Macroeconometrics: Assignment 4

Andreas Fischeneder, Mi Liu, Nirai Tomass

14/07/2022

Exercise 1

Q: Re-estimate the model for the `usmacro` dataset using different prior values for the prior variance (i.e. vary λ_1 and λ_2). Interpret changes in the various outputs of the model. In particular, discuss changes in the estimated autoregressive coefficients, impulse reponses, and forecasts — then explain how they may relate to the prior setup. Discuss why higher/lower values of λ_1 and λ_2 might have an impact on the shape of impulse responses.

```
# autoregressive coefs
ar_coefs <- as.data.frame(matrix(NA, nrow=7, ncol=3))
for (ii in 1:M) {
  for (jj in 1:k) {
    ar_coefs[jj,ii] <- paste0(round(mean(A_store[,jj,ii]),3), " [", round(quantile(A_store[,jj,ii], 0.0
  }
}
ar_coefs1 <- as.data.frame(matrix(NA, nrow=7, ncol=3))
for (ii in 1:M) {
  for (jj in 1:k) {
    ar_coefs1[jj,ii] <- paste0(round(mean(A_store1[,jj,ii]),3), " [", round(quantile(A_store1[,jj,ii], 
  }
}
tab.dat <- ar_coefs
tab.dat <- cbind(c(paste0(colnames(Y), 1), paste0(colnames(Y), 2), "constant"), tab.dat)
names(tab.dat) <- c("variable", colnames(Y))
tab.dat %>% kbl() %>%
        kable_styling(full_width = F) %>% kable_classic() %>%
        add_header_above(c(" " = 1, "Prior: lambda_1=0.1, lambda_2=0.2" = 3)) %>%
        row_spec(0, bold = T) %>% column_spec(1, bold=T)

tab.dat <- ar_coefs1
tab.dat <- cbind(c(paste0(colnames(Y), 1), paste0(colnames(Y), 2), "constant"), tab.dat)
names(tab.dat) <- c("variable", colnames(Y))
tab.dat %>% kbl() %>%
        kable_styling(full_width = F) %>% kable_classic() %>%
        add_header_above(c(" " = 1, "Prior: lambda_1=10, lambda_2=20" = 3)) %>%
        row_spec(0, bold = T) %>% column_spec(1, bold=T)
```

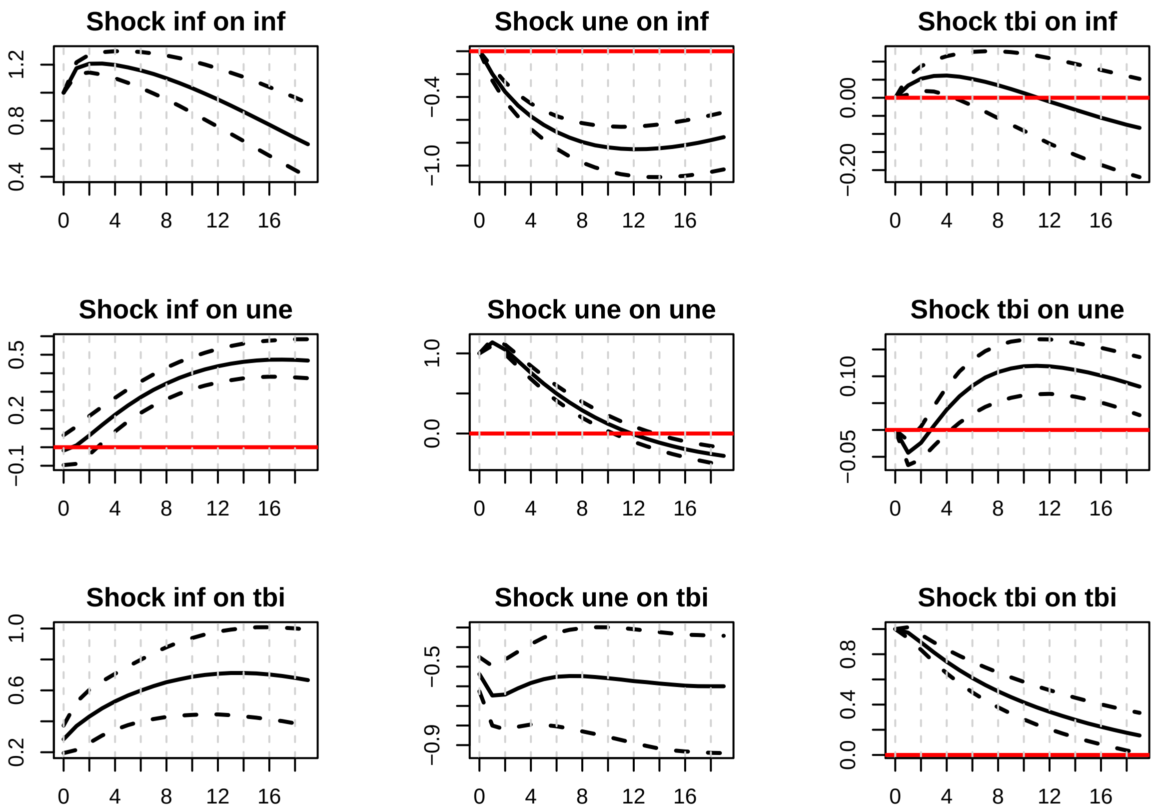
To compute the models, we adapted the code file from MS Teams (`BayesianVARs.R`). We estimated the same VAR model with two lags twice for two different priors for the variance. For the first set-up, we used the usual Minnesota specification with $\lambda_1 = 0.1$ and $\lambda_2 = 0.2$. For the second prior set-up, we increased both

variable	Prior: lambda_1=0.1, lambda_2=0.2		
	inf	**une**	**tbi**
inf1	1.163 [1.084; 1.242]	0.041 [-0.061; 0.143]	0.092 [-0.158; 0.346]
une1	-0.18 [-0.3; -0.063]	1.114 [1.033; 1.194]	-0.124 [-0.39; 0.143]
tbi1	0.034 [-0.015; 0.084]	-0.043 [-0.088; 0.003]	0.971 [0.884; 1.056]
inf2	-0.161 [-0.239; -0.083]	-0.003 [-0.106; 0.102]	-0.017 [-0.274; 0.236]
une2	0.087 [-0.026; 0.201]	-0.206 [-0.281; -0.13]	0.116 [-0.138; 0.367]
tbi2	-0.028 [-0.076; 0.02]	0.063 [0.019; 0.107]	-0.058 [-0.134; 0.019]
constant	0.501 [0.315; 0.685]	0.291 [0.109; 0.473]	0.255 [-0.196; 0.706]

variable	Prior: lambda_1=10, lambda_2=20		
	inf	**une**	**tbi**
inf1	1.525 [1.401; 1.649]	0.018 [-0.095; 0.133]	0.293 [-0.004; 0.599]
une1	-0.205 [-0.342; -0.069]	1.491 [1.363; 1.62]	-0.508 [-0.842; -0.17]
tbi1	0.014 [-0.05; 0.081]	-0.009 [-0.07; 0.053]	1.005 [0.84; 1.165]
inf2	-0.532 [-0.659; -0.406]	-0.006 [-0.123; 0.11]	-0.19 [-0.502; 0.113]
une2	0.159 [0.029; 0.291]	-0.58 [-0.702; -0.457]	0.53 [0.205; 0.85]
tbi2	-0.011 [-0.076; 0.052]	0.041 [-0.019; 0.102]	-0.114 [-0.272; 0.046]
constant	0.282 [0.101; 0.459]	0.3 [0.127; 0.472]	0.107 [-0.343; 0.561]

