

```

/* comb.gss, mww, 12/20/02
  Combined Forecasts
  Compute combined Forecasts
*/

new;
library pgraph;
datadir="C:\\data\\Gauss\\apf\\ddisk2003\\data\\";      @ Data Directory @
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";      @ Country Names @

trimparm = .05;          @ Trimming Parameter for Trimmed Mean @

nperbest = 4;            @ Number of Time Periods for recent beset @

nmin = 32;               @ Number of periods before forecasts start @

nfac = 4;                @ Max Number of Factors @

dibpdate = 73.00;

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 @

lmeth=2;                @ 0 -- fixed
                        1 -- AIC
                        2 -- BIC @

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

output file = comb.out reset; output on;

@ -- Construct Weights for MSE calculation -- @
t=300;
dfac=0.9;
wdisc1 = seqm(1,dfac,t);
dfac=0.95;
wdisc2 = seqm(1,dfac,t);
dfac=1.0;
wdisc3 = seqm(1,dfac,t);

@ -- 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);

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; @
bname;
nu=1; do while nu <= rows(ulstm);
{uname,ucode}=gname(ulstm[nu]);
dname=uname $+ "@" $+ cname;
aname=datadir $+ dname $+ ".q59";

@ -- Load Forecasts from Disk -- @
fname = fpath $+ bname $+ "_" $+
ftocv(scode,1,0) $+ "_" $+
uname $+ "_" $+
ftocv(ucode,1,0) $+ "_" $+
ftocv(lmeth,1,0) $+ "_" $+
ftocv(nph,2,0);
load b=^fname;
tds=b[.,1];
yfor=b[.,2];
if nu .== 1;
yf=b[.,3];
else;
yf=yf~b[.,3];
endif;
nu=nu+1; endo;

tmpmed=miss(zeros(rows(tds),1),0);
tmpmean=miss(zeros(rows(tds),1),0);
tmpmeanb=miss(zeros(rows(tds),1),0);
tmpwdmse1=miss(zeros(rows(tds),1),0);
tmpwdmse2=miss(zeros(rows(tds),1),0);
tmpwdmse3=miss(zeros(rows(tds),1),0);
tmprbest=miss(zeros(rows(tds),1),0);
tmpmedb=miss(zeros(rows(tds),1),0);
tmpmeanb=miss(zeros(rows(tds),1),0);
tmpmeanb=miss(zeros(rows(tds),1),0);
tmpwdmse1b=miss(zeros(rows(tds),1),0);
tmpwdmse2b=miss(zeros(rows(tds),1),0);
tmpwdmse3b=miss(zeros(rows(tds),1),0);
tmprbestb=miss(zeros(rows(tds),1),0);
tmpdi1=miss(zeros(rows(tds),1),0);
tmpdi2=miss(zeros(rows(tds),1),0);
tmpstn1=miss(zeros(rows(tds),1),0);
tmpstn2=miss(zeros(rows(tds),1),0);

```

```

tmpstn3=miss(zeros(rows(tds),1),0);
tmptvp1=miss(zeros(rows(tds),1),0);
tmptvp2=miss(zeros(rows(tds),1),0);
tmptvp3=miss(zeros(rows(tds),1),0);

for t (1,rows(tds),1);

@ -- Mean, Median and Trimmed Mean -- @
z1=yf[t,.'];
z1=packr(z1);
if rows(z1) .>= 3;
  tmpmed[t]=med(z1);
  nz=rows(z1);
  tmpmean[t]=sumc(z1)/nz;
  zs=sortc(z1,1);
  ia=ceil(trimparm*rows(zs));
  tmptmean[t]=meanc(zs[ia+1:rows(zs)-ia]);
endif;

if (tds[t] > 96.99) .and (tds[t] < 97.1);
  trnd=sega(1,1,rows(ulstm));
  tmp=packr(trnd~yf[t,.']);
  trnd1=tmp[.,1];
  "Country";;cname;
  "Series";;sname;
  "Horizon";;nph;
  "1997 Unbalanced Panel";;cname;;sname;;rows(z1);
  "Series in Unbalanced Panel and Indicator (blank) or c";
  inu=1; do while inu <= rows(ulstm);
    strsect(ulstm[inu],1,15);;
    ii = sumc(inu.==trnd1);
    if ii .== 1;
      " ,c";
    else;
      " , ";
    endif;
    inu=inu+1; endo;
  endif;

@ -- Discounted Least Squares -- @
jj=1;
z1=miss(0,0);
t1=t;
for i (1,cols(yf),1);
  if ismiss(yf[t,i]) == 0;
    a = yfor[1:t]~yf[1:t,i];
    pa=packr(a);
    if rows(pa) .>= nmin+nph;
      e=pa[.,1]-pa[.,2];
      e=e[1:rows(e)-nph]; @ Shift back nph periods to compute in real time @
      e2 = e.^2;
      w=wdisc1[1:rows(e2)];
      w=rev(w);
      w=w./sumc(w);
      wmse=w'e2;
      if jj .== 1;
        z1=yf[t,i];

