PharmaSUG 2015 - Paper DV13

Forest Plots: Old Growth versus GMO (Genetically Modified Organism)

Scott Horton, Experis Clinical, Kalamazoo, Michigan

ABSTRACT

By their nature forest plots (e.g. plotting hazard ratios with their confidence intervals for different factors), created via the SASGRAPH GPLOT Procedure, point to the use of the Annotate Facility to produce figures that are informative and inviting in structure and read-ability. We will explore different ways to produce forest plots that vary in the amount of annotation used to produce the figures and assess the plusses and minuses of the presented approaches.

INTRODUCTION

Forest plots are concise graphical displays that allow a reviewer to quickly scan for hazard ratios or risk ratios that are of statistical or clinical significance. They are commonly included in statistical analysis plans of clinical trials, including time-to-event analyses that are typical in cancer trials. Though the review of the statistics of interest can also be done through a table format, a graphical display appeals to the eye and allows location and spread differences to be compared with more speed and ease. The creation of forest plots varies in structure and complexity. We will look at several different approaches to creating this type of figure and the trade-offs that exist when considering how customized the resulting plot can or should be. PROC GPLOT and the GSLIDE PROCEDURE will be used to produce the figures in conjunction with the Annotate Facility.

OLD GROWTH FOREST VERSUS GMO (GENETICALLY MODIFIED ORGANISM)

Opinions vary as to when a mature forest becomes an old-growth forest—an age of 250 to 350 years is often cited. Many factors, including soil conditions and other site qualities, determine the age at which a forest will take on the structural qualities of true old-growth. An old-growth forest is far more structurally diverse than a typical tree plantation. Consequently, associated life forms are far different than those found in a young, second growth forest. (National Park Service)

When a gene from one organism is purposely moved to improve or change another organism, the result is a genetically modified organism (GMO). It can also be called "transgenic" for transfer of genes. There are different ways of moving genes to produce desirable traits. For both plants and animals, one of the more traditional ways is through selective breeding. For example, a plant with a desired trait is chosen and bred to produce more plants with the desirable trait. More recently genes that express the desired trait are physically moved or added to a new plant to enhance the trait in that plant. (University of California at San Diego)

For our purpose, we will consider constructing forest plots using PROC GPLOT without the Annotate Facility as an Old Growth approach; and using varying amounts of the Annotate Facility with PROC GSLIDE as a GMO approach. The approach that might meet a project's needs may vary between projects and could also vary within a project itself.

QUICK AND EASY FOREST PLOTS USING PROC GPLOT

Forest plots can be created with the simplest lines of code. After producing the desired statistics, a simple PROC GPLOT can quickly produce a forest plot. For this paper, we will use hazard ratios with their 95% confidence limits; however, forest plots are not limited to ratios (e.g. using means with their individual confidence intervals also can be displayed in a forest plot when displaying similar results across studies or for a meta-analysis). The ANNOTATE FACILITY was not used in producing the figures in this section of the paper. Y is an index variable with different values for each factor which is being analyzed for its hazard ratio. X contains the values of the hazard ratio and the corresponding lower and upper confidence limits.

DO-IT-IN-YOUR-SLEEP FOREST PLOT

Figure 1 demonstrates how quickly a plot can be produced for clinical review.

```
symbol&i. line=1 interpol=join width=3 value=dot;  /* inside a macro do loop */
...

axis1 label=None major=none minor=none value=none;
axis2 label=("Hazard Ratio (95% Confidence Interval)");
proc gplot;
   plot y*x=factor /vaxis=axis1 haxis=axis2;
```

Figure 1 below is the result of this very simple code. You will note that presenting so many factors becomes problematic when identifying them by color. Symbol statements that specify unique symbols can be used, but this approach will not look as clean when many factor levels are being displayed. In addition, the size of the legend and the text identifying certain factors within the legend are also problems that can arise (as noted below).