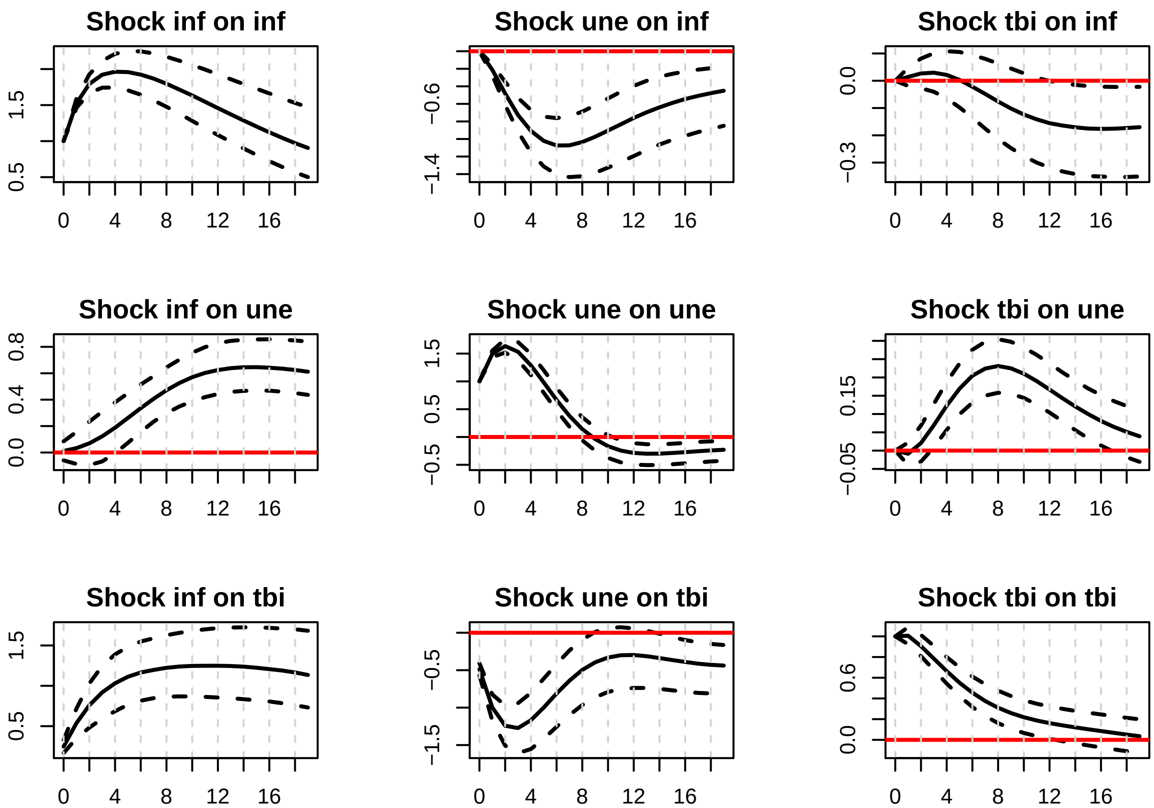
parameters by factor 100, i.e. $\lambda_1 = 10$ and $\lambda_2 = 20$. The table above summarizes the average autoregressive parameters together with their 95% CI based on the posterior draws. What is apparent is that the first set-up with much smaller values for λ_1 and λ_2 induces more shrinkage to the prior of the autoregressive parameters, set to 1 for the own first lag and 0 for all cross lags and all own lags for p>1. From the table above it can be seen that all first lags of the dependent variable itself are closer to 1 for the first prior set-up, and for almost all cross-lag and the second lags of the dependent variable, the autoregressive parameters are closer to 0. Furthermore, in many cases the 95% CI of the autoregressive parameter is smaller with the lower λ prior compared to the higher prior.

```r
# IRFs
IRFchol_low    <- apply(IRFchol_store, c(2,3,4), quantile, 0.16,na.rm=TRUE)
IRFchol_high   <- apply(IRFchol_store, c(2,3,4), quantile, 0.84,na.rm=TRUE)
IRFchol_median <- apply(IRFchol_store, c(2,3,4), median, na.rm=TRUE)
IRFchol_low1    <- apply(IRFchol_store1, c(2,3,4), quantile, 0.16,na.rm=TRUE)
IRFchol_high1   <- apply(IRFchol_store1, c(2,3,4), quantile, 0.84,na.rm=TRUE)
IRFchol_median1 <- apply(IRFchol_store1, c(2,3,4), median, na.rm=TRUE)
par(mfrow=c(3,3),mar=c(4,4,2,2))
for(ii in 1:M){
  for(jj in 1:M){
    min1 <- min(IRFchol_low[ii,jj,])
    max1 <- max(IRFchol_high[ii,jj,])
    plot.ts(IRFchol_median[ii,jj,], ylab="", main=paste0("Shock ",colnames(Y)[jj], " on ",colnames(Y)[i
    lines(IRFchol_low[ii,jj,], lty = 2, lwd=2)
    lines(IRFchol_high[ii,jj,], lty = 2, lwd=2)
    abline(h=0,col="red",lwd=2)
    abline(v=seq(1,nhor,by=2), col="lightgrey", lty=2)
    axis(1, at=seq(1,nhor,by=2), labels=seq(0,nhor-1,by=2))
  }
}
mtext("Prior: Lambda_1=0.1, Lambda_2=0.2", side = 1, line = -1.1, outer = TRUE)
```



Prior: Lambda_1=0.1, Lambda_2=0.2

```
par(mfrow=c(3,3),mar=c(4,4,2,2))
for(ii in 1:M){
  for(jj in 1:M){
    min1 <- min(IRFchol_low1[ii,jj,])
    max1 <- max(IRFchol_high1[ii,jj,])
    plot.ts(IRFchol_median1[ii,jj,], ylab="", main=paste0("Shock ",colnames(Y)[jj], " on ",colnames(Y)[
    lines(IRFchol_low1[ii,jj,], lty = 2, lwd=2)
    lines(IRFchol_high1[ii,jj,], lty = 2, lwd=2)
    abline(h=0,col="red",lwd=2)
    abline(v=seq(1,nhor,by=2), col="lightgrey", lty=2)
    axis(1, at=seq(1,nhor,by=2), labels=seq(0,nhor-1,by=2))
  }
}
mtext("Prior: Lambda_1=10, Lambda_2=20", side = 1, line = -1.1, outer = TRUE)
```

