Paper 167-25

Forcing SAS/GRAPH® Software to Meet My Statistical Needs: A Graphical Presentation of Odds Ratios

Richard M. Mitchell, Westat, Rockville, MD

ABSTRACT

The calculation of odds ratios is a task that SAS® software can provide easily for even the most basic programmer. However, the presentation of this information can become awkward and difficult when a programmer is faced with specific requirements that are not readily addressed in SAS/GRAPH. Specifications including the horizontal display of confidence intervals, a varying number of studies, and symbol size generation that is based on varying study populations, may all add to a challenging task. When first presented with this task, a programmer may be tempted to pursue other, more suitable software packages to graphically present the data. But, with some careful thinking and clever programming, the limitations of SAS/GRAPH can be overcome by utilizing other SAS tools to prepare statistical data for appropriate graphical presentation.

INTRODUCTION

The author was faced with a rather unique task of providing a graphical presentation comparing the odds ratios of multiple international studies for an Individual Patient Data (IPD) meta-analysis [1]. It was necessary to allow flexibility for a varying number of studies and populations, as well as to provide a mechanism for the graph to portray idiosyncrasies between the study data. Through the author's initial research, no SAS examples were identified that would easily help produce a process that would meet all of the client's specifications. Therefore, the author undertook an effort to develop a new process that would effectively present the study data by utilizing a variety of SAS tools in conjunction with SAS/GRAPH. The final graphical presentation of this process is shown later in this paper.

PROCESS SPECIFICATIONS

During the early planning stages of this graphical presentation, the client provided detailed specifications of a desired graph that forced the programmer to create a new approach since compatible options were not directly provided in SAS/GRAPH. In retrospective, the process seems fairly logical and simple, although at the time, the task proved to be time consuming and difficult. Specifications included the horizontal display of confidence intervals, the varying number of studies and populations, and the generation of symbols based on the population size. Each of these individual processes were addressed and subsequently woven together to create a descriptive graph of odds ratio data.

Horizontal display of confidence intervals

SAS/GRAPH conveniently provides the mechanism to display confidence intervals vertically, primarily for high-low type data (e.g. stock quotes). To flip these confidence intervals to a horizontal format, some data step programming was needed prior to accessing SAS/GRAPH.

Varying number of studies and populations

Because the graph was created while the IPD meta-analysis was in progress, flexibility was needed so that changing populations as well as potential subset displays were easily accounted for. Preliminary analyses were performed while data were being received by different studies over the course of several months. Since the decision would not be made until much later on what studies and subjects would be included in the final analysis, flexibility was a high priority in the process specifications. Additionally, the significant amount of time and effort to develop the overall process could be more justified if the graph could be applied easily to other research projects as well.

Symbol generation based on population size

In order to show the wide variety of study contributions to the analysis, the client requested that odds ratio points on the graph be directly related to their population sizes. Although the BUBBLE statement in PROC GPLOT accommodates a basic aspect of this feature, the client desired a more flexible presentation where any type of symbol as well as multiple symbol types could be automatically displayed on the same graph with the confidence interval lines. These different symbol types helped to distinctively describe differences between the studies.

PROCESS COMPONENTS

The automated process that was created to generate a graph of odds ratio data included 6 main components: the definition of odds ratio data and corresponding confidence intervals, the identification of the number of studies and subjects, the production of format codes, the production of additional data points, the generation of varying symbol sizes and shapes, and finally, running the data through PROC GPLOT.

Definition of Odds Ratio Data

To simplify the explanation of the graphical presentation process discussed in this paper, odds ratio data are assumed to already be available in a data set with predefined variable names. Variables contained in this data set would include the study numbers and names, odds ratios, upper and lower confidence intervals, and study sample sizes. A process for obtaining these data through PROC LOGISTIC in SAS/STAT® is presented in "Reporting Results of Multiple Logistic Regression Models Depending on the Availability of Data." [2] Below in Figure 1, are sample data for 13 fictitious studies. The dataset METAODDS includes a record for each study while the dataset TOTAL includes 2 records: 1 that represents the combination of all studies, and 1 that will be used later in the process to provide a space on the graph between the studies and the total.

```
METAODDS:

  X      Y   XMIN     XMAX   SIZE   METANAME

0.759    1   0.103    4.145  2227   Paris
0.731    2   0.400    1.761  4156   Atlanta
0.813    3   0.505    1.346  1352   Bombay
0.254    4   0.105    1.153  1143   Boston
0.000    5   0.000    2.461  1296   * Moscow
0.656    6   0.072    4.827  2455   Miami
0.772    7   0.107    5.495  2194   Ghent
0.000    8   0.000    9.031  1020   * Rome
0.608    9   0.485    0.725  2400   Houston
0.861   10   0.333    1.926   894   Dublin
0.351   11   0.136    0.690  1115   Geneva
0.000   12   0.000    2.987  1595   * Oslo
0.703   13   0.066    4.225  2894   Montreal


TOTAL:

  X      Y   XMIN     XMAX   SIZE   METANAME

 0.750  99   0.600    0.900  2000   ** Total
   .    98     .        .      .
```

