





ภาคผนวก ก

โปรแกรมสร้างกลุ่มประชากรที่มีการแจกแจงปกติสองตัวแปร (PRANDOM1.PR3)

*** Y=GRADE,X=SCORE OF MATHEMATICS

CLEAR

CLOSE ALL

SET TALK OFF

SET SAFETY OFF

SET DECIMALS TO 6

N=1

S=1

R=.7

ROUND = 10000

CREATE TABLE DATARAN1 (ID N(10),Y1 N(10,6),Y2 N(10,6),X N(10,6),;

YX1 N(10,6),YX2 N(10,6), YX3 N(10,6), STA1 N(10,6), CUSTER1 N(10,6),;

STA2 N(10,6), CUSTER2 N(10,6),STA3 N(10,6), CUSTER3 N(10,6),NID N(10))

DELETE ALL

PACK

@ 19,8 SAY 'PROGRAM "CONSTRUCTING BIVARIATE NORMAL DISTRIBUTION" IS WORKING'

@ 20,28 SAY '1 ST POPULATION'

@ 21,30 SAY 'PROCESS TIME'+STR(N)

DO WHILE N <= ROUND

DO WHILE S>=1

RN=RAND()

V1=2*RN - 1

RN=RAND()

V2=2*RN - 1

S=V1*V1 + V2*V2

ENDDO

RNN1 = V1*SQRT ((-2*LOG(S))/S)

RNN2 = V2*SQRT ((-2*LOG(S))/S)

Y11=RNN1

Y22=RNN2

```

XX=50.0000+(SQRT(100.0000)*Y11)
CRR=SQRT(.15)/SQRT(100.0000)
RR=CRR*R
YY=1.6127 + (RR*(XX-50.0000)) + (SQRT(.15*(1-(R*R))) * Y22)
YY1=2.0000 +(RR*(XX-50.0000)) + (SQRT(.15*(1-(R*R))) * Y22)
YY2=2.3873 +(RR*(XX-50.0000)) + (SQRT(.15*(1-(R*R))) * Y22)

```

```

USE DATARAN1
APPEND BLANK
REPLACE ID WITH N
REPLACE Y1 WITH RNN1
REPLACE Y2 WITH RNN2

REPLACE X WITH XX
REPLACE Yx1 WITH YY1

@ 21,30 SAY 'PROCESS TIME'+STR(N)
N=N+1
S=1
ENDDO

```

```

WAIT "
CLEAR
GO TOP
BROW
GO BOTTOM
BROW
GO TOP
COUNT TO NUM
GO TOP

```

```

SUMX=0
SUMY=0

```

SUMXX=0

SUMYY=0

SUMXY1=0

SUMY2=0

SUMYY2=0

SUMY3=0

SUMYY3=0

DO WHILE NOT EOF()

**YX1

SUMX=SUMX+X

SUMY=SUMY+Yx1

SUMXX=SUMXX+(X*X)

SUMYY=SUMYY+(YX1*YX1)

SUMXY1 = SUMXY1+ (X*YX1)

**YX2

SUMY2=SUMY2+Yx2

SUMYY2=SUMYY2+(YX2*YX2)

**YX3

SUMY3=SUMY3+Yx3

SUMYY3=SUMYY3+(YX3*YX3)

SKIP

ENDDO

MEANX=SUMX/NUM

MEANY=SUMY/NUM

VARX=((((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1))

VARY=((((NUM*SUMYY)-(SUMY*SUMY)))/(NUM*(NUM-1))

MEANX=SUMX/NUM

MEANY2=SUMY2/NUM

$$\text{VARX} = (((\text{NUM} * \text{SUMXX}) - (\text{SUMX} * \text{SUMX}))) / (\text{NUM} * (\text{NUM} - 1))$$

$$\text{VARY2} = (((\text{NUM} * \text{SUMYY2}) - (\text{SUMY2} * \text{SUMY2}))) / (\text{NUM} * (\text{NUM} - 1))$$

$$\text{MEANX} = \text{SUMX} / \text{NUM}$$

$$\text{MEANY3} = \text{SUMY3} / \text{NUM}$$

$$\text{VARX} = (((\text{NUM} * \text{SUMXX}) - (\text{SUMX} * \text{SUMX}))) / (\text{NUM} * (\text{NUM} - 1))$$

$$\text{VARY3} = (((\text{NUM} * \text{SUMYY3}) - (\text{SUMY3} * \text{SUMY3}))) / (\text{NUM} * (\text{NUM} - 1))$$

$$\text{RA} = (\text{NUM} * (\text{SUMXY1})) - ((\text{SUMX}) * (\text{SUMY}))$$

$$\text{RB} = (\text{NUM} * (\text{SUMXX})) - ((\text{SUMX}) * (\text{SUMX}))$$

$$\text{RC} = (\text{NUM} * (\text{SUMYY})) - ((\text{SUMY}) * (\text{SUMY}))$$

$$\text{RD} = \text{RB} * \text{RC}$$

$$\text{RE} = \text{SQRT}(\text{RD})$$

$$\text{RXY} = \text{RA} / \text{RE}$$

CALCULATE MIN(X) TO MINX

CALCULATE MAX(X) TO MAXX

CALCULATE MIN(YX1) TO MINYX1

CALCULATE MAX(YX1) TO MAXYX1

? 'MEANX'

?? MEANX

? 'VARX'

?? VARX

? 'MEANY'

?? MEANY

? 'VARY'

?? VARY

? 'MINX'

?? MINX

? 'MAXX'

?? MAXX

? 'MINY'

?? MINYX1

? 'MAXY'

?? MAXYX1

? 'CORRE X WITH YX1'

?? RXY

?

? 'MEANX'

?? MEANX

? 'VARX'

?? VARX

? 'MEANY2'

?? MEANY2

? 'VARY2'

?? VARY2

?

? 'MEANX'

?? MEANX

? 'VARX'

?? VARX

? 'MEANY3'

?? MEANY3

? 'VARY3'

?? VARY3

WAIT "

CLEAR

***2

N=1

S=1

CREATE TABLE DATARAN2 (ID N(10),Y1 N(10,6),Y2 N(10,6),X N(10,6),:

```

YX1 N(10,6),YX2 N(10,6), YX3 N(10,6), STA1 N(10,6), CUSTER1 N(10,6),;
STA2 N(10,6), CUSTER2 N(10,6),STA3 N(10,6), CUSTER3 N(10,6),NID N(10)
DELETE ALL
PACK

```

```
@ 19,8 SAY 'PROGRAM "CONSTRUCTING BIVARIATE NORMAL DISTRIBUTION" IS WORKING'
```

```
@ 20,28 SAY '2 ND POPULATION'
```

```
@ 21,30 SAY 'PROCESS TIME'+STR(N)
```

```
DO WHILE N <= ROUND
```

```
DO WHILE S>=1
```

```
  RN=RAND()
```

```
  V1=2*RN - 1
```

```
  RN=RAND()
```

```
  V2=2*RN - 1
```

```
  S=V1*V1 + V2*V2
```

```
ENDDO
```

```
  RNN1 = V1*SQRT ((-2*LOG(S))/S)
```

```
  RNN2 = V2*SQRT ((-2*LOG(S))/S)
```

```
Y11=RNN1
```

```
Y22=RNN2
```

```
XX=50.0000+(SQRT(100.0000)*Y11)
```

```
CRR=SQRT(.15)/SQRT(100.0000)
```

```
RR=CRR*R
```

```
YY=1.6127 + (RR*(XX-50.0000)) + (SQRT(.15*(1-(R*R))) * Y22)
```

```
YY1=2.0000 + (RR*(XX-50.0000)) + (SQRT(.15*(1-(R*R))) * Y22)
```

```
YY2=2.3873 + (RR*(XX-50.0000)) + (SQRT(.15*(1-(R*R))) * Y22)
```

```
USE DATARAN2
```

```
APPEND BLANK
```

REPLACE ID WITH N

REPLACE Y1 WITH RNN1

REPLACE Y2 WITH RNN2

REPLACE X WITH XX

REPLACE Yx2 WITH YY

@ 21,30 SAY 'PROCESS TIME'+STR(N)

N=N+1

S=1

ENDDO

WAIT "

CLEAR

GO TOP

BROW

GO BOTTOM

BROW

GO TOP

COUNT TO NUM

GO TOP

SUMX=0

SUMY=0

SUMXX=0

SUMYY=0

SUMY2=0

SUMYY2=0

SUMY3=0

SUMYY3=0

SUMXY2 = 0

DO WHILE NOT EOF()

**YX1

```

SUMX=SUMX+X
SUMY=SUMY+Yx1
SUMXX=SUMXX+(X*X)
SUMYY=SUMYY+(YX1*YX1)
**YX2
SUMY2=SUMY2+Yx2
SUMYY2=SUMYY2+(YX2*YX2)
SUMXY2=SUMXY2+(X*YX2)
**YX3
SUMY3=SUMY3+Yx3
SUMYY3=SUMYY3+(YX3*YX3)

SKIP

ENDDO
MEANX=SUMX/NUM
MEANY=SUMY/NUM

VARX=((((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1))
VARY=((((NUM*SUMYY)-(SUMY*SUMY)))/(NUM*(NUM-1))

MEANX=SUMX/NUM
MEANY2=SUMY2/NUM

VARX=((((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1))
VARY2=((((NUM*SUMYY2)-(SUMY2*SUMY2)))/(NUM*(NUM-1))

MEANX=SUMX/NUM
MEANY3=SUMY3/NUM

VARX=((((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1))
VARY3=((((NUM*SUMYY3)-(SUMY3*SUMY3)))/(NUM*(NUM-1))

```

$RA = (NUM * (SUMXY2)) - ((SUMX) * (SUMY2))$
 $RB = (NUM * (SUMXX)) - ((SUMX) * (SUMX))$
 $RC = (NUM * (SUMYY2)) - ((SUMY2) * (SUMY2))$
 $RD = RB * RC$
 $RE = \sqrt{RD}$
 $RXY = RA / RE$

CALCULATE MIN(X) TO MINX
 CALCULATE MAX(X) TO MAXX
 CALCULATE MIN(YX2) TO MINY2
 CALCULATE MAX(YX2) TO MAXY2

? 'MEANX'
 ?? MEANX
 ? 'VARX'
 ?? VARX
 ? 'MEANY'
 ?? MEANY
 ? 'VARY'
 ?? VARY
 ?
 ? 'MEANX'
 ?? MEANX
 ? 'VARX'
 ?? VARX
 ? 'MEANY2'
 ?? MEANY2
 ? 'VARY2'
 ?? VARY2
 ? 'MINX'
 ?? MINX
 ? 'MAXX'
 ?? MAXX

? 'MINY2'
 ?? MINY2
 ? 'MAXY2'
 ?? MAXY2
 ? 'CORRE X WITH YX2'
 ?? RXY

?
 ? 'MEANX'
 ?? MEANX
 ? 'VARX'
 ?? VARX
 ? 'MEANY3'
 ?? MEANY3
 ? 'VARY3'
 ?? VARY3

WAIT "

CLEAR

***3

N=1

S=1

CREATE TABLE DATARAN3 (ID N(10),Y1 N(10,6),Y2 N(10,6),X N(10,6),;
 YX1 N(10,6),YX2 N(10,6), YX3 N(10,6), STA1 N(10,6), CUSTER1 N(10,6),;
 STA2 N(10,6), CUSTER2 N(10,6),STA3 N(10,6), CUSTER3 N(10,6),NID N(10))

DELETE ALL

PACK

@ 19,8 SAY 'PROGRAM "CONSTRUCTING BIVARIATE NORMAL DISTRIBUTION" IS WORKING'

@ 20,28 SAY '3 TH POPULATION'

@ 21,30 SAY 'PROCESS TIME'+STR(N)

DO WHILE N <= ROUND

```

DO WHILE S>=1
  RN=RAND()
  V1=2*RN - 1
  RN=RAND()
  V2=2*RN - 1
  S=V1*V1 + V2*V2
ENDDO

RNN1 = V1*SQRT ((-2*LOG(S))/S)
RNN2 = V2*SQRT ((-2*LOG(S))/S)

Y11=RNN1
Y22=RNN2

XX=50.0000+(SQRT(100.0000)*Y11)
CRR=SQRT(.15)/SQRT(100.0000)
RR=CRR*R
YY=1.6127 + (RR*(XX-50.0000)) + (SQRT(.15*(1-(R*R))) * Y22)
YY1=2.0000 + (RR*(XX-50.0000)) + (SQRT(.15*(1-(R*R))) * Y22)
YY2=2.3873 + (RR*(XX-50.0000)) + (SQRT(.15*(1-(R*R))) * Y22)

USE DATARAN3
APPEND BLANK
REPLACE ID WITH N
REPLACE Y1 WITH RNN1
REPLACE Y2 WITH RNN2

REPLACE X WITH XX
REPLACE Yx3 WITH YY2
@ 21,30 SAY 'PROCESS TIME'+STR(N)
N=N+1
S=1
ENDDO

```

WAIT "

CLEAR

GO TOP

BROW

GO BOTTOM

BROW

GO TOP

COUNT TO NUM

GO TOP

SUMX=0

SUMY=0

SUMXX=0

SUMYY=0

SUMY2=0

SUMYY2=0

SUMY3=0

SUMYY3=0

SUMXY3=0

DO WHILE NOT EOF()

**YX1

SUMX=SUMX+X

SUMY=SUMY+Yx1

SUMXX=SUMXX+(X*X)

SUMYY=SUMYY+(YX1*YX1)

**YX2

SUMY2=SUMY2+Yx2

SUMYY2=SUMYY2+(YX2*YX2)

**YX3

SUMY3=SUMY3+Yx3

SUMYY3=SUMYY3+(YX3*YX3)

SUMXY3=SUMXY3+(X*YX3)

SKIP

ENDDO

MEANX=SUMX/NUM

MEANY=SUMY/NUM

VARX=((((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1)))

VARY=((((NUM*SUMYY)-(SUMY*SUMY)))/(NUM*(NUM-1)))

MEANX=SUMX/NUM

MEANY2=SUMY2/NUM

VARX=((((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1)))

VARY2=((((NUM*SUMYY2)-(SUMY2*SUMY2)))/(NUM*(NUM-1)))

MEANX=SUMX/NUM

MEANY3=SUMY3/NUM

VARX=((((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1)))

VARY3=((((NUM*SUMYY3)-(SUMY3*SUMY3)))/(NUM*(NUM-1)))

RA=(NUM*(SUMXY3)) - ((SUMX)*(SUMY3))

RB=(NUM*(SUMXX)) - ((SUMX)*(SUMX))

RC=(NUM*(SUMYY3)) - ((SUMY3)*(SUMY3))

RD=RB*RC

RE=SQRT(RD)

RXY=RA/RE

CALCULATE MIN(X) TO MINX

CALCULATE MAX(X) TO MAXX

CALCULATE MIN(YX3) TO MINY3

CALCULATE MAX(YX3) TO MAXY3

? 'MEANX'

?? MEANX

? 'VARX'

?? VARX

? 'MEANY'

?? MEANY

? 'VARY'

?? VARY

?

? 'MEANX'

?? MEANX

? 'VARX'

?? VARX

? 'MEANY2'

?? MEANY2

? 'VARY2'

?? VARY2

?

? 'MEANX'

?? MEANX

? 'VARX'

?? VARX

? 'MEANY3'

?? MEANY3

? 'VARY3'

?? VARY3

? 'MINX'

?? MINX

? 'MAXX'

?? MAXX

? 'MINY3'



?? MINY3

? 'MAXY3'

?? MAXY3

? 'CORRE X WITH YX3'

?? RXY

WAIT "

CLOSE ALL

CLEAR

RETURN





ภาคผนวก ข

โปรแกรมกำหนดข้อมูลที่ได้จากการจำลองสถานการณ์ของแต่ละคนว่าเป็น
กลุ่มใดและระดับชั้นประเภทใด (PCSTACUS.PRГ)

CLOSE ALL

CLEAR

SET DECIMAL TO 6

USE DATARAN1

GO TOP

BROW

@ 19,15 SAY 'REPLACE STATRA AND CUSTER FROM 1 ST POPULATION'

@ 20,15 SAY 'REPLACE STATRA FROM 1 ST POPULATION'+STR(ID)

CALCULATE STD(YX1) TO STD

CALCULATE AVG(YX1) TO MEAN

GO TOP

DO WHILE NOT EOF()

IF YX1 > MEAN+STD

REPLACE STA1 WITH 3

ENDIF

IF YX1 < MEAN-STD

REPLACE STA1 WITH 1

ENDIF

IF YX1 >= MEAN-STD AND YX1 <= MEAN+STD

REPLACE STA1 WITH 2

ENDIF

@ 20,15 SAY 'REPLACE STATRA FROM 1 ST POPULATION'+STR(ID)

SKIP

ENDDO

WAIT "

GO TOP

BROW

VON=.T.

@ 20,15 SAY 'REPLACE CUSTER FROM 1 ST POPULATION'+STR(ID)

DO WHILE NOT EOF()

```

DO WHILE VON
  SEED=RAND()
  XN=(SEED+3.1415926)**5
  XN=STR(XN,30,13)

  A=SUBSTR(XN,19,1)
  B=SUBSTR(XN,20,1)
  C=SUBSTR(XN,24,1)
  D=SUBSTR(XN,28,1)

  TABCD=ALLTRIM(A)+ALLTRIM(C)
  XABCD=VAL(TABCD)

  IF XABCD >0 AND XABCD <15
    VON=.F.
  ENDIF
ENDDO

REPLACE CUSTER1 WITH XABCD
@ 20,15 SAY 'REPLACE CUSTER FROM 1 ST POPULATION'+STR(ID)
SKIP
VON=.T.
ENDDO
WAIT "
GO TOP
DO WHILE NOT EOF()
  IF STA1=0
    BROW
  ENDIF
  IF CUSTER1=0
    BROW
  ENDIF
  SKIP
ENDDO

```

GO TOP

CALCULATE MIN(STA1) TO MINSTA

CALCULATE MAX(STA1) TO MAXSTA

CLEAR

? 'MINSTA1'

?? MINSTA

? 'MAXSTA1'

?? MAXSTA

CALCULATE MIN(CUSTER1) TO MINCUS

CALCULATE MAX(CUSTER1) TO MAXCUS

? 'MINCUS1'

?? MINCUS

? 'MAXCUS1'

?? MAXCUS

WAIT "

CLEAR

BROW

***2

USE DATARAN2

GO TOP

BROW

@ 19,15 SAY 'REPLACE STATRA AND CUSTER FROM 2 ND POPULATION'

@ 20,15 SAY 'REPLACE STATRA FROM 2 ND POPULATION'+STR(ID)

CALCULATE STD(YX2) TO STD

CALCULATE AVG(YX2) TO MEAN

GO TOP

DO WHILE NOT EOF()

IF YX2 > MEAN+STD

REPLACE STA2 WITH 3

ENDIF

IF YX2 < MEAN-STD

REPLACE STA2 WITH 1

```

ENDIF
IF YX2 >= MEAN-STD AND YX2 <= MEAN+STD
    REPLACE STA2 WITH 2
ENDIF

@ 20,15 SAY 'REPLACE STATRA FROM 2 ND POPULATION'+STR(ID)
SKIP
ENDDO
WAIT "

GO TOP
BROW
VON=.T.
@ 20,15 SAY 'REPLACE CUSTER FROM 2 ND POPULATION'+STR(ID)
DO WHILE NOT EOF()

    DO WHILE VON
        SEED=RAND()
        XN=(SEED+3.1415926)**5
        XN=STR(XN,30,13)
        A=SUBSTR(XN,19,1)
        B=SUBSTR(XN,20,1)
        C=SUBSTR(XN,24,1)
        D=SUBSTR(XN,28,1)

        TABCD=ALLTRIM(A)+ALLTRIM(C)
        XABCD=VAL(TABCD)

        IF XABCD >0 AND XABCD <15
            VON=.F.
        ENDIF
    ENDDO

    REPLACE CUSTER2 WITH XABCD
@ 20,15 SAY 'REPLACE CUSTER FROM 2 ND POPULATION'+STR(ID)

```

```

SKIP
VON=.T.
ENDDO
WAIT "
GO TOP
DO WHILE NOT EOF()
    IF STA2=0
        BROW
    ENDIF
    IF CUSTER2=0
        BROW
    ENDIF
    SKIP
ENDDO

GO TOP
CALCULATE MIN(STA2) TO MINSTA
CALCULATE MAX(STA2) TO MAXSTA
CLEAR
? 'MINSTA2'
?? MINSTA
? 'MAXSTA2'
?? MAXSTA

CALCULATE MIN(CUSTER2) TO MINCUS
CALCULATE MAX(CUSTER2) TO MAXCUS
? 'MINCUS2'
?? MINCUS
? 'MAXCUS2'
?? MAXCUS

WAIT "
CLEAR
BROW
***
***3

```

USE DATARAN3

GO TOP

BROW

@ 19,15 SAY 'REPLACE STATRA AND CUSTER FROM 3 ND POPULATION'

@ 20,15 SAY 'REPLACE STATRA FROM 3 ND POPULATION'+STR(ID)

CALCULATE STD(YX3) TO STD

CALCULATE AVG(YX3) TO MEAN

GO TOP

DO WHILE NOT EOF()

IF YX3 > MEAN+STD

REPLACE STA3 WITH 3

ENDIF

IF YX3 < MEAN-STD

REPLACE STA3 WITH 1

ENDIF

IF YX3 >= MEAN-STD AND YX3 <= MEAN+STD

REPLACE STA3 WITH 2

ENDIF

@ 20,15 SAY 'REPLACE STATRA FROM 3 ND POPULATION'+STR(ID)

SKIP

ENDDO

WAIT "

GO TOP

BROW

VON=.T.

