```
/* fcst1.gss, mww, 5/20/00
   Construct Forecasts Using Leading Indicators in Bivariate Regression Models
*/
new;
library pgraph;
dext="q59";                    @ Data Extension @
datadir="C:\\data\\Gauss\\apf\\ddisk2003\\data\\";      @ Data Path @
hpath="C:\\data\\Gauss\\apf\\ddisk2003\\gss\\";         @ Home Path @
fpath="C:\\data\\Gauss\\apf\\ddisk2003\\gss\\frc\\";   @ Path For Saving Forecasts @
slst="fyseries.inp";            @ Forecast Series list @
ulst="li.inp";                 @ Name of LI list @
clst="country.inp";            @ List of Country Names @
missc=   1.0e+32 ;             @ Missing Value Code in ASCII Files @

@ Sample Period @
fyds=59;     @ First Year of Data Set @
fmds=1;      @ First Quarter of Data Set @
lyds=99;     @ Last Year of Data Set @
lmds=04;     @ Last Quarter of Data Set @
fyest=60;    @ First Year to For Estimation @
fmest=1;     @ First Quarter For Estimation @
lyest=99;    @ First Year to For Estimation @
lmest=2;     @ First Quarter For Estimation @

@ Other Parameters @
nfstart=40;  @ minimum number of in-sample periods @

nphvec=2|4|8;   @ Number of Periods ahead for regressions
                Dep variable in regression is:
                inf(t+nph)-inf(t), where inf is defined above @

nolag=4;     @ maximum number of own lags to be included
                -- Note regression will include lags from
                   0  to nolag-1
                for a total of nolag values of the series  @
nulag=4;     @ maxium lags of LI include @
lmeth=2;      @  0 -- fixed
                 1 -- AIC
                 2 -- BIC @

@ Calendars, time vectors, etc. @
@ -- Sample Period Pointers -- @
nds =   4*(lyds-fyds)  + (lmds-fmds)+1;  @ Data Set Sample Size @
ifest = 4*(fyest-fyds) + (fmest-fmds)+1; @ Index of First Estimation Period @
ilest = 4*(lyest-fyds) + (lmest-fmds)+1; @ Index of Last Forecast Period @

@ -- Construct Calendar Sequences for Plotting, etc -- @
calds=zeros(nds,2);
calds[1,1]=fyds;
calds[1,2]=fmds;
yr=fyds; mt=fmds;
i=2; do while i <= nds;
 mt=mt+1;
 if mt .> 4; mt=1; yr=yr+1; endif;
 calds[i,1]=yr; calds[i,2]=mt;
i=i+1; endo;
 calest=calds[ifest:rows(calds),.];
 tds=calds[.,1]+(calds[.,2]-ones(rows(calds),1))/4;
```

```
 test=calest[.,1]+(calest[.,2]-ones(rows(calest),1))/4;


@ -- Read in List of Data Series and Tcodes-- @
tmp = datadir $+ slst;
f1=fopen(tmp,"r");
slstm=fgetsat(f1,3000);       @ List of Series to be forecast @
f1=close(f1);

tmp = datadir $+ ulst;
f1=fopen(tmp,"r");
ulstm=fgetsat(f1,3000);       @ List of Leading Indicators @
f1=close(f1);

tmp = datadir $+ clst;
f1=fopen(tmp,"r");
clstm=fgetsat(f1,3000);       @ List of Country lables @
f1=close(f1);

@ -- Load Necessary Procs -- @
#include fcst.prc;

inph=1; do while inph <= rows(nphvec);
 nph=nphvec[inph];
  nc=1; do while nc <= rows(clstm);
   {cname,tmp}=gname(clstm[nc]);

   ns=1; do while ns <= rows(slstm);
     {sname,scode}=gname(slstm[ns]);
   bname=sname $+ "@" $+ cname;
   aname=datadir $+ bname $+ ".q59";
   @ print /flush;; "Loading ";;aname; @
   load x[]=^aname;
   x=miss(x,missc);
