```
clear
addpath('./functions');
dfolder='./data/';
sfolder='./results/';
file=1;
sfile=strcat(dfolder,'dataxx0',num2str(file));
%Load data and transform%%%%%%%
load(sfile);
dataS=standardise(dataS); %standardise data
%%estimation options%%%%%%
T0=20;   %training sample
L=2;     %lag for transition equation
Lx=1;    %lag for idiosyncratic component transition eq
REPS=10000; %Reps
BURN=5000; %burn-in
SKIP=5;  %every SKIP draw is kept after burn-in
maxdraws=100; %max trys to find stable coefficients
CHECK=1;
Sindex=BURN+1:SKIP:REPS;
fsize=length(Sindex);
id=unique(index); %index of countries
NC=length(id); %number of countries
NN=cols(dataS); %number of series
idc=vec(repmat(1:NC,1,1)); %index of countries
%%%%Starting Values and Priors
%initial estimate of the factors
pmatw=extract(dataS,1); %PC estimator of world factor
dataSS=dataS-pmatw*(pmatw\dataS);
pmatc=zeros(rows(pmatw),NC); %PC for countries
for i=1:NC
    dataC=dataSS(:,index==id(i));
    tmp=extract(dataC,1);
pmatc(:,i)=tmp;
end
res=zeros(rows(pmatw),NN); %idiosyncratic
FLOAD0=zeros(NN,2);  %prior mean for Factor loadings
for j=1:NN

        yy=dataS(:,j);
        xx=[pmatw pmatc(:,idc==index(j))];
        BB=xx\yy;
        res(:,j)=yy-xx*BB;
        FLOAD0(j,:)=BB';
end
VFLOAD0=eye(2).*10; %prior variance
%priors for TVP parameters
scale=3.5e-04;
%world factor
[y0w,x0w]=preparex(pmatw(1:T0,:),L,1);
[b00w,s00w,p00w]=getols(y0w,x0w); %OLS AR regression on pre-sample
Q0w=scale*p00w*T0; %OLS covariance times T0 times scaling (scale matrix for IW prior Qw~IW(Q0w,
T0)
Qw=Q0w; %starting value
%country factors
b00c=cell(NC,1);
s00c=cell(NC,1);
p00c=cell(NC,1);
Q0c=cell(NC,1);
Qc=zeros(L+1,L+1,NC);
```

```
for j=1:NC
[y0c,x0c]=preparex(pmatc(1:T0,j),L,1);
[b00c{j},s00c{j},p00c{j}]=getols(y0c,x0c);
Q0c{j}=scale*p00c{j}*T0;  %Qc~IW(Q0c{j},T0) for j=1,2,...NC
Qc(:,:,j)=scale*p00c{j}*T0; %starting value for Qc
end
%idiosyncratic
b00e=cell(NN,1);
s00e=cell(NN,1);
p00e=cell(NN,1);
Q0e=cell(NN,1);
Qe=zeros(Lx,Lx,NN);
for j=1:NN
[y0e,x0e]=preparex(res(1:T0,j),Lx,0);
[b00e{j},s00e{j},p00e{j}]=getols(y0e,x0e);
Q0e{j}=scale*p00e{j}*T0; %Qe~IW(Q0e{j},T0) for j=1,2,...NN
Qe(:,:,j)=scale*p00e{j}*T0; %Starting values
end
%remove training sample
dataS=dataS(T0+1:end,:);
pmatw=pmatw(T0+1:end,:);
pmatc=pmatc(T0+1:end,:);
res=res(T0+1:end,:);
T=rows(dataS);
%priors and starting values for stochastic volatilties as residual^2+small
%number
%world
[y0w,x0w]=preparex(pmatw,L,1);
[~,~,~,epsw]=getols(y0w,x0w); %regression of factor on lags
hlastw=epsw.^2+0.0001; %residual^2+small number
hlastw=[hlastw(1:L+1,:);hlastw];
%country
hlastc=zeros(T+1,NC);
for j=1:NC
[y0c,x0c]=preparex(pmatc(:,j),L,1);
[~,~,~,epsc]=getols(y0c,x0c);%regression of factor on lags
hlastcc=epsc.^2+0.0001;%residual^2+small number
hlastcc=[hlastcc(1:L+1,:);hlastcc];
hlastc(:,j)=hlastcc;
end
%idiosyncratic
hlaste=zeros(T+1,NN);
for j=1:NN
[y0e,x0e]=preparex(res(:,j),Lx,0);
[~,~,~,epse]=getols(y0e,x0e);%regression of factor on lags
hlastee=epse.^2+0.0001;%residual^2+small number
hlastee=[hlastee(1:Lx+1,:);hlastee];
hlaste(:,j)=hlastee;
end
SS0=10;    %variance of initial condition of SVOL
g0=0.1^2;  %prior scale parameter for inverse gamma prior for g
Tg0=1;     %prior degrees of freedom
gw=g0;  %starting values
gc=ones(NC,1).*g0;  %starting values
ge=ones(NN,1).*g0;   %starting values
beta0w=repmat(b00w',T,1);
beta0c=zeros(T,L+1,NC);
for j=1:NC
    beta0c(:,:,j)=repmat(b00c{j}',T,1);
end
```

```
beta0e=zeros(T,Lx,NN);
for j=1:NN
    beta0e(:,:,j)=repmat(b00e{j}',T,1);
end
%initial conditions for the factors
pmat00=[pmatw(L,:) pmatc(L,:)];
for j=1:L-1
    pmat00=[pmat00 [pmatw(L-j,:) pmatc(L-j,:)]];
end
vmat00=eye(cols(pmat00))*1;
save priors
```

Published with MATLAB® R2015b