```

```

    msez=wmse;
else;
    z1=z1|yf[t,i];
    msez=msez|wmse;
endif;
    jj=jj+1;
endif;
endif;
endfor;

if rows(z1) .>= 3;
    w_msez = 1./msez;
    w_msez = w_msez/sumc(w_msez);
    tmpwdmse1[t]=w_msez*z1;
endif;

jj=1;
z1=miss(0,0);
t1=t;
for i (1,cols(yf),1);
    if ismiss(yf[t,i]) == 0;
        a = yfor[1:t]~yf[1:t,i];
        pa=packr(a);
        if rows(pa) .>= nmin+nph;
            e=pa[.,1]-pa[.,2];
            e=e[1:rows(e)-nph]; @ Shift back nph periods to compute in real time @
            e2 = e.^2;
            w=wdisc2[1:rows(e2)];
            w=rev(w);
            w=w./sumc(w);
            wmse=w'e2;
            if jj .== 1;
                z1=yf[t,i];
                msez=wmse;
            else;
                z1=z1|yf[t,i];
                msez=msez|wmse;
            endif;
            jj=jj+1;
        endif;
    endif;
endfor;

if rows(z1) .>= 3;
    w_msez = 1./msez;
    w_msez = w_msez/sumc(w_msez);
    tmpwdmse2[t]=w_msez*z1;
endif;

jj=1;
z1=miss(0,0);
t1=t;
for i (1,cols(yf),1);
    if ismiss(yf[t,i]) == 0;
        a = yfor[1:t]~yf[1:t,i];
        pa=packr(a);
        if rows(pa) .>= nmin+nph;
            e=pa[.,1]-pa[.,2];

```

```

e=e[1:rows(e)-nph]; @ Shift back nph periods to compute in real time @
e2 = e.^2;
w=wdisc3[1:rows(e2)];
w=rev(w);
w=w./sumc(w);
wmse=w'e2;
if jj .== 1;
    z1=yf[t,i];
    msez=wmse;
else;
    z1=z1|yf[t,i];
    msez=msez|wmse;
endif;
jj=jj+1;
endif;
endif;
endfor;

if rows(z1) .>= 3;
    w_msez = 1./msez;
    w_msez = w_msez/sumc(w_msez);
    tmpwdmse3[t]=w_msez'z1;
endif;

@ -- Recent Best -- @
jj=1;
z=miss(0,0);
for i (1,cols(yf),1);
    if ismiss(yf[t,i]) == 0;
        a = yfor[1:t]~yf[1:t,i];
        pa=packr(a);
        if rows(pa) .>= nperbest+nph;
            e=pa[:,1]-pa[:,2];
            e=e[1:rows(e)-nph]; @ Shift back nph periods to compute in real time @
            e=e[rows(e)-nperbest+1:rows(e)]; @ Use only last nperbest obs @
            e2 = e.^2;
            wmse=meanc(e2);
            if jj .== 1;
                z=yf[t,i];
                msez=wmse;
            else;
                z=z|yf[t,i];
                msez=msez|wmse;
            endif;
            jj=jj+1;
        endif;
    endif;
endfor;

if rows(z) .>= 3;
    ii=minindc(msez);
    tmprbest[t]=z[ii];
endif;

@ Construct Balanced Panel for Other Forecasts @
ismpl = tds[1:t,1] .>= dibpdate;
x=selif(yf[1:t,:],ismpl);

