

## Paper 219-27

## Macros for Model Building, Summarizing, and Displaying Results of Stepwise Logistic Regressions

Lori S. Parsons, Ovation Research Group, Seattle, WA

### ABSTRACT

When performing a stepwise logistic regression with PROC LOGISTIC, analysts often want to print only the results of the final model, not the full selection procedure. Beginning with SAS 7.0, the Output Delivery System (ODS) provides users with a method to print a subset of the default output. There is also additional information about the stepwise model that cannot be obtained directly from ODS code, such as the full list of independent variables allowed to enter the model and the number of excluded cases. This information is important to include with the results output. A graphical representation of the model, such as a plot of the Odds Ratios and Confidence Intervals, is also helpful in summarizing the final model. When logistic regression models are built in a hierarchical manner, it becomes important, and far more efficient, to organize the model building code. This paper provides user-written macros, ODS code, and model building code to help SAS users organize, summarize, and display results of stepwise logistic regression models. This paper is written for SAS 7.0 and higher. Stepwise logistic regression models are the focus. However the presented code and model building strategies can be used with any PROC LOGISTIC SELECTION= method.

### BACKGROUND

The default output for stepwise logistic regression includes detailed information for each model building step. Depending on the number of independent variables entering the model, this could result in large amounts of output (and paper, if printed). It is often desirable to save and print only the results of the final model. Beginning with SAS 7.0, the Output Delivery System (ODS) provides a method to select desired data from the default output.

One piece of information in PROC LOGISTIC that cannot be selected directly with ODS table code is the number of cases that were excluded from the analysis due to missing data. This is always an important piece of information to view.

When performing a stepwise logistic regression, the *Summary of Stepwise Selection* lists the independent variables that entered the final model. However, by looking at the results, one cannot determine which independent variables were allowed to enter, but did not enter due to non-significance. It is important to know all the variables that were allowed to enter the model.

It is also very helpful to look at a graph of the odds ratios and confidence intervals of the variables included in the final model. For manuscripts, inclusion of a graph is often required.

When logistic regression models are being built in a hierarchical manner, it becomes important, and far more efficient, to organize the model building code.

### METHODS

To aid analysts, this paper presents a template for organizing model building code. The purpose is to provide a basic setup that can be quickly modified when new analyses are begun. Macro variables are used in the template so that when the template is used for a new study, changes only need to be made in one place.

To aid analysts in summarizing multivariate models, this paper also includes user-written macros that summarize and display results of the models. These include macros to list the full set of independent variables that are allowed to enter the model, calculate and print the number of cases excluded from the model, and plot the odds ratios and confidence intervals for the variables that enter the model.

SAS/STAT® allows users to perform multivariate logistic regression with the LOGISTIC procedure. This paper focuses on the stepwise selection method. However, the presented ODS code, user-written macros, and model building strategies can be used with any PROC LOGISTIC SELECTION= method. The output destination chosen for this paper is the basic listing output, printed to an output file. Any ODS output destinations could be chosen, such as HTML or RTF.

### Template for Model Building

The file **ODSforMLR.txt** contains a template for performing logistic regressions. The purpose of the template is to provide a basis for organizing model building code, macro variables, and included macro code. This template can be easily modified at the start of a new analysis. This file is found in **Appendix 1, Sections 1 – 5** of this paper.

### Template Section 1

This section defines the file locations (subdirectories or folders) for the current analysis. Macro variables are used to define the locations so that a change in location definition needs only to be made in one place.

### Template Section 2

This section uses %INCLUDE statements to insert additional macro code. The included macros used here do not need to be modified when a new analysis is begun. The macros included here are described in greater detail in following sections of this paper.

### Template Section 3

This section contains the macro **MLR**. This macro includes the PROC LOGISTIC code to run a single model. It calls the included macro **PRLRVars**, which prints the list of independent variables that are allowed to enter the model. It selects the logistic regression information to output by using ODS code. It runs the LOGISTIC procedure and then calls the included macro **PRExcluded** that prints the number of cases excluded due to missing data. Finally, it calls the included macro **PlotLR** that produces a graph of the odds ratios and confidence intervals of the final model.

