รายการอ้างอิง

ภาษาไทย

ธีระพร วีระถาวร. ความน่าจะเป็นกับการประยุกต์. พิมพ์ครั้งที่ 2. กรุงเทพมหานคร: วิทยพัฒน์, 2539.

ธีระพร วีระถาวร. ตัวแบบเชิงเส้นและการประยุกต์. พิมพ์ครั้งที่ 2. กรุงเทพมหานคร: วิทยพัฒน์, 2541.

มานพ วราภักดิ์. การจำลองเบื้องต้น. กรุงเทพมหานคร: ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย, 2547

ศิริชัย กาญจนวาสี. การวิเคราะห์พหุระดับ. พิมพ์ครั้งที่ 3. กรุงเทพมหานคร: โรงพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย, 2548.

ภาษาต่างประเทศ

Goldstein H. Multilevel Statistical Models. 2nd ed. Newyork-Toronto: John Wiley & Sons, 1995.

Snijders, T., & Bosker, R. Multilevel Analysis. London: Sage Publications, 1999.

Raudenbush, S. W., Bryk, A. S. Hierarchical Linear Models Applications and Data Analysis Methods. 2nd ed. London: Sage Publications, 2002.

Goldstein H. Multilevel mixed linear model analysis using iterative generalized least squares. Biometrika 73(1986): 43-56.

Goldstein H. Restricted unbiased iterative generalized least squares. Biometrika 76(1989): 622-623.

Van der Leeden, R., F. Busing and E. Meijer, Bootstrap method for two-level models. Department of Psychometrics and Research Methodology. Leiden University, 1997.

Maas, C.J.M. and Hox, J.J. Robustness of multilevel parameter estimates against small sample sizes. Department of Methodology and Statistics. Utrecht University, 2001.

Mass, C.J.M. and Hox, J.J. Sufficient sample sizes for multilevel modeling. Department of Methodology and Statistics. Utrecht University, 2003.

Ayebo, A. and Kozubowski, T.J. An asymmetric generalized of Gaussian and Laplace laws. Department of Mathematics, JPSS 1(2)(2003): 187-210.

ภาคผนวก

ตารางแสดงลักษณะการทำงานของโปรแกรมทั้งหมดที่ใช้ในการวิจัย

อันดับที่	ชื่อโปรแกรม	การทำงานของโปรแกรม	ชื่อโปรแกรมย่อยที่เรียกใช้
โปรแกรมหลัก	est	- สร้างข้อมูลตัวแปรอิสระ ค่าคลาดเคลื่อนสุ่มในระดับที่ 1 และระดับที่ 2 และตัวแปรตาม - คำนวณค่า RB ของวิธี IGLS และ RIGLS - คำนวณค่า RRADASE ของวิธี IGLS และ RIGLS - คำนวณค่า RRADASE ของวิธี ROBUST	ind, geny, IGLS, RIGLS, Robust_Igls, Robust_Rigls, RB, RRADASE
โปรแกรมย่อย			
1	randmcg	- การสร้างเลขสุ่มให้มีการแจกแจงแบบสม่ำเสมอในช่วง (0,1) ด้วยวิธี MCG - การสร้างเลขสุ่มให้มีการแจกแจงแบบปกติ - สร้างเลขสุ่มให้มีการแจกแจงแบบ asymmetric exponential power - การสร้างข้อมูลตัวแปรอิสระ - การสร้างค่าคลาดเคลื่อนในระดับที่ 1 และระดับที่ 2 และข้อมูลตัวแปรตาม	
2	normal		randmcg
3	asym_expo		randmcg
5	ind		normal
6	geny		normal, asym_expo

