence	.	.	.	.	.	.	.	.	0.79
1 < 80%	1071	111	834	123	0.69	0.53	0.89	.	.
1 >= 80%	861	61	1090	103	0.73	0.53	1.00	.	.

```

;

/*--Used for Subgroup labels in column 1--*/
data anno(drop=indent);
  set SubgroupData(keep=row subgroup indent rename=(row=y1));
  retain Function 'Text ' ID 'id1' X1Space 'DataPercent'
         Y1Space 'DataValue ' x1 x2 2 TextSize 7 Width 100 Anchor 'Left ';
  if indent;
  label = tranwrd(subgroup, '>=', '(*ESC*){Unicode '2265'x}');
run;

/*--Used for text under x axis of HR scatter plot in column 7--*/
data anno2;
  retain Function 'Arrow' ID 'id2' X1Space X2Space 'DataValue'
         FillTransparency 0 Y1Space Y2Space 'GraphPercent'
         Scale 1e-40 LineThickness 1 y1 y2 7.0 Width 100
         FillStyleElement 'GraphWalls' LineColor 'Black';
  x1 = 0.90; x2 = 0.25; output;
  x1 = 2.10; x2 = 3.6; output;
  function = 'Text'; y1 = 7.0; y2 = 7.0;
  x1 = 0.92; anchor = 'Right'; label = 'Pioglitazone Better'; Textsize=8;
  output;

```



```

    x1 = 1.07; Anchor = 'Left '; label = 'Placebo Better'; Textsize=8; output;
    run;
data anno; set anno anno2; run;

data forest2(drop=flag);
  set SubgroupData nobs=nobs;
  Head = not indent;
  retain flag 0;
  if head then flag = mod(flag + 1, 2);
  if flag then ref=row;
  if indent then subgroup = ' ';
  run;

/*--Define template for Forest Plot--*/
/*--Template uses a Layout Lattice of 8 columns--*/
proc template;
  define statgraph Forest;
    dynamic /*_show_bands*/ _color _thk;
    begingraph;
      discreteattrmap name='text';
      value '1' / textattrs=(weight=bold); value other;
    enddiscreteattrmap;
    discreteattrvar attrvar=type var=head attrmap='text';

    layout lattice /
      columns=8 columnweights=(0.21 0.06 .07 0.06 .07 0.12 0.33 0.09);

    /*--Column headers--*/
    sidebar / align=top;
    layout lattice /
      rows=2 columns=5 columnweights=(0.21 0.12 0.13 0.40 0.14);
      entry " ";
      entry textattrs=(size=8) halight=left " Pioglitazone";
      entry textattrs=(size=8) halight=left " Placebo";
      entry " ";
      entry " ";
      entry textattrs=(size=8) halight=left "Subgroup";
      entry textattrs=(size=8) halight=left "Events/N (%)";
      entry textattrs=(size=8) halight=left " Events/N (%)";
      entry textattrs=(size=8) halight=left " HR (95% CI)";
      entry halight=center textattrs=(size=8) "P Value*" ;
    endlayout;
  endsidebar;

  /*--First Subgroup column, shows only the Y2 axis--*/
  layout overlay / walldisplay=none xaxisopts=(display=none)
    yaxisopts=(reverse=true display=none
      tickvalueattrs=(weight=bold));
    annotate / id='id1';
    referenceline y=ref / lineattrs=(thickness=_thk color=_color);
    axistable y=row value=subgroup /
      display=(values) textgroup=type;
  endlayout;

  /*--Second column showing Pioglitazone Events/Count --*/
  layout overlay / xaxisopts=(display=none)
    yaxisopts=(reverse=true display=none walldisplay=none);

```

```

referenceline y=ref / lineattrs=(thickness=_thk color=_color);
axistable y=row value=EventCountA /display=(values)
          valuejustify = center;
endlayout;

/*--Third column showing Pioglitizone Percent--*/
layout overlay / xaxisopts=(display=none)
  yaxisopts=(reverse=true display=none) walldisplay=none;
referenceline y=ref / lineattrs=(thickness=_thk color=_color);
axistable y=row value=PctDisplayA /display=(values)
          valuejustify = left;
endlayout;

/*--Fourth column showing Placebo Events/Count --*/
layout overlay / xaxisopts=(display=none)
  yaxisopts=(reverse=true display=none) walldisplay=none;
referenceline y=ref / lineattrs=(thickness=_thk color=_color);
axistable y=row value=EventCountB /display=(values)
          valuejustify = center;
endlayout;

/*--Fifth column showing Placebo Percent--*/
layout overlay / xaxisopts=(display=none)
  yaxisopts=(reverse=true display=none) walldisplay=none;
referenceline y=ref / lineattrs=(thickness=_thk color=_color);
axistable y=row value=PctDisplayB /display=(values)
          valuejustify = left;
endlayout;

/*--Sixth column showing PCIGroup--*/
layout overlay / x2axisopts=(display=none)
  yaxisopts=(reverse=true display=none) walldisplay=none;
referenceline y=ref / lineattrs=(thickness=_thk color=_color);
axistable y=row value=HRCI / display=(values);
endlayout;

/*--Seventh column showing Hazard ratio graph with 95% error bars--*/
layout overlay / xaxisopts=(type=log
  label=' '
  labelattrs=(size=10)
  logopts=(tickvaluepriority=true
  tickvaluelist=(0.25 0.5 1.0 1.5 2.0 3.0)))
  yaxisopts=(reverse=true display=none) walldisplay=none;
annotate / id='id2';
referenceline y=ref / lineattrs=(thickness=_thk color=_color);
scatterplot y=row x=HR / xerrorlower=CIL xerrorupper=CIU
  sizeresponse=SquareSize sizemin=4 sizemax=12
  markerattrs=(symbol=squarefilled);
referenceline x=1;
endlayout;

/*--Eighth column showing P-Values--*/
layout overlay / x2axisopts=(display=none)
  yaxisopts=(reverse=true display=none) walldisplay=none;
referenceline y=ref / lineattrs=(thickness=_thk color=_color);
axistable y=row value=pvalue / display=(values)
          valuejustify = right

```

```

        showmissing=false; /*false removes . for missing pvalues*/
    endlayout;

    endlayout;
    entryfootnote halign=left textattrs=(size=7)
        '* P-Value is the test of interaction between treatment and each
subgroup unadjusted for multiplicity. The adjusted '
        'p-value for multiple testing is 0.94 for all subgroups.';
    endgraph;
end;
run;

/*----Create Graph with PNG output: Raster graphics file format----*/
ods html close;
ods listing gpath="C:\PharmaSUG2017\SAS" image_dpi=&dpi;
ods graphics / reset noscale IMAGEFMT=PNG
    width=&w height=&h imagename='PharmaSUG2017_ForestPlot';
proc sgrender data=Forest2 template=Forest sganno=anno;
dynamic _color='white' _thk=12 ; /*-Essentially invisible band-*/
run;

/*----Create Graph with EPS output: Vector based graphics format----*/
/*----Easily editable by publisher with Adobe Illustrator and Corel Draw----*/
ods listing gpath="C:\PharmaSUG2017\SAS" image_dpi=&dpi;
ods graphics / reset noscale IMAGEFMT=EPS
    width=&w height=&h imagename='PharmaSUG2017_ForestPlot';
proc sgrender data=Forest2 template=Forest sganno=anno;
dynamic _color='white' _thk=12 ;
run;

```