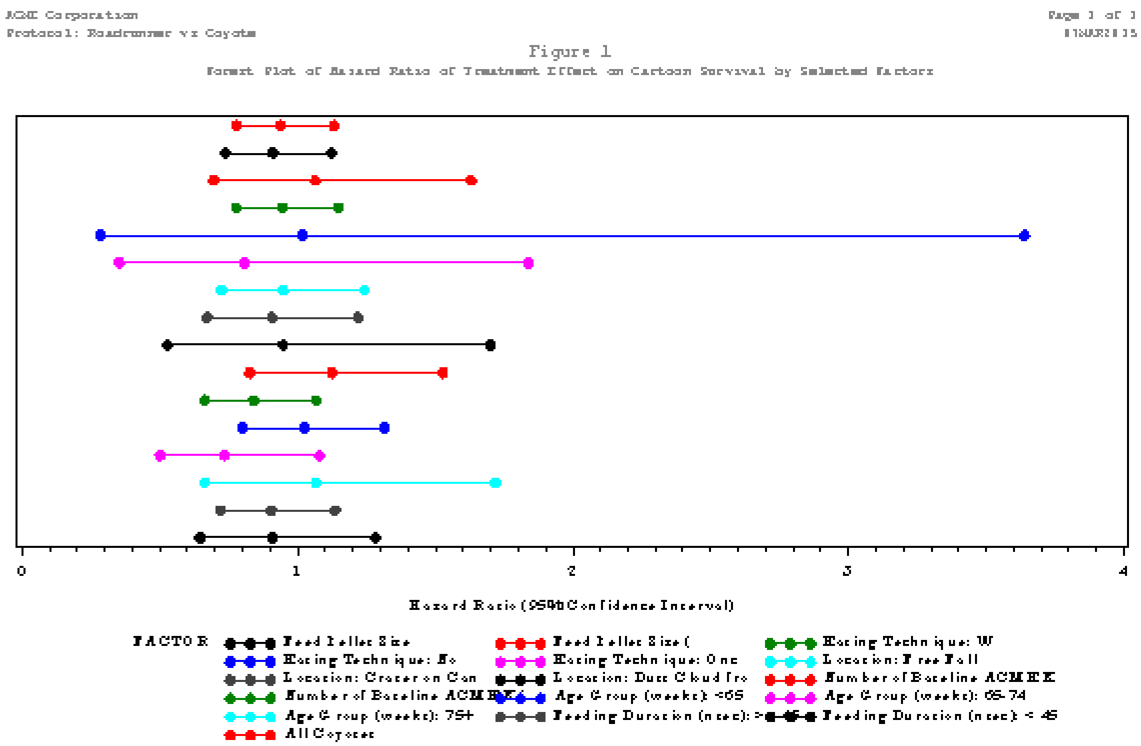


Figure 1. Simple linear-based forest plot

DO-IT-IN-YOUR-SLEEP PLOT WITH A LOG SCALE AND WITH VERTICAL REFERENCE LINES

Figure 1 displays the results on a linear scale. This is usually not desirable for forest plots of ratios and is easily remedied with an adjustment to the definition of the horizontal axis. Since forest plots are designed to review statistics including the hazard or risk of different levels of factors to each other (a ratio), sometimes clinicians like to have vertical reference lines that allow the reviewer to see if a ratio falls outside of a defined range of values. The simple addition of the HREF= option along with LHREF= option allow these reference lines to be plotted on the figure in addition to a reference line at the value of one. A ratio of one represents the hazard or risk as equally likely in the two levels of the factor being compared.

```
axis3 label=("Hazard Ratio (95% Confidence Interval)") logbase=10;

proc gplot;
   plot y*x=factor /vaxis=axis1  haxis=axis3 href=(0.8 1 1.25) lhref=(2 1 2);
```

Since hazard ratios are bounded by zero and ∞ with one as the center point, plotting on a log scale displays a visually-balanced confidence interval around its hazard ratio estimate. In dealing with clinical data, it has been my experience that ratios and their confidence intervals do not stray too far towards either zero or ∞, and thus dealing with graphs that attempt to display such extreme values does not usually occur in the realm of clinical data. The results take us essentially where we want to go without using the ANNOTATE FACILITY. It provides the clinical reviewer with all of the data that is needed (ignoring the color and symbol issues noted above).

You will note that with the addition of a vertical reference line at the value of one, plotting the confidence intervals for the factor levels being compared tends to visually represent a tree to a reviewer's eye; though the name forest can be said to originate from a "forest" of lines being displayed in the plot.

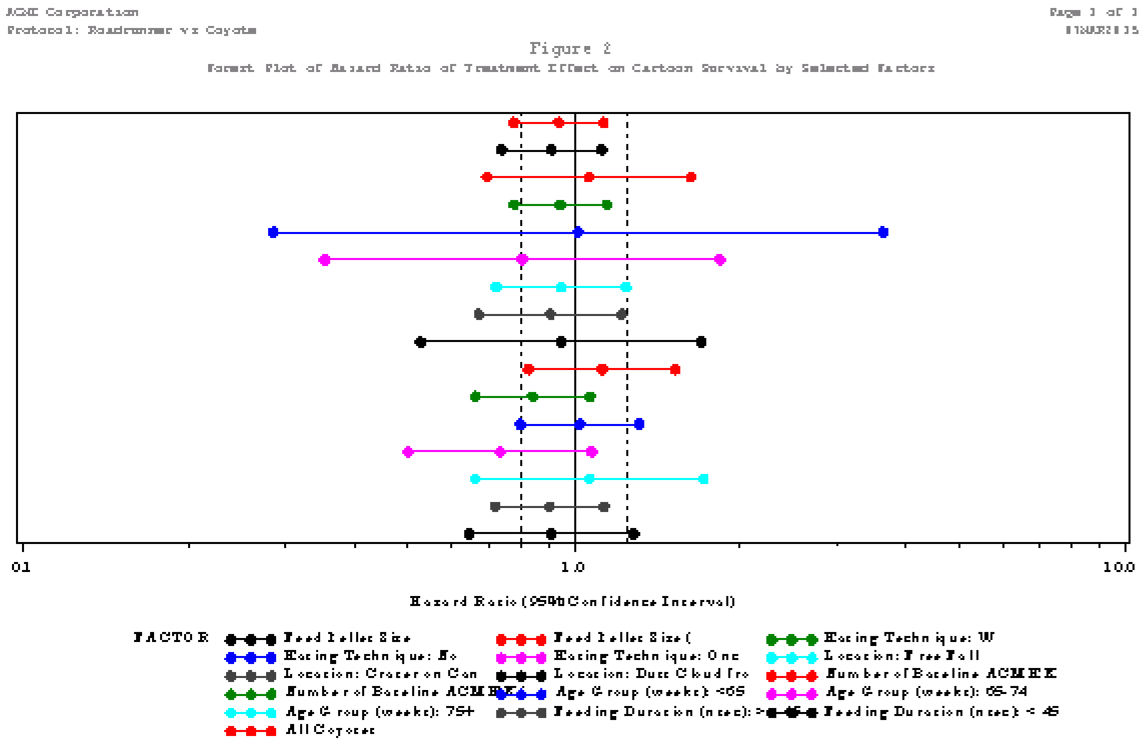


Figure 2. Simple forest plot on log-scale with reference lines

PLOTS USING THE ANNOTATION FACILITY

The ANNOTATE FACILITY provides the flexibility to turn figures into output that meets specific requests of the statistical analysis plan or target clinical reviewer. The contributions to a plot made via the ANNOTATE FACILITY can range from being simple and limited to elaborate and eye-catching. Sometimes the only limit on what is produced is the time available to the programmer or the clinical project deadline.