```

```

yfort=selif(yfor[1:t],ismpl);
@James Block zero -----@
JJJ=x;
@James Block end -----@

tmp=packr(x');
z=tmp';

if (tds[t] > 79.99) .and (tds[t] < 80.1);
trnd=seqa(1,1,rows(ulstm));
tmp=packr(trnd~x');
trnd1=tmp[.,1];
"Country";;cname;
"Series";;sname;
"Horizon";;nph;
"1980 B Panel";;cname;;sname;;cols(z);
"Series in BP and Indicator (blank) or a";
inu=1; do while inu <= rows(ulstm);
  strsect(ulstm[inu],1,15);;
  ii = sumc(inu.==trnd1);
  if ii .== 1;
    " ,a";
  else;
    " , ";
  endif;
  inu=inu+1; endo;
endif;

if cols(z) .> 1;

@ -- Mean, Median and Trimmed Mean -- @
z1=z[rows(z),.];
if rows(z1) .>= 3;
  tmpmedb[t]=med(z1);
  nz=rows(z1);
  tmpmeanb[t]=sumc(z1)/nz;
  zs=sortc(z1,1);
  ia=ceil(trimparm*rows(zs));
  tmptmeanb[t]=meanc(zs[ia+1:rows(zs)-ia]);
endif;

@ -- Discounted Least Squares -- @
jj=1;
z1=miss(0,0);
t1=t;
for i (1,cols(z),1);
  if ismiss(z[rows(z),i]) == 0;
    a = yfort~z[.,i];
    pa=packr(a);
    if rows(pa) .>= nmin+nph;
      e=pa[.,1]-pa[.,2];
      e=e[1:rows(e)-nph]; @ Shift back nph periods to compute in real time @
      e2 = e.^2;
      w=wdisc1[1:rows(e2)];
      w=rev(w);
      w=w./sumc(w);
      wmse=w'e2;
    endif;
  endif;
endfor;

```

```

    if jj .== 1;
        z1=z[rows(z),i];
        msez=wmse;
    else;
        z1=z1|z[rows(z),i];
        msez=msez|wmse;
    endif;
    jj=jj+1;
endif;
endfor;

if rows(z1) .>= 3;
    w_msez = 1./msez;
    w_msez = w_msez/sumc(w_msez);
    tmpwdmse1b[t]=w_msez'z1;
endif;

```

```

jj=1;
z1=miss(0,0);
t1=t;
for i (1,cols(z),1);
    if ismiss(z[rows(z),i]) == 0;
        a = yfort~z[.,i];
        pa=packr(a);
        if rows(pa) .>= nmin+nph;
            e=pa[.,1]-pa[.,2];
            e=e[1:rows(e)-nph]; @ Shift back nph periods to compute in real time @
            e2 = e.^2;
            w=wdisc2[1:rows(e2)];
            w=rev(w);
            w=w./sumc(w);
            wmse=w'e2;
            if jj .== 1;
                z1=z[rows(z),i];
                msez=wmse;
            else;
                z1=z1|z[rows(z),i];
                msez=msez|wmse;
            endif;
            jj=jj+1;
        endif;
    endif;
endfor;

```

```

if rows(z1) .>= 3;
    w_msez = 1./msez;
    w_msez = w_msez/sumc(w_msez);
    tmpwdmse2b[t]=w_msez'z1;
endif;

```

```

jj=1;
z1=miss(0,0);
t1=t;
for i (1,cols(z),1);
    if ismiss(z[rows(z),i]) == 0;

