หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

บรรณานุกรม

ภาษาไทย

วิทยานิพนธ์

สากันย์ สุวรรณการ "วิธีนอนพาราเมตริกสำหรับการประมาณค่าฟังก์ชั่นการอยู่รอดในปัญหา 2 ตัวอย่างที่มีค่าสังเกตไม่สมบูรณ์." วิทยานิพนธ์ ปริญญามหาบัณฑิต ภาควิชาสถิติ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2530.

สะอาด นิวิศพงษ์ "การเปรียบเทียบอำนาจการทดสอบของตัวสถิติทดสอบบางตัวที่ใช้ทดสอบการแจกแจงแบบเอกซ์โปเนนเชียล." วิทยานิพนธ์ ปริญญามหาบัณฑิต ภาควิชาสถิติ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2532.

ภาษาต่างประเทศ

หนังสือ

Lee, E.T. Statistical Methods for Survival Data Analysis, Lifetime Learning Publication : USA, 1980.

Nelson, W. Applied Data Analysis, New York : John Wiley & Sons : 1982.

บทความ

David A. Schoenfeld and Anastasios A. Tsiatis. "A modified logrank test for highly stratified dats." Biometrika, 74, 167-175, 1987.

David P. Harrington and Thomas R. Fleming. "A class of rank test procedures for censored survival data," Biometrika, 69, 553-566, 1982.

ภาคผนวก

ภาคผนวก ก.

