Using SAS® Software to Compute Variances for Poisson Samples

Terry L. Pennington, United States Bureau of the Census, Washington, D.C.

ABSTRACT

Poisson sampling is a method of unequal probability sampling in which each sample unit is selected independently. It is a sample design common for economic surveys at the Bureau of the Census. Such sample designs offer increased sampling efficiency compared to stratified equal-probability sampling. This paper discusses the module for calculating sampling variances for Poisson sampling. The module is a component of the standardized economic processing system that is currently being developed at the Census Bureau. This SAS-based system will process and tabulate survey data. Design-specific variances are then calculated for the various survey data. The SAS code for the Poisson-variance module is displayed and discussed in detail.

INTRODUCTION

Poisson sampling, whereby sample units are independently selected with probability proportional to a measure of size, is commonly used in selecting samples for economic surveys at the Census Bureau. Examples of such surveys include the Annual Survey of Manufactures and the Plant Capacity Survey. In both surveys, large companies have a greater probability of selection than do small companies.

The Poisson-variance module is part of the larger Standard Economic Processing System (StEPS) currently being designed and developed at the Census Bureau. Among numerous other functions, StEPS will allow for data review and correction, imputation, outlier detection, estimation, and variance estimation. The long range goal of StEPS is to improve timeliness, increase efficiency, and avoid programming redundancy, thereby reducing survey processing costs.

The Poisson-variance module allows for weight adjustment and variance calculation via one of two user designated methods. The module itself is controlled by a process control file which allows the user to specify the data items for which the module is to calculate variances, the variance calculation method to be used, and the necessary tabulation, or *by*, variables.

ESTIMATORS FOR POISSON SAMPLES

Estimators. We let N denote the size of the population and n denote the number of sample units that have been selected.

The simple weighted estimator is

$$X' = \sum_{i=1}^{n} (W_i)\ (X_i)$$

where

W_i = the adjusted sample weight
X_i = the data value for observation I

An alternative estimator is

$$X'' = X' \left(\frac{N}{\hat{N}} \right)$$

where

$$\hat{N} = \sum_{i=1}^{n} (W_i)$$

Variance Estimators. The variance of X' is

$$\sigma_{X'}^2 = \sum_{i=1}^{n} (W_i)\ (W_i - 1)\ (X_i)^2$$

and the variance of X" is

$$\sigma_{X''}^2 = \sum_{i=1}^{n} (W_i)^2\ (X_i - \bar{X})^2 - \hat{N}\ \hat{S}_X^2$$

where

$$\hat{S}_X^2 = \frac{1}{\hat{N} - 1} \sum_{i=1}^{n} (W_i)\ (W_i - 1)\ (X_i - \bar{X})^2$$

and

$$\bar{X} = \frac{\sum_{i=1}^{n} (W_i)\ (X_i)}{\hat{N}}$$

Relative Standard Error Estimator. The relative standard error (rse) of X' is

$$V_{X'_i} = \frac{\sigma_{X'_i}}{X'_i}$$

and the rse of X" is

$$V_{X''_i} = \frac{\sigma_{X''_i}}{X''_i}$$

THE VARIANCE MODULE

The variance module, shown below, produces a SAS data set whereby each record contains a tabulation variable(s), variance, and rse. The variance and rse are presented via user-defined variable names. Example output has been included below.

The tabulation variable, variance, and rse names enter the module via a process control file (pcf). Macro *READ_IN* reads the ASCII pcf which contains an unknown number of records. Each record is prefaced by a TYPE variable. TYPE specifies whether the line contains the name of a data variable (DATA), tabulation variable (TAB), calculation method defining variable (METH), or response cell factor variable (RCELL). Each record prefaced by DATA must contain three variable names: a name for the data variable we have chosen to calculate variances for, the name of the variable we would like the result of the variance calculation to be placed in, and the name of the rse. Every other record in the pcf must contain only the TYPE variable plus the one additional variable containing the user-defined name. The file must contain only one record prefaced by METH, and up to 999 of each of the other remaining variable types. Macro *READ_IN* assigns the input variable names to global macro variables. An example of the pcf has been included below.