SIMPLE LABELING OF THE VERTICAL REFERENCE LINES

Let's suppose we receive a request to label the two vertical reference lines that were added to Figure 2. This can be done with the ANNOTATE FACILITY. The use of the facility can be performed by just annotating within one of three defined areas: data area, graphical area, or procedure area. Functions (e.g. %label) can be used to build the data used for annotation. The code below specifies that the graphical area will be utilized.

```
%annomac;              /* compiles the ANNOTATE macros */
...

data reflines;
   %dclanno;              /* sets the type and length of ANNOTATE variables */
   retain xsys ysys '3'; /* using coordinates based on graphics area */
   %label(62,10.2,'1.25',black,0,0,0.90,,5);
   %label(51,10.2,'0.8',black,0,0,0.90,,5);
...

proc gplot data=phregxy anno=reflines;
   plot y*x=factor /vaxis=axis1  haxis=axis3 href=(0.8 1 1.25) lhref=(2 1 2);
```

Figure 3 shows the labeling of the reference lines via the ANNOTATE FACILITY.

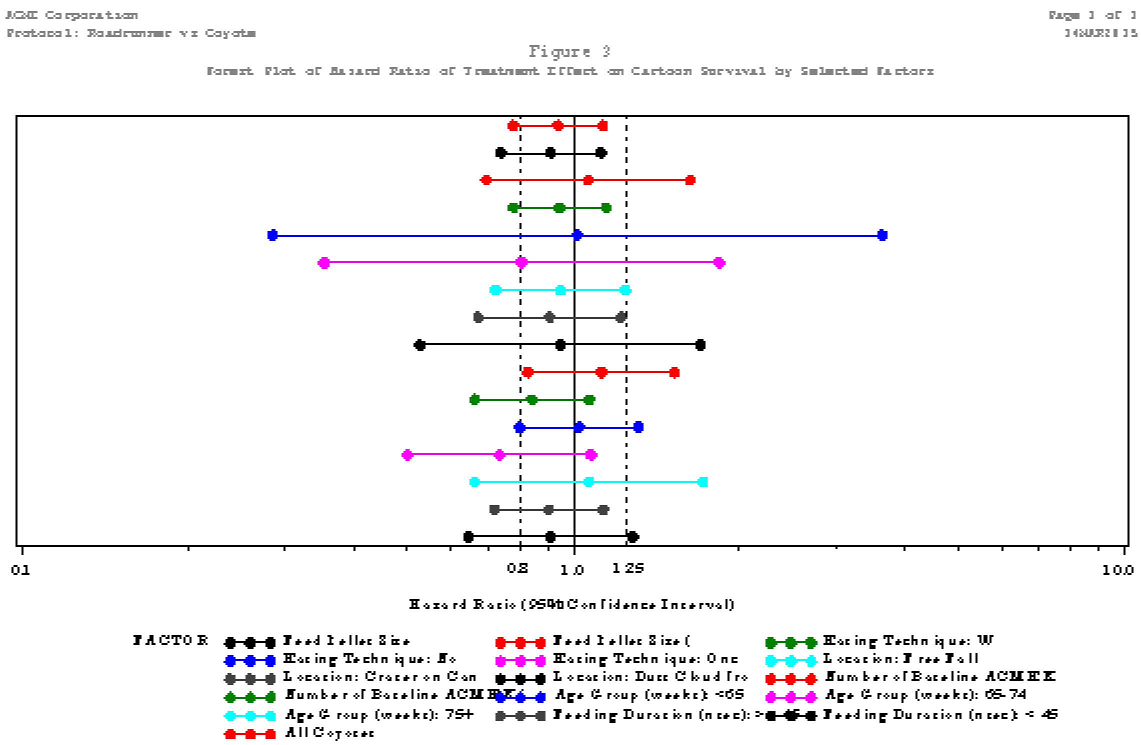


Figure 3. forest plot with simple annotation

Annotate Macros versus other DATA Step Code

As noted, Annotate macros (e.g. %label) can be used to build the data used for annotation. But, the data can also be built via code that does not utilize the macros that were created by SAS® for the ANNOTATION FACILITY.

The %label function has the following parameters: horizontal coordinate, vertical coordinate, text, color, angle of the text string, angle of each character in the text string, size of the font, font, and relative position of the text. It can replace more verbose code in a DATA step creating data that the annotations will be based on.

```
function='label';
x=51;
y=10.2;
text='0.8';
color='black';
angle=0;
rotate=0;
size=0.90;
position=5;
output;
```

Frequent use of the macros keeps the code cleaner.

ANNOTATE AS THE MAIN BUILDER OF A FIGURE INSTEAD OF GPLOT

We have been creating forest plots so far via PROC GPLOT. However, if we feel that “genetic engineering” (i.e. using the ANNOTATE FACILITY) is not only needed, but desirable, we can change course and create the entire plot by using PROC GSLIDE instead of PROC GPLOT. The initial investment in creating the annotation data set is non-trivial, but the resulting possibilities are almost limitless. To accomplish this, a few very simple macros will prove quite useful in creating the data; and some initial macro variables will be defined (which variables can be included in macro calls for different plots, if desired, for flexibility). All code shown is based on plotting via ANNOTATE and XSYS=3 and YSYS=3.

```
%let xstart=25;  /* Start plot of X values at 25% of graphical area */
%let xlen=95;    /* Percentage of graphical area to use for X axis */
```

```
%let ystart=15;  /* Start plot of Y values at 15% of graphical area */
%let ymax=79;    /* End plot of Y values at 79% of graphical area */
%let lblrowhgt=%str((100/44));    /* % decrement between rows of column labels */
```

Macro NEXTROW is used to decrement Y values (plotting is done from the top of the figure) so each factor level is spaced evenly when being plotted. The value stored in macro variable NUMLINES is determined prior to the annotate data set by just counting the total number of factor levels to be displayed and also counting any grouping labels for types of factors that will also be displayed.

```
%macro nextrow;
   yvalue=yvalue-%eval(&ylen./&numlines.); /* decrement due to # factor levels */
%mend;
```