#### **Print Variables Allowed to Enter the Model**

The code described here is found in **Appendix 2**. The file is named **PrintLRVars.txt** and the macro is named **PRLRVars**. This code prints the full list of independent variables that were allowed to enter the model. Both the variable name and the variable label are printed. They are printed in the order that they were allowed to enter the model. PROC MEANS is used as a simple way to create an output dataset that keeps the variables in the desired order. Note that The ODS Output statement does not work with the PROC CONTENTS noprint option. Therefore, the OUT= option is used. Also, the PROC CONTENTS varnum option only affects the printed output, not the order of the variables in the output dataset. Therefore, the sort is required.

#### **Select Initial Model Information**

The Model Information and Response Profile data should always be included in the summary of results. These can be requested with ODS table code. The ODS table definitions are: ModelInfo and ResponseProfile.

#### **Select Final Model Information**

Other results commonly desired are: Summary of Stepwise Selection, Analysis of Maximum Likelihood Estimates, Odd Ratio Estimates, Association of Predicted Probabilities and Observed Responses, Wald Confidence Interval for Adjusted Odds Ratios (when the UNITS option is used), Partition for the Hosmer and Lemeshow Test and the Hosmer and Lemeshow Goodness-of-fit Test results. These can all be requested with ODS code. The ODS table definitions are: ModelBuildingSummary, ParameterEstimates, OddsRatios, Association, CLoddsWald, LackFitPartition, and LackFitChiSq. These output options are included in the presented macro.

#### **Include Additional Model Information**

Any ODS table produced by PROC LOGISTIC can be added to the summary output. A complete list of ODS table names for PROC LOGISTIC can be found in the LOGISTIC procedure chapter of the SAS/STAT Users Guide Version 8.0, in the SAS/STAT online documentation, or in the SAS Explorer window after running a logistic regression model. Additionally, all PROC LOGISTIC ODS table names can be printed to the SAS log by submitting an ODS TRACE ON statement prior to submitting the PROC LOGISTIC code. After

submitting the PROC LOGISTIC code, submit an ODS TRACE OFF statement.

#### **Print Number of Excluded Cases**

It is also very important to look at the number of cases that were excluded from the model due to missing data. In the default output listing, if cases were excluded from the model, SAS prints the following comment "NOTE: X observations were deleted due to missing values for the response or explanatory variables." It is not possible to capture this data directly with ODS table code. However, the number can be calculated with a combination of ODS output code (ODS Output ModelInfo=fModelInfo;) in the Logistic Procedure and some data manipulation in a user-written macro. Included in **Appendix 3** is the file **PrintExcluded.txt** which contains the macro **PRExcluded**. This macro calculates and prints the number of excluded cases.

#### **Graphical look at the results**

Included in **Appendix 4** is the file **PrintFinalLR.txt** which contains the macro **PlotLR**. This macro plots the odds ratios and confidence intervals of the final model. The output chosen here is a .bmp file, but any available device may be chosen. It also may be modified to include plots of all models, not only the final one.

### Template Section 4

This section contains the macro **MMODELS**. This macro defines the output file of results, the independent variables for each model, and the set of models to run. PROC PRINTTO is used to create an output file of all models. The *new* option is used to clear any information that already exists in the defined output file.

Macro variables that build on other macro variables are used to define the independent variables that are allowed to enter the hierarchical models. This is done so that any changes made to lower-level models are likewise made to higher-level models. The programmer does not need to update the change in more than one place, and thus reduces the potential for error.

In the code presented here, four hierarchical models are built. The independent variables are patient characteristics and the dependent variable is hospital mortality. Model 1 allows demographic data to enter the model. The independent variables are defined with the macro variable LR1 in the *%LET LR1 = PTAge Sex White;* statement. Model 2 adds medical history variables to the Model 1 variables. The independent variables are defined with the macro variable LR2 in the *%LET LR2 = &LR1. MHSMOKE MHRENAL MHDIAB MHPREVM MHYPERT MHCHF MHPTCA MHCABG MHSTROKE MHANGINA;* statement. Note that instead of re-listing the variables from model 1, the macro variable from model 1 is used. Similarly, model 3 adds hospital presentation characteristics to Model 2 (and Model 1) variables. The independent variables for model 3 are defined in the *%LET LR3 = &LR2. CPPRESCD ADMITCHF STDEP nswave normal oresult;* statement. Model 4 adds medications within 24 hours to model 3.

