รายการอ้างอิง

ภาษาไทย

กาญจนา พานิชการ.การประมาณค่าพารามิเตอร์ในสมการถดถอยโลจิสติคด้วยวิธีภาวะน่าจะเป็นสูงสุดและฟังก์ชันจำแนกประเภท. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2539.

กัลยา วานิชย์บัญชา. การวิเคราะห์ตัวแปรหลายตัวด้วยเอสพีเอสเอสสำหรับวินโดว์ส.พิมพ์ครั้งที่2. กรุงเทพมหานคร : สำนักพิมพ์จุฬาลงกรณ์มหาวิทยาลัย , 2544.

ธีระพร วีระถาวร. การอนุมานเชิงสถิติชั้นกลาง : โครงสร้างและความหมาย. พิมพ์ครั้งที่ 2. กรุงเทพมหานคร : สำนักพิมพ์จุฬาลงกรณ์มหาวิทยาลัย , 2536.

ภาษาอังกฤษ

Cosslett, Stephen R.Maximum Likelihood Estimator for Choice-Based Samples. Econometrica 49 (1981) : 1289-1316.

Greene , William H . Ecoconometric Analysis , 2nd ed. New York : Macmillan,1993.

Homer ,David W. and Lameshow ,S. Applied Logistic Regression . New York : Wiley ,2000.

King , Gary , and Langche Zeng . Logistic Regression in Rare Events Data . The Society for Political Methodology (2001) : 137-163 .

Manski , Charles F. , and Steven R . Lerman . The Estimation of Choice Probabilities from Choice Based Samples. Ecometrica 45 (1977) :1977-1988 .

Marvyn J. Silvapulle. On the Existence of Maximum Likelihood Estimators for the Binomial Response Models. Journal of the Royal statistical Society Ser.B 43 (1981): 310-313.

Norman R. Draper,Harry Smith. Applied regression analysis : Generalized Linear Models (GLIM). 3rd ed. John Wiley and sons , 1998.

Scott , A.J. , and C.J.Wild Fitting Logistic Models Under Case-Control or Choice Based Sampling . Journal of the Royal Statistical Society Ser B 48 (1986):170-182.

W.R. Gilks, S. Richardson and D. J. Spiegelhalter. Makov Chain Monte Carlo in pratice. London : Chapman and Hall, 1996.

Xie, Yu, and Charles F. Manski. The Logit Model and Response-Based Samples. Sociological Methods and Research 17(1989):283-302.

ภาคผนวก

ตารางแสดงลักษณะการทำงานของฟังก์ชันในโปรแกรม S-plus 2000 ทั้งหมดในการวิจัย

ลำดับที่	ชื่อฟังก์ชัน	การทำงานของฟังก์ชัน
1	rmvnorm (m,mean,cov,rho)	สร้างเลขสุ่มหลายตัวแปรให้มีการแจกแจงปกติที่มีค่าเฉลี่ยด้วย mean และกำหนดระดับความสัมพันธ์ระหว่าง 2 ตัวแปรอิสระเท่ากับ rho
2	rbinom(m,ni,pi)	สร้างตัวเลขสุ่มให้มีการแจกแจงทวินาม เมื่อ m แทน จำนวนกลุ่ม ni แทน จำนวนครั้งของการทดลอง และ pi แทน ความน่าจะเป็นของเหตุการณ์ที่สนใจ
3	mean()	คำนวณหาค่าเฉลี่ยของข้อมูล
4	ginverse()	คำนวณหาค่า inverse ของเมทริกซ์
5	exp()	คำนวณหาค่าเอกซ์โพแนนเชียล
6	matrix(nrow,ncol)	ทำการเก็บข้อมูลในรูปของเมทริกซ์ โดย nrow แทน จำนวนแถวที่ต้องการ และ ncol แทน จำนวนสดมภ์ที่ต้องการ
7	cbind(x)	เก็บข้อมูลในรูปแบบของสดมภ์
8	list()	สร้างหรือเก็บค่าของข้อมูลมีคุณสมบัติเป็นลิซท์
9	cat("/n")	เว้น 1 บรรทัด
10	mahalanobis(beta1,newbeta, cov.beta)	คำนวณค่า Mahalanobis โดยที่ beta1 แทน ค่าเริ่มต้นของ beta ที่ใช้ในการประมาณพารามิเตอร์ newbeta แทนค่า beta ที่ได้จากการประมาณค่า และ cov.beta แทนค่า เมตริกซ์ความแปรปรวนร่วมของbeta ที่ได้จากการประมาณ
11	sqrt(ma)	คำนวณค่า Mahalanobis Distance