Macro CONVERTX is used to convert the X value to its corresponding percentage since we are plotting based on percentage of the graphical area. No semicolon ends the specific line of code since this macro is designed to be used as part of a line of DATA step code.

```
%macro convertx(xval);
   (&xval.-&xmin.)*(&xlen./(&xmax.-&xmin.))+&xstart.
%mend convertx;
```

Data used for annotation is then built via code in a DATA step. An example of such code is below. Use of “FIRST.” variables are needed because these values are repeated due to the nature of the annotation data set being built.

```
*** PLOT THE TYPE OF FACTOR HEADER ***;
if first.factcatcd and n(factcatcd) then do;
   %nextrow
   %label(0,yvalue,factcat,black,0,0,0.95,,6);    /* Label the type of factors */
   *** UNDERLINE STRATIFICATION HEADER ROWS (EX: STRAT AND NON-STRAT ROWS)***;
   %nextrow
   %line(0,yvalue,length(strip(factcat))-5,y,black,1,1);
   %nextrow
end;

*** FACTOR BEING PLOTTED ***;
if first.paramn and paramn then do;
   %label(0,yvalue,factlbl,black,0,0,0.95,,6); /* Label the level of the factor */
   if n(factcatcd) then %nextrow
end;
```

X-values are converted to corresponding % of graphical area values:

```
*** CONVERT X VALUES TO % COORDINATES ***;
if n(hrlowercl) then   xlcl = %convertx(hrlowercl);
if n(hazardratio) then xhr = %convertx(hazardratio);
if n(hruppercl) then   xucl = %convertx(hruppercl);
```

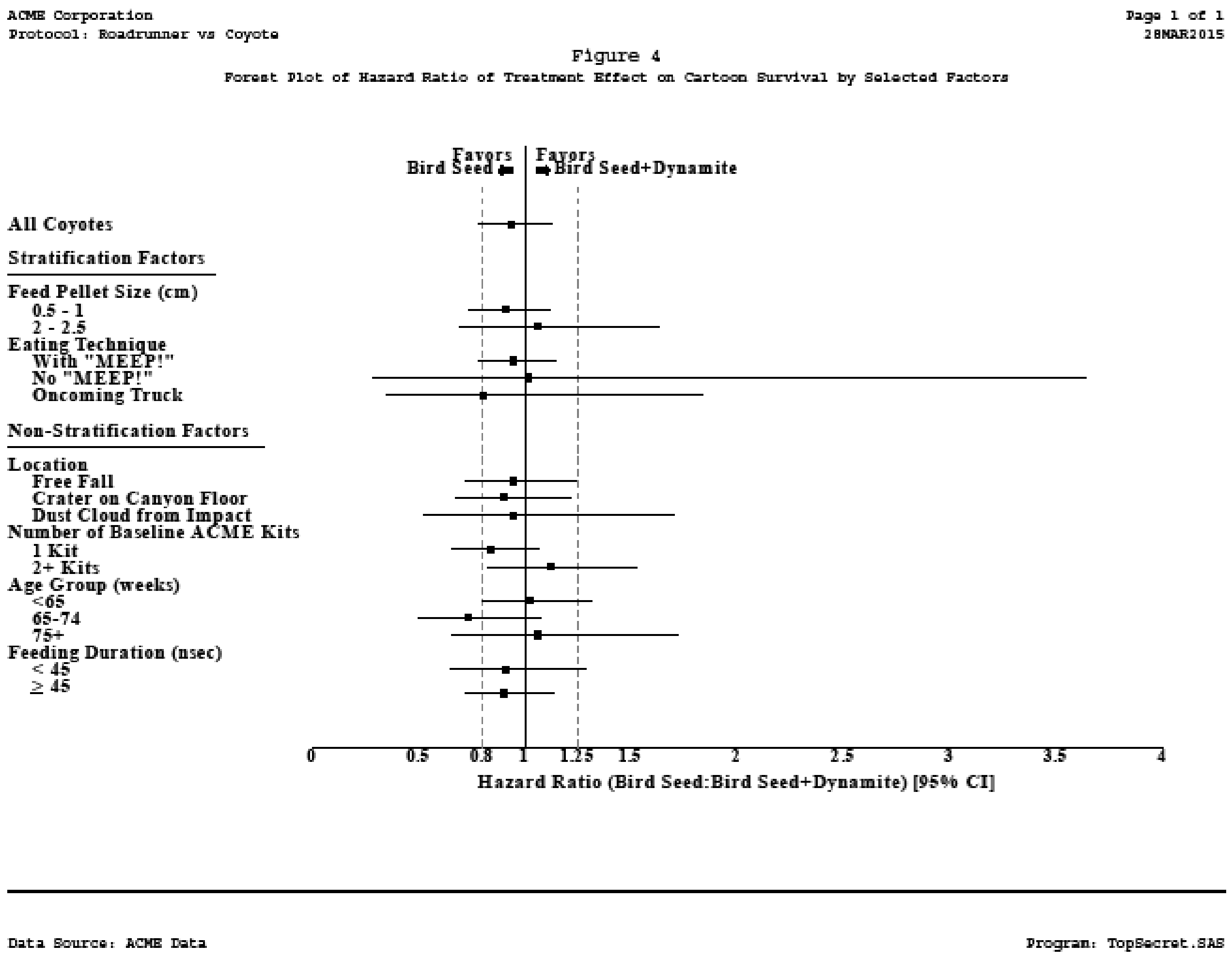
The hazard ratio is plotted as a box character:

```
if n(hazardratio) then do;
   %label(xhr,yvalue,"U",black,0,0,0.35,marker,5);
end;
```

The confidence interval is displayed as a line:

```
if nmiss(hrlowercl, hruppercl)=0 then do;
   %line(xlcl,yvalue,xucl,yvalue,black,1,1);
end;
```

We also replace the legend created when using PROC GPLOT by labeling the data on the same level where the hazard ratio and confidence interval are plotted (code creating labels seen above in `*** PLOT THE TYPE OF FACTOR HEADER ***` and `*** FACTOR BEING PLOTTED ***`). Figure 4 displays the resulting plot created via PROC GSLIDE and using the ANNOTATE FACILITY.