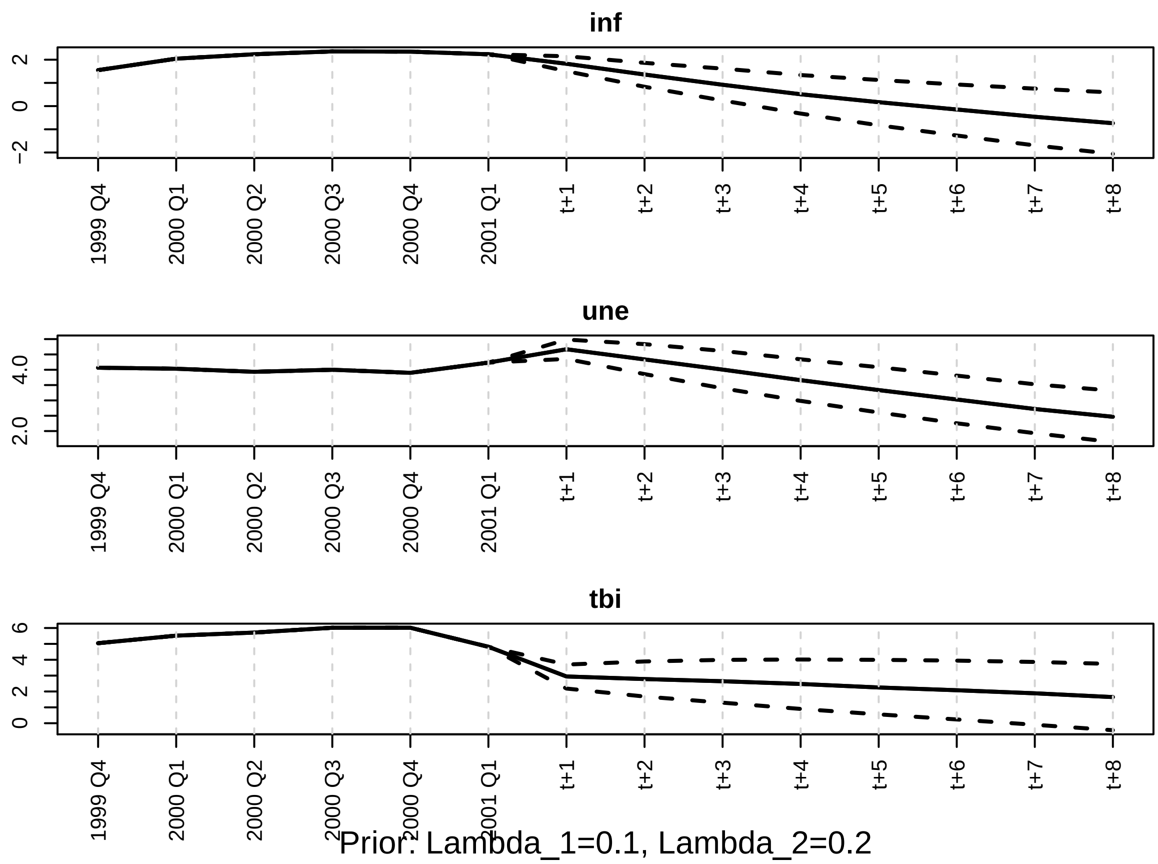


Prior: Lambda_1=10, Lambda_2=20

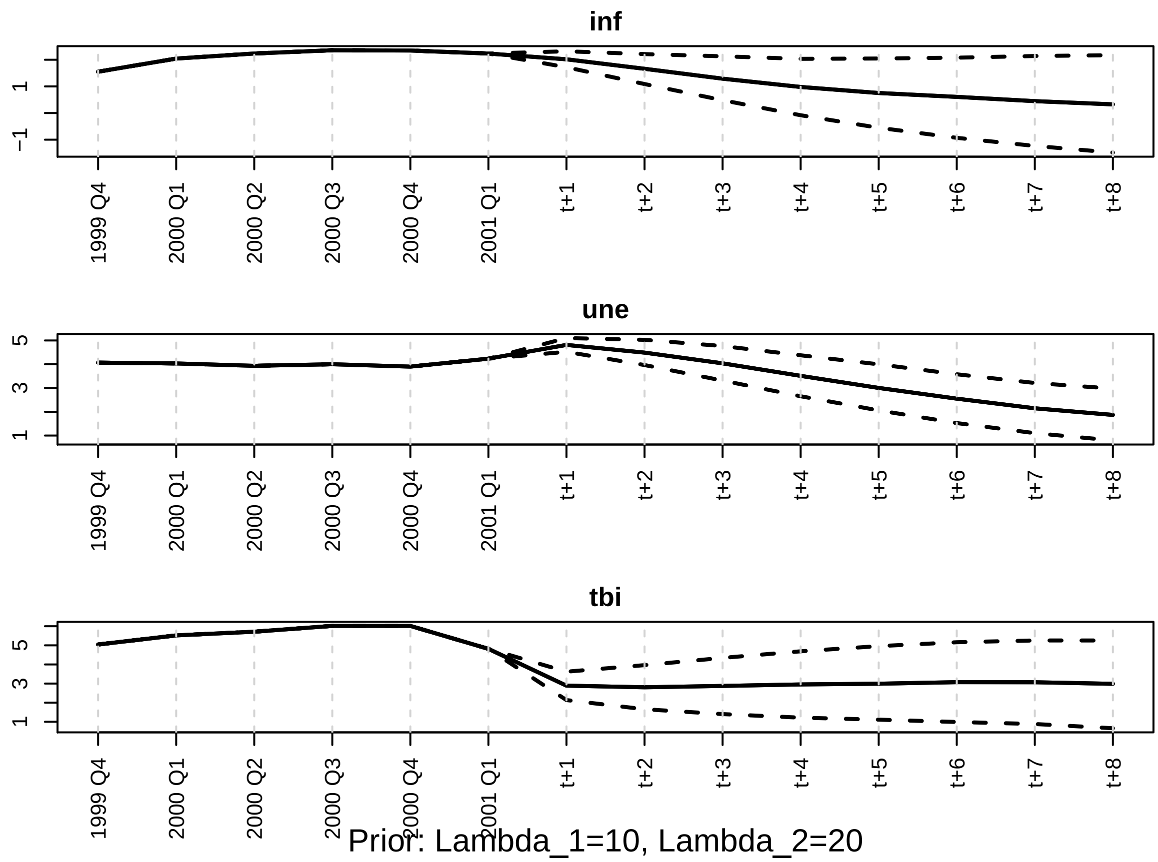
The IRFs for the two different model set-ups are depicted in the plots above. The reaction of the dependent variables to a given shock are actually quite similar for both priors, although there are some differences in the magnitude of the reactions as well as the confidence intervals of the IRFs. The lower we set the values for λ_1 and λ_2, the more shrinkage we induce to our prior for the autoregressive coefficients. As our prior for these coefficients basically implies that all variables follow a random walk, this means that the posterior is more similiar to a random walk the smaller the values for λ_1 and λ_2 are. Therefore, the IRFs become more an more like the IRF of a random walk, which means that it gets pulled slower towards zero.

```r
# Forecasts
yf_low    <- apply(yf_store, c(2,3), quantile, 0.16, na.rm=TRUE)
yf_median <- apply(yf_store, c(2,3), quantile, 0.50, na.rm=TRUE)
yf_high   <- apply(yf_store, c(2,3), quantile, 0.84, na.rm=TRUE)
yf_low    <- cbind(t(Yraw[(bigT-5):bigT,]),yf_low)
yf_median <- cbind(t(Yraw[(bigT-5):bigT,]),yf_median)
yf_high   <- cbind(t(Yraw[(bigT-5):bigT,]),yf_high)
xax <- c(as.character(time[(bigT-5):bigT]),paste0("t+",seq(1,fhorz)))

yf_low1    <- apply(yf_store1, c(2,3), quantile, 0.16, na.rm=TRUE)
yf_median1 <- apply(yf_store1, c(2,3), quantile, 0.50, na.rm=TRUE)
yf_high1   <- apply(yf_store1, c(2,3), quantile, 0.84, na.rm=TRUE)
yf_low1    <- cbind(t(Yraw[(bigT-5):bigT,]),yf_low1)
yf_median1 <- cbind(t(Yraw[(bigT-5):bigT,]),yf_median1)
yf_high1   <- cbind(t(Yraw[(bigT-5):bigT,]),yf_high1)
xax1 <- c(as.character(time[(bigT-5):bigT]),paste0("t+",seq(1,fhorz)))

par(mfrow=c(3,1),mar=c(5,3,2,3))
```

```
for(ii in 1:M){
  min1 <- min(yf_low[ii,])
  max1 <- max(yf_high[ii,])
  plot.ts(yf_median[ii,], ylim=c(min1,max1), main=colnames(Y)[[ii]], ylab="", xlab="", xaxt="n",lwd=2)
  lines(yf_low[ii,], lty=2,lwd=2)
  lines(yf_high[ii,], lty=2,lwd=2)
  lines(yf_median[ii,], lty=1,lwd=2)
  axis(1, at=seq(1,14), labels=xax, las=2)
  abline(v=seq(1,14), col="lightgrey", lty=2)
}
mtext("Prior: Lambda_1=0.1, Lambda_2=0.2", side = 1, line = -1.1, outer = TRUE)
```




```
par(mfrow=c(3,1),mar=c(5,3,2,3))
for(ii in 1:M){
  min1 <- min(yf_low1[ii,])
  max1 <- max(yf_high1[ii,])
  plot.ts(yf_median1[ii,], ylim=c(min1,max1), main=colnames(Y)[[ii]], ylab="", xlab="", xaxt="n",lwd=2)
  lines(yf_low1[ii,], lty=2,lwd=2)
  lines(yf_high1[ii,], lty=2,lwd=2)
  lines(yf_median1[ii,], lty=1,lwd=2)
  axis(1, at=seq(1,14), labels=xax1, las=2)
  abline(v=seq(1,14), col="lightgrey", lty=2)
}
mtext("Prior: Lambda_1=10, Lambda_2=20", side = 1, line = -1.1, outer = TRUE)
```