```

```

a = yfort~z[:,i];
pa=packr(a);
if rows(pa) .>= nmin+nph;
  e=pa[:,1]-pa[:,2];
  e=e[1:rows(e)-nph]; @ Shift back nph periods to compute in real time @
  e2 = e.^2;
  w=wdisc3[1:rows(e2)];
  w=rev(w);
  w=w./sumc(w);
  wmse=w'e2;
  if jj .== 1;
    z1=z[rows(z),i];
    msez=wmse;
  else;
    z1=z1|z[rows(z),i];
    msez=msez|wmse;
  endif;
  jj=jj+1;
endif;
endfor;

if rows(z1) .>= 3;
  w_msez = 1./msez;
  w_msez = w_msez/sumc(w_msez);
  tmpwdmse3b[t]=w_msez'z1;
endif;

@ -- Recent Best -- @
jj=1;
z1=miss(0,0);
t1=t;
for i (1,cols(z),1);
  if ismiss(z[rows(z),i]) == 0;
    a = yfort~z[:,i];
    pa=packr(a);
    if rows(pa) .>= nmin+nph;
      e=pa[:,1]-pa[:,2];
      e=e[1:rows(e)-nph]; @ Shift back nph periods to compute in real time @
      e=e[rows(e)-nperbest+1:rows(e)]; @ Use only last nperbest obs @
      e2 = e.^2;
      w=wdisc3[1:rows(e2)];
      w=rev(w);
      w=w./sumc(w);
      wmse=w'e2;
      if jj .== 1;
        z1=z[rows(z),i];
        msez=wmse;
      else;
        z1=z1|z[rows(z),i];
        msez=msez|wmse;
      endif;
      jj=jj+1;
    endif;
  endif;
endfor;

if rows(z1) .>= 3;

```



```
ii=minindc(msez);
tmprbestb[t]=z1[ii];
endif;
```

```
@ -- Stein Shrinkage -- @
sweight = 0.25;
w = 0;
bols = zeros(cols(z),1);
if rows(z) .> cols(z) + nph + 1;
w = 1 - sweight*(cols(z)/(rows(z)-cols(z)));
w = maxc(0|w);
x=z[1:rows(z)-nph,.];
y=yfort[1:rows(yfort)-nph,1];
bols=inv(x'x)*(x'y);
endif;
bstein = w*bols + (1-w)*(ones(rows(bols),1)/rows(bols));
tmpstn1[t]=z[rows(z),.]*bstein;
```

```
@-----James Block 1 -----@
print "testtesttest";
print t;
print "Country";;cname;
"Series";;sname;
"Horizon";;nph;
if t==162;
if nph==2;
print "w ";;w;
print bols;
print x;
print z;
print JJJ;
endif;
endif;
```

```
@-----End James Block 1 -----@
```

```
@ -- Stein Shrinkage -- @
sweight = 0.5;
w = 0;
bols = zeros(cols(z),1);
if rows(z) .> cols(z) + nph + 1;
w = 1 - sweight*(cols(z)/(rows(z)-cols(z)));
w = maxc(0|w);
x=z[1:rows(z)-nph,.];
y=yfort[1:rows(yfort)-nph,1];
bols=inv(x'x)*(x'y);
endif;
bstein = w*bols + (1-w)*(ones(rows(bols),1)/rows(bols));
tmpstn2[t]=z[rows(z),.]*bstein;
```

```
@ -- Stein Shrinkage -- @
sweight = 1.0;
w = 0;
bols = zeros(cols(z),1);
if rows(z) .> cols(z) + nph + 1;
w = 1 - sweight*(cols(z)/(rows(z)-cols(z)));
w = maxc(0|w);
```

```

x=z[1:rows(z)-nph,.];
y=yfort[1:rows(yfort)-nph,1];
bols=inv(x'x)*(x'y);
endif;
bstein = w*bols + (1-w)*(ones(rows(bols),1)/rows(bols));
tmpstn3[t]=z[rows(z),.]*bstein;