ACME Corporation
Protocol: Roadrunner vs Coyote

Page 1 of 1
28MAR2015

Figure 4
Forest Plot of Hazard Ratio of Treatment Effect on Cartoon Survival by Selected Factors

Data Source: ACME Data

Program: TopSecret.SAS

Figure 4. forest plot created by PROC GSLIDE via the ANNOTATE FACILITY

The text in the legends of figures 1, 2, and 3 that identifies each hazard ratio and its corresponding confidence interval is now being displayed on the same vertical level to the left of the plotted lines. This removes the need to identify the levels of the factors by different colors (and symbols) making the figure less busy to the eye. Colors can obviously still be used, but the necessity of using colors no longer exists to identify each factor level.

PLOTTING VIA THE ANNOTATE FACILITY ON A LOG SCALE

In figure 2 above, we used a simple option on the AXIS statement to plot on a log scale—thus visually balancing the limits of the confidence interval around the hazard ratio. Even though we are now constructing the plot through the annotation facility, we still may want to plot on a log scale. This is easily done by a different macro that converts to percent of the graphical area but adjusts for a log scale.

Macro CONVLOGX is used to convert the X value to its corresponding percentage since we are annotating based on percentage of the graphical area but on a log scale. Just like %CONVERTX, no semicolon ends the one line of code since this macro is designed to be used as part of a line of DATA step code.

```
%macro convlogx(xval);
   (log(&xval./&xmin.))*(&xlen./(log(&xmax./&xmin.)))+&xstart.

%mend convlogx;
```

In specifying the minimum and maximum values for the X-axis, we need to avoid the value of zero since it is undefined for a log. Figure 5 displays the nice symmetry that we observed above in Figure 2.

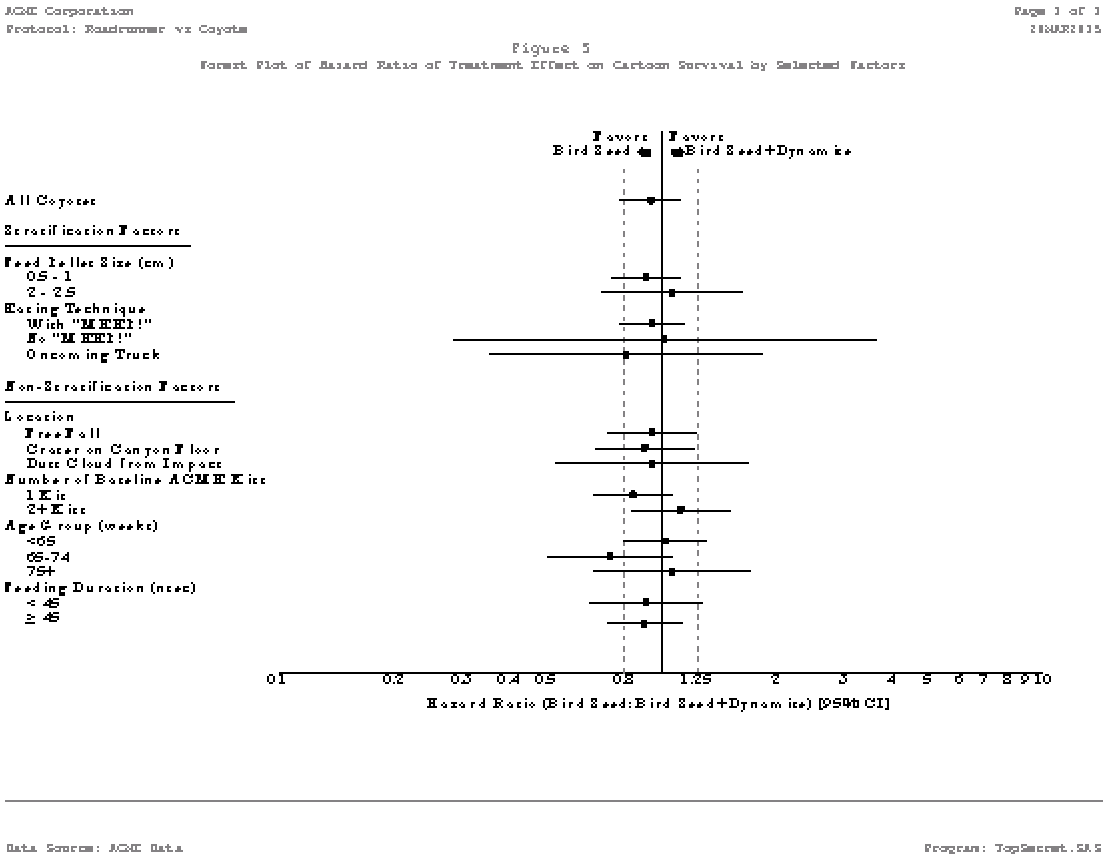


Figure 5. forest plot created by PROC GSLIDE via the ANNOTATE FACILITY on a log scale

PLOTTING VIA THE ANNOTATE FACILITY ON A LOG SCALE AND DISPLAYING STATISTICS

A visual display of the data is nice, but the precision of a displayed number trumps what we think the actual value might be based on just looking at a graphical representation. Although these can be displayed in a matching table, it is more convenient to have them displayed to the right of the plotted lines. This is easily accomplished by using the ANNOTATE FACILITY to add the statistics that match the plotted lines in the figure.