   y=transx(x,scode);       @ -- Transform -- @
   yfor=yfcst(x,scode,nph); @ -- Construct Forecast variable -- @
/*
   title(bname);
   xy(tds,yfor);
   wait;
*/
   @ -- Construct Matrix of Own lags to be used in forecasting -- @
   yreg=miss(zeros(rows(y),nolag),0);
   i=1; do while i <= nolag;
     yreg[i:rows(yreg),i]=y[1:rows(y)+1-i];
   i=i+1; endo;

   @ Set up Matrix of Control Variables @
   cvar=ones(nds,1); @ Constant Term @

   @ --- Compute Univariate Forecast (with controls if desired) --- @
   print /flush;; bname;;"  univariate";;timestr(time);
   yforc=miss(zeros(nds,1),0);       @ Matrix for Storing Forecasts @

   t=ifest+nfstart; do while t <= ilest;
    @ -- Observations to use in regression -- @
    yfort=yfor[ifest:t-nph];
    yregt=yreg[ifest:t-nph,.];
```

```
cvart=cvar[ifest:t-nph,.];

temp=yfort~cvart~yregt;
temp=packr(temp);

if rows(temp) .>= nfstart;
  @ Eliminate Missing Observations @
  yfort=temp[.,1];
  k1=1;
  k2=k1+cols(cvart);
  cvart=temp[.,k1+1:k2];
  k1=k2;
  k2=k1+cols(yregt);
  yregt=temp[.,k1+1:k2];

  @ Determine Lag Length @
  nox=cols(yregt);
  if lmeth ./= 0;
   {nox,crit}=icmod1(yfort,cvart,yregt,lmeth);
  endif;

  @ Run Regression @
  xt=cvart;
  if nox .> 0; xt=xt~yregt[.,1:nox]; endif;
  breg=(invpd(xt'xt))*(xt'yfort);

  @ Construct Forecast @
  zt=cvar[t,.];
  if nox .> 0;
      zt=zt~yreg[t,1:nox];
    endif;
    yforc[t]=zt*breg;
   endif;
  t=t+1; endo;

/*
  @ -- Compute RMSE -- @
  e=yfor-yforcu;
  e=packr(e);
  mse=meanc(e.^2);
  rmse=sqrt(mse);
  format /ro 11,4;
  print /flush "Univariate rmse";;rmse;
*/

  @ -- Save Forecasts to Disk -- @
  fname = fpath $+ bname $+ "_" $+
          ftocv(scode,1,0) $+ "_" $+
          "uniar" $+ "_" $+
          ftocv(lmeth,1,0) $+ "_" $+
          ftocv(nph,2,0);
  b=tds~yfor~yforc;
  save ^fname=b;

@ James Block Start  ---------------------------Saving parameter counts for the univariate models-----------------------------------@
  fname = fpath $+ bname $+ "_" $+
          ftocv(scode,1,0) $+ "_" $+
          "uniar_lag" $+ "_" $+
```

```
           ftocv(lmeth,1,0) $+ "_" $+
           ftocv(nph,2,0) $+ ".txt";
       print fname;
       print nox;

  save ^fname=nox;

@ James Block End ------------------------------------------------------------------------------------------@

   nu=1; do while nu <= rows(ulstm);
     yforc=miss(zeros(nds,1),0);       @ Matrix for Storing Forecasts @
     {uname,ucode}=gname(ulstm[nu]);
     if uname .$== sname; goto bottom; endif;
     dname=uname $+ "@" $+ cname;
     aname=datadir $+ dname $+ ".q59";
     @ print /flush;; "Loading ";;aname; @
     print /flush;;"  ";;bname;;"  ";;uname;
     load x[]=^aname;
     x=miss(x,missc);
     y=transx(x,ucode);       @ -- Transform -- @

/*
     z=packr(y); if rows(z) .> 1;
      title(ulstm[nu]);
      xy(tds,y);
      wait;
     endif;
*/
     @ -- Construct Matrix of lags to be used in forecasting -- @
xreg=miss(zeros(rows(y),nulag),0);
i=1; do while i <= nulag;
  xreg[i:rows(xreg),i]=y[1:rows(y)+1-i];
i=i+1; endo;


t=ifest+nfstart; do while t <= ilest;
