รายการอ้างอิง

ภาษาไทย

จงดี โรจนประศาสน์, การเปรียบเทียบวิธีการประมาณค่าพารามิเตอร์ในสมการถดถอยเชิงเส้นอย่างง่ายเมื่อตัวแปรตามมีค่าที่ถูกตัดทิ้ง. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาสถิติ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2536.

ปรานี รัตนัง, การประมาณสัมประสิทธิ์การถดถอยพหุเมื่อความผิดพลาดมีการแจกแจงแบบเบ้และมีการแจกแจงแบบหางยาวกว่าปกติ. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาสถิติ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2531.

อัมพร ชาตบุษยมาส, การวิเคราะห์ความถดถอยเมื่อตัวแปรตามบางค่ามีค่าขาดหาย. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาสถิติ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2530.

ภาษาอังกฤษ

A. P. Dempster , N. M Laird and D. B. Rubin, " A maximum likelihood from incomplete data via the EM algorithm." Journal of Royal Statistics (1977) : B,28,1-22.

Averill M. Law and W. David Kelton. Simulation Modeling & Analysis. Singapore : McGraw-Hill, 1991.

D.G. Kleinbaum, L.L. Kupper, and E.J. Muller, Applied Regression Analysis and Other Multivariate Methods. Boston : PWS-Kent ,1988.

Elisa T. Lee. Statistical Methods for Survival Data Analysis. California : Wadsworth, 1980.

Glenn Heller and Jeffrey S. Simonoff, "A comparison of estimators for regression with a censored response variable." Biometrica (1990) : 77,3,515-520.

Helmut Schneider, "Estimation in linear models with censored data." Biometrika (1986) :73, 3,741-745.

H. Koul, V. Susarla and J.van Ryzin, "Regression analysis with randomly right-censored data." The annuals of statistics (1981) : 9,6,1276-1288.

J.F. Lawless. Statistical Models and Methods for Lifetime Data. New York : John Wiley & Sons, 1982.

Jonathan Buckley and Ian James, "Linear regression with censored data." Biometrica (1979) : 66,3,429-436.

Josef Schmee and Geral J. Hahn, "A simple method for regression analysis with censored data." Technometrics (1979) : 21,4,417-432.

Murry Aitkin, "A note on the regression analysis of censored data." Technometrics (1981) : 23,2,161-163.

Robert V. Hogg. & Stuart A. Klugman. Loss Distributions. New York : John & Wiley and Sons,1984.

Rupert G. Miller. Survival Analysis . New York : John Wiley and Sons Inc, 1981.

Ruppet G. Miller and Jerry Helpern, "Regression with censored data." Biometrika (1982) : 69, 3,521-531.

S. Chattejee and D.L. McLeish, "Fitting linear regression models to censored data by least squares and maximum likelihood methods." Communication Statistics. (1986) : A,15, 3227-3243

Sue Leurgans, "Linear models,random censoring and synthetic data." Biometrica. (1987) : 74,2, 301-309.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

การสร้างตัวเลขสุ่ม (Random Number)

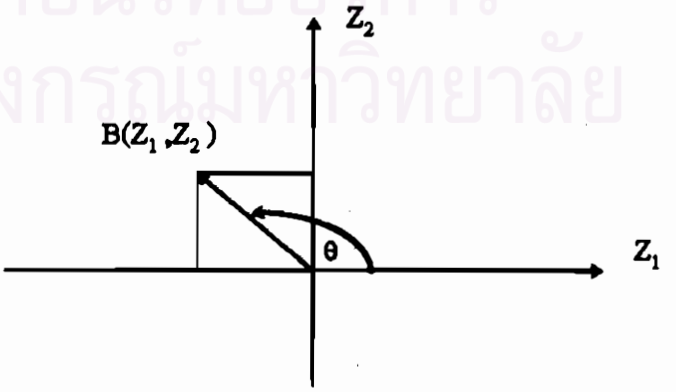
การสร้างลักษณะการแจกแจงแบบต่างๆ นั้น จะต้องอาศัยตัวเลขสุ่มเป็นพื้นฐานในการสร้าง สาหรับการวิจัยครั้งนี้จะใช้วิธีการสร้างตัวเลขสุ่มตามวิธีของ White และ Schmidt (1975) ซึ่งขั้นตอนในการสร้าง จะแสดงรายละเอียดด้วยฟังก์ชันต่อไปนี้

```
FUNCTION RAND(IX)
IX  = IX*16807
IF (IX.LT.0) IX = IX + 2147483647+1
RAND = IX
RAND = RAND*0.465661E-9
RETURN
END
```

ค่า IX จะเป็นค่า SEED หรือค่าเริ่มต้น ซึ่งจะต้องเป็นจำนวนเต็มบวกที่เป็นเลขคี่ RAND จะเป็นค่าของตัวเลขสุ่มที่มีค่าอยู่ระหว่าง 0 กับ 1

การสร้างการแจกแจงปกติ $N(\mu,\sigma^2)$:

การผลิตเลขสุ่มที่มีการแจกแจงแบบปกติใช้วิธีการของ Box Muller (1958) โดยผลิตเลขสุ่มที่มีการแจกแจงปกติมาตรฐาน : $N(0,1)$ พร้อมกัน 2 ค่าและแต่ละค่าเป็นอิสระกันโดยใช้ตัวผลิต (Generater) Z_1 และ Z_2 พิจารณาดังรูปต่อไปนี้



พิจารณาจากรูปจะได้

$Z_1 = B\cos(\theta)$ (1)

$Z_2 = B\sin(\theta)$ (2)

เนื่องจาก $B^2 = Z_1^2 + Z_2^2$ มีการแจกแจงไคสแควร์ด้วยระดับความเป็นอิสระ 2 และเทียบเท่ากับการแจกแจงเอกซโพเนนเชียล ค่าเฉลี่ยเท่ากับ 2 โดยใช้วิธีการแปลงผกผัน (Inverse Transformation) สามารถสร้างเลขสุ่มที่มีการแจกแจงแบบเอกซโพเนนเชียลได้ดังนี้

$$B = (-2 \ln R)^{1/2} \qquad (3)$$

เมื่อ R เป็นเลขสุ่มที่มีการแจกแจงแบบสม่ำเสมอในช่วง (0,1)

จากการสมมาตรของการแจกแจงปกติ (Normal Distribution) จะได้ว่ามุม θ มีการแจกแจงสม่ำเสมอระหว่าง 0 ถึง 2π เรเดียน และรัศมี B กับมุม θ เป็นอิสระกัน จาก (1),(2),(3) เราสามารถสร้างเลขสุ่มที่มีการแจกแจงปกติมาตรฐาน จากเลขสุ่ม 2 ชุด R_1 และ R_2 กล่าวคือ

$$Z_1 = (-2 \ln R_1)^{1/2} \cos(2\pi R_2)$$

$$Z_2 = (-2 \ln R_1)^{1/2} \sin(2\pi R_2)$$

ซึ่ง R1 และ R2 เป็นตัวเลขสุ่มที่สร้างจากฟังก์ชัน FUNCTION RAND(IX) เมื่อได้ตัวเลขสุ่มที่มีการแจกแจงปกติมาตรฐานแล้ว จะทำการแปลงค่าเลขสุ่มดังกล่าวโดยอาศัยฟังก์ชัน

$$GN_1 = \mu + \sigma Z_1$$

$$GN_2 = \mu + \sigma Z_2$$

ซึ่งจะได้ว่า GN_1 และ GN_2 มีการแจกแจงแบบปกติด้วยค่าเฉลี่ยเท่ากับ μ ค่าความแปรปรวนเท่ากับ σ^2 ($GN_i \sim N(\mu, \sigma^2)$; i=1,2)

