Uffe Møller

OC and ASN of the SPRT for the Poisson Process

Preprint
April
1976

8

Institute of Mathematical Statistics
University of Copenhagen

Uffe Møller

OC AND ASN OF THE SPRT FOR THE POISSON PROCESS

Preprint 1976 No. 8

INSTITUTE OF MATHEMATICAL STATISTICS
UNIVERSITY OF COPENHAGEN

April 1976

OC AND ASN OF THE SPRT FOR THE POISSON PROCESS

by
Uffe Møller
Institute of Mathematical Statistics
University of Copenhagen

Summary

The paper presents an algorithm for the operating characteristic (OC) and the average sample number (ASN) functions of the sequential probability ratio test (SPRT) of the mean of a Poisson process.

As a basis of the algorithm an appendix gives a discussion of the accuracy of different formulas for the OC and ASN.

Key words

Poisson process, sequential probability ratio test, operating characteristic, average sample number, algorithm, approximation.

Language: Algol 60

Purpose:

In a Poisson process where λ denotes the mean occurence rate per observational unit, we want to test the hypotesis $\lambda = \lambda_1$ against the alternative $\lambda = \lambda_2$, $\lambda_2 > \lambda_1 > 0$. The sequential probability ratio test (SPRT) has a continuation region of the form

$$- a + \lambda_{12}t < x_t < r + \lambda_{12}t \quad \text{for } t > 0.$$

where x_t denotes the number of events in $[0,t]$,

$$\lambda_{12} = (\lambda_2 - \lambda_1) / (\ln(\lambda_2/\lambda_1)).$$

a and r could be determined as shown by Hald and Møller (1976).

For given $v = \lambda/\lambda_{12} \geqq 0$ and $a,r > 0$ the procedure computes the probability of accepting $\lambda = \lambda_1$ (OC(λ)) and the average sample number (ASN(λ)).

Tables of the OC and ASN for selected values of (v,a,r) have been given by Kiefer and Wolfowitz (1956).

<u>Method</u>:

The OC and ASN functions have been derived by Bartky (1943), Burman (1946) and Dvoretzky et al. (1953).

With OC $(\lambda) = L(a,v)/L(a+r,v)$ and $ASN(\lambda) = OC(\lambda)S(a+r,v) - S(r,v)$ we compute $L(y,v)$ and $S(y,v)$ according to the value of y.

If y is less than 8 we must use the exact formulas rearranged in this way

$$L(y,v) = e^{yv} \sum_{j=0}^{[y]} \left\{(j-y)ve^{-v}\right\}^j / j!$$

$$S(y,v) = \sum_{j=0}^{[y]} \sum_{i=j+1}^{[y]} \left\{(i-y)v\right\}^j / j! * e^{(y-i)v} - [y] - 1$$

which makes computation rather easy.

Otherwise we find the solution, t, to the equation

$$v = t / (e^t - 1)$$

and use the approximation formulas from Bartky (1943):

$$L(y,v) \simeq \begin{cases} 1/(1-v) + e^{-yt}/(1-v-t) & v \neq 1 \\ 2(y + 1/3) & v = 1 \end{cases}$$

$$S(y,v) \simeq \begin{cases} (y+\delta_v)v/(1-v) - L(y,v)(t+v)/t & v \neq 1 \\ (y + 1/3)(y - 5/3) - 1/18 & v = 1 \end{cases}$$

with $\delta_v = 1/t - \frac{1}{2}v/(1-v)$.

<u>Structure</u>:

real procedure *sprtoc (v,a,r,asn)*;

Formal parameters:

v Real value: $v = \lambda/\lambda_{12}$

a Real value: a

r Real value: r

asn Real output: the average sample number

Failure indications:

If v, a, r are outside the allowed range indicated in the Purpose section, *sprtoc* and *asn* will be set to -1.

Restrictions:

None.