@ 20,15 SAY 'REPLACE CUSTER FROM 3 ND POPULATION'+STR(ID)

DO WHILE NOT EOF()

DO WHILE VON

SEED=RAND()

```

XN=(SEED+3.1415926)**5
XN=STR(XN,30,13)
A=SUBSTR(XN,19,1)
B=SUBSTR(XN,20,1)
C=SUBSTR(XN,24,1)
D=SUBSTR(XN,28,1)

TABCD=ALLTRIM(A)+ALLTRIM(C)
XABCD=VAL(TABCD)

IF XABCD >0 AND XABCD <15
  VON=.F.
ENDIF
ENDDO
REPLACE CUSTER3 WITH XABCD
@ 20,15 SAY 'REPLACE CUSTER FROM 3 ND POPULATION'+STR(ID)
SKIP
VON=.T.
ENDDO
WAIT "
GO TOP
DO WHILE NOT EOF()
  IF STA3=0
    BROW
  ENDIF
  IF CUSTER3=0
    BROW
  ENDIF
  SKIP
ENDDO

GO TOP
CALCULATE MIN(STA3) TO MINSTA
CALCULATE MAX(STA3) TO MAXSTA

```

CLEAR

? 'MINSTA3'

?? MINSTA

? 'MAXSTA3'

?? MAXSTA

CALCULATE MIN(CUSTER3) TO MINCUS

CALCULATE MAX(CUSTER3) TO MAXCUS

? 'MINCUS3'

?? MINCUS

? 'MAXCUS3'

?? MAXCUS

WAIT "

CLEAR

BROW

@ 19,15 SAY 'REPLACE STATRA AND CUSTER FROM 3 ND POPULATION'

@ 20,15 SAY 'REPLACE STATRA FROM 3 ND POPULATION'+STR(ID)

WAIT ' END PROGRAM'

CLEAR

CLOSE ALL

RETURN

ภาคผนวก ค

โปรแกรมสู่มตัวอย่างแบบหลายขั้นตอน จัดการข้อมูลสูญหายแบบอีพีเอสเอสอี
คำนวณค่าความแม่นยำและอำนาจการทดสอบ (PMULEP44.PRГ)

มหาวิทยาลัยนเรศวร

***PROGRAM MULTI-STAGE SAMPLING

***METHOD OF EPSSE MISSING DATA TREATMENT METHOD

CLOSE ALL

CLEAR

SET TALK OFF

SET SAFETY OFF

DFVT=1.98

DFVF=3.02

NUMMISS=.20

XMISS=NUMMISS*100

SAMPLE=350

CREATE TABLE CUSSTA (NIDCUS N(10),NCUS N(10),NUMCUS N(10),;
CSTA1 N(10),CSTA2 N(10), CSTA3 N(10))

USE CUSSTA

INDEX ON NCUS TO CUSSTA

USE CUSSTA INDEX CUSSTA

USE

CREATE TABLE DCUSSTA (IDX N(10),IDP N(10),NIDX N(10),IDN N(10),XX N(10,6),;
YYX1 N(10,6),CUS N(10),STA N(10),IDMISS N(10),MISS N(5),GROUP N(1),IDEE N(10))

CREATE TABLE DCSTOTAL (NUM N(10),MX N(10,6),VX N(10,6),MEANY1 N(10,6),;
VARY1 N(10,6),CORXY1 N(10,6),AM N(10,6),AV N(10,6),AR N(10,6),;
WTR N(10,6),NRHY N(10,6),VTT1 N(10,6),NT1HY N(10,6),;
VTT2 N(10,6),NT2HY N(10,6),VTF N(10,6),NTFHY N(10,6),;
NUMG1 N(10),NUMG2 N(10),NUMG3 N(10))

USE DCSTOTAL

DELETE ALL

PACK

SELECT 4

USE DATARAN1

CLEAR

GO TOP

BROW

GO BOTTOM

BROW

GO TOP

COUNT TO NUM

GO TOP

SUMX=0

SUMY=0

SUMXX=0

SUMYY=0

SUMY2=0

SUMYY2=0

SUMY3=0

SUMYY3=0

SUMXY=0

DO WHILE NOT EOF()

**YX1

SUMX=SUMX+X

SUMY=SUMY+Yx1

SUMXX=SUMXX+(X*X)

SUMYY=SUMYY+(YX1*YX1)

**YX2

SUMY2=SUMY2+Yx2

SUMYY2=SUMYY2+(YX2*YX2)

**YX3

SUMY3=SUMY3+Yx3

SUMYY3=SUMYY3+(YX3*YX3)

SUMXY = SUMXY+ (X*YX1)

SKIP

ENDDO

MEANX=SUMX/NUM

MEANY=SUMY/NUM

VARX=(((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1))

VARY=(((NUM*SUMYY)-(SUMY*SUMY)))/(NUM*(NUM-1))

MEANX=SUMX/NUM

MEANY2=SUMY2/NUM

VARX=(((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1))

VARY2=(((NUM*SUMYY2)-(SUMY2*SUMY2)))/(NUM*(NUM-1))

MEANX=SUMX/NUM

MEANY3=SUMY3/NUM

VARX=(((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1))

VARY3=(((NUM*SUMYY3)-(SUMY3*SUMY3)))/(NUM*(NUM-1))

RA=(NUM*(SUMXY)) - ((SUMX)*(SUMY))

RB=(NUM*(SUMXX)) - ((SUMX)*(SUMX))

RC=(NUM*(SUMYY)) - ((SUMY)*(SUMY))

RD=RB*RC

RE=SQRT(RD)

RXY=RA/RE

PMY =MEANY

PVY =VARY

PCORXY=RXY

? 'MEANX'

?? MEANX

? 'VARX '

?? VARX

? 'MEANY'

?? MEANY

? 'VARY'

?? VARY

? 'COR XY'

?? RXY

WAIT "

CLEAR

XLOOP=1

@ 10,18 SAY 'WORKING ABOUT MULTI STAGE SAMPLING '

@ 21,30 SAY 'PROCESS TIME '+ALLTRIM(STR(XLOOP))

DO WHILE XLOOP <=1000

CLOSE ALL

@ 10,18 SAY 'WORKING ABOUT MULTI STAGE SAMPLING'

@ 17,18 SAY 'PROGRAM MULTI STAGE SAMPLING IS WORKING'

@ 18,18 SAY 'WITH NUMBER OF MISSING IS '+ALLTRIM(STR(XMISS))+ '%'+ ' AND CORRELATION
IS '+ (STR(PCORXY,3,3))

@ 19,18 SAY 'AND EPSSE MISSING DATA TREATMENT METHOD'

@ 21,30 SAY 'PROCESS TIME '+ALLTRIM(STR(XLOOP))

@ 10,18 SAY 'WORKING ABOUT SAMPLE NUMBER OF CUSTER AND STRATA'

SELECT 3

USE CUSSTA INDEX CUSSTA

DELETE ALL

PACK

USE CUSSTA INDEX CUSSTA

NC=1

VON=1

DO WHILE VON <= 7

SEED=RAND()

```

XN=(SEED+3.1415926)**5
XN=STR(XN,30,13)
A=SUBSTR(XN,19,1)
B=SUBSTR(XN,20,1)
C=SUBSTR(XN,24,1)
D=SUBSTR(XN,28,1)

TABCD=ALLTRIM(A)+ALLTRIM(C)
XABCD=VAL(TABCD)

IF (XABCD >= 1 AND XABCD <= 14)
  USE CUSSTA INDEX CUSSTA
  SEEK XABCD
  IF FOUND()
    VON=VON
  ELSE
    APPEND BLANK
    REPLACE NIDCUS WITH NC
    REPLACE NCUS WITH XABCD
    NC=NC+1
    VON=VON+1
  ENDIF
ENDIF

ENDDO
SELECT 3
GO TOP
N=NCUS
SUM=0

DO WHILE NOT EOF()
  SELECT 4
  USE DATARAN1
  CALCULATE STD(YX1) TO STDY FOR CUSTER1=N

```

COUNT TO NUM FOR CUSTER1=N

SUM=SUM+(STDY*NUM)

SELECT 3

SKIP

N=NCUS

ENDDO

GO TOP

N=NCUS

DO WHILE NOT EOF()

SELECT 4

CALCULATE STD(YX1) TO STDY FOR CUSTER1=N

COUNT TO NUM FOR CUSTER1=N

NNS=SAMPLE*NUM*STDY

NH=NNS/SUM

NNH=ALLTRIM(STR(NH))

NH=VAL(NNH)

SELECT 3

REPLACE NUMCUS WITH NH

SKIP

N=NCUS

ENDDO

GO TOP

N=NCUS

NSAM=NUMCUS

SUM=0

DO WHILE NOT EOF()

SSTA=1

DO WHILE SSTA <= 3

SELECT 4

USE DATARAN1

CALCULATE STD(YX1) TO STDY FOR CUSTER1=N AND STA1=SSTA

```

COUNT TO NUM FOR CUSTER1=N AND STA1=SSTA
SUM=SUM+(STDY*NUM)
SSTA=SSTA+1
ENDDO
SSTA=1
DO WHILE SSTA<=3
  SELECT 4
  USE DATARAN1
  CALCULATE STD(YX1) TO STDY FOR CUSTER1=N AND STA1=SSTA
  COUNT TO NUM FOR CUSTER1=N AND STA1=SSTA
  NNS=NSAM*NUM*STDY
  NH =NNS/SUM
  NNH=ALLTRIM(STR(NH))
  NH =VAL(NNH)
  SELECT 3
  IF SSTA=1
    REPLACE CSTA1 WITH NH
  ENDIF
  IF SSTA=2
    REPLACE CSTA2 WITH NH
  ENDIF
  IF SSTA=3
    REPLACE CSTA3 WITH NH
  ENDIF
  SSTA=SSTA+1
ENDDO
SELECT 3
SKIP
N=NCUS
NSAM=NUMCUS
SUM=0
ENDDO

***SELECT 3

```

```

USE CUSSTA
GO TOP
DO WHILE NOT EOF()
@ 10,18 SAY '
@ 10,18 SAY 'WORKING ABOUT DATACS1'

```

```

XIDC=NIDCUS
XCUS=NCUS
XNUM=NUMCUS
SLOOP=ALLTRIM(STR(XCUS))
DAT='DAT'+SLOOP
SSTA=1
DO WHILE SSTA <= 3
  SELECT 4
  USE DATARAN1
  COPY TO &DAT FOR CUSTER1=XCUS AND STA1=SSTA
  SELECT 1
  USE &DAT
  COUNT TO NN
  GO TOP

```

```

N=0
DO WHILE NOT EOF()
  N=N+1
  REPLACE NID WITH N
  SKIP
ENDDO
GO BOTTOM

```

*** SAMPLING STATIFIED IN GIVEN CUSTER

```

NN=NN
SELECT 3
IF SSTA=1
  N =CSTA1

```

```

ENDIF
IF SSTA=2
    N=CSTA2
ENDIF
IF SSTA=3
    N=CSTA3
ENDIF
K=NN/N

SELECT 2
USE DCUSSTA
***IDX N:10 IDP N:10 NIDX N:10:6 XX:10:6 YYX1:10:6 STA:10
INDEX ON NIDX TO IDNID
USE DCUSSTA INDEX IDNID
DELETE ALL
PACK
SELECT 2
USE DCUSSTA INDEX IDNID
@ 12,18 SAY 'WORKING ABOUT SYSTEMATIC RANDOM NUMBER'
ACT=.T.
DO WHILE ACT
    SEED=RAND()
    X=(SEED+3.1415926)**5
    X=STR(X,30,13)
    A=SUBSTR(X,19,1)
    B=SUBSTR(X,20,1)
    C=SUBSTR(X,23,1)
    D=SUBSTR(X,25,1)
    E=SUBSTR(X,28,1)

    TABCD=ALLTRIM(A)+ALLTRIM(B)+ALLTRIM(C)+ALLTRIM(D)+ALLTRIM(E)
    XABCD=VAL(TABCD)
    IF (XABCD >= 1 AND XABCD <= NN)
        ACT=.F.
    
```

```

ELSE
  ACT=.T.
ENDIF

ENDDO

J=0
DO WHILE J<=N-1
  NUMX = XABCD+(J*K)
  NUMXX=STR(NUMX)

  NUMX =VAL(NUMXX)
  IF NUMX > NN
    NUMXX=NUMX-NN
    NUMX =NUMXX
  ENDIF

  SELECT 2
  **FILE DCUSSTA
  SEEK NUMX
  IF FOUND()
    ? 'REPEAT1 CODE'
    ? NUMX
    ? XABCD
    ? J
    ? K

    WAIT "
    BROW
  ELSE
    APPEND BLANK
    J=J+1
    REPLACE IDX WITH J
    REPLACE NIDX WITH NUMX
  
```

```

ENDIF
ENDDO
GO TOP
GO BOTTOM

SELECT 1
USE &DAT
INDEX ON NID TO IDS
USE &DAT INDEX IDS
GO TOP
SELECT 2
USE DCUSSTA
INDEX ON NIDX TO IDNID
USE DCUSSTA INDEX IDNID
GO TOP
RC=NIDX
@ 12,18 SAY '
@ 12,18 SAY 'WORKING ABOUT SYSTEMATIC RANDOM SAMPLING IN STRATA'
DO WHILE NOT EOF()

SELECT 1
SEEK RC
IF FOUND()
DX= X
DY= YX1
DID=NID
XID=ID
XCUS=CUSTER1
XSTA=STA1
SELECT 2
REPLACE XX WITH DX
REPLACE YYX1 WITH DY

REPLACE IDN WITH DID
REPLACE IDP WITH XID
REPLACE CUS WITH XCUS

```

REPLACE STA WITH XSTA

```

ELSE
  ? 'NO CODE FOUND'
  ? RC
  SELECT 2
  WAIT "
ENDIF
SELECT 2
SKIP
RC=NIDX
ENDDO
USE DCUSSTA
SORT ON IDX TO DCSS
USE DCSS
DTCS='DTCS'+ALLTRIM(STR($STA))
COPY TO &DTCS
USE &DTCS
$STA=$STA+1
SELECT 1
USE
ENDDO

USE DTCS1
APPEND FROM DTCS2
APPEND FROM DTCS3
USE DTCS1
GO TOP

```

SSLOOP=ALLTRIM(STR(XIDC))

DTCST='DTCST'+SSLOOP

COPY TO &DTCST

USE &DTCST

GO TOP

SELECT 3

```

SKIP

ENDDO

SELECT 8
USE DTCST1
LOOP=2
DO WHILE LOOP <= VAL(SSLOOP)
    NX =ALLTRIM(STR(LOOP))
    DTCST='DTCST'+NX
    APPEND FROM &DTCST
    LOOP=LOOP+1
ENDDO

SELECT 8
USE DTCST1
INDEX ON IDP TO IDIDP
USE DTCST1 INDEX IDIDP

GO TOP
XIDP=0
@ 12,18 SAY '
@ 12,18 SAY 'WORKING ABOUT CHECKING REPEAT CODE'
DO WHILE NOT EOF()
    IF XIDP=IDP
        ? 'REPETE CODE'
        WAIT"
    ELSE
    ENDIF
    XIDP=IDP
SKIP
ENDDO

***BEGIN EPSSE MISSING DATA TREATMENT METHOD

```

***NUMBER OF MISSING DATA EQUAL 5%

@ 12,18 SAY '

@ 12,18 SAY 'WORKING ABOUT EPSSE MISSING DATA TREATMENT METHOD'

SELECT 8

USE DTCST1

GO TOP

COUNT TO NUM

GO TOP

*** CHANGE NUMBER OF MISSING

NMISS=NUM*NUMMISS

N=0

DO WHILE NOT EOF()

 N=N+1

 REPLACE IDMISS WITH N

 SKIP

ENDDO

SELECT 8

USE DTCST1

INDEX ON IDMISS TO IDMISS

USE DTCST1 INDEX IDMISS

GO TOP

NM=1

DO WHILE NM <= NMISS

 SEED=RAND()

 X=(SEED+3.1415926)**5

 X=STR(X,30,13)

 A=SUBSTR(X,19,1)

 B=SUBSTR(X,20,1)

 C=SUBSTR(X,23,1)

 D=SUBSTR(X,25,1)

```
E=SUBSTR(X,28,1)
```

```
TABCD=ALLTRIM(A)+ALLTRIM(B)+ALLTRIM(C)+ALLTRIM(D)+ALLTRIM(E)
```

```
XABCD=VAL(TABCD)
```

```
SEEK XABCD
```

```
IF FOUND()
```

```
  IF MISS = 99999
```

```
  ELSE
```

```
    REPLACE MISS WITH 99999
```

```
    NM=NM+1
```

```
  ENDIF
```

```
ENDIF
```

```
ENDDO
```

```
***ADD PROGRAM
```

```
***PROGRAM EPSSE TREATMENT MISSING DATA METHOD
```

```
*** CHANGE DATA FOR OTHERS PROGRAM
```

```
SELECT 8
```

```
USE DTCST1
```

```
COPY TO DTMI FOR MISS # 99999
```

```
USE
```

```
SELECT 5
```

```
USE DTMI
```

```
COUNT TO NN
```

```
GO TOP
```

```
N=0
```

```
DO WHILE NOT EOF()
```

```
  N=N+1
```

```
  REPLACE IDEE WITH N
```

```
  SKIP
```

```
ENDDO
```

```

USE DTMI
INDEX ON IDEE TO IDEE
USE DTMI INDEX IDEE
GO TOP

```

```

NN=NN
N=NN/10
NN3=9*N
N=NN3
K=NN/N
NUMID=1
VONE=1

```

```

CREATE TABLE DATAE (IDE N(10),IDPE N(10),NIDE N(10),XXE N(10,6).;
                      YYX1E N(10,6))

```

```

USE

```

```

@ 12,18 SAY '

```

```

@ 12,18 SAY 'WORKING ABOUT EPSSE TREATEMENT MISSING DATA METHOD'

```

```

@ 13,18 SAY 'FINDING STATISTIC VALUE '+'ROUND'+ ' '+ALLTRIM(STR(VONE))

```

```

SUMMY  = 0

```

```

SUMMX  = 0

```

```

SUMSDY = 0

```

```

SUMSDX = 0

```

```

SUMCOVXY = 0

```

```

SUMVX  = 0

```

```

SUMVY  = 0

```

```

DO WHILE VONE <= 1000

```

```

    SELECT 6

```

```

    USE DATAE

```

```

    INDEX ON NIDE TO NIDE

```

```

    USE DATAE INDEX NIDE

```

DELETE ALL

PACK

USE DATAE INDEX NIDE

@ 12,18 SAY '

@ 12,18 SAY 'WORKING ABOUT EPSSE TREATMENT MISSING DATA METHOD'

@ 13,18 SAY 'FINDING STATISTIC VALUE '+'ROUND'+ ' '+ALLTRIM(STR(VONE))

ACT=.T.

DO WHILE ACT

SEED=RAND()

X=(SEED+3.1415926)**5

X=STR(X,30,13)

A=SUBSTR(X,19,1)

B=SUBSTR(X,20,1)

C=SUBSTR(X,23,1)

D=SUBSTR(X,25,1)

E=SUBSTR(X,28,1)

TABCD=ALLTRIM(A)+ALLTRIM(B)+ALLTRIM(C)+ALLTRIM(D)+ALLTRIM(E)

XABCD=VAL(TABCD)

IF (XABCD >= 1 AND XABCD <= NN)

ACT=.F.

ELSE

ACT=.T.