ภาคผนวก ก

```
C*************************************************************************
C****    A  COMPARISON ON POWER OF SOME NONPARAMETRIC TEST    ****
C****          FOR COMPARING TWO SURVIVAL DISTRIBUTION        ****
C****                    WITH CENSORED DATA                   ****
C********    BY  MISS SAMILA VICHILNROJ   ID.6023025     ********
C*-----------------------------------------------------------------*
C*     MAIN PROGRAM TO COMPUTE TYPE I ERROR AND POWER OF TEST    *
C*                                                               *
C*************************************************************************
      DIMENSION IDATA1(100),IDATA2(100),IDA1(100),EST(100)
     *,W(100),WW(100),DATA(100),CDATA(100),IDATA(100),A(100)
     *,CCAL(100),PPL(100),U(100),IRANK1(100),IRANK(100),C(100)
     *,ID(100),XY(100),ID1(100),ID2(100),E(100),R(100),A(100)
     *,E1(100),H(100),S2(100),RISK(100),X(100),XM(100),H2(100)
     *, AL1(100),AL2(100),SL1(100),SL2(100),D1(100),D2(100)
     *,PARA(10),SDATA(100)
C*************************************************************************
      REAL L,LOG01,LOG05,PML01,PML05,MLR,MLOG01,MLOG05
      INTEGER RDU
      COMMON IA
     *       /SEED/IX
     *     /SELECT/KK
C*************************************************************************
      N1 = 20
      SN1 = N1
      N2 = 20
      IP = N1+N2
      EX = 0
      BETA1 = 1
C
      NU1 = 2
      KAE = 2
C*************************************************************************
      DO 701 NA=1,3
        READ(5,2) PARA(NA)
  2     FORMAT(F2.1)
 701  CONTINUE
      DO 700 NO3 = 1,4
          READ(5,3)IR1,IR2
  3       FORMAT(2(I2))
      DO 700 NA = 1,1
      NA1 = NA+1
      MU = N1-IR1
      MC = IR1
      NU = N2-IR2
      NC = IR2
      IA = 65539
      IX = 973253
      KK = 0
      LOG01  = 0
      LOG05  = 0
      MLOG01 = 0
      MLOG05 = 0
      GW01   = 0
      GW05   = 0
      TTS01 = 0
      TTS05 = 0
```

```
      COX01 = 0
      COX05 = 0
      PLO01 = 0
      PLO05 = 0
      PML01 = 0
      PML05 = 0
      PGW01 = 0
      PGW05 = 0
      PTIS01 = 0
      PTIS05 = 0
      PCOX01 = 0
      PCOX05 = 0
      IL = 0
      JSAV = 0
      ISAVE = 0
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
      BETA = PARA(3)
      STD  = PARA(3)
      ALPHA = PARA(3)
      BETA2 = PARA(1)
      STD2 = PARA(1)
      ALPHA2 = PARA(1)
      ROU = 500
      DO 7000 MN = 1,ROU
    1   IJ1= 1
        IJ2 = N1
        GO TO (1000,2000,3000),NUI
C______________________________________________________________________
C       NUI = 1 IS EXPONENTIAL
C       NUI = 2 IS LOGNORMAL
C       NUI = 3 IS WEIBULL
C______________________________________________________________________
 1000   DO 1005 J = IJ1,IJ2
        CALL EXPO(BETA,XX)
 1005   DATA(J) = XX
        GO TO 6
 2000   DO 2005 J = IJ1,IJ2
 2005   DATA(J) = EXP(SNORM(EX,STD))
        GO TO 6
 3000   DO 3005 J = IJ1,IJ2
 3005   DATA(J) = WEIBUL(ALPHA,BETA1)
        GO TO 6
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
    6   DO 5 J = IJ1,IJ2
          DATA(J) = ABS(DATA(J))
          IDATA1 (J) = DATA(J)*100
          DATA(J) = IDATA1 (J)/100.
          XY(J) = DATA(J)
          ID1(J) = 1
          ID2(J) = 1
          M(J) = 0
    5   CONTINUE
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
        IJ1 = N1+1
        IJ2 = IP
        GO TO (4000,5000,6000),KAE
 4000   DO 4005 J = IJ1,IJ2
```

```
      CALL EXPO(BETA2,XX)
 4005 DATA(J) = XX
      GO TO 15
 5000 DO 5005 J = IJ1,IJ2
 5005 DATA(J) = EXP(SNORM(EX,STD2))
      GO TO 16
 6000 DO 6005 J = IJ1,IJ2
 6005 DATA(J) = WEIBUL(ALPHA2,BETA1)
      GO TO 16
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
   16 IJ = N1+1
      DO 10 J = IJ,IP
         DATA(J) = ABS(DATA(J))
         IDATA2(J) = DATA(J)*100
         DATA(J) = IDATA2(J)/100.
         XY(J) = DATA(J)
         ID1(J) = 1
         ID2(J) = 2
         M(J) = 0
   10 CONTINUE
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      IF (IR1.GT.0) THEN
         NC = N1-IR1+1
         DO 15 I = NC,N1
            ID1(I) = 0
   15    CONTINUE
      ENDIF
      IF (IR2.GT.0) THEN
         NC = N2-IR2+1+N1
         DO 25 I = NC,IP
            ID1(I) = 0
   25    CONTINUE
      ENDIF
C**********************************************************
C*******       COMPUTE COX-MANTEL TEST          *******
C**********************************************************
      NCEN = MC+NC
      N = IP
      DO 220 I = 1,N
         J = N-I+1
         JJ = J-1
         IF (JJ.LT.1)GOTO 220
           DO 215 K = 1,JJ
             IF(XY(K).NE.XY(J)) GO TO 200
             IF(ID1(K)-ID1(J))205,215,215
  200        IF(XY(K).LE.XY(J)) GO TO 215
  205        X1 = XY(J)
             ITEM = ID1(J)
             ICTE = ID2(J)
             XY(J) = XY(K)
             ID1(J) = ID1(K)
             ID2(J) = ID2(K)
             XY(K) = X1
             ID1(K) = ITEM
             ID2(K) = ICTE
  215      CONTINUE
  220 CONTINUE
```

```
      IF(ID1(1).EQ.0) GO TO 1
      IF (ID1(IP).NE.0) GO TO 1
C
      L = 1
      I = 1
      R(L) = N
  225 IF (ID1(I).EQ.1) GO TO 230
      R(L) = R(L)-1
      I = I+1
      GOTO 225
  230 XM(L) = 1
      XNC = 0
      TEMP = XY(I)
      X(L) = TEMP
  235 IF (I.EQ.N) GOTO 250
      I = I+1
      IF(ID1(I).EQ.0) GO TO 240
      IF(TEMP.NE.XY(I)) GOTO 245
      XM(L) = XM(L)+1
      GO TO 235
  240 XNC = XNC+1
      X(L) = TEMP
      GOTO 235
  245 L = L+1
      R(L) = R(L-1)-XM(L-1)-XNC
      IF (R(L).EQ.1) GO TO 1
      GOTO 230
  250 X(L) = TEMP
      NG = L
C
      L = 0
      DO 270 I = 1,N
        IF(ID2(I).EQ.1) GOTO 270
        L = L+1
        S2(L) = XY(I)
  270 CONTINUE
      IR = L
      K = 1
      DO 285 I = 1,NG
        LL = L+1
        IF (K.EQ.LL) GOTO 276
  275   IF(X(I).LE.S2(K)) GOTO 280
  276   IF(IR.EQ.0) GO TO 280
        IR = IR-1
        IF(K.EQ.0) GOTO 280
        K = K+1
        IF(K.LE.L) GOTO 275
  280   RISK(I) = IR
  285 CONTINUE
      DO 300 J = 1,NG
        IF (R(J).EQ.1) GO TO 1
        A(J) = RISK(J)/R(J)
  300 CONTINUE
      XNU = NU
      SUM = 0
      DO 305 J = 1,NG
  305 SUM = SUM+XM(J)*A(J)
```

```
      SUM = XNU-SUM
      VAR = 0
      DO 310 J = 1,NG
  310 VAR = VAR+(XM(J)*(R(J)-XM(J))*A(J)*(1.-A(J)))/(R(J)-1.)
      COX = SUM/SQRT(VAR)
      IF(COX.LT.-1.2815) COX01  = COX01+1
      IF(COX.LT.-1.645) COX05 = COX05+1
C*************************************************
C**            COMPUTE LOGRANK TEST             **
C*************************************************
      I = 1
      DO 50 IIP = 1,IP
         IF (I.LE.IP) THEN
            M(I) = 1
            IF (I.GE.IP)GOTO 55
               II = I+1
               DO 40 J = II,IP
                IF ((DATA(I).EQ.DATA(J)).AND.(ID1(I).EQ.ID1(J))) THEN
                     M(I)= M(I)+1
                ELSE
                     JJ = J
                     GOTO 45
                ENDIF
                  JJ = J
   40          CONTINUE
   45          I = JJ
            ELSE
               GO TO 55
            ENDIF
   50 CONTINUE
   55 IRANK(1) = IP
      IRANK1(1) = 1
      NP = IP-1
      DO 60 I = 1,NP
         IRANK(I+1) = IP-I
         IRANK1(I+1) = I+1
   60 CONTINUE
      DO 65 IIP = 1,IP
            IF (ID1(IIP).EQ.0) THEN
               IRANK(IIP) = 0
               IRANK1(IIP) = 0
               M(IIP) = 0
               GOTO 65
   64       ENDIF
   65 CONTINUE
      EST(1) = 0
      EEST = 0
      DO 70 I = 1,IP
         IF ((M(I).EQ.0).OR.(IRANK(I).EQ.0))THEN
            EST(I) = 0
            CCAL(I) = 0
         ELSE
            RANK = IRANK(I)
            CAL = M(I)/RANK
            CCAL(I) = CAL
            EST(I) = EEST+CAL
            EEST = EST(I)
```

```
            ENDIF
 70      CONTINUE
         SUM = 0
         SSUMW = 0
         IF (EST(1).EQ.0)THEN
            JSAV = JSAV+1
            WRITE(6,75)MN,JSAV
 75         FORMAT(10X,'ERROR =',5X,I4,5X,'TOTAL =',I4/)
            GOTO 1
         ELSE
            W(1) = 1-EST(1)
            DO 85 I = 2,IP
               IF (EST(I).NE.0) THEN
                  W(I) = 1-EST(I)
               ELSE
                  NI = I-1
                  DO 80 II = 1,NI
                     IF (EST(II).NE.0) ISAVE = II
 80               CONTINUE
                  W(I) = -EST(ISAVE)
               ENDIF
 85       CONTINUE
         ENDIF
         DO 95 I = 1,IP
            IF (M(I).GT.1) THEN
               IM = M(I)
               NI = I+1
               LI = I+IM-1
               DO 86 JJ = NI,LI
                  W(JJ) = W(I)
 86            CONTINUE
               DO 90 J = 1,IM
                  SSUMW = SSUMW+(W(I)**2)
 90            CONTINUE
            ELSE
               IF ((M(I).EQ.0).AND.(IRANK(I).NE.0)) GOTO 95
               SSUMW = SSUMW+(W(I)**2)
            ENDIF
 95      CONTINUE
C*************************************************************
C****       COMPUTE PETO & PETO'S WILCOXON TEST          ****
C*************************************************************
         PL = 1.
         DO 110 I = 1,IP
            IF (IRANK1(I).NE.0) THEN
               P = IP
               RANK = IRANK1(I)
               CAL = (P-RANK)/(P-RANK+1)
               CCAL(I) = CAL
               PL = PL*CAL
               PPL(I) = PL
            ELSE
               CCAL(I) = 0
               PPL(I) = 0
            ENDIF
110      CONTINUE
         DO 115 I = 1,IP
```

```
            IF (M(I).GT.1)THEN
                IM      = M(I)
                LI      = I+IM-1
                DO 114 J = I,LI
114             PPL(J) = PPL(LI)
            ENDIF
115     CONTINUE
        U(1) = PPL(1)
        DO 135 I = 2,IP
           IF (IRANK1(I).NE.0) THEN
                IF (PPL(I-1).NE.0) THEN
                   U(I) = PPL(I)+PPL(I-1)-1
                ELSE
                   NI = I-1
                   DO 120 J = 1,NI
                      IF(PPL(J).NE.0) ISAVE = J
120                CONTINUE
                   U(I) = PPL(I)+PPL(ISAVE)-1
                ENDIF
           ELSE
                NI = I-1
                DO 130 JJ = 1,NI
                   IF (PPL(JJ).NE.0) ISAVE = JJ
130             CONTINUE
                U(I) = PPL(ISAVE)-1
           ENDIF
135     CONTINUE
        DO 138 I = 1,IP
           IF (M(I).GT.1) THEN
              IM = M(I)
              NI = I+1
              LI = I+IM-1
              DO 137 J =NI,LI
137              U(J) = U(I)
           ENDIF
138     CONTINUE
        SUMU = 0
        SSUMU = 0
        DO 140 I = 1,IP
           SSUMU = SSUMU+(U(I)**2)
140     CONTINUE
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
        VL1 = (N1*N2*SSUMW)
        VU1 = (N1*N2*SSUMU)
        V2 = (N1+N2)*(N1+N2-1)
        VARL = VL1/V2
        VARU = VU1/V2
        DO 105 I = 1,IP
              IF (ID2(I).EQ.2)THEN
                 SUM = SUM+W(I)
                 SUMU = SUMU+U(I)
              ENDIF
105     CONTINUE
        L = SUM/SQRT(VARL)
        GW = SUMU/SQRT(VARU)
        IF (L.LT.-1.2815) LOG01 = LOG01+1
        IF (L.LT.-1.645) LOG05 = LOG05+1
```

```
      IF (GW.LT.-1.2815) GW01  = GW01+1
      IF (GW.LT.-1.645) GW05  = GW05+1
C*************************************************************
C***      COMPUTE MODIFIED LOGRANK TEST                      ***
C*************************************************************
      T=0
      P=0.5
      Q =0.5
      DO 146 I=1,IP
         IF (ID2(I).EQ.2) THEN
            II2(I) = 1
         ELSE
            II2(I) = 0
         ENDIF
  146 CONTINUE
      DO 155 I = 1,IP
         AL1(I)=0
         AL2(I)=0
         DO 150 IM=1,IP
            IF(DATA(IM).GE.DATA(I)) THEN
              IDF=1
            ELSE
              IDF=0
            ENDIF
            AL1(I)=((Q*II2(IM)*1)*IDF)+AL1(I)
            AL2(I)=((P*(1-II2(IM)))*IDF)+AL2(I)
  150    CONTINUE
         SPL=(P*AL1(I))+(Q*AL2(I))
         SL1(I)=AL1(I)/SPL
         SL2(I)=AL2(I)/SPL
         T=T+(ID1(I)*((II2(I)*Q*SL2(I))-((1-II2(I))*P*SL1(I))))
  155 CONTINUE
      DO 170 I=1,IP
         COM1 = 0
         COM2 = 0
         DO 165 IM=1,IP
            CAL1 = 0
            DO 160 MP=1,IP
               CA1=(P*Q*II2(MP))+(Q*P*(1-II2(MP)))
               IF(DATA(MP).GE.DATA(IM)) THEN
                 IDF=1
               ELSE
                 IDF=0
               ENDIF
  160          CAL1=(CA1*IDF)+CAL1
               CAL2=(P*Q*II2(IM))+(Q*P*(1-II2(IM)))
               IF(DATA(IM).LE.DATA(I)) THEN
                 IDF=1
               ELSE
                 IDF=0
               ENDIF
               CAL2=CAL2*ID1(IM)*IDF
               CD1=CAL2*SL1(IM)
               CD2=CAL2*SL2(IM)
               IF(CAL1.NE.0) THEN
                 COM1=COM1+(CD1/CAL1)
                 COM2=COM2+(CD2/CAL1)
```

```
                ELSE
                  COM1=COM1
                  COM2=COM2
              ENDIF
  165      CONTINUE
           D1(I)=(ID1(I)*SL1(I))-COM1
           D2(I)=(ID1(I)*SL2(I))-COM2
  170     CONTINUE
          CAL1=0
          CAL2=0
          COM1=0
          COM2=0
          DO 175 I=1,IP
             CAL1=CAL1+((1-II2(I))*(D1(I)**2))
             CAL2=CAL2+(II2(I)*(D2(I)**2))
             COM1=COM1+((1-II2(I))*D1(I))
             COM2=COM2+(II2(I)*D2(I))
  175     CONTINUE
          V=((P**2)*CAL1)+((Q**2)*CAL2)-(((1./M)*COM1)*COM2)
          IF (V.LE.0) THEN
          WRITE(6,238)V
  238     FORMAT(10X,'VAR ERROR =',F10.2)
          GOTO 1
          ENDIF
          MLR=T/SQRT(V)
          IF(MLR.LT.-1.2815) MLOG01=MLOG01+1
          IF(MLR.LT.-1.645) MLOG05=MLOG05+1
C*************************************************************
 7000 CONTINUE
C*************************************************************
C***      COMPUTE PROB TYPEI ERROR OR POWER OF TEST          ***
C*************************************************************
      PLOG1 = LOG01/ROU
      PLOG5 = LOG05/ROU
      PMLO1 = MLOG01/ROU
      PMLO5 = MLOG05/ROU
      PGWO1 = GWO1/ROU
      PGWO5 = GWO5/ROU
      PCOX01 = COX01/ROU
      PCOX05 = COX05/ROU
      WRITE(6,610)NUI,N1,IR1,IR2,BETA,BETA2,STD,STD2,ALPHA,ALPHA2
  610 FORMAT(/10X,'DISTRIBUTION =',I4/10X,'NUMBER GROUP1,2 =',I4/
     *10X,'CENSORED DATA1 =',I4,10X,'CENSORED DATA2 =',I4/
     *10X,'BETA1=',F4.2,5X,'BETA2=',F4.2,/10X,'STD1=',F4.2,5X,
     *'STD2= ',F4.2/10X,'ALPHA1=',F4.2,5X,'ALPHA2= ',F4.2/ /)
      WRITE(6,611)PLOG1,PLOG5,PGWO1,PGWO5,PCOX01,PCOX05,PMLO1,PMLO5
  611 FORMAT(10X,'POWER OF THE TEST',5X,'LOG10= ',F10.5/
     *       10X,'POWER OF THE TEST',5X,'LOG05= ',F10.5//
     *       10X,'POWER OF THE TEST',5X,'WILO10= ',F10.5/
     *       10X,'POWER OF THE TEST',5X,'WILC05= ',F10.5//
     *       10X,'POWER OF THE TEST',5X,'COX10 = ',F10.5/
     *       10X,'POWER OF THE TEST',5X,'COX05 = ',F10.5//
     *       10X,'POWER OF THE TEST',5X,'MLOG10= ',F10.5/
     *       10X,'POWER OF THE TEST',5X,'MLOG05= ',F10.5)
  700 CONTINUE
      STOP
      END
```

```
C**********************************************************************
C*******          SUBROUTINE SORT DATA                  **********
C**********************************************************************
      SUBROUTINE SORT(IP,DATA)
      DIMENSION DATA(IP),
      NL = IP-1
      DO 30 I = 1,NL
      NP = I + 1
      DO 30 K = NP,IP
      IF (DATA(I).LE.DATA(K)) GOTO 30
      SAVE = DATA(I)
      DATA(I) = DATA(K)
      DATA(K) = SAVE
   30 CONTINUE
      RETURN
      END
C**********************************************************************
C*******          FUNCTION NORMAL DISTRIBUTION          ******
C**********************************************************************
      FUNCTION SNORM(DMEAN,SIGMA)
      COMMON /SEED/IX/SELECT/KK
      PI = 3.1415926
      IF (KK.EQ.1) GOTO 10
         CALL RANDUM(IX,IY,YFL)
         RONE = YFL
         CALL RANDUM(IX,IY,YFL)
         RTWO = YFL
         ZONE = SQRT(-2*ALOG(RONE))*COS(2*PI*RTWO)
         ZTWO = SQRT(-2*ALOG(RONE))*SIN(2*PI*RTWO)
            SNORM = ZONE*SIGMA+DMEAN
      KK = 1
      RETURN
   10       SNORM = ZTWO*SIGMA+DMEAN
      KK = 0
      RETURN
      END
C**********************************************************************
C*******       SUBROUTINE  RANDUM          **********
C**********************************************************************
      SUBROUTINE RANDUM(IX,IY,YFL)
      IY = IX*65539
      IF (IY) 5,6,6
    5 IY = IY + 2147483647 + 1
    6 YFL = IY
      YFL = YFL/2147483647
      IX = IY
      RETURN
      END
C**********************************************************************
C********     FUNCTION WEIBULL DISTRIBUTION    *******
C**********************************************************************
      FUNCTION WEIBUL(ALPHA1,BETA1)
      COMMON /SEED/IX
      CALL RANDUM(IX,IY,YFL)
      WEIBUL = BETA1*(-ALOG(1.-YFL))**(1./ALPHA1)
      RETURN
      END
```

```
C*******************************************************
C*******    SUBROUTINE  EXPONENTIAL          *********
C*******************************************************
      SUBROUTINE EXPO(BETA,XX)
      COMMON/SEED/IX
      CALL RANDUM(IX,IY,YFL)
      XX = (-ALOG(YFL))*BETA
      RETURN
      END
```

ภาคผนวก ข.

การผลิตและสุ่มที่มีการแจกแจงแบบเอกซ์โปเนนเชียล

วิธีการหาเลขสุ่มที่มีการกระจายแบบเอกซ์โปเนนเชียล ต้องหาเลขสุ่มที่มีการแจกแจงแบบสม่ำเสมอ (R) ในช่วง (0.1) เมื่อ

$$R = e^{-x/\beta}$$

$$F^{-1}_{(R)} = -\beta x \ln R \quad = x \qquad \ldots(1)$$

แล้วจึงหาเลขสุ่มที่มีการแจกแจง EXPONENTIAL (x) โดยใช้สมการ (1) ในการวิจัยครั้งนี้ได้เขียนโปรแกรมด้วยภาพโฟรแทรน 4 โดยเขียนเป็น SUBROUTINE มีชื่อว่า EXPO ดังนี้

```
SUBROUTINE EXPO (BETA, XX)
COMMON/SEED/IX
CALL RANDUM (IX, IY, YFL)
XX = (-ALOG (YFL)) *BETA
RETURN
END
```

การผลิตเลขสุ่มที่มีการแจกแจงแบบลอกนอร์มอล

การแจกแจงแบบลอกนอร์มอลมีฟังก์ชั่นความหนาแน่นอยู่ในรูปของ

$$f(x) = \begin{cases} \dfrac{1}{x\sigma\sqrt{2\pi}}\, e^{-\frac{1}{2}(\log_e x-\mu)/\ ^2} & ; \ x > 0,\ \sigma > 0,\ -\infty<\mu<\infty \\ 0 & ; \ \text{อื่น ๆ} \end{cases}$$

เมื่อ μ และ σ^2 เป็นค่าเฉลี่ยและความแปรปรวนของ Y ซึ่ง Y = logeX และ Y มีการแจกแจงแบบปกติ โดยมี exp (σ^2) เป็น scale parameter และ μ เป็น shape parameter

ในการวิจัยครั้งนี้ใช้คำสั่งในการสร้างตัวแปรให้มีการแจกแจงแบบลอกนอร์มอล คือ

```
DATA(J) = EXP(SNORM(EX, STD)).
```

เมื่อ FUNCTION NORMAL เป็นดังนี้

```
      FUNCTION SNORM (DMEAN, SIGMA)
      COMMON/SEED/IX/SELECT/KK
      IF (XX. EQ.1) GO TO 10
          CALL RANDUM (IX, IY, YFL)
          RTWO  =  YFL
          ZONE  =  SQRT (-2 * ALOG (RONE)) * COS (2 * PI * RTWO)
          ZTWO  =  SQRT (-2 * ALOG (RONE)) * SIN (2 * PI * RTWO)
          SNORM =  ZONE * SIGMA + DMEAN
      KX = 1
      RETURN
10        SNORM = ZTWO * SIGMA + DMEAN
      KK = 0
      RETURN
      END
```

การผลิตเลขสุ่มที่มีการแจกแจงแบบไวบูลล์

การแจกแจงแบบไวบูลล์ มีฟังก์ชั่นความหนาแน่นอยู่ในรูปของ

$$f(x) = \begin{cases} \dfrac{\alpha x^{\alpha-1}\ \exp(-x/\beta)^{\alpha}}{\beta^2} & ;\ 0 < x < \infty, \infty < 0,\ \beta < 0 \\ 0 & ; \end{cases}$$

โดย β เป็น scale parameter และ α เป็น shape parameter

การสร้างตัวแปรสุ่มให้มีการแจกแจงแบบไวบูลล์ อาศัยเทคนิคการแปลงผกผัน (inverse transformation)

ขั้นที่ 1 cdf. เขียนเป็น $F(x) = 1 - e^{-(x/\beta)}$; $x > 0$

ขั้นที่ 2 ให้ $F(x) = 1 - e^{-(x/\beta)^{\alpha}} = R$

ขั้นที่ 3 หาค่าของ X ในเทอมของ R ได้เป็น

$$X = \beta[-\log_e(1-R)]^{1/\alpha}$$

ดังนั้น คำสั่งในการสร้างตัวแปรให้มีการแจกแจงแบบไวบูลล์ คือ

```
FUNCTION WEIBUL (ALPHA1, BETA1)
COMMON / SEED / IX
CALL RANDUM (IX, IY, YFL)
WEIBUL = BETA1 * (-ALOG(1.0 - YFL) ** (1.0/ALPHA1)
RETURN
END
```

ภาคผนวก ก.

วิธีการประมาณค่าฟังก์ชั่นการอยู่รอด

วิธีนอนพาราเมตริก สำหรับการประมาณค่าฟังก์ชั่นการอยู่รอดที่ใช้กับข้อมูลที่มีค่าสังเกตไม่สมบูรณ์ มีอยู่หลายวิธี เช่น Product - Limit (PL), วิธี Life - gable (actuarial) และวิธี Cox'reqression model ซึ่งในที่นี้จะกล่าวถึงวิธีการประมาณค่าฟังก์ชั่นการอยู่รอดโดยวิธี PL

วิธี Product - Limit

วิธี PL นี้เป็นวิธีที่พัฒนาขึ้นมาใช้ โดย Kaplan & Meier (1950) ซึ่งวิธีนี้เป็นวิธีที่สามารถนำไปใช้กับตัวอย่างขนาดเล็กหรือตัวอย่างขนาดกลางหรือตัวอย่างขนาดใหญ่ก็ได้

ในทางปฏิบัติตัวประมาณ PL สามารถคำนวณได้โดยการจัดเรียงลำดับข้อมูลตัวอย่างขนาด N ที่มีทั้งค่าสังเกตสมบูรณ์และไม่สมบูรณ์นั่นคือ

$$t_{(1)} < t_{(2)} < \ldots < t_{(n)}$$

กำหนดให้ $j = 1, 2, \ldots, n$ และให้ $r = j$ จะได้

$$\hat{S}'_{(t)} = \prod_{r=t_{(r)} \leq t} (n-r)/(n-r+1)$$

เมื่อ $t_{(r)}$ เป็นค่าสังเกตสมบูรณ์ และ $(n-r)/(n-r+1)$ คือสัดส่วนของตัวอย่างที่ยังอยู่รอดที่เวลา $t_{(j)}$ ถ้าค่าสังเกตทุกค่าในตัวอย่างมีค่าสมบูรณ์ ค่าประมาณจะเป็นค่าประมาณแบบทวินาม มีค่าเป็น $\hat{S}_{(t)} = n(t)/N$ เมื่อ $n(t)$ คือจำนวนตัวอย่างที่ยังอยู่รอดที่เวลา t และตัวประมาณ S(t) เรียกว่าตัวประมาณ PL ในกรณีที่เกิดมีข้อมูลซ้ำค่า $S(t) = N-1/n$ เมื่อ i คือ ลำดับของข้อมูลที่เท่ากันที่มีค่าลำดับสูงที่สุด เช่น $t_{(2)} = t_{(3)} = t_{(4)}$ ดังนั้น

$$\hat{S}(t_2) = \hat{S}(t_3) - \hat{S}(t_4) = \frac{N-4}{N}$$

ตัวอย่างการคำนวณหาค่า

จากค่าสังเกต 2 กลุ่ม ดังนี้

กลุ่มที่ 1	3.0	5.7^{+}	6.5	15.0		
กลุ่มที่ 2	4.0^{+}	6.5	8.4^{+}	10.0	10.0^{+}	12.0

เมื่อ + หมายถึง ข้อมูลที่ไม่สมบูรณ์

ตารางที่ ค.1 ตารางแสดงการคำนวณหาค่าปริมาณ ของ

ข้อมูล	ลำดับ (i)	r	(n-r)/(n-r+1)	S(t)
3.0	1	1	9/10	9/10=0.900
4.0^{+}	2	-	-	-
5.7^{+}	3	-	-	-
6.5	4	4	6/7	$^{9}/_{10} \times {}^{6}/_{7} = 0.771)^{a}$
6.5	5	5	5/6	$^{9}/_{10} \times {}^{6}/_{7} \times {}^{5}/_{6} = 0.643^{a}$
8.4^{+}	6	-	-	-
10.0	7	7	3/4	$^{9}/_{10} \times {}^{6}/_{7} \times {}^{5}/_{6} \times {}^{3}/_{4} = 0.482$
10.0^{+}	8	-	-	-
12.0	9	9	1/2	$^{9}/_{10} \times {}^{6}/_{7} \times {}^{5}/_{6} \times {}^{3}/_{4} \times {}^{1}/_{2} = 0.241$
15.0	10	10	0	0

0.643 จะใช้เป็นค่าปริมาณ PL ของ $\hat{S}(6.5)$

การคำนวณหาค่าประมาณ PL นี้จะคำนวณทุก ๆ ค่าสังเกตสมบูรณ์ และถ้ามีข้อมูลที่เป็นค่าซ้ำจะใช้ค่า $\hat{S}(t)$ ที่น้อยที่สุด

ประวัติผู้เขียน

นางสาว ศมีลา วิเชียรโรจน์ เกิดวันที่ 5 ธันวาคม พ.ศ.2508 จังหวัดกรุงเทพฯ ได้รับปริญญาวิทยาศาสตรบัณฑิต สาขาคณิตศาสตร์ จากจุฬาลงกรณมหาวิทยาลัย ปีการศึกษา 2530 ได้เข้าศึกษาในภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณมหาวิทยาลัย เมื่อปีการศึกษา 2530