There are four call statements to the macro **MLR** described in Section 3; one call statement for each model. The call variables are: **Dateset**, the name of the SAS dataset; **Endpoint**, the name of the dependent variable; **Independ**, the macro name of the independent variables (for example &LR1.); **Incl**, number of variables to force into the model (use 0 if none are to be forced); **Ttitle6**, text for TITLE6; **Ttitle7**, text for TITLE7; and **Where**, code for the WHERE statement in PROC LOGISTIC.

Note that **Dataset**, **endpoint**, **ttitle6**, and **where** are defined in the %MMODELS call statement.

### Template Section 5

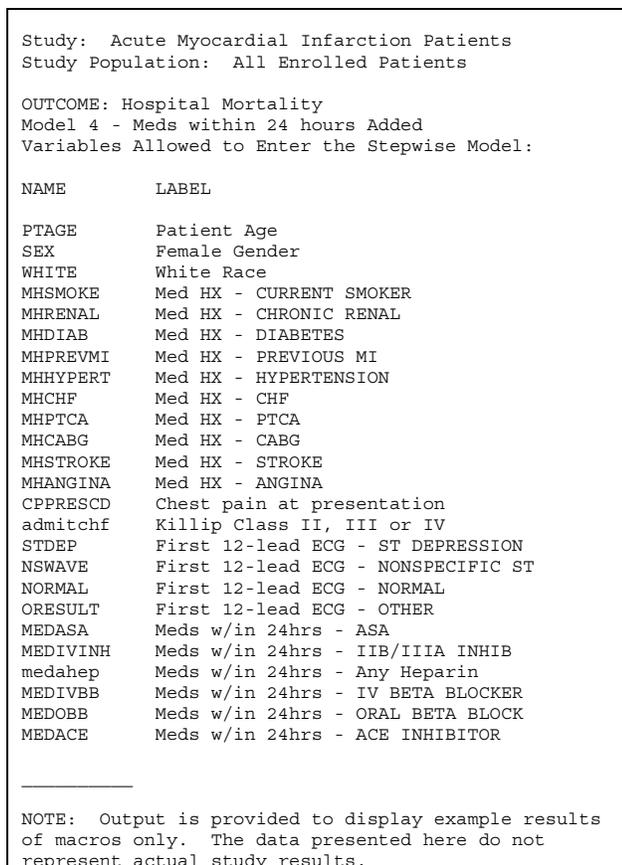
This section is the call statement for the macro **MMODELS** found in section 4.

The call variables are: **Dataset**, name of the SAS dataset; **Endpoint**, the dependent variable; **Ttitle6**, text for Title6; **Where**, code for the WHERE statement in PROC LOGISTIC; and **Outfile**, name of the output file of results.

## RESULTS

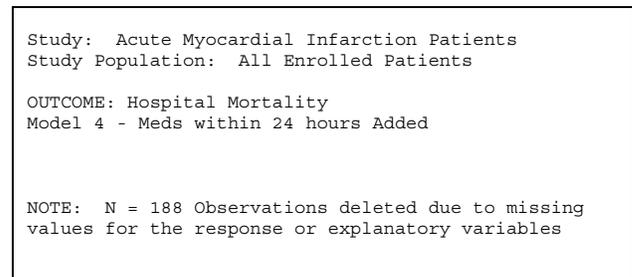
The first page of output will list all the independent variables that were allowed to enter the stepwise model. The variables will be listed in the order that they were allowed to enter the model.