Accuracy:

For a computer working with at least 11 significant digits, the algorithm will normally work with a smallest relative accuracy at 10^{-8}. The only exception is when a is small and r is large at the same time, in this case the relative accuracy of ASN may be as small as 10^{-3}.
If the computer works with less significant digits, the accuracy will be less and we may use the approximation formulas for smaller values of y.

Given the number of significant digits at the computer we are able to give a rough guide for the accuracy of the result using the exact formulas by means of the following table. This could again lead to a new limit on y for using the approximation formulas.

y	Number of lost digits computing L(y,1) by the exact formula.	a+r	Number of lost digits computing ASN by the exact formula with $v = 1$ and $\frac{1}{2} < r/a < 2$.
5	1.3		
6	1.8	6	1.1
7	2.2	7	1.5
8	2.7	8	1.9
9	3.2	9	2.3
10	3.7	10	2.7
12	4.7	12	3.7
14	5.7	14	4.6
16	6.7	16	5.6
18	7.7	18	6.5

The number of correct digits computing L(y,1) by Bartky's approximation formula is approximately $y + 1$.

REFERENCES:

Bartky, W. (1943): Multiple sampling with constant probability. Ann. Math. Statist., 14, 363-377.

Burman, J.P. (1946): Sequential sampling formulae for a binomial population. Journ. Roy. Statist. Soc., Suppl., 8, 98-103.

Dvoretzky, A., J. Kiefer and J. Wolfowitz (1953): Sequential decision problems for processes with continuous time parameter. Testing hypotheses. Ann. Math. Statist., 24, 254-264.

Hald. A. and U. Møller (1976): On the SPRT of the mean of a Poisson process. Preprint no. 5, Inst. Math. Statist., Univ. Copenhagen.

Jones, H.L. (1952): Formulas for the group sequential sampling of attributes. Ann. Math. Statist., 23, 72-87.

Kiefer, J. and J. Wolfowitz (1956): Sequential tests of hypoteses about the mean occurence time of a continuous parameter Poisson process. Naval Res. Logistics Quart., 3, 205-219.