อันดับที่	ชื่อโปรแกรม	การทำงานของโปรแกรม	ชื่อโปรแกรมย่อยที่เรียกใช้
7	IGLS	- การประมาณค่าพารามิเตอร์ และค่าคลาดเคลื่อนมาตรฐานโดยวิธี IGLS	
8	RIGLS	- การประมาณค่าพารามิเตอร์และค่าคลาดเคลื่อนมาตรฐานโดยวิธี RIGLS	
9	Robust_Igls	- คำนวณค่าคลาดเคลื่อนมาตรฐานโดยใช้วิธี ROBUST ในการประมาณพารามิเตอร์ด้วยวิธี IGLS	
10	Robust_Rigls	- คำนวณค่าคลาดเคลื่อนมาตรฐานโดยใช้วิธี ROBUST ในการประมาณพารามิเตอร์ด้วยวิธี RIGLS	
11	RB	- คำนวณค่า RB ของวิธี IGLS และ RIGLS	
12	RRADASE	- คำนวณค่า RRADASE ของวิธี IGLS และ RIGLS	

```
function
[fix_seq,randp_seq,varfix_seq,varrandp_seq,fix_Rseq,randp_Rseq,varfix_Rseq,varrandp_Rseq,ROvarfix_seq,ROvarrandp
_seq,ROvarfix_Rseq,ROvarrandp_Rseq,average_iteration,average_skew_kur,RB_fix,RB_random,Relative_biased,RRSE,R
RDASE2]=est(rep,number,group,var_e,mean_u0,mean_u1,t00_p,t11_p,t00,t11,skew,kur)
[X,xj,vec_xj,W]=ind(number,group);
round=0;
for i=1:rep
  round=round+1
  roundigls=101;
  roundrigls=101;
  tic
  while roundigls>100 || roundrigls>100
    [y,skew_u,kur_u]=geny(number,group,var_e,mean_u0,mean_u1,t00,t00_p,t11_p,skew,kur,X,vec_xj,W);
    [fix,Cov_fix,randp,Cov_randp,roundigls,EEigls,V,Z]=IGLS(y,X,xj,number,group,W);
    if roundigls<=100
      [fix_R,Cov_fix_R,randp_R,Cov_randp_R,roundrigls,EErigls,V_R,Z_R]=RIGLS(y,X,xj,number,group,W);
    else roundrigls=101;
    end
  end
round_iglsseq(i,1)=roundigls;
round_riglsseq(i,1)=roundrigls;
skewi(i,1)=skew_u(1,1);
skewi(i,2)=skew_u(2,1);
kuri(i,1)=kur_u(1,1);
kuri(i,2)=kur_u(2,1);
  [Robust_Covfix,Robust_Covrandp]=Robust_Igls(V,EEigls,X,number,group,randp,Z);
  [Robust_Covfix_R,Robust_Covrandp_R]=Robust_Rigls(V_R,EErigls,X,number,group,randp_R,Z_R);

fix_seq(i,1)=fix(1,1);
fix_seq(i,2)=fix(2,1);
randp_seq(i,1)=randp(1,1);
randp_seq(i,2)=randp(2,1);
randp_seq(i,3)=randp(3,1);
varfix_seq(i,1)=(Cov_fix(1,1));
varfix_seq(i,2)=(Cov_fix(2,2));
```

```
varrandp_seq(i,1)=(Cov_randp(1,1));
varrandp_seq(i,2)=(Cov_randp(2,2));
varrandp_seq(i,3)=(Cov_randp(3,3));

fix_Rseq(i,1)=fix_R(1,1);
fix_Rseq(i,2)=fix_R(2,1);
randp_Rseq(i,1)=randp_R(1,1);
randp_Rseq(i,2)=randp_R(2,1);
randp_Rseq(i,3)=randp_R(3,1);
varfix_Rseq(i,1)=(Cov_fix_R(1,1));
varfix_Rseq(i,2)=(Cov_fix_R(2,2));
varrandp_Rseq(i,1)=(Cov_randp_R(1,1));
varrandp_Rseq(i,2)=(Cov_randp_R(2,2));
varrandp_Rseq(i,3)=(Cov_randp_R(3,3));

ROvarfix_seq(i,1)=(Robust_Covfix(1,1));
ROvarfix_seq(i,2)=(Robust_Covfix(2,2));
ROvarrandp_seq(i,1)=(Robust_Covrandp(1,1));
ROvarrandp_seq(i,2)=(Robust_Covrandp(2,2));
ROvarrandp_seq(i,3)=(Robust_Covrandp(3,3));

ROvarfix_Rseq(i,1)=(Robust_Covfix_R(1,1));
ROvarfix_Rseq(i,2)=(Robust_Covfix_R(2,2));
ROvarrandp_Rseq(i,1)=(Robust_Covrandp_R(1,1));
ROvarrandp_Rseq(i,2)=(Robust_Covrandp_R(2,2));
ROvarrandp_Rseq(i,3)=(Robust_Covrandp_R(3,3));
toc

end
average_iteration{1,2}='average';
average_iteration{1,3}='std';
average_iteration{2,1}='igls';
average_iteration{2,2}=mean(round_iglsseq);
average_iteration{2,3}=std(round_iglsseq);
average_iteration{3,1}='rigls';
average_iteration{3,2}=mean(round_riglsseq);
average_iteration{3,3}=std(round_riglsseq);

average_skew_kur{2,1}='average';
average_skew_kur{1,2}='skew_u0';
average_skew_kur{1,3}='skew_u1';
```

```
average_skew_kur{1,4}='kur_u0';
average_skew_kur{1,5}='kur_u1';
average_skew_kur{2,2}=mean(skewi(:,1));
average_skew_kur{2,3}=mean(skewi(:,2));
average_skew_kur{2,4}=mean(kuri(:,1));
average_skew_kur{2,5}=mean(kuri(:,2));
[RB_fix,RB_random,Relative_biased]=RB(fix_seq,randp_seq,fix_Rseq,randp_Rseq,var_e,t00,t11)
[RRSE,RRDASE2]=RRDSE(rep,fix_seq,randp_seq,varfix_seq,varrandp_seq,fix_Rseq,randp_Rseq,varfix_Rseq,varrandp_R
seq,ROvarfix_seq,ROvarrandp_seq,ROvarfix_Rseq,ROvarrandp_Rseq)

function r = randmcg(p,q)
persistent m a c x
if isempty(x)
  m  = 2^31-1;
  a = 7^5;
  c = 0;
  x = 1;
end

if nargin < 1, p = 1; end
if nargin < 2, q = p; end
r = zeros(p,q);
for k = 1:p*q
  x = rem(a*x + c, m);
  r(k) = x/m;
end

function [norm]=normal(mean1,stdev1)
r1=randmcg(1);
r2=randmcg(1);
znorm=sqrt(-2*log(r1))*cos(2*pi*r2);
norm=(znorm*stdev1)+mean1;


function [asym_expo]=asym_expo(mean2,stdev2,skew,kur)
alpha=1/kur;
g=gam(alpha,1);
u=randmcg(1);
k=(1/(1+(skew^2)));
if u<(1/(1+skew^2))
   l=1/skew;
```

```
else l=-skew;
end
  asym_expo=mean2+stdev2*l*(g^(1/kur));



function [X,xj,vec_xj,W]=ind(number,group)
xj=ones(number,group);
for j=1:group
  for i=1:number
  xj(i,j)=normal(3,16);
  end
W(1+(number)*(j-1):j*number,2*j-1)=1;
W(1+(number)*(j-1):j*number,2*j)=xj(:,j);
end

X=ones(number*group,2);
vec_xj=xj(:);
for i=1:number*group
  X(i,2)=vec_xj(i,1);
end

function [y,skew_u,kur_u]=geny(number,group,var_e,mean_u0,mean_u1,t00,t00_p,t11_p,skew,kur,X,vec_xj,W)
fix=[1;2];
for j=1:group
  u0j=asym_expo(mean_u0,sqrt(t00_p),skew,kur);
  u1j=asym_expo(mean_u1,sqrt(t11_p),skew,kur);
  uj=[u0j;u1j];
  u0(j,1)=u0j;
  u1(j,1)=u1j;
for i=1:number
  ej(i,1)=normal(0,sqrt(var_e)); %level-1 error
end
Xj=X((j-1)*number+1:j*number,:);
yj(:,j)=Xj*fix+Xj*uj+ej;
end
y=yj(:);

function [fix,Cov_fix,randp,Cov_randp,roundigls,EEigls,V,Z]=IGLS(y,X,xj,number,group,W)
Tu=zeros(2*group);
Tuj=zeros(2,2);
```

```
n=number*group;
fix_0=inv(X'*X)*(X'*y);
mse_0=(y'*y-(fix_0)'*X'*y)/((number*group)-2);
randp_0=[mse_0;0;0];
diff_fix=ones(2,1);
diff_randp=ones(3,1);

roundigls=0;
mse=mse_0;
while (diff_fix(1,1)>10^(-5) || diff_fix(2,1)>10^(-5) || diff_randp(1,1)>10^(-5) || diff_randp(2,1)>10^(-5) ||
diff_randp(3,1)>10^(-5))&&(roundigls<=100)
V=W*Tu*W'+mse*eye(number*group);
invV=bbinv(V);
 fix=inv(X'*invV*X)*(X'*invV*y);
EEigls=zeros(number*group,number*group);
for j=1:group
   yj=y((j-1)*number+1:j*number,1);
   Xj=X((j-1)*number+1:j*number,:);
Ej=yj-Xj*fix;
EEiglsj=Ej*Ej';
EEigls((j-1)*number+1:j*number,(j-1)*number+1:j*number)=EEiglsj;
end
vecE=EEigls(:);
col1_Z=eye(n,n);
col2_Z=zeros(n,n);
col3_Z=zeros(n,n);
for j=1:group
o=ones(number,number);
col2_Z((j-1)*(number)+1:j*number,(j-1)*(number)+1:j*number)=o;
col3_Z((j-1)*(number)+1:j*number,(j-1)*(number)+1:j*number)=xj(:,j)*xj(:,j)';
end
Z=ones((number*group)^2,3);
Z(:,1)=col1_Z(:);
Z(:,2)=col2_Z(:);
Z(:,3)=col3_Z(:);
invVV=bbkron(invV,invV);
randp=inv(Z'*invVV*Z)*Z'*invVV*vecE;
Tuj(1,1)=randp(2,1);Tuj(1,2)=0;Tuj(2,1)=0;Tuj(2,2)=randp(3,1);
for j=1:group
   Tu(2*j-1:j*2,2*j-1:2*j)=Tuj;
end
```

```
mse=randp(1,1);
diff_fix=abs(fix-fix_0);
diff_randp=abs(randp-randp_0);
fix_0=fix;
randp_0=randp;
roundigls=roundigls+1;
end
Cov_fix=inv(X'*invV*X);
Cov_randp=2*inv(Z'*invVV*Z);


function [fix_R,Cov_fix_R,randp_R,Cov_randp_R,roundrigls,EErigls,V_R,Z_R]=RIGLS(y,X,xj,number,group,W)
Tu=zeros(2*group);
n=number*group;
fix_0=inv(X'*X)*(X'*y);
mse_0=(y'*y-(fix_0)'*X'*y)/((number*group)-2);
randp_0=[mse_0;0;0];
diff_fix=ones(2,1);
diff_randp=ones(3,1);
roundrigls=0;
mse_R=mse_0;
while (diff_fix(1,1)>10^(-5) || diff_fix(2,1)>10^(-5) || diff_randp(1,1)>10^(-5) || diff_randp(2,1)>10^(-5) ||
diff_randp(3,1)>10^(-5))&&(roundrigls<=100)
V_R=W*Tu*W'+mse_R*eye(number*group);
invV=bbinv(V_R);
fix_R=inv(X'*invV*X)*(X'*invV*y);r
EErigls=zeros(number*group,number*group);
for j=1:group
   yj=y((j-1)*number+1:j*number,1);
   Xj=X((j-1)*number+1:j*number,:);
Ej=yj-Xj*fix_R;
EEriglsj=Ej*Ej';
EErigls((j-1)*number+1:j*number,(j-1)*number+1:j*number)=EEriglsj;
end
EErigls=EErigls+(X*inv(X'*invV*X)*X');vecE=EErigls(:);
col1_Z=eye(n,n);
col2_Z=zeros(n,n);
col3_Z=zeros(n,n);
for j=1:group
o=ones(number,number);
col2_Z((j-1)*(number)+1:j*number,(j-1)*(number)+1:j*number)=o;
```

```
col3_Z((j-1)*(number)+1:j*number,(j-1)*(number)+1:j*number)=xj(:,j)*xj(:,j)';
end
Z_R=ones((number*group)^2,3);
Z_R(:,1)=col1_Z(:);
Z_R(:,2)=col2_Z(:);
Z_R(:,3)=col3_Z(:);
invVV=bbkron(invV,invV);
randp_R=inv(Z_R'*invVV*Z_R)*Z_R'*invVV*vecE;
Tuj(1,1)=randp_R(2,1);Tuj(1,2)=0;Tuj(2,1)=0;Tuj(2,2)=randp_R(3,1);
for j=1:group
   Tu(2*j-1:j*2,2*j-1:2*j)=Tuj;
end
mse_R=randp_R(1,1);
diff_fix=abs(fix_R-fix_0);
diff_randp=abs(randp_R-randp_0);
fix_0=fix_R;
randp_0=randp_R;
roundrigls=roundrigls+1;
end
Cov_fix_R=inv(X'*invV*X);
Cov_randp_R=2*inv(Z_R'*invVV*Z_R);


function [Robust_Covfix,Robust_Covrandp]=Robust_Igls(V,EEigls,X,number,group,randp,Z)
EE=EEigls;
Xj=ones(number,2);
vecE=EE(:);
R=vecE-Z*randp;
invV=bbinv(V)*eye(number*group);
H=0;
C=0;
Hrandp=0;
Crandp=0;
for j=1:group
   Xj(:,2)=X((j-1)*number+1:j*number,2);
   invVj=invV((j-1)*number+1:j*number,(j-1)*number+1:j*number);
   EEj=EE((j-1)*number+1:j*number,(j-1)*number+1:j*number);
   Hj=Xj'*invVj*Xj;
   Cj=Xj'*invVj*EEj*invVj*Xj;
   H=H+Hj;
   C=C+Cj;
```

```
  invVVj=bbkron(invVj,invV);
  Rj=bbmatrix(R((j-1)*(number^2)*group+1:j*(number^2)*group,1));
  RRj=Rj*Rj';
  Zj=Z((j-1)*(number^2)*group+1:j*(number^2)*group,:);
  Hrandpj=Zj'*(invVVj)*Zj;
  Crandpj=Zj'*invVVj*(RRj)*invVVj*Zj;
  Hrandp=Hrandp+Hrandpj;
  Crandp=Crandp+Crandpj;
end
Robust_Covfix=inv(H)*C*inv(H);
Robust_Covrandp=inv(Hrandp)*Crandp*inv(Hrandp);


function [Robust_Covfix_R,Robust_Covrandp_R]=Robust_Rigls(V_R,EErigls,X,number,group,randp_R,Z_R)
invV=bbinv(V_R);
EE=EErigls;
vecE=EE(:);
R=vecE-Z_R*randp_R;
H=0;
C=0;
Hrandp=0;
Crandp=0;
for j=1:group
   Vj=V_R((j-1)*number+1:j*number,(j-1)*number+1:j*number);
   invVj=bbinv(Vj);
   Xj=X((j-1)*number+1:j*number,:);
   EEj=EE((j-1)*number+1:j*number,(j-1)*number+1:j*number);
   Hj=(Xj'*(invVj)*Xj);
   Cj=Xj'*invVj*(EEj)*invVj*Xj;
   H=H+Hj;
   C=C+Cj;

   invVVj=bbkron(invVj,invV);
   Rj=bbmatrix(R((j-1)*(number^2)*group+1:j*(number^2)*group,1));
   Zj=Z_R((j-1)*(number^2)*group+1:j*(number^2)*group,:);
   RRj=Rj*Rj';
   Hrandpj=Zj'*(invVVj)*Zj;
   Crandpj=Zj'*invVVj*(RRj)*invVVj*Zj;
   Hrandp=Hrandp+Hrandpj;
```

```
  Crandp=Crandp+Crandpj;
end
Robust_Covfix_R=inv(H)*C*inv(H);
Robust_Covrandp_R=inv(Hrandp)*Crandp*inv(Hrandp);



function [Relative_biased]=RB(fix_seq,randp_seq,fix_Rseq,randp_Rseq,var_e,t00,t11)
meanfix(1,1)=mean(fix_seq(:,1));
meanfix(2,1)=mean(fix_seq(:,2));
RBfix=sqrt((meanfix-[1;2])'*inv([1 0;0 4])*(meanfix-[1;2]));
meansigma2=mean(randp_seq(:,1));
RBsigma2=sqrt((meansigma2-(var_e))'*((var_e)^(-2))*(meansigma2-(var_e)));
meanlevel_2_var(1,1)=mean(randp_seq(:,2));
meanlevel_2_var(2,1)=mean(randp_seq(:,3));
RBlevel_2_var=sqrt((meanlevel_2_var-[t00;t11])'*inv([t00^2 0;0 t11^2])*((meanlevel_2_var-[t00;t11])));

 meanfix_R(1,1)=mean(fix_Rseq(:,1));
meanfix_R(2,1)=mean(fix_Rseq(:,2));
RBfix_R=sqrt((meanfix_R-[1;2])'*inv([1 0;0 4])*(meanfix_R-[1;2]));
meansigma2_R=mean(randp_Rseq(:,1));
RBsigma2_R=sqrt((meansigma2_R-(var_e))'*((var_e)^(-2))*(meansigma2_R-(var_e)));
meanlevel_2_var_R(1,1)=mean(randp_Rseq(:,2));
meanlevel_2_var_R(2,1)=mean(randp_Rseq(:,3));
RBlevel_2_var_R=sqrt((meanlevel_2_var_R-[t00;t11])'*inv([t00^2 0;0 t11^2])*((meanlevel_2_var_R-[t00;t11])));
   Relative_biased{1,2}='IGLS';
   Relative_biased{1,3}='RIGLS';
   Relative_biased{2,1}='fix parameter';
   Relative_biased{3,1}='level-1 variance';
   Relative_biased{4,1}='level-2 variance';
   Relative_biased{2,2}=RBfix;
   Relative_biased{2,3}=RBfix_R
   Relative_biased{3,2}=RBsigma2;
   Relative_biased{3,3}=RBsigma2_R;
   Relative_biased{4,2}=RBlevel_2_var;
   Relative_biased{4,3}=RBlevel_2_var_R;

function
[RRSE,RRDASE2]=RRDSE(rep,fix_seq,randp_seq,varfix_seq,varrandp_seq,fix_Rseq,randp_Rseq,varfix_Rseq,varrandp_R
seq,ROvarfix_seq,ROvarrandp_seq,ROvarfix_Rseq,ROvarrandp_Rseq)
```

```
MCSE(1,1)=std(fix_seq(:,1));
MCSE(2,1)=std(fix_seq(:,2));
MCSE(3,1)=std(randp_seq(:,1));
MCSE(4,1)=std(randp_seq(:,2));
MCSE(5,1)=std(randp_seq(:,3));


MCSE_R(1,1)=std(fix_Rseq(:,1));
MCSE_R(2,1)=std(fix_Rseq(:,2));
MCSE_R(3,1)=std(randp_Rseq(:,1));
MCSE_R(4,1)=std(randp_Rseq(:,2));
MCSE_R(5,1)=std(randp_Rseq(:,3));

ASEM(1,1)=sqrt(mean(varfix_seq(:,1)));
ASEM(2,1)=sqrt(mean(varfix_seq(:,2)));
ASEM(3,1)=sqrt(mean(varrandp_seq(:,1)));
ASEM(4,1)=sqrt(mean(varrandp_seq(:,2)));
ASEM(5,1)=sqrt(mean(varrandp_seq(:,3)));

ASEM_R(1,1)=sqrt(mean(varfix_Rseq(:,1)));
ASEM_R(2,1)=sqrt(mean(varfix_Rseq(:,2)));
ASEM_R(3,1)=sqrt(mean(varrandp_Rseq(:,1)));
ASEM_R(4,1)=sqrt(mean(varrandp_Rseq(:,2)));
ASEM_R(5,1)=sqrt(mean(varrandp_Rseq(:,3)));

AROSE(1,1)=sqrt(mean(ROvarfix_seq(:,1)));
AROSE(2,1)=sqrt(mean(ROvarfix_seq(:,2)));
AROSE(3,1)=sqrt(mean(ROvarrandp_seq(:,1)));
AROSE(4,1)=sqrt(mean(ROvarrandp_seq(:,2)));
AROSE(5,1)=sqrt(mean(ROvarrandp_seq(:,3)));

AROSE_R(1,1)=sqrt(mean(ROvarfix_Rseq(:,1)));
AROSE_R(2,1)=sqrt(mean(ROvarfix_Rseq(:,2)));
AROSE_R(3,1)=sqrt(mean(ROvarrandp_Rseq(:,1)));
AROSE_R(4,1)=sqrt(mean(ROvarrandp_Rseq(:,2)));
AROSE_R(5,1)=sqrt(mean(ROvarrandp_Rseq(:,3)));

RRDASE(1,1)=sqrt(([ASEM(1,1);ASEM(2,1)]-[MCSE(1,1);MCSE(2,1)])'*inv([MCSE(1,1)^2 0;0
MCSE(2,1)^2])*([ASEM(1,1);ASEM(2,1)]-[MCSE(1,1);MCSE(2,1)]));
RRDASE(2,1)=sqrt((ASEM(3,1)-(MCSE(3,1)))'*((MCSE(3,1))^(-2))*(ASEM(3,1)-(MCSE(3,1))));
```

```
RRDASE(3,1)=sqrt(([ASEM(4,1);ASEM(5,1)]-[MCSE(4,1);MCSE(5,1)])'*inv([MCSE(4,1)^2 0;0
MCSE(5,1)^2])*(([ASEM(4,1);ASEM(5,1)]-[MCSE(4,1);MCSE(5,1)])));

RRDASE_R(1,1)=sqrt(([ASEM_R(1,1);ASEM_R(2,1)]-[MCSE_R(1,1);MCSE_R(2,1)])'*inv([MCSE_R(1,1)^2 0;0
MCSE_R(2,1)^2])*([ASEM_R(1,1);ASEM_R(2,1)]-[MCSE_R(1,1);MCSE_R(2,1)]));
RRDASE_R(2,1)=sqrt((ASEM_R(3,1)-(MCSE_R(3,1)))'*((MCSE_R(3,1))^(-2))*(ASEM_R(3,1)-(MCSE_R(3,1))));
RRDASE_R(3,1)=sqrt(([ASEM_R(4,1);ASEM_R(5,1)]-[MCSE_R(4,1);MCSE_R(5,1)])'*inv([MCSE_R(4,1)^2 0;0
MCSE_R(5,1)^2])*(([ASEM_R(4,1);ASEM_R(5,1)]-[MCSE_R(4,1);MCSE_R(5,1)])));

RRDROASE(1,1)=sqrt(([AROSE(1,1);AROSE(2,1)]-[MCSE(1,1);MCSE(2,1)])'*inv([MCSE(1,1)^2 0;0
MCSE(2,1)^2])*([AROSE(1,1);AROSE(2,1)]-[MCSE(1,1);MCSE(2,1)]));
RRDROASE(2,1)=sqrt((AROSE(3,1)-(MCSE(3,1)))'*((MCSE(3,1))^(-2))*(AROSE(3,1)-(MCSE(3,1))));
RRDROASE(3,1)=sqrt(([AROSE(4,1);AROSE(5,1)]-[MCSE(4,1);MCSE(5,1)])'*inv([MCSE(4,1)^2 0;0
MCSE(5,1)^2])*(([AROSE(4,1);AROSE(5,1)]-[MCSE(4,1);MCSE(5,1)])));

RRDROASE_R(1,1)=sqrt(([AROSE_R(1,1);AROSE_R(2,1)]-[MCSE_R(1,1);MCSE_R(2,1)])'*inv([MCSE_R(1,1)^2 0;0
MCSE_R(2,1)^2])*([AROSE_R(1,1);AROSE_R(2,1)]-[MCSE_R(1,1);MCSE_R(2,1)]));
RRDROASE_R(2,1)=sqrt((AROSE_R(3,1)-(MCSE_R(3,1)))'*((MCSE_R(3,1))^(-2))*(AROSE_R(3,1)-(MCSE_R(3,1))));
RRDROASE_R(3,1)=sqrt(([AROSE_R(4,1);AROSE_R(5,1)]-[MCSE_R(4,1);MCSE_R(5,1)])'*inv([MCSE_R(4,1)^2 0;0
MCSE_R(5,1)^2])*(([AROSE_R(4,1);AROSE_R(5,1)]-[MCSE_R(4,1);MCSE_R(5,1)])));

RRDASE2{2,1}='fix parameter';
RRDASE2{3,1}='level-1 variance';
RRDASE2{4,1}='level-2 variance';
RRDASE2{1,2}='Model base IGLS';
RRDASE2{1,3}='Robust SE IGLS';
RRDASE2{1,5}='Model base RIGLS';
RRDASE2{1,6}='Robust SE RIGLS';
RRDASE2{2,2}=RRDASE(1,1);
RRDASE2{3,2}=RRDASE(2,1);
RRDASE2{4,2}=RRDASE(3,1);
RRDASE2{2,3}=RRDROASE(1,1);
RRDASE2{3,3}=RRDROASE(2,1);
RRDASE2{4,3}=RRDROASE(3,1);
RRDASE2{2,5}=RRDASE_R(1,1);
RRDASE2{3,5}=RRDASE_R(2,1);
RRDASE2{4,5}=RRDASE_R(3,1);
RRDASE2{2,6}=RRDROASE_R(1,1);
RRDASE2{3,6}=RRDROASE_R(2,1);
RRDASE2{4,6}=RRDROASE_R(3,1);
```

ประวัติผู้เขียนวิทยานิพนธ์

ชื่อ นายสิวะโชติ ศรีสุทธิยากร

วัน เดือน ปีเกิด 5 มีนาคม 2526

การศึกษา

ปริญญาตรี	วิทยาศาสตร์บัณฑิต (คณิตศาสตร์) คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
มัธยมศึกษา	รร. สาธิตจุฬาลงกรณ์มหาวิทยาลัย (ฝ่ายมัธยม)
ประถมศึกษา	รร. สาธิตจุฬาลงกรณ์มหาวิทยาลัย (ฝ่ายประถม)

ประสบการณ์ในอดีต

2549	นักกีฬามหาวิทยาลัยเข้าแข่งขันในกีฬามหาวิทยาลัยแห่งประเทศไทย ประเภทกีฬาเทเบิลเทนนิส
2547	ประธานชมรมเทเบิลเทนนิส จุฬาลงกรณ์มหาวิทยาลัย
2545-2547	นักกีฬามหาวิทยาลัยเข้าแข่งขันในกีฬามหาวิทยาลัยแห่งประเทศไทย ประเภทกีฬาเทเบิลเทนนิส
2545	รองประธานชมรมดนตรีไทย สโมสรนิสิต จุฬาลงกรณ์มหาวิทยาลัย

ผลงานในอดีต

2547	โครงงานเรื่องการศึกษาผลกระทบต่อปริมาณน้ำฝน และอุณหภูมิของปรากฏการณ์เอลนิโน และลานีญาต่อประเทศไทย