```

```

@ -- TVP Shrinkage -- @
tvpse=.1;
tmp=packr(yfor[1:t]);
sdtmp=stdc(tmp);
r=0.5*sdtmp*sdtmp; @ R-squared = 0.5 ... calibration @
if rows(z) .> nph + 1;
x=z[1:rows(z)-nph,.];
y=yfort[1:rows(yfort)-nph,1];
btvp=ones(cols(x),1)/cols(x);
@ -- Carry Out KF -- @
p=zeros(cols(x),cols(x));
q=(tvpse*tvpse/(cols(x)*cols(x)))*eye(cols(x));
ikf=1; do while ikf <= rows(y);
p=p+q; @ p2 @
eta=y[ikf]-x[ikf,]*btvp; @ innovation @
h=x[ikf,]*p*x[ikf,]'+r;
px=p*x[ikf,]';
kt=px/h;
btvp=btvp+kt*eta;
p=p-kt*px';
ikf=ikf+1; endo;
tmp1[t]=z[rows(z),.]*btvp;
endif;

```

```

tvpse=.2;
tmp=packr(yfor[1:t]);
sdtmp=stdc(tmp);
r=0.5*sdtmp*sdtmp; @ R-squared = 0.5 ... calibration @
if rows(z) .> nph + 1;
x=z[1:rows(z)-nph,.];
y=yfort[1:rows(yfort)-nph,1];
btvp=ones(cols(x),1)/cols(x);
@ -- Carry Out KF -- @
p=zeros(cols(x),cols(x));
q=(tvpse*tvpse/(cols(x)*cols(x)))*eye(cols(x));
ikf=1; do while ikf <= rows(y);
p=p+q; @ p2 @
eta=y[ikf]-x[ikf,]*btvp; @ innovation @
h=x[ikf,]*p*x[ikf,]'+r;
px=p*x[ikf,]';
kt=px/h;
btvp=btvp+kt*eta;
p=p-kt*px';
ikf=ikf+1; endo;
tmp2[t]=z[rows(z),.]*btvp;
endif;

```

```

tvpse=.4;
tmp=packr(yfor[1:t]);
sdtmp=stdc(tmp);

```

```

r=0.5*sdtmp*sdtmp; @ R-squared = 0.5 ... calibration @
if rows(z) .> nph + 1;
  x=z[1:rows(z)-nph,.];
  y=yfort[1:rows(yfort)-nph,1];
  btvp=ones(cols(x),1)/cols(x);
  @ -- Carry Out KF -- @
  p=zeros(cols(x),cols(x));
  q=(tvpse*tvpse/(cols(x)*cols(x)))*eye(cols(x));
  ikf=1; do while ikf <= rows(y);
    p=p+q; @ p2 @
    eta=y[ikf]-x[ikf,]*btvp; @ innovation @
    h=x[ikf,]*p*x[ikf,]'+r;
    px=p*x[ikf,]';
    kt=px/h;
    btvp=btvp+kt*eta;
    p=p-kt*px';
    ikf=ikf+1; endo;
  tmp3[t]=z[rows(z),.]*btvp;
endif;
@ -- DI, PC -- @
if cols(z) .> nfac+1;
  if rows(z) .> nmin+nph;
    @ -- No Constant Adjustment -- @
    zz=z*z';
    {va,ve}=eighv(zz);
    va=rev(va);
    ve=(rev(ve'))';
    va=va[1:nfac];
    ve=ve[.,1:nfac]; @ Factors @
    vet=ve[1:rows(ve)-nph,.]; @ Lagged for regression @
    x=vet;
    y=yfort[1:rows(yfort)-nph,1];
    aicmeth=1; @ AIC @
    bicmeth=2; @ bic @
    {ix,c}=icmod0(y,x,bicmeth);
    x=x[.,1:ix];
    b=invpd(x'x)*(x'y);
    xt=ve[rows(ve),1:ix];
    tmpdil[t]=xt*b;
  endif;
endif;

@ -- DI, PC -- @
if cols(z) .> nfac+1;
  if rows(z) .> nmin+nph;
    @ -- No Constant Adjustment -- @
    zz=z*z';
    {va,ve}=eighv(zz);
    va=rev(va);
    ve=(rev(ve'))';
    va=va[1:nfac];
    ve=ve[.,1:nfac]; @ Factors @
    vet=ve[1:rows(ve)-nph,.]; @ Lagged for regression @
    x=vet;
    y=yfort[1:rows(yfort)-nph,1];
    aicmeth=1; @ AIC @
    bicmeth=2; @ bic @
    {ix,c}=icmod0(y,x,aicmeth);
  endif;
endif;