โปรแกรมย่อยที่ใช้ในการสร้างเลขสุ่มให้มีการแจกแจงปกติที่มีค่าเฉลี่ยเท่ากับ μ และมีค่าความแปรปรวนเท่ากับ σ^2 คือ SUBROUTINE NORMAL(RMEAN,VAR,GN) รายละเอียดโปรแกรมย่อยแสดงดังนี้

```
SUBROUTINE NORMAL(RMEAN,VARE,GN)
COMMON/SEED/IX,KKIX
SD = SQRT(VARE)
PI = 3.1415926
IF (KKIX .EQ.1) GOTO 100
RONE = RAND(IX)
RTWO = RAND(IX)
ZONE = SQRT(-2*ALOG(RONE))*COS(2*PI*RTWO)
ZTWO = SQRT(-2*ALOG(RONE))*SIN(2*PI*RTWO)
GN  = ZONE*SD+RMEAN
KKIX = 1
GOTO 300
```

```
100  GN   = ZTWO*SD+RMEAN
     KKIX = 0
300  RETURN
     END
```

การสร้างการแจกแจงสม่ำเสมอ U(a,b) :

ฟังก์ชันความหนาแน่น และฟังก์ชันการแจกแจงสะสม แสดงได้ดังนี้

$$f(c) = \frac{1}{b-a} \quad , \quad a<c<b$$

$$F(c) = \begin{cases} (c-a)/(b-a) & , a<c<b \\ 1 & , b>c \end{cases}$$

โดย a เป็นพารามิเตอร์กำหนดตำแหน่ง (Location Parameter)

$(b-a)$ เป็นพารามิเตอร์กำหนดขนาด (Scale Parameter)

มีค่าคาดหวังและความแปรปรวนของการแจกแจงคือ

$$E(c) = (b-a)/2$$

$$Var(c) = (b-a)^2/12$$

การสร้างตัวแปรสุ่มให้มีการแจกแจงแบบสม่ำเสมอ อาศัยการแปลงผกผันโดยมีขั้นตอนการสร้างดังนี้

ขั้นที่ 1 $F(c) = YFL$ โดยที่ YFL คือตัวเลขสุ่มแบบสม่ำเสมอช่วง (0,1)

ขั้นที่ 2 หาค่า c = ((b-a)*YFL) + a

โปรแกรมย่อยที่ใช้ในการสร้างตัวเลขสุ่มให้มีการแจกแจงแบบสม่ำเสมอในช่วง (aa,bb) คือ SUBROUTINE UNIFO(AA,BB,EX) รายละเอียดโปรแกรมย่อยแสดงดังนี้

```
SUBROUTINE UNIFO(AA,BB,EX)
COMMON/SEED/IX,KKIX
YFL= RAND(IX)
```

```
EX = ((BB-AA)*YFL)+AA
RETURN
END
```

การสร้างการแจกแจงแบบดับเบิลเอกซโพเนนเชียล DB (μ,σ):

การแจกแจงแบบดับเบิลเอกซโพเนนเชียล มีฟังก์ชันความหนาแน่นในรูป

$$f(x) = \frac{1}{2\sigma}\exp\left(-|x-\mu|/\sigma\right) \quad ; \ -\infty < x < \infty$$

มีค่าคาดหวังและความแปรปรวนของการแจกแจงคือ

$$E(x) = \mu$$
$$V(x) = 2\sigma^2$$

การสร้างตัวแปรสุ่มให้มีการแจกแจงแบบดับเบิลเอกซโพเนนเชียล อาศัยการแปลงผกผัน โดยมีขั้นตอนการสร้างดังนี้

ขั้นที่ 1 cdf. เขียนเป็น

$$F(x) = \begin{cases} \frac{1}{2}\exp\left((x-\mu)/\sigma\right) & , \ x < 0 \\ 1 - \frac{1}{2}\exp\left(-(x-\mu)/\sigma\right) & , \ x > 0 \end{cases}$$

ขั้นที่ 2 $F(x)$ = *YFL* โดยที่ *YFL* คือตัวเลขสุ่มแบบสม่ำเสมอช่วง (0,1)

ขั้นที่ 3 หาค่า x ได้ดังนี้

$$x = \sigma\left[\ln 2 + \ln(YFL)\right] + \mu \quad ; \ x<0$$
$$x = \mu - \sigma\left[\ln 2 + \ln(1-YFL)\right] \quad ; \ x>0$$

โปรแกรมย่อยที่ใช้ในการสร้างตัวเลขสุ่มให้มีการแจกแจงแบบดับเบิลเอกซโปเนนเชียลคือ คือ FUNCTION DBEX(RMEAN,VARE) รายละเอียดแสดงดังนี้

```
      FUNCTION DBEX(RMEAN,VARE)
      COMMON/SEED/IX,KK
      SD = VARE/2.
      BETA= SQRT(SD)
      YFL = RAND(IX)
      IF (YFL-0.5) 10,10,11
10    DBEX =(BETA*(ALOG(2.)+ALOG(YFL))+RMEAN
      GOTO 15
11    YFL = ALOG(2.)+ALOG(1.-YFL)
      DBEX = -1.*BETA*YFL+RMEAN
15    RETURN
      END
```

การสร้างการแจกแจงแบบล็อกนอร์มอล $LN(\mu,\sigma^2)$

ฟังก์ชันความหนาแน่น แสดงได้ดังนี้

$$f(x) = \frac{1}{x\sigma\sqrt{2\pi}}\exp\left[-\frac{(\ln(x)-\mu)^2}{2\sigma^2}\right] ;x>0,-\infty<\mu<\infty,\sigma>0$$

มีค่าคาดหวังและความแปรปรวนของการแจกแจงคือ

$$E(x) = e^{\mu+\sigma^2/2}$$
$$V(x) = e^{2\mu+\sigma^2/2}\left(e^{\sigma^2}-1\right)$$

การสร้างตัวแปรสุ่มให้มีการแจกแจงแบบล็อกนอร์มอล อาศัยการแปลงผกผันโดยมีขั้นตอนการสร้างดังนี้

ขั้นที่ 1 สร้างตัวแปรสุ่ม GN ให้มีการแจกแจงเป็นแบบปกติด้วยค่าเฉลี่ยเท่ากับ μ และความแปรปรวนเท่ากับ σ^2

ขั้นที่ 2 หา x = exp(GN)

โปรแกรมย่อยที่ใช้ในการสร้างเลขสุ่ม GX ให้มีการแจกแจงแบบล็อกนอร์มอลด้วยพารามิเตอร์ μ และ σ^2 คือ SUBROUTINE LN(RMEAN,VARE,GX) รายละเอียดโปรแกรมย่อยแสดงดังนี้

```
      SUBROUTINE LN(RMEAN,VARE,GX)
      COMMON/SEED/IX,KKIX
      SD = SQRT(VARE)
      PI = 3.1415926
      IF (KKIX .EQ.1) GOTO 100
      RONE = RAND(IX)
      RTWO = RAND(IX)
      ZONE = SQRT(-2*ALOG(RONE))*COS(2*PI*RTWO)
      ZTWO = SQRT(-2*ALOG(RONE))*SIN(2*PI*RTWO)
      GN  = ZONE*SD+RMEAN
      KKIX = 1
      GOTO 300
  100 GN  = ZTWO*SD+RMEAN
      KKIX = 0
  300 GX=EXP(GN)
C     GLN=ALOG(GX)
      RETURN
      END
```

การสร้างการแจกแจงแบบไวบูลล์ (Weibull Distribution)

ฟังก์ชันความหนาแน่น และฟังก์ชันการแจกแจงสะสม แสดงได้ดังนี้

$$f(x) = \alpha\beta^{-\alpha}x^{\alpha-1}\exp\left(-(x/\beta)^{\alpha}\right) \quad ; x>0$$

$$F(x) = \begin{cases} 1-\exp\left(-(x/\beta)^{\alpha}\right) & ; x>0 \\ 0 & ; \text{อื่นๆ} \end{cases}$$

มีค่าคาดหวังและความแปรปรวนของการแจกแจงคือ

$$E(x) = \frac{\beta}{\alpha}\Gamma(1/\alpha)$$

$$V(x) = \frac{\beta^2}{\alpha}\left\{2\Gamma\left(\frac{2}{\alpha}\right) - \frac{1}{\alpha}\left(\Gamma\left(\frac{1}{\alpha}\right)\right)^2\right\}$$