**Figure 1. Variables Allowed to Enter the Model**



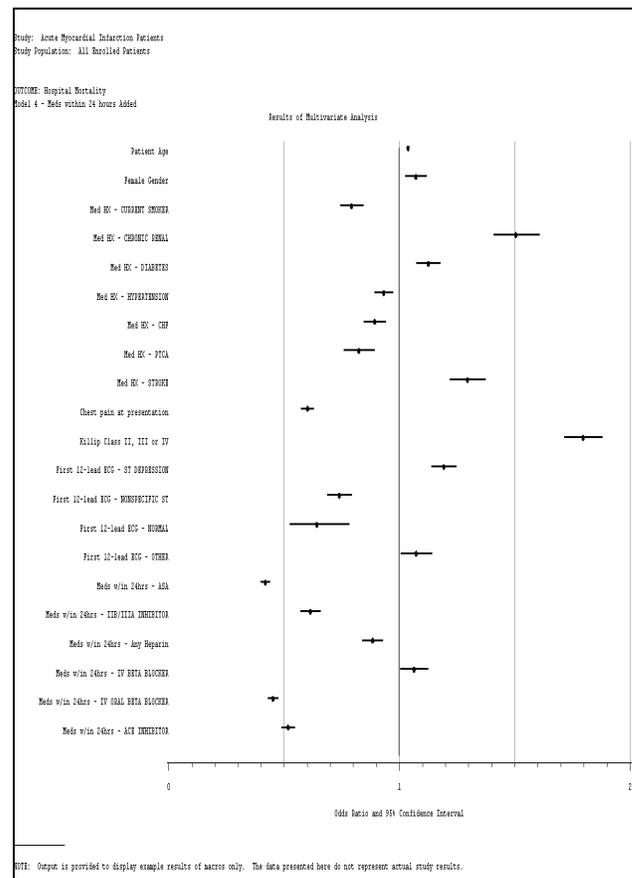
The selected model information will be contained on the next pages of the output. After the model information, the number of cases that were excluded due to missing data will be printed.

**Figure 2. Number of Excluded Cases**



Finally, a graph of the odds ratios and confidence intervals will be produced.

**Figure 3. Odds Ratios and Confidence Intervals**



## CONCLUSIONS

With the model building template and the macro code provided in this paper, an analyst can organize model building code and produce summary output of stepwise logistic regressions.

## REFERENCES

Haworth, Lauren E., *Output Delivery System: The Basics*, Cary, NC: SAS Institute Inc., 2001.

SAS Institute Inc., *The Complete Guide to the SAS® Output Delivery System, Version 8*, Cary, NC: SAS Institute Inc., 1999. 310 pp.

SAS Institute Inc., *SAS/GRAPH® Software: Reference, Version 8*, Cary, NC: SAS Institute Inc., 1999. 1240 pp.

SAS Institute Inc., *SAS/STAT® User's Guide, Version 8*, Cary, NC: SAS Institute Inc., 1999. 3884 pp.

## TRADEMARKS

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.

## CONTACT INFORMATION

Contact the author at:

Lori S. Parsons  
Ovation Research Group  
305 Fir Place  
Edmonds, WA 98020  
Work Phone: (425) 672-8782  
Fax: (425) 672-7282  
Email: lparsons@ovation.org

## APPENDIX 1:

```

/* ***** */
/* File Name: ODSforMLR.txt */
/* Description: */
/* This file contains a template for */
/* model building a Stepwise Logistic */
/* Regression */
/* ***** */
/* ***** */
/* Section 1 */
/* ***** */
/* Define Locations for the Current */
/* Analysis */
/* ~~~~~ */
%let StudyDir = D:\Projects\SASSubroutines\
  ODS_for_MLR;
%let StudyLib = STUDY;
%let OutDir = D:\Projects\SASSubroutines\
  ODS_for_MLR\Output;
/* Define the SAS Format Library */
LIBNAME LIBRARY 'D:\Projects\Nrmi3\Setup\
  NNOV00\SasProgs\Clean\Formats';

/* ***** */
/* Section 2 */
/* ***** */
/* Include macros. These are macros that do */
/* not need to be updated for current study */
/* ~~~~~ */
%INCLUDE "&StudyDir.\SASPrograms\IncludedCode\
  PrintLRVars.txt";
%INCLUDE "&StudyDir.\SASPrograms\IncludedCode\
  PrintExcluded.txt";