Figure 1

Identification of number of studies and subjects

To automatically incorporate study size information on the graph for any given analysis, three macro variables (N_STUD, N_SUBS, MAXSIZE) were created. Assessing how many studies that were to be included in an analysis drives the remainder of the program by identifying the number of lines to be provided on the graph, as well as the number of loops to be executed when creating other key variables. The number of subjects determined the eventual size of plot symbols in the final graph. Since population sizes may vary among projects (e.g. the largest study in Project A may have 1,000 subjects while the largest study in Project B may have 20,000 subjects), SAS code was needed to deal with keeping the symbol sizes proportional. By identifying the maximum size for any group of studies, an algorithm (shown later) could be applied such that the height of symbols could be adjusted accordingly. The programming code is shown in Figure 2 below where an output file is created with PROC MEANS and the data are subsequently transformed into macro variables using CALL SYMPUT.

```
proc means data=metaodds noprint;
  var size;
  output out=meanout sum=sum max=max;

data _null_;
  set meanout end=eof;
  if eof then do;
    call symput("n_stud",trim(left(put(_freq_,8.))));
    call symput("n_subs",trim(left(put(sum,8.))));
    call symput(“maxsize”,trim(left(put(max,8.))));
  end;
```

Figure 2

Produce “Sorted” Format Code

It is a common statistical practice to display odds ratio data in order of decreasing size. To accomplish this task, it was necessary to assign a rank to each study that would be used as the y-axis plot point, while also retaining the original study number so that the appropriate study description could be noted on the graph. By concatenating the rank with the study name, a string of code was produced that would be utilized by PROC FORMAT. For example, Study 1 was perhaps the 5th in order of odds ratios, so its study number was changed to 5. Then a format would be needed to display the actual study name on the graph. In Figure 3 on the following page, the code in the macro variable METASTR might produce something like "1=Dublin 2=Bombay," etc.

```
proc sort data=metaodds; by x xmax; run;

data metadat;
  set total(in=a) metaodds(in=b);
  by x xmax;

  length metastr $ 200;
  retain metastr studcnt;
  metanum=y;
  y=_n_;

if _n_=1 then do;
    metastr="0=' ' %eval(&n_stud+3)=' '";
    studcnt=&n_stud+2;
  end;
  else studcnt=studcnt-1;

  do i=1 to %eval(&n_stud+2);
     if i=studcnt then metastr=" " || trim(left(metastr)) || trim(left(y)) ||
        "='" || trim(left(metaname)) || "' ";
  end;

  if metaname=' ' then y=.;
  yo=xmin; xo=xmax;

  output metadat;

  if _n_=%eval(&n_stud+2) then do;
     call symput("metastr",trim(metastr));
  end;
run;

proc format;
  value stfmt &metastr;
run;
```

Figure 3

Note that the code in Figure 3 adds values to the number of studies such as "&n_stud+2" and "&n_stud+3." These calculations allow for additional rows on the final graphs representing the total, a blank row between the total and the other studies, and finally, a blank row at the top of the graph (a blank row is set at the bottom of the graph by defining row zero as " ").

Produce additional data points

Perhaps the most annoying issue that needed to be addressed in the entire graphical process was the fact that PROC GPLOT draws high-low type lines vertically rather than horizontally. This type of line was needed to present the confidence interval around each study odds ratio, and it was an initial requirement that all studies would be represented on the y-axis. Therefore, plotting data on the x-axis instead was not an option.