การสร้างตัวแปรสุ่มให้มีการแจกแจงแบบไวบูลล์ อาศัยการแปลงผกผันโดยมีขั้นตอนการสร้างดังนี้

ขั้นที่ 1 $F(x) = YFL$ โดยที่ YFL คือตัวเลขสุ่มแบบสม่ำเสมอช่วง (0,1)

ขั้นที่ 2 หาค่า $x = \beta(-\ln(1-YFL))^{(1/\alpha)}$

โปรแกรมย่อยที่ใช้ในการสร้างตัวเลขสุ่มให้มีการแจกแจงแบบไวบูลล์ด้วยพารามิเตอร์ β, α คือ SUBROUTINE WEIBUL(WLAM,WGAM,TWEI) ซึ่งมีรายละเอียดดังนี้

```
      SUBROUTINE WEIBUL(ALPHA,BETA,TWEI)
      COMMON/SEED/IX,KKIX
   10 YFL  = RAND(IX)
      IF ((YFL .EQ.1.0) .OR. (YFL .EQ.0.0)) GOTO 10
      TWEI = BETA*(-ALOG(1.0-YFL))**(1.0/ALPHA)
      RETURN
      END
```

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ข

```
C*-------------------------------------------------------------*
C*----                    MAIN PROGRAM                     ----*
C*----  ESTIMATING PARAMETERS IN MULTIPLE LINEAR REGRESSION WITH  ----*
C*----          TYPE 1 RIGHT CENSORED VARIABLE             ----*
C*-------------------------------------------------------------*
      DIMENSION X(100,5),Y(100),YI(100),EI(100),YCAL(100),
     *     YH1(100),YH2(100),YH3(100),YH4(100),
     *    NDELTA(100),YTC(4),BETA(2),
     *     BH1(5),BH2(5),BH3(5),BH4(5)
     *     ,BRAPE1(5),BRAPE2(5),BMEAN1(5),BMEAN2(5)
     *     ,BRAPE3(5),BRAPE4(5),BMEAN3(5),BMEAN4(5)
      COMMON/SEED/IX,KKIX
     *   /INTT/NM,NL,NCOM,NCEN,TC,NV
     *   /DIM1/X,Y
     *   /DIM2/YI,NDELTA
      DATA ALPHA,BETA/2.0,3.5,3.5/
      DATA XA,XB,XMUE,XSIG2,EALPHA,EBETA/2.,2.,10.,12.,1.0,5.0/
      DATA EMEAN,VARE/0.,100.0/
      DATA ELN,VLN/1.,1.1025/
      DATA YTC/60.,55.,50.,45./
      DATA AMEAN3,AMEAN4,ARAPE3,ARAPE4/0.,0.,0.,0./
C
      NM      = 20
      PERC    = 10.
      TC      = 50.
      NL      = 1000
      NV      = 2
      NCEN    = NM*PERC/100
      NCOM    = NM-NCEN
      NCPERC  = PERC/10
      IX      = 65479
      KKIX    = 0
      NV2     = NV
      NM2     = NM
      NU      = 0
      NC      = 0
      INDEX1  = 1
      INDEX2  = 1
      INDEX3  = 1
      NLOOPX  = 0
```