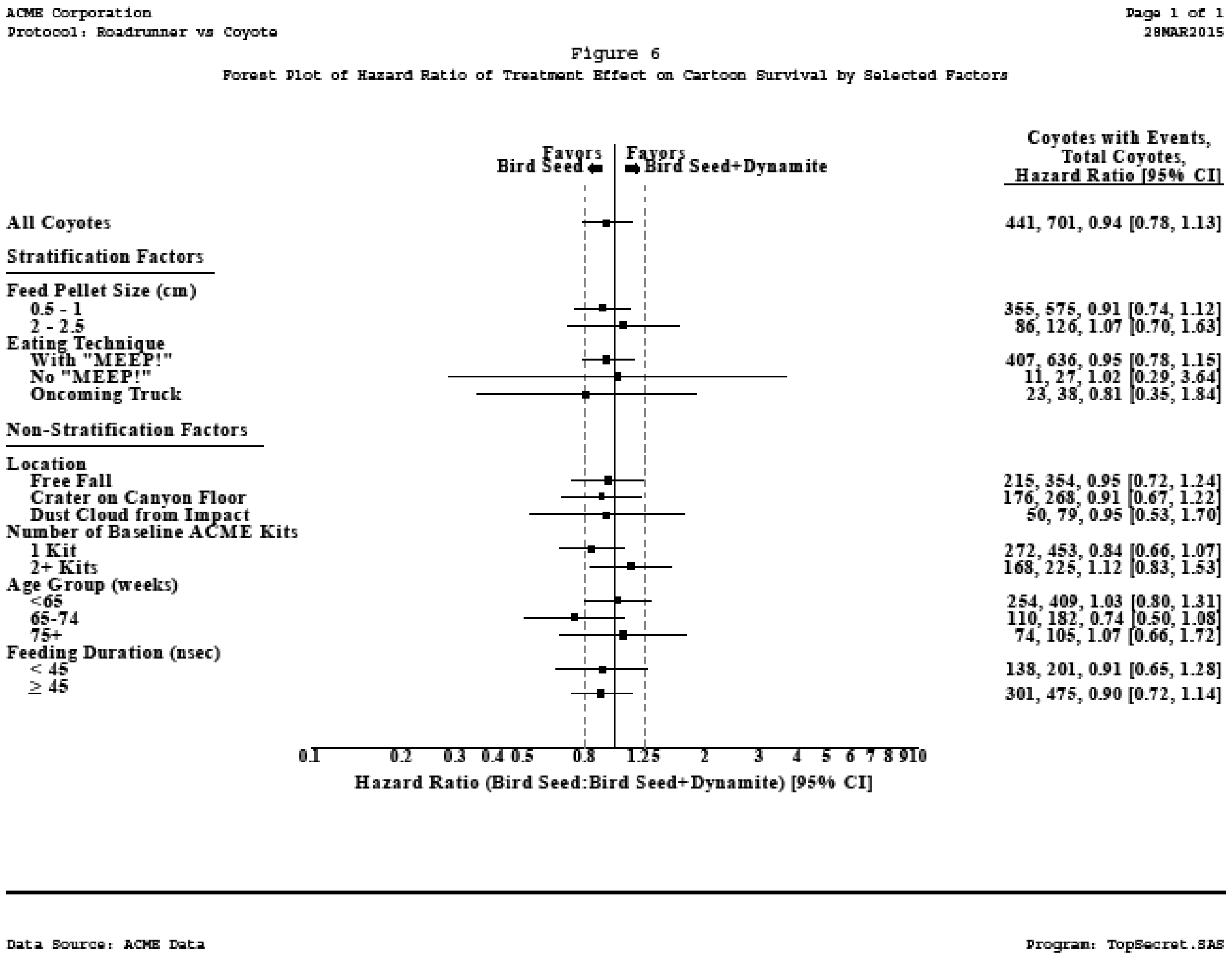
While the data are being created to label the factor levels (specifically the Y-value for each level), we can also create data to plot the statistics. We do this to ensure that the statistics are displayed at the same vertical level as the factor labels. Simply concatenate the events for a factor level, the number of subjects that could have had an event, the hazard ratio, and the confidence interval into a character variable (variable STATS in the example code below). The displaying of these statistics is then rudimentary.

```
   %label(100,yvalue,stats,black,0,0,0.95,,4) ;
```

The value in YVALUE has been calculated as noted above in reference to using the maximum Y value and using the %NEXTROW macro via implicit DATA step looping. We also create the column header with several calls to the %LABEL and an adjustment for the label height.

```
   yvalue=&ymax.+3*&lblrowhgt.;
   %label(99,yvalue,"Coyotes with Events,",black,0,0,0.90,,4);
   yvalue=yvalue-&lblrowhgt.;
   %label(97,yvalue,"Total Coyotes,",black,0,0,0.90,,4);
   yvalue=yvalue-&lblrowhgt.;
   %label(100,yvalue,"Hazard Ratio [95% CI]",black,0,0,0.90,,4);
   yvalue1=yvalue;
   %line(x-length("Hazard Ratio [95% CI]")+3,yvalue1-&underline.,100,yvalue1-
   &underline.,black,1,1);
```

Figure 6 shows the statistics placed in line with the factor level labels and the plotted lines.



