Problem Set 5 Solutions

March 1, 2024

1. Deflate the stock price index by dividing the SP500 data by the GDP Deflator. Note that the TFP series from the data set is in annualized growth rates, so divide it by 4 before estimation. Estimate a VAR(4) for the growth rate of deflated stock prices and the growth rate of TFP.

2. Identify a news shock using the restriction that the shock has no effect contemporaneously on TFP growth.

3. Plot the impulse response functions of (log) TFP and (log) stock prices (point estimates and the 68% confdence bands computed with the bootstrap method) for level responses.

4. Plot the percentage of forecast error variance of (log) TFP and (log) stock prices.

5. Identify a technology shock with the restriction that it is the only one having effect in the long run on (log) TFP. Compute the correlation between the shock identified here and that identified at point 2. Are the results in line with those obtained in Beaudry and Portier (AER 2006)? Comment a bit on this.

6. Now consider a VAR(4) that contains the following variables: the growth rate of TFP, the growth rate of stock prices, the growth rate of real GDP and the growth rate of real consumption in that order. Identify the news shock assuming that the shock has no contemporaneous effect on (log) TFP but the response of (log) TFP at $h = 40$ is maximal.

7. Plot the impulse response functions of (log) TFP and (log) stock prices (point estimates and the 68% confidence bands computed with the bootstrap method).

8. Plot the percentage of forecast error variance of (log) TFP and (log) stock prices, (log) consumption and (log) GDP.

Solution Exercise 1:

The TFP data from Fernald's website is in annualized growth rates, computed as:

$$\Delta TFP_t = [\log(TFP_t) - \log(TFP_{t-1})] \times 400$$

To obtain our usual quarter-on-quarter measure of growth, we therefore divide the series by 4.

```
% TFP data
fernald_data = readtable('quarterly_tfp.xlsx',Sheet='quarterly');
tfp = fernald_data.dtfp_util/4; % We divide by 4 because the ...
    quarterly changes are annualized in the sheet (diff(log(x)))*400 ...
    is used
```

Next, notice that the measure of stock prices we want to use is the S & P 500 series from the FRED-QD data set of McCracken & Ng. This is the composite stock price index and we should deflate it in order to account for the price level. We do this by accounting for the GDP deflator:

$$\Delta SP_t = \left[\log\left(\frac{SP_t}{GDPDEF_t}\right) - \log\left(\frac{SP_{t-1}}{GDPDEF_{t-1}}\right)\right] \times 100$$

```
% Deflate the SP500 series
sp500 = fred_data(:,3)./ fred_data(:,4); % fourth series is GDP deflator

% Turn to growth rates
sp500gr = diff(log(sp500))*100;
```

We estimate a VAR(4) given by:

$$\begin{bmatrix}\Delta TFP_t \\ \Delta SP_t\end{bmatrix} = \begin{bmatrix}\mu_1 \\ \mu_2\end{bmatrix} + A_1 \begin{bmatrix}\Delta TFP_{t-1} \\ \Delta SP_{t-1}\end{bmatrix} + A_2 \begin{bmatrix}\Delta TFP_{t-2} \\ \Delta SP_{t-2}\end{bmatrix} + A_4 \begin{bmatrix}\Delta TFP_{t-3} \\ \Delta SP_{t-3}\end{bmatrix} + A_2 \begin{bmatrix}\Delta TFP_{t-4} \\ \Delta SP_{t-4}\end{bmatrix} + \begin{bmatrix}\epsilon_{1t} \\ \epsilon_{2t}\end{bmatrix}$$

The intuition for including these two particular time series in the VAR is given by Beaudry and Portier (2006) so as to have a measure of the current state of the economy (TFP growth) and a measure of what people (whoever they are) expect for the economy in the future (stock price growth).

We can estimate the VAR(4) with a constant using standard OLS techniques. In this case the dimension of Y will be $(T-p) \times 2$ and X will be $(T-p) \times (1+Np)$. Hence, the matrix of parameter estimates, let's call it $\hat{\beta} = (X'X)^{-1}X'Y$, will have dimensions $(1+Np) \times 2$.

```
tfp_gr = tfp(2:end);
y = [tfp_gr, sp500gr]; % if estimated in growth rates
% y = [cumsum(tfp), log(sp500)*100]; % if estimated in log-levels
[T,N] = size(y);

% Estimate the VAR(4)
p = 4;
c = 1;
[beta, residuals] = VAR(y,p,c);
disp('the maximum eigenvalue is')
disp(max(abs(eig(companionMatrix(beta,c,p)))))
```

Solution Exercise 2:

The first identification strategy requires short run restrictions. The news shock has no contemporaneous effect on TFP growth, but moves expectations. So we heard about some new technology development to arrive in the future, invest accordingly which drives prices today, but the actual arrival of the technology does not happen immediately. It diffuses slowly and only eventually affects TFP. We implement this with our usual Cholesky matrix as follows. First, order the VAR as above $y_t = [\Delta TFP, \Delta SP]'$. Then impose the short run restrictions on the Wold representation.

$$y_t = C(L)SS^{-1}\epsilon_t = D(L)w_t \quad w_t \sim \mathcal{WN}(0, I)$$

```
%% 2. Identify a news shock using contemporaneous restrictions
% Get the variance-covariance matrix
sigma = (residuals' * residuals)./(T-c-p-N*p);

% Compute Wold coefficients
horizon = 40;
wold = woldirf(beta,c,p,horizon);

% Compute the Cholesky factor of sigma
S = chol(sigma, 'lower');

% Estimate the Cholesky IRFs
cholirf = choleskyIRF(wold,S);
```

Solution Exercise 3:

The only slight complication is that the question asks us for log-level responses. This means we have to cumulate the responses. In MATLAB we do the following:

```
%% 3. Plot the IRFs and bootstrapped bands for the log level responses
% Bootstrap the confidence bands
prc = 68;
nboot = 1000;
cumulate = [1,2]; % if estimated in growth rates
% cumulate = []; % if estimated in log-levels
[bootchol, upper, lower, boot_beta] = ...
    bootstrapChol(y,p,c,beta,residuals,nboot,horizon,prc,cumulate);

% Plot Cholesky IRFs
varnames = {'TFP', 'SP500'};
shockname = "News Shock";
shock = 2;
plotchol(cholirf, varnames, shockname, cumulate, shock, upper, ...
    lower, prc)
```

We obtain the following plot of the IRF to the news shock with confidence bands.