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

```
      DO 19 J=1,NM
      CALL WEIBUL(XA,XB,X1GEN)
      CALL NORMAL(XMUE,XSIG2,X2GEN)
      X(J,1) = X1GEN
      X(J,2) = X2GEN
      YCAL(J)= ALPHA+ BETA(1)*X1GEN+BETA(2)*X2GEN
      IF ( YCAL(J) .GE. TC ) NX=NX+1
   19 CONTINUE
C*****************************************************
      DO 99 LOOP=1,NL
      NIL= 0
  199 I = 0
      DO 10 IT =1 ,NM
      Y(IT) =0.
      YI(IT)=0.
      NDELTA(IT)=0
      YH1(IT)=0.
      YH2(IT)=0.
      YH3(IT)=0.
      YH4(IT)=0.
   10 CONTINUE
C     **-------------------------------**
      IL  = 0
      NC = 0
      NU = 0
      NT = 0
    1 I  =I+1
      SUM=0.
      IL  = 0
   30 CALL NORMAL(EMEAN,VARE,EGEN)
C30 CALL LN(ELN,VLN,EGEN1)
C     EGEN = ALOG(EGEN1)
C30 EGEN = DREX(EMEAN,VARE)
      YI(I)= YCAL(I)+EGEN.
      IL  =IL+1
C     **-------------------------**
      IF ((NC .LT. NCEN) .AND. (YI(I) .GT. TC)) THEN
        NC    =NC+1
        NT    =NC+NU
        Y(NT) = TC
        EI(NT) = EGEN
        NDELTA(NT)= 0
      ELSE
```

```
      IF ((NU .LT. NCOM) .AND. (YI(I) .LE. TC)) THEN
      NU   = NU+1
           NT   = NC+NU
           Y(NT) = YI(I)
           EI(NT) = EGEN
           NDELTA(NT) = 1
      ELSE
           GOTO 30
      ENDIF
      IL=0
      ENDIF
60    IF ((NC .EQ. NCEN) .AND. (NU .EQ. NCOM)) GOTO 70
      GOTO 1
C     **-------------------------------------------------------------**
C     STOP GENERATING VALUES AND START TO CALCULATE FOUR METHODS
C     **-------------------------------------------------------------**
 70   CALL BJ(BH1,AH1,SH1,YH1,LOOP,INDEX1)
      IF (INDEX1 .EQ. 0) GOTO 199
      CALL CM(BH2,AH2,SH2,YH2,LOOP,INDEX2)
      IF (INDEX2 .EQ. 0)  GOTO 199
      CALL MLE(BH3,AH3,SH3,YH3,BH4,AH4,SH4,YH4,LOOP,INDEX3)
      IF (INDEX3 .EQ. 0)  GOTO 199
C
      CALL MSE(NC,NU,NT,YH1,RMSEC1,RMSEU1,RMSET1)
      CALL MEANV(LOOP,RMSEC1,RMC21,VRMC21,RMC11,VRMC11,CVC1)
      CALL MEANV(LOOP,RMSEU1,RMU21,VRMU21,RMU11,VRMU11,CVU1)
      CALL MEANV(LOOP,RMSET1,RMT21,VRMT21,RMT11,VRMT11,CVT1)
C
      CALL MSE(NC,NU,NT,YH2,RMSEC2,RMSEU2,RMSET2)
      CALL MEANV(LOOP,RMSEC2,RMC22,VRMC22,RMC12,VRMC12,CVC2)
      CALL MEANV(LOOP,RMSEU2,RMU22,VRMU22,RMU12,VRMU12,CVU2)
      CALL MEANV(LOOP,RMSET2,RMT22,VRMT22,RMT12,VRMT12,CVT2)
C
      CALL MSE(NC,NU,NT,YH3,RMSEC3,RMSEU3,RMSET3)
      CALL MEANV(LOOP,RMSEC3,RMC23,VRMC23,RMC13,VRMC13,CVC3)
      CALL MEANV(LOOP,RMSEU3,RMU23,VRMU23,RMU13,VRMU13,CVU3)
      CALL MEANV(LOOP,RMSET3,RMT23,VRMT23,RMT13,VRMT13,CVT3)
C     **-------------------------------------------------------**
      CALL MSE(NC,NU,NT,YH4,RMSEC4,RMSEU4,RMSET4)
      CALL MEANV(LOOP,RMSEC4,RMC24,VRMC24,RMC14,VRMC14,CVC4)
      CALL MEANV(LOOP,RMSEU4,RMU24,VRMU24,RMU14,VRMU14,CVU4)
      CALL MEANV(LOOP,RMSET4,RMT24,VRMT24,RMT14,VRMT14,CVT4)
```

```
C  **---------- BIASED OF PARAMETERS ----------**
      DO 77 IB=1,NV
      BRAPE3(IB) =BRAPE3(IB)+ABS(BH3(IB)-BETA(IB))/BETA(IB)
      BRAPE4(IB) =BRAPE4(IB)+ABS(BH4(IB)-BETA(IB))/BETA(IB)
      BMEAN3(IB) =BMEAN3(IB)+BH3(IB)
      BMEAN4(IB) =BMEAN4(IB)+BH4(IB)
 77   CONTINUE
      ARAPE3   =ARAPE3+ABS(AH3-ALPHA)/ALPHA
      ARAPE4   =ARAPE4+ABS(AH4-ALPHA)/ALPHA
      AMEAN3   =AMEAN3+AH3
      AMEAN4   =AMEAN4+AH4
 99   CONTINUE
C
999   LOOP=LOOP-1
      DO 78 IB=1,NV
      BRAPE3(IB)=BRAPE3(IB)/(LOOP)
      BRAPE4(IB)=BRAPE4(IB)/(LOOP)
      BMEAN3(IB)=BMEAN3(IB)/(LOOP)
      BMEAN4(IB)=BMEAN4(IB)/(LOOP)
 78   CONTINUE
      ARAPE3  =ARAPE3/(LOOP)
      ARAPE4  =ARAPE4/(LOOP)
      AMEAN3  =AMEAN3/(LOOP)
      AMEAN4  =AMEAN4/(LOOP)
C----------          OUTPUT     ----------
      WRITE(6,150)
150   FORMAT(8X,'NM',2X,'%',3X,'TC',7X,'RMSE-CEN',5X,RMSE-UN',5X,'RMSE-TOTAL',5X,
     *'CV  CEN   UNCEN TOTAL')
      WRITE(6,200)
      WRITE(6,181) NM,PERC,TC,RMC21,RMU21,RMT21,CVC1,CVU1,CVT1
      WRITE(6,182) NM,PERC,TC,RMC22,RMU21,RMT22,CVC2,CVU2,CVT2
      WRITE(6,183) NM,PERC,TC,RMC24,RMU21,RMT24,CVC4,CVU4,CVT3
      WRITE(6,184) NM,PERC,TC,RMC23,RMU21,RMT23,CVC3,CVU3,CVT4
      WRITE(6,201)
      WRITE(6,*) '              RAPE                MEAN'
      WRITE(6,186) ARAPE1,BRAPE1(1),BRAPE1(2),AMEAN1,BMEAN1(1),BMEAN1(2)
      WRITE(6,187) ARAPE2,BRAPE2(1),BRAPE2(2),AMEAN2,BMEAN2(1),BMEAN2(2)
      WRITE(6,188) ARAPE4,BRAPE4(1),BRAPE4(2),AMEAN4,BMEAN4(1),BMEAN4(2)
      WRITE(6,189) ARAPE3,BRAPE3(1),BRAPE3(2),AMEAN3,BMEAN3(1),BMEAN3(2)
      WRITE(6,200)
180   FORMAT(2X,'BJ ',1X,I3,1X,F4.0,1X,F4.0,1X,F9.4,1X,F10.4,1X,F10.4,
     *,1X,F7.4,1X,F7.4,1X,F7.4)
```

```
181 FORMAT(2X,'BJ ',1X,I3,1X,F4.0,1X,F4.0,1X,F9.4,1X,F10.4,1X,F10.4,
   *,1X,F7.4,1X,F7.4,1X,F7.4)
182 FORMAT(2X,'CM ',1X,I3,1X,F4.0,1X,F4.0,1X,F9.4,1X,F10.4,1X,F10.4,
   *,1X,F7.4,1X,F7.4,1X,F7.4)
183 FORMAT(2X,'OLS',1X,I3,1X,F4.0,1X,F4.0,1X,F9.4,1X,F10.4,1X,F10.4,
   *,1X,F7.4,1X,F7.4,1X,F7.4)
184 FORMAT(2X,'MLE',1X,I3,1X,F4.0,1X,F4.0,1X,F9.4,1X,F10.4,1X,F10.4,
   *,1X,F7.4,1X,F7.4,1X,F7.4)
186 FORMAT(2X,'BJ ',16X,F7.4,1X,F7.4,1X,F7.4,5X,F10.4,2X,F10.4,2X,F10.4)
187 FORMAT(2X,'CM ',16X,F7.4,1X,F7.4,1X,F7.4,5X,F10.4,2X,F10.4,2X,F10.4)
188 FORMAT(2X,'OLS',16X,F7.4,1X,F7.4,1X,F7.4,5X,F10.4,2X,F10.4,2X,F10.4)
189 FORMAT(2X,'MLE',16X,F7.4,1X,F7.4,1X,F7.4,5X,F10.4,2X,F10.4,2X,F10.4)

9   WRITE(6,197) ALPHA,BETA(1),BETA(2)
197 FORMAT(5X,'ALPHA =',F6.2,2X,'BETA1 =',F6.2,2X,'BETA2=',F6.2)
    WRITE(6,198) LOOP,VARE,IX
198 FORMAT(5X,'END LOOP=',I4,2X,'ERROR VAR. =',F6.1,2X,'SEED =',I8)
200 FORMAT(110('-'))
201 FORMAT(15X,60('-'))
    STOP
    END
C   ***********************************************
