Paper 256

Poisson Regression Adjustment of Event Rates And Its Macro Procedure ADJ_POIS

Dingyi Zhao, Department of Biostatistics,
UNC-CH, Chapel Hill, NC

Abstract

This paper aims to introduce an adjustment method of event rates and its macro procedure. **SAS ® PROC GENMOD** with the Poisson distribution option can calculate the predicted adjusted rate at the mean levels of the independent variables. Due to nonlinear nature of the model, F(E(X)) is not equal to E(F(X)). A corrected adjusted rate is calculated by multiplying a constant. Macro **ADJ_POIS** can calculate the predicted and corrected adjusted rates and their upper and lower 95% confidence intervals, can display the analysis results and can provide output datasets with adjusted statistics and CIs and with beta estimates, standards errors and P-values.

Method

Assume that the number of incident events **C** for various subgroups is a Poisson random variable, and **c** is the expected number of incident events. **T** is the amount follow-up time (in days).

Let **X** = (1 x1 x2 ......xi......xp)‘;
Betas = (b0 b1 b2 ...... bi......bp);

where xj(j=1......p) are independent variables. bo=intercept, bj(j=1......p) are unknown parameters.

To find the incident rate for xi(0,1) after controlling for x1 to xp but not xi, the Poisson model is

Log **C** = **Betas*X** + Log(**T**).

where **T** = follow-up time for the subgroup defined by independent variables. Log(**T**) has its beta fixed at 1, so is identified in **GENMOD** as “**OFFSET**”.

Let **B_est** = (be0 be1 ...... bei......bep)` are estimate parameters of Betas, and
MX = (1 mx1 mx2 ...... 1 ...... mxp,
1 mx1 mx2 ...... 0 ...... mxp);
where mxk is the mean of xk (k=1......p, but not i). The predicted adjusted rate (at the mean levels of the independent variables) is

ADJVALUE(xi=1,xi=0) = EXP(**B_est*MX**),

its 95% CI is

LOW95% CI = EXP(MX***B_est**-1.96*
SQRT(**VERDIAG**(**MX*COV*MX‘**)));
UP95% CI = EXP(MX***B_est**+1.96*
SQRT(**VERDIAG**(**MX*COV*MX‘**)));
where **COV** is variance-covariance matrix and **VERDIAG** returns a column vector containing the diagonal elements of the matrix **MX*COV*MX‘**.

Due to nonlinear nature of the model, **F(E(X))** is not equal to **E(F(X)**). A corrected adjusted rate per 1000 person-year is calculated by multiplying by constant K, 1000, and 365.25;

ADJUSTED = K*1000*365.25* **ADJVALUE**.

The corrected 95% CI is

CLOW95% CI = K*LOW95%CI;
CUP95% CI = K*UP95%CI;
where **K** = actual count of events overall/sum of adjvalue*T over all subgroups.

Macro Procedure

The macro calling statement is:

%MACRO ADJ_POIS(INFILE=,BYVAR=, EVENTVAR=, FUTIME=, RISK_VAR=, CONTVARS=, GIVENVAR= , GIVENVAL=, PERSON=, EVENTIND=, PRINTOUT=, OUT=,ESTOUT=);

There are 13 parameters described below:

INFILE = Name of input dataset
BYVAR = A list of by variables separated by space
EVENT=An event variable,must be a positive integer
FUTIME = Follow-up time in days
RISK_VAR= A list of dichotomous risk variables separated by spaces
CONTVARS= A list of control variables separated by space
GIVENVAR= A control variable where a reference value will be specified
GIVENVAL= A reference value for the variable in the GIVENVAR parameter. This parameter is required if the GIVENVAR parameter is specified.
PERSON=Person-year unit, 1 or larger number
EVENTIND= Event value to indicate event,default=1
PRINTOUT= YES print all results, otherwise do not specify
OUT= Name of output dataset with adjusted statistics
ESTOUT=Name of output dataset with beta, SE and P-values.