ENDIF

ENDDO

J=0

DO WHILE J<N-1

NUMX=XABCD+(J*K)

NUMXX=STR(NUMX)

NUMX=VAL(NUMXX)

IF NUMX > NN

```

NUMXX=NUMX-NN
NUMX=NUMXX
ENDIF

```

```

SELECT 6

```

```

**FILE DATAE
SEEK NUMX
IF FOUND()
ELSE
APPEND BLANK
J=J+1
REPLACE IDE WITH J
REPLACE NIDE WITH NUMX
ENDIF
ENDDO

```

```

SELECT 5
USE DTMI
INDEX ON IDEE TO IDEE
USE DTMI INDEX IDEE
SELECT 6
USE DATAE
INDEX ON NIDE TO NIDE
USE DATAE INDEX NIDE
GO TOP
RC=NIDE

```

```

@ 12,18 SAY '

```

```

@ 12,18 SAY 'WORKING ABOUT EPSSE TREATEMENT MISSING DATA METHOD'

```

```

DO WHILE NOT EOF()

```

```

SELECT 5
SEEK RC

```

IF FOUND()

XE= XX

YE= YYX1

IDPPE=IDP

SELECT 6

REPLACE XXE WITH XE

REPLACE YYX1E WITH YE

REPLACE IDPE WITH IDPPE

ELSE

? 'NO CODE FOUND'

? RC

SELECT 6

WAIT "

ENDIF

SELECT 6

SKIP

RC=NIDE

ENDDO

USE DATAE

CALCULATE AVG(YYX1E) TO MY

CALCULATE AVG(XXE) TO MX

CALCULATE STD(YYX1E) TO SDY

CALCULATE STD(XXE) TO SDX

COUNT TO NUM

GO TOP

SUMDXY=0

SUMEX =0

SUMEY =0

SUMEXX =0

SUMEYY =0

DO WHILE NOT EOF()

SUMEX =SUMEX +(XXE)

SUMEY =SUMEY +(YYX1E)

SUMEXX=SUMEXX+(XXE*XXE)

SUMEYY=SUMEYY+(YYX1E*YYX1E)

SUMDXY=SUMDXY+((XXE-MX)*(YYX1E-MY))

SKIP

ENDDO

VAREX = ((NUM*SUMEXX)-(SUMEX*SUMEX))/(NUM*(NUM-1))

VAREY = ((NUM*SUMEYY)-(SUMEY*SUMEY))/(NUM*(NUM-1))

COVXY=SUMDXY/(NUM-1)

SUMMY = SUMMY+MY

SUMMX = SUMMX+MX

SUMSDY = SUMSDY+SDY

SUMSDX = SUMSDX+SDX

SUMCOVXY = SUMCOVXY+COVXY

SUMVX = SUMVX +VAREX

SUMVY = SUMVY +VAREY

NUMID=NUMID+1

VONE=VONE+1

ENDDO

MMEX=SUMMX/(VONE-1)

MMEY=SUMMY/(VONE-1)

SSDVX=SUMSDX/(VONE-1)

SSDVY=SUMSDY/(VONE-1)

CCOXY=SUMCOVXY/(VONE-1)

VARIX=SUMVX/(VONE-1)

```

VARIY=SUMVY/(VONE-1)
SDVX =SQRT(VARIX)
SDVY =SQRT(VARIY)
*RXY = (CCOXY)/(SSDVX*SSDVY)
RXY = (CCOXY)/(SDVX*SDVY)
B = RXY * (SSDVY / SSDVX)
A = MMEY -(B * MMEX)

*** CHANGE DATA FOR OTHERS PROGRAM
SELECT 8
USE DTCST1
GO TOP
DO WHILE NOT EOF()
  IF MISS=99999
    XX1=XX
    TM=A + (B * XX1)
    REPLACE YYX1 WITH TM
  ENDIF
  SKIP
ENDDO
USE

VON2=1
CREATE TABLE DATASE (ID N(10),ERROR N(10,6),INTERC N(10,6),COEFFI N(10,6))
USE
@ 14,18 SAY 'FINDING SMALLEST STANDARD ERROR EQUATION '+'ROUND'+ ' '+ALLTRIM(STR
(VON2))

DO WHILE VON2 <= 100
@ 14,18 SAY 'FINDING SMALLEST STANDARD ERROR EQUATION '+'ROUND'+ ' '+ALLTRIM(STR
(VON2))
*** CHANGE DATA FOR OTHERS PROGRAM
SELECT 8
USE DTCST1

```



```

COUNT TO NUM
GO TOP
SUMX=0
SUMY=0
SUMXX=0
SUMYY=0
SUMXY=0
DO WHILE NOT EOF()
  SUMX=SUMX+XX
  SUMY=SUMY+YYX1
  SUMXX=SUMXX+(XX*XX)
  SUMYY=SUMYY+(YYX1*YYX1)
  SUMXY=SUMXY+(XX*YYX1)
  SKIP
ENDDO
MY=SUMY/NUM
MX=SUMX/NUM
VARX=((((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1))
VARY=((((NUM*SUMYY)-(SUMY*SUMY)))/(NUM*(NUM-1))
SDY=SQRT(VARY)
SDX=SQRT(VARX)
RA=(NUM*(SUMXY)) - ((SUMX)*(SUMY))
RB=(NUM*(SUMXX)) - ((SUMX)*(SUMX))
RC=(NUM*(SUMYY)) - ((SUMY)*(SUMY))
RD=RB*RC
RE=SQRT(RD)
RXY=RA/RE
SE=SDY*(SQRT(1-(RXY*RXY)))
B = RXY * (SDY/SDX)
A = MY - (B * MX)
SELECT 7
USE DATASE
APPEND BLANK
REPLACE ID WITH VON2

```

REPLACE SERROR WITH SE
REPLACE INTERC WITH A
REPLACE COEFFI WITH B

SELECT 8

USE DTCST1

GO TOP

DO WHILE NOT EOF()

IF MISS=99999

XX1=XX

TM =A+(B*XX1)

REPLACE YYX1 WITH TM

ENDIF

SKIP

ENDDO

VON2=VON2+1

ENDDO

SELECT 7

USE DATASE

INDEX ON SERROR TO SERROR

USE DATASE INDEX SERROR

GO TOP

A=INTERC

B=COEFFI

SELECT 8

USE DTCST1

GO TOP

DO WHILE NOT EOF()

IF MISS=99999

XX1=XX

TM =A+(B*XX1)

REPLACE YYX1 WITH TM

ENDIF

SKIP

ENDDO

SELECT 8

USE DTCST1 INDEX IDMISS

COPY TO DATACS1

USE DATACS1

REPLACE ALL GROUP WITH 1

CLOSE ALL

@ 12,18 SAY '

@ 13,18 SAY '

@ 14,18 SAY '

***SAMPLING YX2

@ 12,18 SAY '

@ 10,18 SAY '

@ 10,18 SAY 'WORKING ABOUT SAMPLE NUMBER OF CUSTER AND STRATA'

SELECT 3

USE CUSSTA INDEX CUSSTA

DELETE ALL

PACK

USE CUSSTA INDEX CUSSTA

NC=1

VON=1

DO WHILE VON <= 7

SEED=RAND()

XN=(SEED+3.1415926)**5

XN=STR(XN,30,13)

A=SUBSTR(XN,19,1)

B=SUBSTR(XN,20,1)

C=SUBSTR(XN,24,1)

D=SUBSTR(XN,28,1)

TABCD=ALLTRIM(A)+ALLTRIM(C)

XABCD=VAL(TABCD)

IF (XABCD >= 1 AND XABCD <= 14)

USE CUSSTA INDEX CUSSTA

SEEK XABCD

IF FOUND()

VON=VON

ELSE

APPEND BLANK

REPLACE NIDCUS WITH NC

REPLACE NCUS WITH XABCD

NC=NC+1

VON=VON+1

ENDIF

ENDIF

ENDDO

SELECT 3

GO TOP

N=NCUS

SUM=0

DO WHILE NOT EOF()

SELECT 4

USE DATARAN2

CALCULATE STD(YX2) TO STDY FOR CUSTER2=N

COUNT TO NUM FOR CUSTER2=N

SUM=SUM+(STDY*NUM)

SELECT 3

SKIP

```

N=NCUS
ENDDO
GO TOP
N=NCUS
DO WHILE NOT EOF()
    SELECT 4
    CALCULATE STD(YX2) TO STDY FOR CUSTER2=N
    COUNT TO NUM FOR CUSTER2=N
    NNS=SAMPLE*NUM*STDY
    NH=NNS/SUM
    NNH=ALLTRIM(STR(NH))
    NH=VAL(NNH)
    SELECT 3
    REPLACE NUMCUS WITH NH
    SKIP
    N=NCUS
ENDDO
GO TOP
N=NCUS
NSAM=NUMCUS
SUM=0
DO WHILE NOT EOF()

SSTA=1

DO WHILE SSTA <= 3
    SELECT 4
    USE DATARAN2
    CALCULATE STD(YX2) TO STDY FOR CUSTER2=N AND STA2=SSTA
    COUNT TO NUM FOR CUSTER2=N AND STA2=SSTA
    SUM=SUM+(STDY*NUM)
    SSTA=SSTA+1
ENDDO
SSTA=1

```

```

DO WHILE SSTA<=3
  SELECT 4
  USE DATARAN2
  CALCULATE STD(YX2) TO STDY FOR CUSTER2=N AND STA2=SSTA
  COUNT TO NUM FOR CUSTER2=N AND STA2=SSTA
  NNS=NSAM*NUM*STDY
  NH =NNS/SUM
  NNH=ALLTRIM(STR(NH))
  NH =VAL(NNH)
  SELECT 3
  IF SSTA=1
    REPLACE CSTA1 WITH NH
  ENDIF
  IF SSTA=2
    REPLACE CSTA2 WITH NH
  ENDIF
  IF SSTA=3
    REPLACE CSTA3 WITH NH
  ENDIF
  SSTA=SSTA+1
ENDDO

  SELECT 3
  SKIP
  N=NCUS
  NSAM=NUMCUS
  SUM=0
ENDDO

***SELECT 3
USE CUSSTA
GO TOP
DO WHILE NOT EOF()
@ 10,18 SAY '
@ 10,18 SAY 'WORKING ABOUT DATA2S2'

```

```

XIDC=NIDCUS
XCUS=NCUS
XNUM=NUMCUS
SLOOP=ALLTRIM(STR(XCUS))
DAT='DAT'+SLOOP
SSTA=1
DO WHILE SSTA <= 3
    SELECT 4
    USE DATARAN2
    COPY TO &DAT FOR CUSTER2=XCUS AND STA2=SSTA
    SELECT 1
    USE &DAT
    COUNT TO NN
    GO TOP
N=0
DO WHILE NOT EOF()
    N=N+1
    REPLACE NID WITH N
    SKIP
ENDDO
GO BOTTOM

*** SAMPLING STATIFIED IN GIVEN CUSTER
NN=NN
SELECT 3
IF SSTA=1
    N =CSTA1
ENDIF
IF SSTA=2
    N =CSTA2
ENDIF
IF SSTA=3
    N =CSTA3

```

ENDIF

K=NN/N

SELECT 2

USE DCUSSTA

***IDX N:10 IDP N:10 NIDX N:10:6 XX:10:6 YYX1:10:6 STA:10

INDEX ON NIDX TO IDNID

USE DCUSSTA INDEX IDNID

DELETE ALL

PACK

SELECT 2

USE DCUSSTA INDEX IDNID

@ 12,18 SAY 'WORKING ABOUT SYSTEMATIC RANDOM NUMBER'

ACT=.T.

DO WHILE ACT

SEED=RAND()

X=(SEED+3.1415926)**5

X=STR(X,30,13)

A=SUBSTR(X,19,1)

B=SUBSTR(X,20,1)

C=SUBSTR(X,23,1)

D=SUBSTR(X,25,1)

E=SUBSTR(X,28,1)

TABCD=ALLTRIM(A)+ALLTRIM(B)+ALLTRIM(C)+ALLTRIM(D)+ALLTRIM(E)

XABCD=VAL(TABCD)

IF (XABCD >= 1 AND XABCD <= NN)

ACT=.F.

ELSE

ACT=.T.

ENDIF

ENDDO

```

J=0
DO WHILE J<=N-1
    NUMX = XABCD+(J*K)
    NUMXX=STR(NUMX)

    NUMX =VAL(NUMXX)
    IF NUMX > NN
        NUMXX=NUMX-NN
        NUMX =NUMXX
    ENDIF

    SELECT 2
        **FILE DCUSSTA
        SEEK NUMX
        IF FOUND()
            ? 'REPEAT1 CODE'
            ? NUMX
            ? XABCD
            ? J
            ? K
            WAIT "
            BROW
            ELSE
            APPEND BLANK
            J=J+1
            REPLACE IDX WITH J
            REPLACE NIDX WITH NUMX
        ENDIF
    ENDDO
    GO TOP
    GO BOTTOM
SELECT 1
USE &DAT

```

```

INDEX ON NID TO IDS
USE &DAT INDEX IDS
GO TOP
SELECT 2
USE DCUSSTA
INDEX ON NIDX TO IDNID
USE DCUSSTA INDEX IDNID
GO TOP
RC=NIDX
@ 12,18 SAY '
@ 12,18 SAY 'WORKING ABOUT SYSTEMATIC RANDOM SAMPLING IN STRATA'
DO WHILE NOT EOF()

SELECT 1
SEEK RC
IF FOUND()
DX= X
DY= YX2
DID=NID
XID=ID
XCUS=CUSTER2
XSTA=STA2
SELECT 2
REPLACE XX WITH DX
REPLACE YYX1 WITH DY
REPLACE IDN WITH DID
REPLACE IDP WITH XID
REPLACE CUS WITH XCUS
REPLACE STA WITH XSTA

ELSE
? 'NO CODE FOUND'
? RC
SELECT 2
WAIT "
```

```

ENDIF
SELECT 2
SKIP
RC=NIDX
ENDDO
USE DCUSSTA
SORT ON IDX TO DCSS
USE DCSS
DTCS='DTCS'+ALLTRIM(STR(SSTA))
COPY TO &DTCS
USE &DTCS
SSTA=SSTA+1
SELECT 1
USE
ENDDO

USE DTCS1
APPEND FROM DTCS2
APPEND FROM DTCS3
USE DTCS1
SSLOOP=ALLTRIM(STR(XIDC))
DTCST='DTCST'+SSLOOP
COPY TO &DTCST
USE &DTCST
SELECT 3
SKIP

ENDDO

SELECT 8
USE DTCST1
LOOP=2
DO WHILE LOOP <= VAL(SSLOOP)
  NX =ALLTRIM(STR(LOOP))

```

```

DTCST='DTCST'+NX
APPEND FROM &DTCST
LOOP=LOOP+1
ENDDO

```

```

SELECT 8
USE DTCST1
INDEX ON IDP TO IDIDP
USE DTCST1 INDEX IDIDP

```

```

GO TOP
XIDP=0
@ 12,18 SAY '
@ 12,18 SAY 'WORKING ABOUT CHECKING REPEAT CODE'
DO WHILE NOT EOF()
  IF XIDP=IDP
    ? 'REPETE CODE'
    WAIT"
  ELSE
    ENDIF
  XIDP=IDP
SKIP
ENDDO

```

```

***BEGIN EPSSE MISSING DATA TREATEMENT METHOD
***NUMBER OF MISSING DATA EQUAL 5%
@ 12,18 SAY '
@ 12,18 SAY 'WORKING ABOUT EPSSE MISSING DATA TREATEMENT METHOD'
SELECT 8
USE DTCST1
GO TOP
COUNT TO NUM
GO TOP
*** CHANGE NUMBER OF MISSING

```

NMISS=NUM*NUMMISS

N=0

DO WHILE NOT EOF()

 N=N+1

 REPLACE IDMISS WITH N

 SKIP

ENDDO

SELECT 8

USE DTCST1

INDEX ON IDMISS TO IDMISS

USE DTCST1 INDEX IDMISS

GO TOP

NM=1

DO WHILE NM <= NMISS

 SEED=RAND()

 X=(SEED+3.1415926)**5

 X=STR(X,30,13)

 A=SUBSTR(X,19,1)

 B=SUBSTR(X,20,1)

 C=SUBSTR(X,23,1)

 D=SUBSTR(X,25,1)

 E=SUBSTR(X,28,1)

 TABCD=ALLTRIM(A)+ALLTRIM(B)+ALLTRIM(C)+ALLTRIM(D)+ALLTRIM(E)

 XABCD=VAL(TABCD)

 SEEK XABCD

 IF FOUND()

 IF MISS = 99999

```

ELSE
    REPLACE MISS WITH 99999
    NM=NM+1
ENDIF
ENDIF
ENDDO

***ADD PROGRAM
***PROGRAM EPSSE TREATMENT MISSING DATA METHOD
*** CHANGE DATA FOR OTHERS PROGRAM
SELECT 8
USE DTCST1
COPY TO DTMI FOR MISS # 99999
USE

SELECT 5
USE DTMI
COUNT TO NN
GO TOP
N=0

DO WHILE NOT EOF()
    N=N+1
    REPLACE IDEE WITH N
    SKIP
ENDDO

USE DTMI
INDEX ON IDEE TO IDEE
USE DTMI INDEX IDEE
GO TOP

NN=NN
N=NN/10
NN3=9*N

```

N=NN3

K=NN/N

NUMID=1

VONE=1

CREATE TABLE DATAE (IDE N(10),IDPE N(10),NIDE N(10),XXE N(10,6),;
YYX1E N(10,6))

USE

@ 12,18 SAY '

@ 12,18 SAY 'WORKING ABOUT EPSSE TREATMENT MISSING DATA METHOD'

@ 13,18 SAY 'FINDING STATISTIC VALUE '+'ROUND'+' '+ALLTRIM(STR(VONE))

SUMMY = 0

SUMMX = 0

SUMSDY = 0

SUMSDX = 0

SUMCOVXY = 0

SUMVX = 0

SUMVY = 0

DO WHILE VONE <= 1000

SELECT 6

USE DATAE

INDEX ON NIDE TO NIDE

USE DATAE INDEX NIDE

DELETE ALL

PACK

USE DATAE INDEX NIDE

@ 12,18 SAY '

@ 12,18 SAY 'WORKING ABOUT EPSSE TREATMENT MISSING DATA METHOD'

@ 13,18 SAY 'FINDING STATISTIC VALUE '+'ROUND'+' '+ALLTRIM(STR(VONE))

ACT=.T.

DO WHILE ACT

SEED=RAND()

X=(SEED+3.1415926)**5

X=STR(X,30,13)

A=SUBSTR(X,19,1)

B=SUBSTR(X,20,1)

C=SUBSTR(X,23,1)

D=SUBSTR(X,25,1)

E=SUBSTR(X,28,1)

TABCD=ALLTRIM(A)+ALLTRIM(B)+ALLTRIM(C)+ALLTRIM(D)+ALLTRIM(E)

XABCD=VAL(TABCD)

IF (XABCD >= 1 AND XABCD <= NN)

ACT=.F.

ELSE

ACT=.T.

ENDIF

ENDDO

J=0

DO WHILE J<N-1

NUMX=XABCD+(J*K)

NUMXX=STR(NUMX)

NUMX=VAL(NUMXX)

IF NUMX > NN

NUMXX=NUMX-NN

NUMX=NUMXX

ENDIF

SELECT 6

**FILE DATAE

SEEK NUMX

```

        IF FOUND()
            ELSE
                APPEND BLANK
                J=J+1
                REPLACE IDE WITH J
                REPLACE NIDE WITH NUMX
            ENDIF
        ENDDO

SELECT 5
USE DTMI
INDEX ON IDEE TO IDEE
USE DTMI INDEX IDEE
SELECT 6
USE DATAE
INDEX ON NIDE TO NIDE
USE DATAE INDEX NIDE
GO TOP
RC=NIDE
@ 12,18 SAY '
@ 12,18 SAY 'WORKING ABOUT EPSSE TREATMENT MISSING DATA METHOD'

DO WHILE NOT EOF()

    SELECT 5
    SEEK RC
        IF FOUND()
            XE= XX
            YE= YYX1
            IDPPE=IDP

        SELECT 6
        REPLACE XXE WITH XE
        REPLACE YYX1E WITH YE

```

```

REPLACE IDPE WITH IDPPE
ELSE
    ? 'NO CODE FOUND'
    ? RC
    SELECT 6
    WAIT ""

ENDIF
SELECT 6
SKIP
RC=NIDE
ENDDO

USE DATAE
CALCULATE AVG(YX1E) TO MY
CALCULATE AVG(XXE) TO MX
CALCULATE STD(YX1E) TO SDY
CALCULATE STD(XXE) TO SDX
COUNT TO NUM
GO TOP
SUMDXY=0
SUMEX =0
SUMEY =0
SUMEXX =0
SUMEYY =0

DO WHILE NOT EOF()
    SUMEX =SUMEX +(XXE)
    SUMEY =SUMEY +(YX1E)
    SUMEXX=SUMEXX+(XXE*XXE)
    SUMEYY=SUMEYY+(YX1E*YX1E)
    SUMDXY=SUMDXY+((XXE-MX)*(YX1E-MY))
    SKIP
ENDDO

```

$$VAREX = ((NUM * SUMEXX) - (SUMEX * SUMEX)) / (NUM * (NUM - 1))$$

$$VAREY = ((NUM * SUMEYY) - (SUMEY * SUMEY)) / (NUM * (NUM - 1))$$

$$COVXY = SUMDXY / (NUM - 1)$$

$$SUMMY = SUMMY + MY$$

$$SUMMX = SUMMX + MX$$

$$SUMSDY = SUMSDY + SDY$$

$$SUMSDX = SUMSDX + SDX$$

$$SUMCOVXY = SUMCOVXY + COVXY$$

$$SUMVX = SUMVX + VAREX$$

$$SUMVY = SUMVY + VAREY$$

$$NUMID = NUMID + 1$$

$$VONE = VONE + 1$$

$$ENDDO$$

$$MMEX = SUMMX / (VONE - 1)$$

$$MMEY = SUMMY / (VONE - 1)$$

$$SSDVX = SUMSDX / (VONE - 1)$$

$$SSDVY = SUMSDY / (VONE - 1)$$

$$CCOXY = SUMCOVXY / (VONE - 1)$$

$$VARIX = SUMVX / (VONE - 1)$$

$$VARIY = SUMVY / (VONE - 1)$$

$$SDVX = \sqrt{VARIX}$$

$$SDVY = \sqrt{VARIY}$$

$$RXY = (CCOXY) / (SDVX * SDVY)$$

$$B = RXY * (SSDVY / SSDVX)$$

$$A = MMEY - (B * MMEX)$$

*** CHANGE DATA FOR OTHERS PROGRAM

```

SELECT 8
USE DTCST1
GO TOP
DO WHILE NOT EOF()
    IF MISS=99999
        XX1=XX
        TM =A + (B * XX1)
        REPLACE YXX1 WITH TM
    ENDIF
    SKIP
ENDDO
USE

VON2=1
CREATE TABLE DATASE (ID N(10),SERROR N(10,6),INTERC N(10,6),COEFFI N(10,6))
USE
@ 14,18 SAY 'FINDING SMALLEST STANDARD ERROR EQUATION '+'ROUND'+' '+ALLTRIM(STR
(VON2))

DO WHILE VON2 <= 100
@ 14,18 SAY 'FINDING SMALLEST STANDARD ERROR EQUATION '+'ROUND'+' '+ALLTRIM(STR
(VON2))
*** CHANGE DATA FOR OTHERS PROGRAM
SELECT 8
USE DTCST1
COUNT TO NUM
GO TOP
SUMX=0
SUMY=0
SUMXX=0
SUMYY=0
SUMXY=0
DO WHILE NOT EOF()
    SUMX=SUMX+XX

```

```

SUMY=SUMY+YYX1
SUMXX=SUMXX+(XX*XX)
SUMYY=SUMYY+(YYX1*YYX1)
SUMXY=SUMXY+(XX*YYX1)
SKIP
ENDDO
MY=SUMY/NUM
MX=SUMX/NUM
VARX=((NUM*SUMXX)-(SUMX*SUMX))/(NUM*(NUM-1))
VARY=((NUM*SUMYY)-(SUMY*SUMY))/(NUM*(NUM-1))
SDY=SQRT(VARY)
SDX=SQRT(VARX)
RA=(NUM*(SUMXY)) - ((SUMX)*(SUMY))
RB=(NUM*(SUMXX)) - ((SUMX)*(SUMX))
RC=(NUM*(SUMYY)) - ((SUMY)*(SUMY))
RD=RB*RC
RE=SQRT(RD)
RXY=RA/RE
SE=SDY*(SQRT(1-(RXY*RXY)))
B = RXY * (SDY/SDX)
A = MY - (B * MX)
SELECT 7
USE DATASE
APPEND BLANK
REPLACE ID WITH VONZ
REPLACE ERROR WITH SE
REPLACE INTERC WITH A
REPLACE COEFFI WITH B

SELECT 8
USE DTCST1
GO TOP
DO WHILE NOT EOF()
  IF MISS=99999

```

```

XX1=XX
TM =A+(B*XX1)
REPLACE YYX1 WITH TM
ENDIF
SKIP
ENDDO
VON2=VON2+1
ENDDO

```

```

SELECT 7
USE DATASE
INDEX ON SERROR TO SERROR
USE DATASE INDEX SERROR
GO TOP
A=INTERC
B=COEFFI

```

```

SELECT 8
USE DTCST1
GO TOP
DO WHILE NOT EOF()
  IF MISS=99999
    XX1=XX
    TM =A+(B*XX1)
    REPLACE YYX1 WITH TM
  ENDIF
  SKIP
ENDDO

```

```

SELECT 8
USE DTCST1 INDEX IDMISS
COPY TO DATACS2
USE DATACS2

```

REPLACE ALL GROUP WITH 2

CLOSE ALL

@ 12,18 SAY '

@ 13,18 SAY '

@ 14,18 SAY '

***SAMPLING YX3

@ 12,18 SAY '

@ 10,18 SAY '

@ 10,18 SAY 'WORKING ABOUT SAMPLE NUMBER OF CUSTER AND STRATA'

SELECT 3

USE CUSSTA INDEX CUSSTA

DELETE ALL

PACK

USE CUSSTA INDEX CUSSTA

NC=1

VON=1

DO WHILE VON <= 7

SEED=RAND()

XN=(SEED+3.1415926)**5

XN=STR(XN,30,13)

A=SUBSTR(XN,19,1)

B=SUBSTR(XN,20,1)

C=SUBSTR(XN,24,1)

D=SUBSTR(XN,28,1)

TABCD=ALLTRIM(A)+ALLTRIM(C)

XABCD=VAL(TABCD)

IF (XABCD >= 1 AND XABCD <= 14)

USE CUSSTA INDEX CUSSTA

SEEK XABCD

IF FOUND()

```

      VON=VON
    ELSE
      APPEND BLANK
      REPLACE NIDCUS WITH NC
      REPLACE NCUS WITH XABCD
      NC=NC+1
      VON=VON+1
    ENDIF
  ENDIF
ENDDO
SELECT 3
GO TOP
N=NCUS
SUM=0

DO WHILE NOT EOF()
  SELECT 4
  USE DATARAN3
  CALCULATE STD(YX3) TO STDY FOR CUSTER3=N
  COUNT TO NUM FOR CUSTER3=N
  SUM=SUM+(STDY*NUM)

  SELECT 3
  SKIP
  N=NCUS
ENDDO
GO TOP
N=NCUS
DO WHILE NOT EOF()
  SELECT 4
  CALCULATE STD(YX3) TO STDY FOR CUSTER3=N
  COUNT TO NUM FOR CUSTER3=N
  NNS=SAMPLE*NUM*STDY

```

```

NH=NNS/SUM
NNH=ALLTRIM(STR(NH))
NH=VAL(NNH)
SELECT 3
REPLACE NUMCUS WITH NH
SKIP
N=NCUS
ENDDO
GO TOP
N=NCUS
NSAM=NUMCUS
SUM=0
DO WHILE NOT EOF()

SSTA=1

DO WHILE SSTA <= 3
    SELECT 4
    USE DATARAN3
    CALCULATE STD(YX3) TO STDY FOR CUSTER3=N AND STA3=SSTA
    COUNT TO NUM FOR CUSTER3=N AND STA3=SSTA
    SUM=SUM+(STDY*NUM)
    SSTA=SSTA+1
ENDDO
SSTA=1
DO WHILE SSTA<=3
    SELECT 4
    USE DATARAN3
    CALCULATE STD(YX3) TO STDY FOR CUSTER3=N AND STA3=SSTA
    COUNT TO NUM FOR CUSTER3=N AND STA3=SSTA
    NNS=NSAM*NUM*STDY
    NH =NNS/SUM
    NNH=ALLTRIM(STR(NH))
    NH =VAL(NNH)

```

```

SELECT 3
IF SSTA=1
    REPLACE CSTA1 WITH NH
ENDIF
IF SSTA=2
    REPLACE CSTA2 WITH NH
ENDIF
IF SSTA=3
    REPLACE CSTA3 WITH NH
ENDIF
SSTA=SSTA+1
ENDDO
SELECT 3
SKIP
N=NCUS
NSAM=NUMCUS
SUM=0
ENDDO

***SELECT 3
USE CUSSTA
GO TOP
DO WHILE NOT EOF()
@ 10,18 SAY '
@ 10,18 SAY 'WORKING ABOUT DATA33'

XIDC=NIDCUS
XCUS=NCUS
XNUM=NUMCUS
SLOOP=ALLTRIM(STR(XCUS))
DAT='DAT'+SLOOP
SSTA=1
DO WHILE SSTA <= 3
    SELECT 4

```

```

USE DATARAN3
COPY TO &DAT FOR CUSTER3=XCUS AND STA3=SSTA
SELECT 1
USE &DAT
COUNT TO NN
GO TOP

N=0
DO WHILE NOT EOF()
    N=N+1
    REPLACE NID WITH N
SKIP
ENDDO
GO BOTTOM

*** SAMPLING STATIFIED IN GIVEN CUSTER
NN=NN
SELECT 3
IF SSTA=1
    N =CSTA1
ENDIF
IF SSTA=2
    N =CSTA2
ENDIF
IF SSTA=3
    N =CSTA3
ENDIF
K =NN/N

SELECT 2
USE DCUSSTA
***IDX N:10 IDP N:10 NIDX N:10:6 XX:10:6 YYX1:10:6 STA:10
INDEX ON NIDX TO IDNID
USE DCUSSTA INDEX IDNID

```

DELETE ALL

PACK

SELECT 2

USE DCUSSTA INDEX IDNID

@ 12,18 SAY 'WORKING ABOUT SYSTEMATIC RANDOM NUMBER'

ACT=.T.

DO WHILE ACT

SEED=RAND()

X=(SEED+3.1415926)**5

X=STR(X,30,13)

A=SUBSTR(X,19,1)

B=SUBSTR(X,20,1)

C=SUBSTR(X,23,1)

D=SUBSTR(X,25,1)

E=SUBSTR(X,28,1)

TABCD=ALLTRIM(A)+ALLTRIM(B)+ALLTRIM(C)+ALLTRIM(D)+ALLTRIM(E)

XABCD=VAL(TABCD)

IF (XABCD >= 1 AND XABCD <= NN)

ACT=.F.

ELSE

ACT=.T.

ENDIF

ENDDO

J=0

DO WHILE J<=N-1

NUMX = XABCD+(J*K)

NUMXX=STR(NUMX)

NUMX =VAL(NUMXX)

IF NUMX > NN

NUMXX=NUMX-NN

```

NUMX = NUMXX
ENDIF

```

```

SELECT 2

```

```

**FILE DCUSSTA
SEEK NUMX
IF FOUND()
    ? 'REPEAT1 CODE'
    ? NUMX
    ? XABCD
    ? J
    ? K
    WAIT "
    BROW
    ELSE
    APPEND BLANK
    J=J+1
    REPLACE IDX WITH J
    REPLACE NIDX WITH NUMX
ENDIF
ENDDO
GO TOP
GO BOTTOM

```

```

SELECT 1
USE &DAT
INDEX ON NID TO IDS
USE &DAT INDEX IDS
GO TOP
SELECT 2
USE DCUSSTA
INDEX ON NIDX TO IDNID
USE DCUSSTA INDEX IDNID
GO TOP

```

RC=NIDX

@ 12,18 SAY '

@ 12,18 SAY 'WORKING ABOUT SYSTEMATIC RANDOM SAMPLING IN STRATA'

DO WHILE NOT EOF()

SELECT 1

SEEK RC

IF FOUND()

DX= X

DY= YX3

DID=NID

XID=ID

XCUS=CUSTER3

XSTA=STA3

SELECT 2

REPLACE XX WITH DX

REPLACE YYX1 WITH DY

REPLACE IDN WITH DID

REPLACE IDP WITH XID

REPLACE CUS WITH XCUS

REPLACE STA WITH XSTA

ELSE

? 'NO CODE FOUND'

? RC

SELECT 2

*BROW

WAIT "

ENDIF

SELECT 2

SKIP

RC=NIDX

ENDDO

USE DCUSSTA

```

    SORT ON IDX TO DCSS
    USE DCSS
    DTCS='DTCS'+ALLTRIM(STR(SSTA))
    COPY TO &DTCS
    USE &DTCS
    SSTA=SSTA+1
    SELECT 1
    USE
ENDDO

    USE DTCS1
    APPEND FROM DTCS2
    APPEND FROM DTCS3
    USE DTCS1
    SSLOOP=ALLTRIM(STR(XIDC))
    DTCST='DTCST'+SSLOOP
    COPY TO &DTCST
    USE &DTCST
    SELECT 3
    SKIP
ENDDO

    SELECT 8
    USE DTCST1
    LOOP=2
    DO WHILE LOOP <= VAL(SSLOOP)
        NX =ALLTRIM(STR(LOOP))
        DTCST='DTCST'+NX
        APPEND FROM &DTCST
        LOOP=LOOP+1
    ENDDO

    SELECT 8

```

```

USE DTCST1
INDEX ON IDP TO IDIDP
USE DTCST1 INDEX IDIDP

```

```

GO TOP

```

```

XIDP=0

```

```

@ 12,18 SAY '

```

```

@ 12,18 SAY 'WORKING ABOUT CHECKING REPEAT CODE'

```

```

DO WHILE NOT EOF()

```

```

    IF XIDP=IDP

```

```

        ? 'REPETE CODE'

```

```

        WAIT"

```

```

    ELSE

```

```

    ENDIF

```

```

    XIDP=IDP

```

```

SKIP

```

```

ENDDO

```

```

***BEGIN EPSSE MISSING DATA TREATMENT METHOD

```

```

***NUMBER OF MISSING DATA EQUAL 5%

```

```

@ 12,18 SAY '

```

```

@ 12,18 SAY 'WORKING ABOUT EPSSE MISSING DATA TREATMENT METHOD'

```

```

SELECT 8

```

```

USE DTCST1

```

```

GO TOP

```

```

COUNT TO NUM

```

```

GO TOP

```

```

*** CHANGE NUMBER OF MISSING

```

```

NMISS=NUM*NUMMISS

```

```

N=0

```

```

DO WHILE NOT EOF()

```

```

    N=N+1

```

```

    REPLACE IDMISS WITH N

```

```

SKIP
ENDDO
SELECT 8
USE DTCST1
INDEX ON IDMISS TO IDMISS
USE DTCST1 INDEX IDMISS
GO TOP

NM=1

DO WHILE NM <= NMISS

    SEED=RAND()
    X=(SEED+3,1415926)**5
    X=STR(X,30,13)
    A=SUBSTR(X,19,1)
    B=SUBSTR(X,20,1)
    C=SUBSTR(X,23,1)
    D=SUBSTR(X,25,1)
    E=SUBSTR(X,28,1)

    TABCD=ALLTRIM(A)+ALLTRIM(B)+ALLTRIM(C)+ALLTRIM(D)+ALLTRIM(E)
    XABCD=VAL(TABCD)

    SEEK XABCD
    IF FOUND()
        IF MISS = 99999
            ELSE
                REPLACE MISS WITH 99999
                NM=NM+1
            ENDIF
        ENDIF
    ENDDO

***ADD PROGRAM

```

***PROGRAM EPSSE TREATMENT MISSING DATA METHOD

*** CHANGE DATA FOR OTHERS PROGRAM

SELECT 8

USE DTCST1

COPY TO DTMI FOR MISS # 99999

USE

SELECT 5

USE DTMI

COUNT TO NN

GO TOP

N=0

DO WHILE NOT EOF()

N=N+1

REPLACE IDEE WITH N

SKIP

ENDDO

USE DTMI

INDEX ON IDEE TO IDEE

USE DTMI INDEX IDEE

GO TOP

NN=NN

N=NN/10

NN3=9*N

N=NN3

K=NN/N

NUMID=1

VONE=1

CREATE TABLE DATAE (IDE N(10),IDPE N(10),NIDE N(10),XXE N(10,6),;

YYX1E N(10,6))

USE

@ 12,18 SAY '

@ 12,18 SAY 'WORKING ABOUT EPSSE TREATMENT MISSING DATA METHOD'

@ 13,18 SAY 'FINDING STATISTIC VALUE '+'ROUND'+' '+ALLTRIM(STR(VONE))

SUMMY = 0

SUMMX = 0

SUMSDY = 0

SUMSDX = 0

SUMCOVXY = 0

SUMVX = 0

SUMVY = 0

DO WHILE VONE <= 1000

SELECT 6

USE DATAE

INDEX ON NIDE TO NIDE

USE DATAE INDEX NIDE

DELETE ALL

PACK

USE DATAE INDEX NIDE

@ 12,18 SAY '

@ 12,18 SAY 'WORKING ABOUT EPSSE TREATMENT MISSING DATA METHOD'

@ 13,18 SAY 'FINDING STATISTIC VALUE '+'ROUND'+' '+ALLTRIM(STR(VONE))

ACT=.T.

DO WHILE ACT

SEED=Rand()

X=(SEED+3.1415926)**5

X=STR(X,30,13)

A=SUBSTR(X,19,1)

B=SUBSTR(X,20,1)

C=SUBSTR(X,23,1)

```
D=SUBSTR(X,25,1)
```

```
E=SUBSTR(X,28,1)
```

```
TABCD=ALLTRIM(A)+ALLTRIM(B)+ALLTRIM(C)+ALLTRIM(D)+ALLTRIM(E)
```

```
XABCD=VAL(TABCD)
```

```
IF (XABCD >= 1 AND XABCD <= NN)
```

```
  ACT=.F.
```

```
  ELSE
```

```
    ACT=.T.
```

```
  ENDIF
```

```
ENDDO
```

```
J=0
```

```
DO WHILE J<N-1
```

```
  NUMX=XABCD+(J*K)
```

```
  NUMXX=STR(NUMX)
```

```
  NUMX=VAL(NUMXX)
```

```
  IF NUMX > NN
```

```
    NUMXX=NUMX-NN
```

```
    NUMX=NUMXX
```

```
  ENDIF
```

```
SELECT 6
```

```
**FILE DATAE
```

```
SEEK NUMX
```

```
IF FOUND()
```

```
  ELSE
```

```
    APPEND BLANK
```

```
    J=J+1
```

```
    REPLACE IDE WITH J
```

```
    REPLACE NIDE WITH NUMX
```

```
  ENDIF
```

ENDDO

SELECT 5

USE DTMI

INDEX ON IDEE TO IDEE

USE DTMI INDEX IDEE

SELECT 6

USE DATAE

INDEX ON NIDE TO NIDE

USE DATAE INDEX NIDE

GO TOP

RC=NIDE

@ 12,18 SAY '

@ 12,18 SAY 'WORKING ABOUT EPSSE TREATMENT MISSING DATA METHOD'

DO WHILE NOT EOF()

SELECT 5

SEEK RC

IF FOUND()

XE= XX

YE= YYX1

IDPPE=IDP

SELECT 6

REPLACE XXE WITH XE

REPLACE YYX1E WITH YE

REPLACE IDPE WITH IDPPE

ELSE

? 'NO CODE FOUND'

? RC

SELECT 6

WAIT "

```

ENDIF
SELECT 6
SKIP
RC=NIDE
ENDDO

USE DATAE
CALCULATE AVG(YYX1E) TO MY
CALCULATE AVG(XXE) TO MX
CALCULATE STD(YYX1E) TO SDY
CALCULATE STD(XXE) TO SDX
COUNT TO NUM
GO TOP
SUMDXY=0
SUMEX =0
SUMEY =0
SUMEXX =0
SUMEYY =0

DO WHILE NOT EOF()
    SUMEX =SUMEX +(XXE)
    SUMEY =SUMEY +(YYX1E)
    SUMEXX=SUMEXX+(XXE*XXE)
    SUMEYY=SUMEYY+(YYX1E*YYX1E)
    SUMDXY=SUMDXY+((XXE-MX)*(YYX1E-MY))
SKIP
ENDDO

VAREX = ((NUM*SUMEXX)-(SUMEX*SUMEX))/(NUM*(NUM-1))
VAREY = ((NUM*SUMEYY)-(SUMEY*SUMEY))/(NUM*(NUM-1))
COVXY=SUMDXY/(NUM-1)

SUMMY = SUMMY+MY
SUMMX = SUMMX+MX
SUMSDY = SUMSDY+SDY

```

SUMSDX = SUMSDX+SDX

SUMCOVXY = SUMCOVXY+COVXY

SUMVX = SUMVX +VAREX

SUMVY = SUMVY +VAREY

NUMID=NUMID+1

VONE=VONE+1

ENDDO

MMEX=SUMMX/(VONE-1)

MMEY=SUMMY/(VONE-1)

SSDVX=SUMSDX/(VONE-1)

SSDVY=SUMSDY/(VONE-1)

CCOXY=SUMCOVXY/(VONE-1)

VARIX=SUMVX/(VONE-1)

VARIY=SUMVY/(VONE-1)

SDVX =SQRT(VARIX)

SDVY =SQRT(VARIY)

RXY = (CCOXY)/(SDVX*SDVY)

B = RXY * (SSDVY / SSDVX)

A = MMEY -(B * MMEX)

*** CHANGE DATA FOR OTHERS PROGRAM

SELECT 8

USE DTCST1

GO TOP

DO WHILE NOT EOF()

IF MISS=99999

XX1=XX

TM =A + (B * XX1)

REPLACE YYX1 WITH TM

```

ENDIF
SKIP
ENDDO
USE

VON2=1
CREATE TABLE DATASE (ID N(10),SERROR N(10,6),INTERC N(10,6),COEFFI N(10,6))
USE
@ 14,18 SAY 'FINDING SMALLEST STANDARD ERROR EQUATION '+'ROUND'+ ' '+ALLTRIM(STR
(VON2))

DO WHILE VON2 <= 100
@ 14,18 SAY 'FINDING SMALLEST STANDARD ERROR EQUATION '+'ROUND'+ ' '+ALLTRIM(STR
(VON2))
*** CHANGE DATA FOR OTHERS PROGRAM
SELECT 8
USE DTCST1
COUNT TO NUM
GO TOP
SUMX=0
SUMY=0
SUMXX=0
SUMYY=0
SUMXY=0
DO WHILE NOT EOF()
SUMX=SUMX+XX
SUMY=SUMY+YYX1
SUMXX=SUMXX+(XX*XX)
SUMYY=SUMYY+(YYX1*YYX1)
SUMXY=SUMXY+(XX*YYX1)
SKIP
ENDDO
MY=SUMY/NUM
MX=SUMX/NUM

```

$VARX = (((NUM * SUMXX) - (SUMX * SUMX)) / (NUM * (NUM - 1)))$

$VARY = (((NUM * SUMYY) - (SUMY * SUMY)) / (NUM * (NUM - 1)))$

$SDY = \sqrt{VARY}$

$SDX = \sqrt{VARX}$

$RA = (NUM * (SUMXY)) - ((SUMX) * (SUMY))$

$RB = (NUM * (SUMXX)) - ((SUMX) * (SUMX))$

$RC = (NUM * (SUMYY)) - ((SUMY) * (SUMY))$

$RD = RB * RC$

$RE = \sqrt{RD}$

$RXY = RA / RE$

$SE = SDY * (\sqrt{1 - (RXY * RXY)})$

$B = RXY * (SDY / SDX)$

$A = MY - (B * MX)$

SELECT 7

USE DATABASE

APPEND BLANK

REPLACE ID WITH VON2

REPLACE SERROR WITH SE

REPLACE INTERC WITH A

REPLACE COEFFI WITH B

SELECT 8

USE DTCST1

GO TOP

DO WHILE NOT EOF()

IF MISS=99999

XX1=XX

TM=A+(B*XX1)

REPLACE YYX1 WITH TM

ENDIF

SKIP

ENDDO

VON2=VON2+1

ENDDO

```

SELECT 7
USE DATASE
INDEX ON SERROR TO SERROR
USE DATASE INDEX SERROR
GO TOP
A=INTERC
B=COEFFI

```

```

SELECT 8
USE DTCST1
GO TOP
DO WHILE NOT EOF()
  IF MISS=99999
    XX1=XX
    TM =A+(B*XX1)
    REPLACE YYX1 WITH TM
  ENDIF
  SKIP
ENDDO

```

```

***
SELECT 8
USE DTCST1 INDEX IDMISS
COPY TO DATA3
USE DATA3
REPLACE ALL GROUP WITH 3

```

```

@ 12,18 SAY '
@ 13,18 SAY '
@ 14,18 SAY '
@ 12,18 SAY '
@ 10,18 SAY '
@ 10,18 SAY 'WORKING ABOUT ACCURACY AND POWER'

```

```
USE DATACS1
```

```
GO TOP
```

```
COUNT TO NUM
```

```
GO TOP
```

```
SUMX=0
```

```
SUMY=0
```

```
SUMXX=0
```

```
SUMYY=0
```

```
SUMY2=0
```

```
SUMYY2=0
```

```
SUMY3=0
```

```
SUMYY3=0
```

```
SUMXY=0
```

```
DO WHILE NOT EOF()
```

```
  **YX1
```

```
    SUMX=SUMX+XX
```

```
    SUMY=SUMY+Yyx1
```

```
    SUMXX=SUMXX+(XX*XX)
```

```
    SUMYY=SUMYY+(YYX1*YYX1)
```

```
    SUMXY = SUMXY + (XX*YYX1)
```

```
  SKIP
```

```
ENDDO
```

```
MEANX=SUMX/NUM
```

```
MEANY=SUMY/NUM
```

```
VARX=((((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1))
```

```
VARY=((((NUM*SUMYY)-(SUMY*SUMY)))/(NUM*(NUM-1))
```

MEANX=SUMX/NUM

VARX=((((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1))

MEANX=SUMX/NUM

VARX=((((NUM*SUMXX)-(SUMX*SUMX)))/(NUM*(NUM-1))

RA=(NUM*(SUMXY)) - ((SUMX)*(SUMY))

RB=(NUM*(SUMXX)) - ((SUMX)*(SUMX))

RC=(NUM*(SUMYY)) - ((SUMY)*(SUMY))

RD=RB*RC

RE=SQRT(RD)

RXY=RA/RE

***COMPUTE POWER OF TEST OF CORRELATION

VTR=(RXY*SQRT(NUM-2))/SQRT(1-(RXY*RXY))

IF VTR >= DFVT

NNRHY =1

ELSE

NNRHY =0

ENDIF

***END OF COMPUTE POWER OF TEST OF CORRELATION

***COMPUTE POWER OF TEST OF T-TEST

USE DATACS1

COUNT TO NYG1

GO TOP

SUMYG1=0

SUMYYG1=0

DO WHILE NOT EOF()

SUMYG1 =SUMYG1+YYX1

SUMYYG1=SUMYYG1+(YYX1*YYX1)

SKIP

ENDDO

MEANYG1 = SUMYG1/NYG1

VARYG1 = ((NYG1*SUMYYG1)-(SUMYG1*SUMYG1))/((NYG1)*(NYG1-1))

USE DATACS2

COUNT TO NYG2

GO TOP

SUMYG2=0

SUMYYG2=0

DO WHILE NOT EOF()

 SUMYG2 =SUMYG2+YYX1

 SUMYYG2=SUMYYG2+(YYX1*YYX1)

 SKIP

ENDDO

MEANYG2 = SUMYG2/NYG2

VARYG2 = ((NYG2*SUMYYG2)-(SUMYG2*SUMYG2))/((NYG2)*(NYG2-1))

DIFF1 = MEANYG1-MEANYG2

DMEANY1 = ABS(DIFF1)

TD1 = (((NYG1-1)*VARYG1)+((NYG2-1)*VARYG2))* ((1/NYG1)+(1/NYG2))

TN1 = (NYG1+NYG2)-2

EMEANY1 = SQRT(TD1/TN1)

VTT1 = DMEANY1 / EMEANY1

IF VTT1 >= DFVT

 NNT1HY =1

ELSE

 NNT1HY =0

ENDIF

USE DATACS3

COUNT TO NYG3

GO TOP

SUMYG3=0

SUMYYG3=0

```

DO WHILE NOT EOF()
    SUMYG3 =SUMYG3+YYX1
    SUMYYG3=SUMYYG3+(YYX1*YYX1)
    SKIP
ENDDO

MEANYG3 = SUMYG3/NYG3
VARYG3 = ((NYG3*SUMYYG3)-(SUMYG3*SUMYG3))/((NYG3)*(NYG3-1))

DIFF2 = MEANYG2-MEANYG3
DMEANY2 = ABS(DIFF2)
TD2 = (((NYG2-1)*VARYG2)+((NYG3-1)*VARYG3))* ((1/NYG2)+(1/NYG3))
TN2 = (NYG2+NYG3)-2
EMEANY2 = SQRT(TD2/TN2)
VTT2 = DMEANY2 / EMEANY2

IF VTT2 >= DFVT
    NNT2HY =1
ELSE
    NNT2HY =0
ENDIF

*****END OF COMPUTE POWER OF TEST OF T-TEST
***COMPUTE POWER OF TEST OF ONE WAY ANALYSIS OF VARIANCE
NT = NYG1+NYG2+NYG3
SYY= SUMYYG1+SUMYYG2+SUMYYG3
SY = SUMYG1+SUMYG2+SUMYG3
SST = SYY - ((SY*SY)/NT)
SSB = (((SUMYG1*SUMYG1)/NYG1)+((SUMYG2*SUMYG2)/NYG2)+((SUMYG3*SUMYG3)/NYG3))-
(SY*SY/NT)
SSW = SST-SSB
MSB = SSB/2
MSW = SSW/(NT-3)
VTF = MSB/MSW

```

```
IF VTF >= DFVF
```

```
  NNTFHY =1
```

```
ELSE
```

```
  NNTFHY =0
```

```
ENDIF
```

```
*****END OF COMPUTE POWER OF TEST OF ONE WAY ANALYSIS OF VARIANCE
```

```
? 'MEANX'
```

```
?? MEANX
```

```
? 'VARX '
```

```
?? VARX
```

```
? 'MEANY'
```

```
?? MEANY
```

```
? 'VARY '
```

```
?? VARY
```

```
? 'COR XY'
```

```
?? RXY
```

```
USE DCSTOTAL
```

```
APPEND BLANK
```

```
REPLACE NUM WITH XLOOP
```

```
REPLACE MX WITH MEANX
```

```
REPLACE VX WITH VARX
```

```
REPLACE MEANY1 WITH MEANY
```

```
REPLACE VARY1 WITH VARY
```

```
REPLACE CORXY1 WITH RXY
```

```
REPLACE AM WITH MEANY-PMY
```

```
REPLACE AV WITH VARY -PVY
```

```
REPLACE AR WITH RXY -PCORXY
```

```
REPLACE VVTR WITH VTR
```

```
REPLACE NRHY WITH NNRHY
```

```
REPLACE VTT1 WITH VTT1
```

```
REPLACE NT1HY WITH NNT1HY
```

```
REPLACE WTT2 WITH VTT2
REPLACE NT2HY WITH NNT2HY
REPLACE WTF WITH VTF
REPLACE NTFHY WITH NNTFHY
REPLACE NUMG1 WITH NYG1
REPLACE NUMG2 WITH NYG2
REPLACE NUMG3 WITH NYG3
```

```
CLEAR
@ 21,30 SAY 'PROCESS TIME '+ALLTRIM(STR(XLOOP))
XLOOP=XLOOP+1
```

```
ENDDO
GO TOP
BROW
GO BOTTOM
BROW
GO TOP
COUNT TO N
GO TOP
```

```
SAM=0
SAV=0
SAR=0
SNRHY =0
SNT1HY =0
SNT2HY =0
SNTFHY =0
```

```
SAMM=0
SAVV=0
SARR=0
```

```
DO WHILE NOT EOF()
```

```

SAM=SAM+AM
SMM=SMM+(AM*AM)
SAV=SAV+AV
SAVV=SAVV+(AV*AV)
SAR=SAR+AR
SARR=SARR+(AR*AR)
SNRHY=SNRHY+NRHY
SNT1HY=SNT1HY+NT1HY
SNT2HY=SNT2HY+NT2HY
SNTFHY=SNTFHY+NTFHY

SKIP
ENDDO

MAM=SAM/N
MAV=SAV/N
MAR=SAR/N
MNRHY=SNRHY/N
MNT1HY=SNT1HY/N
MNT2HY=SNT2HY/N
MNTFHY=SNTFHY/N

DM=(MAM*100)/PMY
DV=(MAV*100)/PVY
DR=(MAR*100)/PCORXY

MAMM=SMM/N
MAVV=SAVV/N
MARR=SARR/N

CLEAR

? 'THE DIFFERENCE OF AVERAGE OF MEAN WITH MEAN POPULATION IS(%)
?? DM

```

? 'THE DIFFERENCE OF AVERAGE OF VARIANCE WITH VARIANCE POPULATION IS(%)

?? DV

? 'THE DIFFERENCE OF AVERAGE OF CORRELATION WITH CORRELATION POPULATION IS(%)'

?? DR

?

?

? 'THE SQUARE OF ERROR OF MEAN IS

?? MAMM

?

? 'THE SQUARE OF ERROR OF VARIANCE IS

?? MAVV

?

? 'THE SQUARE OF ERROR OF CORRELATION IS

?? MARR

?

? 'THE STATISTICAL POWER OF CORRELATION IS

?? MNRHY

?

? 'THE STATISTICAL POWER OF T-TEST (1*SD) IS

?? MNT1HY

?

? 'THE STATISTICAL POWER OF T-TEST (2*SD) IS

?? MNT2HY

?

? 'THE STATISTICAL POWER OF ANALYSIS OF VARIANCE IS'

?? MNTFHY

?

WAIT "

CLEAR

@ 17,18 SAY 'PROGRAM MULTI STAGE SAMPLING IS WORKING'

@ 18,18 SAY 'WITH NUMBER OF MISSING IS '+ALLTRIM(STR(XMISS))+ '%'+ ' AND CORRELATION IS
'+(STR(PCORXY,3,3))

@ 19,18 SAY 'AND EPSSE MISSING DATA TREATMENT METHOD'

@ 23,30 SAY 'END'PROGRAM'

COPY TO MEP20C30

USE MEP20C30

BROW

CLOSE ALL

WAIT "

CLEAR

RETURN





ภาคผนวก ง

ค่าสถิติพื้นฐานของข้อมูลที่ได้จากการจำลองสถานการณ์

ค่าสถิติพื้นฐานของข้อมูลที่ได้จากการจำลองสถานการณ์

ข้อมูลที่นำมาศึกษาวิจัยครั้งนี้ได้จากการจำลองสถานการณ์โดยใช้เทคนิค มอนติ คาร์โล ซิมูเลชัน (Monte carlo Simulation Technique) เป็นข้อมูลผลสัมฤทธิ์ทางการเรียนของนักเรียนประกอบด้วยเกรดเฉลี่ยเป็นตัวแปรเกณฑ์ (Y) และตัวแปรทำนายคือคะแนนสอบของนักเรียน (X) ผู้วิจัยจะได้นำเสนอค่าสถิติพื้นฐานของข้อมูลที่ได้จากการจำลองสถานการณ์ตามลำดับดังนี้

1. ค่าสถิติการแจกแจงของข้อมูลที่ได้จากการจำลองสถานการณ์
2. การทดสอบความเป็นปกติ
3. ค่าเฉลี่ยและความแปรปรวนของข้อมูลที่ได้จากการจำลองสถานการณ์ จำแนกตามระดับชั้นและกลุ่ม
4. การทดสอบความแตกต่างของค่าเฉลี่ยและการทดสอบความเป็นเอกพันธ์ของความแปรปรวน

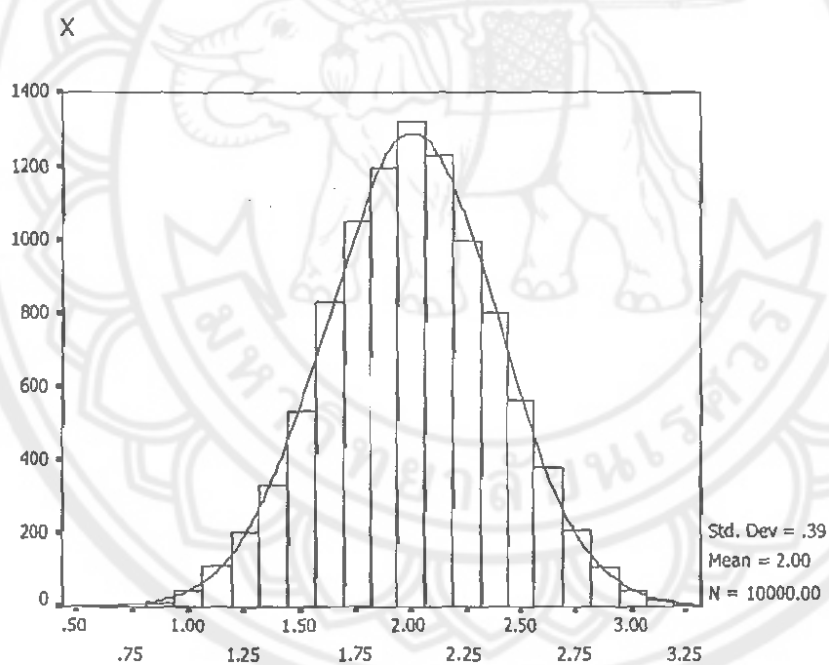
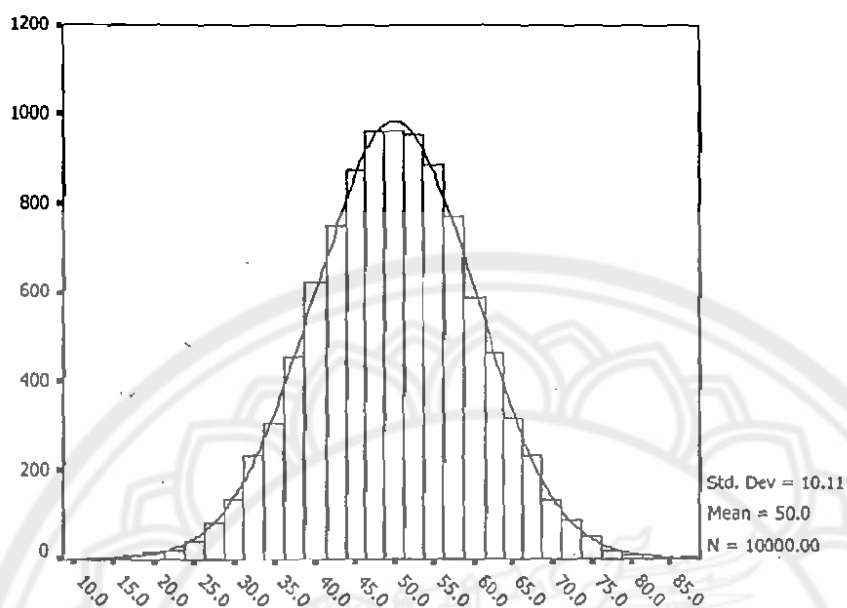
1. ค่าสถิติการแจกแจงของข้อมูลที่ได้จากการจำลองสถานการณ์

ผู้วิจัยจะนำเสนอค่าสถิติการแจกแจงของข้อมูลที่ได้จากการจำลองสถานการณ์ซึ่งประกอบด้วย ความสัมพันธ์ระหว่างตัวแปร ค่าต่ำสุด ค่าสูงสุด ค่าเฉลี่ย ค่าความแปรปรวน ค่าความถี่ และค่าความเบ้ ดังแสดงในตารางที่ 35-37

ตาราง 35 ค่าสถิติพื้นฐานของข้อมูลตามทฤษฎีและที่ได้จากการจำลองสถานการณ์
จำนวนประชากร 10,000 คน ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30

ประชากรกลุ่มที่ 1			ประชากรกลุ่มที่ 2			ประชากรกลุ่มที่ 3		
ค่าสถิติ	ทฤษฎี	จำลอง สถานการณ์	ค่าสถิติ	ทฤษฎี	จำลอง สถานการณ์	ค่าสถิติ	ทฤษฎี	จำลอง สถานการณ์
CORR	.30	.3035	CORR	.30	.3013	CORR	.30	.2963
MINX	-	10.4490	MINX	-	10.3104	MINX	-	5.4747
MAXX	-	87.5863	MAXX	-	86.5897	MAXX	-	85.4271
MEANX	50.00	50.0211	MEANX	50.00	49.9074	MEANX	50.00	49.9375
VARIX	100	102.2212	VARIX	100	98.6291	VARIX	100	99.2323
SKEWX	0	.0110	SKEWX	0	-.0549	SKEWX	0	-.0139
KURTX	3	3.0341	KURTX	3	3.0452	KURTX	3	3.0075
MINY	-	.4577	MINY	-	.2414	MINY	-	.9310
MAXY	-	3.2887	MAXY	-	3.2659	MAXY	-	3.8210
MEANY	2.00	2.0023	MEANY	1.61	1.6109	MEANY	2.38	2.3804
VARIY	.15	.1486	VARIY	.15	.1509	VARIY	.15	.1522
SKEWY	0	-.0183	SKEWY	0	.0322	SKEWY	0	.0272
KURTY	3	2.9486	KURTY	3	2.9659	KURTY	3	3.0070

ตารางที่ 35 แสดงให้เห็นลักษณะการแจกแจงของข้อมูล ซึ่งได้แสดงภาพได้
การแจกแจงตามภาพประกอบที่ 78-80

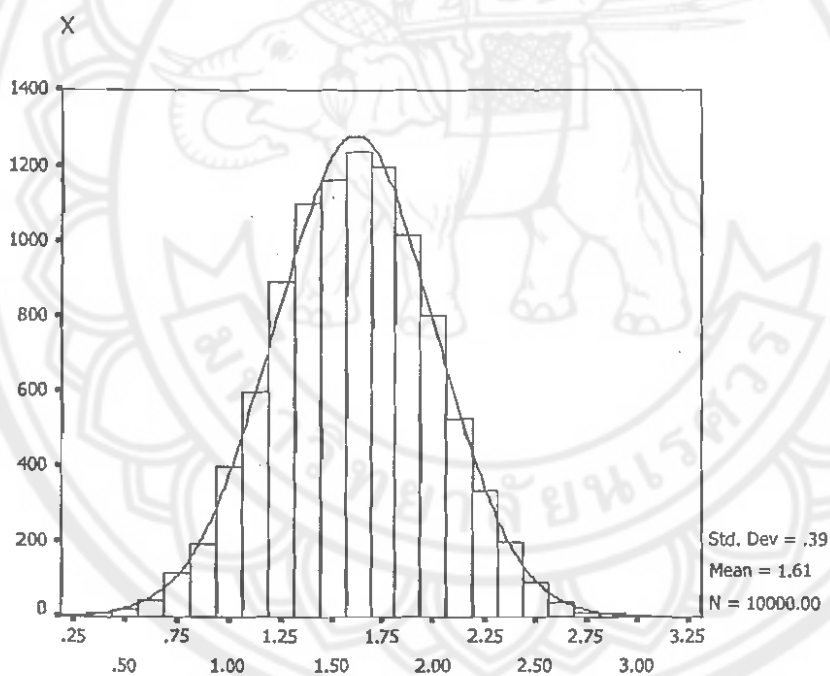
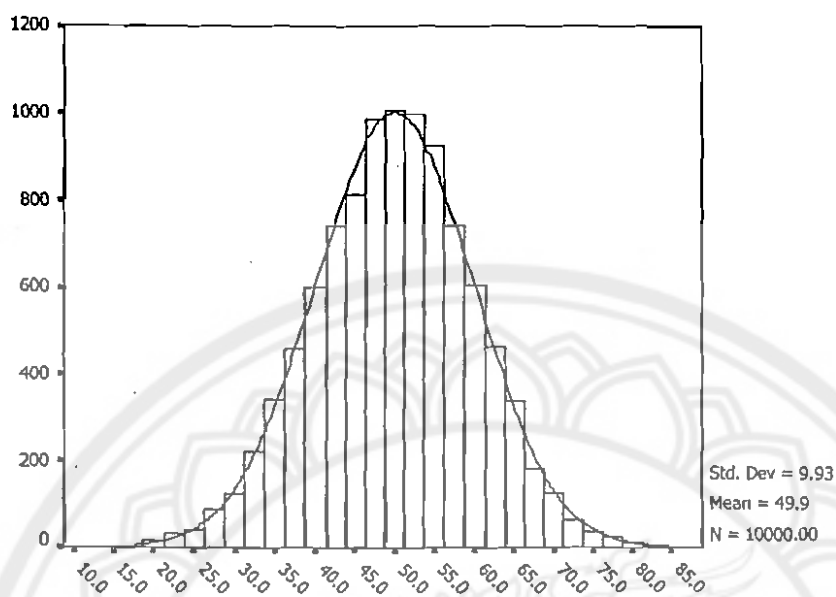


ภาพ 78 กราฟแสดงลักษณะการกระจายของคะแนนสอบ

$(x, \mu = 50, \sigma^2 = 100)$ และเกรดเฉลี่ย

$(y, \mu = 2.00, \sigma^2 = .15)$ ของประชากรกลุ่มที่ 1

ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30



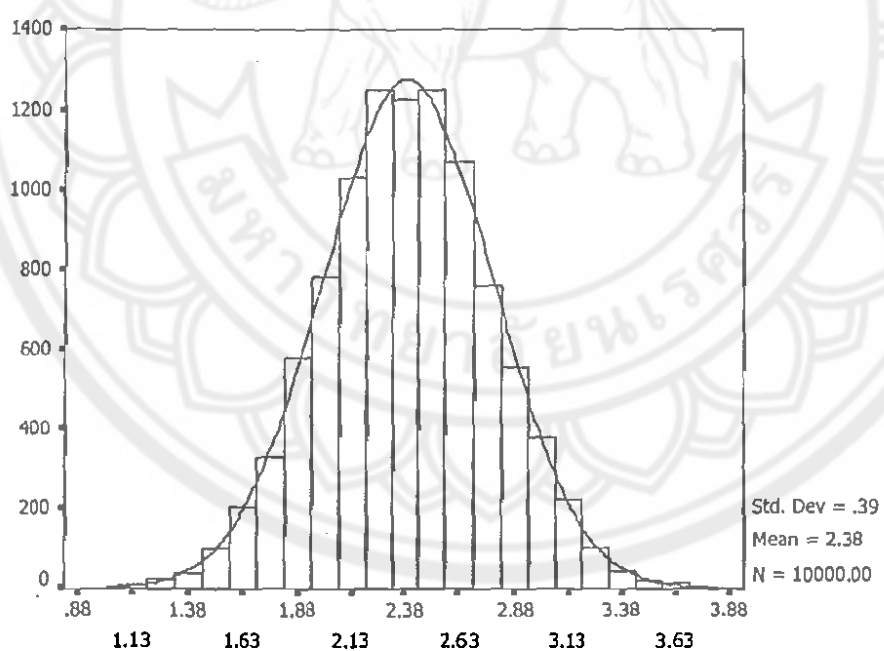
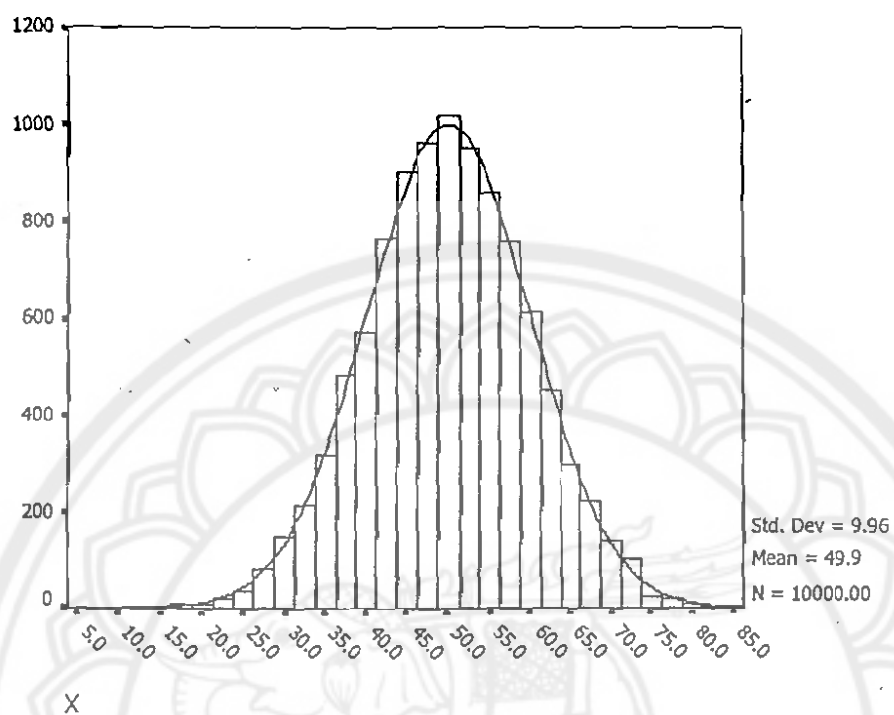
Y

ภาพ 79 กราฟแสดงลักษณะการกระจายของคะแนนสอบ

$(X, \mu = 50, \sigma^2 = 100)$ และเกรดเฉลี่ย

$(Y, \mu = 1.61, \sigma^2 = .15)$ ของประชากรกลุ่มที่ 2

ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30



Y

ภาพ 80 กราฟแสดงลักษณะการกระจายของคะแนนสอบ

$(x, \mu = 50, \sigma^2 = 100)$ และเกรดเฉลี่ย

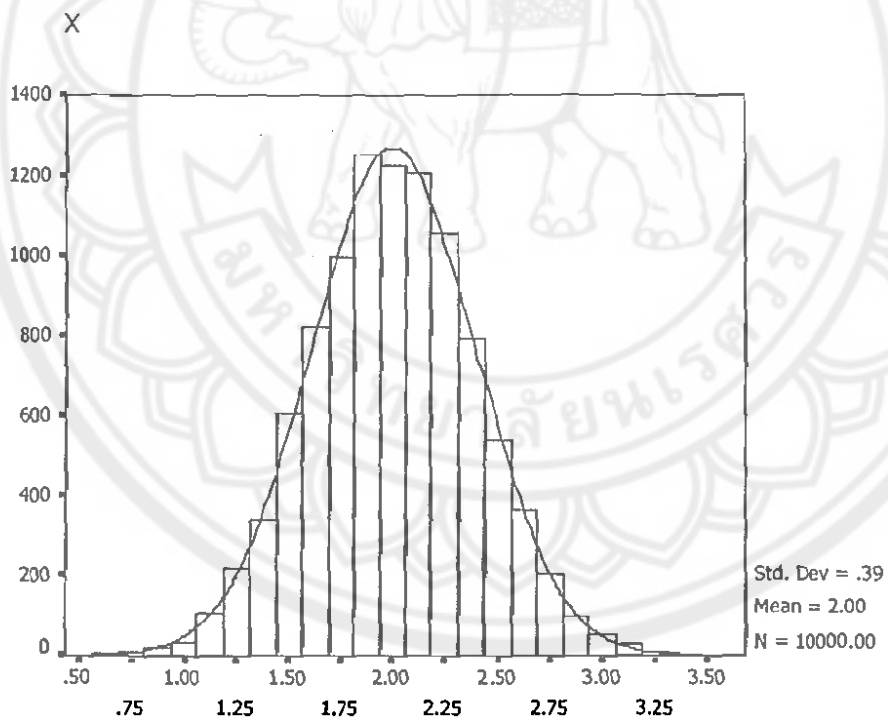
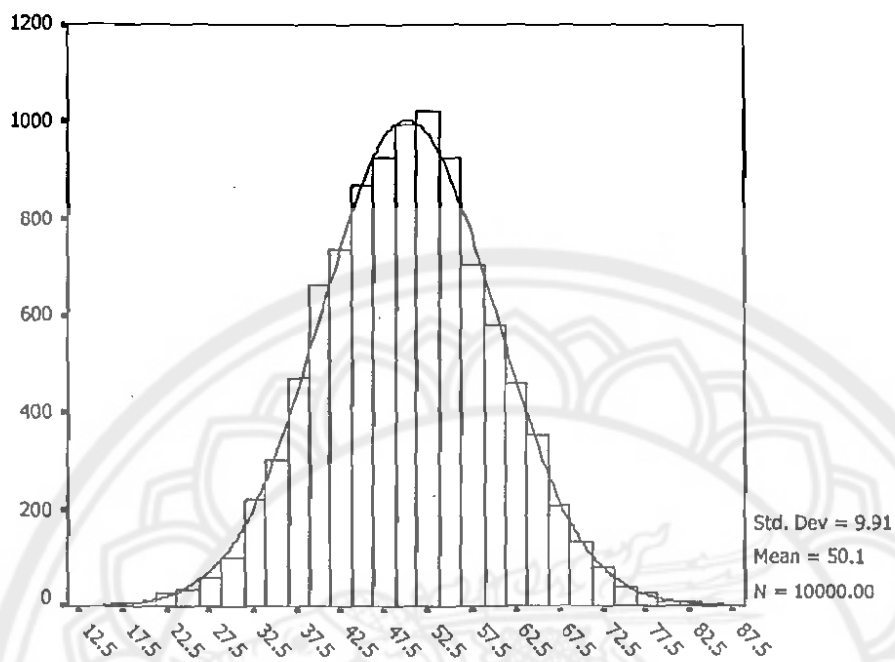
$(y, \mu = 2.38, \sigma^2 = .15)$ ของประชากรกลุ่มที่ 3

ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30

ตาราง 36 ค่าสถิติพื้นฐานของข้อมูลตามทฤษฎีและที่ได้จากการจำลองสถานการณ์
จำนวนประชากร 10,000 คน ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .50

ประชากรกลุ่มที่ 1			ประชากรกลุ่มที่ 2			ประชากรกลุ่มที่ 3		
ค่าสถิติ	ทฤษฎี	จำลอง สถานการณ์	ค่าสถิติ	ทฤษฎี	จำลอง สถานการณ์	ค่าสถิติ	ทฤษฎี	จำลอง สถานการณ์
CORR	.50	.5064	CORR	.50	.5058	CORR	.50	.5049
MINX	-	12.8538	MINX	-	8.9487	MINX	-	11.3829
MAXX	-	87.8513	MAXX	-	87.2465	MAXX	-	82.3781
MEANX	50.00	50.1063	MEANX	50.00	49.9065	MEANX	50.00	50.0297
VARIX	100	98.2574	VARIX	100	101.1922	VARIX	100	101.2029
SKEWX	0	.0280	SKEWX	0	-.0136	SKEWX	0	.0020
KURTX	3	3.0476	KURTX	3	3.0088	KURTX	3	2.9689
MINY	-	.4533	MINY	-	.2482	MINY	-	.7925
MAXY	-	3.5719	MAXY	-	3.1119	MAXY	-	3.7848
MEANY	2.00	1.9987	MEANY	1.61	1.6105	MEANY	2.38	2.3897
VARIY	.15	.1531	VARIY	.15	.1490	VARIY	.15	.1505
SKEWY	0	.0274	SKEWY	0	-.0015	SKEWY	0	.0052
KURTY	3	3.0054	KURTY	3	2.9756	KURTY	3	2.9779

ตารางที่ 36 แสดงให้เห็นลักษณะการแจกแจงของข้อมูล ซึ่งได้แสดงภาพโค้ง
การแจกแจงตามภาพประกอบที่ 81-83



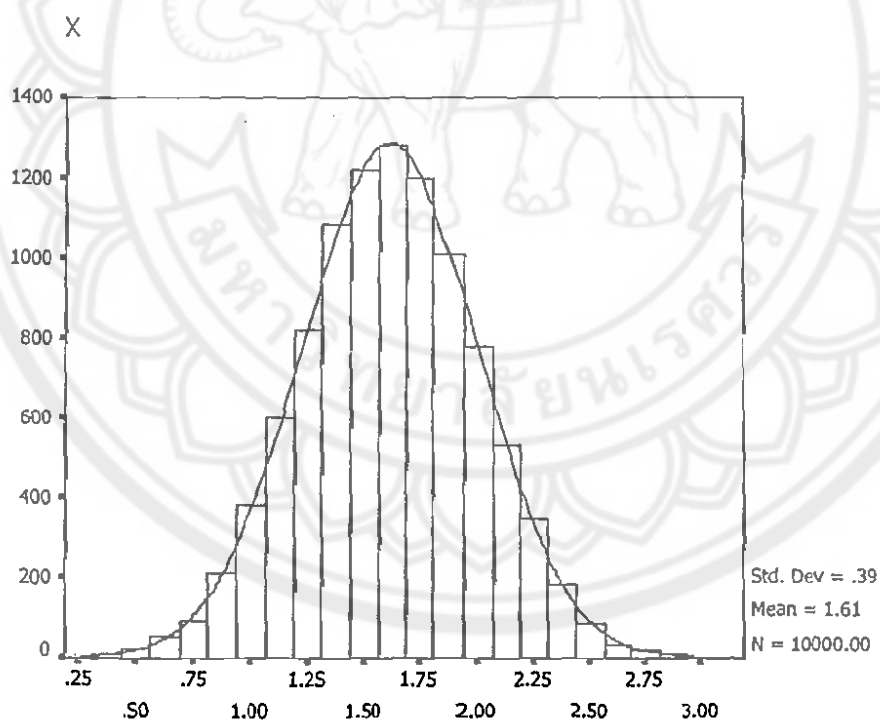
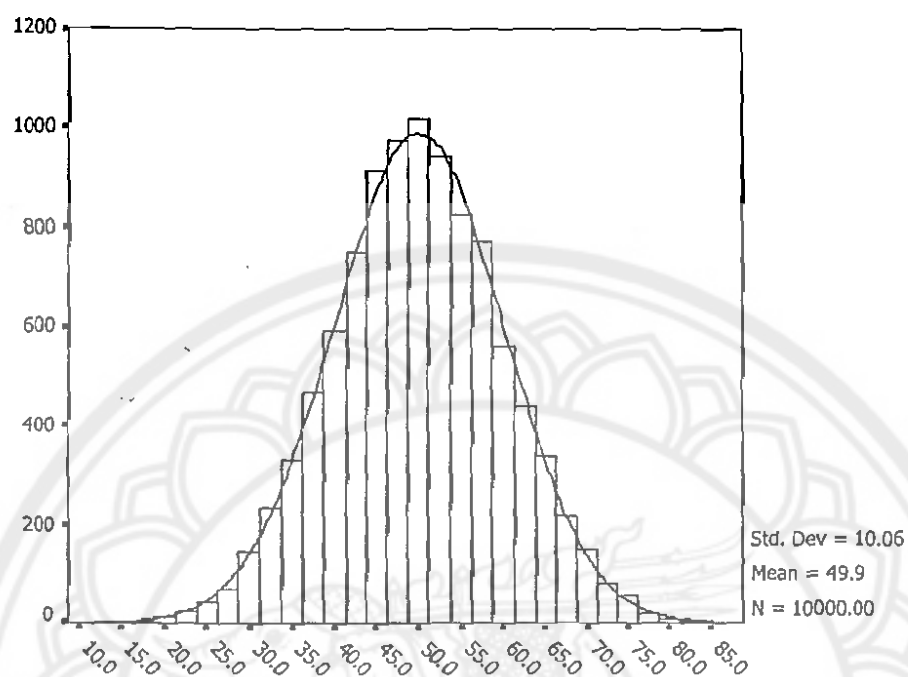
Y

ภาพ 81 กราฟแสดงลักษณะการกระจายของคะแนนสอบ

$(x, \mu = 50, \sigma^2 = 100)$ และเกรดเฉลี่ย

$(y, \mu = 2.00, \sigma^2 = .15)$ ของประชากรกลุ่มที่ 1

ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .50



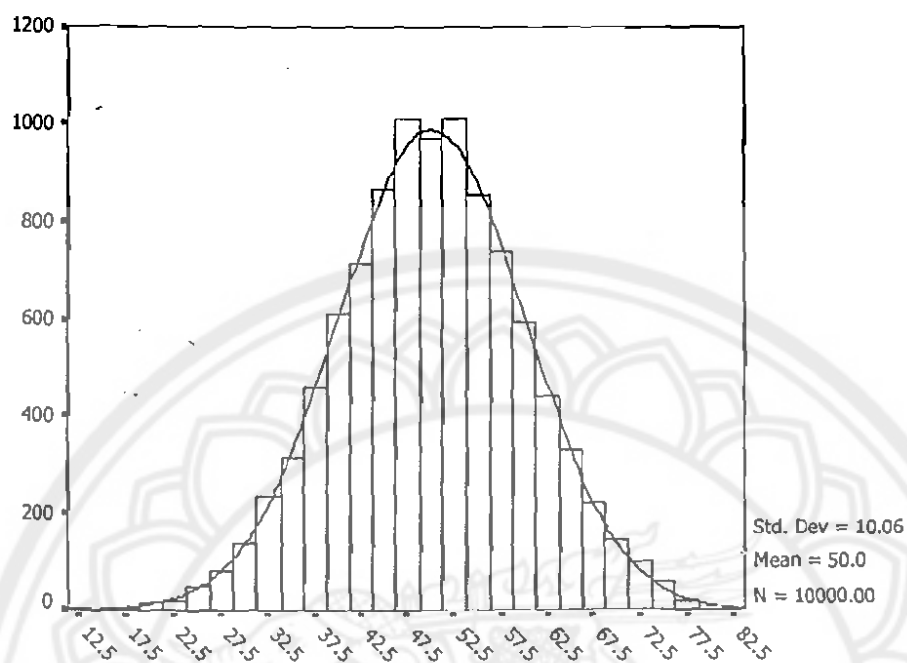
Y

ภาพ 82 กราฟแสดงลักษณะการกระจายของคะแนนสอบ

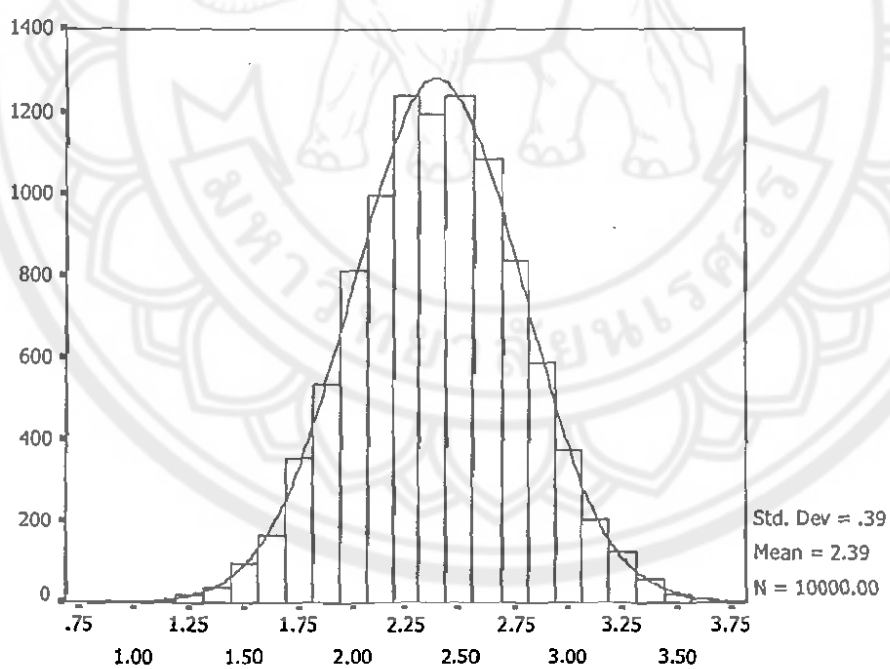
$(x, \mu = 50, \sigma^2 = 100)$ และเกรดเฉลี่ย