One solution to this dilemma was to identify 6 coordinates on the graph for each study that would later be joined together "manually." These points were plotted based on the study number on the y-axis and represented the lower confidence interval (LCI) and the upper confidence interval (UCI) where 0.2 was added and subtracted from each of these to produce the additional points. The resulting coordinates were as follows:

1. LCI (x-axis) and study number (y-axis)
2. LCI (x-axis) and study number + 0.2 (y-axis)
3. LCI (x-axis) and study number – 0.2 (y-axis)
4. UCI (x-axis) and study number (y-axis)
5. UCI (x-axis) and study number + 0.2 (y-axis)
6. UCI (x-axis) and study number – 0.2 (y-axis)

These 6 coordinates were plotted individually, and then connected as shown in Figure 4 below:

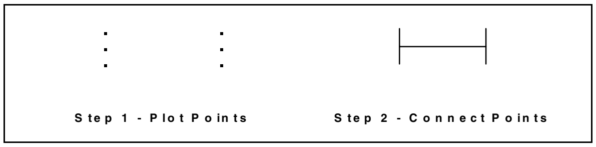


Figure 4

The code that assigned coordinates to each study is shown below in Figure 5:

```
data alldata;
  set metadat;
  %macro doall (study);
    %do %while (&study le (&n_stud+2));
      if y=0.2+&study then do;

        * Apply LCI for next 3 points;
        xx&study=xmin;
        yy&study=y+0.2;  output;  * Point at 0.2 above study #;
        yy&study=y-0.2;  output;  * Point at 0.2 below study #;
        yy&study=y;      output;  * Point at study #;

        * Apply UCI for next 3 points;
        xx&study=xmax;  output;  * Point at study #;
        yy&study=y+0.2;  output;  * Point at 0.2 above study #;
        yy&study=y-0.2;  output;    * Point at 0.2 below study #;

      end;
      %let study=%eval(&study+1);    * Add 1 to study #;
    %end;
  %mend doall;
  %doall (1)
```

Figure 5

Generate Varying Symbol Sizes and Shapes

Before utilizing PROC GPLOT to plot the study data, it was necessary to create an automated process that would uniquely define symbol sizes and shapes to represent the different population sizes, as well as studies with nonconforming data. Based on the number of studies for a given analysis, global macro variables were created that could be called within PROC GPLOT. These variables were used to produce any given number of

SYMBOL statements, each with a height (and width) based on the size of the study, and a shape that would appropriately categorize groups of studies. In the example used for this paper, a filled square symbol was used for most studies, while a filled left arrow was used to represent studies with an odds ratio of 0 and thus an unknown lower confidence interval. One may also use varying shapes to represent distinct groups such as North American studies vs. South American studies (not shown). While this program offers several default symbols, the user may easily modify the code to use other symbol types (e.g. dot, circle, diamond, etc.), as well as line colors that may better suit their analysis and visual needs.

The code in Figure 6 shows how two macros are used to first define a population size and symbol type for each study (&SYMSIZE), then secondly to generate an individual symbol statement for all studies in the analysis (&SYMS). Note that if the lower confidence interval for a study is 0.01 (converted from 0 for use on a log scale), then an arrow symbol is used, while all other studies are assigned a square symbol. The user may expand on this code as necessary depending on the complexity of their analysis data. Also note that all studies are proportionally fit on any given graph by the algorithm, symbol height=(study size*11)/maximum study size in analysis.

```
%macro symsize;
  data _null_;
    set metadat;
    retain sizeh1-sizeh%eval(&n_stud+1)
           fontv1-fontv%eval(&n_stud+1);
    length fontv1-fontv%eval(&n_stud+1) $ 20;
    array sizes sizeh1-sizeh%eval(&n_stud+1);
    array fvs $ fontv1-fontv%eval(&n_stud+1);
    do i=1 to (&n_stud)+1;
      if i=y then do;
        sizes{i}=sizeh*11/&maxsize;
        if xmin=0.01 then fvs{i}='font=marker v=A';
        else fvs{i}='font=specialu v=K';
        if i=&n_stud then output;
      end;
    end;
    %do i=1 %to (&n_stud)+1;
      call symput("sh&i",trim(left(put(sizeh&i,6.2))));
      call symput("fv&i",trim(left(put(fontv&i,20.))));
      %global sh&i fv&i;
    %end;
  run;
%mend symsize;

%symsize

%macro syms;
  %do i=1 %to (&n_stud)+2;
    symbol&i &&fv&i l=1 interpol=none h=&&sh&i color=green;
  %end;
%mend syms;


%syms
```