ACME Corporation
Protocol: Roadrunner vs Coyote

Page 1 of 1
28MAR2015

Figure 6
Forest Plot of Hazard Ratio of Treatment Effect on Cartoon Survival by Selected Factors

	Coyotes with Events, Total Coyotes, Hazard Ratio [95% CI]
All Coyotes	441, 701, 0.94 [0.78, 1.13]
Stratification Factors	
Feed Pellet Size (cm)	
0.5 - 1	355, 575, 0.91 [0.74, 1.12]
2 - 2.5	86, 126, 1.07 [0.70, 1.63]
Eating Technique	
With "MEEP!"	407, 636, 0.95 [0.78, 1.15]
No "MEEP!"	11, 27, 1.02 [0.29, 3.64]
Oncoming Truck	23, 38, 0.81 [0.35, 1.84]
Non-Stratification Factors	
Location	
Free Fall	215, 354, 0.95 [0.72, 1.24]
Crater on Canyon Floor	176, 268, 0.91 [0.67, 1.22]
Dust Cloud from Impact	50, 79, 0.95 [0.53, 1.70]
Number of Baseline ACME Kits	
1 Kit	272, 453, 0.84 [0.66, 1.07]
2+ Kits	168, 225, 1.12 [0.83, 1.53]
Age Group (weeks)	
<65	254, 409, 1.03 [0.80, 1.31]
65-74	110, 182, 0.74 [0.50, 1.08]
75+	74, 105, 1.07 [0.66, 1.72]
Feeding Duration (nsec)	
< 45	138, 201, 0.91 [0.65, 1.28]
≥ 45	301, 475, 0.90 [0.72, 1.14]

Data Source: ACME Data

Program: TopSecret.SAS

Figure 6. forest plot created by PROC GSLIDE via the ANNOTATE FACILITY on a log scale with statistics

PLOTTING VIA THE ANNOTATE FACILITY ON A LOG SCALE AND DISPLAYING STATISTICS AND IMPLEMENTING TRUNCATION

The actual variation found between the hazard ratios and their confidence intervals can vary in amount and location. We may want to limit the allowed values for one or both sides of the figure depending upon the actual data or the interest of the clinicians. Since the actual statistics are now being displayed, truncation does not really limit the visual effect upon a reviewer's eye to a material degree. However, we want to have a visual cue that truncation occurred; this is accomplished by adding a character at the end of a displayed interval affected by truncation.

By allowing both sides to be truncated independently, a figure can be exactly tailored to show a desired range without trying to influence a reviewer—as both the truncation is always identified and the exact statistics are displayed. This truncation is accomplished by limiting the values that can be plotted, and adjusting the scale to the desired minimum and maximum. Macro parameters CUTLO and CUTHI are used to assign what macro variable XMIN and XMAX contain – which variables are used to convert values to the scale being plotted.

```
%let xmin=&cutlo.;
%let xmax=&cuthi.;
```

Statistics that exceed the either CUTLO or CUTHI are truncated on a by-observation basis (remember, each observation contains the hazard ratio, and lower and upper confidence interval limits).

```
if hruppercl gt &cuthi. then cuthi=1;
if n(hrlowercl) and hrlowercl lt &cutlo. then cutlo=1;
foundcutlo=max(foundcutlo,cutlo);
foundcuthi=max(foundcuthi,cuthi);
```

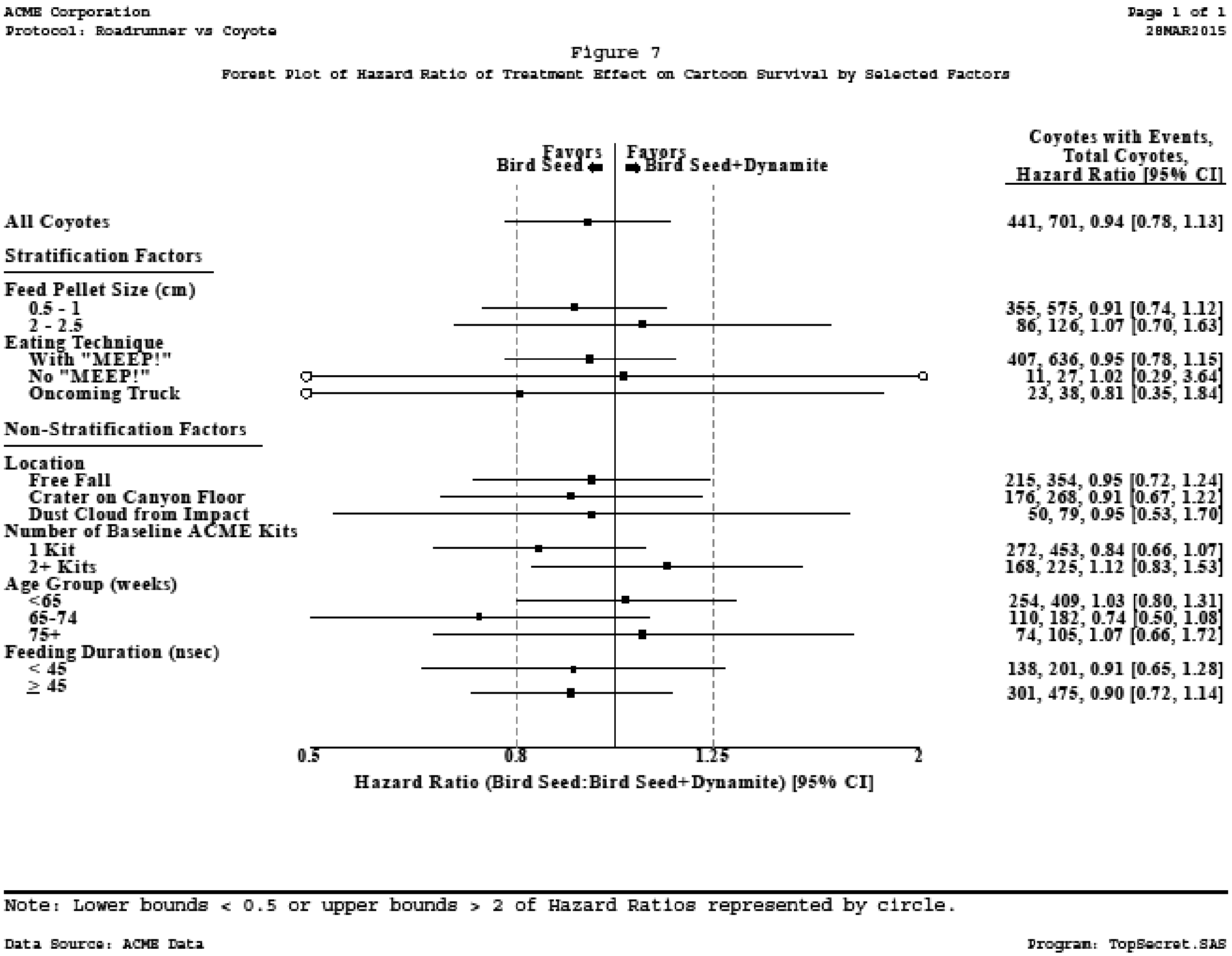
The actual values prior to any truncation are kept for display in the figure as the exact numerical values (on the right side of the figure), and then the truncation is performed to limit them as desired for the plotting the forest plot lines.

```
stats = strip(put(events,3.))||', '||strip(put(totpats,9.))||', '||
   compress(ratioc)||' ['||compress(lowerc)||', '||compress(upperc)||']') ;

*** TRUNCATE VALUE TO PLOT IF OUTSIDE OF SPECIFIED PLOT BOUNDARIES ***;
if cutlo then hrlowercl=&cutlo.;
if cuthi then hruppercl=&cuthi.;
```