C   *******    STOP MAIN PROGRAMME     ******
C   ***********************************************
    SUBROUTINE BJ(BETA,ALPHA,SIGMA,YH,LOOP,INDEX)
    DIMENSION X(100,5),Y(100),NDELTA(100),YSTAR(100),YI(100),
   *    XU(100,5),YU(100),YH(100),IDE(100),
   *    YCMEW(100),YUMEW(100),SU(100),SC(100),EU(100),EC(100),
   *    ICRANK(100),IRC(100),IRU(100),VK(100),WIK(100),E(100),
   *    BK(3,5),AK(3),SK(3),EM(5),BETA(5)
    DOUBLE PRECISION YH1,SC,SU,SUM,VK
    COMMON/SEED/IX,KKIX
   *  /INTT/NM,NL,NCOM,NCEN,TC,NV
   *  /DIM1/X,Y
   *  /DIM2/YI,NDELTA
C
C   ESTIMATE INITIAL PARAMETER
C
    KNO=50
    K=1
    NU=0
    NC=0
    DO 5 JK = 1,NM
```

```
      IF (NDELTA(JK) .EQ. 0) GOTO 5
      NU= NU+1
      DO 3 I = 1,NV
      XU(NU,I)= X(JK,I)
    3 CONTINUE
      YU(NU) = Y(JK)
      IDB(NU)= NDELTA(JK)
    5 CONTINUE
C   *-------------------------------------
      NM2 = NU
      CALL LS(XU,YU,NM2,BM,ALFA,S)
      OLSA =ALFA
      OLSB1=BM(1)
      OLSB2=BM(2)
      OLSS =S
      CALL TRANP(BM,ALFA,S,BK,AK,SK)
C     **********************************************
C     CALCULATE EI AT ALPHA =0 FOR  ALL OBSERVATIONS
C     START ITERLATION
C     **********************************************
   20 K = K+1
      KK= K-1
      DO 12 JK = 1,NM
      YH1=0.
      DO 10 J =1,NV
      YH1 = YH1+BM(J)*X(JK,J)
   10 CONTINUE
      E(JK)= ABS(Y(JK)-YH1)
   12 CONTINUE
C
C     FIND UNCENSORED AND CENSORED
C
      NU =0
      NC =0
      DO 14 I=1,NM
      IF (NDELTA(I) .EQ. 1) THEN
      NU = NU+1
      EU(NU)=E(I)
      IRU(NU)=I
      ELSE
      NC=NC+1
      EC(NC) =E(I)
      IRC(NC)=I
```

```
      ENDIF
   14 CONTINUE
C
C     FIND PL ESITMATOR BASED ON UNCENSORED OBSERVATIONS
C
      CALL SORT2(EU,NU,IRU)
      CALL SORT2(EC,NC,IRC)
C     IF(EC(NC) .GT. EU(NU)) THEN
C     NU =NU+1
C     EU(NU)=EC(NC)
C     IRU(NU)=IRC(NC)
C     NC=NC-1
C     ENDIF
C
      CALL PL(EU,EC,NU,NC,SU)
      CALL PL(EC,EU,NC,NU,SC)
C     ------------------------------------------
C     WRITE(6,*) '     EU       SU       EC       SC'
C     ------------------------------------------
      DO 16 I=1,NM
      ICRANK(I)=0
      IF (NDELTA(I) .EQ. 1) GOTO 16
      DO 15 J=1,NC
      IF (E(I) .EQ. EC(J)) ICRANK(I)=J
   15 CONTINUE
   16 CONTINUE
C
C     FIND BETA*X=YUMEW ---UNCENSORED OBSERVATION
C
      DO 19 I=1,NU
      SUM=0.
      YUMEW(I)=0.
      DO 18 II=1,NV
      SUM=SUM+BM(II)*X(IRU(I),II)
   18 CONTINUE
      YUMEW(I)= SUM
   19 CONTINUE
      CALL MASS(SU,NU,VK)
C103 CONTINUE
      DO 40 I=1,NM
      IF (NDELTA(I) .EQ. 1) THEN
      YSTAR(I)=Y(I)
      ELSE
```

```
      SUM=0.
      YCMEW(I)=0.
      DO 30 IJ = 1,NV
      SUM=SUM+BM(IJ)*X(I,IJ)
 30   CONTINUE
      YCMEW(I)=SUM
      SUM=0.
      DO 35 IU=1,NU
      IF (SC(ICRANK(I)) .GT. 0.) THEN
      WIK(IU)=VK(IU)/SC(ICRANK(I))
      ELSE
      WIK(IU)=0.
      ENDIF
      SUM=SUM+WIK(IU)*(Y(IRU(IU))-YUMEW(IU))
 35   CONTINUE
      YSTAR(I)=YCMEW(I)+SUM
      ENDIF
 40   CONTINUE
C     #
      DO 42 IJ =1,NM
      DO 41 IJJ=1,NV
      XU(IJ,IJJ)=X(IJ,IJJ)
 41   CONTINUE
 42   CONTINUE
      NM2=NM
      CALL LS(XU,YSTAR,NM2,BM,ALFA,S)
C     #
C     **SIGMA ** BJ** NONPARA **
C     #
      SUM=0.
      DO  45 I1=1,NU
      SUM = SUM+EU(I1)
 45   CONTINUE
      EUBAR= SUM/NU
      SUM  = 0.
      DO 47 IV=1,NU
      SUM = SUM+(EU(IV)-EUBAR)**2
47    CONTINUE
      SBJ = SUM/(NU-NV-1)
      CALL TRANP(BM,ALFA,SBJ,BK,AK,SK)
C     #
C     WRITE(6,46) K,AK(3),BK(3,1),BK(3,2),SK(3),S
C46   FORMAT(2X,I3,3X,F12.6,3X,F12.6,3X,F12.6,3X,F12.6,2X,F10.6)
```

```
      IF (K .GE. 3) THEN
      IF (K .GT. KNO) GOTO 59
       KK = 2
      CALL DIFP(BK,AK,SK,KK,IND)
      IF  (IND .EQ. 0) GOTO 50
      INDEX=1
      GOTO 60
50    KK=1
      CALL DIFP(BK,AK,SK,KK,IND)
      IF (IND .EQ. 0)  GOTO 20
      INDEX=1
      GOTO 60
      ELSE
      KK=2
      CALL DIFP(BK,AK,SK,KK,IND)
      IF (IND .EQ. 0)  GOTO 20
      INDEX=1
      GOTO 60
     ENDIF
C     **-------------------------------------**
59    INDEX=0
      IF (INDEX .EQ. 0) GOTO 191
C     **---------- PARAMETER ----------**
60    ALPHA = (AK(3)+AK(KK))/2.
      SIGMA = (SK(3)+SK(KK))/2.
      DO 65 I2=1,NV
      BETA(I2)=(BK(3,I2)+BK(KK,I2))/2.
65    CONTINUE
C     WRITE(6,92) LOOP,ALPHA,BETA(1),BETA(2),SIGMA
C92   FORMAT(3X,I4,2X,F12.4,3X,F12.4,3X,F12.4,3X,F12.4)
C     **---------- FIND Y ESTIMATED ----------**
      DO 71 I=1,NM
      SUM=0.
      DO 70 J=1,NV
      SUM = SUM+BETA(J)*X(I,J)
 70   CONTINUE
      YH(I)=ALPHA+SUM
 71   CONTINUE
191   RETURN
      END
```

```
C  ***********************************************
C  *****              SUBROUTINE CM              *****
C  ***********************************************
      SUBROUTINE CM(BETA,ALPHA,SIGMA,YH,LOOP,INDEX)
      DIMENSION BETA(5),BM(5),BK(3,5),AK(3),SK(3),
     *     YH(100),SIG(100),
     *     XU(100,5),YU(100),NDEL(100),YSTAR(100),
     *     X(100,5),Y(100),YI(100),NDELTA(100),IDEL(100)
      COMMON/INTT/NM,NL,NCOM,NCEN,TC,NV
     *  /DIM1/X,Y
     *  /DIM2/YI,NDELTA
      K       =1
      IN      =0
      SUM1    =0.
      SUM2    =0.
      NU      =0
      NC      =0
      KNO     =50
C  **--- UNCENSORED DATA ---**
      DO 5 I=1,NM
      IF (NDELTA(I).EQ. 0 ) GOTO 5
      NU=NU+1
      DO 3 J=1,NV
      XU(NU,J)=X(I,J)
3     CONTINUE
      YU(NU)=Y(I)
C     IDEL(NU)=NDELTA(I)
5     CONTINUE
      NM2=NU
C     **-----------------**
      CALL LS(XU,YU,NM2,BM,ALFA,S)
      CALL TRANP(BM,ALFA,S,BK,AK,SK)
      DO 1 I=1,NM
      DO 2 J=1,NV
      XU(I,J)=X(I,J)
 2    CONTINUE
 1    CONTINUE
C     ---------------------------------------
 20   K  =K+1
C     KK =K-1
      DO 40 I = 1,NM
      SUM = 0.
      IF (NDELTA(I) .EQ. 0) THEN
```

```
      DO 15 I3 = 1,NV
      SUM = SUM + X(I,I3)*BM(I3)
15     CONTINUE
      Y2 = ALFA+SUM
      IF (Y2 .LE. TC ) THEN
      YSTAR(I)=TC
      ELSE
      YSTAR(I)=Y2
      ENDIF
     ELSE
      YSTAR(I)=Y(I)
      ENDIF
140  CONTINUE
      NM2= NM
      CALL LS(XU,YSTAR,NM2,BM,ALFA,S)
      CALL TRANP(BM,ALFA,S,BK,AK,SK)
      IF (K .GE. 3) THEN
      IF (K .GT. KNO) GOTO 59
      KK = 2
      CALL DIFP(BK,AK,SK,KK,IND)
      IF  (IND .EQ. 0) GOTO 50
      INDEX=1
      GOTO 60
50     KK=1
      CALL DIFP(BK,AK,SK,KK,IND)
      IF (IND .EQ. 0)  GOTO 20
      INDEX=1
      GOTO 60
     ELSE
      KK=2
      CALL DIFP(BK,AK,SK,KK,IND)
      IF (IND .EQ. 0)  GOTO 20
      INDEX=1
      GOTO 60
     ENDIF
C    **-------------------------------------**
59   INDEX=0
     IF (INDEX .EQ. 0) GOTO 191
C    **---------- PARAMETER ----------**
60   ALPHA = (AK(3)+AK(KK))/2.
     SIGMA = (SK(3)+SK(KK))/2.
     DO 65 I2=1,NV
      BETA(I2)=(BK(3,I2)+BK(KK,I2))/2.
```

```
 65   CONTINUE
C     WRITE(6,92) LOOP,ALPHA,BETA(1),BETA(2),SIGMA
C92   FORMAT(1X,'LAM',I4,2X,F12.4,3X,F12.4,3X,F12.4,3X,F12.4)
C     **---------- FIND Y ESTIMATED ----------**
      DO 71 I=1,NM
       SUM=0.
       DO 70 J=1,NV
       SUM = SUM+BETA(J)*X(I,J)
  70   CONTINUE
       YH(I)=ALPHA+SUM
  71  CONTINUE
 191  RETURN
       END
C     ************************************************
       SUBROUTINE MLE(BETA,ALPHA,SIGMA,YH1,BLS,ALS,SLS,YH2,LOOP,INDEX)
       DIMENSION  BETA(5),BM(5),BK(3,5),AK(3),SK(3),
     *  Z(100),FZ(100),FXZ(100),HZ(100),W(100),AM(100),
     *  X(100,5),Y(100),YI(100),NDELTA(100),BLS(5),
     *  IDEL(100),XU(100,5),YU(100),YSTAR(100),YH1(100),YH2(100)
       DOUBLE PRECISION ZI
       COMMON/INTT/NM,NL,NCOM,NCEN,TC,NV
     *   /DIM1/X,Y
     *   /DIM2/YI,NDELTA
       K   = 1
       INDEX= 1
       SUM1 = 0.
       SUM2 = 0.
       NU   = 0
       NC   = 0
       KNO  = 50
C      **------- ALL DATA -----**
       DO 5 I=1,NM
       DO 3 J=1,NV
       XU(I,J)=X(I,J)
3      CONTINUE
       YSTAR(I)=Y(I)
5      CONTINUE
       NM2=NM
C      **-------------------**
       CALL LS(XU,YSTAR,NM2,BM,ALFA,S)
       CALL TRANP(BM,ALFA,S,BK,AK,SK)
       ALS   = ALFA
       BLS(1) = BM(1)
```

```
      BLS(2) = BM(2)
      SLS  = S
C     ------------------------------------------
 20   K  = K+1
C     KK  = K-1
      SUM  = 0.
      SUM1 = 0.
      SUM2 = 0.
C
      DO 8 I1=1,NM
      IF (NDELTA(I1) .EQ. 0) GOTO 8
      SUM=0.
      DO 7 I2=1,NV
      SUM=SUM+BM(I2)*X(I1,I2)
 7    CONTINUE
      AM(I1)=ALFA+SUM
      SUM1=SUM1+(Y(I1)-AM(I1))**2
 8    CONTINUE
C
      DO 40 I = 1,NM
      IF (NDELTA(I) .EQ. 0) THEN
      SUM=0.
      DO 15 IJ = 1,NV
      SUM=SUM+BM(IJ)*X(I,IJ)
 15   CONTINUE
      AM(I) = ALFA+SUM
      Z(I)  = (Y(I)-AM(I))/SQRT(SK(3))
C     ------------  FIND F(Z)  ------------
      IF (Z(I) .GE. 0.) THEN
       IF (Z(I) .GE. 4.) THEN
         FZ(I) = 0.
         FXZ(I)= 1.
       ELSE
         FZ(I) = EXP(-Z(I)*Z(I)/2.)/2.506628275
         ZI  = Z(I)
C         CALL PP(ZI,PXZ1)
         PXZ1  = CDFN(ZI)
         FXZ(I)= PXZ1
       ENDIF
      ELSE
       IF (Z(I).LE.-4.) THEN
         FZ(I)  = 0.
         FXZ(I) = 0.
```

```
        ELSE
          FZ(I) = EXP(-Z(I)*Z(I)/2.)/2.506628275
          ZI  = Z(I)
C          CALL PP(ZI,PXZ1)
          PXZ1 = CDFN(ZI)
          FXZ(I)= PXZ1
        ENDIF
       ENDIF
C     ------------ FIND HZ(I) ------------
      IF ((FZ(I) .EQ. 0.) .OR. (1.-FXZ(I) .EQ. 0.)) THEN
        HZ(I)=0.
        W(I) =AM(I)
      ELSE
        HZ(I)=(FZ(I)/(1.-FXZ(I)))
        W(I) =AM(I)+SQRT(SK(3))*HZ(I)
      ENDIF
C     ---------- ESTIMATE CENSORED DATA ----------
      YSTAR(I)=W(I)
      SUM2 = SUM2 +(1.+Z(I)*HZ(I))
      ENDIF
 40   CONTINUE
      NM2=NM
C     ---------- SIGMA SQARE(K+1) AND BETA(K+1) ------
      CALL LS(XU,YSTAR,NM2,BM,ALFA,S)
      SUM2=SUM2*SK(3)
      S=(SUM1+SUM2)/NM
      CALL TRANP(BM,ALFA,S,BK,AK,SK)
C
      IF (K .GE. 3) THEN
      IF (K .GT. KNO) GOTO 59
      KK = 2
      CALL DIFP(BK,AK,SK,KK,IND)
      IF  (IND .EQ. 0) GOTO 50
      INDEX=1
      GOTO 60
 50   KK=1
      CALL DIFP(BK,AK,SK,KK,IND)
      IF (IND .EQ. 0)  GOTO 20
      INDEX=1
      GOTO 60
      ELSE
      KK=2
      CALL DIFP(BK,AK,SK,KK,IND)
```

```
      IF (IND .EQ. 0) GOTO 20
      INDEX=1
      GOTO 60
      ENDIF
C     **------ ITERATION DIVERGE ----------**
59    INDEX=0
C     **---------- PARAMETER ----------**
60    ALPHA = (AK(3)+AK(KK))/2.
      SIGMA = (SK(3)+SK(KK))/2.
      DO 65 I2=1,NV
      BETA(I2)=(BK(3,I2)+BK(KK,I2))/2.
65    CONTINUE
      IF (ALPHA   .GE. 30.0)  INDEX =0
      IF ((BETA(1) .LE. 0.) .OR. (BETA(1) .GE. 7.0 )) INDEX =0
      IF ((BETA(2) .LE. 0.) .OR. (BETA(2) .GE. 7.0 )) INDEX =0
      IF ( INDEX .EQ. 0) GOTO 191
C     **---------- FIND Y ESTIMATED ----------**
      DO 71 I=1,NM
      SUM=0.
      SUML=0.
      DO 70 J=1,NV
      SUM = SUM+BETA(J)*X(I,J)
      SUML= SUML+BLS(J)*X(I,J)
 70   CONTINUE
      YH1(I)= ALPHA+SUM
      YH2(I)= ALS+SUML
 71   CONTINUE
191   RETURN
      END
C     ***************************************************
      SUBROUTINE LS(XNEW,YNEW,NMNEW,BETA,ALFA,SIGMA)
      DIMENSION  XNEW(100,5),YNEW(100),YH(100),
     *      SXIXJ(5,5),SUMXIY(5),SUMXI(5),
     *      XTY(5),BETA(5)
      COMMON/INTT/NM,NL,NCOM,NCEN,TC,NV