```

```

        x=x[:,1:ix];
        b=invpd(x'x)*(x'y);
        xt=ve[rows(ve),1:ix];
        tmpdi2[t]=xt*b;
    endif;
endif;

endif; @ For Balanced Panel Calculation @

endfor; @ t @
@ -- Save Forecasts to Disk -- @

fstr="med";
b=tds~yfor~tmpmed;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;

fstr="mean";
b=tds~yfor~tmpmean;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;

fstr="tmean";
b=tds~yfor~tmptmean;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;

fstr="wdmse1";
b=tds~yfor~tmpwdmse1;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;

fstr="wdmse2";
b=tds~yfor~tmpwdmse2;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;

fstr="wdmse3";

```

```
b=tds~yfor~tmpwdmse3;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;
```

```
fstr="rbest";
b=tds~yfor~tmpbest;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;
```

```
fstr="medb";
b=tds~yfor~tmpmedb;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;
```

```
fstr="meanb";
b=tds~yfor~tmpmeanb;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;
```

```
fstr="tmeanb";
b=tds~yfor~tmptmeanb;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;
```

```
fstr="wdmse1b";
b=tds~yfor~tmpwdmse1b;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;
```

```
fstr="wdmse2b";
b=tds~yfor~tmpwdmse2b;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
```

```

    fstr $+ "_" $+
    ftocv(lmeth,1,0) $+ "_" $+
    ftocv(nph,2,0);
save ^fname=b;

fstr="wdmse3b";
b=tds~yfor~tmpwdmse3b;
fname = fpath $+ bname $+ "_" $+
    ftocv(scode,1,0) $+ "_" $+
    fstr $+ "_" $+
    ftocv(lmeth,1,0) $+ "_" $+
    ftocv(nph,2,0);
save ^fname=b;

fstr="rbestb";
b=tds~yfor~tmprbestb;
fname = fpath $+ bname $+ "_" $+
    ftocv(scode,1,0) $+ "_" $+
    fstr $+ "_" $+
    ftocv(lmeth,1,0) $+ "_" $+
    ftocv(nph,2,0);
save ^fname=b;

fstr="di1";
b=tds~yfor~tmpdi1;
fname = fpath $+ bname $+ "_" $+
    ftocv(scode,1,0) $+ "_" $+
    fstr $+ "_" $+
    ftocv(lmeth,1,0) $+ "_" $+
    ftocv(nph,2,0);
save ^fname=b;

fstr="di2";
b=tds~yfor~tmpdi2;
fname = fpath $+ bname $+ "_" $+
    ftocv(scode,1,0) $+ "_" $+
    fstr $+ "_" $+
    ftocv(lmeth,1,0) $+ "_" $+
    ftocv(nph,2,0);
save ^fname=b;

fstr="stn1";
b=tds~yfor~tmpstn1;
fname = fpath $+ bname $+ "_" $+
    ftocv(scode,1,0) $+ "_" $+
    fstr $+ "_" $+
    ftocv(lmeth,1,0) $+ "_" $+
    ftocv(nph,2,0);
save ^fname=b;

fstr="stn2";
b=tds~yfor~tmpstn2;
fname = fpath $+ bname $+ "_" $+
    ftocv(scode,1,0) $+ "_" $+
    fstr $+ "_" $+
    ftocv(lmeth,1,0) $+ "_" $+
    ftocv(nph,2,0);
save ^fname=b;

```

```

fstr="stn3";
b=tds~yfor~tmpstn3;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;

fstr="tvp1";
b=tds~yfor~tmptvp1;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;

fstr="tvp2";
b=tds~yfor~tmptvp2;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;

fstr="tvp3";
b=tds~yfor~tmptvp3;
fname = fpath $+ bname $+ "_" $+
        ftocv(scode,1,0) $+ "_" $+
        fstr $+ "_" $+
        ftocv(lmeth,1,0) $+ "_" $+
        ftocv(nph,2,0);
save ^fname=b;
ns=ns+1; endo;
nc=nc+1; endo;

inph=inph+1; endo;

output off;

```