A Program to Compute Odds Ratios and Confidence Intervals from LOGISTIC Output
Eric Yu I Wang, Aili Gong
University of Southern California
Department of Preventive Medicine
and David P. MacKinnon
Arizona State University
Department of Psychology

Abstract:

The odds ratio and its 95% Confidence Interval are widely used in epidemiology and clinical trial research. The purpose of this paper is to describe a program to automatically compute these statistics with the SASjSTAT LOGISTIC procedure and the SAS MACRO product. The program can be used for both binary and ordinal response measures with one or many independent variables. The use of the program is demonstrated in two examples from prevention research.

Introduction:

The odds ratio was originally proposed by Cornfield (1951) as measure of the degree of association between an antecedent factor and an outcome. It widely used in clinical trial and epidemiology research particularly because of its simple interpretation as the odds of the outcome occurring given the antecedent factor. Despite the importance of odds ratio, SASjSTAT procedures do not provide the odds ratio in the output of the CATMOD, LOGIST, or LOGISTIC regression programs. The simple 2 by 2 odds ratio is presented when the MEASURES option is requested in the PROC FREQ procedure. The purpose of this paper is to describe the use of the PROC LOGISTIC procedure and the SAS MACRO language to compute and display the odds ratio and its 95% confidence interval, standardized parameter estimates, along with the parameter estimates and standard errors generated from PROC LOGISTIC. The standardized parameter estimates, odds ratio and its confidence limits should make it easier to interpret the results of PROC LOGISTIC. The standardized parameter estimates are used to judge the effects of the predictors on a uniform scale. And in the case of large models with many predictors, hand computation of these measures is cumbersome. This program greatly reduces the time to compute and double check these values.

Statistical Background:

The logistic density function is defined as follows:

\[ \pi(x) = \frac{\exp(\beta_0 + \beta_1 x)}{1 + \exp(\beta_0 + \beta_1 x)} \]

where \( x \) is the covariate vector, and \( \beta_1 \) is the parameter of estimates. The logit transformation of \( x \) yields the LOGISTIC regression equation as follows:

\[ g(x) = \ln \left( \frac{\pi(x)}{1 - \pi(x)} \right) = \beta_0 + \beta_1 x \]

The odds ratio and its 95% confidence interval is usually the parameter of interest in a LOGISTIC regression due to its relatively straightforward interpretation. It is written as the following:

odds ratio of \( \beta = \exp (\beta_1) \)

confidence interval = \( \exp \{ \beta_1 + Z_{0.05}*se(\beta_1) \} \)

The variance of the Logistic distribution is approximately \( \pi^2/3 \). The standardized estimate for the slope parameter is computed by dividing the slope parameter estimate by the ratio of the standard deviation of the underlying distribution, (inverse of the link function) to sample standard deviation of the explanatory variable. (SAS Technical Report P-200 SASjSTAT Software: Callis and Logistic Procedures, 1990, Page 201) For example, if we have age as independent variable in the model the standardized estimate of age = \( \beta_{age} \cdot \text{std(age)} / \sqrt{\pi^2/3} \)

We use,

```
PROC means mean std n noprint;
var %set var;
output out=std out (keep=&keep)
  std=%set var;
PROC transpose data = std out
prefix=sd
out=s std(keep=NAME sd1);
```
to get standard deviation of the independent variable

We use:

```
PROC LOGIST data=log_data covout
outest=new;
model &dev = %5et var;
to get parameter estimate and estimated covariance matrix to calculate standard deviation of the intercept and parameter estimate.
```
Macro Logit description and use:

Attached is a listing of the SAS MACRO logit program to compute odds ratios, confidence limits and standardized parameter estimates from the output of the LOGISTIC program. First, logistic regression parameter estimates are computed in the LOGISTIC program and stored in an output file. Second the LOGISTIC output data set then serves as input to the section of the program that computes the odds ratio, confidence limits, and standardized parameter estimates. Finally, the odds ratio, confidence limits, and standardized parameter estimates are printed along with the usual PROLOGISTIC output. The program is written for use with PC SAS.

The program requires the user to input:

1) dataset name
2) dependent variable (binary or ordinal)
3) sets of independent variables, now there is a limit of 10 independent variables but it is relatively easy to expand to more independent variables.
4) index k is the number of independent variables.
5) since PROC LOGISTIC output is different for binary and ordinal response measures. The binary only case has an intercept, and ordinal case has j-1 intercepts (j is the response level). The macro variable ordinal = 1 indicates that the dependent variables is ordinal, and ordinal = 0 indicates that the dependent variable is binary. The program could be easily extended for the LOGISTIC regression output in CATMOD or LOGIST. The most important difference is the manner in which output files are written and organized in these two programs.

Examples:

Two examples are used to illustrate the logit program. The first example comes from a logistic regression in the study of the predictors of drug use and the second example is from an ordinal logistic regression in the study of the impact of the alcohol warning label.

Drug use. Cigarette use in the last month is the dependent measure in the first example, with predictors of grade, socioeconomic status, and monthly cigarette use one year before. The data is from a large community drug prevention program (Pentz et al., 1989). A person who reported smoking in the last month on year earlier was 9.7 (UCL=12.9, LCL=7.2) times more likely to be a smoker a year later. The standardized estimate for prior smoking status was substantial (.38). Students in the ninth grade were 2.2 (UCL=3.3, LCL=1.5) times more likely to be smokers than eighth graders.

Beliefs about Fetal Alcohol Syndrome. The second example comes from a study of the impact of the new alcohol warning label (MacKinnon, Pentz, and Stacy, 1990). Ordinal logistic regression was used to determine whether gender, alcohol use, race, party attendance, beliefs about the positive consequences about alcohol and the number of friends who use alcohol predict whether the respondent believes that alcohol can harm an unborn baby if the mother consumes alcohol while she is pregnant. The dependent measure was a four point scale ranging from yes definitely, probably, I don’t think so, and no. The odds in this case reflect the average odds across the 4-1=3 logits corresponding to the j-1 transitions across the scale of the dependent measure. Females were 3 (UCL=4.1, LCL=2.2) times more likely to believe that alcohol can harm an unborn baby. Non-whites were .8 (UCL=.9, LCL=.7) as likely as Whites. Non-whites and males were less likely to believe that maternal alcohol consumption can harm an unborn baby.

Summary:

The macro logit program generates the usual LOGISTIC output along with odds ratios, 95% confidence intervals, and standardized estimates. The program is very useful in our data analysis because it automatically computes these statistics which often make more practical sense that the parameter estimate alone. The program is especially useful when there are many predictor variables as shown in the example of the predictors of beliefs about drinking while pregnant.

PROGRAM LISTING:

****** 1 *****;
%MACRO KEEPVAR;
%LETKEEP= ;
%DO N=1 %TO&I;
%LETKEEP = &KEEP %UPCASE(&&V&N) ;
%END;
%MEND KEEPVAR;

****** 2 *****;
%MACRO SET_VAR;
%DO N=1 %TO&I;
%UPCASE(&&V&N)
%END;
%MEND SET_VAR;

****** 3 ************;
%MACRO SET_DATA(DATA,1=);
%DO N=1 %TO &I;
%DATA&N
%END;
%MEND SET_DATA;
%MACRO STD(VAR=J);
%IF &VAR NE %THEN %DO;
%DO N=-1 %TO &J;
DATA STD DATA(KEEP= NAME VARIANCE);
SET COV;
IF NAME = %UPCASE("&VAR&N")
VARIANCE= &VAR&N;
%END;
%END;
%ELSE %DO;
%DO N=-1 %TO &J;
DATA INC P&N(KEEP= NAME VARIANCE);
SET COV;
IF NAME = %UPCASE("INTERCP&N")
VARIANCE= INTERCP&N;
%END;
%END;
%ELSE%DO;
%DON=-1 %TO &J;
DATA CONSTANT;
SET COY;
NAME=SUBSTR(NAME_1,7);
IF NAME='INTERCP';
%GLOBALM;
DATA NULL;
IF 0 THEN SET CONSTANT POINT= NOBS=COUNT;
CALL SYMPUT('M',LEFT(PUT(COUNT,2.)));
STOP;
%STD(VAR=J= &M)
RUN;
%END;
%MENDSTD;

****** 5 ******;
%MACRO INTCEP;
%IF %EVAL(&ORDINAL) %THEN %DO;
DATA CONSTANT;
SET COY;
NAME=SUBSTR(NAME_1,7);
IF NAME='INTERCP';
%GLOBALM;
DATA NULL;
IF 0 THEN SET CONSTANT POINT= NOBS=COUNT;
CALL SYMPUT('M',LEFT(PUT(COUNT,2.)));
STOP;
%STD(VAR=J= &M)
RUN;
%END;
%ELSE %DO;
%DO N=-1 %TO &J;
DATA INTERCEPT;
SET %SET DATA(DATA=INCP, i=&M);
%END;
%MEND INTCEP;

*************************************************************.
%GLOBALM;
%LET 1=10;
%KEEPVAR
DELETE MISSING VALUE
****************************************************************************.*
DATA LOG_DATA(KEEP= %UPCASE(&DEV) &KEEP);
SET &DATA ;
%DO N=-1 %TO &K;
IF &V&N = . THEN DELETE;
%END;
%END;
%MEND INTECP;

***********************************************

** Where:
* data = SAS data set you want analyzed,
* dev = dependent variable (ordinal or binary),
* v1, ..., V10 = independent variable,
* k = # of independent var had value,
* ordinal = 1 == > dependent variable is ordinal,
* ordinal = 0 == > dependent variable is binary.
* *
* programmer: Eric Yu I Wang, Aili Gong, DAVID P. MACKINNON
* Institute for Health Promotion and Disease Prevention Research
* University of Southern California
*
******************************************************************************

***** KEEP THOSE VARIABLES IN THE LOGIST MODEL
******************************************************************************.

%GLOBALM;
%LET I=10;
%KEEPVAR
DELETE MISSING VALUE
****************************************************************************.*
DATA LOG_DATA(KEEP= %UPCASE(&DEV) &KEEP);
SET &DATA ;
%DO N=-1 %TO &K;
IF &V&N = . THEN DELETE;
%END;
%END;
%MEND INTECP;

***************************************************************

* use SAS/STAT logist procedure to generate odds ratio and
* 95% confidence interval
*
MODEL &DEV = %SET_VAR;

CREATE PARAMETER ESTIMATE DATA SET

DATA ES;
SET NEW;
IF NAME = 'ESTIMATE';
PROC TRANSPOSE DATA = ES
OUT = TRANS(KEEP = NAME_ESTIMATE);
DATA P_ESTIMA;
SET TRANS;
J+1;

CREATE STANDARD ERROR OF INTERCEPT
AND
INDEPENDENT VARIABLES

DATA COV;
SET NEW;
IF _TYPE_ = 'COV';
%INTCEP
%STD(VAR=V,J=&K)

DATA COMBINE;
SET INTERCEPT
%SET_DATA(DATA=STD_DA,I=&K); J+1;
PROC SORT DATA = COMBINE; BY NAME;
PROC SORT DATA = S STD ; BY NAME;
DATA STDMERGE; MERGE COMBINE S
BY
_NAME_;PROC SORT DATA = STDMERGE; PROC SORT DATA = P_ESTIMA;
BYj;

CALCULATE:
* 1) STANDARD ERROR
* 2) CHI-SQUARE
* 3) P-VALUE
* 4) STANDARDIZED ESTIMATE
* 5) ODDS RATIO
* 6) 95% CONFIDENCE INTERVAL

DATA BETA;
MERGE P_ESTIMA STDMERGE ; BY j;

SD_ERROR= SQRT(VARIANCE);
T VALUE= ESTIMATE/SD_ERROR;
CHI_SQRE= (T_VALUE)**2;
P _VALUE= (1-PROBNORM(ABS(T _VALUE)))**2;
STANDEST= ESTIMATE*SD1 /(SQRT(3.14159265
*2)/3));
ESTIMATE= -ESTIMATE;

ODDS= EXP(ESTIMATE);
UP_LIMIT= EXP(ESTIMATE + 1.96*SD_ERROR);
LO_LIMIT= EXP(ESTIMATE - 1.96*SD_ERROR);

STANDARDIZE OUTPUT FORMAT

PROC PRINT DATA = BETA SPLT = '*' NOOBS;
VAR NAME_ESTIMATE SD_ERROR CHI_SQRE
P_VALUE STANDEST ODDS UP LIMIT LO LIMIT;
label
NAME = 'Variable'
ESTIMATE = 'Parameter* Estimate'
SD_ERROR = 'Standard* Error'
CHI S QRE = 'Wald* Chi-Square'
P _VALUE = 'Pr > * Chi-Square'
STANDEST = 'Standardized*Estimate'
ODDS = 'Odds*Ratio'
UP LIMIT = '95% *Ucl'
LO LIMIT = '95% *Lcl';
title1 'Analysis of Maximum Likelihood Estimates';
title2 'Odds Ratio and 95% Confidence Interval,'
title3 'Response Variable: &dev';
title4 'Independent Variable: %SET_VAR';
title5 'RUN';
title1; %MEND LOGIT;

References