Finally, we compare the forecasts for the three variables for up to eight periods ahead for the two prior set-ups. Again, the predicted developments of the three variables are quite similar. Both inflation and unemployment are expected to decrease over time for both priors. The interest rate is expected to fall slightly with the smaller prior for the the λs, but stays roughly constant with the second specification. Finally, the confidence in the prediction is higher with the first set-up, compared to the second one.

Exercise 2

Read Kilian (2009), who discusses how to disentangle different oil shocks; focus on section II. Load the provided data by Kilian (2009), which contains a measure of change in oil production, a measure of real economic activity, and the real price of oil.

2.1

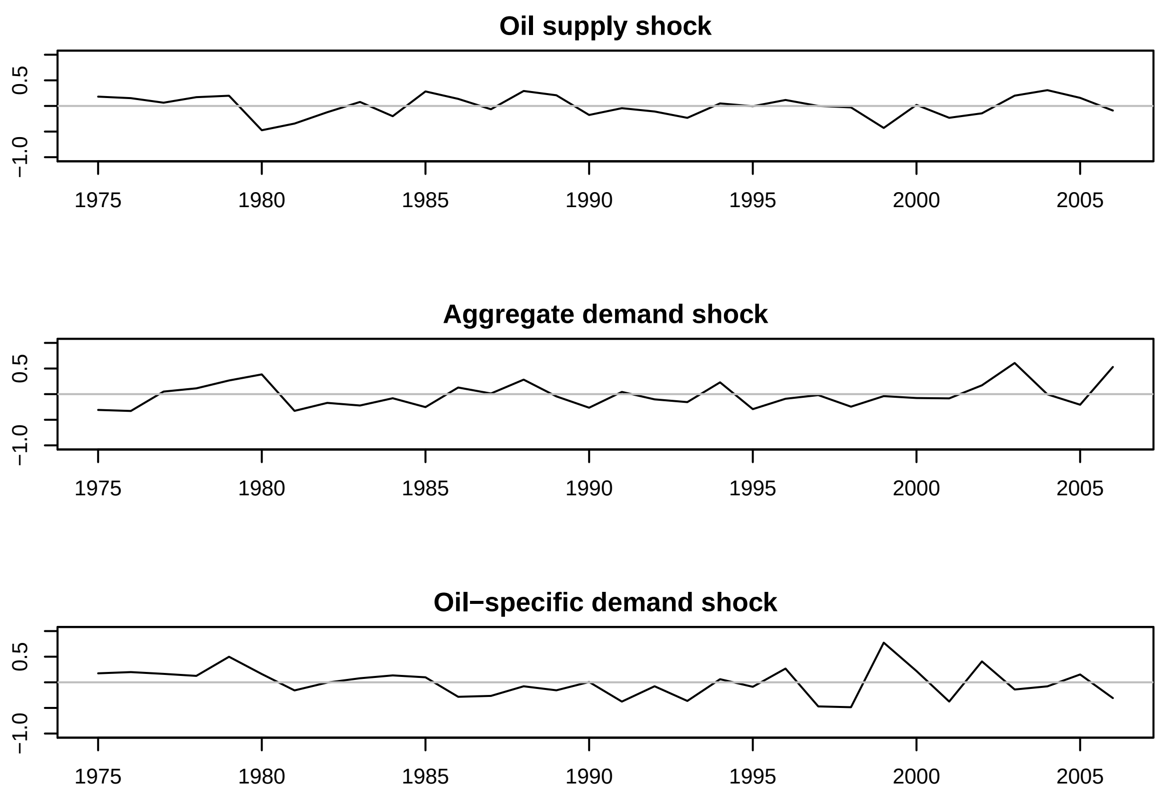
Q: Using the code for the Bayesian VAR, estimate the VAR described in section II.A and recover the structural form of it by recursive ordering of variables.

We estimated the VAR used by Kilian (2009) and recovered its structural form using recursive ordering of variables. To adjust for the aggressive shrinkage with the original Minnesota Prior, we increased both λ_1 and $lambda_2$ by factor 3 (since a quarter consists of three months), as higher $lambda$ values induce less shrinkage.