When values are truncated, a data-driven footnote is created

```
*** FOOTNOTES DEPEND UPON IF A CONFIDENCE INTERVAL VALUE WAS TRUNCATED OR NOT ***;
%macro foots;
   %if &foundcutlo. eq 1 or &foundcuthi. eq 1 %then %do;
         %if &foundcutlo. eq 1 %then %do;
               %if &foundcuthi. eq 1 %then %do;
                     *** BOTH A LOWER AND AN UPPER VALUE TRUNCATED ***;
                     footnote1 height=10pt justify=left "&topbrdr.Note: Lower
                     bounds < &cutlo. or upper bounds > &cuthi. of Hazard Ratios
                     represented by circle." ;
               %end;
               %else %do;
                     *** A LOWER VALUE TRUNCATED ***;
                     footnote1 height=10pt justify=left "&topbrdr.Note: Lower
                     bounds < &cutlo. of Hazard Ratios represented by circle." ;
               %end;
         %end;
         %else %do;
               *** AN UPPER VALUE TRUNCATED ***;
               footnote1 height=10pt justify=left "&topbrdr.Note: Upper bounds >
               &cuthi. of Hazard Ratios represented by circle." ;
         %end;
   %end;
%mend;

%foots
```

The resulting figure is clean, informative, and compact.



ACME Corporation
Protocol: Roadrunner vs Coyote

Page 1 of 1
28MAR2015

Figure 7
Forest Plot of Hazard Ratio of Treatment Effect on Cartoon Survival by Selected Factors

	Coyotes with Events, Total Coyotes, Hazard Ratio [95% CI]
All Coyotes	441, 701, 0.94 [0.78, 1.13]
Stratification Factors	
Feed Pellet Size (cm)	
0.5 - 1	355, 575, 0.91 [0.74, 1.12]
2 - 2.5	86, 126, 1.07 [0.70, 1.63]
Eating Technique	
With "MEEP!"	407, 636, 0.95 [0.78, 1.15]
No "MEEP!"	11, 27, 1.02 [0.29, 3.64]
Oncoming Truck	23, 38, 0.81 [0.35, 1.84]
Non-Stratification Factors	
Location	
Free Fall	215, 354, 0.95 [0.72, 1.24]
Crater on Canyon Floor	176, 268, 0.91 [0.67, 1.22]
Dust Cloud from Impact	50, 79, 0.95 [0.53, 1.70]
Number of Baseline ACME Kits	
1 Kit	272, 453, 0.84 [0.66, 1.07]
2+ Kits	168, 225, 1.12 [0.83, 1.53]
Age Group (weeks)	
<65	254, 409, 1.03 [0.80, 1.31]
65-74	110, 182, 0.74 [0.50, 1.08]
75+	74, 105, 1.07 [0.66, 1.72]
Feeding Duration (nsec)	
< 45	138, 201, 0.91 [0.65, 1.28]
≥ 45	301, 475, 0.90 [0.72, 1.14]

Note: Lower bounds < 0.5 or upper bounds > 2 of Hazard Ratios represented by circle.

Data Source: ACME Data

Program: TopSecret.SAS

Figure 7. forest plot created by PROC GSLIDE via the ANNOTATE FACILITY on a log scale with statistics and graphical truncation

CONCLUSION

So, are you inclined to stick with PROC GPLOT (the *Old Growth* method)? It is simple, quick, and the easiest to maintain. Using PROC GPLOT with reference lines and potentially plotting on a log scale is the simplest approach. Or do you want to move into the *Genetically Modified Organism* realm with some reservations (just annotating your vertical reference lines) or jump in wholeheartedly by constructing the entire plot including a whole section displaying the statistics as text via the ANNOTATE FACILITY and PROC GSLIDE? Both have their merits. However, once the code is developed for an initial figure via PROC GLIDE and the ANNOTATE FACILITY, creating a macro and having many different presentation options by just varying the call of the macro is non-complex and results in reviewer-pleasing output.

REFERENCES ITE

- National Park Service, U.S. Department of the Interior, Mount Rainier National Park. “Old-Growth Forest.” Available at http://www.nps.gov/mora/planyourvisit/upload/old-growth-forest-sep11.pdf
- University of California at San Diego. “Genetically Modified Organisms (GMO).” Available at http://www.bt.ucsd.edu/gmo.html

CONTACT INFORMATION

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

Name: Scott Horton
Enterprise: Experis Clinical
City, State: Kalamazoo, MI
E-mail: Contact via LinkedIn.com