Macro *MERGE* merges the SAS data set containing the data values, MASTER, with the SAS data set containing the weight adjustment values, RFACT. The data sets are merged by user-specified tabulation variables and the weight is adjusted. If the weight is not to be adjusted, then a data set must be created with the appropriate tabulation variables and corresponding weights of 1.

Finally, macro *CALC* calculates the variances based upon the user-specified variance calculation method of VP1 or VP2. VP1 calculates the variance for X', while VP2 calculates the variance for X".

```
%************************************;
%* POISSON SAMPLING VARIANCE MODULE   *;
%************************************;

  %LET LIB      = STEPS;
        /*LIBNAME WHERE DATA MAY BE FOUND*/
  %LET LOCATION = '/home/tpenning/steps';
       /*ACTUAL LOCATION OF LIBRARY*/
  %LET MASTER   = MASTER2;
       /*NAME OF MASTER DATA SET*/
  %LET RFACT    = RFACT2;
       /*NAME OF DATA SET CONTAINING*/
       /*WEIGHT ADJUSTMENT VARIABLE*/
  %LET PCF      =
          "/home/tpenning/steps/pcf2a.dat";
      /*NAME AND LOCATION OF PROCESS CONTROL
    FILE*/
  %LET WGT      = WEIGHT; /*NAME OF WEIGHT VAR*/

%MACRO MAIN;

     %*THE FOLLOWING IS AN ALPHABETIC LIST OF
       THE GLOBAL MACRO VARIABLES:

       &NUM_DATA     NUMBER OF DATA VARIABLES
       &NUM_RCEL     NUMBER OF RESPONSE CELL
                     VARIABLES
       &NUM_TAB      NUMBER OF TABULATION
                     VARIABLES
       &&XT&I        DATA VARIABLE NAME
       &&R&I         RESULT RELATIVE STANDARD
                     ERROR NAME
       &&RCELL&I     LIST OF RESPONSE CELL
                     VARIABLES
       &&TAB&I       LIST OF TABULATION
                     VARIABLES
       &&V&I         RESULT VARIANCE VARIABLE
                     NAME
     ;

     /*READ IN THE ASCII PROCESS CONTROL FILE*/

  %READ_IN;

     /* MERGE - MERGES MASTER AND RFACT SAS DATA
        SETS */

  %MERGE;

     /* CALC - DOES THE ACTUAL VARIANCE
        CALCULATION */

  %CALC;
RUN;

%MEND MAIN;


%MACRO READ_IN;

  /*THE FOLLOWING READS IN THE PROCESS CONTROL
    FILE (PCF).  EACH LINE IN THE PCF CONTAINS
    A VARIABLE CALLED TYPE.  TYPE SPECIFIES
    WHETHER THAT LINE REPRESENTS
    A DATA (DATA), TABULATION (TAB), CALCULATION
    METHOD-DEFINING (METH),
    OR RESPONSE FACTOR CELL VARIABLE
    (RCELL).  TYPE VARIABLES
    MAY BE UPPER OR LOWER CASE.*/
```

```
  %* RULES FOR PCF FILE:

   VARIABLES                          LIMITATION:
   1.)  TABULATION VARIABLES             999
   2.)  DATA VARIABLES                   999
   3.)  RESULT VARIANCE VARIABLES        999
   4.)  RESULT RSE VARIABLES             999
   5.)  RESPONSE FACTOR CELL VARIABLE    999
          (RCELL)
   6.)  COMPUTATIONAL METHOD VARIABLE     01
        (METH)
            METH = VP1, VP2;

   %*EXAMPLE PCF FILE:

   /*1*/  TAB tab_nam
   /*2*/  DATA data_nam v_datanm r_datanm
   /*3*/  RCELL rcell_nm
   /*4*/  METH vp1

    WHERE
    /*1- PRIMARY TABULATION VAR RECODE*/
    /*2- DATA VAR, RESULT VAR, RSE VAR*/
    /*3- PRIMARY RESPONSE FACTOR CELL VAR
         RECODE*/
    /*4-  METHOD VAR VP1*/
   ;

DATA _NULL_;