2.2

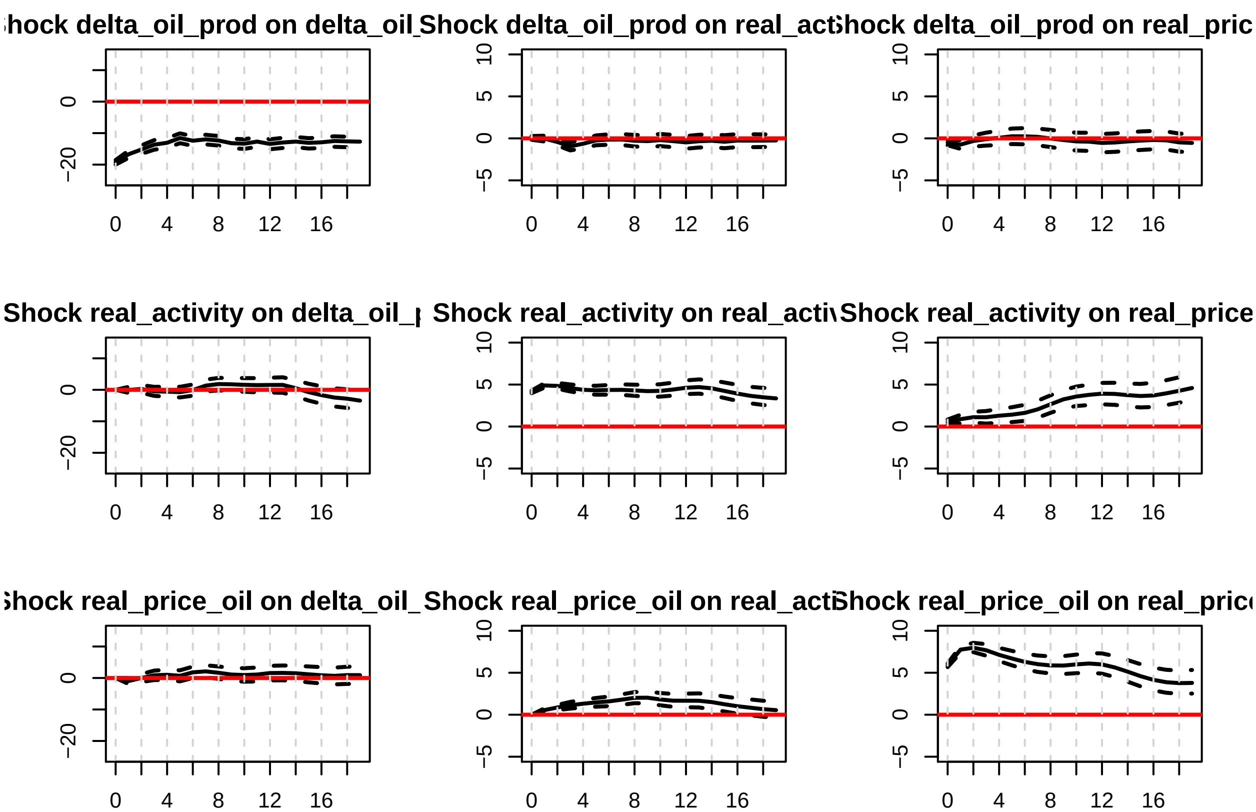
Q: Replicate figures 2 and 3 of Kilian, 2009.

```r
# Replicate Fig. 2
err <- t(apply(apply(E_store, c(2, 3), median), 1, function(x) solve(shock.chol) %*% x))
err <- ts(err, start = 1975, frequency = 12)
par(mfrow=c(3,1),mar=c(5,3,2,3))
plot(as.zoo(-aggregate.ts(err, FUN = mean))[, 1],
     ylim = c(-1, 1), xlab = "", ylab = "", main = "Oil supply shock")
abline(h=0, col="grey")
plot(as.zoo(aggregate.ts(err, FUN = mean))[, 2],
     ylim = c(-1, 1), xlab = "", ylab = "", main = "Aggregate demand shock")
abline(h=0, col="grey")
plot(as.zoo(aggregate.ts(err, FUN = mean))[, 3],
     ylim = c(-1, 1), xlab = "", ylab = "", main = "Oil-specific demand shock")
abline(h=0, col="grey")
```




```r
# Replicate Fig. 3
for (i in 1:3) {
  IRFchol_store[, 1, i, ] <- t(apply(IRFchol_store[, 1, i, ], 1, cumsum))
}

#Quantiles over the first dimension (number of saved draws)
IRFchol_low    <- apply(IRFchol_store, c(2,3,4), quantile, 0.16,na.rm=TRUE)
```

```
IRFchol_high   <- apply(IRFchol_store, c(2,3,4), quantile, 0.84,na.rm=TRUE)
IRFchol_median <- apply(IRFchol_store, c(2,3,4), median, na.rm=TRUE)

#Start plotting the IRFs w.r.t. different shocks
lims = list(c(-25, 15), c(-5, 10), c(-5, 10))
par(mfrow=c(3,3),mar=c(4,4,2,2))
for(jj in 1:M){
  for(ii in 1:M){
    ylim = lims[[ii]]
    plot.ts(IRFchol_median[ii,jj,], ylab="", main=paste0("Shock ",colnames(Y)[jj], " on ",colnames(Y)[i
    lines(IRFchol_low[ii,jj,], lty = 2, lwd=2)
    lines(IRFchol_high[ii,jj,], lty = 2, lwd=2)
    abline(h=0,col="red",lwd=2)
    abline(v=seq(1,nhor,by=2), col="lightgrey", lty=2)
    axis(1, at=seq(1,nhor,by=2), labels=seq(0,nhor-1,by=2))
  }
}
```