$(y, \mu = 1.61, \sigma^2 = .15)$ ของประชากรกลุ่มที่ 2

ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .50



X



Y

ภาพ 83 กราฟแสดงลักษณะการกระจายของคะแนนสอบ

$(x, \mu = 50, \sigma^2 = 100)$ และเกรดเฉลี่ย

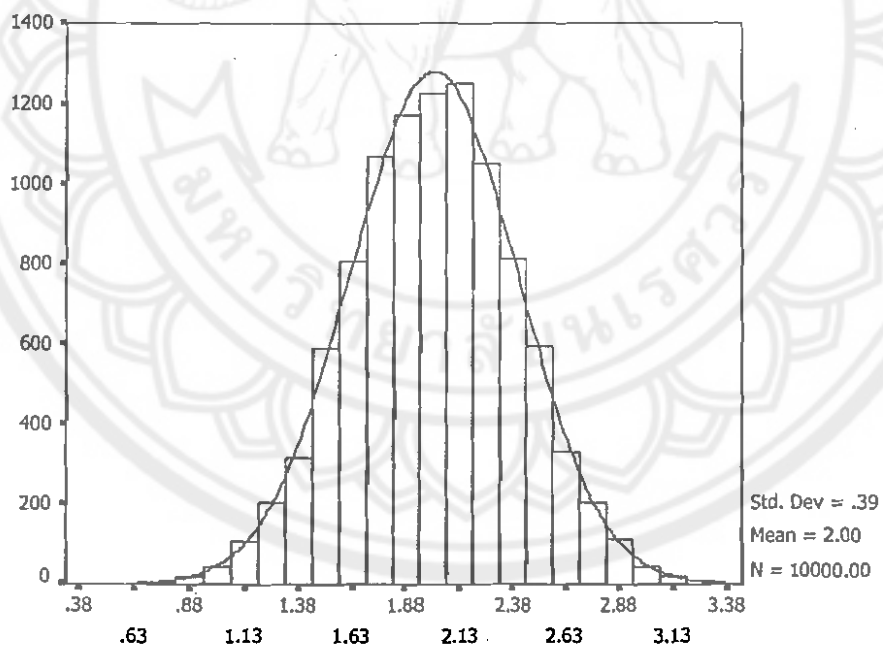
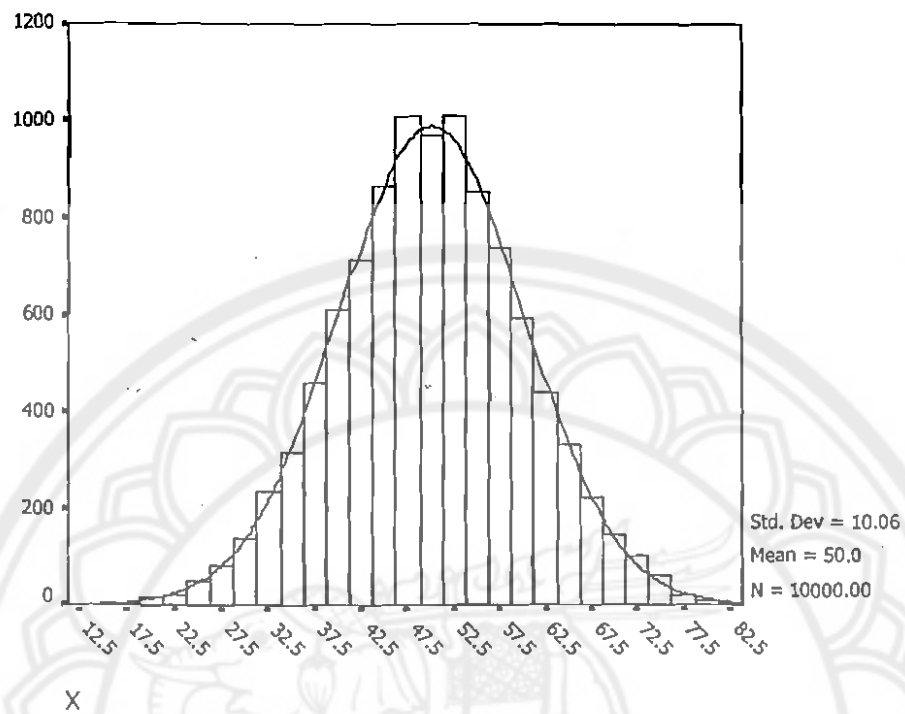
$(y, \mu = 2.38, \sigma^2 = .15)$ ของประชากรกลุ่มที่ 3

ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .50

ตาราง 37 ค่าสถิติพื้นฐานของข้อมูลตามทฤษฎีและที่ได้จากการจำลองสถานการณ์
จำนวนประชากร 10,000 คน ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .70

ประชากรกลุ่มที่ 1			ประชากรกลุ่มที่ 2			ประชากรกลุ่มที่ 3		
ค่าสถิติ	ทฤษฎี	จำลอง สถานการณ์	ค่าสถิติ	ทฤษฎี	จำลอง สถานการณ์	ค่าสถิติ	ทฤษฎี	จำลอง สถานการณ์
CORR	.70	.7038	CORR	.70	.7016	CORR	.70	.6936
MINX	-	11.3829	MINX	-	9.7312	MINX	-	13.1356
MAXX	-	82.3781	MAXX	-	86.2530	MAXX	-	89.1975
MEANX	50.00	50.0297	MEANX	50.00	49.9885	MEANX	50.00	49.9040
VARIX	100	101.2029	VARIX	100	100.0286	VARIX	100	99.0938
SKEWX	0	.0020	SKEWX	0	-.0038	SKEWX	0	-.0074
KURTX	3	2.9689	KURTX	3	2.9870	KURTX	3	2.9692
MINY	-	.4120	MINY	-	.3131	MINY	-	.8545
MAXY	-	3.3929	MAXY	-	2.9397	MAXY	-	3.7536
MEANY	2.00	2.0023	MEANY	1.61	1.6089	MEANY	2.38	2.3871
VARIY	.15	.1511	VARIY	.15	.1498	VARIY	.15	.1508
SKEWY	0	-.0030	SKEWY	0	-.0462	SKEWY	0	.0153
KURTY	3	2.9751	KURTY	3	2.9538	KURTY	3	2.9562

ตารางที่ 37 แสดงให้เห็นลักษณะการแจกแจงของข้อมูล ซึ่งได้แสดงภาพโค้ง
การแจกแจงตามภาพประกอบที่ 84-86

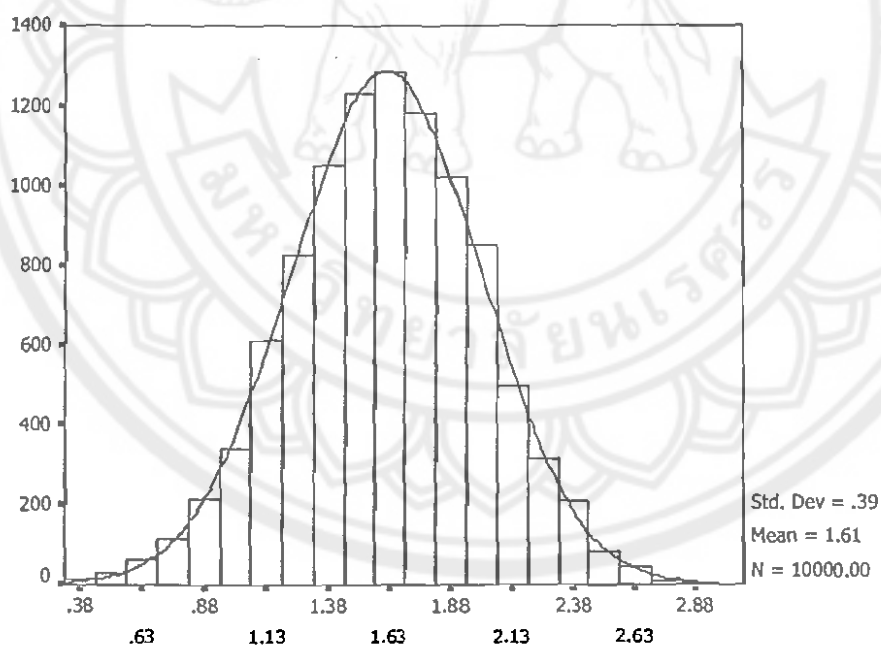
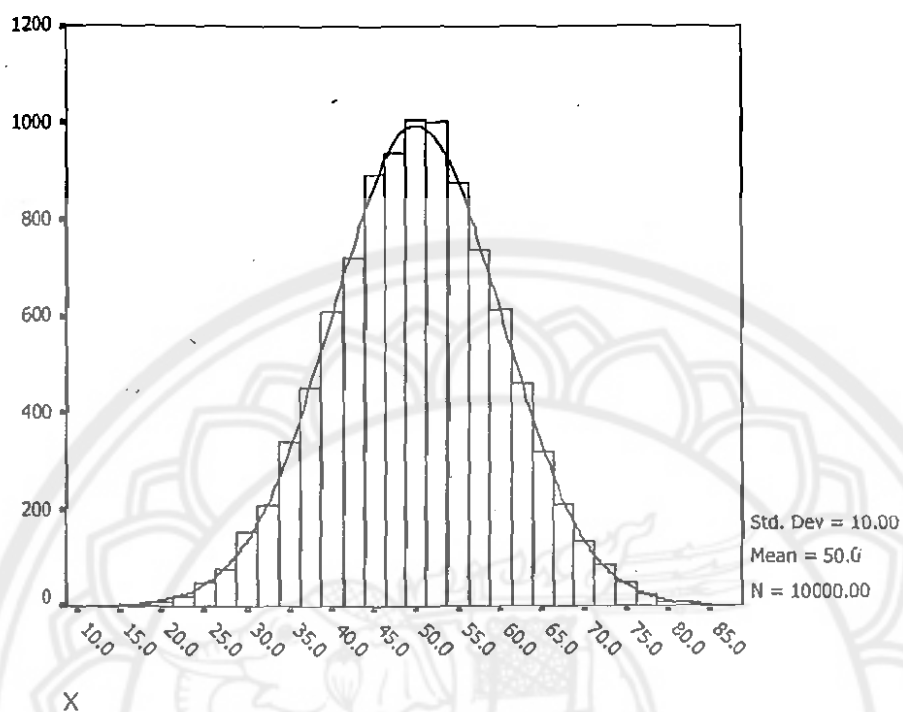


ภาพ 84 กราฟแสดงลักษณะการกระจายของคะแนนสอบ

$(x, \mu = 50, \sigma^2 = 100)$ และเกรดเฉลี่ย

$(y, \mu = 2.00, \sigma^2 = .15)$ ของประชากรกลุ่มที่ 1

ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .70

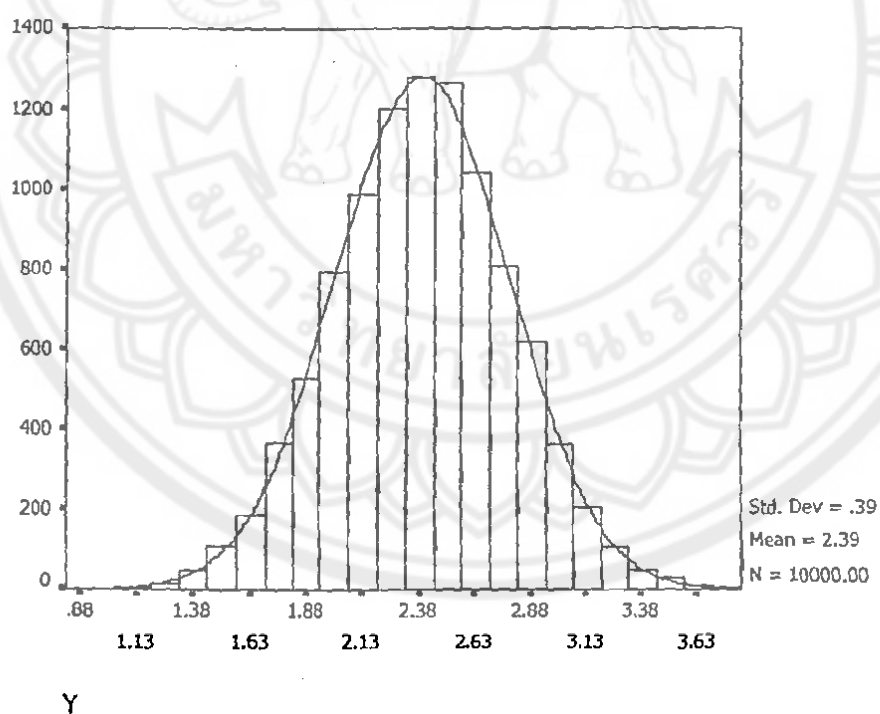
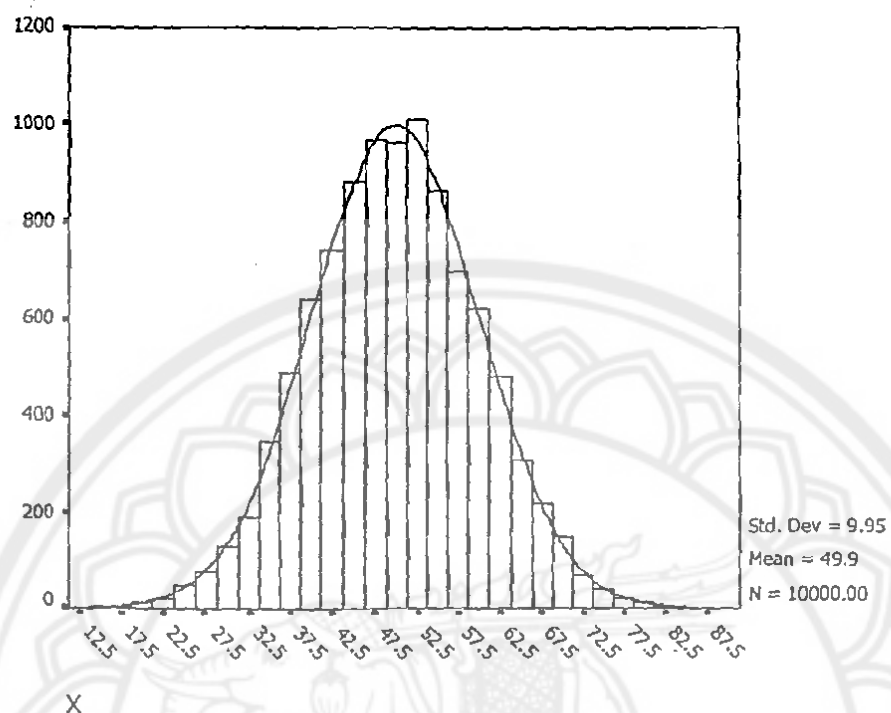


ภาพ 85 กราฟแสดงลักษณะการกระจายของคะแนนสอบ

$(X, \mu = 50, \sigma^2 = 100)$ และเกรดเฉลี่ย

$(Y, \mu = 1.61, \sigma^2 = .15)$ ของประชากรกลุ่มที่ 2

ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .70



ภาพ 86 กราฟแสดงลักษณะการกระจายของคะแนนสอบ

$(X, \mu = 50, \sigma^2 = 100)$ และเกรดเฉลี่ย

$(Y, \mu = 2.38, \sigma^2 = .15)$ ของประชากรกลุ่มที่ 3

ความสัมพันธ์ระหว่างตัวแปรเท่ากับ .70

ตารางที่ 35-37 และแผนภาพที่ 78-86 แสดงค่าสถิติพื้นฐาน ลักษณะการแจกแจงของข้อมูล ข้อมูลที่ผู้วิจัยนำมาศึกษาในครั้งนี้ได้จากการจำลองสถานการณ์โดยกำหนดให้มีลักษณะการแจกแจงแบบปกติสองตัวแปร (Bivariate normal distribution) ค่าสถิติพื้นฐานจากตารางที่ 35 36 และ 37 พบว่า ตัวแปรที่มีความสัมพันธ์ .30 .50 และ .70 มีค่าความเบ้และความโด่งอยู่ในเกณฑ์ของการแจกแจงแบบปกติ คือ มีค่าความเบ้ใกล้เคียง 0 และค่าความโด่งเข้าใกล้ 3 ซึ่งสามารถตรวจสอบลักษณะการแจกแจงด้วยรูปภาพได้ตามแผนภาพที่ 78-86 จะเห็นได้ว่าข้อมูลที่นำมาศึกษาทั้งหมดมีลักษณะการแจกแจงแบบปกติ

2. การทดสอบความเป็นปกติ

จากการนำเสนอค่าสถิติการแจกแจงของข้อมูลที่ได้จากการจำลองสถานการณ์ในตารางที่ 35-37 และแผนภาพที่ 78-86 จะเห็นได้ว่าข้อมูลที่นำมาศึกษาทั้งหมดมีลักษณะการแจกแจงแบบปกติ แต่เพื่อให้เกิดความมั่นใจได้ว่าข้อมูลที่นำมาศึกษามีลักษณะการแจกแจงแบบปกติจริง ผู้วิจัยจึงนำข้อมูลที่ได้จากการจำลองสถานการณ์มาทดสอบการแจกแจงแบบปกติโดยใช้การทดสอบ Kolmogorov-Smirnov Test (K-S Test) เป็นการทดสอบที่เปรียบเทียบค่าฟังก์ชันการแจกแจงสะสมของข้อมูลตัวอย่างกับค่าฟังก์ชันการแจกแจงสะสมของข้อมูลภายใต้สมมติฐานว่า ประชากร/ข้อมูลมีการแจกแจงแบบปกติ (กัลยา วาณิชย์บัญชา, 2540. หน้า 140) ผลการทดสอบดังแสดงในตารางที่ 38

ตาราง 38 ผลการทดสอบการแจกแจงปกติของประชากรกลุ่มที่ 1, 2 และ 3 เมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30 .50 และ .70

ประชากร	ตัวแปร	ความสัมพันธ์ระหว่างตัวแปร					
		$\rho = .30$		$\rho = .50$		$\rho = .70$	
		K-S	p-values	K-S	p-values	K-S	p-values
ประชากร กลุ่มที่ 1	คะแนนสอบ (X)	.455	.986	.742	.641	.755	.619
	เกรดเฉลี่ย (Y)	.485	.973	.410	.996	.660	.776
ประชากร กลุ่มที่ 2	คะแนนสอบ (X)	1.004	.265	.557	.305	.444	.989
	เกรดเฉลี่ย (Y)	.932	.350	.916	1.000	.551	.922
ประชากร กลุ่มที่ 3	คะแนนสอบ (X)	.470	.980	.755	.796	.601	.862
	เกรดเฉลี่ย (Y)	.604	.858	.619	.550	.503	.962

ตารางที่ 38 พบว่า ค่า p-values มากกว่า .05 จึงทำให้ยอมรับสมมติฐานศูนย์ (H_0) ที่ว่าข้อมูลมีการแจกแจงแบบปกติทุกกรณี แสดงว่าข้อมูลทั้งหมดที่ผู้วิจัยนำมาศึกษาในครั้งนี้มีการแจกแจงแบบปกติจริง

