```
clear
addpath('functions');
% a TVP-VAR with dlog(GDP) dlog(CPI) and R for the US 1962 2004
%load data
data=xlsread('\data\usdata.xls')/100;
N=size(data,2);
L=2;   %number of lags in the VAR
Y=data;
X=[ lag0(Y,1) lag0(Y,2) ones(size(Y,1),1) ];
Y=Y(3:end,:);
X=X(3:end,:);
%step 1 set starting values and priors using a pre-sample of 10 years
T0=40;
y0=Y(1:T0,:);
x0=X(1:T0,:);
b0=x0\y0;
```

$$\beta_0 = (X'_{0t}X_{0t})^{-1}(X'_{0t}Y_{0t})$$

```
e0=y0-x0*b0;
sigma0=(e0'*e0)/T0;
V0=kron(sigma0,inv(x0'*x0));
```

$$p_{0\backslash 0} = \Sigma_0 \otimes (X'_{0t}X_{0t})^{-1}$$

```
%priors for the variance of the transition equation
Q0=V0*T0*3.5e-04;
```

$$p_{0\backslash 0} \times T_0 \times \tau$$

```
%prior for the variance of the transition equation error
P00=V0;
```

$$p_{0\backslash 0}$$

```
% variance of the intial state vector variance of state variable p[t-1/t-1]
beta0=vec(b0)';
```

$$\beta_{0\backslash 0} = vec(\beta_0)'$$

```
% intial state vector   %state variable  b[t-1/t-1]
%initialise
Q=Q0;
R=sigma0;
%remove intial Sample
Y=Y(T0+1:end,:);
X=X(T0+1:end,:);
T=rows(X);
%Gibbs sampling algorithm Step 2
reps=110000;
burn=109000;
mm=1;
for m=1:reps
m
%%Step 2a Set up matrices for the Kalman Filter
ns=cols(beta0);
F=eye(ns); fixed
mu=0;  fixed
beta_tt=[];          %will hold the filtered state variable
ptt=zeros(T,ns,ns);    % will hold its variance
beta11=beta0;
p11=P00;
% %%%%%%%%%%%Step 2b run Kalman Filter
for i=1:T
   x=kron(eye(N),X(i,:));
```

$$(I_N \otimes X_t)$$

```
    %Prediction
beta10=mu+beta11*F';
```

$$\beta_{t|t-1} = \mu + F\beta_{t-1|t-1}$$

```
p10=F*p11*F'+Q;
```

$$p_{t\backslash t-1} = Fp_{t-1\backslash t-1}F' + Q$$

```
yhat=(x*(beta10)')';
```

$$X_t\beta_{t|t-1}$$

```
eta=Y(i,:)-yhat;
```

$$\eta_{t\backslash t-1} = Y_t - X_t\beta_{t|t-1}$$

```
feta=(x*p10*x')+R;
```

$$f_{t\backslash t-1} = X_t p_{t\backslash t-1} X_t' + R$$

```
%updating
K=(p10*x')*inv(feta);
```

$$K_t = p_{t\backslash t-1} X_t' f_{t\backslash t-1}^{-1}$$

```
beta11=(beta10'+K*eta')';
```

$$\beta_{t\backslash t} = \beta_{t|t-1} + K_t\eta_{t\backslash t-1}$$

```
p11=p10-K*(x*p10);
```

$$p_{t\backslash t} = p_{t\backslash t-1} - K_t X_t p_{t\backslash t-1}$$

```
ptt(i,:,:)=p11;
beta_tt=[beta_tt;beta11];
end
%%%%%%%%%%%end of Kalman Filter%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%step 2c Backward recursion to calculate the mean and variance of the distribution of the state
%vector
chck=-1;
while chck<0 while loop to ensure VAR stable at each point in time
beta2 = zeros(T,ns);   %this will hold the draw of the state variable
wa=randn(T,ns);
error=zeros(T,N);
roots=zeros(T,1);
i=T;  %period t
p00=squeeze(ptt(i,:,:));
beta2(i,:)=beta_tt(i:i,:)+(wa(i:i,:)*chol(p00)); %draw for beta in period t from N(beta_tt,ptt)
```

$$\beta_T \sim N(\beta_{T\backslash T}, p_{T\backslash T})$$

```
error(i,:)=Y(i,:)-X(i,:)*reshape(beta2(i:i,:),N*L+1,N);  %var residuals calculate var residuals in the same loop for convenience
roots(i)=stability(beta2(i,:)',N,L); checking stability at ith time period roots(i)=1 if stability violated
%periods t-1..to .1
for i=T-1:-1:1
pt=squeeze(ptt(i,:,:));
bm=beta_tt(i:i,:)+(pt*F'*inv(F*pt*F'+Q)*(beta2(i+1:i+1,:)-beta_tt(i,:)*F')')';
```