ลำดับที่	ชื่อฟังก์ชัน	การทำงานของฟังก์ชัน
12	lsfit(x,emp.logit,intercept=F)$coef	คำนวณค่าประมาณพารามิเตอร์ด้วยวิธี Linear least squres โดยที่ x แทน เมทริกซ์ของตัวแปรอิสระที่สดมภ์แรกเป็น 1 หมด และ emp.logit แทน ค่าตัวแปรตามที่ได้จาก empirical logits
13	ml.logit(x,y,m,n,beta.start, maxits,eps)	คำนวณค่าด้วยวิธี Newton – Raphson ของวิธีความควรจะเป็นสูงสุดโดยที่ x แทน เมทริกซ์ของตัวแปรอิสระ y แทน เวกเตอร์ของตัวแปรตาม m แทน จำนวนกลุ่ม n แทน ค่าของ ni beta.start แทน ค่าเริ่มต้น B_0 ที่ใช้ในการประมาณพารามิเตอร์ด้วยวิธี Newton – Raphson , maxits แทน จำนวนรอบมากสุดที่ยอมให้ค่าลู่เข้า eps แทนค่าผลต่างที่ยอมรับได้ $\lvert B_r - B_{r+1}\rvert < eps$
14	we.logit(x,y,m,n,beta.start, w1,w0,maxits,eps)	คำนวณค่าด้วยวิธี Newton – Raphson ของวิธีถ่วงน้ำหนักโดยที่ x แทน เมทริกซ์ของตัวแปรอิสระ y แทน เวกเตอร์ของตัวแปรตาม m แทน จำนวนกลุ่ม n แทน ค่าของ ni beta.start แทน ค่าเริ่มต้น B_0 ที่ใช้ในการประมาณพารามิเตอร์ด้วยวิธี Newton – Raphson w1 และ w0แทน สัดส่วนการถ่วงน้ำหนัก maxits แทน จำนวนรอบมากสุดที่ยอมให้ค่าลู่เข้า eps แทน ค่าผลต่างที่ยอมรับได้ $\lvert B_r - B_{r+1}\rvert < eps$

ลำดับที่	ชื่อฟังก์ชัน	การทำงานของฟังก์ชัน
15	pc.logit_function(x,m,n,pi.M, pi.Mhat,beta.start,maxits,eps)	คำนวณค่าด้วยวิธี Newton – Raphson ของวิธีการปรับแก้เบื้องต้นโดยที่ x แทน เมทริกซ์ของตัวแปรอิสระ m แทน จำนวนกลุ่ม n แทน ค่าของ ni , pi.M และ pi.Mhat แทน ค่าเฉลี่ยความน่าจะเป็นของเหตุการณ์ที่สนใจของประชากรและตัวอย่างตามลำดับ , beta.start แทน ค่าเริ่มต้น B_0 ที่ใช้ในการประมาณพารามิเตอร์ด้วยวิธี Newton – Raphson , maxits แทน จำนวนรอบมากสุดที่ยอมให้ค่าลู่เข้า eps แทนค่าผลต่างที่ยอมรับได้ $\lvert B_r - B_{r+1} \rvert < eps$

โปรแกรมสำหรับการดำเนินการวิจัย

ตัวอย่างโปรแกรม S-plus 2000 สำหรับการประมาณค่าพารามิเตอร์ของตัวแบบถดถอยโลจิสติค

```
pi.M_c(0.1,0.3,0.5,0.8)
one_rep(1,m)
beta.start_rep((1,inde+1))

/* ฟังก์ชันการสร้างตัวแปรอิสระ */
x_rmvnorm(m,mean=c(rep(0,inde)),cov=diag(1,inde),rho=ro)
xmat_cbind(one,x)



/* วิธีความควรจะเป็นสูงสุด */

ml.logit_function(x,y,m,n,beta.start,maxits=20,eps=0.0000001)
{   if(missing(beta.start))
    {   emp.logit_log((y+0.5)/(n-y+0.5))
        newbeta_lsfit(x,emp.logit,intercept=F)$coef
    }    else {newbeta_beta.start}
    iter_0
     converged_F
    while((!converged)&(iter<maxits))
    {
            iter_iter+1
            beta_newbeta
            tmp_exp(-(x%*%beta))
            pi_1/(1+tmp)
            mu_n*pi
            w_as.vector(n*pi*(1-pi))
```

```
            xtwx_t(w*x)%*%x
            xtwxinv_ginverse(xtwx)
            newbeta_beta+xtwxinv%*%t(x)%*%(y-mu)
            converged_all(abs(newbeta-beta)<eps)
    }
        cat("\n")
        tmp_exp(x%*%newbeta)
        pi_tmp/(1+tmp)
        cov.ml_xtwxinv
        beta1_matrix(1,ncol=(inde+1),byrow=T)
        ma_mahalanobis(beta1,newbeta,cov.ml)
        md_sqrt(ma)
        result_list(beta.ml=newbeta,md=md)
        result
}


/* วิธีการถ่วงน้ำหนัก */


we.logit_function(x,y,m,n,beta.start,w1,w0,maxits=20,eps=0.0000001)
{   if(missing(beta.start))
    {   emp.logit_log((y+0.5)/(n-y+0.5))
         newbeta_lsfit(x,emp.logit,intercept=F)$coef
    } else {newbeta_beta.start}
    iter_0
     converged_F
    while((!converged)&(iter<maxits))
    {
            iter_iter+1
            beta_newbeta
            tmp_exp(-(x%*%beta))
```

```
            pi_1/(1+tmp)
            a_((w1*y*(1-pi))-(w0*(n-y)*pi))
            ub_t(x)%*%a
            b_as.vector(pi*(1-pi)*((w1*y)+w0*(n-y)))
            hb_t(x)%*%(b*x)
            hbin_ginverse(hb)
            newbeta_beta+(hbin%*%ub)
            converged_all(abs(newbeta-beta)<eps)
    }
        cat("\n")
        cov.beta_hbin
        beta1_matrix(1,ncol=(inde+1),byrow=T)
        ma_mahalanobis(beta1,newbeta,cov.beta)
        md_sqrt(ma)
        result_list(beta.we=newbeta,md=md)
        result
}
```