  @ -- Observations to use in regression -- @
  yfort=yfor[ifest:t-nph];
  yregt=yreg[ifest:t-nph,.];
  xregt=xreg[ifest:t-nph,.];
  cvart=cvar[ifest:t-nph,.];

  temp=yfort~cvart~xregt~yregt;
  temp=packr(temp);

  if rows(temp) .>= nfstart;
   @ Eliminate Missing Observations @
    yfort=temp[.,1];
    k1=1;
    k2=k1+cols(cvart);
    cvart=temp[.,k1+1:k2];
    k1=k2;
    k2=k1+cols(xregt);
    xregt=temp[.,k1+1:k2];
    k1=k2;
    k2=k1+cols(yregt);
    yregt=temp[.,k1+1:k2];
```

```
        @ Determine Lag Length @
         nux=cols(xregt);
         nox=cols(yregt);
         if lmeth ./= 0;
          @ -- Add First Column of xregt to cvart, this imposes the constraint
            that at least one lag xregt is included in model -- @
          {nux,nox,crit}=icmod2(yfort,cvart~xregt[.,1],xregt[.,2:cols(xregt)],yregt,lmeth);
          nux=nux+1;
         endif;
        @ Run Regression @
         xt=cvart;
         if nux .> 0; xt=xt~xregt[.,1:nux]; endif;
         if nox .> 0; xt=xt~yregt[.,1:nox]; endif;
         breg=(invpd(xt'xt))*(xt'yfort);

        @ Construct Forecast @
         zt=cvar[t,.];
         if nux .> 0; zt=zt~xreg[t,1:nux]; endif;
         if nox .> 0; zt=zt~yreg[t,1:nox]; endif;
         yforc[t]=zt*breg;
       endif;
      t=t+1; endo;

      bottom: @ Skipped if Own Series is Used as LI @

/*      @ -- Compute RMSE -- @
      e=yfor-yforc;
      e=packr(e);
      mse=meanc(e.^2);
       rmse=sqrt(mse);
       format /ro 11,4;
       print /flush "rmse";;rmse;
*/

     @ -- Save Forecasts to Disk -- @
     fname = fpath $+ bname $+ "_" $+
             ftocv(scode,1,0) $+ "_" $+
             uname $+ "_" $+
             ftocv(ucode,1,0) $+ "_" $+
             ftocv(lmeth,1,0) $+ "_" $+
             ftocv(nph,2,0);
     b=tds~yfor~yforc;
     save ^fname=b;

 @ James Block #2 Start   -------------------Saving parameter counts nux+nou for the non-univariate models---------------------@
    fname = fpath $+ bname $+ "_" $+
             ftocv(scode,1,0) $+ "_" $+
             uname $+ "_" $+
             ftocv(ucode,1,0) $+ "_lag_nux" $+
             ftocv(lmeth,1,0) $+ "_" $+
             ftocv(nph,2,0);
     save ^fname=nux;

     fname = fpath $+ bname $+ "_" $+
             ftocv(scode,1,0) $+ "_" $+
             uname $+ "_" $+
             ftocv(ucode,1,0) $+ "_lag_nox" $+
             ftocv(lmeth,1,0) $+ "_" $+
```

```
            ftocv(nph,2,0);
        save ^fname=nox;

        fname = fpath $+ bname $+ "_" $+
            ftocv(scode,1,0) $+ "_" $+
            uname $+ "_" $+
            ftocv(ucode,1,0) $+ "_lag_nux_nox" $+
            ftocv(lmeth,1,0) $+ "_" $+
            ftocv(nph,2,0);
        nuxnox=nux+nox+1;        @nox is count of AR variables, nux is count of lags of leading indicator, the +1 accounts for the intercept giving a parameter count @

         print fname;
         print nuxnox;

         save ^fname=nuxnox;


   @ James Block #2 End  ------------------------------------------------------------------------------@

     nu=nu+1; endo;
   ns=ns+1; endo;
  nc=nc+1; endo;
inph=inph+1; endo;
```