2021-02-19_SVAR_m1_EViews_formal

October 15, 2021

1 SVAR Model R Script

1.0.1 Summary

- Benchmark Model

1. 5 variables

   - overnight rate(%)
   - Sentiment
   - () (number of permits)
   - -()- (housing loan)
   - ()- (housing price index)

2. Lag = 7

$$\begin{bmatrix} \varepsilon_t^{R} \\ \varepsilon_t^{Sent} \\ \varepsilon_t^{LPermit} \\ \varepsilon_t^{dLloan} \\ \varepsilon_t^{dLhp} \end{bmatrix} = \begin{bmatrix} a_{11} & 0 & 0 & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 & 0 \\ a_{31} & a_{32} & a_{33} & 0 & 0 \\ a_{41} & a_{42} & a_{43} & a_{44} & 0 \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} \end{bmatrix} \begin{bmatrix} e_t^{mp} \\ e_t^{exp} \\ e_t^{hs} \\ e_t^{hd} \\ e_t^{sp} \end{bmatrix}$$

1.0.2 Set up environment

```
In [1]: getwd()
```

'/Users/Andy 1/google_drive/0_Preserved/Thesis/7_writing/model_playground/model/0_benchmark'

```
In [2]: #
        #Path = "/Users/Andy 1/Google  (r08323004@g.ntu.edu.tw)/0 Semesters/Thesis/6_VAR_model
        #setwd(Path)
        source("code/VAR_functions.R")           #  VARsource.R

        inv_tol = 1e-20 #(singular matrix)
```

```
 Attaching packages  tidyverse 1.3.1

 ggplot2 3.3.5      purrr   0.3.4
 tibble  3.1.4      dplyr   1.0.7
 tidyr   1.1.3      stringr 1.4.0
 readr   1.4.0      forcats 0.5.1

 Conflicts  tidyverse_conflicts()
 dplyr::arrange()   masks plyr::arrange()
 purrr::compact()   masks plyr::compact()
 dplyr::count()     masks plyr::count()
 dplyr::failwith()  masks plyr::failwith()
 dplyr::filter()    masks stats::filter()
 dplyr::id()        masks plyr::id()
 dplyr::lag()       masks stats::lag()
 dplyr::mutate()    masks plyr::mutate()
 dplyr::rename()    masks plyr::rename()
 dplyr::summarise() masks plyr::summarise()
 dplyr::summarize() masks plyr::summarize()

Loading required package: matrixcalc

Loading required package: lattice
```

```
In [3]: options(warn=-1)    # warning
        #options(warn=0)
        options(scipen=999) #
```

1.0.3

```
In [4]: ######  ######
        file = "data/df.csv"
        data = read.csv(file = file, header = TRUE)
        #data = na.omit(data)
        # 4-variable model
        By <- data %>% select(R, Sent, Permit_TW1, Loan3, hp_tw) %>% as.matrix
```