2.3

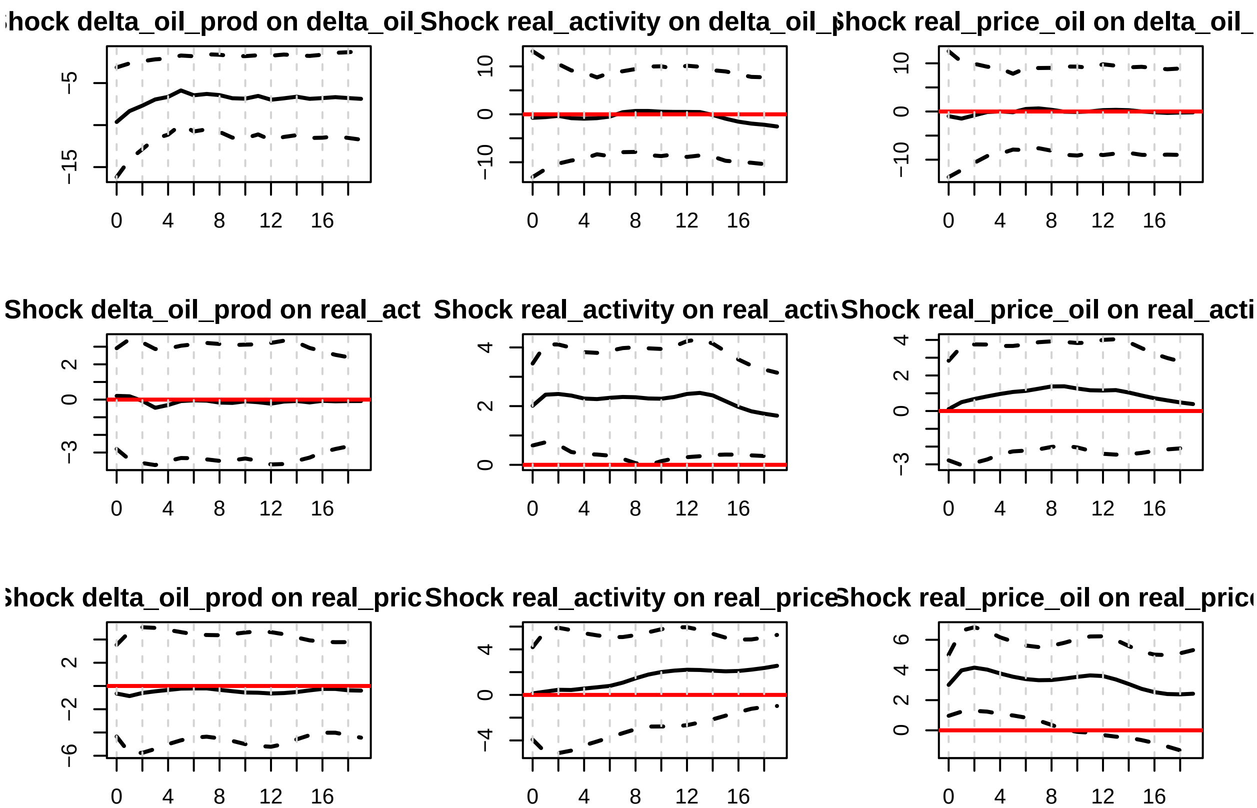
Q: Think of reasonable sign restrictions to identify the model at hand; discuss and implement them. Recreate figure 3 using these restrictions and discuss differences to the one identified by recursive ordering

From an econometric point, the sign restrictions need to be set such that the columns are not linearly dependent to allow identification. From an economic point, they further have to be set such that they make sense economically. After consulting the literature, we followed the specification by Kilian & Murphy (2012) and set the following sign restrictions:

$$\begin{pmatrix} \varepsilon^{OP} \\ \varepsilon^{RA} \\ \varepsilon^{P} \end{pmatrix} = \begin{bmatrix} - & + & + \\ - & + & - \\ + & + & + \end{bmatrix} \begin{pmatrix} e^{OS} \\ e^{AD} \\ e^{OD} \end{pmatrix}$$

which implies that an oil supply (OS) shock will by construction lower oil production (OP), but also lowers real activity (RA) and increases real price of oil (P). An unexpected shock to aggregate demand (AD) on the other hand leads to an increase in all three variables, while an oil-specific demand shock (OD) increases oil production and the real price of oil, but lowers real activity. Using the code for Bayesian VARs, we again estimate the model but now with the sign restrictons from above to identify the structural form. To induce less aggressive shrinkage, we again increase λ_1 and λ_2 like before.

```
# Replicate Fig. 3 with Sign Restrictions

for (i in 1:3) {
  IRFsign_store[, 1, i, ] <- t(apply(IRFsign_store[, 1, i, ], 1, cumsum))
}

IRFsign_low    <- apply(IRFsign_store, c(2,3,4), quantile, 0.16,na.rm=TRUE)
IRFsign_high   <- apply(IRFsign_store, c(2,3,4), quantile, 0.84,na.rm=TRUE)
IRFsign_median <- apply(IRFsign_store, c(2,3,4), median, na.rm=TRUE)

par(mfrow=c(3,3),mar=c(4,4,2,2))
for(ii in 1:M){
  for(jj in 1:M){
    min1 <- min(IRFsign_low[ii,jj,])
    max1 <- max(IRFsign_high[ii,jj,])
    plot.ts(IRFsign_median[ii,jj,], ylab="", xlab="", main=paste0("Shock ",colnames(Y)[jj], " on ",colna
    lines(IRFsign_low[ii,jj,], lty = 2, lwd=2)
    lines(IRFsign_high[ii,jj,], lty = 2, lwd=2)
    abline(h=0,col="red",lwd=2)
    abline(v=seq(1,nhor,by=2), col="lightgrey", lty=2)
    axis(1, at=seq(1,nhor,by=2), labels=seq(0,nhor-1,by=2))
  }
}
```



2.4

Q: Think of other variables that might be influenced by oil market shocks and collect data that fits the frequency and time period of the provided data. Transform the additional data appropriately and estimate the reduced form of a suitable VAR model. Identify the different shocks (using recursive ordering and/or sign restrictions), compute impulse responses and discuss your results.

We speculate that the unemployment rate in the US is affected by oil market shocks. To test if the time series for unemployment we use an ADF test.

```
adf.test(USDATA$UNRATE)
```

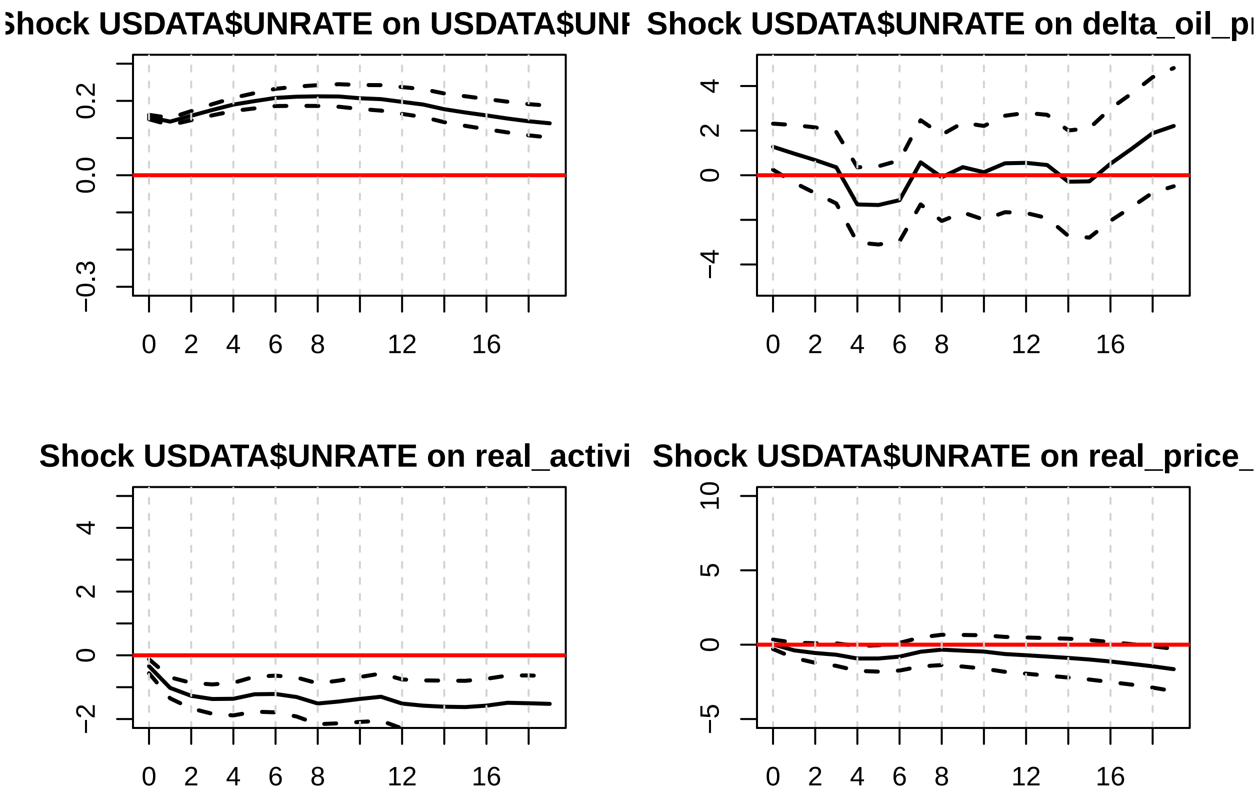
```
##
##  Augmented Dickey-Fuller Test
##
## data:  USDATA$UNRATE
## Dickey-Fuller = -3.9764, Lag order = 7, p-value = 0.01035
## alternative hypothesis: stationary
```