The macro provides the following results:

(1) Sample size
(2) Event number
(3) Both predicated and corrected adjusted event rates and their upper and lower 95% confidence intervals
(4) Output dataset of betas, SES and P-values
(5) Output dataset of adjusted statistics.

Examples

The dataset called as ANAL includes 9 variables:
(1) Event variable=ischemic stroke(ISC)
(2) Age
(3) Center-race= 4 indicator variables FB,FW,MW,WW
(4) By variable = Gender
(5) Risk variable = IMT(0,1)
(6) FUTIME = Futime.

We are interested in finding Age-, field center and race- adjusted ischemic stroke incident rates (per 1000 person-years) with 95% confidence intervals for levels of IMT by gender.

```
%ADJ_POIS(
INFILE=ANAL,
BYVAR=Gender,
EVENTVAR=ISC,
FUTIME=Futime,
RISK_VAR=IMT,
CONTVARS=FB FW MW WW,
GIVENVAR=Age,
GIVENVAL=55,
PERSON=1000,
EVENTIND=1,
PRINTOUT=YES,
OUT=OUTA,
ESTOUT=ESTA);
```

The final output dataset OUTA looks like below:

Gender	IMT	Size	Eventnum	Adjusted	LowCI	UpCi
F	0	7518	75	1.3264	0.9920	1.7736
F	1	347	15	4.3620	2.4767	7.6823
M	0	5691	85	2.3796	1.8722	3.0246
M	1	658	24	4.6950	2.9865	7.3809

The final output dataset ESTA is:

Variables	Beta	SE	Z	P-value	Gender
INT	-17.2893	1.1335	-15.2528	0.0000	F
AGE	0.0967	0.0200	4.8444	0.0000	F
FB	0.5665	0.2759	1.5072	0.1318	F
FW	-1.6568	0.4159	-3.9833	0.0001	F
MW	-1.0207	0.3261	-3.1300	0.0018	F
WW	-0.3903	0.2585	-1.5101	0.1310	F
IMT	1.1904	0.2875	4.1403	0.0000	F
INT	-16.4566	1.0490	-15.6873	0.0000	M
AGE	0.0898	0.0184	4.8929	0.0000	M
FB	-0.1879	0.5307	-0.3541	0.7232	M
FW	-0.8831	0.2855	-3.0934	0.0020	M
MW	-0.6259	0.2545	-2.4590	0.0139	M
WW	-0.9250	0.2744	-3.3712	0.0008	M
IMT	0.6795	0.2381	2.8536	0.0043	M

Conclusion

Our goal in writing this paper was to introduce an Poisson regression adjustment method for event rates and to use macro to calculate adjustment rates and to generate table. Although Poisson regression adjustment can be used to analysis many categorical data, the independent variables are chosen not to generate too many categories.

Acknowledgments

The author would like to thank Nancy Anderson for providing input and reviewing drafts of this paper.

Reference

Chiang CL: Standard error of the age-adjusted death rate. Vital Stat Special Rep 47: 275-285,1961

Timothy C. Wilcosky, Lloyd E. Chambless:A comparison of direct adjustment and regression adjustment of epidemiologic measures. J Chron Dis Vol. 38. No. 10. pp. 849-856,1985

Marshall M. Joffe: Standardized estimates from categorical regression models. Statistics in medicine, Vol, 14, 2131-2141, 1995

SAS Institute Inc.1996, SAS/STAT software change and enhancements

SAS Institute Inc. SAS guide to macro processing

Author Contact

Dingyi Zhao (UCCDYZ.CSCC@MHS.UNC.EDU)
Collaborative Studies Coordinating Center
Dept. Of Biostatistics,
University of North Carolina-Chapel Hill
137 E. Franklin Street, Suite 203,
Chapel Hill, NC 27514