  RETAIN NUM_DATA NUM_TAB NUM_RCEL 0;
  INFILE &PCF MISSOVER END=END;
  INPUT TYPE $ @;
    SELECT (TRIM(LEFT(UPCASE(TYPE))));
      WHEN ('DATA')
        DO;
          INPUT DATA_NAM $ VAR_NAM $ RSE_NAM $;
          NUM_DATA + 1;
          CHAR_NX =
            TRIM(LEFT(PUT(NUM_DATA,$3.)));
          CALL SYMPUT("XT"||CHAR_NX,DATA_NAM);
          CALL SYMPUT("V"||CHAR_NX,VAR_NAM);
          CALL SYMPUT("R"||CHAR_NX,RSE_NAM);
        END;
      WHEN ('TAB')
        DO;
          INPUT TAB_NAM $;
          NUM_TAB + 1;
          CHAR_NX =
            TRIM(LEFT(PUT(NUM_TAB,$3.)));
          CALL SYMPUT("TAB"||CHAR_NX,TAB_NAM);
        END;
      WHEN ('METH')
        DO;
          INPUT METHOD $;
          %GLOBAL METH;
          CALL SYMPUT("METH",METHOD);
        END;

      WHEN ('RCELL')
        DO;
          INPUT RCELLNAM $;
          NUM_RCEL + 1;
          CHAR_NX =
            TRIM(LEFT(PUT(NUM_RCEL,$3.)));
          CALL
            SYMPUT("RCELL"||CHAR_NX,RCELLNAM);
        END;
      OTHERWISE;
    END;
  IF END=1
    THEN
      DO;
        %GLOBAL NUM_DATA;
        CALL SYMPUT("NUM_DATA",NUM_DATA);
        %GLOBAL NUM_TAB;
        CALL SYMPUT("NUM_TAB",NUM_TAB);
        %GLOBAL NUM_RCEL;
        CALL SYMPUT("NUM_RCEL",NUM_RCEL);
      END;
  RUN;

  %DO I = 1 %TO &NUM_DATA;
      %GLOBAL &&XT&I;
      %GLOBAL &&V&I;
      %GLOBAL &&R&I;
  %END;

  %DO I = 1 %TO &NUM_TAB;
      %GLOBAL &&TAB&I;
  %END;

  %DO I = 1 %TO &NUM_RCEL;
      %GLOBAL &&RCELL&I;
  %END;
RUN;

%MEND READ_IN;

%MACRO MERGE;

  %* CALCULATE ADJWGT, MERGE RFACT AND MASTER*;

  /*MACRO XT, TAB, AND RCELL ARE USED
    TO GENERATE THE LIST
    OF DATA, RESULT, AND TAB VARIABLES
    TO BE USED IN THE
    KEEP AND BY STATEMENTS BELOW.*/

  %MACRO TAB;
    %DO I = 1 %TO &NUM_TAB;
       &&TAB&I
    %END;
  %MEND TAB;

  %MACRO XT;
    %DO I = 1 %TO &NUM_DATA;
       &&XT&I
    %END;
  %MEND XT;

  %MACRO RCELL;
    %DO I = 1 %TO &NUM_RCEL;
       &&RCELL&I
    %END;
  %MEND RCELL;

    /*SET THE MASTER DATA FILE TO A
      TEMPORARY DATA SET*/

   DATA MASTER;
     SET &LIB..&MASTER(KEEP = CFN &WGT %XT
         %TAB %RCELL STATUS);
   RUN;

    /*SET THE DATA FILE CONTAINING THE WEIGHT
      ADJUSTMENT FACTORS (RFACT) TO A TEMPORARY
      DATA SET*/

   DATA RFACT;
     SET &LIB..&RFACT(KEEP = %RCELL ADJ);
   RUN;

    /*SORT THE TEMPORARY MASTER DATA FILE*/

  PROC SORT DATA = MASTER NOEQUALS TAGSORT;
    BY %RCELL;
  RUN;

    /*SORT THE TEMPORARY DATA FILE
      CONTAINING THE WEIGHT ADJUSTMENT
      FACTORS*/
```

```
  PROC SORT DATA = RFACT NOEQUALS TAGSORT;
    BY %RCELL;
  RUN;

    /*MERGE TEMPORARY MASTER WITH
      TEMPORARY RFACT    */
    /* BY RCELL.  CALCULATE ADJUSTED
       WEIGHT (ADJWGT).*/
    /*IF ADJ = 1, THEN ADJWGT = WGT.*/

   DATA TEST;
      MERGE MASTER RFACT;
      BY %RCELL;
      ADJWGT = &WGT * ADJ;
   RUN;

%MEND MERGE;

%MACRO CALC;

  %*CALCULATE VARIANCES USING METHOD
    1 OR METHOD 2   *;

  %*METHOD 1 VARIABLE DEFINITIONS (METH = VP1):
    ----------------------------------------

    P1_XT&I = ADJWGT*(ADJWGT-1)*(Yi**2)
              "P1" STANDS FOR PART1 OF THE
               VARIANCE CALCULATION
    V_XT&I  IS THE SUMMATION OF P1_XT(I)
            ABOVE (I.E., THE VARIANCE)
    P2_XT&I = ADJWGT*XT(I)
              "P2" STANDS FOR PART 2 OF THE
               VARIANCE CALCULATION
    X_XT&I  IS THE SUMMATION OF P2_XT(I) OR
            THE WEIGHTED SUM
    RSE&I   REPRESENTS THE RELATIVE STANDARD
            ERROR
  ;

  %*METHOD 2 VARIABLE DEFINITIONS (METH = VP2):
    ----------------------------------------

    P2_XT&I = ADJWGT*XT(I)
    ADJWGT    IS THE ADJUSTED WEIGHT
    TOT_WGT   SUMMATION OF ADJUSTED WEIGHTS,
              I.E., THE ESTIMATED POPULATION
              SIZE
    X_XT&I    IS THE SUMMATION OF P2_XT(I) OR
              THE WEIGHTED SUM
    P_XT&I    IS THE ESTIMATED POPULATION MEAN
              I.E., THE WEIGHTED SUM DIVIDED BY
              THE ESTIMATED POP. SIZE
    SXT&I     (ADJWGT*(ADJWGT-1)*
              ((XT(I)-P_XT(I))**2))
    PXT&I     (ADJWGT**2)*((XT(I)-P_XT(I))**2)
              THE FIRST PART OF THE VARIANCE
              FORMULA PRIOR TO SUMMATION
    TSX&I     SUMMATION OF SXT
    TPX&I     SUMMATION OF PXT
    N_S2Y&I   PARTIAL RESULT
    V_XT&I    IS THE SUMMATION OF P1_XT(I)
              ABOVE (I.E., THE VARIANCE)
    RSE&I     REPRESENTS THE RELATIVE STANDARD
              ERROR;

  /*THE FOLLOWING MACROS ARE USED TO GENERATE
    LISTS OF VARIABLES USED IN THE KEEP,
    VAR, BY, AND SUM STATEMENTS.
    THEY ARE LISTED ALPHABETICALLY.*/

  %MACRO P_XT;
    %DO I = 1 %TO &NUM_DATA;
       P_XT&I
    %END;
  %MEND P_XT;

  %MACRO P1_XT;
    %DO I = 1 %TO &NUM_DATA;
       P1_XT&I
    %END;
  %MEND P1_XT;

  %MACRO P2_XT;
    %DO I = 1 %TO &NUM_DATA;
       P2_XT&I
    %END;
  %MEND P2_XT;

  %MACRO PXT;
    %DO I = 1 %TO &NUM_DATA;
       PXT&I
    %END;
  %MEND PXT;

  %MACRO R;   /*GLOBAL VARIABLE*/
    %DO I = 1 %TO &NUM_DATA;
       &&R&I
    %END;
  %MEND R;

  %MACRO RSE;
    %DO I = 1 %TO &NUM_DATA;
       RSE&I
    %END;
  %MEND RSE;

  %MACRO SXT;
    %DO I = 1 %TO &NUM_DATA;
       SXT&I
    %END;
  %MEND SXT;

  %MACRO TAB;  /*GLOBAL VARIABLE*/
    %DO I = 1 %TO &NUM_TAB;
       &&TAB&I
    %END;
  %MEND TAB;

  %MACRO TPX;
    %DO I = 1 %TO &NUM_DATA;
       TPX&I
    %END;
  %MEND TPX;

  %MACRO TSX;
    %DO I = 1 %TO &NUM_DATA;
       TSX&I
    %END;
  %MEND TSX;

  %MACRO X_XT;
    %DO I = 1 %TO &NUM_DATA;
       X_XT&I
    %END;
  %MEND X_XT;

  %MACRO XT;      /*GLOBAL VARIABLE*/
    %DO I = 1 %TO &NUM_DATA;
       &&XT&I
    %END;
  %MEND XT;

  %MACRO V;     /*GLOBALVARIBLE*/
    %DO I = 1 %TO &NUM_DATA;
       &&V&I
    %END;
  %MEND V;
```

```
 %MACRO V_XT;
   %DO I = 1 %TO &NUM_DATA;
      V_XT&I
   %END;
%MEND V_XT;

 %IF &METH = VP1
    %THEN
       %DO;
          DATA TEST1(KEEP = %TAB
                %P1_XT %P2_XT);
            SET TEST;
            BY %TAB;
            IF STATUS = 'A';
            %DO I = 1 %TO &NUM_DATA;
              IF &&XT&I NE .
                THEN
                  DO;
                    P1_XT&I =  ADJWGT*
                    (ADJWGT - 1)*(&&XT&I**2);
                    P2_XT&I =  ADJWGT*&&XT&I;
                  END;
                ELSE
                  DO;
                    P1_XT&I = .;
                    P2_XT&I = .;
                  END;