      DOUBLE PRECISION XTXM(5,5),XINV(5,5),SUM2,SUM1,SUMY
      NVNEW=NV
      DO 96 I = 1,NVNEW
      DO 95 J = 1,NVNEW
      SUM = 0.
      DO 90 K=1,NMNEW
      SUM  = SUM +XNEW(K,I)*XNEW(K,J)
 90   CONTINUE
```

```
      SXIXJ(I,J) = SUM
   95 CONTINUE
   96 CONTINUE
C
      DO 115 I =1,NVNEW
      SUM1 = 0.
      SUM2 = 0.
      DO 110 K=1,NMNEW
      SUM1 = SUM1+XNEW(K,I)
      SUM2 = SUM2+XNEW(K,I)*YNEW(K)
  110 CONTINUE
      SUMXI(I) = SUM1
      SUMXIY(I)= SUM2
  115 CONTINUE
C
      SUMY =0.
      DO 120 K = 1,NMNEW
      SUMY = SUMY +YNEW(K)
  120 CONTINUE
C
      DO 122 I = 1,NVNEW
      DO 121 J = 1,NVNEW
      XTXM(I,J)= SXIXJ(I,J)-(SUMXI(I)*SUMXI(J))/NMNEW
  121 CONTINUE
  122 CONTINUE
      DO 125 J = 1,NVNEW
      XTY(J) = SUMXIY(J)-(SUMXI(J)*SUMY)/NMNEW
  125 CONTINUE
C
      CALL INV2(XTXM,XINV,NVNEW)
      DO 128 I =1,NVNEW
      SUM = 0.
      DO  126 J = 1,NVNEW
      SUM = SUM+XINV(I,J)*XTY(J)
  126 CONTINUE
      BETA(I)=SUM
  128 CONTINUE
      SUMXBA = 0.
      DO  131 I =1,NVNEW
      SUMXBA = SUMXBA +BETA(I)*(SUMXI(I)/NMNEW)
  131 CONTINUE
      ALFA = SUMY/NMNEW -SUMXBA
C     ******** SIGMA ********
```

```
      SUM2=0.
      DO 140 I = 1,NMNEW
      SUM1=0.
      YH(K)=0.
      DO 135 K = 1,NVNEW
      SUM1 = SUM1+BETA(K)*XNEW(I,K)
  135 CONTINUE
      YH(I)= ALFA+SUM1
      SUM2 = SUM2+(YNEW(I)-YH(I))**2
  140 CONTINUE
      SIGMA = SUM2/(NMNEW-NVNEW-1)
C     WRITE(6,*) ' SIGMA LS=',SIGMA
      RETURN
      END
C     ***********************************************
C     *** FUNCTION FIND VALUE OF STANDARD NORMAL ***
C     ***********************************************
      FUNCTION CDFN(ZO)
      DOUBLE PRECISION ZO
      TLZO = (1./2.5066282746)*EXP(-1.*((ZO**2)/2.))
      TLWW = 1.0/(1.0+0.33267*ABS(ZO))
      TLP  = 1.0-TLZO*(0.4361836*TLWW-0.1201676*(TLWW**2)
     *     +0.937298*(TLWW**3))
      IF (ZO.GE.0.0) THEN
      CDFN = TLP
      ELSE
      CDFN = 1.0- TLP
      ENDIF
      RETURN
      END
C     ***********************************************
      FUNCTION DBEX(RMEAN,VARE)
      COMMON/SEED/IX,KK
      SD = VARE/2.
      BETA= SQRT(SD)
      YFL = RAND(IX)
      IF (YFL-0.5) 10,10,11
   10 DBEX =BETA*(ALOG(2.)+ALOG(YFL))
      GOTO 15
   11 YFL = ALOG(2.)+ALOG(1.-YFL)
      DBEX =-1.*BETA*YFL
   15 RETURN
      END
```

```
C     ***********************************************
      SUBROUTINE DIFP(BK,AK,SK,KK,IND)
      DIMENSION BK(3,5),AK(3),SK(3),IND1(5)
      COMMON/INTT/NM,NL,NCOM,NCEN,TC,NV
      Z =.005
      DO 10 I=1,NV
      DIFB=ABS(BK(3,I)-BK(KK,I))
      IF (DIFB .LE. Z ) THEN
      IND1(I)=1
      ELSE
      IND1(I)=0
      ENDIF
 10   CONTINUE
      IND2=1
      DO 20 J=1,NV
      IND2=IND2*IND1(J)
 20   CONTINUE
      DIFA=ABS(AK(3)-AK(KK))
      DIFS=ABS(SK(3)-SK(KK))
      IF ((IND2 .EQ. 1 ) .AND. (DIFA .LE. Z)
     *  .AND. (DIFS .LE. Z)) THEN
      IND=1
      ELSE
      IND=0
      ENDIF
      RETURN
      END
C     ***********************************************
C     *****     SUBROUTINE INVERSE METRIX     ******
C     ***********************************************
      SUBROUTINE INV2(XTWX,OXTWX,N)
C     REAL IXTWX
      DOUBLE PRECISION XTWX(5,5),OXTWX(5,5),A(5,5)
      DO 5 I=1,N
      DO 5 J=1,N
       A(I,J)=(XTWX(I,J))
 5    CONTINUE
      DO 10 L=1,N
      A(L,L)=-1.0D0/A(L,L)
      DO 20 I=1,N
      IF (I-L) 30,20,30
 30   A(I,L)=-1.0D0*A(I,L)*A(L,L)
 20   CONTINUE
```

```
      DO 40 I=1,N
      DO 40 J=1,N
      IF ((I-L)*(J-L)) 50,40,50
50        A(I,J)=A(I,J)-A(I,L)*A(L,J)
40    CONTINUE
      DO 10 J=1,N
      IF (J-L) 60,10,60
60      A(L,J)=-1.0D0*A(L,J)*A(L,L)
10    CONTINUE
      DO 70 I=1,N
      DO 70 J=1,N
C     OXTWX(I,J)=-1.*ENGL(A(I,J))
      OXTWX(I,J)=-1.0*A(I,J)
 70   CONTINUE
      RETUEN
      END
C     ***********************************************
      SUBROUTINE MASS(S,NM2,V)
      COMMON/INTT/NM,NL,NCOM,NCEN,TC,NV
      DIMENSION S(100),V(100),F(100)
      DOUBLE PRECISION S,V,F
      DO 10 I= 1,NM2
      IF( I .EQ. 1) THEN
      V(I)=1.-S(I)
      ELSE
      II=I-1
      F(I)=1.-S(I)
      F(II)=1.-S(II)
      V(I)=F(I)-F(II)
      ENDIF
10    CONTINUE
      RETURN
      END


C     ***********************************************
      SUBROUTINE MSE(NC,NU,NT,YH,RMSEC,RMSEU,RMSET)
      DIMENSION YI(100),YH(100),NDELTA(100)
      COMMON/INTT/NM,NL,NCOM,NCEN,TC,NV
     *   /DIM2/YI,NDELTA
      SSEC =0.
      SSEU =0.
      SSET =0.
```

```
      DO 10 I=1,NM
      SSE = (YH(I)-YI(I))**2
C     WRITE(6,*) YI(I),YH(I),SSE,NDELTA(I)
      IF (NDELTA(I) .EQ. 0 ) THEN
      SSEC = SSEC+SSE
      ELSE
      SSEU = SSEU+SSE
      ENDIF
      SSET = SSET+SSE
 10   CONTINUE
      RMSEC = SQRT(SSEC/NC)
      RMSEU = SQRT(SSEU/NU)
      RMSET = SQRT(SSET/NT)
C     WRITE(6,*) 'RMSEC U T',RMSEC,RMSEU,RMSET
C     WRITE(6,*) 'NC NU NT ',NC,NU,NT,SSEC,SSEU,SSET
      RETURN
      END
C     ************************************************
      SUBROUTINE NORMAL(RMEAN,VARE,GN)
      COMMON/SEED/IX,KKIX
     *  /INTT/NM,NL,NCOM,NCEN,TC,NV
      SD = SQRT(VARE)
      PI = 3.1415926
      IF (KKIX .EQ.1) GOTO 100
        RONE = RAND(IX)
        RTWO = RAND(IX)
        ZONE = SQRT(-2*ALOG(RONE))*COS(2*PI*RTWO)
        ZTWO = SQRT(-2*ALOG(RONE))*SIN(2*PI*RTWO)
        GN  = ZONE*SD+RMEAN
        KKIX = 1
        GOTO 300
100   GN  = ZTWO*SD+RMEAN
      KKIX = 0
300   RETURN
      END
C     ************************************************
      SUBROUTINE LN(RMEAN,VARE,GX)
      COMMON/SEED/IX,KKIX