Appendix

```
%MACRO ADJ_POIS(INFILE=,BYVAR=,EVENT=,FUTIME=,
RISK_VAR=,CONTVARS=,GIVENVAR=,GIVENVAL=,
PERSON=, EVENTIND=,PRINTOUT=,OUT=,ESTOUT=);

*-------------------------------------------------------------------------------*
|  COUNT SUM OF PERSON-YEARS OF FOLLOW-UP TIME
| AND COUNT NUMBER OF EVENTS. BY BYVAR , RISK
| VARS. AND ADJUSTED VARIABLES.
*-------------------------------------------------------------------------------*;

PROC SORT DATA = &INFILE OUT=ANALFILE;
BY &BYVAR &RISK_VAR &CONTVARS &GIVENVAR;
RUN;
*-------------------------------------------------------------------------------*
|  COUNT SUM OF PERSON-YEARS OF FOLLOW-UP TIME
|  BY BYVAR ,  RISK VARS. AND ADJUSTED VRIABLES.
*-------------------------------------------------------------------------------*;

PROC MEANS DATA = ANALFILE   NOPRINT;
 BY &BYVAR &RISK_VAR &CONTVARS &GIVENVAR;
 VAR &FUTIME;
 OUTPUT OUT=TOTAL SUM = N;
RUN;

*-------------------------------------------------------------------------------*
|  TRANSLATE A STRING 'X1 X2 X3' TO X1*X2*X3.
*-------------------------------------------------------------------------------*;

%MACRO TRANS(XX=,TT=);
 %IF %LENGTH(&XX)>0 %THEN  %DO;
 %GLOBAL &TT;
 CALL SYMPUT("&TT",TRANSLATE("&XX","*",' '));
 %END;
%MEND;

DATA _NULL_;
   %TRANS(XX=&RISK_VAR,TT=RISKMVAR);
    %TRANS(XX=&CONTVARS,TT=CONTMVAR);
    %TRANS(XX=&BYVAR,TT=BYSORT);
RUN;

%MACRO TRANS2(XX2,TT2);
 %IF %LENGTH(&XX2)>0 %THEN %DO;
 %GLOBAL &TT2;
 CALL SYMPUT("&TT2",TRANSLATE("&XX2",',',' '));
 %END;
%MEND;

DATA _NULL_;
   %TRANS2(&RISK_VAR,RISKSK);
   %TRANS2(&CONTVARS,CONTSK);
RUN;

%GLOBAL CONT_NUM RISK_NUM VAR_NUM;

%MACRO COUNT(COUNTN,COUNTVAR);
 %GLOBAL &COUNTN._NUM;
 %IF %LENGTH(&COUNTVAR)>0 %THEN %DO;
   %LET COUNT1=1;
   %DO %UNTIL(%SCAN(&COUNTVAR,&COUNT1)=);
     %LET COUNT1=%EVAL(&COUNT1+1);
   %END;
   %LET &COUNTN._NUM = %EVAL(&COUNT1-1);
 %END;
 %ELSE %DO;
   %LET &COUNTN._NUM = 0;
 %END;
 %PUT &&&COUNTN._NUM;
%MEND;

%COUNT(BY,&BYVAR);
%PUT NUMBER OF BY VARIABLE IS &BY_NUM;
%COUNT(CONT,&CONTVARS);
%PUT NUMBER OF ADJUSTE VARIABLE IS &CONT_NUM;
%COUNT(RISK,&RISK_VAR);
%PUT NUMBER OF RISK VARIABLE IS &RISK_NUM;

%IF &GIVENVAR NE %THEN %DO;
 %LET VAR_NUM =%EVAL(&CONT_NUM+&RISK_NUM+2);
%END;
%ELSE %DO;
%LET VAR_NUM = %EVAL(&CONT_NUM+&RISK_NUM+1);
%END;

%PUT VAR_NUM = &VAR_NUM;
%IF &BYVAR NE %THEN %DO;
%IF &GIVENVAR NE & &RISK_VAR NE & &CONTVARS NE
%THEN %DO; %LET TABS =
&BYVAR*&RISKMVAR*&CONTMVAR*&GIVENVAR;
%END;
%ELSE %IF &GIVENVAR NE & &RISK_VAR NE &
&CONTVARS = %THEN %DO;
%LET TABS = &BYVAR*&RISKMVAR*&GIVENVAR;
%END;
%ELSE %IF &GIVENVAR NE & &RISK_VAR = &
&CONTVARS NE %THEN %DO;
%LET TABS = &BYVAR*&CONTMVAR*&GIVENVAR;
%END;
%ELSE %IF &GIVENVAR = & &RISK_VAR NE &
&CONTVARS NE %THEN %DO;
%LET TABS = &BYVAR*&RISKMVAR*&CONTMVAR;
%END;
%ELSE %IF &GIVENVAR NE & &RISK_VAR = &
&CONTVARS = %THEN %DO;
%LET TABS = &BYVAR*&GIVENVAR;
%END;
%ELSE %IF &GIVENVAR = & &RISK_VAR = & &CONTVARS
NE %THEN %DO; %LET TABS = &BYVAR*&CONTMVAR;
%END;
%ELSE %IF &GIVENVAR = & &RISK_VAR NE  &
&CONTVARS = %THEN %DO;
%LET TABS = &BYVAR*&RISKMVAR;
%END;
%END;
%ELSE %IF &BYVAR = %THEN %DO;
```

```
%IF &GIVENVAR NE & &RISK_VAR NE & &CONTVARS NE
%THEN %DO;
%LET TABS = &RISKMVAR*&CONTMVAR*&GIVENVAR;
%END;
%ELSE %IF &GIVENVAR NE & &RISK_VAR NE &
&CONTVARS = %THEN %DO;
%LET TABS = &RISKMVAR*&GIVENVAR;
%END;
%ELSE %IF &GIVENVAR NE & &RISK_VAR = &
&CONTVARS NE %THEN %DO;
%LET TABS = &CONTMVAR*&GIVENVAR;
%END;
%ELSE %IF &GIVENVAR = & &RISK_VAR NE &
&CONTVARS NE %THEN %DO;
%LET TABS = &RISKMVAR*&CONTMVAR;
%END;
%ELSE %IF &GIVENVAR NE & &RISK_VAR = &
&CONTVARS = %THEN %DO; %LET TABS = &GIVENVAR;
%END;
%ELSE %IF &GIVENVAR = & &RISK_VAR = & &CONTVARS
NE %THEN %DO; %LET TABS = &CONTMVAR;
%END;
%ELSE %IF &GIVENVAR = & &RISK_VAR NE  &
&CONTVARS = %THEN %DO;
%LET TABS = &RISKMVAR;
%END;
%END;

%IF &EVENTIND = %THEN %DO;
%LET EVENTVAL = 1;
%END;
%ELSE %DO;
%LET EVENTVAL = &EVENTIND;
%END;

PROC FREQ DATA =ANALFILE NOPRINT;
 WHERE &EVENT = &EVENTVAL;
 TABLE &TABS / MISSING SPARSE LIST OUT = INCIDENT;
RUN;

DATA NUMBER;
 MERGE TOTAL(KEEP=&BYVAR &CONTVARS
        &GIVENVAR &RISK_VAR N)
 INCIDENT(KEEP=&BYVAR &CONTVARS &GIVENVAR
&RISK_VAR COUNT RENAME = (COUNT=C));
 BY &BYVAR &RISK_VAR &CONTVARS &GIVENVAR;
 LN = LOG(N);
RUN;

%GLOBAL _PRINT_;
%IF %UPCASE(&PRINTOUT) NE YES %THEN %DO;
%LET _PRINT_=OFF;
%END;

PROC GENMOD DATA = NUMBER;
 %IF &BYVAR NE %THEN %DO;
 BY &BYVAR;
 %END;
 MODEL C = &GIVENVAR &CONTVARS &RISK_VAR/
 DIST = POISSON
 LINK = LOG
 OFFSET = LN
 LRCI COVB;
 MAKE 'PARMEST' OUT = BETAS;
 MAKE 'COV' OUT = COVS;
RUN;


%IF &CONTVARS NE %THEN %DO;
PROC MEANS DATA = ANALFILE;
 VAR &CONTVARS;
 OUTPUT OUT= PROP MEAN=PROP1-PROP&CONT_NUM;
RUN;

%LET KEEPPROP= PROP1-PROP&CONT_NUM;
%END; %ELSE %DO; %LET KEEPPROP = ; %END;

DATA ADJ_COEF(KEEP= INT &KEEPPROP &GIVENVAR);
 %IF &CONTVARS NE %THEN %DO;
 SET PROP(KEEP=&KEEPPROP); %END;
 %IF &GIVENVAL > .Z %THEN %DO;
 &GIVENVAR = &GIVENVAL; %END; INT = 1;
RUN;

%IF &BYVAR NE %THEN %DO;
PROC FREQ DATA = ANALFILE;
 TABLE &BYSORT/OUT=BYL;
RUN;

DATA _NULL_;
 SET BYL(KEEP=&BYVAR) END=LAST;
 %DO BI = 1 %TO &BY_NUM;
 %LET BYV = %SCAN(&BYVAR,&BI);
 CALL SYMPUT("BY&BI"||LEFT(_N_),TRIM(&BYV));
 %END;
 IF LAST THEN CALL SYMPUT('BYLEVEL',_N_);
RUN;
%END; %ELSE %DO; %LET BYLEVEL = 1;
%END;

%MACRO CONTS;
 %IF &CONTVARS NE %THEN %DO;
 %DO COI = 1 %TO &CONT_NUM;
 PROP&COI %END;  %END;
%MEND;

%MACRO RISKS;
 %DO RISKJ = 1 %TO &RISK_NUM;  RISK&RISKJ
 %END;
%MEND;

PROC IML;

%IF %UPCASE(&PRINTOUT) = YES %THEN %DO;
RESET PRINT; %END;

PROP={INT &GIVENVAR %CONTS};
USE ADJ_COEF; READ ALL VAR PROP INTO PX;
CLOSE;

%IF &RISK_NUM >0 %THEN %DO;
ONE=I(&RISK_NUM);
ZERO = SHAPE(0,1,&RISK_NUM);
INIT = ZERO//ONE;
X = REPEAT(PX,NROW(INIT))||INIT;
RISK={%RISKS};
%END;
%ELSE %DO; X= PX;
%END;

USE BETAS; BETAVAR={ESTIMATE};
 READ ALL VAR BETAVAR INTO BETA
 WHERE(PARM^="SCALE");
CLOSE;

 USE COVS; COVVAR ='PRM1':"PRM&VAR_NUM";
    READ ALL VAR COVVAR INTO COV;
 CLOSE;
```

```
*--------------------------------------------------------------------------------*
| CALCULATE PREDICATED PROPORTION OF RISK
|VARIABLES BY BYVAR
*--------------------------------------------------------------------------------*;
%DO LI = 1 %TO &BYLEVEL;
%LET BEGROW = %EVAL((&LI-1)*&VAR_NUM+1);
%LET ENDROW = %EVAL(&LI*&VAR_NUM);
XBETA   = X*BETA[&begrow:&endrow,];
XCOVX   = VECDIAG(X*COV[&BEGROW:&ENDROW,]*X');
BETAS  = BETA[&begrow:&endrow,];
SES    = SQRT(VECDIAG(COV[&BEGROW:&ENDROW,]));
BETAS = BETAS/SES;
PBETAS = 2*(1-PROBNORM(ABS(TBETAS)));
STDERR = SQRT(XCOVX);
LOW95 = XBETA-1.96*STDERR;