Figure 6

As a last step in the data preparation process, the X (odds ratio) and Y (study number) values are assigned to new variables names that correspond to their study numbers so that a separate SYMBOL may be applied to each study (e.g. X1, Y1, X2, Y2, etc.). Code for this process is shown in Figure 7 below:

```
data final(drop=i);
  set alldata;
  array xarray{*} x1-x%eval(&n_stud+2);
  array yarray{*} y1-y%eval(&n_stud+2);
  do i=1 to (&n_stud)+2;
    if i=y then do;
      xarray{i}=x;
      yarray{i}=y;
    end;
  end;
run;
```

Figure 7

Running the Data Through PROC GPLOT

By automatically processing the study data to define the number, size, and shapes of symbols, as well as the identification of confidence interval plot points, the shortcomings of the PROC GPLOT options may be overcome to produce the final product. The code that incorporates these macro variables is shown below in Figure 8:

```
axis1  label=(height=2.9
        font=swiss 'Odds Ratio')
       minor=none
       logbase=10 order=(0.01 0.1 1 10)
        value=(height=2.5 font=swiss '0.01' '0.1' '1' '10');
axis2  label=(height=2.9
        font=swiss 'Study')
        minor=none
        order=( 0 to %eval(&n_stud+3) by 1)
        value=(height=2.5 font=swiss);

proc gplot data=final;
  plot
  %macro plotpts;

    * Produce lines to connect 6 points;

    %do i=1 %to (&n_stud + 2);
        yy&i*xx&i=%eval(&n_stud+3)
    %end;

    * Produce symbols to add to each line;

     %do i=1 %to (&n_stud + 2);
        y&i*x&i=&i
    %end;
  %mend;

  %plotpts / overlay haxis=axis1 vaxis=axis2 frame href=1;

  symbol%eval(&n_stud+3) f=marker v=none l=1 w=1 i=join;
  symbol1 font=marker v=P l=1 h=2.7 interpol=none;
```

Figure 8

The first step to perform when using PROC GPLOT is to define the x-axis and the y-axis. Note that data are presented on a log-10 scale to normalize the odds ratio data. The user may easily modify these intervals to fit their own data. As an earlier step, data values of zero were changed to 0.01 to accommodate the log scale (not shown).

Note that nearly the entire coding scheme within the PROC GPLOT section is macro generated. Multiple lines and symbols are requested through PROC GPLOT based on whatever number of macro variables were created earlier in the process. Without looking at the resulting SAS log, it may be difficult to understand just what is actually being plotted and why. From the data provided in this example, the following items are plotted:

- Confidence intervals – For each of the 13 studies and the total summary, 6 coordinates each were plotted and then joined (INTERPOL=JOIN) resulting in a total of 84 points. These points were represented in the code as yy and xx with each study's corresponding number attached to the end (e.g. yy1 and xx1).
- Symbols – For each of the 13 studies and the total summary, 1 square symbol was plotted that represented the odds ratio for that study's data. These points were only used to plot the symbols and were plotted separately (INTERPOL=NONE) from those used to draw the confidence intervals. As noted earlier, a left arrow was used in place of a square when the LCI was unknown.

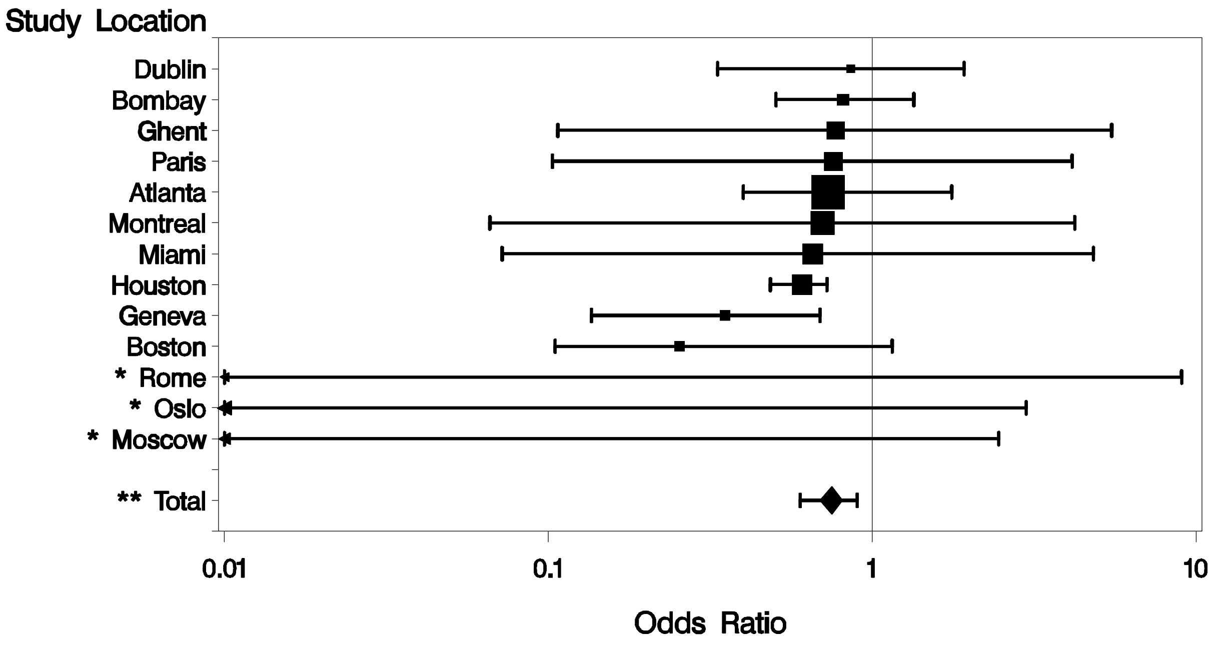
All confidence intervals and symbols were overlaid such that PROC GPLOT provided 98 plots (84 points for confidence interval lines and 14 points for odds ratio symbols) as a single graph. With loops set up throughout the program to generate variables and plot data, the user may produce graphs based on any number of studies and corresponding populations.

Note that macro variables created earlier in the process can be used in the TITLE and FOOTNOTE statements to automatically describe the data in terms of the number of studies and the number of subjects within these studies.

Presenting the Data

By feeding a dataset with the appropriate variables into the program, a graphical presentation of odds ratio data may be automatically produced (shown below in Figure 9) that provides a wealth of information. This graph is not

Odds Ratios and 95% Confidence Intervals for the Association of
Needle Exchange Program Participation with Transmission of HIV Through IV Drug Use



* Lower limit of confidence interval less than 0.01; No HIV transmission through IV drug use among Needle Exchange Program participants
** TOTAL: meta–analytic summary based on the combination of all studies
1. Areas of symbols (individual studies) are proportional to the sample size for the indicated study.
2. Sample size for total model (meta–analytic summaries) is 24741 for 13 Individual Studies
3. The project name, study locations, and odds ratio results in this graph represent fictitious data and are not intended to portray any real studies.

Program Source: sugiodds.sas Run Date: 10JAN00

Figure 9

restricted to any given number of studies and can be used generally for any analysis in which the user desires to display similar odds ratio data.

CONCLUSION

By looking beyond the tools that are available in SAS/GRAPH, users may realize that there are an infinite number of tasks that can be accomplished with SAS software as a whole. The only requirements are that programmers have the necessary problem solving skills and patience to reach the finish line. In the case of the example provided in this paper, the author spent a considerable amount of time developing a process that may perhaps be more easily implemented in another existing software package. However, now that the process has been completed and built to not just accommodate a single task, the author's code can and has been used across multiple projects. The containment of the process in SAS allows for easy flow of data and the flexibility for other programmers to incorporate the process into their work. Finally, the author is still able to honestly say that there's not yet been a task in SAS/GRAPH that he could not perform.

REFERENCES

1. The International Perinatal HIV Group. The Mode of Delivery and the Risk of Vertical Transmission of Human Immunodeficiency Virus Type 1: A Meta-Analysis of 15 Prospective Cohort Studies. The New England Journal of Medicine, 1999, 340: 977-987.

2. Mitchell, R. Reporting Results of Multiple Logistic Regression Models Depending on the Availability of Data. Proceedings of the Twenty-Third Annual SAS Users Group International Conference, 1998, 1227-1231.

ACKNOWLEDGEMENTS

The author wishes to thank Michele Martin for her contributions during the early development stages of this graph.

Corresponding Author:

Richard M. Mitchell
Westat
1650 Research Boulevard, WB 496
Rockville, MD 20850
(301) 251-4386 (voice)
(301) 738-8379 (fax)
MITCHER1@WESTAT.COM