             %END;
          RUN;

          PROC SUMMARY DATA = TEST1;
            VAR %P1_XT %P2_XT;
            BY %TAB;
            OUTPUT OUT = VP1 (KEEP = %TAB
                   %V_XT %X_XT)
                   SUM = %V_XT %X_XT;
          RUN;

          DATA VP1 (KEEP = %TAB %V %R);
            SET VP1;
            BY %TAB;
            %DO I = 1 %TO &NUM_DATA;
              RSE&I = ((SQRT(V_XT&I))/X_XT&I);
              &&V&I = V_XT&I;
              &&R&I = RSE&I;
            %END;
          RUN;

       %END;

/*******************/

     %ELSE
        %IF &METH = VP2
          %THEN
            %DO;

              DATA TEST2(KEEP = %TAB
                ADJWGT %P2_XT %XT);
                SET TEST;
                BY %TAB;
                IF STATUS = 'A';
                %DO I = 1 %TO &NUM_DATA;
                  IF &&XT&I NE .
                    THEN
                      DO;
                        P2_XT&I   = ADJWGT*
                        &&XT&I;
                      END;
                    ELSE
                      DO;
                        P2_XT&I   = .;
                        ADJWGT    = .;
                      END;
              %END;
              RUN;

              PROC SUMMARY DATA = TEST2;
                 VAR ADJWGT %P2_XT;
                 BY %TAB;
                 OUTPUT OUT = VP2A (KEEP =
                        TOT_WGT %TAB %X_XT N)
                        SUM = TOT_WGT %X_XT
                              N = N;
               RUN;

               DATA VP2A;
                 SET VP2A;
                 BY %TAB;
                 %DO I = 1 %TO &NUM_DATA;
                   P_XT&I = X_XT&I/TOT_WGT;
                 %END; RUN;

              DATA TEST2(KEEP = %TAB %SXT %PXT);
                MERGE TEST2 VP2A;
                BY %TAB;
                %DO I = 1 %TO &NUM_DATA;
                   IF &&XT&I NE .
                     THEN
                       DO;
                         SXT&I    = ((ADJWGT)*
                          (ADJWGT-1)*((&&XT&I-
                           P_XT&I)**2));
                         PXT&I    = (ADJWGT**2)*
                          ((&&XT&I - P_XT&I)**2);
                       END;
                     ELSE
                       DO;
                         SXT&I    = .;
                         PXT&I    = .;
                       END;
                %END;
              RUN;

              PROC SUMMARY DATA = TEST2;
                VAR %SXT %PXT;
                BY %TAB;
                OUTPUT OUT  = VP2B (KEEP =
                          %TAB %TSX %TPX N2)
                          SUM  = %TSX %TPX N = N2;
              RUN;

              DATA VP2(KEEP = %TAB %V %R);
                 MERGE VP2A VP2B;
                 BY %TAB;
                 %DO I = 1 %TO &NUM_DATA;
                   N_S2Y&I = ((TOT_WGT/
                    (TOT_WGT-1))*TSX&I);
                   V_XT&I  = TPX&I - N_S2Y&I;
                   RSE&I   = (SQRT(V_XT&I))/
                     X_XT&I;
                   &&V&I   = V_XT&I;
                   &&R&I   = RSE&I;
                 %END;
              RUN;
        %END;
    RUN;

%MEND CALC;


%* CALL TO START PROGRAM*;

%MAIN;
RUN;
```