UP95  = XBETA+1.96*STDERR;
CILOW = EXP(LOW95);
CIUP  = EXP(UP95);
ADJVALUE = EXP(XBETA);
ESTMTRX = BETAS||SES||TBETAS||PBETAS;
CESTNAME= {BETA SE Z PVALUE};
%IF %LENGTH(&GIVENVAR)>0 &
%LENGTH(&CONTVARS)>0 &
      %LENGTH(&RISK_VAR)>0 %THEN %DO;
      RESTNAME= {INT,&GIVENVAR,&CONTSK,&RISKSK};
%END;
%ELSE %IF &GIVENVAR =  & %LENGTH(&CONTVARS)>0
& %LENGTH(&RISK_VAR)>0 %THEN %DO;
    RESTNAME= {INT,&CONTSK,&RISKSK};
%END;
%ELSE %IF %LENGTH(&GIVENVAR)>0 & &CONTVARS =
& %LENGTH(&RISK_VAR)>0 %THEN %DO;
   RESTNAME= {INT,&GIVENVAR,&RISKSK};
%END;
%ELSE %IF %LENGTH(&GIVENVAR)>0 &
%LENGTH(&CONTVARS)>0 &
  &RISK_VAR = %THEN %DO;
  RESTNAME= {INT,&GIVENVAR,&CONTSK};
%END;
%ELSE %IF &GIVENVAR =  & %LENGTH(&CONTVARS)>0
 & &RISK_VAR = %THEN %DO;
     RESTNAME= {INT,&CONTSK};
%END;
%ELSE %IF %LENGTH(&GIVENVAR)>0 & &CONTVARS =
 & &RISK_VAR = %THEN %DO;
      RESTNAME= {INT,&GIVENVAR};
%END;
%ELSE %IF &GIVENVAR =  & &CONTVARS =  &
     %LENGTH(&RISK_VAR)>0 %THEN %DO;
       RESTNAME= {INT,&RISKSK};
%END;
%ELSE %DO;
        RESTNAME= {INT};
%END;
CREATE EST&LI FROM ESTMTRX[COLNAME=CESTNAME
                                ROWNAME=RESTNAME];
APPEND FROM ESTMTRX[ROWNAME=RESTNAME];
CLOSE EST&LI;

%IF %LENGTH(&RISK_VAR)>0 %THEN %DO;
CIMATRX=INIT||CILOW||CIUP||ADJVALUE||XBETA||STDERR;
              CINAMES ={&RISK_VAR CILOW CIUP
                ADJVALUE XBETA STDERR};
%END;
%ELSE %IF %LENGTH(&RISK_VAR)>0 %THEN %DO;
 CIMATRX =CILOW||CIUP||ADJVALUE||XBETA||STDERR;
 CINAMES ={CILOW CIUP ADJVALUE XBETA STDERR};
%END;
 CREATE CI&LI FROM CIMATRX[COLNAME = CINAMES];
APPEND FROM CIMATRX;
CLOSE CI&LI;
%END; QUIT;

*----------------------------------------------------------------------------*
|    CALCULATE PROPORTIONS OF EVENT.                    |
*----------------------------------------------------------------------------*;
PROC MEANS DATA = ANALFILE;  VAR &EVENT;
          OUTPUT OUT=YSUM SUM = YNUMBER;
RUN;

DATA TEMP;
   LENGTH YNAME $8.;
   %IF %LENGTH(&RISK_VAR)>0 %THEN %DO;
    IF _N_ = 1 THEN SET YSUM(KEEP=YNUMBER);
   %END;
    SET %DO QI = 1 %TO &BYLEVEL;
          CI&QI(IN=IN&QI)
          %END;;
  %IF %LENGTH(&BYVAR)>0 %THEN %DO;
        %DO SI = 1 %TO &BY_NUM;