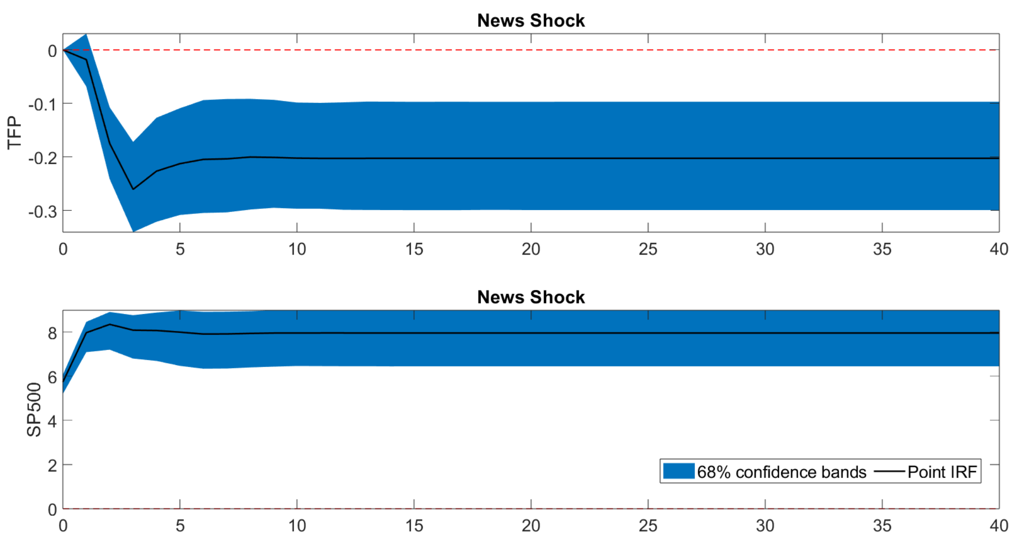

Figure 1: *Level IRFs with short-run identification.*

The confidence bands are computed using the usual bootstrapping technique with resampling from the model residuals. Here it looks like the news either did not reflect technology or the shock materialized in a different way than expected. So whatever drove stock prices up today ended up producing lower TFP in the future. The result contrasts with that of Beaudry & Portier (2006). They use a VECM which directly models the cointegrating relationship between stock prices and TFP. Typically, in such cases we would estimate the model in log-levels instead of growth rates. The problem in this small model is that both series are trending which would give nearly explosive eigenvalues of the companion form matrix. We can increase the lag order a bit and will have to increase the bootstrap sample size as well as the quantiles to make sure the IRF stays within bounds.

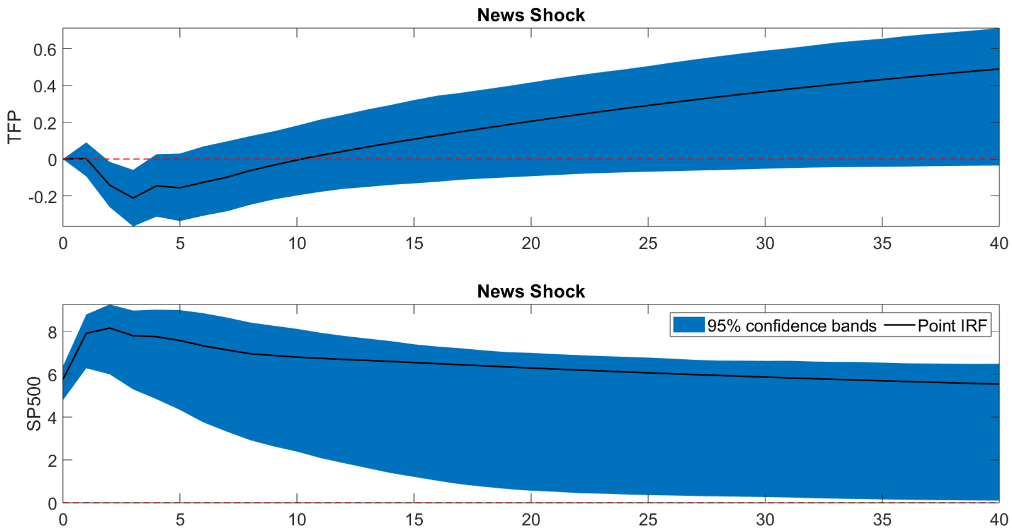

Figure 2: *Level IRFs with short-run identification estimated on a VAR(6) with level data instead of growth rates.*

Now we do see that TFP eventually increases. However, our OLS estimate bias kicks in hard due to the non-stationarity, especially as we increase the horizon. We will continue with the data in growth rates to practice the techniques, but we should be aware of this potential drawback.

Solution Exercise 4:

Here we compute the forecast error variance decomposition (FEVD) of the two variables due to the news shock and the other shock. Since we are asked for the level FEVD we use the cumulated IRFs in

$$FEVD_j^n = \frac{\sum_{h=0}^{H} IRF_{n,j,h}^2}{\sum_{k=1}^{K}\sum_{h=0}^{H} IRF_{n,k,h}^2}$$

This gives us the FEVD of the n^{th} variable for the j^{th} shock up to horizon H.

```
%% 4. Compute and plot the FEVD
level_irf = cumsum(cholirf,3); % if estimated in growth rates
% level_irf = cholirf;
fevd = variance_decomp(level_irf, shock);
```

```

% Plot the FEVD
plot_vardec(fevd, varnames, shockname)
```

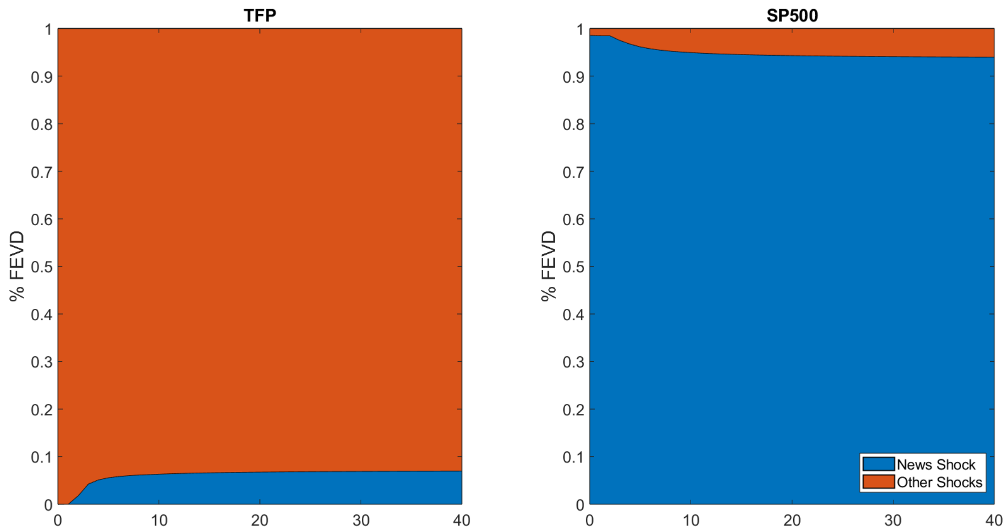

Figure 3: *Forecast Error Variance Decomposition Cholesky Level Shocks.*

The values at horizons 0,8,40 are

Horizon	TFP		S&P500	
	Other	News	Other	News
0	1	0	0.0141	0.9859
8	0.9381	0.0619	0.0472	0.9528
40	0.9296	0.0704	0.0597	0.9403

The News Shock drives most fluctuations in stock prices whereas it is hardly responsible for any fluctuations in log TFP at any horizon. Again, this is a sign of model problems, probably stemming from modeling everything in growth rates.

Solution Exercise 5:

Recall that there are two potential structural shocks in the model. With the long-run identification we want to impose that the IRF of log TFP at horizon 40 from one of these shocks is equal to zero. Here we impose that the second shock, associated to stock prices has no effect on the level response of TFP in the long run. Therefore, our long run IRF has to fulfill:

$$IRF_{40} = \begin{pmatrix} IRF^{s1}_{v1} & 0 \\ IRF^{s1}_{v2} & IRF^{s2}_{v2} \end{pmatrix}$$

As usual, we start with the Wold representation (here abstracting from the constant):

$$y_t = C(L)\epsilon_t$$

To implement this strategy we use the procedure of Blanchard & Quah (1989). The first bell that has to ring is **"the response on the log-level of TFP"**. So we have to work with cumulated IRFs $C(1) = C^{40}$ for our identification. Great! Now we have to find a matrix, let's call it K, that provides the zero-restriction above and whose inverse gives us that the structural shocks are white noise, i.e. $K^{-1}\epsilon_t \sim \mathcal{WN}(0, I)$.

Approach 1: You can use the approach of BQ directly, where the structural MA representation is given by:

$$y_t = C(L)K\eta_t$$

with $K = C(1)^{-1}S^*$, $S^* = chol(C(1)\Omega C(1)')$ and $\eta_t = K^{-1}\epsilon_t$. Notice that the long run responses in this notation are given by

$$C(1)K = C(1)C(1)^{-1}S^* = S^* = \begin{bmatrix} s^*_{11} & 0 \\ s^*_{21} & s^*_{22} \end{bmatrix}$$

since S^* is a lower triangular Cholesky factor. So this implements our restriction correctly. To see whether the shocks w_t are in fact white noise, consider

$$\begin{aligned} E[w_t] &= E[K^{-1}\epsilon_t] = K^{-1}E[\epsilon_t] = 0 \\ Var[w_t] &= E[K^{-1}\epsilon_t\epsilon_t'K^{-1'}] = S^{-1*}C(1)\Omega C(1)'S^{-1*'} \\ &= S^{-1*}S^*S^{*'}S^{-1*'} = I \end{aligned}$$

In `MATLAB` we do the following:

```
%% 5. Long-run identification
C1 = sum(wold,3); % If estimated in growth rates
% C1 = squeeze(wold(:,:,end));
K = C1\chol(C1*sigma*C1','lower');

% Compute the LR responses
irf_bq = bqIRF(wold, K);
```

where the function `bqIRF()` works very similarly to the function we had made for the Cholesky case.

```
function [bqirf] = bqIRF(wold, K, scaling)
[N,¬, horizon] = size(wold);
bqirf = zeros(N,N,horizon);

for h=1:horizon

    bqirf(:,:,h) = wold(:,:,h) * K;

end

if nargin > 2
    bqirf = bqirf ./ (bqirf(scaling(1),scaling(1),1) * 1/scaling(2));
end

end
```

Again, the function here has an input in case you may wish to scale the response to the shock of interest to have a certain size. Keep in mind that whenever you do such scaling, the shock does not have unit variance anymore which may mess up your FEVD!

This will give you $K = \begin{bmatrix} 0.7564 & 0.2121 \\ -0.8898 & 5.7193 \end{bmatrix}$.

Approach 2: Alternatively, you can implement the restrictions using the `fsolve()` nonlinear solver of `MATLAB`. We have four unknown elements in the matrix $K_{2\times2}$. So we need four equations to solve for the elements. For this, you set up a system of equations as follows: First, from the fact that $Var[K^{-1}\epsilon_t] = I$

$$Var[K^{-1}\epsilon_t] = K^{-1}\Omega K^{-1'} = I$$

which gives you three restrictions (the diagonal elements have to be equal to 1 and the symmetric off-diagonal element has to be zero). Second,

$$C^{40}K = \begin{pmatrix} IRF_{v1}^{s1} & 0 \\ IRF_{v2}^{s1} & IRF_{v2}^{s2} \end{pmatrix}$$

which gives you the last restriction, namely the zero restriction we imposed. Solving this system gives you the same matrix K as does the above approach. In `MATLAB` we do the following:

```
%% Approach 2: Alternative trial for long-run restriction
% Using fsolve for a system of non-linear equations K

options=optimset('Display', 'off', ...
    'TolFun',.0000000001,'MaxIter',100000,'MaxFunEvals',100000); % ...
    more iterations

```

```
C1=sum(C,3); % sum the IRFs across the horizons
start = [1,-2,3,4];
sol = fsolve(@(k) mysystem(k,sigma,C1), start, options);
K1 = [sol(1), sol(2); sol(3), sol(4)];
```

The system to solve looks like this in MATLAB:

```
function [sys] = mysystem(k,Omega,C1)

    K = [k(1), k(2); k(3), k(4)];

    n = size(Omega,2);
    first = inv(K)*Omega*inv(K)' - eye(n); % 2 by 2 matrix
    f1 = first(1,1);
    f2 = first(1,2);
    f3 = first(2,2);
    second = C1(1,:)*K(:,2); % scalar
    f4 = second;
    sys = [f1;f2;f3;f4];

end
```

Approach 3: Finally, you can also use the more general way you saw as the **orthonormal representation** of y_t. This is given by

$$y_t = C(L)SHH'S^{-1}\epsilon_t = H(L)u_t$$

where $HH' = H'H = I$, $S = chol(\Omega)$, $H(L) = C(L)SH$ and $u_t = H'S^{-1}\epsilon_t$. We have to pin down the matrix H such that it fulfills our restriction that the long run impact of the second shock on the first variable (TFP) in the level is zero. Define $D(1) = C(1)S = D^{40}$. The first restriction for the nonlinear system is given by

$$D^{40}H = \begin{pmatrix} IRF^{s1}_{v1} & 0 \\ IRF^{s1}_{v2} & IRF^{s2}_{v2} \end{pmatrix}$$

and the remaining three of the restrictions come from orthogonality of H: $H'H = I$. So again, we have four restrictions for four unknowns. We can solve this similarly to the above using `fsolve()`. This will again give you the same result.

```
%% Third Approach: Using the orthonormal representaiton yt = ...
    C(L)SHH'S^{-1}et

D1 = sum(cholirf,3);
start = [1,-2,3,4];
sol1 = fsolve(@(h) mysystem1(h,D1), start, options);
H = [sol1(1), sol1(2); sol1(3), sol1(4)];
K2 = S * H;
```

All the approaches yield the following level IRFs (column 1 is response to technology shock):

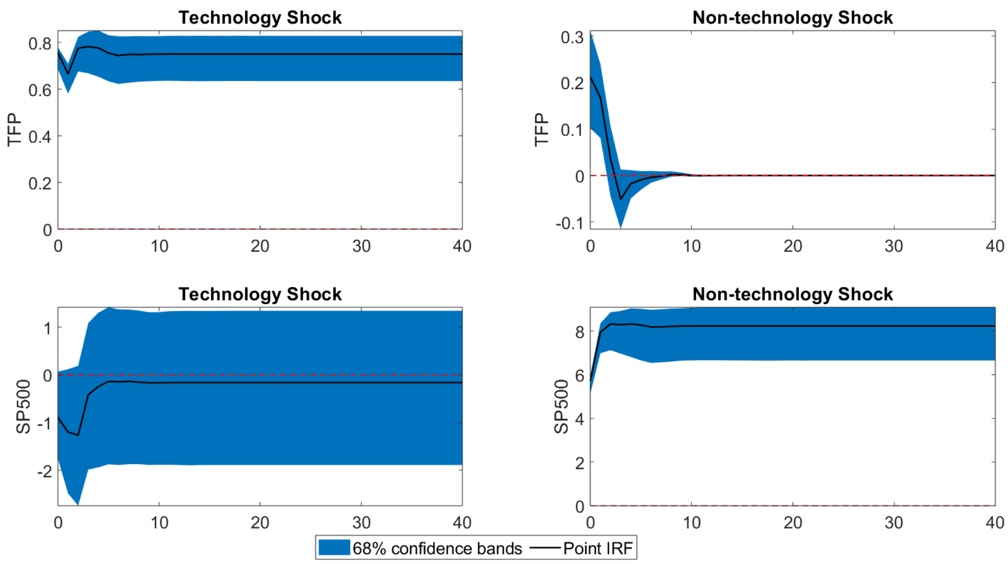

Figure 4: *Level IRFs with long-run identification.*

We see again a discrepancy with respect to the result of Beaudry & Portier (2006). They argue that the news shock from the Cholesky exercise and the technology shock from the BQ exercise produce in essence the same IRFs. Our exercise shows an immediate reaction in TFP that is ruled out in the Cholesky case. Therefore, we cannot expect the news and the technology shock to be very correlated for our model in growth rates.

Computing the correlation between the two shocks:
We have identified a news shock as the second vector in $w_t = S^{-1}\epsilon_t$ and a technology shock as the second vector in $\eta_t = K^{-1}\epsilon_t$. In Beaudry and Portier (2006) the correlation is very high at 0.97, suggesting that the shock is nearly the same in both cases. The correlation we get is not as high and negative at -0.2700. This suggests that we have two different shocks, as distinct from the original paper. You can find some replication code here Replication Codes Beaudry and Portier. Note that they use a VECM model as the variables TFP and SP500 are cointegrated. The VECM approach uses the variables in growth rates and corrects for the error arising from omitting the cointegrating relationship. Alternatively, we could estimate the model in log-levels. This would approximate the cointegrating relationship, but we have to be careful since OLS is downward-biased, the long-term IRFs will be poorly estimated and we are imposing identifying restrictions on the long run.

When we do the two exercises in log-levels, we do indeed find a higher positive correlation of 0.6512:

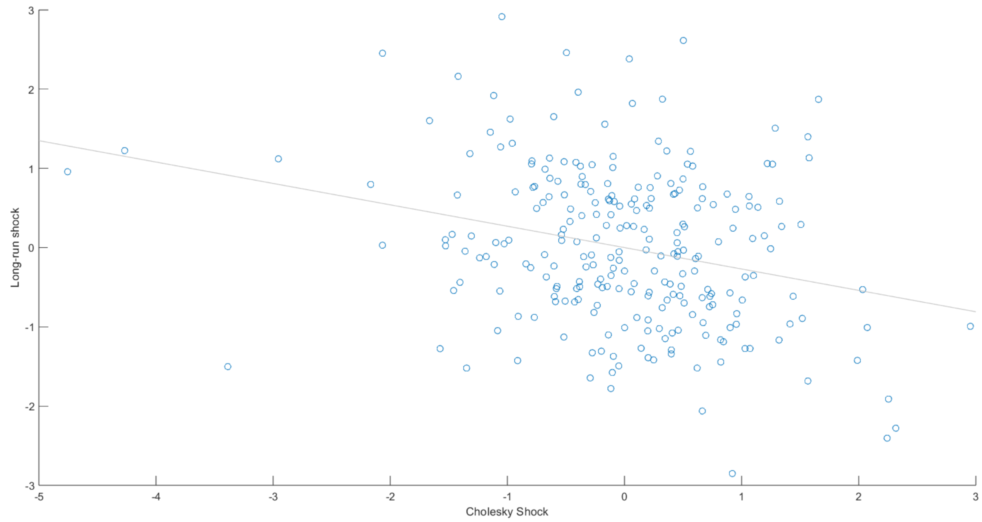

Figure 5: *Correlation between news shock and technology shock.*

Solution Exercise 6:

Now we augment the bivariate system with two additional variables, the growth rate of real consumption and the growth rate of real GDP. Now $N = 4$:

$$y_t = \begin{bmatrix} \Delta TFP \\ \Delta SP \\ \Delta C \\ \Delta Y \end{bmatrix}$$

For the identification we have one short run restriction *(The news shock has no contemporaneous effect on log TFP)* and a long run restriction *(The news shock has maximum effect on TFP at horizon 40)*. We will implement both restrictions using the **orthonormal representation**. Just to repeat, this is given by

$$y_t = C(L)SHH'S^{-1}\epsilon_t = H(L)u_t$$

We will use **partial identification**, since we are only interested in one shock. This means we have to restrict **only one column** h_j **of** H. The IRFs will then be given by

$$IRF = C(L)Sh_j$$

In principle, any column will do. All elements of the above equation can be obtained from OLS and the Wold coefficients, all we need is the vector h_j. Hence, we start by estimating and inverting as well as computing the Cholesky matrix S for the new larger system:

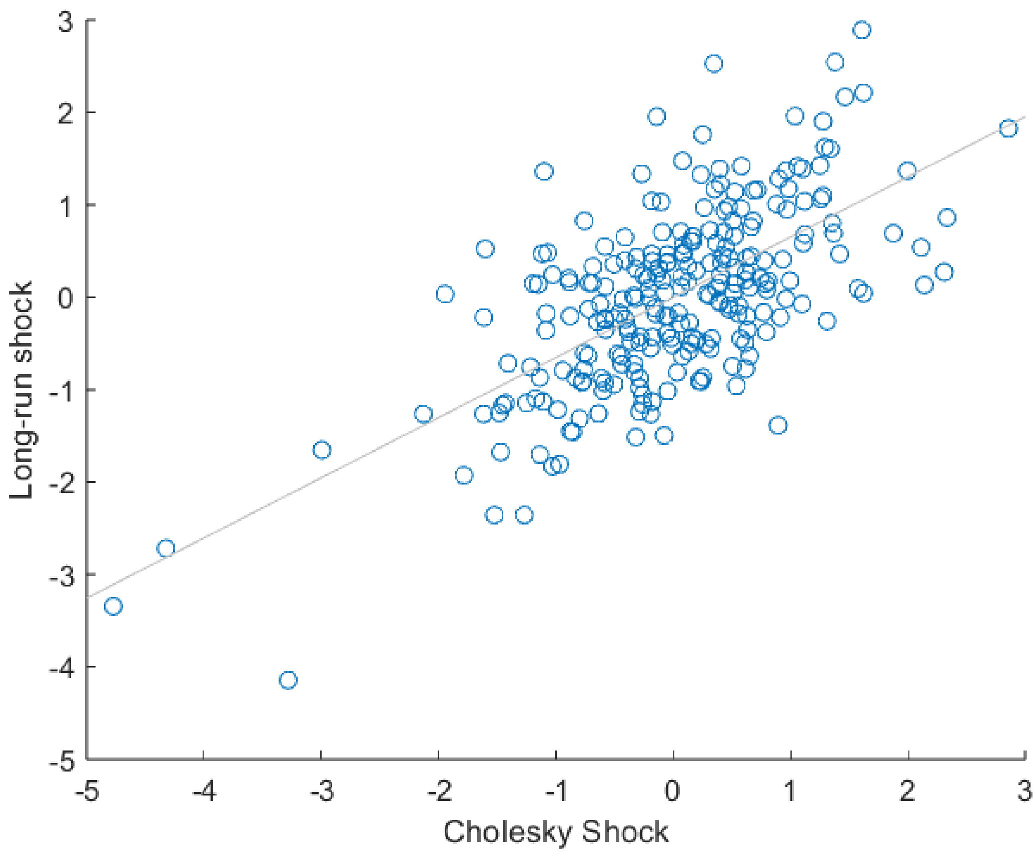

Figure 6: *Correlation between news shock and technology shock if model estimated in levels.*

```
% Create the growth rates of GDP and Consumption
GDP_gr = diff(log(fred_data(:,1)))*100;
CONS_gr = diff(log(fred_data(:,2)))*100;
y = [tfp_gr, sp500gr, GDP_gr, CONS_gr];
[T,N] = size(y);

% Estimate the VAR(4)
[beta, residuals] = VAR(y,p,c);
disp('the maximum eigenvalue is')
disp(max(abs(eig(companionMatrix(beta,c,p)))))

% Get the variance-covariance matrix
sigma = (residuals' * residuals)./(T-c-p-N*p);

% Compute Wold coefficients
wold = woldirf(beta,c,p,horizon);
```

```

% Compute the Cholesky factor of sigma
S = chol(sigma, 'lower');

% Estimate the Cholesky IRFs
cholirf = choleskyIRF(wold,S);
```

Start with the implementation of the first assumption.

1. The news shock has no contemporaneous effect on log TFP.

Contemporaneous effects are given by $I \times S \times H$. Since this is the very first response, there is no difference between level and growth rate responses. Hence, the short run restriction comes from:

$$\begin{pmatrix} s_{11} & 0 & 0 & 0 \\ s_{21} & s_{22} & 0 & 0 \\ s_{31} & s_{32} & s_{33} & 0 \\ s_{41} & s_{42} & s_{43} & s_{44} \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \end{pmatrix} = \begin{pmatrix} 0 \\ IRF0^{sj}_{v2} \\ IRF0^{sj}_{v3} \\ IRF0^{sj}_{v4} \end{pmatrix}$$

We know that S is a Cholesky factor of Σ. Cholesky factors always have real positive elements on the diagonal. Hence, $s_{11} \neq 0$. For the IRF of log TFP to be unaffected by our news shock it must therefore be that

$$s_{11}h_1 = 0 \Leftrightarrow h_1 = 0$$

So 1 element of h_j is identified. Let's turn to the long run restriction to get the other 3 restrictions.

2. The news shock has maximum effect on TFP at horizon 40.

The long run responses of the log of TFP to the news shock are captured by the top element of $H(1)$

$$H(1) = D(1)h_j = \begin{pmatrix} D^{40}_{11} & D^{40}_{12} & D^{40}_{13} & D^{40}_{14} \\ D^{40}_{21} & D^{40}_{22} & D^{40}_{23} & D^{40}_{24} \\ D^{40}_{31} & D^{40}_{32} & D^{40}_{33} & D^{40}_{34} \\ D^{40}_{41} & D^{40}_{42} & D^{40}_{43} & D^{40}_{44} \end{pmatrix} \begin{pmatrix} 0 \\ h_2 \\ h_3 \\ h_4 \end{pmatrix} = \begin{pmatrix} max \\ IRF40^{sj}_{v2} \\ IRF40^{sj}_{v3} \\ IRF40^{sj}_{v4} \end{pmatrix}$$

Again, we take D(1) to be the cumulated response of the Cholesky shocks. We have the following maximization program:

$$\begin{aligned} max_{h_2,h_3,h_4} \quad & D^{40}_{12}h_2 + D^{40}_{13}h_3 + D^{40}_{14}h_4 && (1) \\ s.t. \quad & h_2^2 + h_3^2 + h_4^2 = 1 && (2) \end{aligned}$$

Remember that in addition to maximization we have to have $||h_j||= 1$, i.e. the length of the column h_j must be equal to one to fulfill the property of the orthogonality matrix. The following routines can be used in MATLAB:

1. Directly without the explicit constraint:

```
% Using fminsearch
var = 1; %TFP is ordered first
startvalue=ones(3,1)./norm(ones(3,1));
h_est = fminsearch(@(h) ...
    lr_target(h,cholirf,var),startvalue,options);
h = [0;h_est./norm(h_est)];

function obj = lr_target(h, cholirf, tfp_pos)

    h = h./norm(h); % Impose that h has unit length
    h1 = [0; h];    % Impose that the initial response (C0Sh = ...
        ISh = Sh) delivers a zero response of TFP

    lr = sum(cholirf,3)*h1;      % The LR response is C(1)Sh
    obj = -lr(tfp_pos);          % The routine is for ...
        minimization and we want a maximum

end
```

2. Using a constrained optimization set up

```
%% Second approach: fmincon
h_est = fmincon(@(h) ...
    lr_target1(h,cholirf,var),startvalue,[],[],[],[],[],[],@constraint);
h1 = [0;h_est./norm(h_est)];

function obj = lr_target1(h, cholirf, tfp_pos)

    h1 = [0; h];    % Impose that the initial response (C0Sh = ...
        ISh = Sh) delivers a zero response of TFP

    lr = sum(cholirf,3)*h1;      % The LR response is C(1)Sh
    obj = -lr(tfp_pos);          % The routine is for ...
        minimization and we want a maximum

end

function [c, ceq] = constraint(h)

    c = [];
    ceq = h'*h - 1;

end
```

3. Using Givens rotations (see last section of this document for more details on this).

```
% Using the rotation matrices
startvalue=zeros(2,1);
theta_est=fminunc(@(theta) -h_vec_max(theta,D),startvalue,options);

theta_13=theta_est(1);
theta_14=theta_est(2);

h=[0;-cos(theta_13)*cos(theta_14); -sin(theta_13)*cos(theta_14); ...
    -sin(theta_14)];

function max_effect=h_vec_max(theta,D)

theta_13=theta(1);
theta_14=theta(2);

h=[0;-cos(theta_13)*cos(theta_14); -sin(theta_13)*cos(theta_14); ...
    -sin(theta_14)];

% Recall that we want the maximum response on the level, so we ...
    need to take
% cumsum
D_last=cumsum(D(:,:,:),3);

% The target function is C_40
max_effect=squeeze(D_last(1,:,41))*h;


end
```

All of the procedures yield the same h_j:

$$h_j = \begin{bmatrix} 0 \\ -0.6278 \\ 0.7470 \\ 0.2187 \end{bmatrix}$$

Solution Exercise 7:

Now we can implement our steps to get the structural IRFs with short and long run restrictions.

1. Estimate the four variable VAR(4) with a constant.

2. Compute Wold IRFs.

3. Compute Cholesky IRFs.

4. Partially identify the news shock.

5. Compute cumulated IRFs and bootstrap confidence bands applying the identification procedure.

```
%% Compute point IRFs
function irf = partial_irf(cholirf,h)

    hor = size(cholirf,3);
    irf = zeros(size(cholirf,1), hor);

    for i = 1:hor
        irf(:,i) = cholirf(:,:,i)*h;
    end

end
```

Cumulation will be done inside the plotting function.

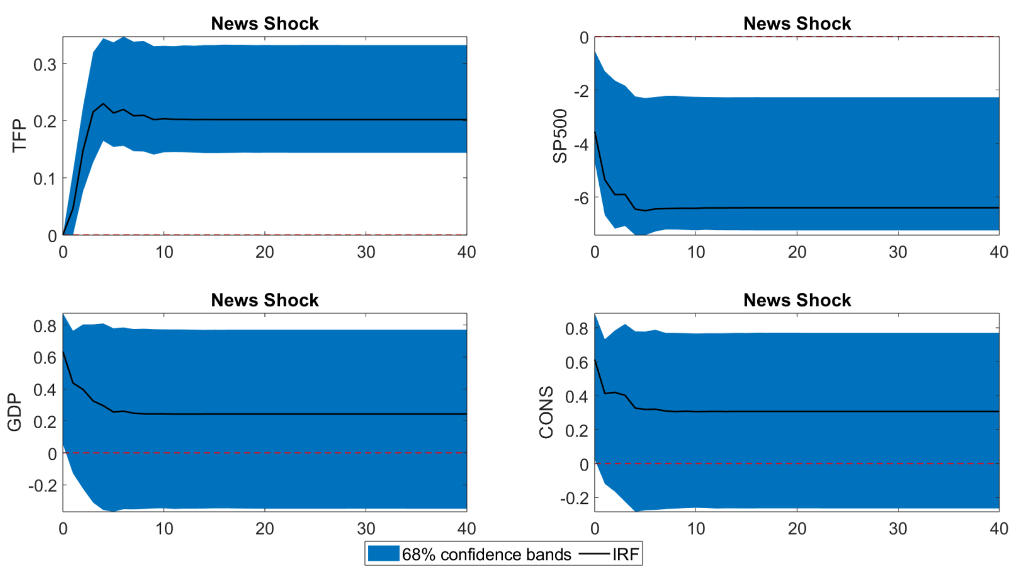

Figure 7: *Level IRFs to news shock with short and long run restrictions.*

The news shock increases TFP in the long run and both consumption and GDP increase, although non-significantly, so the shock is expansionary. Stock prices fall, very similar to the approach in the Cholesky case. Notice that again, if we estimate in log levels, the results are more sensible:

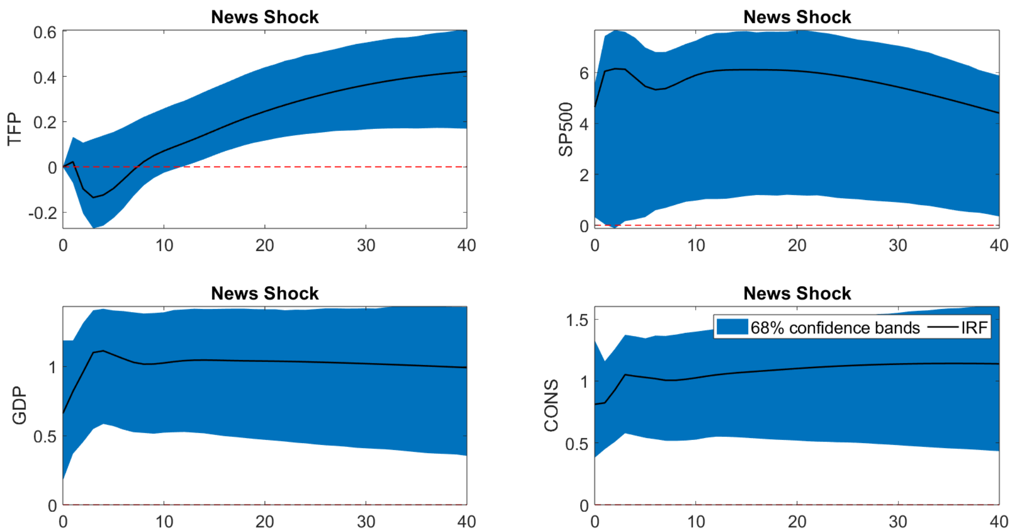

Figure 8: *Level IRFs to news shock with short and long run restrictions. VAR(4) estimated in log levels.*

Solution Exercise 8:

Notice that in the previous steps we only identified one shock, because we used partial identification. That means that we cannot compute the denominator in

$$FEVD_j^n = \frac{\sum_{h=0}^{H} IRF_{n,j,h}^2}{\sum_{k=1}^{K}\sum_{h=0}^{H} IRF_{n,k,h}^2}$$

from our maximization identification. For that, we would need the IRFs for all shocks, but we have only one. As a substitute, we can use the total variation from the Wold shocks.

```
%% 8. Perform forecast error variance decomposition

% Compute the total variation
irf_chol_cumu = cumsum(cholirf,3);
tot_var = squeeze(cumsum(sum((irf_chol_cumu.^2),2),3));

% Cumulate the IRF from the Shock
irf_cumu = cumsum(max_irf,2);
fevd = cumsum((irf_cumu.^2),2)./tot_var;
plot_vardec(fevd,varnames, shockname)

% Total variation from Wold
```

```
wold_cumu = cumsum(wold,3);
tot_var_wold = zeros(N,horizon+1);
temp = 0;
for hh = 1:horizon+1
    temp = temp + wold_cumu(:,:,hh)*sigma*wold_cumu(:,:,hh)';
    tot_var_wold(:,hh) = diag(temp);
end
```

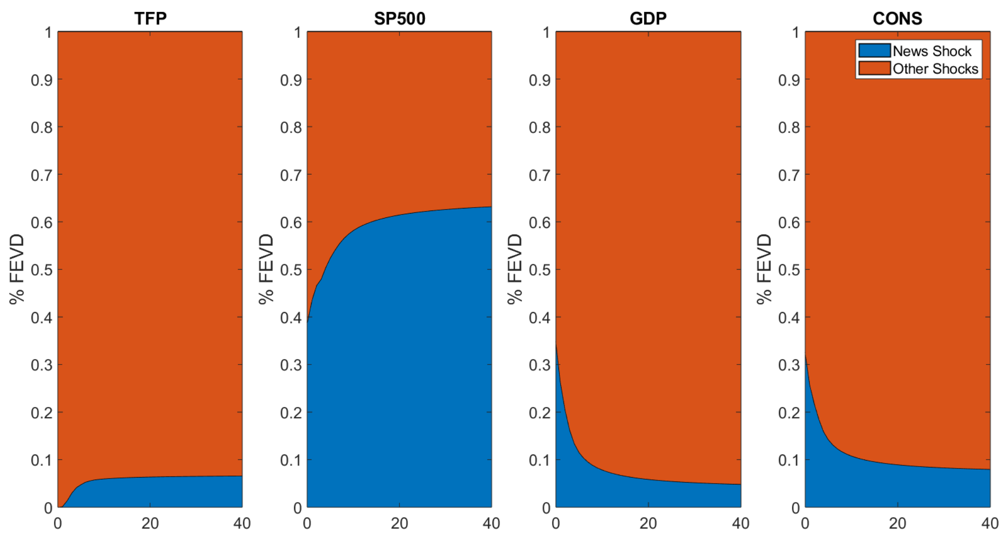

Figure 9: *Forecast Error Variance Decomposition short and long run identified level news shock.*

The values at horizons 0,8,40 are

Horizon	TFP		S&P500		GDP		C	
	Other	News	Other	News	Other	News	Other	News
0	1	0	0.6114	0.3886	0.6506	0.3494	0.6738	0.3262
8	0.9420	0.0580	0.4340	0.5660	0.9117	0.0883	0.8832	0.1168
40	0.9338	0.0662	0.3678	0.6322	0.9515	0.0485	0.9200	0.0800

Our news shock is not a very important driver of the business cylce as it explains only small variation in both GDP and consumption. However, we may have identified something other than a technological news shock since TFP is not driven very heavily by the shock. Again this could be due to the omitted cointegrating relationship.

Using rotation matrices for Exercise 6:

In theory, we should parameterize the matrix H in order to include orthogonality restrictions by using rotation matrices. The rotation matrix H is found by the product of 6 rotation matrices (we needed n(n-1)/2 matrices) in our case. The six matrices that compose H as presented below:

$$H_{1,2} = \begin{bmatrix} cos(\theta_{1,2}) & sin(\theta_{1,2}) & 0 & 0 \\ -sin(\theta_{1,2}) & cos(\theta_{1,2}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} H_{1,3} = \begin{bmatrix} cos(\theta_{1,3}) & 0 & sin(\theta_{1,3}) & 0 \\ 0 & 1 & 0 & 0 \\ -sin(\theta_{1,3}) & 0 & cos(\theta_{1,3}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_{1,4} = \begin{bmatrix} cos(\theta_{1,4}) & 0 & 0 & sin(\theta_{1,4}) \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -sin(\theta_{1,4}) & 0 & 0 & cos(\theta_{1,4}) \end{bmatrix} H_{2,3} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos(\theta_{2,3}) & sin(\theta_{2,3}) & 0 \\ 0 & -sin(\theta_{2,3}) & cos(\theta_{2,3}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_{2,4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos(\theta_{2,4}) & 0 & sin(\theta_{2,4}) \\ 0 & 0 & 1 & 0 \\ 0 & -sin(\theta_{2,4}) & 0 & cos(\theta_{2,4}) \end{bmatrix} H_{3,4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & cos(\theta_{3,4}) & sin(\theta_{3,4}) \\ 0 & 0 & -sin(\theta_{2,4}) & cos(\theta_{2,4}) \end{bmatrix}$$

$$H = H_{1,2} * H_{1,3} * H_{1,4} * H_{2,3} * H_{2,4} * H_{3,4}$$

If you multiply this out the first (and simplest) column of H turns out to be

$$h_1 = \begin{pmatrix} cos(\theta_{1,2})cos(\theta_{1,3})cos(\theta_{1,4}) \\ -cos(\theta_{1,3})cos(\theta_{1,4})sin(\theta_{1,2}) \\ -cos(\theta_{1,4})sin(\theta_{1,3}) \\ -sin(\theta_{1,4}) \end{pmatrix}$$

To implement the first restriction, i.e. that the first element of $h_1 = 0$ we can set $\theta_{1,2} = \frac{\pi}{2}$. This will give us

$$h_1 = \begin{pmatrix} 0 \\ -cos(\theta_{1,3})cos(\theta_{1,4}) \\ -cos(\theta_{1,4})sin(\theta_{1,3}) \\ -sin(\theta_{1,4}) \end{pmatrix}$$

Now we have to run a different maximization program to before. The orthogonality constraints are already implemented and all that is left is to

$$max_{\theta_{1,3},\theta_{1,4}} \quad - D^{40}_{12}cos(\theta_{1,3})cos(\theta_{1,4}) - D^{40}_{13}cos(\theta_{1,4})sin(\theta_{1,3}) - D^{40}_{14}sin(\theta_{1,4})$$

This will give you the same results as the constrained optimization problem from before.