3. ค่าเฉลี่ยและค่าความแปรปรวนของข้อมูลที่ได้จากการจำลองสถานการณ์ จัดตามชั้นและกลุ่ม

ผู้วิจัยได้วิเคราะห์ค่าสถิติพื้นฐาน คือ ค่าเฉลี่ยและความแปรปรวนของข้อมูลที่ได้จากการจำลองสถานการณ์เพื่อต้องการให้ข้อมูลภายในชั้นมีความเป็นเอกพันธ์ ระหว่างชั้นมีความเป็นวิวิธพันธ์ และในการจัดข้อมูลแบบกลุ่ม ในแต่ละกลุ่มมีความเป็นวิวิธพันธ์ ระหว่างกลุ่มมีความเป็นเอกพันธ์ ผลการจัดข้อมูลแสดงในตารางที่ 39-41

ตาราง 39 ค่าเฉลี่ยและความแปรปรวนของข้อมูลเกรดเฉลี่ยที่ได้จากการจำลองสถานการณ์
เมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30 จัดตามชั้นและกลุ่ม

ประชากรกลุ่มที่ 1 ($\mu = 2.00, \sigma^2 = .15$)			ประชากรกลุ่มที่ 2 ($\mu = 1.61, \sigma^2 = .15$)			ประชากรกลุ่มที่ 3 ($\mu = 2.38, \sigma^2 = .15$)		
ระดับชั้น	μ	σ^2	ระดับชั้น	μ	σ^2	ระดับชั้น	μ	σ^2
1	1.41	.0301	1	1.03	.0288	1	1.79	.0293
2	2.00	.0429	2	1.61	.0451	2	2.38	.0440
3	2.59	.0269	3	2.20	.0308	3	2.98	.0312
กลุ่ม	μ	σ^2	กลุ่ม	μ	σ^2	กลุ่ม	μ	σ^2
1	2.03	.1468	1	1.61	.1495	1	2.38	.1504
2	2.00	.1344	2	1.60	.1409	2	2.39	.1658
3	2.00	.1576	3	1.60	.1619	3	2.38	.1480
4	2.01	.1450	4	1.62	.1648	4	2.38	.1376
5	2.00	.1520	5	1.61	.1527	5	2.38	.1581
6	1.99	.1375	6	1.61	.1405	6	2.39	.1486
7	2.01	.1351	7	1.62	.1524	7	2.38	.1494
8	2.00	.1639	8	1.63	.1474	8	2.38	.1471
9	2.00	.1506	9	1.61	.1565	9	2.40	.1562
10	2.02	.1466	10	1.61	.1598	10	2.39	.1549
11	1.98	.1521	11	1.60	.1547	11	2.35	.1558
12	2.00	.1595	12	1.61	.1441	12	2.37	.1494
13	2.00	.1497	13	1.61	.1400	13	2.39	.1562
14	1.99	.1475	14	1.61	.1468	14	2.37	.1498

ตาราง 40 ค่าเฉลี่ยและความแปรปรวนของข้อมูลเกรดเฉลี่ยที่ได้จากการจำลองสถานการณ์
เมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .50 จัดตามชั้นและกลุ่ม

ประชากรกลุ่มที่ 1 ($\mu = 2.00, \sigma^2 = .15$)			ประชากรกลุ่มที่ 2 ($\mu = 1.61, \sigma^2 = .15$)			ประชากรกลุ่มที่ 3 ($\mu = 2.38, \sigma^2 = .15$)		
ระดับชั้น	μ	σ^2	ระดับชั้น	μ	σ^2	ระดับชั้น	μ	σ^2
1	1.41	.0290	1	1.02	.0292	1	1.80	.0296
2	2.00	.0448	2	1.61	.0432	2	2.39	.0439
3	2.60	.0315	3	2.20	.0291	3	2.98	.0303
กลุ่ม	μ	σ^2	กลุ่ม	μ	σ^2	กลุ่ม	μ	σ^2
1	1.99	.1564	1	1.62	.1488	1	2.39	.1537
2	2.00	.1491	2	1.61	.1426	2	2.38	.1425
3	2.00	.1536	3	1.63	.1525	3	2.39	.1516
4	1.96	.1420	4	1.62	.1491	4	2.40	.1404
5	2.00	.1466	5	1.61	.1490	5	2.40	.1512
6	2.00	.1444	6	1.60	.1485	6	2.40	.1324
7	1.98	.1388	7	1.61	.1404	7	2.37	.1481
8	2.01	.1659	8	1.61	.1430	8	2.40	.1454
9	2.01	.1548	9	1.60	.1525	9	2.40	.1471
10	1.98	.1518	10	1.62	.1621	10	2.40	.1548
11	2.03	.1639	11	1.61	.1454	11	2.40	.1491
12	1.99	.1592	12	1.61	.1663	12	2.36	.1590
13	2.03	.1580	13	1.60	.1457	13	2.38	.1710
14	2.01	.1546	14	1.61	.1400	14	2.41	.1587

ตาราง 41 ค่าเฉลี่ยและความแปรปรวนของข้อมูลเกรดเฉลี่ยที่ได้จากการจำลองสถานการณ์
เมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .70 จัดตามชั้นและกลุ่ม

ประชากรกลุ่มที่ 1 ($\mu = 2.00, \sigma^2 = .15$)			ประชากรกลุ่มที่ 2 ($\mu = 1.61, \sigma^2 = .15$)			ประชากรกลุ่มที่ 3 ($\mu = 2.38, \sigma^2 = .15$)		
ระดับชั้น	μ	σ^2	ระดับชั้น	μ	σ^2	ระดับชั้น	μ	σ^2
1	1.41	.0302	1	1.01	.0316	1	1.80	.0279
2	2.00	.0447	2	1.61	.0443	2	2.39	.0434
3	2.59	.0295	3	2.19	.0279	3	2.98	.0307
กลุ่ม	μ	σ^2	กลุ่ม	μ	σ^2	กลุ่ม	μ	σ^2
1	2.03	.1466	1	1.61	.1586	1	2.40	.1481
2	2.00	.1508	2	1.61	.1490	2	2.41	.1526
3	2.00	.1539	3	1.61	.1706	3	2.38	.1485
4	2.02	.1469	4	1.59	.1526	4	2.39	.1507
5	2.00	.1476	5	1.62	.1479	5	2.39	.1485
6	2.01	.1724	6	1.59	.1400	6	2.38	.1525
7	1.99	.1371	7	1.63	.1416	7	2.39	.1437
8	1.99	.1471	8	1.57	.1517	8	2.40	.1428
9	1.99	.1568	9	1.60	.1438	9	2.37	.1597
10	2.01	.1434	10	1.63	.1496	10	2.42	.1478
11	1.98	.1481	11	1.63	.1494	11	2.38	.1483
12	2.02	.1534	12	1.63	.1421	12	2.38	.1597
13	2.02	.1440	13	1.60	.1522	13	2.37	.1529
14	1.99	.1641	14	1.60	.1439	14	2.38	.1537

ตารางที่ 39-41 แสดงค่าเฉลี่ยและความแปรปรวนของข้อมูลเกรดเฉลี่ยที่ได้จากการ
จำลองสถานการณ์เมื่อความสัมพันธ์ระหว่างตัวแปรมีค่าเท่ากับ .30 .50 และ .70 จัดตามชั้นและ
กลุ่ม เมื่อพิจารณาค่าเฉลี่ยและความแปรปรวนของข้อมูลตามชั้น พบว่า ค่าเฉลี่ยของข้อมูลต่าง
ชั้นกันมีความแตกต่างกันแต่ความแปรปรวนมีค่าน้อย แสดงว่าภายในชั้นเดียวกันข้อมูลจะมี

ลักษณะคล้าย ๆ กัน ซึ่งแสดงถึงข้อมูลภายในชั้นมีความเป็นเอกพันธ์ แต่ต่างชั้นลักษณะข้อมูลจะแตกต่างกันซึ่งแสดงถึงระหว่างชั้นมีความเป็นวิวิธพันธ์

เมื่อพิจารณาค่าเฉลี่ยและความแปรปรวนของข้อมูลตามกลุ่ม พบว่า ค่าเฉลี่ยของข้อมูลในแต่ละกลุ่มมีค่าใกล้เคียงกันมาก แสดงว่าประชากรในแต่ละกลุ่มประกอบด้วยสมาชิกที่มีลักษณะคล้ายกัน ซึ่งแสดงถึงข้อมูลระหว่างกลุ่มมีความเป็นเอกพันธ์ และเมื่อพิจารณาค่าความแปรปรวนในแต่ละกลุ่ม พบว่า มีค่าใกล้เคียงกับค่าความแปรปรวนของประชากร ($\sigma^2 = .15$) แสดงว่าข้อมูลในแต่ละกลุ่มมีความแตกต่างกัน ซึ่งแสดงถึงความเป็นวิวิธพันธ์ภายในกลุ่ม

4. การทดสอบค่าเฉลี่ยและความแปรปรวนจำแนกตามระดับชั้นและกลุ่ม

จากการนำเสนอค่าเฉลี่ยและความแปรปรวนของข้อมูลเกรดเฉลี่ยที่ได้จากการจำลองสถานการณ์ เมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30 .50 และ .70 จัดตามชั้นและกลุ่มในตารางที่ 39-41 จะเห็นว่าข้อมูลมีลักษณะเป็นชั้นและกลุ่ม แต่เพื่อให้เกิดความมั่นใจได้ว่าข้อมูลที่นำมาศึกษาจัดตามชั้นและกลุ่มจริง ผู้วิจัยได้นำค่าเฉลี่ยที่จัดตามชั้นและกลุ่มมาทดสอบความแตกต่างของค่าเฉลี่ยว่าเท่ากันหรือไม่ โดยใช้การวิเคราะห์ความแปรปรวนแบบทางเดียว (One Way ANOVA) และทดสอบความเป็นเอกพันธ์ของความแปรปรวนโดยใช้ Bartlett-Box Test ผลการวิเคราะห์ข้อมูลดังตารางที่ 42-45

ตาราง 42 ผลการวิเคราะห์ความแปรปรวนค่าเฉลี่ยของเกรดเฉลี่ยที่ได้จากการจำลองสถานการณ์
เมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30 .50 และ .70 จัดตามชั้น

ประชากร	ความสัมพันธ์ระหว่างตัวแปร					
	$\rho = .30$		$\rho = .50$		$\rho = .70$	
	F	p-values	F	p-values	F	p-values
ประชากรกลุ่มที่ 1	14388.02	.00	14056.03	.00	13909.79	.00
ประชากรกลุ่มที่ 2	13747.36	.00	14238.26	.00	13858.85	.00
ประชากรกลุ่มที่ 3	14204.10	.00	14082.87	.00	14384.91	.00

ตารางที่ 42 ผลการวิเคราะห์ความแปรปรวนเพื่อเปรียบเทียบค่าเฉลี่ยของเกรดเฉลี่ยที่ได้จากการจำลองสถานการณ์ เมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30 .50 และ .70 ทุกกลุ่มประชากร ที่จัดตามชั้น แตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .01

ตาราง 43 การทดสอบความเป็นเอกพันธ์ของความแปรปรวนของเกรดเฉลี่ยที่ได้จากการจำลอง
สถานการณ์ เมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30 .50 และ .70 จัดตามชั้น

ประชากร	ความสัมพันธ์ระหว่างตัวแปร					
	$\rho = .30$		$\rho = .50$		$\rho = .70$	
	Bartlett-Box F	p-values	Bartlett-Box F	p-values	Bartlett-Box F	p-values
ประชากรกลุ่มที่ 1	88.89	.00	79.40	.00	83.50	.00
ประชากรกลุ่มที่ 2	89.48	.00	79.72	.00	84.56	.00
ประชากรกลุ่มที่ 3	72.69	.00	76.19	.00	80.91	.00

ตารางที่ 43 พบว่า ความแปรปรวนของเกรดเฉลี่ยเมื่อความสัมพันธ์ระหว่างตัวแปร
เท่ากับ .30 .50 และ .70 ทุกกลุ่มประชากร ที่จัดตามชั้น แตกต่างกันอย่างมีนัยสำคัญทางสถิติ
ที่ระดับ .01

ตาราง 44 ผลการวิเคราะห์ความแปรปรวนค่าเฉลี่ยของเกรดเฉลี่ยที่ได้จากการจำลองสถานการณ์
เมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30 .50 และ .70 จัดตามกลุ่ม

ประชากร	ความสัมพันธ์ระหว่างตัวแปร					
	$\rho = .30$		$\rho = .50$		$\rho = .70$	
	F	p-values	F	p-values	F	p-values
ประชากรกลุ่มที่ 1	.767	.70	1.699	.05	1.139	.32
ประชากรกลุ่มที่ 2	.368	.98	.372	.98	1.547	.09
ประชากรกลุ่มที่ 3	.841	.62	.976	.41	.955	.49

ตารางที่ 44 ผลการวิเคราะห์ความแปรปรวนเพื่อเปรียบเทียบค่าเฉลี่ยของเกรดเฉลี่ยที่ได้จากการจำลองสถานการณ์ เมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30 .50 และ .70 ทุกกลุ่มประชากร ที่จัดตามกลุ่ม ไม่แตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .01

ตาราง 45 การทดสอบความเป็นเอกพันธ์ของความแปรปรวนของเกรดเฉลี่ยที่ได้จากการจำลองสถานการณ์ เมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30 .50 และ .70 จัดตามกลุ่ม

ประชากร	ความสัมพันธ์ระหว่างตัวแปร					
	$\rho = .30$		$\rho = .50$		$\rho = .70$	
	Bartlett-Box F	p-values	Bartlett-Box F	p-values	Bartlett-Box F	p-values
ประชากรกลุ่มที่ 1	1.23	.25	.97	.48	1.22	.26
ประชากรกลุ่มที่ 2	1.02	.43	.89	.57	.97	.48
ประชากรกลุ่มที่ 3	.67	.80	1.37	.17	.40	.97

ตารางที่ 45 พบว่า ความแปรปรวนของเกรดเฉลี่ยเมื่อความสัมพันธ์ระหว่างตัวแปรเท่ากับ .30 .50 และ .70 ทุกกลุ่มประชากร ที่จัดตามกลุ่ม ไม่แตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ .01

จากผลการทดสอบค่าเฉลี่ยและความแปรปรวนระหว่างชั้น พบว่า ค่าเฉลี่ยของข้อมูลที่แตกต่างกันจะมีความแตกต่างกันและก็มีค่าความแปรปรวนแตกต่างกัน ความแปรปรวนในแต่ละชั้นมีค่าน้อยแสดงว่าข้อมูลภายในชั้นเดียวกันจะมีลักษณะคล้ายกัน ซึ่งแสดงถึงความเป็นเอกพันธ์ภายในชั้น แต่ค่าเฉลี่ยของข้อมูลที่แตกต่างกันแสดงว่าข้อมูลต่างกันจะมีความแตกต่างกัน ซึ่งแสดงถึงความเป็นวิวิธพันธ์ระหว่างชั้น

และเมื่อพิจารณาผลการทดสอบค่าเฉลี่ยและความแปรปรวนระหว่างกลุ่ม พบว่า ค่าเฉลี่ยของข้อมูลที่ต่างกลุ่มจะไม่แตกต่างกันและค่าความแปรปรวนก็ไม่แตกต่างกัน ความแปรปรวนในแต่ละกลุ่มมีค่าใกล้เคียงกับค่าความแปรปรวนของประชากร ($\sigma^2 = .15$) แสดงว่าข้อมูลภายในกลุ่มมีลักษณะที่แตกต่างกันคล้ายกับลักษณะของประชากร ซึ่งแสดงถึง

ความเป็นวิวิธพันธุ์ภายในกลุ่ม แต่ค่าเฉลี่ยและความแปรปรวนของข้อมูลที่ต่างกลุ่มไม่แตกต่างกัน แสดงว่าข้อมูลต่างกลุ่มกันจะมีลักษณะคล้ายกัน ซึ่งแสดงถึงความเป็นเอกพันธุ์ระหว่างกลุ่ม

จากผลการวิเคราะห์ที่กล่าวมาแสดงให้เห็นว่าลักษณะข้อมูลที่ผู้วิจัยจำลองสถานการณ์ ขึ้นมาแล้วจัดตามชั้นและกลุ่มสอดคล้องกับหลักการของการกำหนดประชากรเป็นชั้นและกลุ่มจริง





ภาคผนวก จ

ผลการวิเคราะห์ความแปรปรวนค่าเฉลี่ยของความแม่นยำของค่าเฉลี่ย
เลขคณิต ความแปรปรวนและสัมประสิทธิ์สหสัมพันธ์ จำแนกตามวิธีการสุ่ม
ตัวอย่าง จำนวนข้อมูลสุญหาย ความสัมพันธ์ระหว่างตัวแปรและวิธีการจัดการ
ข้อมูลสุญหาย

ผลการวิเคราะห์ความแปรปรวนค่าเฉลี่ยของความแม่นยำของค่าเฉลี่ยเลขคณิต ความแปรปรวนและสัมประสิทธิ์สหสัมพันธ์ จำแนกตามวิธีการสุ่มตัวอย่าง จำนวนข้อมูลสูญหาย ความสัมพันธ์ระหว่างตัวแปรและวิธีการจัดการข้อมูลสูญหาย

ตาราง 46 ผลการวิเคราะห์ความแปรปรวนค่าเฉลี่ยของความแม่นยำของค่าเฉลี่ยเลขคณิต

เงื่อนไขการเปรียบเทียบความแม่นยำของค่าเฉลี่ยเลขคณิต	F	p-values
1. SAM (stratified),NUMMISS (5%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	4.03	.00
2. SAM (stratified),NUMMISS (10%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	10.84	.00
3. SAM (stratified),NUMMISS (20%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	4.73	.00
4. SAM (stratified),NUMMISS (30%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	9.24	.00
5. SAM(cluster),NUMMISS (5%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	1.30	.24
6. SAM(cluster),NUMMISS (10%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	1.37	.20
7. SAM(cluster),NUMMISS (20%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	2.75	.18
8. SAM(cluster),NUMMISS (30%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	3.64	.00
9. SAM(multi-stage),NUMMISS (5%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	4.39	.00
10. SAM(multi-stage),NUMMISS (10%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	5.53	.00
11. SAM(multi-stage),NUMMISS (20%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	5.96	.00
12. SAM(multi-stage),NUMMISS (30%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	6.12	.00

ตาราง 47 ผลการวิเคราะห์ความแปรปรวนค่าเฉลี่ยของความแม่นยำของความแปรปรวน

เงื่อนไขการเปรียบเทียบความแม่นยำของความแปรปรวน	F	p-values
1. SAM (stratified),NUMMISS (5%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	133.32	.00
2. SAM (stratified),NUMMISS (10%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	465.63	.00
3. SAM (stratified),NUMMISS (20%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	1507.17	.00
4. SAM (stratified),NUMMISS (30%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	2792.00	.00
6. SAM(cluster),NUMMISS (5%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	3.92	.00
6. SAM(cluster),NUMMISS (10%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	59.92	.00
7. SAM(cluster),NUMMISS (20%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	429.19	.00
8. SAM(cluster),NUMMISS (30%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	1105.88	.00
9. SAM(multi-stage),NUMMISS (5%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	93.47	.00
10. SAM(multi-stage),NUMMISS (10%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	380.67	.00
11. SAM(multi-stage),NUMMISS (20%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	1171.47	.00
12. SAM(multi-stage),NUMMISS (30%),DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	2371.65	.00

ตาราง 48 ผลการวิเคราะห์ความแปรปรวนค่าเฉลี่ยของความแม่นยำของสัมประสิทธิ์สหสัมพันธ์

เงื่อนไขการเปรียบเทียบความแม่นยำของสัมประสิทธิ์สหสัมพันธ์	F	p-values
1. SAM (stratified), NUMMISS (5%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	119.28	.00
2. SAM (stratified), NUMMISS (10%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	135.32	.00
3. SAM (stratified), NUMMISS (20%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	104.17	.00
4. SAM (stratified), NUMMISS (30%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	103.45	.00
7. SAM(cluster), NUMMISS (5%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	87.80	.00
6. SAM(cluster), NUMMISS (10%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	97.73	.00
7. SAM(cluster), NUMMISS (20%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	102.78	.00
8. SAM(cluster), NUMMISS (30%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	157.76	.00
9. SAM(multi-stage), NUMMISS (5%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	77.21	.00
10. SAM(multi-stage), NUMMISS (10%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	101.22	.00
11. SAM(multi-stage), NUMMISS (20%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	89.58	.00
12. SAM(multi-stage), NUMMISS (30%), DATA(.30,.50,.70) and METHOD (LD , EM , EPSSE)	99.01	.00