            %LET BYV = %SCAN(&BYVAR,&SI);
            %DO QQ = 1 %TO &BYLEVEL;
                 IF IN&QQ THEN &BYV ="&&BY&SI.&QQ";
            %END;
         %END;
  %END;
   YNAME = %UPCASE("&EVENT");
RUN;

DATA ESTDATA;
   LENGTH YNAME $8.;
   SET %DO QI = 1 %TO &BYLEVEL;
EST&QI(IN=IN&QI) %END;;
      %IF %LENGTH(&BYVAR)>0 %THEN %DO;
            %DO SI = 1 %TO &BY_NUM;
              %LET BYV = %SCAN(&BYVAR,&SI);
              %DO QQ = 1 %TO &BYLEVEL;
                IF IN&QQ THEN &BYV ="&&BY&SI.&QQ";
       %END; %END; %END;
       YNAME = %UPCASE("&EVENT");
RUN;

%IF &OUT= %THEN %DO;  %LET OUTDATA = OUT;%END;
%ELSE %DO; %LET OUTDATA =&OUT; %END;

%IF &ESTOUT= %THEN %DO; %LET ESTDAT = ESTOUT;
%END; %ELSE %DO; %LET ESTDAT = &ESTOUT; %END;

PROC APPEND BASE = &OUTDATA DATA = TEMP;
PROC APPEND BASE = &ESTDAT DATA = ESTDATA;
%END;

%IF %LENGTH(&RISK_VAR)>0 %THEN %DO;
  PROC MEANS DATA = ANALFILE ;
        BY &BYVAR &RISK_VAR;
        VAR &FUTIME &EVENT ;
       OUTPUT OUT= XNNUM SUM= XNUMBER
EVENTNUM;
   RUN;

   PROC SORT DATA = &OUTDATA;
        BY &BYVAR &RISK_VAR;
   RUN;

   DATA &OUTDATA;
      MERGE &OUTDATA(IN=INA)
XNNUM(KEEP=&BYVAR &RISK_VAR _FREQ_
EVENTNUM XNUMBER RENAME=(_FREQ_=SIZE));
```

```
    BY &BYVAR &RISK_VAR; IF INA;
    NPREDIT = XNUMBER*ADJVALUE;
  RUN;

  PROC MEANS ; VAR NPREDIT;
     OUTPUT OUT = OVERALL SUM = OVERALL;
  RUN;

PROC MEANS DATA = &OUTDATA NOPRINT;
  VAR YNUMBER; OUTPUT OUT = SUMY MEAN = SUMY;
RUN;

DATA KNUMBER;
  MERGE SUMY(KEEP=SUMY)
OVERALL(KEEP=OVERALL);
        K = SUMY/OVERALL;
RUN;

DATA &OUTDATA;
  IF _N_ = 1 THEN SET KNUMBER;
  SET &OUTDATA;
  ADJUSTED = K*&PERSON*365.25*ADJVALUE;
   LOWCI    = K*&PERSON*365.25*CILOW;
  UPCI     = K*&PERSON*365.25*CIUP;
RUN;
%END;
%ELSE %DO;
    ADJUSTED = &PERSON*365.25*ADJVALUE;
     LOWCI    = &PERSON*365.25*CILOW;
     UPCI     = &PERSON*365.25*CIUP;
%END;

%IF %UPCASE(&PRINTOUT) = YES %THEN %DO;
PROC PRINT DATA = &OUTDATA;
        ID YNAME &RISK_VAR SIZE EVENTNUM
ADJUSTED LOWCI UPCI;
RUN;
%END;

%IF &ESTOUT= %THEN %DO;
   PROC DATASETS LIB=WORK;
    DELETE ESTOUT;  RUN;
%END;

%IF &OUT = %THEN %DO;
   PROC DATASETS LIB=WORK;
       DELETE OUT;  RUN;
%END;

%MEND;
```

→