     *  /INTT/NM,NL,NCOM,NCEN,TC,NV
      SD = SQRT(VARE)
      PI = 3.1415926
      IF (KKIX .EQ.1) GOTO 100
      RONE = RAND(IX)
```

```
      RTWO = RAND(IX)
      ZONE = SQRT(-2*ALOG(RONE))*COS(2*PI*RTWO)
      ZTWO = SQRT(-2*ALOG(RONE))*SIN(2*PI*RTWO)
      GN  = ZONE*SD+RMEAN
      KKIX = 1
      GOTO 300
  100 GN  = ZTWO*SD+RMEAN
      KKIX = 0
  300 GX=EXP(GN)
C     GLN=ALOG(GX)
      RETURN
      END
C     ***************************************************
C     *******      FUNCTION RANDOM      ******
C     ***************************************************
      FUNCTION RAND(IX)
      IX  = IX*16807
      IF (IX.LT.0) IX = IX + 2147483647+1
      RAND = IX
      RAND = RAND*0.465661E-9
      RETURN
      END
C     ***************************************************
      SUBROUTINE PL(U,C,NU,NC,S)
      DIMENSION U(100),C(100),S(100),L(200),V(100)
      DOUBLE PRECISION S,PT
      COMMON/INTT/NM,NL,NCOM,NCEN,TC,NV
      K =1
      KK =0
      NT = NU+NC
      IF (NC .EQ. 0 ) GOTO 15
      DO 10 I= 1,NU
      IF(KK .EQ. NC) GOTO 1
      DO 20 J = K,NC
      K1 = J
      IF(C(J) .GE. U(I)) GOTO 1
   20 KK= KK+1
    1 K = K1
   10 L(I)= I+KK
      GOTO 11
   15 DO 19 I=1,NU
   19 L(I)=I
   11 CONTINUE
```

```
C     WRITE(6,100) (L(I),I=1,NU)
C100  FORMAT(2X,'RANKS',10I5)
C     COMPUTE P(T) FOR ALL UNCENSORED BASED ON L(RANK)
      V1= 0.
      PT=1.0
      XNT = NM
      DO 12 I=1,NU
      XL=L(I)
      PT=PT*((XNT-XL)/(XNT-XL+1.))
      S(I)=PT
      IF ((XNT-XL) .GT. 0. ) GOTO 21
      VV=0.
      GOTO 25
21    CONTINUE
      V1=V1+1./((XNT-XL)*(XNT-XL+1.0))
      VV=SQRT((PT**2)*V1)
25    CONTINUE
      V(I)=VV
12    CONTINUE
      DO 30 J=1,NU
30    CONTINUE
      RETURN
      END
C     ***************************************************
C     *****        SUBROUTINE PP          *****
C     ***************************************************
      SUBROUTINE PP(ZZ,PZ)
      R = EXP(-ZZ*ZZ/2.)/2.5066282746
      WW= 1./(1.+.33267*ABS(ZZ))
      P =1.- R*(.4361836*WW-.1201676*(WW**2)
     *    +.937298*(WW**3))
      IF (ZZ .GE.0.0) THEN
        PZ = P
      ELSE
        PZ = 1.-P
      ENDIF
      RETURN
      END
C     ***************************************************
      SUBROUTINE SORT1(EE,NN)
      DIMENSION EE(100),X(100,5),Y(100)
      COMMON/DIM1/X,Y
      DO 1 I=1,NN
```

```
      J = NN-I+1
      JJ = J-1
      IF (JJ .LT. 1) GOTO 1
      DO 2 K=1,JJ
      IF (EE(K) .LE. EE(J)) GOTO 2
4     E1=EE(J)
      X1=X(J,1)
      X2=X(J,2)
      EE(J)=EE(K)
      X(J,1)=X(K,1)
      X(J,2)=X(K,2)
      EE(K) =E1
      X(K,1)=X1
      X(K,2)=X2
2     CONTINUE
1     CONTINUE
      RETURN
      END
C     ********************************
      SUBROUTINE SORT2(EE,N,IR)
      DIMENSION EE(100),IR(100)
      DO 1 I=1,N
      J     = N-I+1
      JJ = J-1
      IF (JJ .LT. 1) GOTO 1
      DO 2 K=1,JJ
      IF (EE(K) .LE. EE(J)) GOTO 2
4     E1=EE(J)
      IR1=IR(J)
      EE(J)=EE(K)
      IR(J)=IR(K)
      EE(K)=E1
      IR(K)=IR1
2     CONTINUE
1     CONTINUE
      RETURN
      END
C     ************************************************
      SUBROUTINE TRANP(BM,ALFA,S,BK,AK,SK)
      COMMON/INTT/NM,NL,NCOM,NCEN,TC,NV
      DIMENSION BM(5),BK(3,5),AK(3),SK(3)
      DO 10 I=1,NV
      BK(1,I)=BK(2,I)
```

```
      BK(2,I)=BK(3,I)
      BK(3,I)=BM(I)
   10 CONTINUE
      AK(1)=AK(2)
      AK(2)=AK(3)
      AK(3)=ALFA
      SK(1)=SK(2)
      SK(2)=SK(3)
      SK(3)=S
      RETURN
      END
C     ****************************************
      SUBROUTINE  UNCEN(XNEW,YNEW,NMNEW)
      DIMENSION X(100,5),Y(100),NDELTA(100),YI(100),
     *      XNEW(100,5),YNEW(100)
      COMMON/INTT/NM,NL,NCOM,NCEN,TC,NV
     *   /DIM1/X,Y
     *   /DIM2/YI,NDELTA
      NMNEW = 0
      DO 85 I = 1,NM
      IF (NDELTA(I) .EQ. 1) THEN
      NMNEW = NMNEW+1
      DO 80 J = 1,NV
      XNEW(NMNEW,J) = X(NMNEW,J)
   80 CONTINUE
      YNEW(NMNEW)= Y(I)
      ENDIF
   85 CONTINUE
      RETURN
      END
C *****  SUBROUTINE UN (DISCRETE UNIFORM)  *****
      SUBROUTINE  UN(AAD,BBD,IEX)
      COMMON/SEED/IX,KKIX
     *   /INTT/NM,NL,NCOM,NCEN,TC,NV
      U     = RAND(IX)
      EX    = AAD+ ((BBD-AAD+1)*U)
      IEX   = EX
      RETURN
      END
C     ************************************************
      SUBROUTINE UNIFO(AA,BB,EX)
      COMMON/SEED/IX,KKIX
      YFL = RAND(IX)
```

```
      EX = ((BB-AA)*YFL)+ AA
      RETURN
      END
C     ***************************************************
      SUBROUTINE WEIBUL(ALPHA,BETA,TWEI)
      COMMON/SEED/IX,KKIX
10    YFL   = RAND(IX)
      IF ((YFL .EQ.1.0) .OR. (YFL .EQ.0.0)) GOTO 10
      TWEI = BETA*(-ALOG(1.0-YFL))**(1.0/ALPHA)
      RETURN
      END
C     ***************************************************
      SUBROUTINE MEANV(IR,A1,XMSE12,VMSE12,XMSE11,VMSE11,CV1)
      IF(IR.NE.1) THEN
      XMSE12 = XMSE11 + (A1 - XMSE11)/IR
      VMSE12 = (1.-1./(IR-1))*VMSE11 + IR*((XMSE12-XMSE11)**2)
      SV1 = SQRT(VMSE12)
      IF(XMSE12.EQ.0.) GOTO 2445
      CV1 = SV1/XMSE12
2445  XMSE11 = XMSE12
      VMSE11 = VMSE12
      GOTO 2999
      ENDIF
      XMSE11 = A1
      VMSE11 = 0.
2999  RETURN
      END
```

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียน

นางสาวจำเนียน จำนงค์รักษ์ เกิดวันที่ 22 มกราคม 2511 สำเร็จการศึกษาปริญญาสถิติศาสตรบัณฑิต จากภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2529 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาการประกันภัย ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2536

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย