ภาคผนวก

ผนวก ก

โปรแกรมคอมพิวเตอร์ที่ใช้ในงานวิจัย

การจำลองและการประมวลผลข้อมูลทำได้โดยการใช้โปรแกรมสำเร็จรูป *SAS* เวอร์ชัน 9.0 ในการจำลองตัวแปรอธิบาย ตัวแปรตาม และคำนวณค่าสัมประสิทธิ์การตัดสินใจที่ปรับค่า $\left(R^2_{adj}\right)$ พิจารณาโปรแกรมสำหรับประมวลผลได้ดังนี้

ก.1 โปรแกรมสำหรับจำลองข้อมูล

กรณีที่ตัวแบบตัวแปรแฝงของการถดถอยโลจิสติกทวิภาค ประกอบด้วยตัวแปรอธิบาย 1 ตัว คือ $Y^* = \eta_{adj(l)} + b_1 x_{i1} + \varepsilon$ ภายใต้เงื่อนไขของสัดส่วนผลตอบสนอง ระดับความเชื่อถือได้ในตัวแปรอธิบาย และอัตราการจำแนกผิดในตัวแปรตาม สามารถจำลองตัวแปรอธิบาย และตัวแปรตามโดยการเขียนมาร์กโคร (macro) ได้ดังนี้

```
%macro case_x1(n_sample =,nsim =,k =);

option nodate nonumber;
data summary;
run;

%let loop_num = 1;
%do %while (&loop_num<= &nsim);

PROC IML;
     start D;
     seed = 0 ;
      pi_b0 = 0.1 ;  pi_b1 = 0.1 ;  pi_b2 = 0.3 ;  pi_b3 = 0.5 ;
      rho_0 = 0.1 ;  rho_1 = 0.1 ;  rho_2 = 0.5 ;  rho_3 = 0.9 ;
      per_1 = 2 ;    per_2 = 10 ;
      rel_1 = 0.3 ;  rel_2 = 0.5 ;  rel_3 = 0.7 ;  rel_4 = 0.9 ;
      V_0   = 10 ;
```

กำหนดค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐาน สำหรับตัวแบบที่มีการแจกแจงแบบปรกติ และมีจำนวนตัวแปรอธิบาย 1 ตัว ดังนี้

```
      mu_x1 = 2 ;   sigma_x1 = 2 ;
```

สำหรับกรณีที่ตัวแบบกำหนดมีจำนวนตัวแปรอธิบาย 5 ตัว กำหนดค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐานสำหรับตัวแบบที่มีการแจกแจงแบบปรกติ ดังนี้

```
mu_x1 = 2 ; mu_x2 = 0 ; mu_x3 = 3 ; mu_x4 = 1 ; mu_x5 = -4 ;
sigma_x1 = 2 ;sigma_x2 = 1 ;  sigma_x3 = 1.414213562;
sigma_x4 = 1.732050808; sigma_x5 = 2;
```

สำหรับกรณีที่ตัวแบบกำหนดมีจำนวนตัวแปรอธิบาย 10 ตัว กำหนดค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐานสำหรับตัวแบบที่มีการแจกแจงแบบปรกติ ดังนี้

```
mu_x1 = 2 ; mu_x2 = 0 ; mu_x3 = 3 ; mu_x4 = 1 ; mu_x5 = -4 ;
mu_x6 = -2 ;mu_x7 = 0 ; mu_x8 = -12 ;mu_x9 = -6 ;mu_x10 = 15 ;
sigma_x1 = 2 ;sigma_x2 = 1 ;sigma_x3 = 1.414213562;
sigma_x4 = 1.732050808; sigma_x5 = 2;sigma_x6 = 2.236067978 ;
sigma_x7 = 2.449489743 ;sigma_x8 = 2.645751311;
sigma_x9 = 2.828427125; sigma_x10 = 3;
```

กำหนดอัตราการจำแนกผิดในตัวแปรตาม (2% , 10%)

```
n_per1 = j(1,1,0);
      do i = 1 to 1;do j = 1 to 1;
      n_per1[1,1] = &n_sample*(per_1/100) ;
```
(การจำแนกค่าผิดทั้งหมด 2%)
```
      end; end;   *print n_per1;
n_per11 = j(1,1,0);
      do i = 1 to 1;do j = 1 to 1;
      n_per11[1,1] = n_per1/2 ;
```
(การจำแนกค่าผิดอย่างละ 1%)
```
      end; end;   *print n_per11;
n_per2 = j(1,1,0);
      do i = 1 to 1;do j = 1 to 1;
      n_per2[1,1] = &n_sample*(per_2/100) ;
```
(การจำแนกค่าผิดทั้งหมด 10%)
```
      end; end;    *print n_per2;
n_per22 = j(1,1,0);
      do i = 1 to 1;    do j = 1 to 1;
      n_per22[1,1] = n_per2/2 ;
```
(การจำแนกค่าผิดอย่างละ 5%)
```
      end; end;   *print n_per22;
```

การคำนวณค่าสถิติต่างๆ ที่ใช้ในการประมาณค่าสัมประสิทธิ์การถดถอยของตัวแบบตัวแปรแฝง

```
compute = j(6,1,0);
      do i = 1 to 1;    do j = 1 to 1;
     compute[1,1] = probit(1-pi_b0);
      end; end;
      do i = 2 to 2;    do j = 1 to 1;
     compute[2,1] = probit(1-pi_b1);
      end; end;         do i = 3 to 3;
      do j = 1 to 1;
     compute[3,1] = probit(1-pi_b2);
      end; end;
      do i = 4 to 4;    do j = 1 to 1;
```

```
     compute[4,1] = probit(1-pi_b3);
      end; end;
      do i = 5 to 5;    do j = 1 to 1;
     compute[5,1]  = V_0*(1-rho_0);
      end; end;
      do i = 6 to 6;    do j = 1 to 1;
     compute[6,1]  = sqrt(compute[5,1]);
      end; end;
*print compute;
```

การคำนวณค่าสถิติที่ใช้ในการกำหนดค่าเริ่มต้นของ η หรือ η_0

กรณีที่ตัวแบบมีตัวแปรอธิบาย 1 ตัว

```
Mu_x = j(1,1,0);
      do i = 1 to 1;    do j = 1 to 1;
      Mu_x[1,1] = (mu_x1/sigma_x1) ;
      end;  end;
*print Mu_x;
```

กรณีที่ตัวแบบมีตัวแปรอธิบาย 5 ตัว

```
Mu_x[1,1] = (mu_x1/sigma_x1) + (mu_x2/sigma_x2) + (mu_x3/sigma_x3) +
            (mu_x4/sigma_x4) + (mu_x5/sigma_x5) ;
```

กรณีที่ตัวแบบมีตัวแปรอธิบาย 10 ตัว

```
Mu_x[1,1] = (mu_x1/sigma_x1) + (mu_x2/sigma_x2) + (mu_x3/sigma_x3) +
            (mu_x4/sigma_x4) + (mu_x5/sigma_x5) + (mu_x6/sigma_x6) +
            (mu_x7/sigma_x7) + (mu_x8/sigma_x8) + (mu_x9/sigma_x9) +
            (mu_x10/sigma_x10);
```

การคำนวณค่าเริ่มต้นของ η หรือ η_0

```
eta_0 = j(1,1,0);
      do i = 1 to 1;    do j = 1 to 1;
      eta_0[1,1] = (1/&k)*(compute[6,1])*(sqrt(rho_0/(1-rho_0)))*
      Mu_x[1,1];
      end;  end;
*print eta_0;
```

การคำนวณค่า c_0

```
C_0 = j(1,1,0);
      do i = 1 to 1;    do j = 1 to 1;
      C_0[i,j] = (sqrt(V_0))*(compute[1,1])+ eta_0 ;
      end;  end;
*print C_0;
```

ก.2 โปรแกรมสำหรับประมาณค่าสัมประสิทธิ์การถดถอยของตัวแบบโลจิสติก

กรณีที่ตัวแบบการถดถอยโลจิสติกประกอบด้วยตัวแปรอธิบาย 1 ตัว

กรณีที่กำหนดค่าสัมประสิทธิ์การตัดสินใจที่แท้จริง 0.1 0.5 และ 0.9

```
B_rho1 = j(1,1,0);
      do i = 1 to 1;    do j = 1 to 1;
      B_rho1[1,1] = (compute[6,1]/sigma_x1)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
*print B_rho1;
```

กรณีที่ตัวแบบการถดถอยโลจิสติกประกอบด้วยตัวแปรอธิบาย 5 ตัว

กรณีที่กำหนดค่าสัมประสิทธิ์การตัดสินใจที่แท้จริง 0.1 0.5 และ 0.9

```
B_rho1 = j(5,1,0);
      do i = 1 to 1;    do j = 1 to 1;
       B_rho1[1,1] = (compute[6,1]/sigma_x1)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 2 to 2;    do j = 1 to 1;
       B_rho1[2,1] = (compute[6,1]/sigma_x2)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 3 to 3;    do j = 1 to 1;
       B_rho1[3,1] = (compute[6,1]/sigma_x3)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 4 to 4;    do j = 1 to 1;
       B_rho1[4,1] = (compute[6,1]/sigma_x4)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 5 to 5;    do j = 1 to 1;
       B_rho1[5,1] = (compute[6,1]/sigma_x5)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
*print B_rho1;
```

กรณีที่ตัวแบบการถดถอยโลจิสติกประกอบด้วยตัวแปรอธิบาย 10 ตัว

กรณีที่กำหนดค่าสัมประสิทธิ์การตัดสินใจที่แท้จริง 0.1 0.5 และ 0.9

```
B_rho1 = j(10,1,0);
      do i = 1 to 1;    do j = 1 to 1;
       B_rho1[1,1] = (compute[6,1]/sigma_x1)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 2 to 2;    do j = 1 to 1;
       B_rho1[2,1] = (compute[6,1]/sigma_x2)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 3 to 3;    do j = 1 to 1;
       B_rho1[3,1] = (compute[6,1]/sigma_x3)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 4 to 4;    do j = 1 to 1;
       B_rho1[4,1] = (compute[6,1]/sigma_x4)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 5 to 5;    do j = 1 to 1;
       B_rho1[5,1] = (compute[6,1]/sigma_x5)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 6 to 6;    do j = 1 to 1;
       B_rho1[6,1] = (compute[6,1]/sigma_x6)*(sqrt(rho_1/(1-rho_1)));
```

```
      end;  end;
      do i = 7 to 7;    do j = 1 to 1;
       B_rho1[7,1] = (compute[6,1]/sigma_x7)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 8 to 8;    do j = 1 to 1;
       B_rho1[8,1] = (compute[6,1]/sigma_x8)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 9 to 9;    do j = 1 to 1;
       B_rho1[9,1] = (compute[6,1]/sigma_x9)*(sqrt(rho_1/(1-rho_1)));
      end;  end;
      do i = 10 to 10;  do j = 1 to 1;
       B_rho1[10,1]=(compute[6,1]/sigma_x10)*(sqrt(rho_1/(1-rho_1)));
      end;  end;  *print B_rho1;
```

จำลองชุดข้อมูลของความคลาดเคลื่อนของตัวแบบตัวแปรแฝงที่มีการแจกแจงแบบปรกติ $\left(\varepsilon \sim N(0,\sigma^2_{\varepsilon(l)})\right)$

กรณีที่มีตัวแปรอธิบาย 1 5 และ 10 ตัว

```
error = j(&n_sample,&k,0);
      do i = 1 to &n_sample;  do j = 1 to &k;
       error[i,j] = 0 + compute[6,1] *rannor(seed); end; end;
*print error;
```

คำนวณค่าเฉลี่ยของความคลาดเคลื่อนของตัวแบบตัวแปรแฝง

```
Mean_Error = j(1,1,0);
      do i = 1 to 1;    do j = 1 to 1;
       Mean_Error[i,j] = Sum(error)/&n_sample ; end;  end;
*print Mean_Error;
```

คำนวณค่าความแปรปรวนสำหรับการจำลองชุดข้อมูลตัวแปรอธิบายภายใต้เงื่อนไขของระดับความเชื่อถือได้ในตัวแปรอธิบายที่กำหนด

```
var_rel1 = j(1,1,0) ;
      do i = 1 to 1 ;   do j = 1 to 1 ;
      var_rel1[i,j] = abs((1/&k)*(error[i,j]/(1-rel_1))) ;end ;end ;
      *print var_rel1 ;
var_rel2 = j(1,1,0) ;
      do i = 1 to 1 ;   do j = 1 to 1 ;
      var_rel2[i,j] = abs((1/&k)*(error[i,j]/(1-rel_2))) ;end ;end ;
      *print var_rel2 ;
var_rel3 = j(1,1,0) ;
      do i = 1 to 1 ;   do j = 1 to 1 ;
      var_rel3[i,j] = abs((1/&k)*(error[i,j]/(1-rel_3))) ;end ;end ;
      *print var_rel3 ;
var_rel4 = j(1,1,0) ;
      do i = 1 to 1 ;   do j = 1 to 1 ;
      var_rel4[i,j] = abs((1/&k)*(error[i,j]/(1-rel_4))) ;end ;end ;
      *print var_rel4 ;
```

จำลองชุดข้อมูลตัวแปรอธิบายและหาค่าเฉลี่ยภายใต้เงื่อนไขของสัดส่วนผลตอบสนองและอัตราการจำแนกผิดในตัวแปรตามที่กำหนด

กรณีที่มีตัวแปรอธิบาย 1 5 และ 10 ตัว

```
X1 = j(&n_sample,&k-9,0);
      do i = 1 to &n_sample;  do j = 1 to &k-9;
       X1[i,j] = mu_x1 + sigma_x1 * rannor(seed);end;end;*print X1;
X2 = j(&n_sample,&k-9,0);
      do i = 1 to &n_sample;  do j = 1 to &k-9;
       X2[i,j] = mu_x2 + sigma_x2 * rannor(seed);end;end;*print X2;
X3 = j(&n_sample,&k-9,0);
      do i = 1 to &n_sample;  do j = 1 to &k-9;
       X3[i,j] = mu_x3 + sigma_x3 * rannor(seed);end;end;*print X3;
X4 = j(&n_sample,&k-9,0);
      do i = 1 to &n_sample;  do j = 1 to &k-9;
       X4[i,j] = mu_x4 + sigma_x4 * rannor(seed);end;end;*print X4;
X5 = j(&n_sample,&k-9,0);
      do i = 1 to &n_sample;  do j = 1 to &k-9;
       X5[i,j] = mu_x5 + sigma_x5 * rannor(seed);end;end;*print X5;
X6 = j(&n_sample,&k-9,0);
      do i = 1 to &n_sample;  do j = 1 to &k-9;
       X6[i,j] = mu_x6 + sigma_x6 * rannor(seed);end;end;*print X6;
X7 = j(&n_sample,&k-9,0);
      do i = 1 to &n_sample;  do j = 1 to &k-9;
       X7[i,j] = mu_x7 + sigma_x7 * rannor(seed);end;end;*print X7;
X8 = j(&n_sample,&k-9,0);
      do i = 1 to &n_sample;  do j = 1 to &k-9;
       X8[i,j] = mu_x8 + sigma_x8 * rannor(seed);end;end;*print X8;
X9 = j(&n_sample,&k-9,0);
      do i = 1 to &n_sample;  do j = 1 to &k-9;
       X9[i,j] = mu_x9 + sigma_x9 * rannor(seed);end;end;*print X9;
X10 = j(&n_sample,&k-9,0);
      do i = 1 to &n_sample;  do j = 1 to &k-9;
      X10[i,j]=mu_x10 + sigma_x10 * rannor(seed);end;end;*print X10;
Mean_X = j(1,10,0);
      do i = 1 to 1;    do j = 1 to 1;
       Mean_X[1,1] = Sum(X1)/&n_sample; end; end;
      do i = 1 to 1;    do j = 2 to 2;
       Mean_X[1,2] = Sum(X2)/&n_sample; end; end;
      do i = 1 to 1;    do j = 3 to 3;
       Mean_X[1,3] = Sum(X3)/&n_sample; end; end;
      do i = 1 to 1;    do j = 4 to 4;
       Mean_X[1,4] = Sum(X4)/&n_sample; end; end;
      do i = 1 to 1;    do j = 5 to 5;
       Mean_X[1,5] = Sum(X5)/&n_sample; end; end;
      do i = 1 to 1;    do j = 6 to 6;
       Mean_X[1,6] = Sum(X6)/&n_sample; end; end;
      do i = 1 to 1;    do j = 7 to 7;
       Mean_X[1,7] = Sum(X7)/&n_sample; end; end;
      do i = 1 to 1;    do j = 8 to 8;
       Mean_X[1,8] = Sum(X8)/&n_sample; end; end;
      do i = 1 to 1;    do j = 9 to 9;
       Mean_X[1,9] = Sum(X9)/&n_sample; end; end;
      do i = 1 to 1;     do j = 10 to 10;
       Mean_X[1,10] = Sum(X10)/&n_sample; end; end; *print Mean_X;
```

จำลองชุดข้อมูลตัวแปรอธิบายและหาค่าเฉลี่ยภายใต้เงื่อนไขของระดับความเชื่อถือได้ในตัวแปรอธิบาย

กรณีที่มีตัวแปรอธิบาย 1 5 และ 10 ตัว

```
X1_rel1 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X1_rel1[i,j] = mu_x1 + (sqrt(var_rel1)) * rannor(seed);
      end; end; *print X1_rel1;
X2_rel1 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X2_rel1[i,j] = mu_x2 + (sqrt(var_rel1)) * rannor(seed);
      end; end; *print X2_rel1;
X3_rel1 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X3_rel1[i,j] = mu_x3 + (sqrt(var_rel1)) * rannor(seed);
      end; end; *print X3_rel1;
X4_rel1 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X4_rel1[i,j] = mu_x4 + (sqrt(var_rel1)) * rannor(seed);
      end; end; *print X4_rel1;
X5_rel1 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X5_rel1[i,j] = mu_x5 + (sqrt(var_rel1)) * rannor(seed);
      end; end; *print X5_rel1;
X6_rel1 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X6_rel1[i,j] = mu_x6 + (sqrt(var_rel1)) * rannor(seed);
      end; end; *print X6_rel1;
X7_rel1 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X7_rel1[i,j] = mu_x7 + (sqrt(var_rel1)) * rannor(seed);
      end; end; *print X7_rel1;
X8_rel1 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X8_rel1[i,j] = mu_x8 + (sqrt(var_rel1)) * rannor(seed);
      end; end; *print X8_rel1;
X9_rel1 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X9_rel1[i,j] = mu_x9 + (sqrt(var_rel1)) * rannor(seed);
      end; end; *print X9_rel1;
X10_rel1 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X10_rel1[i,j] = mu_x10 + (sqrt(var_rel1)) * rannor(seed);
      end; end; *print X10_rel1;
X1_rel2 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X1_rel2[i,j] = mu_x1 + (sqrt(var_rel2)) * rannor(seed);
      end; end; *print X1_rel2;
X2_rel2 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X2_rel2[i,j] = mu_x2 + (sqrt(var_rel2)) * rannor(seed);
      end; end; *print X2_rel2;
X3_rel2 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X3_rel2[i,j] = mu_x3 + (sqrt(var_rel2)) * rannor(seed);
      end; end; *print X3_rel2;
```

```
X4_rel2 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X4_rel2[i,j] = mu_x4 + (sqrt(var_rel2)) * rannor(seed);
      end; end; *print X4_rel2;
X5_rel2 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X5_rel2[i,j] = mu_x5 + (sqrt(var_rel2)) * rannor(seed);
      end; end; *print X5_rel2;
X6_rel2 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X6_rel2[i,j] = mu_x6 + (sqrt(var_rel2)) * rannor(seed);
      end; end; *print X6_rel2;
X7_rel2 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X7_rel2[i,j] = mu_x7 + (sqrt(var_rel2)) * rannor(seed);
      end; end; *print X7_rel2;
X8_rel2 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X8_rel2[i,j] = mu_x8 + (sqrt(var_rel2)) * rannor(seed);
      end; end; *print X8_rel2;
X9_rel2 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X9_rel2[i,j] = mu_x9 + (sqrt(var_rel2)) * rannor(seed);
      end; end; *print X9_rel2;
X10_rel2 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X10_rel2[i,j] = mu_x10 + (sqrt(var_rel2)) * rannor(seed);
      end; end; *print X10_rel2;
X1_rel3 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X1_rel3[i,j] = mu_x1 + (sqrt(var_rel3)) * rannor(seed);
      end; end; *print X1_rel3;
X2_rel3 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X2_rel3[i,j] = mu_x2 + (sqrt(var_rel3)) * rannor(seed);
      end; end; *print X2_rel3;
X3_rel3 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X3_rel3[i,j] = mu_x3 + (sqrt(var_rel3)) * rannor(seed);
      end; end; *print X3_rel3;
X4_rel3 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X4_rel3[i,j] = mu_x4 + (sqrt(var_rel3)) * rannor(seed);
      end; end; *print X4_rel3;
X5_rel3 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X5_rel3[i,j] = mu_x5 + (sqrt(var_rel3)) * rannor(seed);
      end; end; *print X5_rel3;
X6_rel3 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X6_rel3[i,j] = mu_x6 + (sqrt(var_rel3)) * rannor(seed);
      end; end; *print X6_rel3;
X7_rel3 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X7_rel3[i,j] = mu_x7 + (sqrt(var_rel3)) * rannor(seed);
      end; end; *print X7_rel3;
```

```
X8_rel3 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X8_rel3[i,j] = mu_x8 + (sqrt(var_rel3)) * rannor(seed);
      end; end; *print X8_rel3;
X9_rel3 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X9_rel3[i,j] = mu_x9 + (sqrt(var_rel3)) * rannor(seed);
      end; end; *print X9_rel3;
X10_rel3 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X10_rel3[i,j] = mu_x10 + (sqrt(var_rel3)) * rannor(seed);
      end; end; *print X10_rel3;
X1_rel4 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X1_rel4[i,j] = mu_x1 + (sqrt(var_rel4)) * rannor(seed);
      end; end; *print X1_rel4;
X2_rel4 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X2_rel4[i,j] = mu_x2 + (sqrt(var_rel4)) * rannor(seed);
      end; end; *print X2_rel4;
X3_rel4 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X3_rel4[i,j] = mu_x3 + (sqrt(var_rel4)) * rannor(seed);
      end; end; *print X3_rel4;
X4_rel4 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X4_rel4[i,j] = mu_x4 + (sqrt(var_rel4)) * rannor(seed);
      end; end; *print X4_rel4;
X5_rel4 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X5_rel4[i,j] = mu_x5 + (sqrt(var_rel4)) * rannor(seed);
      end; end; *print X5_rel4;
X6_rel4 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X6_rel4[i,j] = mu_x6 + (sqrt(var_rel4)) * rannor(seed);
      end; end; *print X6_rel4;
X7_rel4 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X7_rel3[i,j] = mu_x7 + (sqrt(var_rel4)) * rannor(seed);
      end; end; *print X7_rel4;
X8_rel4 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X8_rel4[i,j] = mu_x8 + (sqrt(var_rel4)) * rannor(seed);
      end; end; *print X8_rel4;
X9_rel4 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X9_rel4[i,j] = mu_x9 + (sqrt(var_rel4)) * rannor(seed);
      end; end; *print X9_rel4;
X10_rel4 = j(&n_sample,&k-9,0);
    do i = 1 to &n_sample;     do j = 1 to &k-9;
      X10_rel4[i,j] = mu_x10 + (sqrt(var_rel4)) * rannor(seed);
      end; end; *print X10_rel4;
Mean_X_rel1 = j(1,10,0);
    do i = 1 to 1;     do j = 1 to 1;
      Mean_X_rel1[1,1] = Sum(X1_rel1)/&n_sample; end; end;
    do i = 1 to 1;     do j = 2 to 2;
      Mean_X_rel1[1,2] = Sum(X2_rel1)/&n_sample; end; end;
```

```
    do i = 1 to 1;     do j = 3 to 3;
       Mean_X_rel1[1,3] = Sum(X3_rel1)/&n_sample; end; end;
    do i = 1 to 1;     do j = 4 to 4;
       Mean_X_rel1[1,4] = Sum(X4_rel1)/&n_sample; end; end;
    do i = 1 to 1;     do j = 5 to 5;
       Mean_X_rel1[1,5] = Sum(X5_rel1)/&n_sample; end; end;
    do i = 1 to 1;     do j = 6 to 6;
       Mean_X_rel1[1,6] = Sum(X6_rel1)/&n_sample; end; end;
    do i = 1 to 1;     do j = 7 to 7;
       Mean_X_rel1[1,7] = Sum(X7_rel1)/&n_sample; end; end;
    do i = 1 to 1;     do j = 8 to 8;
       Mean_X_rel1[1,8] = Sum(X8_rel1)/&n_sample; end; end;
    do i = 1 to 1;     do j = 9 to 9;
       Mean_X_rel1[1,9] = Sum(X9_rel1)/&n_sample; end; end;
    do i = 1 to 1;     do j = 10 to 10;
       Mean_X_rel1[1,10]= Sum(X10_rel1)/&n_sample; end; end;
*print Mean_X_rel1;

Mean_X_rel2 = j(1,10,0);
    do i = 1 to 1;     do j = 1 to 1;
       Mean_X_rel2[1,1] = Sum(X1_rel2)/&n_sample; end; end;
    do i = 1 to 1;     do j = 2 to 2;
       Mean_X_rel2[1,2] = Sum(X2_rel2)/&n_sample; end; end;
    do i = 1 to 1;     do j = 3 to 3;
       Mean_X_rel2[1,3] = Sum(X3_rel2)/&n_sample; end; end;
    do i = 1 to 1;     do j = 4 to 4;
       Mean_X_rel2[1,4] = Sum(X4_rel2)/&n_sample; end; end;
    do i = 1 to 1;     do j = 5 to 5;
       Mean_X_rel2[1,5] = Sum(X5_rel2)/&n_sample; end; end;
    do i = 1 to 1;     do j = 6 to 6;
       Mean_X_rel2[1,6] = Sum(X6_rel2)/&n_sample; end; end;
    do i = 1 to 1;     do j = 7 to 7;
       Mean_X_rel2[1,7] = Sum(X7_rel2)/&n_sample; end; end;
    do i = 1 to 1;     do j = 8 to 8;
       Mean_X_rel2[1,8] = Sum(X8_rel2)/&n_sample; end; end;
    do i = 1 to 1;     do j = 9 to 9;
       Mean_X_rel2[1,9] = Sum(X9_rel2)/&n_sample; end; end;
    do i = 1 to 1;     do j = 10 to 10;
       Mean_X_rel2[1,10] = Sum(X10_rel2)/&n_sample; end; end;
*print Mean_X_rel2;

Mean_X_rel3 = j(1,10,0);
    do i = 1 to 1;     do j = 1 to 1;
       Mean_X_rel3[1,1] = Sum(X1_rel3)/&n_sample; end; end;
    do i = 1 to 1;     do j = 2 to 2;
       Mean_X_rel3[1,2] = Sum(X2_rel3)/&n_sample; end; end;
    do i = 1 to 1;     do j = 3 to 3;
       Mean_X_rel3[1,3] = Sum(X3_rel3)/&n_sample; end; end;
    do i = 1 to 1;     do j = 4 to 4;
       Mean_X_rel3[1,4] = Sum(X4_rel3)/&n_sample; end; end;
    do i = 1 to 1;     do j = 5 to 5;
       Mean_X_rel3[1,5] = Sum(X5_rel3)/&n_sample; end; end;
    do i = 1 to 1;     do j = 6 to 6;
       Mean_X_rel3[1,6] = Sum(X6_rel3)/&n_sample; end; end;
    do i = 1 to 1;     do j = 7 to 7;
       Mean_X_rel3[1,7] = Sum(X7_rel3)/&n_sample; end; end;
    do i = 1 to 1;     do j = 8 to 8;
       Mean_X_rel3[1,8] = Sum(X8_rel3)/&n_sample; end; end;
```

```
    do i = 1 to 1;     do j = 9 to 9;
       Mean_X_rel3[1,9] = Sum(X9_rel3)/&n_sample; end; end;
    do i = 1 to 1;     do j = 10 to 10;
       Mean_X_rel3[1,10] = Sum(X10_rel3)/&n_sample; end; end;
*print Mean_X_rel3;

Mean_X_rel4 = j(1,10,0);
    do i = 1 to 1;     do j = 1 to 1;
       Mean_X_rel4[1,1] = Sum(X1_rel4)/&n_sample; end; end;
    do i = 1 to 1;     do j = 2 to 2;
       Mean_X_rel4[1,2] = Sum(X2_rel4)/&n_sample; end; end;
    do i = 1 to 1;     do j = 3 to 3;
       Mean_X_rel4[1,3] = Sum(X3_rel4)/&n_sample; end; end;
    do i = 1 to 1;     do j = 4 to 4;
       Mean_X_rel4[1,4] = Sum(X4_rel4)/&n_sample; end; end;
    do i = 1 to 1;     do j = 5 to 5;
       Mean_X_rel4[1,5] = Sum(X5_rel4)/&n_sample; end; end;
    do i = 1 to 1;     do j = 6 to 6;
       Mean_X_rel4[1,6] = Sum(X6_rel4)/&n_sample; end; end;
    do i = 1 to 1;     do j = 7 to 7;
       Mean_X_rel4[1,7] = Sum(X7_rel4)/&n_sample; end; end;
    do i = 1 to 1;     do j = 8 to 8;
       Mean_X_rel4[1,8] = Sum(X8_rel4)/&n_sample; end; end;
    do i = 1 to 1;     do j = 9 to 9;
       Mean_X_rel4[1,9] = Sum(X9_rel4)/&n_sample; end; end;
    do i = 1 to 1;     do j = 10 to 10;
       Mean_X_rel4[1,10] = Sum(X10_rel4)/&n_sample; end; end;
*print Mean_X_rel4;
```

คำนวณค่าของ η ตามเงื่อนไขต่างๆ

```
eta_m_rho1 = j(1,1,0); do i = 1 to 1;  do j = 1 to 1;
      eta_m_rho1[1,1] = (Mean_X*B_rho1) + Mean_Error; end; end;
      *print eta_m_rho1;
eta_m_rho2 = j(1,1,0); do i = 1 to 1;  do j = 1 to 1;
      eta_m_rho2[1,1] = (Mean_X*B_rho2) + Mean_Error; end; end;
      *print eta_m_rho2;
eta_m_rho3 = j(1,1,0);  do i = 1 to 1; do j = 1 to 1;
      eta_m_rho3[1,1] = (Mean_X*B_rho3) + Mean_Error; end; end;
      *print eta_m_rho3;
eta_m_rel1 = j(1,1,0);  do i = 1 to 1; do j = 1 to 1;
      eta_m_rel1[1,1] = (Mean_X_rel1*B_rho3) + Mean_Error; end; end;
      *print eta_m_rel1;
eta_m_rel2 = j(1,1,0);  do i = 1 to 1; do j = 1 to 1;
      eta_m_rel2[1,1] = (Mean_X_rel2*B_rho3) + Mean_Error; end; end;
      *print eta_m_rel2;
eta_m_rel3 = j(1,1,0);  do i = 1 to 1; do j = 1 to 1;
      eta_m_rel3[1,1] = (Mean_X_rel3*B_rho3) + Mean_Error; end; end;
      *print eta_m_rel3;
eta_m_rel4 = j(1,1,0);  do i = 1 to 1; do j = 1 to 1;
      eta_m_rel4[1,1] = (Mean_X_rel4*B_rho3) + Mean_Error; end; end;
      *print eta_m_rel4;
eta_d_pi_b1 = j(1,1,0); do i = 1 to 1; do j = 1 to 1;
      eta_d_pi_b1[1,1] = C_0 - ((sqrt(V_0))*compute[2,1]); end; end;
      *print eta_d_pi_b1
eta_d_pi_b2 = j(1,1,0); do i = 1 to 1; do j = 1 to 1;
      eta_d_pi_b2[1,1] = C_0 - ((sqrt(V_0))*compute[3,1]); end; end;
      *print eta_d_pi_b2;
eta_d_pi_b3 = j(1,1,0); do i = 1 to 1; do j = 1 to 1;
```

```
      eta_d_pi_b3[1,1] = C_0 - ((sqrt(V_0))*compute[4,1]); end; end;
      *print eta_d_pi_b3;
eta_adj_r1b1 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;   do j = 1 to &k;
      eta_adj_r1b1[i,j] = eta_d_pi_b1 - eta_m_rho1;  end;  end;
      *print eta_adj_r1b1;
eta_adj_r1b2 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;   do j = 1 to &k;
      eta_adj_r1b2[i,j] = eta_d_pi_b2 - eta_m_rho1;  end;  end;
      *print eta_adj_r1b2;
eta_adj_r1b3 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;   do j = 1 to &k;
      eta_adj_r1b3[i,j] = eta_d_pi_b3 - eta_m_rho1; end; end;
      *print eta_adj_r1b3;
eta_adj_r2b1 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;   do j = 1 to &k;
      eta_adj_r2b1[i,j] = eta_d_pi_b1 - eta_m_rho2; end; end;
      *print eta_adj_r2b1;
eta_adj_r2b2 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;    do j = 1 to &k;
      eta_adj_r2b2[i,j] = eta_d_pi_b2 - eta_m_rho2; end; end;
      *print eta_adj_r2b2;
eta_adj_r2b3 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;    do j = 1 to &k;
      eta_adj_r2b3[i,j] = eta_d_pi_b3 - eta_m_rho2; end; end;
      *print eta_adj_r2b3;
eta_adj_r3b1 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;     do j = 1 to &k;
      eta_adj_r3b1[i,j] = eta_d_pi_b1 - eta_m_rho3; end; end;
      *print eta_adj_r3b1;
eta_adj_r3b2 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;     do j = 1 to &k;
      eta_adj_r3b2[i,j] = eta_d_pi_b2 - eta_m_rho3; end; end;
      *print eta_adj_r3b2;
eta_adj_r3b3 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;     do j = 1 to &k;
      eta_adj_r3b3[i,j] = eta_d_pi_b3 - eta_m_rho3; end; end;
      *print eta_adj_r3b3;
eta_adj_rel1 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;     do j = 1 to &k;
      eta_adj_rel1[i,j] = eta_d_pi_b3 - eta_m_rel1; end; end;
      *print eta_adj_rel1;
eta_adj_rel2 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;     do j = 1 to &k;
      eta_adj_rel2[i,j] = eta_d_pi_b3 - eta_m_rel2; end; end;
      *print eta_adj_rel2;
eta_adj_rel3 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;     do j = 1 to &k;
      eta_adj_rel3[i,j] = eta_d_pi_b3 - eta_m_rel3; end; end;
      *print eta_adj_rel3;
eta_adj_rel4 =j(&n_sample,&k,0);
      do i = 1 to &n_sample;     do j = 1 to &k;
      eta_adj_rel4[i,j] = eta_d_pi_b3 - eta_m_rel4; end; end;
      *print eta_adj_rel4;
```

จำลองชุดข้อมูลตัวแปรตามภายใต้เงื่อนไขของสัดส่วนผลตอบสนอง ระดับความเชื่อถือได้ในตัวแปรอธิบาย และอัตราการจำแนกผิดในตัวแปรตามที่กำหนด กรณีที่ตัวแบบมีตัวแปรอธิบาย 1 5 และ 10 ตัว

```
Y1_r1b1 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
      Y1_r1b1[i,1] = ((X[i,1]*B_rho1[1,1])+(X[i,2]*B_rho1[2,1])+
                     (X[i,3]*B_rho1[3,1])+(X[i,4]*B_rho1[4,1])+
                     (X[i,5]*B_rho1[5,1])+(X[i,6]*B_rho1[6,1])+
                     (X[i,7]*B_rho1[7,1])+(X[i,8]*B_rho1[8,1])+
                     (X[i,9]*B_rho1[9,1])+(X[i,10]*B_rho1[10,1]))+
                     (eta_adj_r1b1[i,1] + error[i,1]);end;end;
      *print Y1_r1b1;

Y1_r1b2 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
      Y1_r1b2[i,1] = ((X[i,1]*B_rho1[1,1])+(X[i,2]*B_rho1[2,1])+
                     (X[i,3]*B_rho1[3,1])+(X[i,4]*B_rho1[4,1])+
                     (X[i,5]*B_rho1[5,1])+(X[i,6]*B_rho1[6,1])+
                     (X[i,7]*B_rho1[7,1])+(X[i,8]*B_rho1[8,1])+
                     (X[i,9]*B_rho1[9,1])+(X[i,10]*B_rho1[10,1]))+
                     (eta_adj_r1b2[i,1] + error[i,1]);end;end;
      *print Y1_r1b2;
Y1_r1b3 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
      Y1_r1b3[i,1] = ((X[i,1]*B_rho1[1,1])+(X[i,2]*B_rho1[2,1])+
                     (X[i,3]*B_rho1[3,1])+(X[i,4]*B_rho1[4,1])+
                     (X[i,5]*B_rho1[5,1])+(X[i,6]*B_rho1[6,1])+
                     (X[i,7]*B_rho1[7,1])+(X[i,8]*B_rho1[8,1])+
                     (X[i,9]*B_rho1[9,1])+(X[i,10]*B_rho1[10,1]))+
                     (eta_adj_r1b3[i,1] + error[i,1]);end; end;
      *print Y1_r1b3;
Y1_r2b1 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
      Y1_r2b1[i,1] = ((X[i,1]*B_rho2[1,1])+(X[i,2]*B_rho2[2,1])+
                     (X[i,3]*B_rho2[3,1])+(X[i,4]*B_rho2[4,1])+
                     (X[i,5]*B_rho2[5,1])+(X[i,6]*B_rho2[6,1])+
                     (X[i,7]*B_rho2[7,1])+(X[i,8]*B_rho2[8,1])+
                     (X[i,9]*B_rho2[9,1])+(X[i,10]*B_rho2[10,1]))+
                     (eta_adj_r2b1[i,1] + error[i,1]);end; end;
      *print Y1_r2b1;
Y1_r2b2 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
       Y1_r2b2[i,1] =((X[i,1]*B_rho2[1,1])+(X[i,2]*B_rho2[2,1])+
                     (X[i,3]*B_rho2[3,1])+(X[i,4]*B_rho2[4,1])+
                     (X[i,5]*B_rho2[5,1])+(X[i,6]*B_rho2[6,1])+
                     (X[i,7]*B_rho2[7,1])+(X[i,8]*B_rho2[8,1])+
                     (X[i,9]*B_rho2[9,1])+(X[i,10]*B_rho2[10,1]))+
                     (eta_adj_r2b2[i,1] + error[i,1]);end; end;
      *print Y1_r2b2;
Y1_r2b3 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
       Y1_r2b3[i,1] = ((X[i,1]*B_rho2[1,1])+(X[i,2]*B_rho2[2,1])+
                      (X[i,3]*B_rho2[3,1])+(X[i,4]*B_rho2[4,1])+
                      (X[i,5]*B_rho2[5,1])+(X[i,6]*B_rho2[6,1])+
                      (X[i,7]*B_rho2[7,1])+(X[i,8]*B_rho2[8,1])+
                      (X[i,9]*B_rho2[9,1])+(X[i,10]*B_rho2[10,1]))+
                      (eta_adj_r2b3[i,1] + error[i,1]);end; end;
      *print Y1_r2b3;
Y1_r3b1 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
       Y1_r3b1[i,1] = ((X[i,1]*B_rho3[1,1])+(X[i,2]*B_rho3[2,1])+
                      (X[i,3]*B_rho3[3,1])+(X[i,4]*B_rho3[4,1])+
```

```
                        (X[i,5]*B_rho3[5,1])+(X[i,6]*B_rho3[6,1])+
                        (X[i,7]*B_rho3[7,1])+(X[i,8]*B_rho3[8,1])+
                        (X[i,9]*B_rho3[9,1])+(X[i,10]*B_rho3[10,1]))+
                        (eta_adj_r3b1[i,1] + error[i,1]);end;end;
      *print Y1_r3b1;
Y1_r3b2 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
       Y1_r3b2[i,1] = ((X[i,1]*B_rho3[1,1])+(X[i,2]*B_rho3[2,1])+
                        (X[i,3]*B_rho3[3,1])+(X[i,4]*B_rho3[4,1])+
                        (X[i,5]*B_rho3[5,1])+(X[i,6]*B_rho3[6,1])+
                        (X[i,7]*B_rho3[7,1])+(X[i,8]*B_rho3[8,1])+
                        (X[i,9]*B_rho3[9,1])+(X[i,10]*B_rho3[10,1]))+
                        (eta_adj_r3b2[i,1] + error[i,1]);end;end;
      *print Y1_r3b2;
Y1_r3b3 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
       Y1_r3b3[i,1] = ((X[i,1]*B_rho3[1,1])+(X[i,2]*B_rho3[2,1])+
                        (X[i,3]*B_rho3[3,1])+(X[i,4]*B_rho3[4,1])+
                        (X[i,5]*B_rho3[5,1])+(X[i,6]*B_rho3[6,1])+
                        (X[i,7]*B_rho3[7,1])+(X[i,8]*B_rho3[8,1])+
                        (X[i,9]*B_rho3[9,1])+(X[i,10]*B_rho3[10,1]))+
                        (eta_adj_r3b3[i,1] + error[i,1]);end;end;
      *print Y1_r3b3;
Y1_rel1 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
       Y1_rel1[i,1] = ((X[i,11]*B_rho3[1,1])+(X[i,12]*B_rho3[2,1])+
                        (X[i,13]*B_rho3[3,1])+(X[i,14]*B_rho3[4,1])+
                        (X[i,15]*B_rho3[5,1])+(X[i,16]*B_rho3[6,1])+
                        (X[i,17]*B_rho3[7,1])+(X[i,18]*B_rho3[8,1])+
                        (X[i,19]*B_rho3[9,1])+(X[i,20]*B_rho3[10,1]))+
                        (eta_adj_rel1[i,1] + error[i,1]); end; end;
      *print Y1_rel1;
Y1_rel2 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
       Y1_rel2[i,1] = ((X[i,21]*B_rho3[1,1])+(X[i,22]*B_rho3[2,1])+
                        (X[i,23]*B_rho3[3,1])+(X[i,24]*B_rho3[4,1])+
                        (X[i,25]*B_rho3[5,1])+(X[i,26]*B_rho3[6,1])+
                        (X[i,27]*B_rho3[7,1])+(X[i,28]*B_rho3[8,1])+
                        (X[i,29]*B_rho3[9,1])+(X[i,30]*B_rho3[10,1]))+
                        (eta_adj_rel2[i,1] + error[i,1]);end; end;
      *print Y1_rel2;
Y1_rel3 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
       Y1_rel3[i,1] = ((X[i,31]*B_rho3[1,1])+(X[i,32]*B_rho3[2,1])+
                        (X[i,33]*B_rho3[3,1])+(X[i,34]*B_rho3[4,1])+
                        (X[i,35]*B_rho3[5,1])+(X[i,36]*B_rho3[6,1])+
                        (X[i,37]*B_rho3[7,1])+(X[i,38]*B_rho3[8,1])+
                        (X[i,39]*B_rho3[9,1])+(X[i,40]*B_rho3[10,1]))+
                        (eta_adj_rel3[i,1] + error[i,1]);end; end;
      *print Y1_rel3;
Y1_rel4 = j(&n_sample,&k-9,0);do i = 1 to &n_sample;do j = 1 to &k-9;
       Y1_rel4[i,1] = ((X[i,41]*B_rho3[1,1])+(X[i,42]*B_rho3[2,1])+
                        (X[i,43]*B_rho3[3,1])+(X[i,44]*B_rho3[4,1])+
                        (X[i,45]*B_rho3[5,1])+(X[i,46]*B_rho3[6,1])+
                        (X[i,47]*B_rho3[7,1])+(X[i,48]*B_rho3[8,1])+
                        (X[i,49]*B_rho3[9,1])+(X[i,40]*B_rho3[10,1]))+
                        (eta_adj_rel4[i,1] + error[i,1]);end; end;
      *print Y1_rel3;
```

แปลงค่าตัวแปรตามภายใต้เงื่อนไขของสัดส่วนผลตอบสนอง ระดับความเชื่อถือได้ในตัวแปรอธิบาย และอัตราการจำแนกผิดในตัวแปรตามที่กำหนด ให้มีค่า 0 หรือ 1

```
Y_r1b1 = j(&n_sample,&k,0); do i = 1 to &n_sample; do j = 1 to &k;
       if(Y1_r1b1[i,j] >= C_0) then Y_r1b1[i,j] = 1;
       if(Y1_r1b1[i,j] < C_0) then Y_r1b1[i,j] = 0;end ;end ;
*print Y_r1b1 ;
 Y_r1b2 = j(&n_sample,&k,0); do i = 1 to &n_sample; do j = 1 to &k;
       if(Y1_r1b2[i,j] >= C_0) then Y_r1b2[i,j] = 1;
       if(Y1_r1b2[i,j] < C_0) then Y_r1b2[i,j] = 0;end ; end ;
 *print Y_r1b2 ;
 Y_r1b3 = j(&n_sample,&k,0);do i = 1 to &n_sample;do j = 1 to &k;
       if(Y1_r1b3[i,j] >= C_0) then Y_r1b3[i,j] = 1;
       if(Y1_r1b3[i,j] < C_0) then Y_r1b3[i,j] = 0;end ; end ;
 *print Y_r1b3 ;
Y_r2b1 = j(&n_sample,&k,0);do i = 1 to &n_sample;do j = 1 to &k;
       if(Y1_r2b1[i,j] >= C_0) then Y_r2b1[i,j] = 1;
       if(Y1_r2b1[i,j] < C_0) then Y_r2b1[i,j] = 0;end ; end ;
 *print Y_r2b1 ;
 Y_r2b2 = j(&n_sample,&k,0);do i = 1 to &n_sample;do j = 1 to &k;
       if(Y1_r2b2[i,j] >= C_0) then Y_r2b2[i,j] = 1;
       if(Y1_r2b2[i,j] < C_0) then Y_r2b2[i,j] = 0;end ; end ;
 *print Y_r2b2 ;
 Y_r2b3 = j(&n_sample,&k,0);do i = 1 to &n_sample;do j = 1 to &k;
       if(Y1_r2b3[i,j] >= C_0) then Y_r2b3[i,j] = 1;
       if(Y1_r2b3[i,j] < C_0) then Y_r2b3[i,j] = 0;end ; end ;
 *print Y_r2b3 ;
Y_r3b1 = j(&n_sample,&k,0);do i = 1 to &n_sample; do j = 1 to &k;
       if(Y1_r3b1[i,j] >= C_0) then Y_r3b1[i,j] = 1;
       if(Y1_r3b1[i,j] < C_0) then Y_r3b1[i,j] = 0;end ; end ;
 *print Y_r3b1 ;
 Y_r3b2 = j(&n_sample,&k,0); do i = 1 to &n_sample; do j = 1 to &k;
       if(Y1_r3b2[i,j] >= C_0) then Y_r3b2[i,j] = 1;
       if(Y1_r3b2[i,j] < C_0) then Y_r3b2[i,j] = 0;end ; end ;
 *print Y_r3b2 ;
 Y_r3b3 = j(&n_sample,&k,0); do i = 1 to &n_sample; do j = 1 to &k;
       if(Y1_r3b3[i,j] >= C_0) then Y_r3b3[i,j] = 1;
       if(Y1_r3b3[i,j] < C_0) then Y_r3b3[i,j] = 0;end ; end ;
 *print Y_r3b3 ;
Y_rel1 = j(&n_sample,&k,0); do i = 1 to &n_sample; do j = 1 to &k;
       if(Y1_rel1[i,j] >= C_0) then Y_rel1[i,j] = 1;
       if(Y1_rel1[i,j] < C_0) then Y_rel1[i,j] = 0; end ; end ;
 *print Y_rel1 ;
 Y_rel2 = j(&n_sample,&k,0); do i = 1 to &n_sample; do j = 1 to &k;
       if(Y1_rel2[i,j] >= C_0) then Y_rel2[i,j] = 1;
       if(Y1_rel2[i,j] < C_0) then Y_rel2[i,j] = 0; end ; end ;
 *print Y_rel2 ;
 Y_rel3 = j(&n_sample,&k,0); do i = 1 to &n_sample; do j = 1 to &k;
       if(Y1_rel3[i,j] >= C_0) then Y_rel3[i,j] = 1;
       if(Y1_rel3[i,j] < C_0) then Y_rel3[i,j] = 0; end ; end ;
 *print Y_rel3 ;
 Y_rel4 = j(&n_sample,&k,0); do i = 1 to &n_sample; do j = 1 to &k;
       if(Y1_rel4[i,j] >= C_0) then Y_rel4[i,j] = 1;
       if(Y1_rel4[i,j] < C_0) then Y_rel4[i,j] = 0; end ; end ;
 *print Y_rel4 ;
```

จัดชุดข้อมูลของสัมประสิทธิ์การถดถอยให้อยู่ในรูปเมตริกซ์

```
D = j(&n_sample,&k+22,0);
    do i = 1 to &n_sample;do j=1 to 4; D[i,j]=X[i,j];end ;end;
    do i = 1 to &n_sample;do j=5 to 5; D[i,5]=error[i,1] ;end; end;
    do i = 1 to &n_sample;do j=6 to 6; D[i,6]= Y_r1b1[i,1];end; end;
    do i = 1 to &n_sample;do j=7 to 7; D[i,7]= Y_r1b2[i,1];end; end;
    do i = 1 to &n_sample;do j=8 to 8; D[i,8]= Y_r1b3[i,1];end; end;
    do i = 1 to &n_sample;do j=9 to 9; D[i,9]= Y_r2b1[i,1];end; end;
    do i = 1 to &n_sample;do j=10to 10;D[i,10]=Y_r2b2[i,1];end; end;
    do i = 1 to &n_sample;do j=11to 11;D[i,11]=Y_r2b3[i,1];end; end;
    do i = 1 to &n_sample;do j=12to 12;D[i,12]= Y_r3b1[i,1];end; end;
    do i = 1 to &n_sample;do j=13to 13;D[i,13]= Y_r3b2[i,1];end; end;
    do i = 1 to &n_sample;do j=14to 14;D[i,14]= Y_r3b3[i,1];end; end;
    do i = 1 to &n_sample;do j=15to 15;D[i,15]= Y_new11[i,1];end;end;
    do i = 1 to &n_sample;do j=16to 16;D[i,16]= Y_new12[i,1];end;end;
    do i = 1 to &n_sample;do j=17to 17;D[i,17]= Y_new13[i,1];end;end;
    do i = 1 to &n_sample;do j=18to 18;D[i,18]= Y_new21[i,1];end;end;
    do i = 1 to &n_sample;do j=19to 19;D[i,19]= Y_new22[i,1];end;end;
    do i = 1 to &n_sample;do j=20to 20;D[i,20]= Y_new23[i,1];end;end;
    do i = 1 to &n_sample;do j=21to 21;D[i,21]= Y_rel1[i,1];end;end;
    do i = 1 to &n_sample;do j=22to 22;D[i,22]= Y_rel2[i,1];end;end;
    do i = 1 to &n_sample;do j=23to 23;D[i,23]= Y_rel3[i,1];end;end;
*print D ; finish;  run D;
create Data from D; append from D; show datasets;show contents;
*proc print data = Data;*run;
```

การประมาณค่าสัมประสิทธิ์การถดถอยของตัวแบบโลจิสติกกรณีที่ตัวแบบประกอบด้วยตัวแปรอธิบาย 1 5 และ 10 ตัว

```
proc logistic descending noprint ;
model COL42 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_r1b1 predicted = Pi_i_r1b1;run;
*proc print data = data_new_r1b1 ;*var COL42 Pi_i_r1b1 ;*run ;

proc logistic descending noprint ;
model COL43 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_r1b2 predicted = Pi_i_r1b2;run;
*proc print data = data_new_r1b2 ;*var COL43 Pi_i_r1b2 ;*run ;

proc logistic descending noprint ;
model COL44 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_r1b3 predicted = Pi_i_r1b3;run;
*proc print data = data_new_r1b3 ;*var COL44 Pi_i_r1b3 ;*run ;

proc logistic descending noprint ;
model COL45 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_r2b1 predicted = Pi_i_r2b1;run;
*proc print data = data_new_r2b1 ;*var COL45 Pi_i_r2b1 ;*run ;

proc logistic descending noprint ;
model COL46 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_r2b2 predicted = Pi_i_r2b2;run;
*proc print data = data_new_r2b2 ;*var COL46 Pi_i_r2b2 ;*run ;
```

```
proc logistic descending noprint ;
model COL47 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_r2b3 predicted = Pi_i_r2b3;run;
*proc print data = data_new_r2b3 ;*var COL47 Pi_i_r2b3 ;*run ;

proc logistic descending noprint ;
model COL48 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_r3b1 predicted = Pi_i_r3b1;run;
*proc print data = data_new_r3b1 ;*var COL48 Pi_i_r3b1 ;*run ;

proc logistic descending noprint ;
model COL49 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_r3b2 predicted = Pi_i_r3b2;run;
*proc print data = data_new_r3b2 ;*var COL49 Pi_i_r3b2 ;*run ;

proc logistic descending noprint ;
model COL50 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_r3b3 predicted = Pi_i_r3b3;run;
*proc print data = data_new_r3b3 ;*var COL50 Pi_i_r3b3 ;*run ;

proc logistic descending noprint ;
model COL51 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_m11 predicted = Pi_i_m11;run;
*proc print data = data_new_m11 ;*var COL51 Pi_i_m11 ;*run ;

proc logistic descending noprint ;
model COL52 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_m12 predicted = Pi_i_m12;run;
*proc print data = data_new_m12 ;*var COL52 Pi_i_m12 ;*run ;

proc logistic descending noprint ;
model COL53 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_m13 predicted = Pi_i_m13;run;
*proc print data = data_new_m13 ;*var COL53 Pi_i_m13 ;*run ;

proc logistic descending noprint ;
model COL54 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_m21 predicted = Pi_i_m21;run;
*proc print data = data_new_m21 ;*var COL54 Pi_i_m21 ;*run ;

proc logistic descending noprint ;
model COL55 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_m22 predicted = Pi_i_m22;run;
*proc print data = data_new_m22 ;*var COL55 Pi_i_m22 ;*run ;

proc logistic descending noprint ;
model COL56 = COL1 COL2 COL3 COL4 COL5 COL6 COL7 COL8 COL9 COL10;
output out = data_new_m23 predicted = Pi_i_m23;run;
*proc print data = data_new_m23 ;*var COL56 Pi_i_m23 ;*run ;

proc logistic descending noprint ;
model COL57 = COL11 COL12 COL13 COL14 COL15 COL16 COL17 COL18 COL19
COL20 ;output out = data_new_rel1 predicted = Pi_i_rel1;run;
*proc print data = data_new_rel1 ;*var COL57 Pi_i_rel1 ;*run ;

proc logistic descending noprint ;
model COL58 = COL21 COL22 COL23 COL24 COL25 COL26 COL27 COL28 COL29
COL30 ;output out = data_new_rel2 predicted = Pi_i_rel2;run;
```

```
*proc print data = data_new_rel2 ;*var COL58 Pi_i_rel2 ;*run ;

proc logistic descending noprint ;
model COL59 = COL31 COL32 COL33 COL34 COL35 COL36 COL37 COL38 COL39
COL40 ;output out = data_new_rel3 predicted = Pi_i_rel3;run;
*proc print data = data_new_rel3 ;*var COL59 Pi_i_rel3 ;*run ;

proc logistic descending noprint ;
model COL60 = COL31 COL32 COL33 COL34 COL35 COL36 COL37 COL38 COL39
COL40 ;output out = data_new_rel4 predicted = Pi_i_rel4;run;
*proc print data = data_new_rel4 ;*var COL60 Pi_i_rel4 ;*run ;

ods OUTPUT ;proc Means noprint;title 'Descriptive Statistics';
var COL42 COL43 COL44 COL45 COL46 COL47 COL48 COL49 COL50
    COL51 COL52 COL53 COL54 COL55 COL56 COL57 COL58 COL59 COL60;
OUTPUT OUT=OUTA MEAN=MY_r1b1 MY_r1b2 MY_r1b3 MY_r2b1 MY_r2b2 MY_r2b3
    MY_r3b1 MY_r3b2 MY_r3b3 MY_m11 MY_m12 MY_m13 MY_m21 MY_m22
    MY_m23 MY_rel1 MY_rel2 MY_rel3 MY_rel4;run ;
data Mean_y(keep = order MY_r1b1 MY_r1b2 MY_r1b3 MY_r2b1
    MY_r2bMY_r2b3 MY_r3b1 MY_r3b2 MY_r3b3 MY_m11 MY_m12 MY_m13
    MY_m21 MY_m22 MY_m23 MY_rel1 MY_rel2 MY_rel3 MY_rel4) ;
set OUTA ; run ;*proc print data = Mean_y ;*run ;

data Mean_ym ;  set Mean_y ;  retain _seed_ 0;
      do _i_ = 1 to &n_sample;
      Mean_ym_r1b1 = MY_r1b1 ;       Mean_ym_r1b2 = MY_r1b2 ;
      Mean_ym_r1b3 = MY_r1b3 ;       Mean_ym_r2b1 = MY_r2b1 ;
      Mean_ym_r2b2 = MY_r2b2 ;       Mean_ym_r2b3 = MY_r2b3 ;
      Mean_ym_r3b1 = MY_r3b1 ;       Mean_ym_r3b2 = MY_r3b2 ;
      Mean_ym_r3b3 = MY_r3b3 ;       Mean_ym_m11 = MY_m11 ;
      Mean_ym_m12 = MY_m12 ;         Mean_ym_m13 = MY_m13 ;
      Mean_ym_m21 = MY_m21 ;         Mean_ym_m22 = MY_m22 ;
      Mean_ym_m23 = MY_m23 ;         Mean_ym_rel1 = MY_rel1 ;
      Mean_ym_rel2 = MY_rel2 ;       Mean_ym_rel3 = MY_rel3 ;
      Mean_ym_rel4 = MY_rel4 ;output; end; drop _seed_ _i_; run;
*proc print data = Mean_ym ;
*var Mean_ym_r1b1 Mean_ym_r1b2 Mean_ym_r1b3 Mean_ym_r2b1 Mean_ym_r2b2
     Mean_ym_r2b3 Mean_ym_r3b1 Mean_ym_r3b2 Mean_ym_r3b3 Mean_ym_m11
     Mean_ym_m12 Mean_ym_m13   Mean_ym_m21 Mean_ym_m22 Mean_ym_m23
     Mean_ym_rel1 Mean_ym_rel2 Mean_ym_rel3 Mean_ym_rel4;*run;
```

ก.3 โปรแกรมสำหรับการคำนวณค่าสัมประสิทธิ์การตัดสินใจที่ปรับค่า $\left(R^2_{adj}\right)$ แบบต่างๆ

กรณีที่ตัวแบบการถดถอยโลจิสติกประกอบด้วยตัวแปรอธิบาย 1 5 และ 10 ตัว

เมื่อค่าสัมประสิทธิ์การตัดสินใจที่แท้จริงมีค่า 0.1 และสัดส่วนผลตอบสนองมีค่า 0.1 0.3 และ 0.5

```
data Dif_r1b1 ;  set data_new_r1b1 ;  set Mean_ym ;
AA_r1b1 =(COL6 - Pi_i_r1b1)**2 ;BB_r1b1 = (COL6 - Mean_ym_r1b1)**2 ;
LM_r1b1 =(COL6*(log(Pi_i_r1b1)))+((1-COL6)*(log(1-Pi_i_r1b1)));
L0_r1b1 =(COL6*(log(Mean_ym_r1b1)))+((1-COL6)*(log(1-Mean_ym_r1b1)));
run ; *proc print data = Dif_r1b1 ;
*var COL6 Pi_i_r1b1 AA_r1b1 BB_r1b1 LM_r1b1 L0_r1b1;*run;

proc means data = Dif_r1b1 noprint;
var AA_r1b1 BB_r1b1 LM_r1b1 L0_r1b1 COL6;
output out = out_dif_r1b1 sum = AA_r1b1 BB_r1b1 LM_r1b1 L0_r1b1
SCOL6; run; *proc print data = out_dif_r1b1 ;*run ;

data R_r1b1 ;  set out_dif_r1b1 ;
BASE_r1b1 = SCOL6/&n_sample ; AAA_r1b1 = (&n_sample-1)*AA_r1b1 ;
BBB_r1b1 = (&n_sample-&k-1)*BB_r1b1 ;CCC_r1b1 = AAA_r1b1 / BBB_r1b1 ;
DDD_r1b1 = ((LM_r1b1 -((&k+1)/2)));  EEE_r1b1 = (L0_r1b1 - 0.5);
FFF_r1b1 = DDD_r1b1 / EEE_r1b1 ;
GGG_r1b1 = LM_r1b1 -(((&k+1)*(&n_sample-1))/(&n_sample-&k-1));
HHH_r1b1 = (L0_r1b1 - 1) ;     III_r1b1 = GGG_r1b1 / HHH_r1b1 ;
JJJ_r1b1 = AA_r1b1 / BB_r1b1 ; KKK_r1b1 = LM_r1b1 / L0_r1b1 ;
R_OMS_r1b1 = 1 - CCC_r1b1 ;    R_LMS_r1b1 = 1 - FFF_r1b1 ;
R_LAIC_r1b1 = 1 - III_r1b1 ;   R_OLM_r1b1 = 1 - JJJ_r1b1 ;
R_LLM_r1b1 = 1 - KKK_r1b1 ;    run ;/*proc print data = R_r1b1 ;
var R_OMS_r1b1 R_LMS_r1b1 R_LAIC_r1b1 R_OLM_r1b1 R_LLM_r1b1
    BASE_r1b1;run;*/
```

เมื่อค่าสัมประสิทธิ์การตัดสินใจที่แท้จริงมีค่า 0.5 และสัดส่วนผลตอบสนองมีค่า 0.1 0.3 และ 0.5

```
data Dif_r2b1 ; set data_new_r2b1 ;  set Mean_ym ;
AA_r2b1 = (COL9 - Pi_i_r2b1)**2 ;BB_r2b1 = (COL9 - Mean_ym_r2b1)**2 ;
LM_r2b1 = (COL9*(log(Pi_i_r2b1)))+((1-COL9)*(log(1-Pi_i_r2b1)));
L0_r2b1 = (COL9*(log(Mean_ym_r2b1)))+((1-COL9)*(log(1-ean_ym_r2b1)));
run ;*proc print data = Dif_r2b1 ;
*var COL9 Pi_i_r2b1 AA_r2b1 BB_r2b1 LM_r2b1 L0_r2b1;*run;

proc means data = Dif_r2b1 noprint;
var AA_r2b1 BB_r2b1 LM_r2b1 L0_r2b1 COL9;
output out = out_dif_r2b1 sum = AA_r2b1 BB_r2b1 LM_r2b1 L0_r2b1
SCOL9;run;*proc print data = out_dif_r2b1 ;*run ;

data R_r2b1 ;
set out_dif_r2b1 ;  BASE_r2b1 = SCOL9/&n_sample ;
AAA_r2b1 = (&n_sample-1)*AA_r2b1;BBB_r2b1 =(&n_sample-&k-1)*BB_r2b1 ;
CCC_r2b1 = AAA_r2b1 / BBB_r2b1 ; DDD_r2b1 = ((LM_r2b1 -((&k+1)/2)));
EEE_r2b1 = (L0_r2b1 - 0.5);      FFF_r2b1 = DDD_r2b1 / EEE_r2b1 ;
GGG_r2b1 = LM_r2b1 -(((&k+1)*(&n_sample-1))/(&n_sample-&k-1));
HHH_r2b1 = (L0_r2b1 - 1) ;       III_r2b1 = GGG_r2b1 / HHH_r2b1 ;
JJJ_r2b1 = AA_r2b1 / BB_r2b1 ;   KKK_r2b1 = LM_r2b1 / L0_r2b1 ;
R_OMS_r2b1 = 1 - CCC_r2b1 ;      R_LMS_r2b1 = 1 - FFF_r2b1 ;
R_LAIC_r2b1 = 1 - III_r2b1 ;     R_OLM_r2b1 = 1 - JJJ_r2b1 ;
R_LLM_r2b1 = 1 - KKK_r2b1 ;      run ;
*proc print data = R_r2b1 ;
```

```
*var R_OMS_r2b1 R_LMS_r2b1 R_LAIC_r2b1 R_OLM_r2b1 R_LLM_r2b1
BASE_r2b1;*run;
```

เมื่อค่าสัมประสิทธิ์การตัดสินใจที่แท้จริงมีค่า 0.9 และสัดส่วนผลตอบสนองมีค่า 0.1 0.3 และ 0.5

```
data Dif_r3b1 ;  set data_new_r3b1 ;  set Mean_ym ;
AA_r3b1 = (COL12 - Pi_i_r3b1)**2;BB_r3b1 =(COL12 - Mean_ym_r3b1)**2 ;
LM_r3b1 = (COL12*(log(Pi_i_r3b1)))+((1-COL12)*(log(1-Pi_i_r3b1)));
L0_r3b1 = (COL12*(log(Mean_ym_r3b1)))+((1-COL12)*
          (log(1-Mean_ym_r3b1)));run ;*proc print data = Dif_r3b1 ;
*var COL12 Pi_i_r3b1 AA_r3b1 BB_r3b1 LM_r3b1 L0_r3b1;*run;

proc means data = Dif_r3b1 noprint;
var AA_r3b1 BB_r3b1 LM_r3b1 L0_r3b1 COL12;
output out = out_dif_r3b1 sum = AA_r3b1 BB_r3b1 LM_r3b1 L0_r3b1
SCOL12;run;*proc print data = out_dif_r3b1 ;*run ;

data R_r3b1 ;  set out_dif_r3b1 ;  BASE_r3b1 = SCOL12/&n_sample ;
AAA_r3b1 = (&n_sample-1)*AA_r3b1;BBB_r3b1 =(&n_sample-&k-1)*BB_r3b1 ;
CCC_r3b1 = AAA_r3b1 / BBB_r3b1 ; DDD_r3b1 = ((LM_r3b1 -((&k+1)/2)));
EEE_r3b1 = (L0_r3b1 - 0.5);  FFF_r3b1 = DDD_r3b1 / EEE_r3b1 ;
GGG_r3b1 = LM_r3b1 -(((&k+1)*(&n_sample-1))/(&n_sample-&k-1));
HHH_r3b1 = (L0_r3b1 - 1) ;   III_r3b1 = GGG_r3b1 / HHH_r3b1 ;
JJJ_r3b1 = AA_r3b1 / BB_r3b1 ;   KKK_r3b1 = LM_r3b1 / L0_r3b1 ;
R_OMS_r3b1 = 1 - CCC_r3b1 ;   R_LMS_r3b1 = 1 - FFF_r3b1 ;
R_LAIC_r3b1 = 1 - III_r3b1 ;  R_OLM_r3b1 = 1 - JJJ_r3b1 ;
R_LLM_r3b1 = 1 - KKK_r3b1 ;run ;*proc print data = R_r3b1 ;
*var R_OMS_r3b1 R_LMS_r3b1 R_LAIC_r3b1 R_OLM_r3b1 R_LLM_r3b1
BASE_r3b1;*run;
```

เมื่ออัตราการจำแนกผิดในตัวแปรตามมีค่าเท่ากับ เท่ากับ 2% และ 10% และมีการจำแนกค่า 0 ผิดทั้งหมด

```
data Dif_m11 ;  set data_new_m11 ;  set Mean_ym ;
AA_m11 = (COL15- Pi_i_m11)**2 ; BB_m11 = (COL15 - Mean_ym_m11)**2 ;
LM_m11 = (COL15*(log(Pi_i_m11)))+((1-COL15)*(log(1-Pi_i_m11)));
L0_m11 = (COL15*(log(Mean_ym_m11)))+((1-COL15)*(log(1-Mean_ym_m11)));
run ; *proc print data = Dif_m11 ;
*var COL15 Pi_i_m11 AA_m11 BB_m11 LM_m11 L0_m11;*run;

proc means data = Dif_m11 noprint;
var AA_m11 BB_m11 LM_m11 L0_m11 COL15;
output out = out_dif_m11 sum = AA_m11 BB_m11 LM_m11 L0_m11 SCOL15;
run; *proc print data = out_dif_m11 ; *run ;
data R_m11 ;  set out_dif_m11 ;  BASE_m11 = SCOL15/&n_sample ;
AAA_m11 = (&n_sample-1)*AA_m11 ; BBB_m11 = (&n_sample-&k-1)*BB_m11 ;
CCC_m11 = AAA_m11 / BBB_m11 ;    DDD_m11 = LM_m11 -((&k+1)/2);
EEE_m11 = L0_m11 - 0.5;    FFF_m11 = DDD_m11 / EEE_m11 ;
GGG_m11 = LM_m11 -(((&k+1)*(&n_sample-1))/(&n_sample-&k-1));
HHH_m11 = (L0_m11 - 1) ;   III_m11 = GGG_m11 / HHH_m11 ;
JJJ_m11 = AA_m11 / BB_m11 ;   KKK_m11 = LM_m11 / L0_m11 ;
R_OMS_m11 = 1 - CCC_m11 ;   R_LMS_m11 = 1 - FFF_m11 ;
R_LAIC_m11 = 1 - III_m11 ;  R_OLM_m11 = 1 - JJJ_m11 ;
R_LLM_m11 = 1 - KKK_m11 ;  run ; *proc print data = R_m11 ;
*var R_OMS_m11 R_LMS_m11 R_LAIC_m11 R_OLM_m11 R_LLM_m11 BASE_m11;
*run;
```

เมื่ออัตราการจำแนกผิดในตัวแปรตามมีค่าเท่ากับ เท่ากับ 2% และ 10% และมีการจำแนกค่า 1 ผิดทั้งหมด

```
data Dif_m12 ;  set data_new_m12 ;  set Mean_ym ;
AA_m12 = (COL16 - Pi_i_m12)**2 ;  BB_m12 = (COL16 - Mean_ym_m12)**2 ;
LM_m12 = (COL16*(log(Pi_i_m12)))+((1-COL16)*(log(1-Pi_i_m12)));
L0_m12 = (COL16*(log(Mean_ym_m12)))+((1-COL16)*(log(1-Mean_ym_m12)));
run ;*proc print data = Dif_m12 ;
*var COL16 Pi_i_m12 AA_m12 BB_m12 LM_m12 L0_m12;*run;

proc means data = Dif_m12 noprint;
var AA_m12 BB_m12 LM_m12 L0_m12 COL16;
output out = out_dif_m12 sum = AA_m12 BB_m12 LM_m12 L0_m12 SCOL16;
run; *proc print data = out_dif_m12 ; *run ;

data R_m12 ;  set out_dif_m12 ;  BASE_m12 = SCOL16/&n_sample ;
AAA_m12 = (&n_sample-1)*AA_m12 ; BBB_m12 = (&n_sample-&k-1)*BB_m12 ;
CCC_m12 = AAA_m12 / BBB_m12 ;    DDD_m12 = LM_m12 -((&k+1)/2);
EEE_m12 = L0_m12 - 0.5;          FFF_m12 = DDD_m12 / EEE_m12 ;
GGG_m12 = LM_m12 -(((&k+1)*(&n_sample-1))/(&n_sample-&k-1));
HHH_m12 = (L0_m12 - 1) ;         III_m12 = GGG_m12 / HHH_m12 ;
JJJ_m12 = AA_m12 / BB_m12 ;      KKK_m12 = LM_m12 / L0_m12 ;
R_OMS_m12 = 1 - CCC_m12 ;        R_LMS_m12 = 1 - FFF_m12 ;
R_LAIC_m12 = 1 - III_m12 ;       R_OLM_m12 = 1 - JJJ_m12 ;
R_LLM_m12 = 1 - KKK_m12 ; run ; *proc print data = R_m12 ;
*var R_OMS_m12 R_LMS_m12 R_LAIC_m12 R_OLM_m12 R_LLM_m12 BASE_m12;
*run;
```

เมื่ออัตราการจำแนกผิดในตัวแปรตามมีค่าเท่ากับ 2% และ 10% และมีการจำแนกค่า 0 และ 1 ผิดอย่างละครึ่ง

```
data Dif_m13 ;  set data_new_m13 ;  set Mean_ym ;
AA_m13 = (COL17 - Pi_i_m13)**2 ;  BB_m13 = (COL17 - Mean_ym_m13)**2 ;
LM_m13 = (COL17*(log(Pi_i_m13)))+((1-COL17)*(log(1-Pi_i_m13)));
L0_m13 = (COL17*(log(Mean_ym_m13)))+((1-COL17)*(log(1-Mean_ym_m13)));
run ; *proc print data = Dif_m13 ;
*var COL17 Pi_i_m13 AA_m13 BB_m13 LM_m13 L0_m13;*run;

proc means data = Dif_m13 noprint;
var AA_m13 BB_m13 LM_m13 L0_m13 COL17;
output out = out_dif_m13 sum = AA_m13 BB_m13 LM_m13 L0_m13 SCOL17;
run; *proc print data = out_dif_m13 ; *run ;

data R_m13 ;  set out_dif_m13 ;   BASE_m13 = SCOL17/&n_sample ;
AAA_m13 = (&n_sample-1)*AA_m13 ;  BBB_m13 = (&n_sample-&k-1)*BB_m13 ;
CCC_m13 = AAA_m13 / BBB_m13 ;     DDD_m13 = LM_m13 -((&k+1)/2);
EEE_m13 = L0_m13 - 0.5;           FFF_m13 = DDD_m13 / EEE_m13 ;
GGG_m13 = LM_m13 -(((&k+1)*(&n_sample-1))/(&n_sample-&k-1));
HHH_m13 = (L0_m13 - 1) ;          III_m13 = GGG_m13 / HHH_m13 ;
JJJ_m13 = AA_m13 / BB_m13 ;       KKK_m13 = LM_m13 / L0_m13 ;
R_OMS_m13 = 1 - CCC_m13 ;         R_LMS_m13 = 1 - FFF_m13 ;
R_LAIC_m13 = 1 - III_m13 ;        R_OLM_m13 = 1 - JJJ_m13 ;
R_LLM_m13 = 1 - KKK_m13 ;run ; *proc print data = R_m13 ;
*var R_OMS_m13 R_LMS_m13 R_LAIC_m13 R_OLM_m13 R_LLM_m13 BASE_m13;
*run
```

เมื่อระดับความเชื่อถือได้ในตัวแปรอธิบายมีค่าเท่ากับ 0.3 0.5 0.7 และ 0.9

```
data Dif_rel1 ;  set data_new_rel1 ;  set Mean_ym ;
AA_rel1 = (COL21 - Pi_i_rel1)**2; BB_rel1 =(COL21 - Mean_ym_rel1)**2;
LM_rel1 = (COL21*(log(Pi_i_rel1)))+((1-COL21)*(log(1-Pi_i_rel1)));
L0_rel1 = (COL21*(log(Mean_ym_rel1)))+((1-COL21)*
          (log(1-Mean_ym_rel1))); run ;*proc print data = Dif_rel1 ;
*var COL21 Pi_i_rel1 AA_rel1 BB_rel1 LM_rel1 L0_rel1; *run;

proc means data = Dif_rel1 noprint;
var AA_rel1 BB_rel1 LM_rel1 L0_rel1 COL21;
output out = out_dif_rel1 sum = AA_rel1 BB_rel1 LM_rel1 L0_rel1
SCOL21;run;*proc print data = out_dif_rel1 ; *run ;

data R_rel1 ;  set out_dif_rel1 ; BASE_rel1 = SCOL21/&n_sample ;
AAA_rel1 = (&n_sample-1)*AA_rel1;BBB_rel1 = (&n_sample-&k-1)*BB_rel1;
CCC_rel1 = AAA_rel1 / BBB_rel1 ; DDD_rel1 = LM_rel1 -((&k+1)/2);
EEE_rel1 = L0_rel1 - 0.5;        FFF_rel1 = DDD_rel1 / EEE_rel1 ;
GGG_rel1 = LM_rel1 -(((&k+1)*(&n_sample-1))/(&n_sample-&k-1));
HHH_rel1 = (L0_rel1 - 1) ;       III_rel1 = GGG_rel1 / HHH_rel1 ;
JJJ_rel1 = AA_rel1 / BB_rel1 ;   KKK_rel1 = LM_rel1 / L0_rel1 ;
R_OMS_rel1 = 1 - CCC_rel1 ;      R_LMS_rel1 = 1 - FFF_rel1 ;
R_LAIC_rel1 = 1 - III_rel1 ;     R_OLM_rel1 = 1 - JJJ_rel1 ;
R_LLM_rel1 = 1 - KKK_rel1 ; run ;*proc print data = R_rel1 ;
*var R_OMS_rel1 R_LMS_rel1 R_LAIC_rel1 R_OLM_rel1 R_LLM_rel1
BASE_rel1;*run;
```

คำนวณค่าความคลาดเคลื่อนกำลังสองเฉลี่ย (Mean Square Error, MSE) ของค่าประมาณสัมประสิทธิ์การตัดสินใจที่ปรับค่า

```
data MSE;
set R_r1b1 ;set R_r1b2 ;set R_r1b3 ;set R_r2b1 ;set R_r2b2 ;
set R_r2b3 ;set R_r3b1 ;set R_r3b2 ;set R_r3b3 ;set R_m11 ;
set R_m12 ;set R_m13 ;set R_m21 ;set R_m22 ;
set R_m23 ;set R_rel1 ;set R_rel2 ;set R_rel3 ;

ER_OMS_r1b1=(0.1 - R_OMS_r1b1)**2; ER_LMS_r1b1 =(0.1-R_LMS_r1b1)**2;
ER_LAIC_r1b1 = (0.1-R_LAIC_r1b1)**2;ER_OLM_r1b1=(0.1-R_OLM_r1b1)**2;
ER_LLM_r1b1 = (0.1-R_LLM_r1b1)**2;

ER_OMS_r1b2=(0.1-R_OMS_r1b2)**2; ER_LMS_r1b2=(0.1-R_LMS_r1b2)**2;
ER_LAIC_r1b2 =(0.1-R_LAIC_r1b2)**2;ER_OLM_r1b2 = (0.1-R_OLM_r1b2)**2;
ER_LLM_r1b2 = (0.1-R_LLM_r1b2)**2;

ER_OMS_r1b3 =(0.1-R_OMS_r1b3)**2;ER_LMS_r1b3 =(0.1-R_LMS_r1b3)**2;
ER_LAIC_r1b3 = (0.1-R_LAIC_r1b3)**2;ER_OLM_r1b3 =(0.1-R_OLM_r1b3)**2;
ER_LLM_r1b3 = (0.1-R_LLM_r1b3)**2;

ER_OMS_r2b1 =(0.5-R_OMS_r2b1)**2;ER_LMS_r2b1 =(0.5-R_LMS_r2b1)**2;
ER_LAIC_r2b1 = (0.5-R_LAIC_r2b1)**2;ER_OLM_r2b1 =(0.5-R_OLM_r2b1)**2;
ER_LLM_r2b1 = (0.5-R_LLM_r2b1)**2;

ER_OMS_r2b2 =(0.5-R_OMS_r2b2)**2;ER_LMS_r2b2 =(0.5-R_LMS_r2b2)**2;
ER_LAIC_r2b2 = (0.5-R_LAIC_r2b2)**2;ER_OLM_r2b2 =(0.5-R_OLM_r2b2)**2;
ER_LLM_r2b2 = (0.5-R_LLM_r2b2)**2;

ER_OMS_r2b3 =(0.5-R_OMS_r2b3)**2;ER_LMS_r2b3 =(0.5-R_LMS_r2b3)**2;
ER_LAIC_r2b3 = (0.5-R_LAIC_r2b3)**2;ER_OLM_r2b3 =(0.5-R_OLM_r2b3)**2;
```

```
ER_LLM_r2b3 = (0.5-R_LLM_r2b3)**2;

ER_OMS_r3b1 =(0.9-R_OMS_r3b1)**2;ER_LMS_r3b1 =(0.9-R_LMS_r3b1)**2;
ER_LAIC_r3b1 = (0.9-R_LAIC_r3b1)**2;ER_OLM_r3b1=(0.9-R_OLM_r3b1)**2;
ER_LLM_r3b1 = (0.9-R_LLM_r3b1)**2;

ER_OMS_r3b2 =(0.9 - R_OMS_r3b2)**2;ER_LMS_r3b2=(0.9 - R_LMS_r3b2)**2;
ER_LAIC_r3b2 =(0.9 - R_LAIC_r3b2)**2;ER_OLM_r3b2=(0.9-R_OLM_r3b2)**2;
ER_LLM_r3b2 = (0.9 - R_LLM_r3b2)**2;

ER_OMS_r3b3 =(0.9 - R_OMS_r3b3)**2;ER_LMS_r3b3=(0.9 - R_LMS_r3b3)**2;
ER_LAIC_r3b3 =(0.9 - R_LAIC_r3b3)**2;ER_OLM_r3b3 =(0.9R_OLM_r3b3)**2;
ER_LLM_r3b3 = (0.9 - R_LLM_r3b3)**2;

ER_OMS_m11 =(0.9 - R_OMS_m11)**2;ER_LMS_m11 =(0.9 - R_LMS_m11)**2;
ER_LAIC_m11 = (0.9 - R_LAIC_m11)**2;ER_OLM_m11 =(0.9 - R_OLM_m11)**2;
ER_LLM_m11 = (0.9 - R_LLM_m11)**2;

ER_OMS_m12 =(0.9 - R_OMS_m12)**2;ER_LMS_m12 =(0.9 - R_LMS_m12)**2;
ER_LAIC_m12 = (0.9 - R_LAIC_m12)**2;ER_OLM_m12 =(0.9 - R_OLM_m12)**2;
ER_LLM_m12 = (0.9 - R_LLM_m12)**2;


ER_OMS_m13 =(0.9 - R_OMS_m13)**2;ER_LMS_m13 =(0.9 - R_LMS_m13)**2;
ER_LAIC_m13 = (0.9 - R_LAIC_m13)**2;ER_OLM_m13 =(0.9 - R_OLM_m13)**2;
ER_LLM_m13 = (0.9 - R_LLM_m13)**2;

ER_OMS_m21 =(0.9 - R_OMS_m21)**2;ER_LMS_m21 =(0.9 - R_LMS_m21)**2;
ER_LAIC_m21 = (0.9 - R_LAIC_m21)**2;ER_OLM_m21 =(0.9 - R_OLM_m21)**2;
ER_LLM_m21 = (0.9 - R_LLM_m21)**2;

ER_OMS_m22 =(0.9 - R_OMS_m22)**2;   ER_LMS_m22 =(0.9 - R_LMS_m22)**2;
ER_LAIC_m22 = (0.9 - R_LAIC_m22)**2;ER_OLM_m22 =(0.9 - R_OLM_m22)**2;
ER_LLM_m22 = (0.9 - R_LLM_m22)**2;

ER_OMS_m23 =(0.9 - R_OMS_m23)**2;   ER_LMS_m23 =(0.9 - R_LMS_m23)**2;
ER_LAIC_m23 = (0.9 - R_LAIC_m23)**2;ER_OLM_m23 =(0.9 - R_OLM_m23)**2;
ER_LLM_m23 = (0.9 - R_LLM_m23)**2;

ER_OMS_rel1 =(0.9 - R_OMS_rel1)**2;ER_LMS_rel1=(0.9 - R_LMS_rel1)**2;
ER_LAIC_rel1 =(0.9- R_LAIC_rel1)**2;ER_OLM_rel1=(0.9- R_OLM_rel1)**2;
ER_LLM_rel1 = (0.9 - R_LLM_rel1)**2;

ER_OMS_rel2 =(0.9 - R_OMS_rel2)**2;ER_LMS_rel2 =(0.9-R_LMS_rel2)**2;
ER_LAIC_rel2 =(0.9 -R_LAIC_rel2)**2;ER_OLM_rel2 =(0.9-R_OLM_rel2)**2;
ER_LLM_rel2 = (0.9 - R_LLM_rel2)**2;

ER_OMS_rel3 =(0.9 - R_OMS_rel3)**2;ER_LMS_rel3 =(0.9 -R_LMS_rel3)**2;
ER_LAIC_rel3 =(0.9 -R_LAIC_rel3)**2;ER_OLM_rel3 =(0.9-R_OLM_rel3)**2;
ER_LLM_rel3 = (0.9 - R_LLM_rel3)**2; run; *proc print data = MSE;

ER_OMS_rel4 =(0.9 - R_OMS_rel4)**2;ER_LMS_rel4 =(0.9 -R_LMS_rel4)**2;
ER_LAIC_rel4 = (0.9-R_LAIC_rel4)**2;ER_OLM_rel4 =(0.9-R_OLM_rel4)**2;
ER_LLM_rel4 = (0.9 - R_LLM_rel4)**2; run; *proc print data = MSE;

var  ER_OMS_r1b1  ER_LMS_r1b1  ER_LAIC_r1b1  ER_OLM_r1b1  ER_LLM_r1b1
     ER_OMS_r1b2  ER_LMS_r1b2  ER_LAIC_r1b2  ER_OLM_r1b2  ER_LLM_r1b2
     ER_OMS_r1b3  ER_LMS_r1b3  ER_LAIC_r1b3  ER_OLM_r1b3 ER_LLM_r1b3;
run;

proc print data = MSE;
var  ER_OMS_r2b1  ER_LMS_r2b1  ER_LAIC_r2b1  ER_OLM_r2b1  ER_LLM_r2b1
     ER_OMS_r2b2  ER_LMS_r2b2  ER_LAIC_r2b2  ER_OLM_r2b2  ER_LLM_r2b2
     ER_OMS_r2b3  ER_LMS_r2b3  ER_LAIC_r2b3  ER_OLM_r2b3 ER_LLM_r2b3;
run;
```

```
proc print data = MSE;
var  ER_OMS_r3b1  ER_LMS_r3b1  ER_LAIC_r3b1  ER_OLM_r3b1  ER_LLM_r3b1
     ER_OMS_r3b2  ER_LMS_r3b2  ER_LAIC_r3b2  ER_OLM_r3b2  ER_LLM_r3b2
     ER_OMS_r3b3  ER_LMS_r3b3  ER_LAIC_r3b3  ER_OLM_r3b3 ER_LLM_r3b3;
run;

proc print data = MSE;
var  ER_OMS_m11  ER_LMS_m11  ER_LAIC_m11  ER_OLM_m11  ER_LLM_m11
     ER_OMS_m12  ER_LMS_m12  ER_LAIC_m12  ER_OLM_m12  ER_LLM_m12
     ER_OMS_m13  ER_LMS_m13  ER_LAIC_m13  ER_OLM_m13  ER_LLM_m13;
run;

proc print data = MSE;
var  ER_OMS_m21  ER_LMS_m21  ER_LAIC_m21  ER_OLM_m21  ER_LLM_m21
     ER_OMS_m22  ER_LMS_m22  ER_LAIC_m22  ER_OLM_m22  ER_LLM_m22
     ER_OMS_m23  ER_LMS_m23  ER_LAIC_m23  ER_OLM_m23  ER_LLM_m23;
run;

proc print data = MSE;
var  ER_OMS_rel1  ER_LMS_rel1  ER_LAIC_rel1  ER_OLM_rel1  ER_LLM_rel1
     ER_OMS_rel1  ER_LMS_rel1  ER_LAIC_rel1  ER_OLM_rel1  ER_LLM_rel1
     ER_OMS_rel1  ER_LMS_rel1  ER_LAIC_rel1  ER_OLM_rel1 ER_LLM_rel1;
run;

proc print data = MSE;
var  ER_OMS_rel2  ER_LMS_rel2  ER_LAIC_rel2  ER_OLM_rel2  ER_LLM_rel2
     ER_OMS_rel2  ER_LMS_rel2  ER_LAIC_rel2  ER_OLM_rel2  ER_LLM_rel2
     ER_OMS_rel2  ER_LMS_rel2  ER_LAIC_rel2  ER_OLM_rel2 ER_LLM_rel2;
run;

proc print data = MSE;
var  ER_OMS_rel3  ER_LMS_rel3  ER_LAIC_rel3  ER_OLM_rel3  ER_LLM_rel3
     ER_OMS_rel3  ER_LMS_rel3  ER_LAIC_rel3  ER_OLM_rel3  ER_LLM_rel3
     ER_OMS_rel3  ER_LMS_rel3  ER_LAIC_rel3  ER_OLM_rel3 ER_LLM_rel3;
run;
```

โปรแกรมแสดงการสรุปผลจาก 1,000 ครั้งของตัวแบบการถดถอยโลจิสติก

```
data summary;  set summary MSE;

if ER_OMS_r1b1<0 then delete;       if ER_LMS_r1b1<0 then delete;
if ER_LAIC_r1b1<0 then delete;      if ER_OLM_r1b1<0 then delete;
if ER_LLM_r1b1<0 then delete;       if ER_OMS_r1b2<0 then delete;
if ER_LMS_r1b2<0 then delete;       if ER_LAIC_r1b2<0 then delete;
if ER_OLM_r1b2<0 then delete;       if ER_LLM_r1b2<0 then delete;
if ER_OMS_r1b3<0 then delete;       if ER_LMS_r1b3<0 then delete;
if ER_LAIC_r1b3<0 then delete;      if ER_OLM_r1b3<0 then delete;
if ER_LLM_r1b3<0 then delete;       if ER_OMS_r2b1<0 then delete;
if ER_LMS_r2b1<0 then delete;       if ER_LAIC_r2b1<0 then delete;
if ER_OLM_r2b1<0 then delete;       if ER_LLM_r2b1<0 then delete;
if ER_OMS_r2b2<0 then delete;       if ER_LMS_r2b2<0 then delete;
if ER_LAIC_r2b2<0 then delete;      if ER_OLM_r2b2<0 then delete;
if ER_LLM_r2b2<0 then delete;       if ER_OMS_r2b3<0 then delete;
if ER_LMS_r2b3<0 then delete;       if ER_LAIC_r2b3<0 then delete;
if ER_OLM_r2b3<0 then delete;       if ER_LLM_r2b3<0 then delete;
if ER_OMS_r3b1<0 then delete;       if ER_LMS_r3b1<0 then delete;
if ER_LAIC_r3b1<0 then delete;      if ER_OLM_r3b1<0 then delete;
if ER_LLM_r3b1<0 then delete;       if ER_OMS_r3b2<0 then delete;
if ER_LMS_r3b2<0 then delete;       if ER_LAIC_r3b2<0 then delete;
if ER_OLM_r3b2<0 then delete;       if ER_LLM_r3b2<0 then delete;
if ER_OMS_r3b3<0 then delete;       if ER_LMS_r3b3<0 then delete;
```

```
if ER_LAIC_r3b3<0 then delete;      if ER_OLM_r3b3<0 then delete;
if ER_LLM_r3b3<0 then delete;       if ER_OMS_m11<0 then delete;
if ER_LMS_m11<0 then delete;        if ER_LAIC_m11<0 then delete;
if ER_OLM_m11<0 then delete;        if ER_LLM_m11<0 then delete;
if ER_OMS_m12<0 then delete;        if ER_LMS_m12<0 then delete;
if ER_LAIC_m12<0 then delete;       if ER_OLM_m12<0 then delete;
if ER_LLM_m12<0 then delete;        if ER_OMS_m13<0 then delete;
if ER_LMS_m13<0 then delete;        if ER_LAIC_m13<0 then delete;
if ER_OLM_m13<0 then delete;        if ER_LLM_m13<0 then delete;
if ER_OMS_m21<0 then delete;        if ER_LMS_m21<0 then delete;
if ER_LAIC_m21<0 then delete;       if ER_OLM_m21<0 then delete;
if ER_LLM_m21<0 then delete;        if ER_OMS_m22<0 then delete;
if ER_LMS_m22<0 then delete;        if ER_LAIC_m22<0 then delete;
if ER_OLM_m22<0 then delete;        if ER_LLM_m22<0 then delete;
if ER_OMS_m23<0 then delete;        if ER_LMS_m23<0 then delete;
if ER_LAIC_m23<0 then delete;       if ER_OLM_m23<0 then delete;
if ER_LLM_m23<0 then delete;        if ER_OMS_rel1<0 then delete;
if ER_LMS_rel1<0 then delete;       if ER_LAIC_rel1<0 then delete;
if ER_OLM_rel1<0 then delete;       if ER_LLM_rel1<0 then delete;
if ER_OMS_rel2<0 then delete;       if ER_LMS_rel2<0 then delete;
if ER_LAIC_rel2<0 then delete;      if ER_OLM_rel2<0 then delete;
if ER_LLM_rel2<0 then delete;       if ER_OMS_rel3<0 then delete;
if ER_LMS_rel3<0 then delete;       if ER_LAIC_rel3<0 then delete;
if ER_OLM_rel3<0 then delete;       if ER_LLM_rel3<0 then delete;
if ER_OMS_rel4<0 then delete;       if ER_LMS_rel4<0 then delete;
if ER_LAIC_rel4<0 then delete;      if ER_OLM_rel4<0 then delete;
if ER_LLM_rel4<0 then delete;
run;

data summary;
set summary R_r1b1 R_r1b2 R_r1b3 R_r2b1 R_r2b2 R_r2b3 R_r3b1 R_r3b2
R_r3b3 R_m11 R_m12 R_m13 R_m21 R_m22 R_m23 R_rel1 R_rel2 R_rel3;
if 0.1 = . then delete;if 0.5 = . then delete;
if 0.9 = . then delete;i=_N_;  if i>&nsim then delete;  run;

var R_OMS_r1b1 R_LMS_r1b1 R_LAIC_r1b1 R_OLM_r1b1 R_LLM_r1b1 BASE_r1b1
R_OMS_r1b2 R_LMS_r1b2 R_LAIC_r1b2 R_OLM_r1b2 R_LLM_r1b2 BASE_r1b2
R_OMS_r1b3 R_LMS_r1b3 R_LAIC_r1b3 R_OLM_r1b3 R_LLM_r1b3 BASE_r1b3;
run;

proc print data = summary;
var R_OMS_r2b1 R_LMS_r2b1 R_LAIC_r2b1 R_OLM_r2b1 R_LLM_r2b1 BASE_r2b1
R_OMS_r2b2 R_LMS_r2b2 R_LAIC_r2b2 R_OLM_r2b2 R_LLM_r2b2 BASE_r2b2
R_OMS_r2b3 R_LMS_r2b3 R_LAIC_r2b3 R_OLM_r2b3 R_LLM_r2b3 BASE_r2b3;
run;

proc print data = summary;
var R_OMS_r3b1 R_LMS_r3b1 R_LAIC_r3b1 R_OLM_r3b1 R_LLM_r3b1 BASE_r3b1
R_OMS_r3b2 R_LMS_r3b2 R_LAIC_r3b2 R_OLM_r3b2 R_LLM_r3b2 BASE_r3b2
R_OMS_r3b3 R_LMS_r3b3 R_LAIC_r3b3 R_OLM_r3b3 R_LLM_r3b3 BASE_r3b3;
run;

proc print data = summary;
var R_OMS_m11 R_LMS_m11 R_LAIC_m11 R_OLM_m11 R_LLM_m11 BASE_m11
R_OMS_m12 R_LMS_m12 R_LAIC_m12 R_OLM_m12 R_LLM_m12 BASE_m12
R_OMS_m13 R_LMS_m13 R_LAIC_m13 R_OLM_m13 R_LLM_m13 BASE_m13;
run;

proc print data = summary;
var R_OMS_m21 R_LMS_m21 R_LAIC_m21 R_OLM_m21 R_LLM_m21 BASE_m21
R_OMS_m22 R_LMS_m22 R_LAIC_m22 R_OLM_m22 R_LLM_m22 BASE_m22
```

```
R_OMS_m23 R_LMS_m23 R_LAIC_m23 R_OLM_m23 R_LLM_m23 BASE_m23;
run;

proc print data = summary;
var R_OMS_rel1 R_LMS_rel1 R_LAIC_rel1 R_OLM_rel1 R_LLM_rel1 BASE_rel1
R_OMS_rel2 R_LMS_rel2 R_LAIC_rel2 R_OLM_rel2 R_LLM_rel2 BASE_rel2
R_OMS_rel3 R_LMS_rel3 R_LAIC_rel3 R_OLM_rel3 R_LLM_rel3 BASE_rel3;
run;

%let loop_num=%eval(&loop_num+1);
%end;

proc freq data = summary noprint;run;
ods OUTPUT ;
proc Means data = summary noprint;
      title 'Descriptive Statistics';
      var R_OMS_r1b1 R_LMS_r1b1 R_LAIC_r1b1 R_OLM_r1b1 R_LLM_r1b1
            R_OMS_r1b2 R_LMS_r1b2 R_LAIC_r1b2 R_OLM_r1b2 R_LLM_r1b2
            R_OMS_r1b3 R_LMS_r1b3 R_LAIC_r1b3 R_OLM_r1b3 R_LLM_r1b3
            R_OMS_r2b1 R_LMS_r2b1 R_LAIC_r2b1 R_OLM_r2b1 R_LLM_r2b1
            R_OMS_r2b2 R_LMS_r2b2 R_LAIC_r2b2 R_OLM_r2b2 R_LLM_r2b2
            R_OMS_r2b3 R_LMS_r2b3 R_LAIC_r2b3 R_OLM_r2b3 R_LLM_r2b3
            R_OMS_r3b1 R_LMS_r3b1 R_LAIC_r3b1 R_OLM_r3b1 R_LLM_r3b1
            R_OMS_r3b2 R_LMS_r3b2 R_LAIC_r3b2 R_OLM_r3b2 R_LLM_r3b2
            R_OMS_r3b3 R_LMS_r3b3 R_LAIC_r3b3 R_OLM_r3b3 R_LLM_r3b3
            R_OMS_m11 R_LMS_m11 R_LAIC_m11 R_OLM_m11 R_LLM_m11
            R_OMS_m12 R_LMS_m12 R_LAIC_m12 R_OLM_m12 R_LLM_m12
            R_OMS_m13 R_LMS_m13 R_LAIC_m13 R_OLM_m13 R_LLM_m13
            R_OMS_m21 R_LMS_m21 R_LAIC_m21 R_OLM_m21 R_LLM_m21
            R_OMS_m22 R_LMS_m22 R_LAIC_m22 R_OLM_m22 R_LLM_m22
            R_OMS_m23 R_LMS_m23 R_LAIC_m23 R_OLM_m23 R_LLM_m23
            R_OMS_rel1 R_LMS_rel1 R_LAIC_rel1 R_OLM_rel1 R_LLM_rel1
            R_OMS_rel2 R_LMS_rel2 R_LAIC_rel2 R_OLM_rel2 R_LLM_rel2
            R_OMS_rel3 R_LMS_rel3 R_LAIC_rel3 R_OLM_rel3 R_LLM_rel3
            R_OMS_rel4 R_LMS_rel4 R_LAIC_rel4 R_OLM_rel4 R_LLM_rel4
      ER_OMS_r1b1 ER_LMS_r1b1 ER_LAIC_r1b1 ER_OLM_r1b1 ER_LLM_r1b1
      ER_OMS_r1b2 ER_LMS_r1b2 ER_LAIC_r1b2 ER_OLM_r1b2 ER_LLM_r1b2
      ER_OMS_r1b3 ER_LMS_r1b3 ER_LAIC_r1b3 ER_OLM_r1b3 ER_LLM_r1b3
      ER_OMS_r2b1 ER_LMS_r2b1 ER_LAIC_r2b1 ER_OLM_r2b1 ER_LLM_r2b1
      ER_OMS_r2b2 ER_LMS_r2b2 ER_LAIC_r2b2 ER_OLM_r2b2 ER_LLM_r2b2
      ER_OMS_r2b3 ER_LMS_r2b3 ER_LAIC_r2b3 ER_OLM_r2b3 ER_LLM_r2b3
      ER_OMS_r3b1 ER_LMS_r3b1 ER_LAIC_r3b1 ER_OLM_r3b1 ER_LLM_r3b1
      ER_OMS_r3b2 ER_LMS_r3b2 ER_LAIC_r3b2 ER_OLM_r3b2 ER_LLM_r3b2
      ER_OMS_r3b3 ER_LMS_r3b3 ER_LAIC_r3b3 ER_OLM_r3b3 ER_LLM_r3b3