/* วิธีปรับแก้เบื้องต้น */

```
pc.logit_function(x,m,n,pi.M,pi.Mhat,beta.start,maxits=20,eps=0.0000001)
{
    ni_rep(n,m)
    pc_log(((1-pi.M)/pi.M)*(pi.Mhat/(1-pi.Mhat)),exp(1))
    mat.pc_matrix(rep(pc,m),ncol=1,byrow=F)
    e.pc_exp(-(x%*%beta.start)+mat.pc)
    pi.pc_1/(1+e.pc)
    y1_rbinom(m,ni,pi.pc)
    iter_0
     converged_F
```

```
        newbeta_beta.start
        while((!converged)&(iter<maxits))
        {
                iter_iter+1
                beta_newbeta
                tmp_exp(-(x%*%beta)+mat.pc)
                pi_1/(1+tmp)
                mu_n*pi
                w_as.vector(n*pi*(1-pi))
                xtwx_t(w*x)%*%x
                xtwxinv_ginverse(xtwx)
                newbeta_beta+xtwxinv%*%t(x)%*%(y1-mu)
                converged_all(abs(newbeta-beta)<eps)
        }
                cat("\n")
                cov.pc_xtwxinv
                beta1_matrix(1,ncol=(inde+1),byrow=T)
                ma_mahalanobis(beta1,newbeta,cov.pc)
                md_sqrt(ma)
                result_list(beta.pc=newbeta,md=md)
                result
}

/*  ฟังก์ชันการประมาณค่าพารามิเตอร์ */

logitRe_function(k,n,m,pi.M,inde,xmat,beta.start)
{
    ni_rep(n,m)
    x_xmat
    e.xb_exp(-(x%*%beta.start))
    pi_1/(1+e.xb)
```

```
    y.bar_mean(pi)
    y_rbinom(m,ni,pi)
    w1_pi.M/pi.Mhat
    w0_(1-pi.M)/(1-pi.Mhat)


    for(i in 1:m )
    {
        fit1_ml.logit(x,y,m,n,beta.start)
        sum1_sum1+fit1$md
        fit2_we.logit(x,y,m,n,beta.start,w1,w0)
        sum2_sum2+fit2$md
        fit3_pc.logit(x,m,n,pi.M,pi.Mhat,beta.start)
        sum3_sum3+fit3$md
    }

    AMH.ml_sum1/k
    AMH.we_sum2/k
    AMH.pc_sum3/k

    output_cbind(AMH.ml,AMH.we,AMH.pc)
    output
}



sum1_0
sum2_0
sum3_0
```

```
out10_list(matrix(nrow=3),matrix(nrow=3),matrix(nrow=3),matrix(nrow=3))
out20_list(matrix(nrow=3),matrix(nrow=3),matrix(nrow=3),matrix(nrow=3))
out30_list(matrix(nrow=3),matrix(nrow=3),matrix(nrow=3),matrix(nrow=3))

n_10
ni_rep(n,m)
for (i in 1:4)
{
    out10[[i]]_logitRe(k,n,m,pi.M[i],inde,xmat,beta.start)
}

n_20
ni_rep(n,m)
out20_list(matrix(nrow=3),matrix(nrow=3),matrix(nrow=3),matrix(nrow=3))
for (i in 1:4)
{
    out20[[i]]_logitRe(k,n,m,pi.M[i],inde,xmat,beta.start)
}

n_30
ni_rep(n,m)
out30_list(matrix(nrow=3),matrix(nrow=3),matrix(nrow=3),matrix(nrow=3))
for (i in 1:4)
{
    out30[[i]]_logitRe(k,n,m,pi.M[i],inde,xmat,beta.start)
}
out10
out20
out30
```

ประวัติผู้เขียนวิทยานิพนธ์

นางสาว ทัศนาพร จงเกตุกรณ์ เกิดเมื่อวันที่ 27 กุมภาพันธ์ 2522 ที่ จ. นครสวรรค์ สำเร็จการศึกษาปริญญาตรีวิทยาศาสตรบัณฑิต สาขาสถิติ มหาวิทยาลัยเชียงใหม่ เมื่อปีการศึกษา 2543 เข้าศึกษาต่อในระดับปริญญามหาบัณฑิต สาขาสถิติ ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี