APPENDICES

APPENDIX A

LISTING OF GAUSS PROGRAM FOR COMPARISON OF DIFFERENT ESTIMATORS IN THE MEASUREMENT-ERROR MODEL

A.1 MAIN PROGRAM

```
start:;
cls;
"              Simulated comparison for measurement error model";
"              ==============================================";
print;
print;
"                              David R. Fox";
print;
"                          University of Wyoming";
"                              January 1989";
print;
print;
count2=0;
cov=eye(2);
/* coeff=zeros(20,1); */;
"Enter parameter values :";
print;
"     How many observations for each sample : ";;
nobs=con(1,1);
print;
"     Length of simulation run : ";;
nrun=con(1,1);
errc=zeros(nrun,1);
erro=zeros(nrun,1);
errm=zeros(nrun,1);
errf=zeros(nrun,1);
print;
/* "     How many correlation coefficients : ";;
ncoef=con(1,1);
print;
format /rd 2,0;
i=1;
do while i<=ncoef;
"     Enter coefficient #";;i;;" : ";;
coeff[i,1]=con(1,1);
```

```
i=i+1;
endo; */;
"     Enter correlation coefficient : ";;
rho=con(1,1);
print;
"     Enter sigma-squared e : ";;
se=con(1,1);
print;
"     Enter sigma-squared u : ";;
su=con(1,1);
print;
"     Enter sigma-squared x : ";;
varx=con(1,1);
lamda=se/su;
repeat=0;
loop1:;
count2=0;
if repeat>0;
/* count=count+1;
if count>ncoef; */;
print;
print;
"      Do you want to repeat for new parameters (y/n) : ";;
reply=cons;
if reply $=="Y" or reply $=="y";
goto start;
else;
format /rd 9,4;
stop;
endif;
endif;
/* endif;
rho=coeff[count,1]; */;
theta=rho*lamda^.5;
cov[1,1]=se;
cov[1,2]=rho*(se*su)^.5;
cov[2,1]=cov[1,2];
cov[2,2]=su;
{d,Q}=eigrs2(cov);
dd=diagrv(eye(2),d^.5);
a=q*dd;
loop2:;
repeat=1;
count2=count2+1;
if count2>nrun;
goto loop1;
endif;
"Iteration #";;count2;
z=rndn(nobs+1,2);
w=z*a';
/* w1=w[.,1]-meanc(w[.,1]);w2=w[.,2]-meanc(w[.,2]);ww=w1~w2;ww'ww/(nobs-
e=w[.,1];
u=w[.,2];
```

```
x=rndn(nobs+1,1)*varx^.5+ones(nobs+1,1)*10;
xx=x[1:nobs,1]+u[1:nobs,1];
y=0.5+xx+e[1:nobs,1];
x0=x[nobs+1,1];
y0=.5+x0+e[nobs+1,1];
sx=(xx)'(xx)-sumc(xx)^2/nobs;
sy=y'y-sumc(y)^2/nobs;
sxy=xx'y-sumc(xx)*sumc(y)/nobs;
beta1c=sxy/sx;
beta0c=(sumc(y)-beta1c*sumc(xx))/nobs;
format /rd 9,6;
/* beta0c;
beta1c; */;
errc[count2,1]=((y0-beta0c)/beta1c)-x0;
beta1o=(sy-sx+((sy-sx)^2+4*sxy^2)^.5)/2/sxy;
beta0o=(sumc(y)-beta1o*sumc(xx))/nobs;
erro[count2,1]=((y0-beta0o)/beta1o)-x0;
beta1m=(sy-lamda*sx+((sy-lamda*sx)^2-4*(sxy-theta*sx)*(theta*sy-lamda*sx
/2/(sxy-theta*sx);
beta0m=(sumc(y)-beta1m*sumc(xx))/nobs;
errm[count2,1]=((y0-beta0m)/beta1m)-x0;
lamda1=(sx-sxy^2/sy)/su;
if lamda1>1;
gamma1=((sx-su)*sxy+2*sxy*su/nobs)/(sxy^2+(sx*sy-sxy^2)/nobs);
else;
gamma1=sxy/sy;
endif;
gamma0=(sumc(xx)-gamma1*sumc(y))/nobs;
errf[count2,1]=(gamma0+gamma1*y0)-x0;
goto loop2;
```

A.2 AUXILIARY ROUTINE :Q.ARC

```
proc (0)=aa;
meanc(errc);meanc(erro);meanc(errm);meanc(errf);
print;
print;
errc'errc/nobs;erro'erro/nobs;errm'errm/nobs;errf'errf/nobs;
q=errc~erro~errm~errf;
endp;
```

A.3 RESULTS.ARC

```
load path=d:\simul;
bias=zeros(81,4);
mse=zeros(81,4);
i=1;
do while i<=81;
if i<10;
```

```
ll=1;
else;
ll=2;
endif;
j=1;
ext=ftos(i,"lf",ll,0);
ext;
file="run" $+ ext;
file;
load q=^file;
do while j<=4;
bias[i,j]=meanc(q[.,j]);
mse[i,j]=q[.,j]'q[.,j]/200;
j=j+1;
endo;
i=i+1;
endo;
```

A.4 PLOT.ARC

```
i=1;
_qmajor=2;
library qgraph;
window(2,2);
beggraph;
do while i<=4;
{c,m,f}=hist(bias[.,i],12);
i=i+1;
endo;
endgraph;
beggraph;
i=1;
do while i<=4;
{c,m,f}=hist(mse[.,i],12);
i=i+1;
endo;
endgraph;
```

APPENDIX B

LISTING OF GAUSS PROGRAM TO COMPUTE UNCONDITIONAL CONFIDENCE LEVELS FOR THE CSS PROCEDURE

```
start:;
cls;
"              Experimentwise error rate for the CSS procedure";
"              ==============================================";
print;
print;
"                             David R. Fox";
print;
"                         University of Wyoming";
"                             January 1989";
print;
print;
count2=1;
index=1;
"Enter parameter values :";
print;
"     How many observations for each sample : ";;
nobs=con(1,1);
print;
"     Length of each simulation run : ";;
nrun=con(1,1);
print;
"              <<<<<<  Vectors must be present for cv,F,T   >>>>>>";
i=rows(cv);
j=rows(f);
k=rows(t);
alpha=zeros(i*j*k,4);
cvcount=1;
do while cvcount<=i;
fcount=1;
do while fcount<=j;
tcount=1;
do while tcount<=k;
tcrit=t[tcount,1];
fcrit=(2*f[fcount,1])^.5;
format /ld 6,3;
```

```
"     Current iteration : cv=";;cv[cvcount,1];;
" F=";;f[fcount,1];;" t=";;t[tcount,1];
ind=zeros(nrun,1);
count2=1;
do while count2<=nrun;
x=rndu(nobs+1,1);
e=cv[cvcount,1]*rndn(nobs+1,1);
y=x+e;
xbar=meanc(x);
x0=x[nobs+1,1];
y0=y[nobs+1,1];
xx=x[1:nobs,1];
sx=xx'xx-sumc(xx)^2/nobs;
yy=y[1:nobs,1];
xx=ones(nobs,1)~xx;
beta=inv(xx'xx)*xx'yy;
xhat=(y0-beta[1,1])/beta[2,1];
ehat=yy-xx*beta;
sigma=(ehat'ehat/(nobs-2))^.5;
delta=sigma*(tcrit+fcrit*(1/nobs+(xhat-xbar)^2/sx)^.5);
yupper=y0+delta;
ylower=y0-delta;
xupper=(yupper-beta[1,1])/beta[2,1];
xlower=(ylower-beta[1,1])/beta[2,1];
if x0>=xlower and x0<=xupper;
ind[count2,1]=1;
endif;
count2=count2+1;
endo;
format /rd 12,6;
alpha[index,1]=cv[cvcount,1];
alpha[index,2]=f[fcount,1];
alpha[index,3]=t[tcount,1];
alpha[index,4]=1-sumc(ind)/nrun;
tcount=tcount+1;
index=index+1;
endo;
fcount=fcount+1;
endo;
cvcount=cvcount+1;
endo;
stop;
```

APPENDIX C

LISTING OF GAUSS PROGRAM FOR CONFIDENCE INTERVAL SIMULATION

```
start:;
bell=chrs(7);
cls;
"              Confidence interval simulation for calibration";
"              ================================================";
print;
print;
"                             David R. Fox";
print;
"                         University of Wyoming";
"                             January 1989";
print;
print;
count2=1;
"Enter parameter values :";
print;
"      How many observations for each sample : ";;
nobs=con(1,1);
print;
"      Length of each simulation run : ";;
nrun=con(1,1);
print;
"      Enter coefficient of variation : ";;
cv=con(1,1);
print;
"      Enter F-value for CSS procedure : ";;
f=con(1,1);
print;
"      Enter T-value for CSS procedure : ";;
tcss=con(1,1);
print;
"      Enter T-value for all other procedures : ";;
tcrit=con(1,1);
fcrit=(2*f)^.5;
alpha=zeros(1,5);
length=zeros(nrun,5);
format /ld 4,0;
ind=zeros(nrun,5);
count2=1;
```

```
timeO=hsec/100;
do while count2<=nrun;
timen=hsec/100;
format /ld 3,0;
if timen-time0>30;
"           Still working . . .   ";;count2/nrun*100;;"% completed";
time0=timen;
endif;
x=rndu(nobs+1,1);
e=cv*rndn(nobs+1,1);
y=x+e;
x0=x[nobs+1,1];
y0=y[nobs+1,1];
xx=x[1:nobs,1];
sx=xx'xx-sumc(xx)^2/nobs;
yy=y[1:nobs,1];
xbar=meanc(xx);
ybar=meanc(yy);
sy=yy'yy-sumc(yy)^2/nobs;
sxy=xx'yy-sumc(xx)*sumc(yy)/nobs;
xx=ones(nobs,1)~xx;
beta=inv(xx'xx)*xx'yy;
xhatc=(y0-beta[1,1])/beta[2,1];
ehatc=yy-xx*beta;
sigmac=(ehatc'ehatc/(nobs-2))^.5;
/*          CSS PROCEDURE  #4      */;
delta=sigmac*(tcss+fcrit*(1/nobs+(xhatc-xbar)^2/sx)^.5);
yupper=y0+delta;
ylower=y0-delta;
xupper=(yupper-beta[1,1])/beta[2,1];
xlower=(ylower-beta[1,1])/beta[2,1];
length[count2,4]=xupper-xlower;
if x0>=xlower and x0<=xupper;
ind[count2,4]=1;
endif;
/*          CLASSICAL PROCEDURE  #2     */;
delta=tcrit*sigmac*(1+1/nobs+(xhatc-xbar)^2/sx)^.5;
yupper=y0+delta;
ylower=y0-delta;
xupper=(yupper-beta[1,1])/beta[2,1];
xlower=(ylower-beta[1,1])/beta[2,1];
length[count2,2]=xupper-xlower;
if x0>=xlower and x0<=xupper;
ind[count2,2]=1;
endif;
/*          INVERSE PROCEDURE  #1         */;
gamma1=beta[2,1]*sx/sy;
gamma0=xbar-gamma1*ybar;
xhati=gamma0+gamma1*y0;
ehati=x[1:nobs]-(gamma0+gamma1*yy);
sigmai=(ehati'ehati/(nobs-2))^.5;
delta=tcrit*sigmai*(1+1/nobs+(y0-ybar)^2/sy)^.5;
xupper=xhati+delta;
```

```
xlower=xhati-delta;
length[count2,1]=xupper-xlower;
if x0>=xlower and x0<=xupper;
ind[count2,1]=1;
endif;
/*          CLASSICAL (GRAYBILL) PROCEDURE   #3          */;
a=beta[2,1]^2-sigmac^2*tcrit^2/sx;
b=xbar+beta[2,1]*(y0-ybar)/a;
delta=(sigmac/a)*tcrit*(a*(1+1/nobs)+(y0-ybar)^2/sx)^.5;
xupper=b+delta;
xlower=b-delta;
length[count2,3]=xupper-xlower;
if x0>=xlower and x0<=xupper;
ind[count2,3]=1;
endif;
/*          ORTHOGONAL PROCEDURE   #5              */;
b1=(sy-sx+((sy-sx)^2+4*sxy^2)^.5)/2/sxy;
b0=ybar-b1*xbar;
xhato=(y0-b0)/b1;
theta=arctan(b1);
ehato=yy-(b0+b1*x[1:nobs,1]);
/* sigmao=((ehato'ehato/(nobs-2))^.5);  */;
sigmao=cos(theta)^2*sigmac+sin(theta)^2*sigmai;
delta=tcrit*sigmao*(1+1/nobs+(xhato-xbar)^2/sx)^.5;
yupper=y0+delta;
ylower=y0-delta;
xupper=(yupper-b0)/b1;
xlower=(ylower-b0)/b1;
length[count2,5]=xupper-xlower;
if x0>=xlower and x0<=xupper;
ind[count2,5]=1;
endif;
count2=count2+1;
endo;
format /rd 12,6;
alpha[1,1]=sumc(ind[.,1])/nrun;
alpha[1,2]=sumc(ind[.,2])/nrun;
alpha[1,3]=sumc(ind[.,3])/nrun;
alpha[1,4]=sumc(ind[.,4])/nrun;
alpha[1,5]=sumc(ind[.,5])/nrun;
print;
"       Probability of interval capturing unknown x :";
alpha;
"       Average interval length : ";
(meanc(length))';
bell;bell;bell;
stop;
```

APPENDIX D

ELEMENTS OF THE GRADIENT VECTOR AND HESSIAN MATRIX

We give here the computing formulae required to determine the elements of the gradient vector and Hessian matrix for the components of variance estimation associated with the example given in §2.7.2.

We have :

$$\phi(\sigma_e^2;\sigma_u^2;\rho) = \sum_{i=1}^{t} (s_i^2 w_i - \ln w_i)$$

where

$$w_i = \frac{1}{\sigma_i^2}$$

and

$$\sigma_i^2 = \sigma_e^2 + \frac{c^2}{x_i^4}\sigma_u^2 - \frac{2c}{x_i^2}\rho\sigma_e\sigma_u$$

D.1 ELEMENTS OF THE GRADIENT VECTOR

From equation (2.59) we have

$$\underline{g}(\underline{\theta}) = \left[\frac{\partial\Phi}{\partial\sigma_e^2} \quad \frac{\partial\Phi}{\partial\sigma_u^2} \quad \frac{\partial\Phi}{\partial\rho} \right]^T$$

where

$$\frac{\partial \Phi}{\partial \sigma_e^2} = \sum_{i=1}^{N} (w_i - w_i^2 s_i^2) \frac{\partial \sigma_i^2}{\partial \sigma_e^2}$$

$$\frac{\partial \Phi}{\partial \sigma_u^2} = \sum_{i=1}^{N} (w_i - w_i^2 s_i^2) \frac{\partial \sigma_i^2}{\partial \sigma_u^2}$$

$$\frac{\partial \Phi}{\partial \rho} = \sum_{i=1}^{N} (w_i - w_i^2 s_i^2) \frac{\partial \sigma_i^2}{\partial \rho}$$

and

$$\frac{\partial \sigma_i^2}{\partial \sigma_e^2} = 1 - \frac{c\rho\sigma_u}{x_i^2\sigma_e}$$

$$\frac{\partial \sigma_i^2}{\partial \sigma_u^2} = \frac{c^2}{x_i^4} - \frac{c\rho\sigma_e}{x_i^2\sigma_u}$$

$$\frac{\partial \sigma_i^2}{\partial \rho} = \frac{-2c\sigma_u\sigma_e}{x_i^2}$$

D.2 ELEMENTS OF THE HESSIAN MATRIX

$\underline{H}(\underline{\theta})$ is given in equation (2.60).

Elements of this matrix are :

$$\frac{\partial^2 \Phi}{\partial \sigma_e^4} = \sum_{i=1}^{N} \left\{ \left[(w_i^2 - 2s_i^2 w_i^3) \right] \left[1 - \frac{c\rho\sigma_u}{x_i^2 \sigma_e} \right]^2 + \frac{1}{2} \left[s_i^2 w_i^2 - w_i \right] \left[\frac{c\rho\sigma_u}{x_i^2 \sigma_e^3} \right] \right\}$$

$$\frac{\partial^2 \Phi}{\partial \sigma_u^2 \partial \sigma_e^2} = \sum_{i=1}^{N} \left\{ \left[(w_i^2 - 2s_i^2 w_i^3) \right] \left[\frac{c^2}{x_i^4} - \frac{c\rho\sigma_e}{x_i^2 \sigma_u} \right]^2 - \frac{1}{2} \left[s_i^2 w_i^2 - w_i \right] \left[\frac{c\rho}{\sigma_e \sigma_u x_i^2} \right] \right\}$$

$$\frac{\partial^2 \Phi}{\partial \rho \partial \sigma_e^2} = \sum_{i=1}^{N} \left\{ \left[(w_i^2 - 2s_i^2 w_i^3) \right] \left[\frac{-2c\sigma_u\sigma_e}{x_i^2} \right] \left[1 - \frac{c\rho\sigma_u}{x_i^2 \sigma_e} \right] - \left[s_i^2 w_i^2 - w_i \right] \left[\frac{c\sigma_u}{\sigma_e x_i^2} \right] \right\}$$

$$\frac{\partial^2 \Phi}{\partial \rho \partial \sigma_u^2} = \sum_{i=1}^{N} \left\{ \left[(w_i^2 - 2s_i^2 w_i^3) \right] \left[\frac{-2c\sigma_u\sigma_e}{x_i^2} \right] \left[\frac{c^2}{x_i^4} - \frac{c\rho\sigma_e}{x_i^2 \sigma_u} \right] - \left[s_i^2 w_i^2 - w_i \right] \left[\frac{c\sigma_e}{\sigma_u x_i^2} \right] \right\}$$

$$\frac{\partial^2 \Phi}{\partial \sigma_u^4} = \sum_{i=1}^{N} \left\{ \left[(w_i^2 - 2s_i^2 w_i^3) \right] \left[\frac{c^2}{x_i^4} - \frac{c\rho\sigma_e}{x_i^2 \sigma_u} \right] + \frac{1}{2} \left[s_i^2 w_i^2 - w_i \right] \left[\frac{c\rho\sigma_e}{x_i^2 \sigma_u^3} \right] \right\}$$

$$\frac{\partial^2 \Phi}{\partial \rho^2} = \sum_{i=1}^{N} \left\{ \left[(w_i^2 - 2s_i^2 w_i^3) \right] \left[\frac{-2c\sigma_u\sigma_e}{x_i^2} \right]^2 \right\}$$

APPENDIX E

FORTRAN PROGRAM LISTING FOR NEWTON-RAPHSON COMPONENTS OF VARIANCE ESTIMATION

E.1 MAIN PROGRAM

```
      double precision x(3),f,g(3),w(30),bl(3),bu(3),y(30),s(30)
      common k,y,s
      integer iw(5)
      open(unit=21,file='varcomp.dat',status='old',readonly)
      do 1 i=1,30
      read(21,210,end=98)y(i),s(i)
 1    write(6,699)y(i),s(i)
 699  format(2x,2(1x,f12.6))
 98   k=i-1
 210  format(f6.0,2x,f12.0)
      n=3
      liw=5
      lw=30
      ifail=1
      ibound=0
      bl(1)=1e-6
      bl(2)=1e-6
      bl(3)=-0.99
      bu(1)=1e6
      bu(2)=1e6
      bu(3)=0.99
      write(6,610)
 610  format(//2x,'Enter initial estimates : ')
      write(6,650)
 650  format(//2x,'Sigma-squared V=',$)
      read(5,500)x(1)
      write(6,651)
 651  format(//2x,'Sigma-squared U=',$)
      read(5,500)x(2)
      write(6,652)
 652  format(//2x,'Rho=',$)
      read(5,500)x(3)
 500  format(f12.0)
      call e04laf(n,ibound,bl,bu,x,f,g,iw,liw,w,lw,ifail)
```

```
      if(ifail.ne.0)write(6,600)ifail
      if(ifail.eq.1)go to 99
  600 format(//2x,'Error exit type ',i3,'see NAG documentation')
      write(6,601)f
      write(6,602)(x(j),j=1,n)
      write(6,603)(g(j),j=1,n)
  601 format(//2x,'Function value on exit is ',f12.6)
  602 format(//2x,'at the point ',3f9.4)
  603 format(//2x,'The corresponding gradient is' /15x,3f12.4)
  99  stop
      end
```

E.2 SUBROUTINE FUNCT2 : FUNCTION AND GRADIENT VECTOR EVALUATION

```
      subroutine funct2(n,xc,fc,gc)
      common k,y,s
      double precision gc(n),xc(n),fc,y(30),s(30),w,z,x1,x2,x3,x4
      fc=0
      gc(1)=0
      gc(2)=0
      gc(3)=0
      do 1 i=1,k
      z=y(1)**2
      w=xc(1)+xc(2)*(1800/z)**2-(3600/z)*xc(3)*(xc(1)*xc(2))**0.5
      w=1/w
      fc=fc+s(i)*w-log(w)
      x1=1-xc(3)*1800*((xc(2)/xc(1))**0.5)/z
      x2=(1800/z)**2-xc(3)*1800*((xc(1)/xc(2))**0.5)/z
      x3=-(3600/z)*(xc(1)*xc(2))**0.5
      x4=w-s(i)*w**2
      gc(1)=gc(1)+x1*x4
      gc(2)=gc(2)+x2*x4
      gc(3)=gc(3)+x3*x4
  1   continue
      return
      end
```

E.3 SUBROUTINE HESS2 : EVALUATION OF HESSIAN MATRIX

```
      subroutine hess2(n,xc,heslc,lh,hesdc)
      common k,y,s
      double precision xc(n),heslc(lh),hesdc(n),x1,x2,x3,x4,x5
      double precision x6,x7,x8,x9,x10,x11,y(30),s(30),w,z
      heslc(1)=0
      heslc(2)=0
      heslc(3)=0
      hesdc(1)=0
      hesdc(2)=0
```

```
      hesdc(2)=0
      do 1 i=1,k
      z=y(i)**2
      w=xc(1)+xc(2)*(1800/z)**2-(3600/z)*xc(3)*(xc(1)*xc(2))**0.5
      w=1/w
      x1=2*(w**3)*s(i)-w**2
      x2=w-s(i)*w**2
      x3=(1800/z)**2-((xc(1)/xc(2))**0.5)*1800*xc(3)/z
      x4=1-(1800*xc(3)/z*(xc(2)/xc(1))**0.5)
      x5=-1800*xc(3)/((xc(1)*xc(2)**0.5)*z)
      x6=-(3600/z)*(xc(1)*xc(2))**0.5
      x8=-1800/z*(xc(2)/xc(1))**0.5
      x7=x8
      x9=-1800/z*(xc(1)/xc(2))**0.5
      heslc(1)=heslc(1)+(x1*x3*x4+0.5*x2*x5)
      heslc(2)=heslc(2)+(x1*x6*x4+x2*x7)
      heslc(3)=heslc(3)+(x1*x6*x3+x2*x9)
      x10=-(1800*xc(3)*xc(2)**0.5)/(z*xc(1)**1.5)
      x11=-(1800*xc(3)*xc(1)**0.5)/(z*xc(2)**1.5)
      hesdc(1)=hesdc(1)+(x1*x4**2-0.5*x2*x10)
      hesdc(2)=hesdc(2)+(x1*x3**2-0.5*x2*x11)
      hesdc(3)=hesdc(3)+(x1*x6**2)
1     continue
      return
      end
```

APPENDIX F

DERIVATION OF (Y^*-Y) AND (X^*-X) FOR ORTHOGONAL LEAST-SQUARES ESTIMATION

F.1 $(y_i^* - y_i)$

Using equation (3.25) it can be established that

$$y_i^* = \frac{1}{1 + \sum_{k=1}^{p} \beta_k^2} \left[\beta_1 x_{1i} + \beta_2 x_{2i} + \ldots + \beta_p x_{pi} + \sum_{j=1}^{p} \beta_j^2 \right]$$

and thus

$$(y_i^* - y_i) = \frac{\left[\sum_{j=1}^{p} \beta_j x_{ji} + y_i \sum_{j=1}^{p} \beta_j^2 \right] - \left[1 + \sum_{j=1}^{p} \beta_j^2 \right] y_i}{1 + \sum_{k=1}^{p} \beta_k^2}$$

$$= \frac{\sum_{j=1}^{p} \beta_j x_{ji} - y_i}{1 + \sum_{k=1}^{p} \beta_k^2}$$

F.2 $(x_{ji}^* - x_{ji})$

Again using equation (3.25) we have that the projection of x_{ji} is x_{ji}^* where :

$$x_{ji}^* = \frac{1}{1 + \sum_{k=1}^{p} \beta_k^2} \left[1 + \beta_1^2 + \beta_2^2 + \ldots + \beta_{j-1}^2 + \beta_{j+1}^2 + \beta_p^2 - \sum_{\substack{k=1 \\ k \neq j}}^{p} \beta_k \beta_j x_k \right]$$

$$= \frac{1}{1 + \sum_{k=1}^{p} \beta_k^2} \left[\left[1 + \sum_{k=1}^{p} \beta_k^2 \right] x_{ji} - \sum_{k=1}^{p} \beta_k \beta_j x_{ki} \right]$$

and

$$(x_{ji}^* - x_{ji}) = \frac{\left[1 + \sum_{k=1}^{p} \beta_k^2 \right] x_{ji} - \sum_{k=1}^{p} \beta_j \beta_k x_{ki} - x_{ji} \left[1 + \sum_{k=1}^{p} \beta_k^2 \right]}{1 + \sum_{k=1}^{p} \beta_k^2}$$

thus

$$(x_{ji}^* - x_{ji}) = \frac{- \sum_{k=1}^{p} \beta_j \beta_k x_{ki}}{1 + \sum_{k=1}^{p} \beta_k^2}$$

APPENDIX G

DERIVATION OF ORTHOGONAL LEAST-SQUARES ESTIMATES

The orthogonal least-squares criterion of equation(3.28) is given as :

$$\mathbb{Z} = \sum_{i=1}^{n} \left[\left[\frac{\sum_{j=1}^{p} \beta_j x_{ji} - y_i}{1 + \sum_{k=1}^{p} \beta_k^2} \right]^2 + \left[\frac{-\sum_{k=1}^{p} \beta_j \beta_k x_{ki}}{1 + \sum_{k=1}^{p} \beta_k^2} \right]^2 \right]$$

We differentiate $\mathbb{Z}$ with respect to β_j and set the result to zero. Thus :

$$\frac{\partial \mathbb{Z}}{\partial \beta_j} = 0 \Rightarrow$$

$$\sum_{i=1}^{n} \left[\frac{\partial}{\partial \beta_j} \left[\frac{\sum_{j=1}^{p} \beta_j x_{ji} - y_i}{1 + \sum_{k=1}^{p} \beta_k^2} \right]^2 + \frac{\partial}{\partial \beta_j} \sum_{j=1}^{p} \left[\frac{-\sum_{k=1}^{p} \beta_j \beta_k x_{ki}}{1 + \sum_{k=1}^{p} \beta_k^2} \right]^2 \right] = 0$$

$$\Rightarrow \sum_{i=1}^{n}\left[2\left[\frac{\sum_{j=1}^{p}\beta_j x_{ji} - y_i}{1+\sum_{k=1}^{p}\beta_k^2}\right]\left[\frac{x_{ji}\left[1+\sum_{k=1}^{p}\beta_k^2\right] - 2\beta_j\left[\sum_{k=1}^{p}\beta_k x_{ki} - y_i\right]}{\left[1+\sum_{k=1}^{p}\beta_k^2\right]^2}\right]\right.$$

$$+ 2\sum_{\substack{m=1\\ m\neq j}}^{p}\left[\frac{\sum_{k=1}^{p}\beta_m\beta_k x_{ki}}{1+\sum_{k=1}^{p}\beta_k^2}\right]\frac{\partial}{\partial\beta_j}\left[\frac{\beta_m\beta_k x_{ki}}{1+\sum_{k=1}^{p}\beta_k^2}\right]$$

$$\left. + 2\left[\frac{\sum_{k=1}^{p}\beta_j\beta_k x_{ki}}{1+\sum_{k=1}^{p}\beta_k^2}\right]\left[\sum_{\substack{k=1\\ k\neq j}}^{p}\frac{\partial}{\partial\beta_j}\left[\frac{\beta_m\beta_k x_{ki}}{1+\sum_{k=1}^{p}\beta_k^2}\right] + \frac{\partial}{\partial\beta_j}\left[\frac{\beta_j^2\, x_{ji}}{1+\sum_{k=1}^{p}\beta_k^2}\right]\right]\right]$$

$$= 0$$

After some algebraic manipulation we obtain

$$\frac{\partial Z}{\partial\beta_j} = 0 \quad \Rightarrow$$

$$\sum_{=1}^{n}\left[2\left[\frac{\sum_{k=1}^{p}\beta_k x_{ki} - y_i}{1+\sum_{k=1}^{p}\beta_k^2}\right]\left[\frac{x_{ji}\left[1+\sum_{k=1}^{p}\beta_k^2\right] - 2\beta_j\left[\sum_{k=1}^{p}\beta_k x_{ki} - y_i\right]}{\left[1+\sum_{k=1}^{p}\beta_k^2\right]^2}\right]\right]$$

$$+ 2\left[\frac{\sum_{k=1}^{p}\beta_j\beta_k x_{ki}}{1+\sum_{k=1}^{p}\beta_k^2}\right]\left[\sum_{\substack{k=1\\k\neq j}}^{p}\frac{\beta_k x_{ki}\left[1+\sum_{k=1}^{p}\beta_k^2\right] - 2\beta_j^2\beta_k\, x_{ki}}{\left[1+\sum_{k=1}^{p}\beta_k^2\right]^2}\right] = 0$$

$$\Rightarrow \sum_{i=1}^{n}\left[\left[\sum_{k=1}^{p}\beta_k x_{ki} - y_i\right]\left[x_{ji}\left[1+\sum_{k=1}^{p}\beta_k^2\right] - 2\beta_j\left[\sum_{k=1}^{p}\beta_k x_{ki} - y_i\right]\right]\right.$$

$$\left. + \left[\sum_{k=1}^{p}\beta_j\beta_k x_{ki}\right]\left[\sum_{\substack{k=1\\k\neq j}}^{p}\beta_k x_{ki}\left[1+\sum_{k=1}^{p}\beta_k^2\right] - 2\beta_j\beta_k x_{ki}\right]\right] = 0$$

For simplicity, let $\lambda = 1 + \sum_{k=1}^{p}\beta_k^2$ and $\hat{y}_i = \sum_{k=1}^{p}\beta_k x_{ki}$, thus the last expression can be written as :

$$\sum_{i=1}^{n}\left[[\hat{y}_i - y_i]\left[\lambda x_{ji} - 2\beta_j[\hat{y}_i - y_i]\right] + \beta_j\hat{y}_i\left[\sum_{\substack{k=1\\k\neq j}}^{p}\beta_k x_{ki}\ [\lambda - 2\beta_j^2]\right]\right] = 0$$

After further algebraic manipulation this reduces to

$$2\beta_j^4\hat{y}_i x_{ji} - 2\beta_j^3\hat{y}_i^2 - \lambda\beta_j^2\hat{y}_i x_{ji} + \beta_j[\lambda\hat{y}_i^2 - 2(y_i^2 + \hat{y}_i^2)] + \lambda x_{ji}(\hat{y}_i - y_i)$$

which is a quartic in β_j . This has no closed-form solution since the coefficients of the β_j terms involve $\hat{y}_i$'s which themselves depend on β_j.

APPENDIX H

PROGRAM LISTING FOR MULTIVARIATE ORTHOGONAL PARAMETER ESTIMATION

H.1 MAIN PROGRAM

```
cls;
count=1;
pcount=1;
"                Orthogonal Least-Squares for Multivariate Calibration";
"                =====================================================";
print;
"                                 David R. Fox";
print;
"                            University of Wyoming";
"                                February 1989.";
print;
print;
"  Input requirements : ";
"  =================";
print;
"                    Matrix Y (nxp) of dependent variable values";
print;
"                    Matrix X (nxq) of independent variable values";
print;
print;
"  Output : ";
"  ======";
print;
"                    Matrix B (qxp) of paramter estimates which minimize
"                    generalized distance.";
print;
dos pause;
cls;
print;
"       Enter epsilon for convergence : ";;
epsilon=con(1,1);
cls;
/*      Determine p , q and n      */;
p=cols(y);
q=cols(x);
n=rows(y);
borthog=zeros(q,p);
```

```
/*      Obtain OLS estimate for b  */;
b=inv(x'x)*x'y;
do while pcount<=p;
/*      Extract current components of Y and B   */;
yc=y[.,pcount];
bc=b[.,pcount];
/*      Perform Newton-Raphson iteration
        Note : Projection matrix is computed in Proc calc */;
kk=1;
inc=999999;
format /ld 4,0;
do while inc>epsilon;
h=hessp(&calc,bc);
g=gradp(&calc,bc);
delta=inv(h)*g';
inc=delta'delta;
bc=bc-delta;
"Iteration # ";;kk;
kk=kk+1;
format /rd 12,6;
bc;
endo;
print;
format /ld 4,0;
"     Convergence established at iteration ";;kk;;" for column ";;pcount
print;
borthog[.,pcount]=bc;
pcount=pcount+1;
endo;
print;
"                                 DONE !! ";
print;
"    Orthogonal matrix is :";
format /rd 12,6;
borthog;
print;
"    OLS matrix is : ";
b;
stop;
```

H.2 SUBROUTINE FOR CALCULATION OF ELEMENTS OF HESSIAN MATRIX.

```
proc calc(bc);
local w;
/*      Form matrix Q   */;
Qc=eye(q)-inv(1+bc'bc)*bc*bc';
/*      Form matrix Z   */;
Zc=x~yc;
/*      Construct projection matrix P   */;
```

```
P2=Qc*bc;
P4=bc'qc*bc;
Pc=(Qc~P2)|(P2'~P4);
/*     Obtain X and Y projections , store into xp and yp    */;
Zp=zc*Pc;
xp=zp[.,1:q];
yp=zp[.,q+1];
w=(yp-yc)'(yp-yc)+sumc(diag((xp-x)'(xp-x)));
retp(w);
endp;
```

APPENDIX I

PROGRAM LISTING FOR MULTIVARIATE BIAS/MSE SIMULATION

I.1 MAIN PROGRAM

```
cls;
count=1;
pcount=1;
"                Simulation of multivariate calibration estimators";
"                =================================================";
print;
"                                 David R. Fox";
print;
"                            University of Wyoming";
"                                February 1989.";
print;
print;
print;
print;
"        This program simulates multivariate X and Y data having";
"        a specified covariance structure. X is 2-dimensional and";
"        Y is 4-dimensional.";
print;
print;
/* "          *     means and variances for X1 and X2";
"          *     coefficients of variation for Y1,Y2,Y3,and Y4.";
"          *     correlation between errors in X1 and X2";
"          *     correlation between errors in Y's";
"          *     correlation between errors in X's and errors in Y's";
"          *     measurement error variances for X1 and X2";
print;
dos pause;
cls;
"                                  DATA ENTRY";
"                                  ==========";
print;
print; */;
dos pause;
loop1:;
dos e:input;
cls;
load data[15,1]=e:mvc.dat;
```

```
load beta[2,4]=e:beta.dat;
rhox=data[11,1];rhoy=data[12,1];rhoxy=data[13,1];nsamp=data[14,1];
nsim=data[15,1];
cv=data[7:10,1];varx=data[3:4,1];mux=data[1:2,1];varu=data[5:6,1];
print;
print;
print;
print;
"             Do you want to change any of the input data (y/n) ? : ";;
reply=cons;
if reply $=="Y" or reply $=="y";
goto loop1;
endif;
ncount=1;
biasi=zeros(nsim,2);
biasc=zeros(nsim,2);
biaso=zeros(nsim,2);
do while ncount<=nsim;
print;
format /ld 3,0;
"  Simulation run # ";;ncount;
format /rd 9,4;
/*                       construct sigmax        */;
s12=rhox*(varx[1,1]*varx[2,1])^.5;
sigmax=varx[1,1]~s12|s12~varx[2,1];
/*                       spectral decomposition of sigmax        */;
{va,p}=eigrs2(sigmax);
va=va^.5;
d1=eye(2);
d1=diagrv(d1,va);
a=p*d1;
/*                       generate (n+1) observations from N2(0,1)
z1=rndn(nsamp+1,2);
/*                       compute matrix of xs    */;
m1=ones(nsamp+1,1)*mux[1,1];
m2=ones(nsamp+1,1)*mux[2,1];
m=m1~m2;
x=z1*a+m;
/*                       compute muY     */;
muy=mux'*beta;
/*                       assemble sigmae         */;
sigmae=zeros(4,4);
i=1;
j=1;
do while i<=4;
j=1;
do while j<=4;
if i==j;
sigmae[i,i]=(cv[i,1]*muy[1,i])^2;
else;
sigmae[i,j]=rhoy*cv[i,1]*muy[1,i]*cv[j,1]*muy[1,j];
endif;
j=j+1;
```

```
endo;
i=i+1;
endo;
/*                      assemble sigmaeu        */;
sigmaeu=zeros(4,2);
i=1;
j=1;
do while i<=4;
j=1;
do while j<=2;
sigmaeu[i,j]=rhoxy*(cv[i,1]*muy[1,i]*varx[j,1])^.5;
j=j+1;
endo;
i=i+1;
endo;
/*                      assemble sigmau         */;
s12=rhox*(varu[1,1]*varu[2,1])^.5;
sigmau=varu[1,1]~s12|s12~varu[2,1];
/*                      assemble sigma          */;
sigma=sigmae~sigmaeu|sigmaeu'~sigmau;
/*                      spectral decomposition of sigma       */;
{var,vai,p,vei}=eigrg2(sigma);
k=1;
do while k<=6;
if var[k,1]<0;
var[k,1]=0;
endif;
k=k+1;
endo;
var=var^.5;
d2=eye(6);
d2=diagrv(d2,var);
b=p*d2;
/*                      generate (n+1) observations from N6(0,1)
z2=rndn(nsamp+1,6);
er=z2*b;
e=er[.,1:4];
u=er[.,5:6];
capx=x+u;
capy=x*beta+e;
x0=x[nsamp+1,.];
y0=capy[nsamp+1,.];
capx=capx[1:nsamp,.];
capy=capy[1:nsamp,.];
/*                  <<<<<<<<  PARAMETER ESTIMATION  >>>>>>>
/*

*/;
/*                              Inverse estimation      */;
ghat=inv(capy'capy)*capy'capx;
biasi[ncount,.]=x0-y0*ghat;
/*                              Classical estimation    */;
bhat=inv(capx'capx)*capx'capy;
```

```
resid=capy-capx*bhat;
shat=resid'resid/nsamp;
shat=inv(shat);
biasc[ncount,.]=x0-(inv(bhat*shat*bhat')*bhat*shat*y0')';
/*                              Orthogonal estimator    */;
borthog=orthog;
resid=capy-capx*borthog;
shat=resid'resid/nsamp;
shat=inv(shat);
biaso[ncount,.]=x0-(inv(borthog*shat*borthog')*borthog*shat*y0')';
ncount=ncount+1;
endo;
format /rd 9,4;
stop;
```

I.2 INITIALIZATION SUBROUTINE.

```
capx=zeros(nsamp+1,2);
capy=zeros(nsamp+1,4);
bhat=zeros(2,4);
bc=zeros(2,1);
p=4;
q=2;
qc=zeros(q,q);
zc=zeros(nsamp,q+1);
p2=zeros(q,1);
p4=zeros(1,1);
pc=zeros(q+1,q+1);
zp=zc;
xp=zeros(nsamp,q);
yp=zeros(nsamp,1);
yc=zeros(nsamp,1);
run e:mvccalc.arc;
run gradp.g;
run hessp.g;
```

I.3 ITERATIVE PROCEDURE FOR CALCULATION OF ORTHOGONAL ESTIMATES.

```
proc orthog;
local n,kk,pcount,inc,h,g,delta,borthog,newstart,mss;
newstart=0;
pcount=1;
/*      Determine p , q and n      */;
p=cols(capy);
q=cols(capx);
n=rows(capy);
borthog=zeros(q,p);
```

```
format /ld 2,0;
do while pcount<=p;
"       Orthogonal estimate : Iterating on column ";;pcount;;
/*      Extract current components of Y and B   */;
yc=capy[.,pcount];
bc=bhat[.,pcount];
/*      Perform Newton-Raphson iteration
        Note : Projection matrix is computed in Proc calc */;
kk=1;
inc=999999;
do while inc>.00001;
if kk>15;
if newstart>4;
print;
" No convergence after 5 restarts - giving up !!";
borthog=zeros(q,p);
mss=zeros(1,p);
borthog=miss(borthog,mss);
retp(borthog);
endif;
print;
" Convergence not obtained after 15 iterations - trying new initial esti
newstart=newstart+1;
bc=rndu(q,1)*10;
kk=1;
endif;
h=hessp(&calc,bc);
g=gradp(&calc,bc);
delta=inv(h)*g';
inc=delta'delta;
bc=bc-delta;
kk=kk+1;
endo;
"(";;kk;;")";
borthog[.,pcount]=bc;
pcount=pcount+1;
endo;
retp(borthog);
endp;
```

I.4 SUBROUTINE FOR GRADIENT AND HESSIAN ELEMENTS.

```
proc calc(bc);
local w;
/*      Form matrix Q   */;
Qc=eye(q)-inv(1+bc'bc)*bc*bc';
/*      Form matrix Z   */;
Zc=capx~yc;
/*      Construct projection matrix P   */;
P2=Qc*bc;
```

```
P4=bc'qc*bc;
Pc=(Qc~P2)|(P2'~P4);
/*      Obtain X and Y projections , store into xp and yp    */;
Zp=zc*Pc;
xp=zp[.,1:q];
yp=zp[.,q+1];
w=(yp-yc)'(yp-yc)+sumc(diag((xp-capx)'(xp-capx)));
retp(w);
endp;
```

I.5 DBASEIII SUBROUTINE FOR DATA ENTRY.

```
set talk off
set exact on
set confirm on
clear memory
store chr(7) to bell
store '' to fn
clear
use e:mvc.dbf
@1,1 say date()
@1,20 say 'MULTIVARIATE CALIBRATION SIMULATION'
@1,70 say 'DATA ENTRY'
@2,20 say '==================================='
@2,72 say time()
@5,10 say 'The following information is required :'
@7,5 say '*    Mean of X1'
@8,5 say '*    Variance of X1'
@9,5 say '*    Mean of X2'
@10,5 say '*    Variance of X2'
@11,5 say '*    Measurement error variance for X1'
@12,5 say '*    Measurement error variance for X2'
@13,5 say '*    Coefficients of variation for Y1,Y2,Y3, and Y4'
@14,5 say '*    Correlation between Xs'
@15,5 say '*    Correlation between Ys'
@16,5 say '*    Corrrelation between Xs and Ys'
@22,25 say 'Press Q to exit procedure'
wait '                       Press any other key to continue'to continue
if continue="Q".or.continue="q"
quit
endif
store '' to fn
store .f. to flag3
clear
@1,1 say date()
@1,20 say 'MULTIVARIATE CALIBRATION SIMULATION'
@1,70 say 'DATA ENTRY'
@2,20 say '==================================='
@2,72 say time()
@3,5 say 'Mean X1 : ' get mux1
```

```
@3,42 say 'Mean X2 : ' get mux2
@5,1 say 'Variance X1 : ' get varx1
@5,38 say 'Variance X2 : ' get varx2
@7,25 say 'Measurement-error variances :'
@8,25 say '----------------------------'
@10,12 say 'For X1 : ' get varu1
@10,45 say 'For X2 : ' get varu2
@12,25 say 'Coefficients of Variation : '
@13,25 say '---------------------------'
@15,15 say 'For Y1 : ' get cy1
@15,45 say 'For Y2 : ' get cy2
@17,15 say 'For Y3 : ' get cy3
@17,45 say 'For Y4 : ' get cy4
@19,15 say 'Correlation coefficients for error components :'
@20,15 say '-----------------------------------------------'
@21,5 say 'Between Xs : ' get rhox
@21,27 say 'Between Ys : ' get rhoy
@21,50 say 'Between Xs and Ys : ' get rhoxy
@23,10 say 'Sample size : ' get nsamp
@23,40 say 'Length of simulation run : ' get nsim
read
copy                    to                    e:mvc.dat                    fields
mux1,mux2,varx1,varx2,varu1,varu2,cy1,cy2,cy3,cy4,rhox,rhoy,rhoxy,nsamp,
clear
go top
@1,1 say date()
@1,20 say 'MULTIVARIATE CALIBRATION SIMULATION'
@1,70 say 'DATA ENTRY'
@2,20 say '==================================='
@2,72 say time()
@5,25 say 'ENTER ELEMENTS OF B MATRIX'
@6,25 say '=========================='
@10,5  say 'B11 : ' get b11
@10,23 say 'B12 : ' get b12
@10,43 say 'B13 : ' get b13
@10,63 say 'B14 : ' get b14
@12,5  say 'B21 : ' get b21
@12,23 say 'B22 : ' get b22
@12,43 say 'B23 : ' get b23
@12,63 say 'B24 : ' get b24
read
copy to e:beta.dat fields b11,b12,b13,b14,b21,b22,b23,b24 sdf
```

APPENDIX J

DERIVATION OF CONDITIONAL MULTIVARIATE NORMAL P.D.F.

Let **X** and **Y** be two random vectors and $\begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix}$ be multivariate normally distributed with mean vector

$$\mathbb{E}\begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix} = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}$$

and non-singular covariance matrix

$$\mathrm{Cov}\begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} \\ C_{12}^{T} & C_{22} \end{bmatrix} .$$

Define $\tilde{\mathbf{Y}}$ as the random vector **Y** given $\mathbf{X} = \mathbf{x}$, where **x** is a vector of deterministic scalars. Then $\tilde{\mathbf{Y}}$ has the multivariate normal distribution with mean

$$\mathbb{E}[\tilde{\mathbf{Y}}] = C_{12}^{T}C_{11}^{-1}(\mathbf{x} - \mu_x) + \mu_y$$

and covariance matrix

$$\mathrm{Cov}[\tilde{\mathbf{Y}}] = C_{22} - C_{12}^{T}C_{11}^{-1}C_{12} \quad .$$

<u>Proof</u> :

$$f_{X,Y}(x,y) = \frac{\exp\left[-\frac{1}{2}(X^T-\mu_x^T\ ,Y^T-\mu_y^T)\begin{bmatrix} C_{11} & C_{12} \\ C_{12}^T & C_{22}\end{bmatrix}^{-1}\begin{bmatrix} X-\mu_x \\ Y-\mu_y\end{bmatrix}\right]}{(2\pi)^{\frac{\nu_1+\nu_2}{2}}\begin{vmatrix} C_{11} & C_{12} \\ C_{12}^T & C_{22}\end{vmatrix}^{\frac{1}{2}}}$$

where

ν_1 is the dimension of X

and ν_2 is the dimension of Y .

The marginal density for X is

$$f_X(x) = \frac{\exp\left[-\frac{1}{2}(X^T-\mu_x^T\ )\ C_{11}^{-1}\ (X-\mu_x\ )\right]}{(2\pi)^{\nu_1/2}\ \left|C_{11}\right|^{\frac{1}{2}}}$$

The conditional density for $\tilde{Y} = (Y \mid X = x)$ is :

$$f_{\tilde{Y}}(y) = f_{Y|X=x}(y) = f_{X,Y}(x,y)/f_X(x)$$

$$= \kappa \exp\left[\frac{1}{2}(x-\mu_x)^T C_{11}^{-1}(x-\mu_x)-\frac{1}{2}(x-\mu_x\ ,y-\mu_y)^T\begin{bmatrix} M_{11} & M_{12} \\ M_{12}^T & M_{22}\end{bmatrix}\begin{bmatrix} x-\mu_x \\ y-\mu_y\end{bmatrix}\right]$$

where

$$\kappa = \frac{\left|C_{11}\right|^{\frac{1}{2}}}{(2\pi)^{\nu_2/2}\begin{vmatrix} C_{11} & C_{12} \\ C_{12}^T & C_{22}\end{vmatrix}^{\frac{1}{2}}}$$

and

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{12}^{T} & M_{22} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} \\ C_{12}^{T} & C_{22} \end{bmatrix}^{-1} .$$

Furthermore, it can be shown that

$$M_{11} = (C_{11} - C_{12}C_{22}^{-1}C_{12}^{T})^{-1} ;$$
$$M_{22} = (C_{22} - C_{12}^{T}C_{11}^{-1}C_{12})^{-1} ;$$
$$M_{12} = -C_{11}^{-1}C_{12}M_{22}$$
$$= -C_{11}^{-1}C_{12}(C_{22} - C_{12}^{T}C_{11}^{-1}C_{12})^{-1} .$$

Thus $f_{\tilde{Y}}(y) =$

$$\kappa \exp\left[\tfrac{1}{2}(x - \mu_x)^{T}(C_{11}^{-1} - M_{11})(x - \mu_x)\right]\exp\left[-\tfrac{1}{2}\left[(y - \mu_y)^{T}M_{22}(y - \mu_y) + 2(x - \mu_x)^{T}M_{12}(y - \mu_y)\right]\right]$$

The last exponent of the previous expression can be written as

$$(y - \mu_y)^{T}M_{22}(y - \mu_y) + 2(x - \mu_x)^{T}M_{12}(y - \mu_y) =$$

$$(y - \mu_y - m)^{T}M_{22}(y - \mu_y - m) - (x - \mu_x)^{T}M_{12}M_{22}^{-1}M_{12}^{T}(x - \mu_x)$$

where

$$m = -M_{22}^{-1}M_{12}^{T} (x - \mu_x) .$$

After some algebraic manipulation we obtain

$$f_{\tilde{Y}}(y) = \kappa \exp[(y - \mu_y - m)^T M_{22}(y - \mu_y - m)] \quad .$$

This is the multivariate normal density with mean vector $\mu_y + m$ and covariance matrix M_{22}^{-1} .

Thus :

$$\mathbb{E}[\tilde{Y}] = \mu_y + m = \mu_y - M_{22}^{-1}M_{12}^T(x - \mu_x)$$

$$= \mu_y - (C_{22} - C_{12}^T C_{11}^{-1} C_{12})[-C_{11}^{-1}C_{12}(C_{22} - C_{12}^T C_{11}^{-1}C_{12})^{-1}]^T (x - \mu_x)$$
$$= \mu_y + (C_{22} - C_{12}^T C_{11}^{-1} C_{12})(C_{22} - C_{12}^T C_{11}^{-1}C_{12})^{-1} C_{12}^T C_{11}^{-1}(x - \mu_x)$$
$$= \mu_y + C_{12}^T C_{11}^{-1}(x - \mu_x) \quad .$$

and

$$\text{Cov}[\tilde{Y}] = M_{22}^{-1} = C_{22} - C_{12}^T C_{11}^{-1} C_{12} \qquad \text{QED.}$$

APPENDIX K

DELETION STATISTICS : REVISED PARAMETER AND COVARIANCE ESTIMATES

K.1 REVISED PARAMETER ESTIMATES

The derivation of equation (3.61) is as follows :

Without loss of generality, we can partition X as $\left[\dfrac{X_{(I)}}{X_I}\right]$ where $X_{(I)}$ is a (n-m) x p matrix and X_I is m x p.

Thus

$$X^TX = [X_{(I)}^T \; X_I^T]\left[\dfrac{X_{(I)}}{X_I}\right] = X_{(I)}^TX_{(I)} + X_I^TX_I$$

and

$$X_{(I)}^TX_{(I)} = X^TX - X_I^TX_I$$

$$\Rightarrow \qquad [X_{(I)}^TX_{(I)}]^{-1} = [X^TX - X_I^TX_I]^{-1}$$

Using the fact that $(A - UV^T)^{-1} = A^{-1} + A^{-1}U(I - V^TA^{-1}U)^{-1}V^TA^{-1}$ with $A = (X^TX)$, $U = X_I^T$ and $V = X_I$ we have

$$[X^TX - X_I^TX_I]^{-1} = (X^TX)^{-1} + (X^TX)^{-1}X_I^T[I - X_I(X^TX)^{-1}X_I^T]^{-1}X_I(X^TX)^{-1} .$$

We identify the hat matrix : $H_I = X_I(X^TX)^{-1}X_I^T$ and therfore

$$[X_{(I)}^T X_{(I)}]^{-1} = (X^TX)^{-1} + (X^TX)^{-1}X_I^T[I - H_I]^{-1}X_I(X^TX)^{-1} .$$

Now, the matrix of revised parameter estimates $\hat{\beta}_{(I)}$ is

$$\hat{\beta}_{(I)} = [X_{(I)}^T X_{(I)}]^{-1}X_{(I)}^T Y_{(I)}$$

Furthermore

$$X = \begin{bmatrix} X_{(I)} \\ \hline X_I \end{bmatrix} \quad \text{and} \quad Y = \begin{bmatrix} Y_{(I)} \\ \hline Y_I \end{bmatrix}$$

thus

$$X^TY = [X_{(I)}^T \; X_I^T]\begin{bmatrix} Y_{(I)} \\ \hline Y_I \end{bmatrix} = X_{(I)}^T Y_{(I)} + X_I^T Y_I$$

$$\Rightarrow \qquad X_{(I)}^T Y_{(I)} = X^TY - X_I^TY_I .$$

Hence

$$\begin{aligned}
\hat{\beta}_{(I)} &= [X_{(I)}^T X_{(I)}]^{-1}[X^TY - X_I^TY_I] \\
&=\{(X^TX)^{-1} + (X^TX)^{-1}X_I^T[I - H_I]^{-1}X_I(X^TX)^{-1}\}[X^TY - X_I^TY_I] \\
&= (X^TX)^{-1}X^TY + (X^TX)^{-1}X_I^T[I - H_I]^{-1}X_I(X^TX)^{-1}X^TY - (X^TX)^{-1}X_I^TY_I \\
&\qquad - (X^TX)^{-1}X_I^T[I - H_I]^{-1}X_I(X^TX)^{-1}X_I^TY_I \\
&= \hat{\beta} + (X^TX)^{-1}X_I^T[I - H_I]^{-1}X_I\hat{\beta} - (X^TX)^{-1}X_I^TY_I \\
&\qquad - (X^TX)^{-1}X_I^T[I - H_I]^{-1}X_I(X^TX)^{-1}X_I^TY_I \\
&= \hat{\beta} + (X^TX)^{-1}X_I^T[I - H_I]^{-1}\hat{Y}_I - (X^TX)^{-1}X_I^TY_I - (X^TX)^{-1}X_I^T[I - H_I]^{-1}H_IY_I
\end{aligned}$$

Thus
$$\begin{aligned}
\hat{\beta}_{(I)} - \hat{\beta} &= (X^TX)^{-1}X_I^T\{[I - H_I]^{-1}\hat{Y}_I - Y_I - [I - H_I]^{-1}H_IY_I\} \\
&= (X^TX)^{-1}X_I^T\{-Y_I - [I - H_I]^{-1}H_IY_I + [I - H_I]^{-1}\hat{Y}_I\}
\end{aligned}$$

Now, $-Y_I - [I - H_I]^{-1}H_IY_I = -[I + (I - H_I)^{-1}H_I]Y_I$.

Next consider the term $I + (I - H_I)^{-1}H_I$ and again use $(A - UV^T)^{-1} = A^{-1} + A^{-1}U(I - V^TA^{-1}U)^{-1}V^TA^{-1}$ with $A = I$; $U = I$; and $V^T = H_I$ thus

$$(I - H_I)^{-1} = I + (I - H_I)^{-1}H_I$$

and therefore

$$-Y_I - [I - H_I]^{-1}H_IY_I = -(I - H_I)^{-1}Y_I$$

and so

$$\hat{\beta}_{(I)} - \hat{\beta} = (X^TX)^{-1}X_I^T\{-(I - H_I)^{-1}Y_I + (I - H_I)^{-1}\hat{Y}_I\}$$
$$= (X^TX)^{-1}X_I^T(I - H_I)^{-1}[\hat{Y}_I - Y_I]$$

Letting $R_I = \hat{Y}_I - Y_I$ we thus obtain

$$\hat{\beta}_{(I)} - \hat{\beta} = (X^TX)^{-1}X_I^T(I - H_I)^{-1}R_I$$

K.2 REVISED RESIDUAL SUM OF SQUARES

Prior to deletion our estimate of the error-covariance matrix Σ is

$$(n-p)\hat{\Sigma} = Y^TY - \hat{\beta}^TX^TY$$

After deletion of the i^{th} observation this becomes :

$$(n-m-p)\hat{\Sigma}_{(I)} = Y^TY - Y_I^TY_I - \hat{\beta}_{(I)}^T(X^TY - X_I^TY_I)$$
$$\Rightarrow \quad (n-m-p)\hat{\Sigma}_{(I)} = (n-p)\hat{\Sigma} + \hat{\beta}\, X^TY - Y_I^TY_I - \hat{\beta}_{(I)}^T(X^TY - X_I^TY_I)$$
$$= (n-p)\hat{\Sigma} - \{R_I^T(I - H_I)^{-1}X_I(X^TX)^{-1}\}X^TY - Y_I^TY_I + \hat{\beta}_{(I)}^TX_I^TY_I$$

$$= (n-p)\hat{\Sigma} - R_I^T(I - H_I)^{-1}X_I\hat{\beta} - Y_I^TY_I + \hat{\beta}_{(I)}^TX_I^TY_I$$
$$= (n-p)\hat{\Sigma} - R_I^T(I - H_I)^{-1}\hat{Y}_I - Y_I^TY_I + \hat{\beta}_{(I)}^TX_I^TY_I$$
$$= (n-p)\hat{\Sigma} - R_I^T(I - H_I)^{-1}\hat{Y}_I - \{Y_I^TY_I - \hat{\beta}_{(I)}^TX_I^TY_I\}$$

Consider the expression inside the braces {} :

$$Y_I^TY_I - \hat{\beta}_{(I)}^TX_I^TY_I = (Y_I^T - \hat{\beta}_{(I)}^TX_I^T)Y_I = (Y_I - X_I\hat{\beta}_{(I)})^TY_I$$

Now,

$$Y_I - X_I\hat{\beta}_{(I)} = Y_I - X_I\{[X_{(I)}^TX_{(I)}]^{-1}[X^TY - X_I^TY_I]\}$$
$$=Y_I - X_I\{[(X^TX)^{-1} + (X^TX)^{-1}X_I^T(I - H_I)^{-1}X_I(X^TX)^{-1}](X^TY - X_I^TY_I)\}$$
$$=Y_I - \{[X_I(X^TX)^{-1} + X_I(X^TX)^{-1}X_I^T(I - H_I)^{-1}X_I(X^TX)^{-1}](X^TY - X_I^TY_I)\}$$
$$=Y_I - \{X_I(X^TX)^{-1}X^TY + X_I(X^TX)^{-1}X_I^T(I - H_I)^{-1}X_I(X^TX)^{-1}X^TY$$
$$- X_I(X^TX)^{-1}X_I^TY_I - X_I(X^TX)^{-1}X_I^T(I - H_I)^{-1}X_I(X^TX)^{-1}X_I^TY_I\}$$
$$=Y_I - X_I(X^TX)^{-1}X^TY - X_I(X^TX)^{-1}X_I^T(I - H_I)^{-1}X_I(X^TX)^{-1}X^TY$$
$$+ X_I(X^TX)^{-1}X_I^TY_I + X_I(X^TX)^{-1}X_I^T(I - H_I)^{-1}X_I(X^TX)^{-1}X_I^TY_I$$

Recall that $H_I = X_I(X^TX)^{-1}X_I^T$; $R_I = \hat{Y}_I - Y_I$; and $\hat{Y}_I = X_I\hat{\beta}$, thus the right hand side of the previous expression becomes

$$=Y_I - \hat{Y}_I - H_I(I - H_I)^{-1}\hat{Y}_I + H_IY_I + H_I(I - H_I)^{-1}H_IY_I$$

Now,

$$H_IY_I + H_I(I - H_I)^{-1}H_IY_I =[I + H_I(I - H_I)^{-1}]H_IY_I$$

and since both H_I and $(I - H_I)^{-1}$ are symmetric their product is commutative and hence

$$H_I Y_I + H_I (I - H_I)^{-1} H_I Y_I = [I + (I - H_I)^{-1} H_I] H_I Y_I$$

Finally,

$$\begin{aligned}
&Y_I - \hat{Y}_I - H_I (I - H_I)^{-1} \hat{Y}_I + (I - H_I)^{-1} H_I Y_I \\
&= Y_I - \hat{Y}_I + (I - H_I)^{-1} H_I \{Y_I - \hat{Y}_I\} \\
&= Y_I - \hat{Y}_I \{I + (I - H_I)^{-1} H_I\} \\
&= Y_I - \hat{Y}_I (I - H_I)^{-1} \\
&= -R_I (I - H_I)^{-1}
\end{aligned}$$

Therefore :

$$\begin{aligned}
Y_I^T Y_I - \hat{\beta}_{(I)}^T X_I^T Y_I &= (Y_I - X_I \hat{\beta}_{(I)})^T Y_I \\
&= -[R_I (I - H_I)^{-1}]^T Y_I \\
&= -(I - H_I)^{-1} R_I^T Y_I
\end{aligned}$$

and

$$\begin{aligned}
(n-m-p)\hat{\Sigma}_{(I)} &= (n-p)\hat{\Sigma} - R_I^T (I - H_I)^{-1} \hat{Y}_I + R_I^T (I - H_I)^{-1} Y_I \\
&= (n-p)\hat{\Sigma} - R_I^T (I - H_I)^{-1} [\hat{Y}_I - Y_I].
\end{aligned}$$

Thus

$$(n-m-p)\hat{\Sigma}_{(I)} = (n-p)\hat{\Sigma} - R_I^T (I - H_I)^{-1} R_I \qquad \text{QED.}$$

APPENDIX L

PROGRAM LISTING FOR CALIBRATION IN A NON-STATIONARY FIELD

L.1 MAIN PROGRAM

```
cls;
count=1;
mseold=9999999999;
note="";
loadp gradp=c:\gauss\cp\gradp;
loadp hessp=c:\gauss\cp\hessp;
format /rd 3,0;
"                     Calibration in a correlated field";
"                     ================================";
"                    (Generalized Least-Squares Version)";
print;
"                              David R. Fox";
print;
"                          University of Wyoming";
"                             November 1988";
print;
print;
"  Input requirements :   It is assumed there are k locations for which
"  ==================     following data is available at each location :
print;
"          Xi : (v x 1)  position vector for ith. location.";
"          Wi : (n x p)  matrix of n observations on p independent variab
"          Vi : (n x 1)  vector of observations on dependent variable.";
print;
"  Given the linear model :  Vi=Xi*Bi + Ei where Bi is a (p x 1) vector
"  (unknown) parameters, the program will estimate using a least-squares
"  procedure, the matrix A (p x v) where it is further assumed that :";
print;
"                          Bi = A*Xi";
print;
dos pause;
cls;
print;
print;
print;
print;
"                        HAVE YOU LOADED THE GLSCALC PROGRAM  (y/n) ? : ";
```

```
reply=cons;
if reply $/="Y" and reply $/="y";
stop;
endif;
start:;
cls;
"                              DATA ENTRY";
"                              ==========";
print;
print;
loop1:;
"          Is data to come from keyboard or file (k/f) : ";;
device=cons;
if device $/="K" and device $/="k" and device $/="F" and device $/="f";
print;
"               * * * ERROR - MUST BE EITHER K OR T !";
"                        Please reenter";
"                     ";;
dos pause;
goto start;
elseif device $=="f" or device $=="F";
loop5:;
"          Enter drive and (optionally) a path for stored matrices : ";;
source=cons;
load path=^source;
loadm w;
loadm v;
loadm x;
nu=rows(x);
k=cols(x);
n=rows(w)/k;
p=cols(w);
goto flag1;
endif;
"          How many locations are there ? : ";;
k=con(1,1);
k=floor(k);
if k<=0;
"          < NOT A VALID SELECTION - REENTER >";
goto loop1;
endif;
print;
loop2:;
"          How many observations at each of these locations ? : ";;
n=con(1,1);
n=floor(n);
if n<=0;
"          < NOT A VALID SELECTION - REENTER >";
goto loop2;
endif;
print;
loop3:;
"          How many INDEPENDENT variables are there (Wi's) ? :";;
```

```
p=con(1,1);
p=floor(p);
if p<=0;
"         < NOT A VALID SELECTION - REENTER >";
goto loop3;
endif;
"         Is a constant (B0) term to be included in the model (y/n) : ";
reply1=cons;
if reply1 $=="Y" or reply1 $=="y";
p=p+1;
note="(First value corresponds to B0 term)";
endif;
if p>n;
print;
"                 * * * ERROR IN INPUT DATA * * *";
PRINT;
"                 There are more parameters than observations";
print;
"                      Please reenter from the beginning";
dos pause;
goto start;
endif;
print;
loop4:;
print;
"         What is the DIMENSION of the field (1,2,or 3) : ";;
nu=con(1,1);
if nu/=1 and nu/=2 and nu/=3;
print;
"                * * * ERROR - MUST BE 1,2, OR 3 ! * * *";
goto loop4;
endif;
nu1=nu;
cls;
if reply1$=="Y" or reply1$=="y";
nu=nu+1;
endif;
"                         SUMMARY OF DESIGN PARAMETERS";
"                         ============================";
print;
print;
format /ld 3,0;
"                       You have specified the following : ";
print;
print;
"     At each of ";;k;;" locations in ";;nu1;;"dimensional space";
"     there are ";;p-1;;" independent variables.";
print;
"     The number of observations on each variable at each location is ";
print;
print;
print;
"              Is this information correct (y/n) : ";;
```

```
reply=cons;
if reply $/= "Y" and  reply $/="y";
print;
print;
dos pause;
goto start;
endif;
i=1;
do while i<=k;
cls;
j=1;
"                              DATA ENTRY";
"                              ==========";
print;
print;
"LOCATION ";;i;
print;
"Enter the ";;nu;;"POSITION coordinates : "$+note;
x1=con(nu,1);
if i==1;
x=x1;
else;
x=x~x1;
endif;
do while j<=n;
print;
"INDEPENDENT variable information  (LOCATION ";;i;;")";
"=========================================================";
"Enter the ";;p;;"values for observation ";;j;;note;
w1=con(1,p);
print;
"Enter the value of the DEPENDENT variable : ";;
v1=con(1,1);
if j==1 and i==1;
w=w1;
v=v1;
goto jump1;
endif;
w=w|w1;
v=v|v1;
jump1:;
j=j+1;
endo;
i=i+1;
endo;
print;
print;
"       Do you want to save these matrices to a file (y/n) : ";;
reply=cons;
if reply $=="Y" or reply $== "y";
print;
"       Enter drive (and optionally a path ) : ";;
destn=cons;
```

```
save path=^destn;
save w;
"     File ";;destn;;"W.FMT successfully saved";
save x;
"     File ";;destn;;"X.FMT successfully saved";
save v;
"     File ";;destn;;"V.FMT successfully saved";
endif;
/* loadp gradp=c:\gauss\cp\gradp; */
/*"  Matrix W";
w;
"  Matrix V";
v;
"  Matrix X";
x;  */
/* loadp fcalc=c:\gauss\cp\fcalc; */;
flag1:;
aold=zeros(p*nu,1);
print;
print;
"       Do you want OLS estimation or GLS estimation (Type O or G) : ";;
olsgls=cons;
if olsgls $=="G" or olsgls $=="g";
vv=reshape(v,k,n);vv=vv';
kk=1;
do while kk<=k;
vv[.,kk]=vv[.,kk]-meanc(vv[.,kk]);
kk=kk+1;
endo;
cov=vv'*vv/n;
print;
"        Are observations WITHIN a location independent (y/n) : ";;
reply=cons;
jump8:;
if reply $=="y" or reply$=="Y";
psi=cov.*.eye(n);
else;
psi=cov.*.ones(n,n);
loadexe path=c:\gauss\gxe;
loadp pinv=c:\gauss\cp\pinv;
psi=pinv(psi);
goto jump9;
endif;
if n<k and reply $=="y" or  reply $=="Y";
print;
print;
"       NOTE : Since the number of observations WITHIN each location" ;
"              is less than the number of locations, will have to "  ;
"              compute the Moore-Penrose inverse of the psi matrix.";
loadexe path=c:\gauss\gxe;
loadp pinv=c:\gauss\cp\pinv;
psi=pinv(psi);
else;
```

```
psi=inv(psi);
endif;
else;
psi=eye(k*n);
endif;
jump9:;
if count>1;
goto jump7;
endif;
z=zeros(k*n,k*p);
kk=1;
do while kk<=k;
kk1=(kk-1)*n+1;
kk2=kk*n;
kk3=(kk-1)*p+1;
kk4=kk*p;
w1=w[kk1:kk2,1:p];
z[kk1:kk2,kk3:kk4]=w1;
kk=kk+1;
endo;
/* z; */
jump7:;
itern=1;
kk=1;
inc=9999;
/* a0=ones(p*nu,1); */
gamhat=inv(z'*psi*z)*z'*psi*v;
gamhat=reshape(gamhat,k,p);gamhat=gamhat';
a0=gamhat*x'*inv(x*x');
format /rd 9,4;
print;
print;
"Two-stage regression estimate of matrix A : ";;a0;
dos pause;
a0=ones(p*nu,1);
cls;
if count>1;
goto jump6;
endif;
"      Enter tolerance for determining stopping criterion : ";;
epsilon=con(1,1);
jump6:;
do while inc>epsilon;
"Iteration ";;kk;
pp=gradp(&glscalc,a0);
ph=hessp(&glscalc,a0);
delta=inv(ph)*pp';
inc=delta'delta;
a0=a0-delta;
kk=kk+1;
a0;
endo;
cls;
```

```
print;
print;
a0=reshape(a0,p,nu);
"                    Convergence established at iteration ";;kk;
print;
print;
format /rd 9,4;
"                    Matrix A = ";
a0;
print;
print;
"      Will now compute the predicted values of dependent variable";
"      using this A matrix . . .";
print;
"                       ";;dos pause;
i=1;
do while i<=k;
i1=(i-1)*n;
x1=x[.,i];
i2=1;
do while i2<=n;
if i2==1;
w1=w[i1+i2,.];
else;
w1=w1|w[i1+i2,.];
endif;
i2=i2+1;
endo;
if i==1;
vhat=w1*a0*x1;
else;
vhat=vhat|w1*a0*x1;
endif;
i=i+1;
endo;
print;
"    V and Vhat . . .";
v~vhat;
resid=v-vhat;
mse=resid'resid;
mse=mse/k/n;
print;
format /re 15,9;
"            Mean Square Error is ";;mse;
print;
format /rd 9,4;
if olsgls $=="o" or olsgls $=="O";
stop;
else;
print;
"            Do you want to continue iterating on psi matrix (y/n) : ";;
cont=cons;
if cont $=="n" or cont $=="N";
```

```
stop;
endif;
if abs(mseold-mse)>=epsilon;
aold=a0;
mseold=mse;
resid=(reshape(resid,k,n))';
cov=(resid'resid)/n;
count=count+1;
goto jump8;
endif;
endif;
```

L.2 AUXILIARY PROCEDURE FOR COMPUTATION OF EQUATION (4.29)

```
proc glscalc(arg);
local q,i,i1,i2,w1,v1,x1,f,f1,gn,a1,b1;
a1=reshape(arg,p,nu);
/* i=1;
do while i<=k;
i1=(i-1)*n;
i2=1;
do while i2<=n;
if i2==1;
w1=w[i1+i2,.];
v1=v[i1+i2,1];
else;
w1=w1|w[i1+i2,.];
v1=v1|v[i1+i2,1];
endif;
i2=i2+1;
endo;
x1=x[.,i];
f1=(v1-w1*a1*x1)'psi*(v1-w1*a1*x1);
if i==1;
f=f1;
else;
f=f+f1;
endif;
i=i+1;
endo; */
b1=vec(a1*x);
f=v-z*b1;
q=f'*psi*f;
retp(q);
endp;
```

BIBLIOGRAPHY

Aitchison,T.W. and Dunsmore,I.R. (1975) Statistical prediction analysis. Cambridge : University Press

Ali,M.A. and Singh,N. (1981) An alternative estimator in inverse linear regression. J.Statist.Comp. and Simulation 14 1-15

Anderson,T.W. (1984) An introduction to multivariate statistical analysis. John Wiley and Sons

Andrews,D.F. and Herzberg,A.M. (1973) A simple method of constructing exact tests for sequentially designed experiments. Biometrika 60 489-497

Andrews,D.F. and Pregibon,D. (1977) Finding the outliers that matter. J.R.Statist.Soc.Ser. B 40,1 85-93

Anscombe,F.J. and Tukey,J.W. (1963) The examination of residuals Technometrics 5,2 141-160

Atkinson,A.C. (1981) Two graphical displays for outlying and influential observations in regression Biometrika 68,1 13-20

Atkinson,A.C. (1985) Plots,Transformations, and regression : An introduction to graphical methods of diagnostic regression analysis. Clarendon Press, Oxford

Avery,R.B. (1977) Error components and seemingly unrelated regressions. Econometrica 45,1 199-209

Barker,D.R. and Diana,L.M. (1974) Simple method for fitting data when both variables have uncertainties. Aust.J.Physics 42 224-227

Beale,E.M.L. (1970) Note on procedures for variable selection in multiple regression. Technometrics 12,4 909-913

Belsley,D.A.,Kuh,E. and Welsch,R.E. (1980) Regression diagnostics. New York Wiley

Bendre,S.M. and Kale,B.K. (1985) Masking effect on tests for outliers in exponential models. J.Amer.Statits.Assocn. 80,392 1020-1025

Berkson,J. (1950) Are there two regressions ?
J.Amer.Statist.Assoc. 164-180

Berkson,J. (1969) Estimation of a linear function for a calibration line : Consideration of a recent proposal.
Technometrics 11 649-660

Borgman,L.E. (1982) Techniques for computer simulation of Ocean Waves.
Proc. Int. School of Physics, Course LXXX - Topics in Ocean Physics
A.R. Osborne (ed.) , North-Holland Publishing Co.

Brown,G.H. (1979) An optimization criterion for linear inverse estimation. Technometrics 21 575-579

Brown,P.J. (1982) Multivariate calibration (with discussion).
J.Roy.Statist.Soc.,Ser.B 44 287-321

Buonaccorsi,J.P. (1986) Design considerations for calibration.
Technometrics 28,2 149-155

Buonaccorsi,J.P. (1988) Errors-in-variables with systematic biases.
Unpublished manuscript.

Buse,A. (1979) Goodness-of-fit in the seemingly unrelated regression model. J.Econometrics 10 109-113

Butler,R.W. (1984) The significance attained by the best-fitting regressor variable. J.Amer.Statist.Assocn. 79,386 341-348

Campbell,N.A. (1978) The influence function as an aid in outlier detection in discriminant analysis. Appl.Statist. 27,3 251-258

Carrol,R.J. and Ruppert,D. (1981) On prediction and the power transformation family. Biometrika 68,3 609-615

Carrol,R.J. and Spiegelman,C. (1986) The effect of ignoring small measurement errors in precision instrument calibration.
J.Qual.Tech. 18,3 170-173

Carrol,R.J., Spiegelman,C.H. and Sacks,J. (1988) A quick and easy multiple-use calibration curve procedure.
Technometrics 30,2 137-141

Chambers,R.L. and Heathcote,C.R. (1981) On the estimation of slope and identification of outliers in linear regression.
Biometrika 68,1 21-23

Clutton Brock,M. (1967) Likelihood distributions for estimating functions when both variables are subject to error.
Technometrics 9,2 261-269

Cook,R.D. (1977) Detection of influential observations in linear regression. Technometrics 19 15-18

Cook,R.D. (1986) Assessment of local influence. J.Roy.Statist.Soc.Ser.B 48,2 133-169

Cramer,H. (1954) Mathematical methods of statistics. Princeton University Press.

Dempster,A.P. and Gasko-Green,M. (1981) New tools for residual analysis. The Annals of Statistics 9,5 945-959

Dey,A. (1985) Orthogonal fractional factorial designs. John Wiley and Sons

Dhrymes,P.J. (1971) Equivalence of iterative Aitken and maximum likelihood estimators for a system of regression equations. Aust.Economic papers 10 20-24

Dhrymes,P.J. (1971) Distributed lags : Problems of estimation and foundation. San Francisco : Holden-Day

Dhrymes,P.J. (1974) Econometrics : Statistical Applications and Foundations. Springer-Verlag

Dobrigal,A.,Fraser,D.A.S.,Gebotys,R. (1987) Linear calibration and conditional inference. Comm.Statist.Theory Meth. 16,4 1037-1048

Dunsmore,I.R. (1968) A Bayesian approach to calibration. J.Roy.Statist.Soc.,Ser B 31 160-170

Dwivedi,T.D. and Srivastiva,V. (1977) Optimality of least squares in the seemingly unrelated regression model. J.Econometrics 7 391-395

Eisenhart,C. (1939) The interpretation of certain regression methods and their use in biological and Industrial Research. Ann.Math.Statist. 10 162-186

Fearn,T. (1983) A misuse of ridge regression in the calibration of a near infrared reflectance instrument. Appl.Statist. 32,1 73-79

Fieller,E.C. (1940) The biological standardization of Insulin. J.Roy.Statist.Soc.(Supplement) 7 1-54

Fieller,E.C. (1954) Some problems in interval estimation. J.Roy.Statist.Soc.,Ser.B 16 175-185

Fomby,T.B.,Carter Hill,R., and Johnson,S.R. (1984) Advanced Econometric methods. Springer-Verlag

Fox,D.R. (1985) A statistical appraisal of vehicle speed determination from airborne observation. Tech. Report,Department of Mathematics and Statistics, Curtin University, Perth.

Fox,D.R. (1987) Components of variance estimation in a calibration problem where both variables are subject to error. Tech. Report,Department of Mathematics and Statistics, Curtin University, Perth.

Fox,D.R. (1987) Statistical calibration when both variables are subject to error. Tech. Report,Department of Mathematics and Statistics, Curtin University, Perth.

Fuller,W.A. (1987) Measurement Error Models. John Wiley and Sons

Ganse,R.A.,Amemiya,Y., and Fuller,W.A. (1983) Prediction when both variables are subject to error, with application to earthquake magnitudes. J.Amer.Statist.Assocn. 78,384 761-765

Goldberger,A.S. (1984) Reverse regression and salary discrimination. J. Human Resources 19,3 293-318

Graybill,F.A. (1976) Theory and Application of the linear model. Duxbury Press

Graybill,F.A. (1983) Matrices with applications in Statistics. Wadsworth International

Guttman,I. (1982) Linear models : An introduction. John Wiley and Sons

Hader,R.J. and Grandage,A.H.E. (1958) Simple and multiple regression analyses in experimental design in Industry. New York Wiley

Halperian,M. and Gurian,J. (1971) A note on estimation in straight line regression when both variables are subject to error. J.Amer.Statist.Assoc. 54,177 173-205

Halperin,M. (1970) On inverse estimation in linear regression. Technometrics 12,4 727-736

Halperin,M. (1970) On Inverse estimation in Linear Regression. Technometrics 12 727-736

Harris,R.J. (1975) A primer of multivariate statistics. Academic Press

Herbert,J.H. (1987) Measurement error and the estimation of regression coefficients - A brief case study Proc.Amer.Statist.Assocn. San Francisco 1987

Hoadley,B. (1970) A Bayesian look at inverse linear regression. J.Amer.Statist.Assoc. 65 356-369

Hoaglin,D.C. and Welsch,R.E. (1978) The hat matrix in regression and ANOVA. Amer.Statist. 32 17-22

Hoaglin,D.C.,Iglewicz,B., and Tukey,J.W. (1986) Performance of some resistant rules for outlier labeling.
J.Amer.Statist.Assocn. 81,396 991-999

Huber,P.J. (1983) Minimax aspects of bounded influence regression.
J.Amer.Statist.Assocn. 78,381 66-77

Hunter,G. and Lamboy,W (1981) A Bayesian Analysis of the linear calibration problem (with discussion).
Technometrics 23,4 323-328

Hunter,J.S. (1981) Calibration and the straight line : Current statistical practices. J.Assoc.Offic.Anal.Chem. 64 574-583

Jennings,D.E. (1986) Outliers and residual distributions in logistic regression. J.Amer.Statist.Assocn. 81,396 987-990

Johnson,R.A. and Wichern,D.W. (1982) Applied Multivariate Statistical Analysis. Prentice-Hall

Johnson,W. and Geisser,S. (1983) A predictive view of the detection and characterization of Influential observations in regression analysis. J.Amer.Statist.Assocn. 78,381 137-144

Judge,G.G. et. al. (1980) The theory and practice of Econometrics.
John Wiley and Sons

Karson,M.J. (1982) Multivariate Statistical Methods.
Iowa State University Press

Kmenta,J. (1986) Elements of Econometrics. MacMillian Publishing Co.

Knafl,G et. al. (1984) Nonparametric calibration.
Technometrics 26,3 233-241

Knafl,G.,Sacks,J. and Spiegelman,C.H. (1985) Confidence bands for regression. J.Amer.Statist.Assoc. 80 683-691

Krutchkoff,R.G. (1967) Classical and Inverse regression methods of calibration. Technometrics 9 425-439

Krutchkoff,R.G. (1968) Letter to the editor.
Technometrics 10 430-431

Krutchkoff,R.G. (1969) Classical and Inverse Regression methods of calibration in extrapolation. Technometrics 11 605-608

Krutchkoff,R.G. (1970) Letter to the editor.
Technometrics 12 433-434

Leamer,E.E. (1984) Global sensitivity results for generalized least squares estimates. J.Amer.Statist.Assocn. 79,388 867-870

Lechner,J.A. et. al. (1982) An implementation of the Scheffe approach to calibration using spline functions,illustrated by a pressure-volume calibration. Technometrics 24,3 229-234

Lewis,T.O. and Odell,P.L. (1971) Estimation in Linear Models. Prentice-Hall Inc.

Lwin,T. (1981) Discussion of Hunter,W. and Lamboy,W.F. (1981) paper. Technometrics 23,4 339-341

Lwin,T. and Maritz,J.S. (1980) A note on the problem of statistical calibration. J.Roy.Statist.Soc.Ser C. 29 135-141

Lwin,T. and Maritz,J.S. (1982) An analysis of the linear calibration controversy from the perspective of compound estimation. Technometrics 27,4 235-242

Madansky,A. (1959) The fitting of straight lines when both variables are subject to error. J.Amer.Statist.Assoc. 54 173-205

Maddula,G.S. (1988) Introduction to Econometrics. Macmillan New York.

Magnus,J.R. (1978) Maximum likelihood estimation of the GLS model with unknown parameters in the disturbance covariance matrix. J.Econometrics 7 281-312

Mandel,J. (1957) Fitting a straight line to certain types of cumulative data. J.Amer.Statist.Assoc. 52 552-566

Mandel,J. (1984) Fitting straight lines when both variables are subject to error. J.Qual.Tech. 16,1 1-14

Martinelle,S. (1970) On the choice of regression in linear calibration. Comments on a paper by R.G. Krutchkoff. Technometrics 12 157-161

Martinelle,S. (1970) On the choice of regression in linear calibration. Technometrics 12 157-161

Monahan,E.C. (1971) Oceanic Whitecaps. J.Physical Oceanography 1 139-144

NAG (1984) NAG program documentation. NAG, Oxford U.K.

Naes,T. (1985) Multivariate calibration when the error covariance matrix is structured. Technometrics 27,3 301-311

Naszodi,L.J. (1978) Elimination of the bias in the course of calibration. Technometrics 20 201-205

Nelder,J.A. and Wedderburn,R.W.M. (1972) Generalized linear models. J.R.Statist.Soc. (A), 135 370-384

O'Muircheartaich,I.G., Gaver,D (1986) Estimation of sea-surface wind speed from whitecap cover : statistical approaches compared empirically and by simulation. Int.J.Remote Sensing 7,8 985-999

Oberhofer,W. and Kmenta,J. (1974) A general procedure for obtaining maximum likelihood estimates in generalized regression models. Econometrica 42 579-590

Oden,A. (1973) Simultaneous confidence intervals in inverse linear regression. Biometrika 60,2 339-343

Oman,S.D (1985) An exact formula for the mean squared error of the inverse estimator in the linear calibration problem. J.Statist.Plan. and Inf. 11 189-196

Oman,S.D. (1982) Analyzing residuals in calibration problems. Technometrics 26,4 347-353

Oman,S.D. (1988) Confidence regions in multivariate calibration. Annals of Statistics 16,1 174-187

Oman,S.D. and Wax,Y. (1984) Estimating Fetal age by ultrasound measurements : An example of multivariate calibration. Biometrics 40 947-960

Orban,J.E. (1981) Discussion of Hunter,W.G. and Lamboy,W.F. Tecnometrics 23,4 342-343

Ott,R.L., and Myers,R.H. (1968) Optimal experimental designs for estimating the independent variable in regression. Technometrics 10 811-823

Picard,R.R and Cook,R.D. (1984) Cross-Validation of regression models. J.Amer.Statist.Assoc. 79,387 575-583

Pierce,D.A. and Schafer,D.W. (1986) Residuals in generalized linear models. J.Amer.Statist.Assocn. 81,396 977-986

Polasek,W. (1984) Regression diagnostics for general linear regression models. J.Amer.Statist.Assocn. 79,386 336-340

Pregibon,D. (1981) Logistic regression diagnostics. The Annals of Statistics 9,4 705-724

Press,S.J. (1972) Applied multivariate analysis. Holt,Rinehart and Winston

Randles,R.H. (1984) On tests applied to residuals. J.Amer.Statist.Assocn. 79,386 349-354

Reilman,M.A. and Gunst,R.F. (1986) Stochastic regression with errors in both variables. J.Qual.Tech. 18,3 162-169

Rosenblatt,J.,Spiegelman,C.H. (1981) Discussion of Hunter,W. and Lamboy,W.G. (1981) paper. Technometrics 23,4 329-333

Scheffe,H. (1973) A statistical theory of calibration. Ann.Statist. 1 1-37

Schmidt,P. (1978) A note on the estimation of seemingly unrelated regression systems. J.Econometrics 7 259-261

Shukla,G.K. (1972) On the problem of calibration. Technometrics 14 547-553

Shukla,G.K. and Datta,P. (1985) Comparison of the inverse estimator with the classical estimator subject to a preliminary test in linear calibration. J.Statist.Plan.and Inf. 12 93-102

Sjostrom,M. et. al. (1983) A multivariate calibration problem in analytical chemistry solved by partial least-squares models in latent variables. Analytica Chimica Acta 150 61-70

Smith,R.L. (1987) Measuring marathon courses : An application of statistical calibration theory. Appl.Statist. 36,3 283-295

Spezzaferri,F. (1985) A note on multivariate calibration experiments. Biometrics 41 267-272

Spiegelman,C.H. (1984) An iterative calibration curve procedure. J.Nat.Bureau of Standards 89,2 187-192

Spiegelman,C.H. (1984) A new statistic for detecting influential observations in a Scheffe type calibration curve. Aust.J.Statist. 26,3 290-297

Spiegelman,C.H. and Studden,W. 1980 Design aspects of Scheffe's calibration theory using linear splines. J.Res.Nat.Bureau of Standards 85,4 295-304

Srivastava,V.K. and Singh,A.K. (1987) A class of estimators for linear calibration problem. Proc.Am.Statist.Assocn. San Francisco 1987. (Statist.Comp. Sec) 175-177

Srivastava,V.K.,& Dwivedi,T.D. (1979) Estimation of seemingly unrelated regression equations : A brief survey. J.Econometrics 10 15-32

Strang,G. (1980) Linear Algebra and its applications. Harcourt,Brace Jovanovich

Street,A.P. and Street,D.J. (1987) Combinatorics of experimental design. Oxford University Press

Stukel,T.A. (1988) Calibration in binary response dose-response models Unpublished manuscript.

Velleman,P.F. and Welsch,R.E. (1981) Efficient computing of regression diagnostics. The American Statist. 35,4 234-242

Vellman,P.F. and Welsch,R.E. (1981) Efficient computing of regression diagnostics. Amer.Statist. 35 234-242

Williams,E.J. (1969) Regression methods in calibration problems. Bull.Inst.Internat.Statist. 43 17-28

Williams,E.J. (1969) A note on regression methods in calibration. Technometrics 11 189-192

Wolff,F.A. (1908) The temperature formula of the Weston Standard Cell Bull.Bur.of Standards 5 309-337

Wood,J. (1982) Estimating the age of an animal : An application of multivariate calibration. Proc.Int.Biometrics Conf. 1982

Zellner,A. (1962) An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. J.Amer.Statist.Assoc. 57 348-368