Wald, A. (1947): Sequential analysis. Wiley, New York.

```
REAL PROCEDURE SPRTOC(V, A, R, ASN);
VALUE                   V, A, R;
REAL                    V, A, R, ASN;

COMMENT  COMPUTES OC AND ASN OF THE SPRT FOR A POISSON PROCESS;

  BEGIN
  REAL T, DELTA, AR, TERM, EPS, LR, LAR, SR, SAR, V1, THIRD, D18,
    HALF, ONE, TWO, FOUR;
  INTEGER IR, IAR, I, J;

  EPS   := 1.0₁₀-8;                      COMMENT  SMALL NUMBER;
  D18   := 0.0555555555555555556;    COMMENT  1/18;
  THIRD := 0.3333333333333333333;    COMMENT  1/3;
  HALF  := 0.5;                          COMMENT  1/2;
  ONE   := 1.0;                          COMMENT  1;
  TWO   := 2.0;                          COMMENT  2;
  FOUR  := 4.0;                          COMMENT  4;

  COMMENT  CHECK PARAMETERS;
  IF V LSS 0.0 OR A LEQ 0.0 OR R LEQ 0.0 THEN
    BEGIN
    COMMENT  INVALID PARAMETERS;
    SPRTOC := ASN := -ONE;
    GOTO  EXIT
    END;

  IR  := ENTIER(R - EPS);
  AR  := A + R;
  IAR := ENTIER(AR - EPS);

  IF IAR GEQ 8 THEN
    BEGIN
    COMMENT  COMPUTE PARAMETERS FOR APPROXIMATION FORMULAS:
      SOLUTION, T, TO       V = T/(EXP(T) - 1)
      AND                DELTA = 1/T - 0.5×V/(1-V)              ;

    V1 := ONE - V;    TERM := V1 × THIRD;
    T := V1 × (TWO + TERM×(TWO + TERM×(FOUR + TERM×(8.8 + TERM×20.8))));
    IF ABS(V1) GTR 0.01 THEN
      BEGIN
      REAL DERIV, TOLD, FUN, ET1;

      COMMENT  FOR V ≠ 1 USE NEWTON ITERATION;

LOOP:
      ET1 := EXP(T) - ONE;
      FUN := T / ET1;
      DERIV := (ONE - FUN)/ET1 - FUN;
      TOLD := T;
      T := T - (FUN-V)/DERIV;
      IF ABS(T-TOLD) GTR EPS×ABS(T) THEN GOTO LOOP
      END;
    DELTA := IF ABS(T) GTR EPS THEN ONE/T - HALF×V/V1 ELSE THIRD
    END  APPROXIMATION PARAMETERS;

  COMMENT  COMPUTE LR AND SR;
```

```
IF IR LSS 8 THEN
  BEGIN
  REAL ARRAY EXPS, BASE, POTS(0:IAR);
  REAL DIDJ, EXPNV, EXPNAV;

  COMMENT  COMPUTE L(R,V) AND S(R,V) BY SUMMATION;

  EXPNV := EXP(-V);
  EXPNAV := EXP(-A×V);
  EXPS(0) := TERM := EXP(R×V);

  FOR I := 1 STEP 1 UNTIL IR DO
    BEGIN
    TERM := TERM × EXPNV;   EXPS(I) := TERM;
    COMMENT  EXPS(I) = EXP(R-I);
    END  I;

  TERM :=  - R × V;
  LR := EXPS(0);
  SR := IR + 1;
  FOR I := 1 STEP 1 UNTIL IR DO
    BEGIN
    TERM := TERM + V;
    BASE(I) := TERM;    POTS(I) := ONE;
    COMMENT  BASE(I) = (I-R)×V;
    SR := SR - EXPS(I)
    END  I;
  FOR J := 1 STEP 1 UNTIL IR DO
    BEGIN
    DIDJ := ONE / J;
    LR := LR + BASE(J)×POTS(J)×EXPS(J)×DIDJ;
    FOR I := J+1 STEP 1 UNTIL IR DO
      BEGIN
      POTS(I) := TERM := BASE(I)×POTS(I)×DIDJ;
      COMMENT  POTS(I) = ((I-R)×V)××J/FAK(J);
      SR := SR - TERM×EXPS(I)
      END  I;
    END  J;

  IF IAR LSS 8 THEN
    BEGIN

    COMMENT  COMPUTE L(A+R,V) AND S(A+R,V) BY SUMMATION;

    TERM := EXPS(IR);
    FOR I := IR+1 STEP 1 UNTIL IAR DO
      BEGIN
      TERM := TERM × EXPNV;    EXPS(I) := TERM;
      COMMENT  EXPS(I) = EXP(R-I);
      END  I;

    TERM := - AR × V;
    LAR := EXPS(0);
    SAR :=  - (IAR + 1) × EXPNAV;
    FOR I := 1 STEP 1 UNTIL IAR DO
      BEGIN
      TERM := TERM + V;
      BASE(I) := TERM;    POTS(I) := ONE;
      COMMENT  BASE(I) = (I-(A+R)) × V;
      SAR := SAR + EXPS(I)
      END  I;
```

```
        FOR J := 1 STEP 1 UNTIL IAR DO
          BEGIN
          DIDJ := ONE / J;
          LAR := LAR + BASE(J)×POTS(J)×EXPS(J)×DIDJ;
          FOR I := J+1 STEP 1 UNTIL IAR DO
            BEGIN
            POTS(I) := TERM := BASE(I)×POTS(I)×DIDJ;
            COMMENT  POTS(I) = ((I-(A+R))×V)××J / FAK(J);
            SAR := SAR + TERM×EXPS(I)
            END  I;
          END  J;

        TERM := LR / LAR;
        SPRTOC := TERM × EXPNAV;
        ASN := TERM×SAR + SR;
        COMMENT  A+R <= 8 FINISHED;
        END
                        ELSE
        BEGIN

        COMMENT  COMPUTE L(A+R,V) AND S(A+R,V) BY APPROXIMATION;

        IF ABS(T) LSS EPS THEN
          BEGIN
          TERM := AR + THIRD;
          LAR := TWO × TERM;
          SAR := TERM×(TERM-TWO) - D18;
          END
                        ELSE
          BEGIN
          LAR := ONE/V1 + EXP(-AR×T)/(V1-T);
          SAR := (AR+DELTA)×V/V1 - LAR×(T+V)/T
          END;
        SPRTOC := TERM := LR / LAR;
        ASN := TERM×SAR + SR
        END;
      COMMENT   R <= 8 FINISHED;
      END
                            ELSE
      BEGIN

      COMMENT  USE APPROXIMATION FOR ALL TERMS;

      IF ABS(T) LSS EPS THEN
        BEGIN
        SPRTOC := (R+THIRD) / (AR+THIRD);
        ASN := A × (R+THIRD + D18/(AR+THIRD))
        END
                        ELSE
        BEGIN
        TERM := (V1-T)/V1;
        SPRTOC := TERM := (TERM + EXP(-R×T))/(TERM + EXP(-AR×T));
        ASN := ((AR+DELTA)×TERM - R-DELTA) × V / V1
        END
      END  APPROXIMATION;
  EXIT:
    END  SPRTOC;
```

APPENDIX

THE ACCURACY OF THE FORMULAS FOR THE OC AND ASN.

1. The exact formulas

It is well-known - see e.g. Burman (1946) - that the OC of the SPRT is

$$P(\lambda) = H(a,r,v) = L(r,v)/L(a+r,v), \quad v = \lambda/\lambda_{12} \tag{1}$$

with

$$L(y,v) = e^{yv} \sum_{i=0}^{[y]} \left\{(i-y)ve^{-v}\right\}^i/i! \tag{2}$$

where $a,r > 0$, $v \geqq 0$ and $[y]$ denotes the integral part of y.

It it also known that the ASN function may be found by means of

$$ASN(\lambda) = M(a,r,v)v =$$

$$H(a,r,v)\left\{\sum_{i=1}^{[a+r]} L(a+r-i,v)-[a+r]-1\right\} - \left\{\sum_{i=1}^{[r]} L(r-i,v)-[r]-1\right\} \tag{3}$$

It is easy to verify that the definition of $[y]$ could be changed to

$$[y] = \max \{i \in Z \mid i < y\},$$

which is more convenient for computational purposes.

For small values of a and r these formulas give accurate results within a reasonable amount of computation. But as a and r become larger this is not necessarily true.

It is easy to show that for fixed y, the minimum for the sum in (2) is found for $v = 1$. This means that in investigations of the accuracy it is reasonable to consider $L(y,1)$. From approximations we know that for large values of y we will have $L(y,1) \simeq 2(y+1/3)$ and this means that the sum in (2) will be approximately $2ye^{-y}$. The first term in the alternating sum is 1 and the next are of the same order of magnitude. For increasing values of y the accuracy of (2) and (3) is thus decreasing rapidly.

At UNIVAC 1110 where double precision real numbers have 17-18 correct decimals we may use (1) and (3) for $a+r < 20$ yielding at least 6-7 significant digits. If we want the same accuracy for larger values of a and r, we must use an approximation formula with this property.

WALD'S APPROXIMATION

The most well-known approximation is given by Jones (1952) adapting the formulas found by Wald (1947) for the approximate values of (a,r) to the exact (a,r).

$$P(\lambda) \simeq P_W(\lambda) = \begin{cases} \{e^{(r+1/3)t}-1\} \;/\; \{e^{(r+1/3)t}-e^{-at}\} & v \neq 1 \\ (r+1/3) \;/\; (a+r+1/3) & v = 1 \end{cases} \qquad (4)$$

where t is the solution of $v = t/(e^t-1)$, $v \neq 1$. (5)

This approximation could also be written in terms of

$$L(y,v) \simeq L_W(y,v) = \begin{cases} \{e^{-(y+1/3)t}-1\} \;/\; (1-v-t) & v \neq 1 \\ 2\;(y+1/3) & v = 1 \end{cases} \qquad (6)$$