      ER_OMS_m11 ER_LMS_m11 ER_LAIC_m11 ER_OLM_m11 ER_LLM_m11
      ER_OMS_m12 ER_LMS_m12 ER_LAIC_m12 ER_OLM_m12 ER_LLM_m12
      ER_OMS_m13 ER_LMS_m13 ER_LAIC_m13 ER_OLM_m13 ER_LLM_m13
      ER_OMS_m21 ER_LMS_m21 ER_LAIC_m21 ER_OLM_m21 ER_LLM_m21
      ER_OMS_m22 ER_LMS_m22 ER_LAIC_m22 ER_OLM_m22 ER_LLM_m22
      ER_OMS_m23 ER_LMS_m23 ER_LAIC_m23 ER_OLM_m23 ER_LLM_m23
      ER_OMS_rel1 ER_LMS_rel1 ER_LAIC_rel1 ER_OLM_rel1 ER_LLM_rel1
      ER_OMS_rel2 ER_LMS_rel2 ER_LAIC_rel2 ER_OLM_rel2 ER_LLM_rel2
      ER_OMS_rel3 ER_LMS_rel3 ER_LAIC_rel3 ER_OLM_rel3 ER_LLM_rel3
      ER_OMS_rel4 ER_LMS_rel4 ER_LAIC_rel4 ER_OLM_rel4 ER_LLM_rel4;
OUTPUT OUT = MR   MEAN =
      MR_OMS_r1b1 MR_LMS_r1b1 MR_LAIC_r1b1 MR_OLM_r1b1 MR_LLM_r1b1
      MR_OMS_r1b2 MR_LMS_r1b2 MR_LAIC_r1b2 MR_OLM_r1b2 MR_LLM_r1b2
      MR_OMS_r1b3 MR_LMS_r1b3 MR_LAIC_r1b3 MR_OLM_r1b3 MR_LLM_r1b3
```

```
      MR_OMS_r2b1 MR_LMS_r2b1 MR_LAIC_r2b1 MR_OLM_r2b1 MR_LLM_r2b1
      MR_OMS_r2b2 MR_LMS_r2b2 MR_LAIC_r2b2 MR_OLM_r2b2 MR_LLM_r2b2
      MR_OMS_r2b3 MR_LMS_r2b3 MR_LAIC_r2b3 MR_OLM_r2b3 MR_LLM_r2b3
      MR_OMS_r3b1 MR_LMS_r3b1 MR_LAIC_r3b1 MR_OLM_r3b1 MR_LLM_r3b1
      MR_OMS_r3b2 MR_LMS_r3b2 MR_LAIC_r3b2 MR_OLM_r3b2 MR_LLM_r3b2
      MR_OMS_r3b3 MR_LMS_r3b3 MR_LAIC_r3b3 MR_OLM_r3b3 MR_LLM_r3b3
      MR_OMS_m11 MR_LMS_m11 MR_LAIC_m11 MR_OLM_m11 MR_LLM_m11
      MR_OMS_m12 MR_LMS_m12 MR_LAIC_m12 MR_OLM_m12 MR_LLM_m12
      MR_OMS_m13 MR_LMS_m13 MR_LAIC_m13 MR_OLM_m13 MR_LLM_m13
      MR_OMS_m21 MR_LMS_m21 MR_LAIC_m21 MR_OLM_m21 MR_LLM_m21
      MR_OMS_m22 MR_LMS_m22 MR_LAIC_m22 MR_OLM_m22 MR_LLM_m22
      MR_OMS_m23 MR_LMS_m23 MR_LAIC_m23 MR_OLM_m23 MR_LLM_m23
      MR_OMS_rel1 MR_LMS_rel1 MR_LAIC_rel1 MR_OLM_rel1 MR_LLM_rel1
      MR_OMS_rel2 MR_LMS_rel2 MR_LAIC_rel2 MR_OLM_rel2 MR_LLM_rel2
      MR_OMS_rel3 MR_LMS_rel3 MR_LAIC_rel3 MR_OLM_rel3 MR_LLM_rel3
      MR_OMS_rel4 MR_LMS_rel4 MR_LAIC_rel4 MR_OLM_rel4 MR_LLM_rel4
MER_OMS_r1b1 MER_LMS_r1b1 MER_LAIC_r1b1 MER_OLM_r1b1 MER_LLM_r1b1
MER_OMS_r1b2 MER_LMS_r1b2 MER_LAIC_r1b2 MER_OLM_r1b2 MER_LLM_r1b2
MER_OMS_r1b3 MER_LMS_r1b3 MER_LAIC_r1b3 MER_OLM_r1b3 MER_LLM_r1b3
MER_OMS_r2b1 MER_LMS_r2b1 MER_LAIC_r2b1 MER_OLM_r2b1 MER_LLM_r2b1
MER_OMS_r2b2 MER_LMS_r2b2 MER_LAIC_r2b2 MER_OLM_r2b2 MER_LLM_r2b2
MER_OMS_r2b3 MER_LMS_r2b3 MER_LAIC_r2b3 MER_OLM_r2b3 MER_LLM_r2b3
MER_OMS_r3b1 MER_LMS_r3b1 MER_LAIC_r3b1 MER_OLM_r3b1 MER_LLM_r3b1
MER_OMS_r3b2 MER_LMS_r3b2 MER_LAIC_r3b2 MER_OLM_r3b2 MER_LLM_r3b2
MER_OMS_r3b3 MER_LMS_r3b3 MER_LAIC_r3b3 MER_OLM_r3b3 MER_LLM_r3b3
MER_OMS_m11 MER_LMS_m11 MER_LAIC_m11 MER_OLM_m11 MER_LLM_m11
MER_OMS_m12 MER_LMS_m12 MER_LAIC_m12 MER_OLM_m12 MER_LLM_m12
MER_OMS_m13 MER_LMS_m13 MER_LAIC_m13 MER_OLM_m13 MER_LLM_m13
MER_OMS_m21 MER_LMS_m21 MER_LAIC_m21 MER_OLM_m21 MER_LLM_m21
MER_OMS_m22 MER_LMS_m22 MER_LAIC_m22 MER_OLM_m22 MER_LLM_m22
MER_OMS_m23 MER_LMS_m23 MER_LAIC_m23 MER_OLM_m23 MER_LLM_m23
MER_OMS_rel1 MER_LMS_rel1 MER_LAIC_rel1 MER_OLM_rel1 MER_LLM_rel1
MER_OMS_rel2 MER_LMS_rel2 MER_LAIC_rel2 MER_OLM_rel2 MER_LLM_rel2
MER_OMS_rel3 MER_LMS_rel3 MER_LAIC_rel3 MER_OLM_rel3 MER_LLM_rel3
MER_OMS_rel4 MER_LMS_rel4 MER_LAIC_rel4 MER_OLM_rel4 MER_LLM_rel4;
run ;

proc print data=MR;
var MR_OMS_r1b1 MR_LMS_r1b1 MR_LAIC_r1b1 MR_OLM_r1b1 MR_LLM_r1b1
MER_OMS_r1b1 MER_LMS_r1b1 MER_LAIC_r1b1 MER_OLM_r1b1 MER_LLM_r1b1;
run;

proc print data=MR;
var MR_OMS_r1b2 MR_LMS_r1b2 MR_LAIC_r1b2 MR_OLM_r1b2 MR_LLM_r1b2
MER_OMS_r1b2 MER_LMS_r1b2 MER_LAIC_r1b2 MER_OLM_r1b2 MER_LLM_r1b2;
run;

proc print data=MR;
var MR_OMS_r1b3 MR_LMS_r1b3 MR_LAIC_r1b3 MR_OLM_r1b3 MR_LLM_r1b3
MER_OMS_r1b3 MER_LMS_r1b3 MER_LAIC_r1b3 MER_OLM_r1b3 MER_LLM_r1b3;
run;

proc print data=MR;
var MR_OMS_r2b1 MR_LMS_r2b1 MR_LAIC_r2b1 MR_OLM_r2b1 MR_LLM_r2b1
MER_OMS_r2b1 MER_LMS_r2b1 MER_LAIC_r2b1 MER_OLM_r2b1 MER_LLM_r2b1;
run;

proc print data=MR;
var MR_OMS_r2b2 MR_LMS_r2b2 MR_LAIC_r2b2 MR_OLM_r2b2 MR_LLM_r2b2
MER_OMS_r2b2 MER_LMS_r2b2 MER_LAIC_r2b2 MER_OLM_r2b2 MER_LLM_r2b2;
```

```
run;

proc print data=MR;
var MR_OMS_r2b3 MR_LMS_r2b3 MR_LAIC_r2b3 MR_OLM_r2b3 MR_LLM_r2b3
MER_OMS_r2b3 MER_LMS_r2b3 MER_LAIC_r2b3 MER_OLM_r2b3 MER_LLM_r2b3;
run;

proc print data=MR;
var MR_OMS_r3b1 MR_LMS_r3b1 MR_LAIC_r3b1 MR_OLM_r3b1 MR_LLM_r3b1
MER_OMS_r3b1 MER_LMS_r3b1 MER_LAIC_r3b1 MER_OLM_r3b1 MER_LLM_r3b1;
run;

proc print data=MR;
var MR_OMS_r3b2 MR_LMS_r3b2 MR_LAIC_r3b2 MR_OLM_r3b2 MR_LLM_r3b2
MER_OMS_r3b2 MER_LMS_r3b2 MER_LAIC_r3b2 MER_OLM_r3b2 MER_LLM_r3b2;
run;

proc print data=MR;
var MR_OMS_r3b3 MR_LMS_r3b3 MR_LAIC_r3b3 MR_OLM_r3b3 MR_LLM_r3b3
MER_OMS_r3b3 MER_LMS_r3b3 MER_LAIC_r3b3 MER_OLM_r3b3 MER_LLM_r3b3;
run;

proc print data=MR;
var MR_OMS_m11 MR_LMS_m11 MR_LAIC_m11 MR_OLM_m11 MR_LLM_m11
MER_OMS_m11 MER_LMS_m11 MER_LAIC_m11 MER_OLM_m11 MER_LLM_m11;
run;

proc print data=MR;
var MR_OMS_m12 MR_LMS_m12 MR_LAIC_m12 MR_OLM_m12 MR_LLM_m12
MER_OMS_m12 MER_LMS_m12 MER_LAIC_m12 MER_OLM_m12 MER_LLM_m12;
run;

proc print data=MR;
var MR_OMS_m13 MR_LMS_m13 MR_LAIC_m13 MR_OLM_m13 MR_LLM_m13
MER_OMS_m13 MER_LMS_m13 MER_LAIC_m13 MER_OLM_m13 MER_LLM_m13;
run;

proc print data=MR;
var MR_OMS_m21 MR_LMS_m21 MR_LAIC_m21 MR_OLM_m21 MR_LLM_m21
MER_OMS_m21 MER_LMS_m21 MER_LAIC_m21 MER_OLM_m21 MER_LLM_m21;
run;

proc print data=MR;
var MR_OMS_m22 MR_LMS_m22 MR_LAIC_m22 MR_OLM_m22 MR_LLM_m22
MER_OMS_m22 MER_LMS_m22 MER_LAIC_m22 MER_OLM_m22 MER_LLM_m22;
run;

proc print data=MR;
var MR_OMS_m23 MR_LMS_m23 MR_LAIC_m23 MR_OLM_m23 MR_LLM_m23
MER_OMS_m23 MER_LMS_m23 MER_LAIC_m23 MER_OLM_m23 MER_LLM_m23;
run;

proc print data=MR;
var MR_OMS_rel1 MR_LMS_rel1 MR_LAIC_rel1 MR_OLM_rel1 MR_LLM_rel1
MER_OMS_rel1 MER_LMS_rel1 MER_LAIC_rel1 MER_OLM_rel1 MER_LLM_rel1;
run;

proc print data=MR;
var MR_OMS_rel2 MR_LMS_rel2 MR_LAIC_rel2 MR_OLM_rel2 MR_LLM_rel2
MER_OMS_rel2 MER_LMS_rel2 MER_LAIC_rel2 MER_OLM_rel2 MER_LLM_rel2;
run;

proc print data=MR;
var MR_OMS_rel3 MR_LMS_rel3 MR_LAIC_rel3 MR_OLM_rel3 MR_LLM_rel3
```

```
MER_OMS_rel3 MER_LMS_rel3 MER_LAIC_rel3 MER_OLM_rel3 MER_LLM_rel3;
run;

proc print data=MR;
var MR_OMS_rel4 MR_LMS_rel4 MR_LAIC_rel4 MR_OLM_rel4 MR_LLM_rel4
MER_OMS_rel4 MER_LMS_rel4 MER_LAIC_rel4 MER_OLM_rel4 MER_LLM_rel4;
run;

%mend case_x1;
%mend case_x5;
%mend case_x10;
```

เรียกใช้มาร์กโคร (macro) ตามขนาดตัวอย่างต่างๆ ได้ดังนี้

กรณีตัวแบบมีจำนวนตัวแปรอธิบาย 1 ตัว

```
%case_x1(n_sample = 50,nsim = 1000,k = 1);
%case_x1(n_sample = 250,nsim = 1000,k = 1);
%case_x1(n_sample = 500,nsim = 1000,k = 1);
%case_x1(n_sample = 1000,nsim = 1000,k = 1);
```

กรณีตัวแบบมีจำนวนตัวแปรอธิบาย 5 ตัว

```
%case_x5(n_sample = 50,nsim = 1000,k = 5);
%case_x5(n_sample = 250,nsim = 1000,k = 5);
%case_x5(n_sample = 500,nsim = 1000,k = 5);
%case_x5(n_sample = 1000,nsim = 1000,k = 5);
```

กรณีตัวแบบมีจำนวนตัวแปรอธิบาย 10 ตัว

```
%case_x10(n_sample = 50,nsim = 1000,k = 10);
%case_x10(n_sample = 250,nsim = 1000,k = 10);
%case_x10(n_sample = 500,nsim = 1000,k = 10);
%case_x10(n_sample = 1000,nsim = 1000,k = 10);
```