```

```

%INCLUDE "&StudyDir.\SASPrograms\IncludedCode\
  PrintFinalLRModel.txt";

/* ***** */
/* Section 3 */
/* ***** */
/* Set up the Logistic Regression Macro */
/* to run a single model */
/* ~~~~~ */
%MACRO MLR(Dataset,endpoint,independ,incl,
  ttitle6, ttitle7,where);
/* Dataset= The SAS dataset name */
/* Endpoint= Dependent variable */
/* Independ= Macro name for the independ vars */
/* Incl= Number of vars to force in model*/
/* ttitle6= Text for Title6 */
/* ttitle7 = Text for Title7 */
/* where= Code for the WHERE option */

/* Define Data Library for current study */
LIBNAME &StudyLib. "&StudyDir.\DataSetX";

/* Choose the desired output options */
Options NoCenter pageno=1 orientation=landscape
  leftmargin=.5 rightmargin=.5 topmargin=.5
  bottommargin=.2 pagesize=61 linesize=145
  font="Courier New" 8;
%let t1 = " ";
%let t2 = "Study: Acute Myocardial Infarction
  Patients";
%let t3 = "Study Population: All Enrolled
  Patients";
%let t4 = " "; %let t5 = " ";
%let t6 = &ttitle6.; %let t7 = &ttitle7.;
%let t8 = " ";
%let f1 = "_____"; %let f2 = " ";
%let f3 = "NOTE: Output is provided to display
  example results of macros only. The data
  presented here do not represent actual study
  results. ";
%let f4 = " ";

Title1 &t1.; Title2 &t2.; Title3 &t3.;
Title4 &t4.; Title5 &t5.; Title6 &t6.;
Title7 &t7.; Title8 &t8.;
Footnote1 &f1.; Footnote2 &f2.;
Footnote3 &f3.; Footnote4 &f4.;

/* Print Vars Allowed to enter model */
%PRLRVars (&StudyLib.&Dataset.&independ.);

/* Select LR Information to output */
ods select ModelInfo ResponseProfile
  ModelBuildingSummary ParameterEstimates
  OddsRatios Association CLoddsWald
  LackFitPartition LackFitChiSq RSquare;

/* NOTE: Output code must come after the select
  code. Capture data in temporary datasets */
ods output ModelInfo=fModelInfo;
ods output RSquare=fRSquare;
ods output OddsRatios=fOddsRatios;

/* Run the Logistic Regression Model */
PROC LOGISTIC DATA=&StudyLib.&Dataset. Descend
  NameLen=40;
MODEL &endpoint. = &independ.
  /SELECTION = STEPWISE INCLUDE = &INCL.
  RISKLIMITS LACKFIT RSQUARE PARMLABEL;
UNITS PTAGE=10 /DEFAULT=1;
WHERE &where.;
RUN;
/* Print number excluded from the model */
%PRExcluded (&StudyLib.&Dataset.);

```

```

/* Graph Odds Ratio and Confidence Intervals */
%LET T8 = "Results of Multivariate Analysis";
%PLOTLR (FinalModel, &t1., &t2., &t3., &t4., &t5., &t6.
, &t7., &t8., &f1., &f2., &f3., &f4);
%let t8 = " ";

/* Turn ODS output "Off" */
ods output close;
/* Delete temporary files */
Proc DataSets Nolist;
Delete fModelInfo fRsquare fOddsRatios;
Run;
%MEND MLR;

/* ***** */
/* Section 4 */
/* ***** */
/* Define the Logistic Regression Model */
/* ~~~~~ */
%MACRO MMODELS
(Dataset, endpoint, ttitle6, where, outfile);
/* Dataset= The SAS dataset name */
/* Endpoint= The dependent variable */
/* ttitle6= Text for Title6 */
/* where= The code for WHERE option */
/* outfile= The name of the output file */

/* Turn proc printto on. */
/* Clear information in file */
proc printto print="&outdir.\&Outfile."
new; run;
ods LISTING ;

/* Define model independent variables */
%LET LR1 = ptage sex white ;
%LET LR2 = &LR1. MHSMOKE MHRENAL MHDIA
MHPREVM I MHHYPERT MHCHF MHP TCA MHCABG
MHSTROKE MHANGINA;
%LET LR3 = &LR2. CPPRESCD ADMITCHF STDEP
nswave normal oresult ;
%LET LR4 = &LR3. MEDASA MEDIVINH MEDAHEP
MEDIVBB MEDOBB MEDACE ;

/* Define the models */
%MLR (&dataset., &endpoint., &LR1., 0, &ttitle6.,
'Model 1-Demographics', &where.);
%MLR (&dataset., &endpoint., &LR2., 0, &ttitle6.,
'Model 2-Medical Histories Added', &where. );
%MLR (&dataset., &endpoint., &LR3., 0, &ttitle6.,
'Model 3-Presentation Characteristics
Added', &where. );
%MLR (&dataset., &endpoint., &LR4., 0, &ttitle6.,
'Model 4-Meds within 24 hours Added', &where.);

/* Turn proc printto off */
proc printto; run;
%MEND MMODELS;

/* ***** */
/* Section 5 */
/* ***** */
/* Call Statement to Run the Models */
/* ~~~~~ */
%MMODELS
(Patients, deceased, 'OUTCOME: Hospital
Death', (denom=1), AllPatients_Deceased);

```

## APPENDIX 2:

```

/* ***** */
/* File Name: PrintLRVars.txt */
/* Description: */
/* (1) Print the full list of variable names */
/* and variable labels that are allowed to */
/* enter the stepwise LR */

```

```

/*(2) Print the variables in the order they */
/* are allowed to enter the model */
/* ***** */
%MACRO PRLRVars (datafile, vars);
TITLE8 'Variables Allowed to Enter the Stepwise
Model:';

/* Get a dataset of the variables that will */
/* be entered into the model. Keep variables */
/* in the order that they will be entered */
/* Note: PROC MEANS is used as a simple way */
/* to create an output dataset with variables */
/* remaining in the desired order */
proc means data=&datafile. (obs=1) noprint;
var &vars.;
output out=PLRTEMP0 (drop=_type_ _freq_
_stat_);
run;

/* Output the variable names and labels */
/* NOTE: */
/* 1) The ODS Output statement does not work */
/* with the PROC CONTENTS noprint option. */
/* Therefore, the OUT= option is used */
/* 2) The PROC CONTENTS varnum option only */
/* affects the printed output, not the order */
/* of the variables in the output dataset. */
/* Therefore, the file needs to be sorted */
proc contents data=PLRTEMP0 out=PLRTEMP1
noprint;
run;
proc sort data = PLRTEMP1 out = PLRTEMP2;
by varnum;
run;
/* Print the model variable names and labels */
proc print data=PLRTEMP2 noobs;
var name label;
run;
TITLE8 ;
%MEND PRLRVars;

```

## APPENDIX 3:

```

/* ***** */
/* File Name: PrintExcluded.txt */
/* Description: */
/* Calculate and print number cases excluded */
/* ***** */
%MACRO PRExcluded (datafile);
/* Get number of cases included in Model */
data ModelN (keep=ModelN);
obsnum=4;
set fmodelinfo point=obsnum ;
ModelN = input(value,7.);
label ModelN = 'Number of cases
included in Model';
output; stop;
run;

/* Get number in the original population */
proc contents data=&datafile. out=OrigN
(keep=noobs) noprint; run;
/* Calculate the difference */
data MissingN;
merge ModelN OrigN (obs=1);
p1 = 'NOTE: N = ';
deleted = Nobs-ModelN;
p2 = 'Observations deleted due to missing
values for the response or explanatory
variables';
run;
/* Print deleted number of cases */
proc print data=MissingN noobs label;
var p1 deleted p2 ;
label p1='00'x;

```

```

        label deleted='00'x;
        label p2='00'x;
    run;
/* Delete temp files
Proc DataSets Nolist;
    Delete ModelN OrigN MissingN;
Run;
%MEND PRExcluded;

```

## APPENDIX 4:

```

/* *****
/* File Name: PrintFinalLR.txt
/* Description:
/* Print the Odds ratios and confidence
/* intervals of the final model
/* *****
%macro
PlotLR(GraphFile,t1,t2,t3,t4,t5,t6,t7,t8,f1,f2,f
3,f4);
/* GraphFile= File containing the graph
/* ~~~~~
/* Merge in the Variable Labels
/* PLRTEMP2 created in file PrintLRVars.txt
/* Sort for merge */
data PLRTEMP2_2 (keep=Effect Label);
    length Effect $8;
    set PLRTEMP2 (rename = (Name=Effect));
run;
proc sort data = plrtemp2_2 out=plrtemp2_3;
    by effect;
run;
/* Create a var to maintain the original order*/
data foddsratios_2;
    set foddsratios;
    order+1;
    retain order;
run;
/* Sort for merge */
proc sort data=foddsratios_2 out=foddsratios_3;
    by effect;
run;
/* Merge */
data fOddsRatios_4;
merge fOddsRatios_3(in=a) PLRTEMP2_3(in=b);
    if a; by effect;
run;
/* Re-sort to get original order */
proc sort data=foddsratios_4 out=foddsratios_5;
    by order;
run;
/* ~~~~~
/* Create a file with graph values
/* Note: CI values are output (gx2) but values
not plotted (do not want a symbol)
/* If you want the CI values to have a symbol,
add gy * gx1 and overlay to the plot statement
in GPLOT
%let tvars = 0;
data graphit (keep = effect gx1 gx2 gy);
    set fOddsRatios_5 nobs = last;
    ty + 1;
/* compute total number of independent variables
to be plotted (maximum on the y-axis)
call symput('tvars',last);
/* compute the maximum upper confidence interval
(maximum on the x-axis)
maxci = max(maxci,UpperCL);
retain maxci;
/* add one more ref line past max upper CI */
call symput('tmaxci',round(maxci) + .5);
Do i = 1 to 3;
    if i = 1 then gx1 = OddsRatioEst;
    else if i = 2 then gx2 = LowerCL;
    else gx2 = UpperCL;

```

```

        gy = last + 1 - ty;
        output;
    end;
    retain ty;
run;
/* ~~~~~
/* Create file with draw stments to connect
/* lower and upper CI points and label vars
/* data annoplot (keep=function x y text position
xsys ysys hsys);
    set fOddsRatios_5 nobs = last;
    length text $40;
    length function $8;
/* Position according to DATA Value coordinate
system */
    retain xsys ysys hsys '2';
/* Reference Line */
    first + 1; retain first;
    if first = 1 then do;
        position = ' '; text = ' ';
        function = 'move';
        x = 1; y = 0; output;
        position = ' '; text = ' ';
        function = 'draw';
        x = 1; y = last; output;
    end;
/* Lines to connect CI and print labels
ty + 1;
gy = last + 1 - ty;
retain ty;
do i = 1 to 3;
    if i = 1 then do;
        position = ' '; text = ' ';
        function = 'move';
        x = LowerCL; y = gy; output;
    end;
    else if i = 2 then do;
        position = ' '; text = ' ';
        function = 'draw';
        x = UpperCL; y = gy; output;
    end;
    else do;
/* Position labels
        position = '4';
        function = 'label';
        x = 0; y = gy; text=Label; output;
    end;
    end;
run;
/* ~~~~~
/* Plot OR and CI
/* This creates a bmp file
filename gout "&OutDir.\&GraphFile..bmp";
goptions reset=all;
goptions htitle=1 htext=1 rotate=landscape
gsfname=gout gsfmode=replace device=bmp;
proc gplot data=graphit;
    plot gy * gx1 / chref=gray href=.5 to
        &tmaxci. by .5 vaxis=axis1
        haxis=axis2 anno=annoplot;
    symbol1 v=dot h=1pct;
    axis1 origin=(45,) order=(0 to &tvars.)
        label=none value=none major=none
        minor=none style=0;
    axis2 order=(0 to &tmaxci.) label=('Odds
        Ratio and 95% Confidence Interval');
    title1 j=left &t1.; title2 j=left &t2.;
    title3 j=left &t3.; title4 j=left &t4.;
    title5 j=left &t5.; title6 j=left &t6.;
    title7 j=left &t7.; title8 &t8.;
    footnotel j=left &f1.; footnote2 j=left &f2.;
    footnote3 j=left &f3.; footnote4 j=left &f4.;
run;
goptions reset=all;
%mend PlotLR;

```