$$\beta_{t\backslash t,B_{t+1}} = \beta_{t\backslash t} + p_{t\backslash t}F'(Fp_{t\backslash t}F' + Q)^{-1}(\beta_{t+1} - \mu - F\beta_{t\backslash t})$$

```
pm=pt-pt*F'*inv(F*pt*F'+Q)*F*pt;
```

$$p_{t\backslash t,B_{t+1}} = p_{t\backslash t} - p_{t\backslash t}F'(Fp_{t\backslash t}F' + Q)^{-1}Fp_{t\backslash t}$$

```
beta2(i:i,:)=bm+(wa(i:i,:)*chol(pm));
```

$$\beta_t\backslash\beta_{t+1} \sim N(\beta_{t\backslash t,B_{t+1}}, p_{t\backslash t,B_{t+1}})$$

```
error(i,:)=Y(i,:)-X(i,:)*reshape(beta2(i:i,:),N*L+1,N);
roots(i)=stability(beta2(i,:)',N,L);
end
if sum(roots)==0
    chck=1;
end
end
% step 3 sample Q from the IW distribution
errorq=diff(beta2);
```

$$\tilde\beta_t^1 - \tilde\beta_{t-1}^1$$

```
scaleQ=(errorq'*errorq)+Q0;
```

$$\left(\tilde\beta_t^1 - \tilde\beta_{t-1}^1\right)'\left(\tilde\beta_t^1 - \tilde\beta_{t-1}^1\right) + Q_0$$

```
Q=iwpQ(T+T0,inv(scaleQ));
```

Sample Q from its conditional posterior distribution

```
%step4 sample R from the IW distribution
```

```
scaleR=(error'*error);
```

$$\left(Y_t-\left(c_t^1+\sum_{j=1}^{P}B_{j,t}^1Y_{t-j}\right)\right)'\left(Y_t-\left(c_t^1+\sum_{j=1}^{P}B_{j,t}^1Y_{t-j}\right)\right)+R_0$$

Sample R from its conditional posterior distribution

```
R=iwpQ(T,inv(scaleR));
if m>burn
       %save output from Gibbs sampler
       out1(mm,1:T,:)=beta2;
       out2(mm,1:N,1:N)=R;
       out3(mm,1:N*(N*L+1),1:N*(N*L+1))=Q;
        mm=mm+1;
end
end
%save results
save tvp.mat out1 out2 out3
%compute irf to a policy shock using sign restrictions
horz=40;% impulse response horizon
irfmat=zeros(size(out1,1),T,horz,N); %empty matrix to save impulse response to a policy shock
for i=1:size(out1,1);
    sigma=squeeze(out2(i,:,:));
   %sign restrictions
        chck=-1;
        while chck<0
        K=randn(N,N);
        QQ=getQR(K);
        A0hat=chol(sigma);
        A0hat1=(QQ*A0hat);  %candidate draw
        for m=1:N
        %check signs in each row
        e1=A0hat1(m,1)<0;  %Response of Y
        e2=A0hat1(m,2)<0;  %Response of P
        e3=A0hat1(m,3)>0;  %Response of R

        if e1+e2+e3==3
            MP=A0hat1(m,:);
            chck=10;
        else
            %check signs but reverse them
         e1=-A0hat1(m,1)<0;  %Response of Y
        e2=-A0hat1(m,2)<0;  %Response of P
        e3=-A0hat1(m,3)>0;  %Response of R

        if e1+e2+e3==3
            MP=-A0hat1(m,:);
            chck=10;
        end
        end
        end
        end
        %re-shuffle rows of A0hat1 and insert MP in the first row
        A0x=[]; %will hold rows of A0hat1 not equal to MP
        for m=1:N
            ee=sum(abs(A0hat1(m,:))==abs(MP));
            if ee==0
                A0x=[A0x;A0hat1(m,:)];
            end
        end
        A0new=[A0x;MP]; %A0 matrix to be used for impulse response
    shock=[0 0 1];
    for j=1:size(out1,2)
```

```
        btemp=squeeze(out1(i,j,:));
        btemp=reshape(btemp,N*L+1,N);
        zz=irfsim(btemp,N,L,A0new,shock,horz+L);
        zz=zz./repmat(zz(1,3),horz,N);
        irfmat(i,j,:,:)=zz;
    end
end
TT=1964.75:0.25:2010.5;
HH=0:horz-1;
irf1=squeeze(median(irfmat(:,:,:,1),1));
irf2=squeeze(median(irfmat(:,:,:,2),1));
irf3=squeeze(median(irfmat(:,:,:,3),1));
figure(1)
subplot(2,2,1);
mesh(TT,HH,irf1')
ylabel('Impulse Horizon');
xlabel('Time');
axis tight
title('GDP growth');
subplot(2,2,2);
mesh(TT,HH,irf2')
ylabel('Impulse Horizon');
xlabel('Time');
axis tight
title('Inflation');
subplot(2,2,3);
mesh(TT,HH,irf3')
ylabel('Impulse Horizon');
xlabel('Time');
axis tight
title('Federal Funds Rate');
```