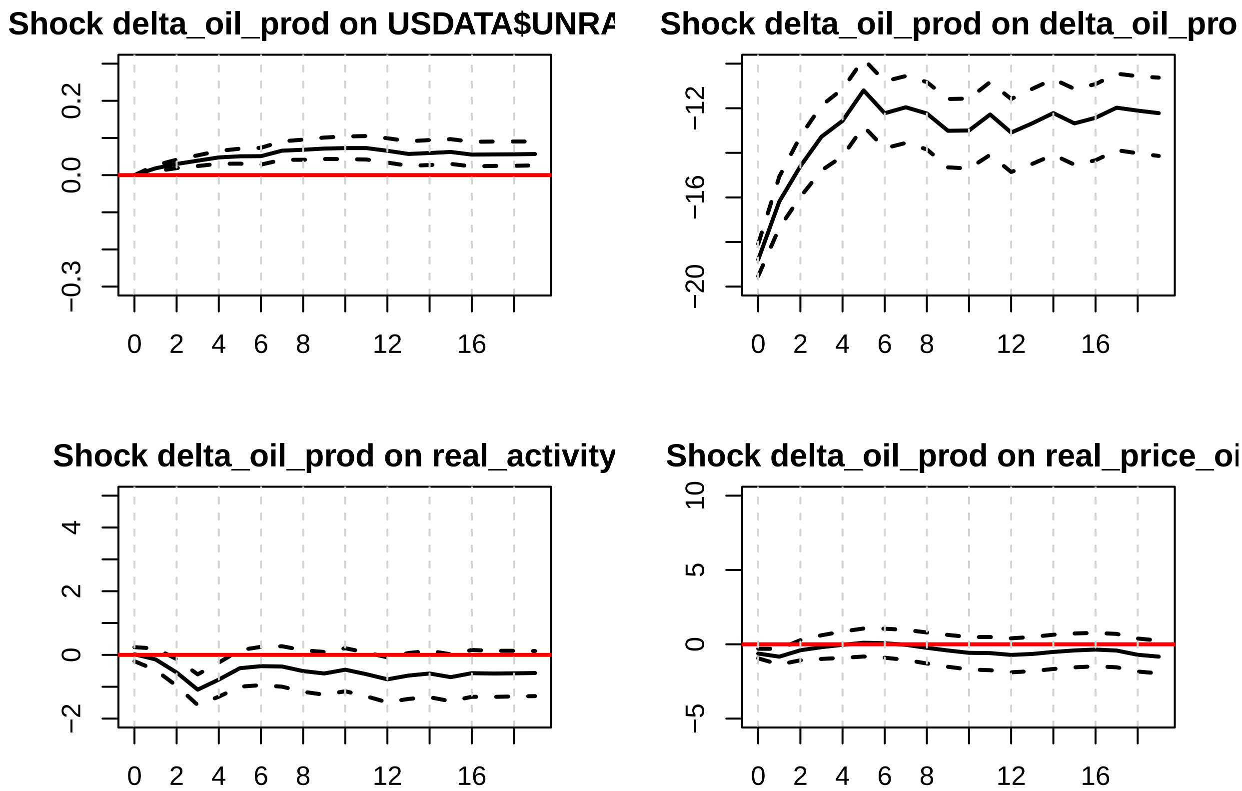
This test shows us that we can reject the null that the time series is non-stationary.

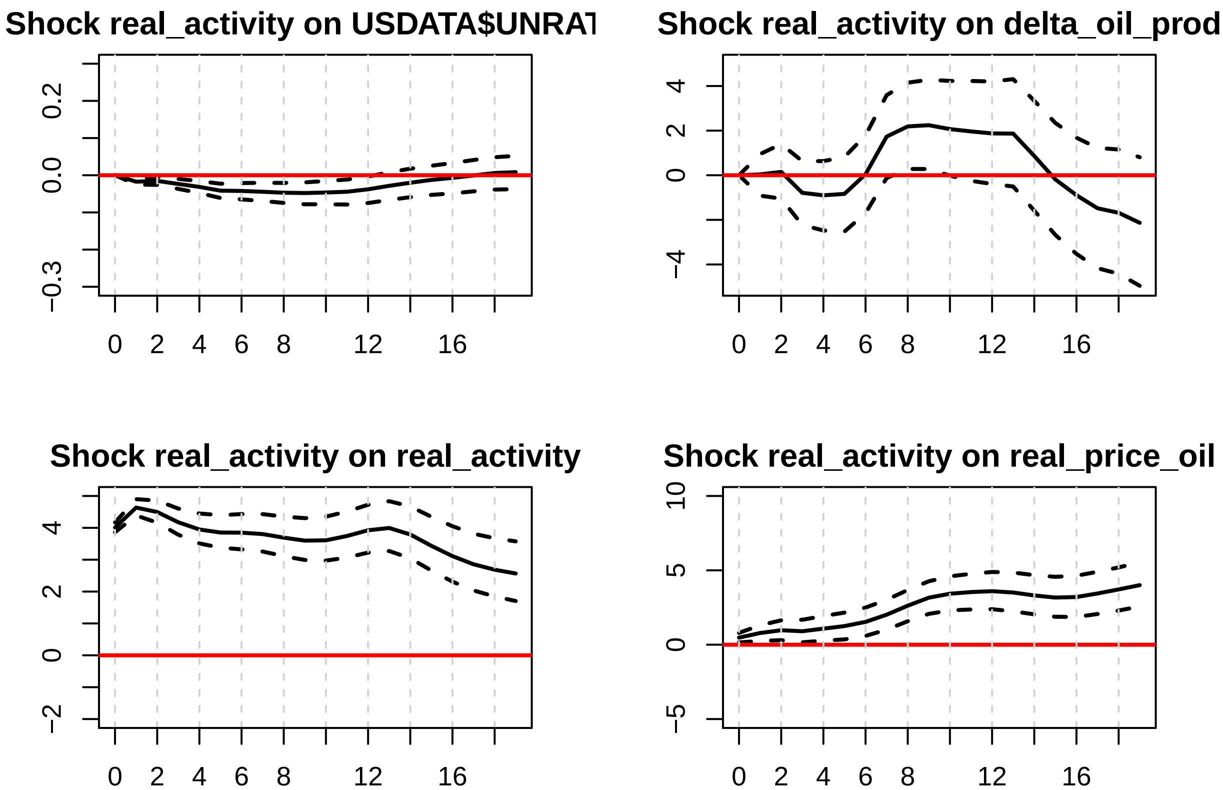
We then proceed by fitting a model similar to the one above, but with the unemployment time series added. For the identification, we use Cholesky decomposition and add unemployment as the last variable, meaning that unemployment is not affected by any other variables contemporaneously, but unemployment affects all other variables contemporaneously. This can be seen as justified economically, as unemployment is usually more sluggish than other macroeconomic variables.