The approximation to the ASN is found from

$$M(a,r,v) \simeq M_W(a,r,v) = \begin{cases} \{(a+r+1/3)P_W(\lambda) - (r+1/3)\} \;/\; (1-v) & v \neq 1 \\ a(r+1/3) & v = 1 \end{cases} \qquad (7)$$

Table 1 shows a summary of the investigation of the accuracy of (4) and (7) for $0.001 \leqq P(\lambda) \leqq 0.999$. For both formulas the accuracy is shown in the case where it is smallest, which is a $P(\lambda) = 0.999$ for (4) and at $P(\lambda) = 0.001$ for (7). The accuracy is measured in terms of $-\log_{10}\{(f_W(\lambda)-f(\lambda))/f(\lambda)\}$, which gives the number of correct significant digits in the approximation. Investigating the OC we use the tail probability for f.

Table 1a.

Table of $-\log_{10} \{(P_W(\lambda)-P(\lambda)) / (1-P(\lambda))\}$ for $P(\lambda) = 0.999$.

		r				
		2	4	6	8	10
	4	1.46				
	8	1.46	1.98	2.52		
a+r	12	1.46	1.98	2.50	2.86	3.20
	16	1.46	1.98	2.50	2.86	3.14
	20	1.46	1.98	2.50	2.86	3.13

Table 1b.

Table of $-\log_{10} \{(ASN_W(\lambda)-ASN(\lambda)) / ASN(\lambda)\}$ for $P(\lambda) = 0.001$.

		r				
		2	4	6	8	10
	4	1.34				
	8	1.82	2.03	1.95		
a+r	12	2.09	2.29	2.34	2.30	2.17
	16	2.17	2.46	2.54	2.57	2.55
	20	2.27	2.58	2.68	2.74	2.75

BARTKY'S APPROXIMATION

Bartky (1943) has given approximations to the OC for the binomial SPRT From which we can derive for $P(\lambda) \simeq P_B(\lambda) = L_B(r,v)/L_B(a+r,v)$

$$L_B(y,v) = \begin{cases} 1/(1-v) + e^{-yt}/(1-v-t) & v \neq 1 \\ 2(y+1/3) & v = 1 \end{cases} \qquad (8)$$

where t is given in (5).

For the ASN we get

$$\sum_{i=1}^{[y]} L(y-i)-[y]-1 \simeq \begin{cases} (y+\delta_v)v/(1-v) - L_B(y,v)(t+v)/t & v \neq 1 \\ (y+1/3)(y-5/3) - 1/18 & v = 1 \end{cases} \qquad (9)$$

with $\delta_v = 1/t - \frac{1}{2}\, v/(1-v)$.

This leads to

$$M(a,r,v) \simeq M_B(a,r,v) = \begin{cases} \{(a+r+\delta_v)P_B(\lambda) - (r+\delta_v)\}/(1-v) & v \neq 1 \\ a\{r + 1/3 + 1/(18(a+r+1/3))\} & v = 1 \end{cases} \qquad (10)$$

In the same way as for Wald's approximation we show the relative accuracy of Bartky's formulas in table 2.

Table 2a.

Table of $-\log_{10} \{(P_B(\lambda)-P(\lambda))/(1-P(\lambda))\}$ for $P(\lambda) = 0.999$.

		r				
		2	4	6	8	10
	4	1.02				
	8	1.02	3.05	5.32		
a+r	12	1.02	3.05	5.35	7.29	> 8
	16	1.02	3.05	5.35	7.29	> 8
	20	1.02	3.05	5.35	> 7	> 8

Table 2b.

Table of $-\log_{10} \{(ASN_B(\lambda) - ASN(\lambda))/ASN(\lambda)\}$ for $P(\lambda) = 0.001$.

		r				
		2	4	6	8	10
	4	1.79				
	8	2.55	3.75	4.61		
a+r	12	2.75	4.38	5.99	6.74	5.00
	16	2.84	4.63	6.50	8.20	8.69
	20	2.89	4.75	6.75	> 8	> 8