EXAMPLE PROCESS CONTROL FILE

```
TAB STRATUM
DATA TOT1 V_TOT1 R_TOT1
DATA TOT2 V_TOT2 R_TOT2
RCELL STRATUM
METH VP1
```

EXAMPLE OUTPUT

VARIANCES FOR X'
METH = VP1

OBS	STRATUM	V_TOT1	R_TOT1	V_TOT2	R_TOT2
1	01	207833081.59	0.01078	831332326.37	0.01078
2	02	107652056.07	0.03475	430608224.26	0.03475
3	03	24559743.91	0.01986	98238975.64	0.01986
4	04	1297142301.30	0.02894	5188569205.21	0.02894
5	08	2875651458.41	0.07036	11502605833.62	0.07036
6	10	113817101.56	0.00161	455268406.25	0.00161
7	11	358967314464.23	0.06549	1435869257856.9	0.06549
8	12	1144178639.00	0.01432	4576714556.00	0.01432
9	20	39304105.55	0.00296	157216422.22	0.00296
10	21	2052323095.26	0.04148	8209292381.03	0.04148
11	22	89808981.65	0.01461	359235926.60	0.01461
12	30	1832888394.30	0.14816	7331553577.21	0.14816

Note: In order to create a second variable, TOT2, in the test file for this example, variable TOT1 was doubled. Therefore, variances of TOT2 are four times greater than those of TOT1 and relative standard errors of TOT1 and TOT2 are equivalent.

CONCLUSION

The SAS system is an effective tool for creating a generalized application, such as the poisson variance module. Over time, such applications conserve resources, increase efficiency, and reduce turnaround time.

REFERENCES

Sardnal, Swensson, Wretman (1992), *Model Assisted Survey Sampling*, New York, NY: Springer Series in Statistics.

Census Bureau (12/11/95), *Standard Economic Processing System; Document 1: StEPS Concepts and Overview*

SAS Institute (1990), *SAS Guide to Macro Processing*, Version 6, 2nd Edition, Cary, NC: SAS Institute Inc.

SAS is a registered trademark of the SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

The author may be contacted at:
United States Bureau of the Census
ESMPD
1021-4
Suitland Federal Center
Washington, DC 20233
(301) 457-4426
Terry.L.Pennington@ccmail.census.gov