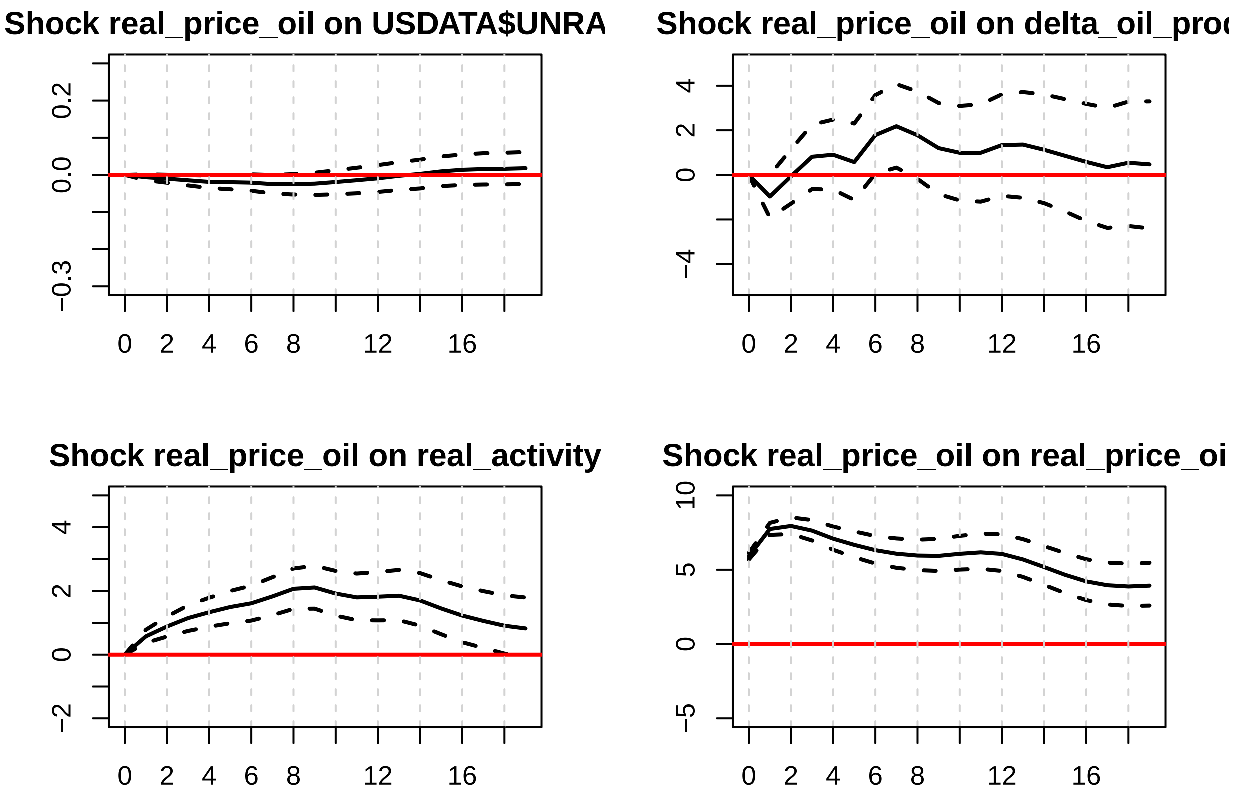
We then replicate figure 3 of the paper with the added variable:

```
for (i in 1:4) {
  IRFchol_store[, 2, i, ] <- t(apply(IRFchol_store[, 2, i, ], 1, cumsum))
}
#Quantiles over the first dimension (number of saved draws)
IRFchol_low <- apply(IRFchol_store, c(2, 3, 4), quantile, 0.16, na.rm = TRUE)
IRFchol_high <- apply(IRFchol_store, c(2, 3, 4), quantile, 0.84, na.rm = TRUE)
IRFchol_median <- apply(IRFchol_store, c(2, 3, 4), median, na.rm = TRUE)


#Start plotting the IRFs w.r.t. different shocks
lims = list(c(-0.3, 0.3),c(-5, 5), c(-2, 5), c(-5, 10))
par(mfrow = c(2, 2), mar = c(4, 4, 2, 2))
for (jj in 1:M) {
  for (ii in 1:M) {
    ylim = lims[[ii]]
    if(ii==2 & jj==2){
      ylim = c(-20,-10)
    }
    plot.ts(IRFchol_median[ii, jj,], ylab = "", main = paste0("Shock ", colnames(Y)[jj], " on ", colname
    lines(IRFchol_low[ii, jj,], lty = 2, lwd = 2)
    lines(IRFchol_high[ii, jj,], lty = 2, lwd = 2)
    abline(h = 0, col = "red", lwd = 2)
    abline(v = seq(1, nhor, by = 2), col = "lightgrey", lty = 2)
    axis(1, at = seq(1, nhor, by = 2), labels = seq(0, nhor - 1, by = 2))
  }
}
```


hock USDATA$UNRATE on USDATA$UN
Shock USDATA$UNRATE on delta_oil_p
Shock USDATA$UNRATE on real_activi
Shock USDATA$UNRATE on real_price_


Shock delta_oil_prod on USDATA$UNRA
−0.3
0.0
0.2
0 2 4 6 8 12 16
Shock delta_oil_prod on delta_oil_pro
−20
−16
−12
0 2 4 6 8 12 16
Shock delta_oil_prod on real_activity
−2
0
2
4
0 2 4 6 8 12 16
Shock delta_oil_prod on real_price_oi
−5
0
5
10
0 2 4 6 8 12 16


Shock real_activity on USDATA$UNRAT
0.2
0.0
−0.3
0 2 4 6 8 12 16
Shock real_activity on delta_oil_prod
4
2
0
−4
0 2 4 6 8 12 16
Shock real_activity on real_activity
4
2
0
−2
0 2 4 6 8 12 16
Shock real_activity on real_price_oil
10
5
0
−5
0 2 4 6 8 12 16



As we see from the IRFs, a negative oil production increases US unemployment, while a positive shock to real activity and decreases unemploymentThe effect of the real price of oil on unemployment is unclear, however.

A positive shock to US unemployment, meanwhile, has an unclear effect on oil production, and has a substantial negative effect on real activity, and a somewhat negative effect on the price of oil.