2 Raw Data

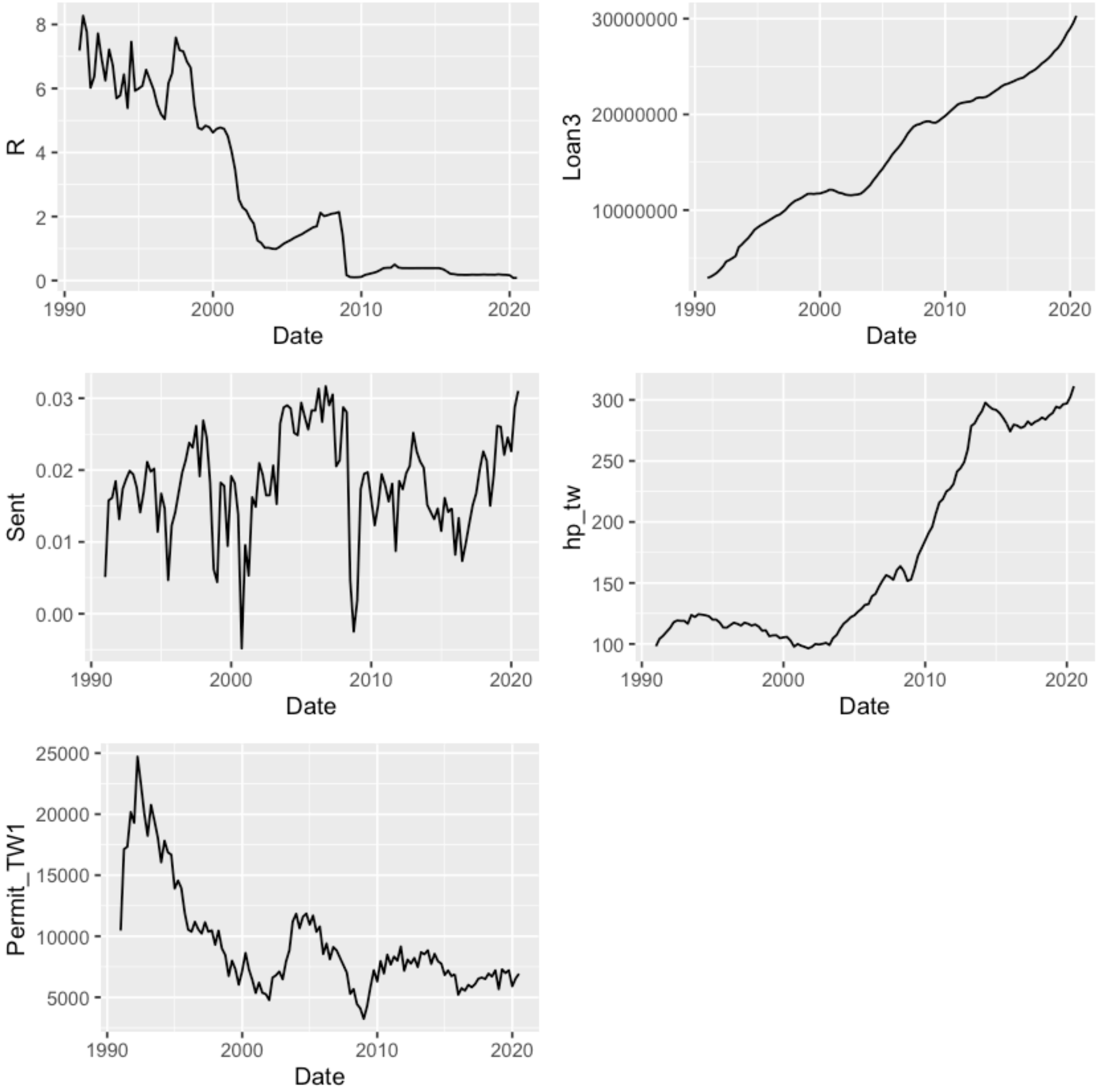
```
In [5]: dim(By)
```

1. 120 2. 5

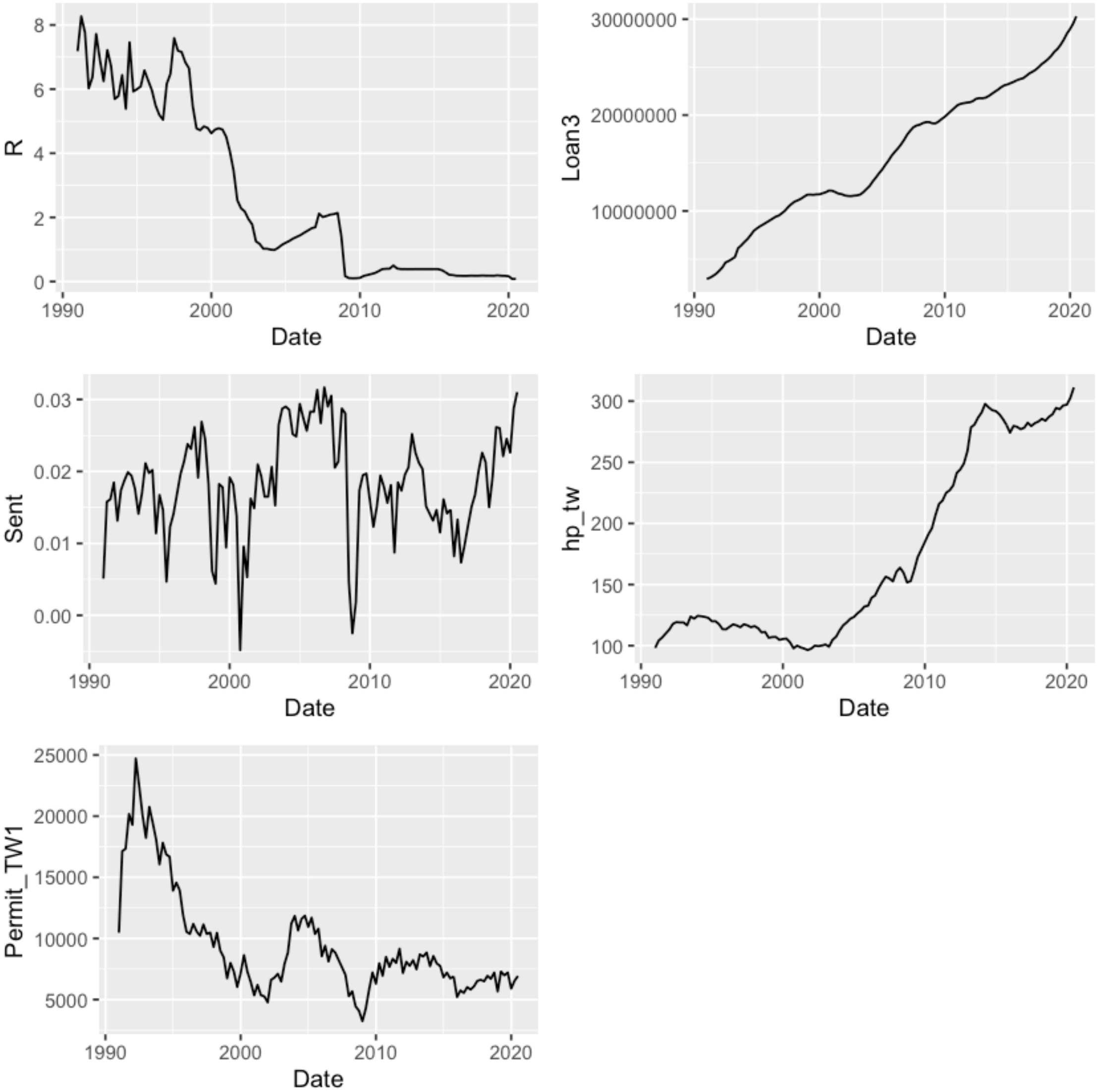
```
In [6]: data$Date <- as.Date(data$Date)
```

```
In [7]: raw_level_R <- ggplot(data, aes(x = Date, y = R))+
            geom_line()

        raw_level_Sent <- ggplot(data, aes(x = Date, y = Sent))+
            geom_line()

        raw_level_Permit_TW1 <- ggplot(data, aes(x = Date, y = Permit_TW1))+
            geom_line()

        raw_level_Loan3 <- ggplot(data, aes(x = Date, y = Loan3))+
            geom_line()

        raw_level_hp_tw <- ggplot(data, aes(x = Date, y = hp_tw))+
            geom_line()
```

```
In [8]: multiplot(raw_level_R,raw_level_Sent,
                  raw_level_Permit_TW1, raw_level_Loan3,
                  raw_level_hp_tw,
                  cols = 2)
```



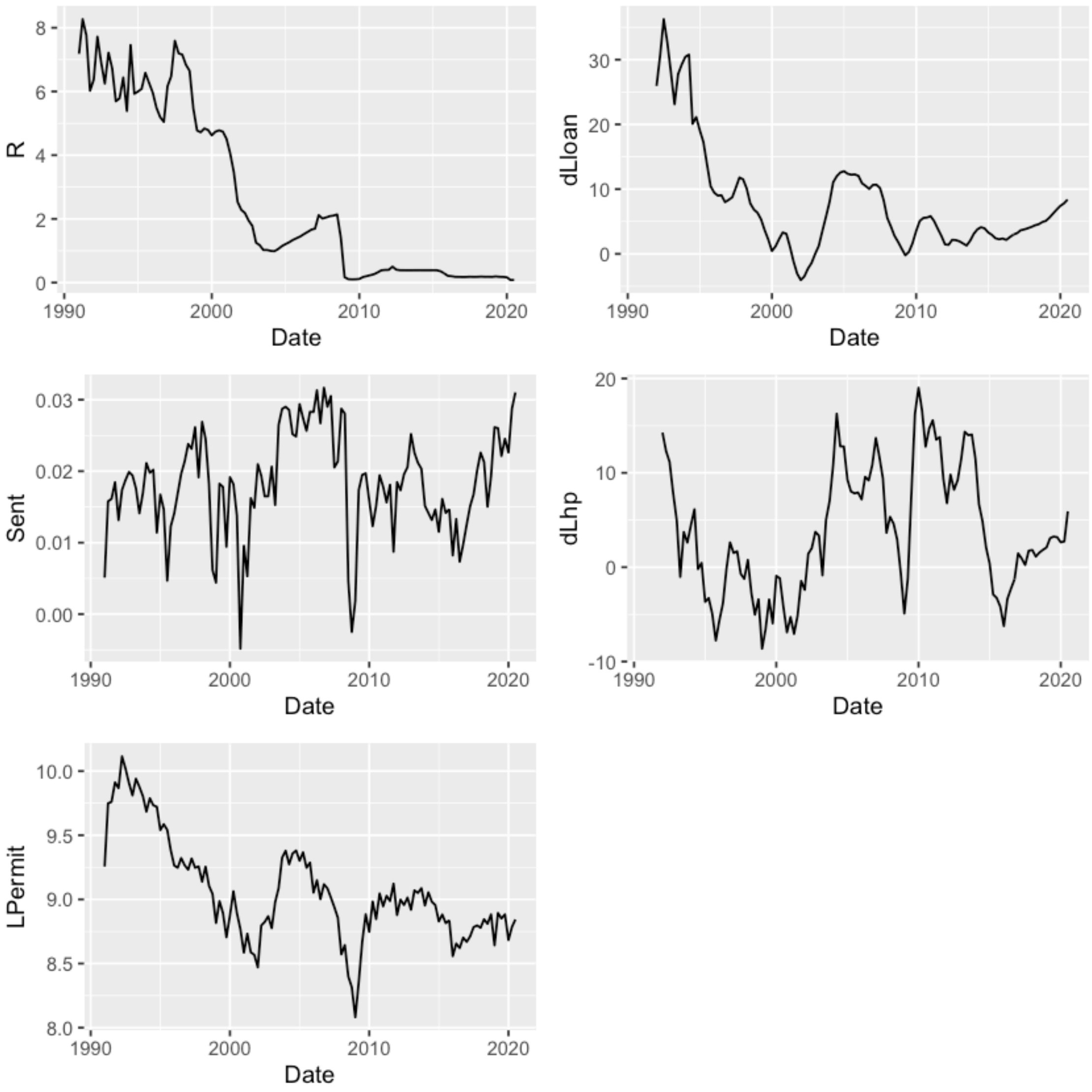
```
In [9]: ggsave(filename = "raw.png",
               plot = multiplot(raw_level_R,raw_level_Sent,
                                raw_level_Permit_TW1, raw_level_Loan3,
                                raw_level_hp_tw,
                                cols = 2),
               width = 30, height = 20, units = "cm",
               device = "png")
```



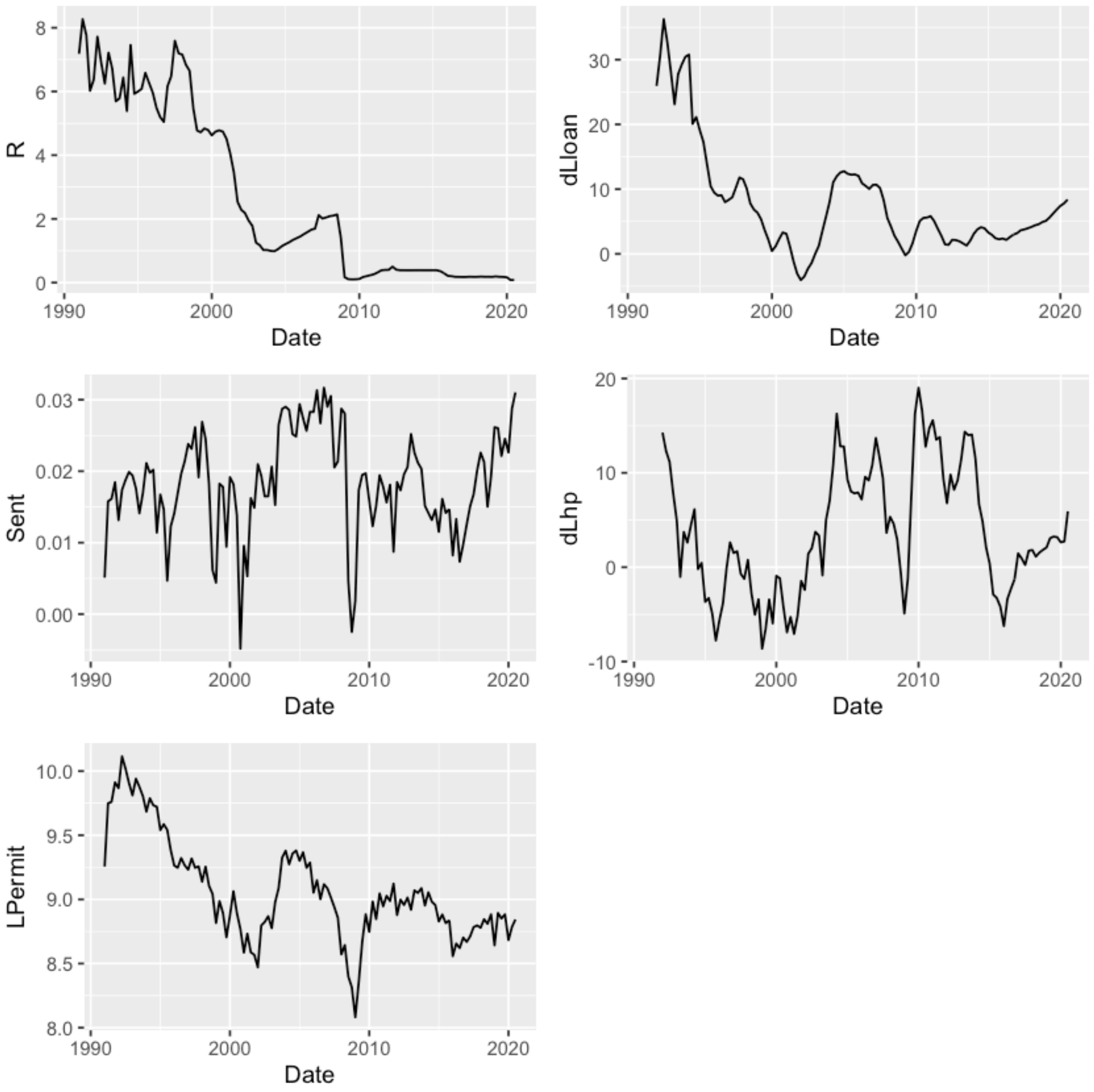
2.0.1 Data Manipulation

```
In [10]: By <- data %>% select(R, Sent, Permit_TW1, Loan3, hp_tw) %>%
             mutate(LPermit_TW1 = log(Permit_TW1),
                    dLloan = c(rep(NA, 4), 100*diff(log(Loan3), 4)),
                    dLhp = c(rep(NA,4), 100*diff(log(hp_tw), 4))) %>%
             select(R, Sent, LPermit = LPermit_TW1, dLloan, dLhp) %>%
             drop_na() %>%
             as.matrix
```

```
In [11]: data_new = data %>% select(Date, R, Sent, Permit_TW1, Loan3, hp_tw) %>%
             mutate(LPermit_TW1 = log(Permit_TW1),
                    dLloan = c(rep(NA, 4), 100*diff(log(Loan3), 4)),
                    dLhp = c(rep(NA,4), 100*diff(log(hp_tw), 4))) %>%
             select(Date, R, Sent, LPermit = LPermit_TW1, dLloan, dLhp)

         raw_level_R <- ggplot(data_new, aes(x = Date, y = R))+
             geom_line()

         raw_level_Sent <- ggplot(data_new, aes(x = Date, y = Sent))+
             geom_line()

         raw_level_LPermit <- ggplot(data_new, aes(x = Date, y = LPermit))+
             geom_line()

         raw_level_dLloan <- ggplot(data_new, aes(x = Date, y = dLloan))+
             geom_line()

         raw_level_dLhp <- ggplot(data_new, aes(x = Date, y = dLhp))+
             geom_line()

         multiplot(raw_level_R, raw_level_Sent,
                   raw_level_LPermit, raw_level_dLloan,
                   raw_level_dLhp,
                   cols = 2)
```



```
In [12]: ggsave(filename = "raw_new.png",
                plot = multiplot(raw_level_R, raw_level_Sent,
                                 raw_level_LPermit, raw_level_dLloan,
                                 raw_level_dLhp,
                                 cols = 2),
                width = 30, height = 20, units = "cm",
                device = "png")
```



2.0.2

```
In [13]: #-----  -----#
         VAR.P = 7                        #
         CONST = TRUE                     #
         Y     = VAR.Y(By, VAR.P)         #   Y
         X     = VAR.X(By, VAR.P)         #   X
```

```
In [14]: hrz=19 # the length of response
         shock_sign = -1 # control the positive/negative shock
```

2.1 Reduced Form VAR

```
In [15]: ######  ######
         (Coef.OLS     = VAR.OLS(Y, X, CONST)                     )
         (Sigma.OLS    = VAR.Sigma.OLS(Y, X, Coef.OLS, CONST)  )
         (Sigma.MLE    = VAR.Sigma.MLE(Y, X, Coef.OLS, CONST))
```

```
                                0.968389752    18.1265564    0.204402241   0.0582493370   -0.0170735991   0.19
                                -0.004998645   0.7242579     0.004089702   0.0008880673   -0.0003599237   -0.00
   A matrix: 5 Œ 36 of type dbl 0.027032539    6.5372498     0.561221943   0.0357627348   0.0025379838    -0.1
                                1.433599003    -25.6988821   2.174987030   1.3990142646   0.0693112238    -2.2
                                -0.074422212   103.0204145   2.137888156   0.5738638082   0.8515385513    -2.8
                               0.0731177057   0.00012011970   -0.00029355467   -0.1034498092   -0.081841335
                               0.0001201197   0.00002822604   0.00007950886    0.0008817479    0.007591137
   A matrix: 5 Œ 5 of type dbl -0.0002935547  0.00007950886   0.01024740890    0.0116281224    0.072531005
                               -0.1034498092  0.00088174792   0.01162812240    0.8628052551    0.575549186
                               -0.0818413346  0.00759113748   0.07253100481    0.5755491861    5.725868119
                               0.0487451371   0.00008007980   -0.00019570311   -0.0689665395   -0.054560890
                               0.0000800798   0.00001881736   0.00005300591    0.0005878319    0.005060758
   A matrix: 5 Œ 5 of type dbl -0.0001957031  0.00005300591   0.00683160594    0.0077520816    0.048354003
                               -0.0689665395  0.00058783194   0.00775208160    0.5752035034    0.383699457
                               -0.0545608898  0.00506075832   0.04835400321    0.3836994574    3.817245413
```

2.1.1 lag

```
In [16]: # AIC
         VAR.P = 7
```

3 SVAR

3.1 Identification Conditions

```
In [17]: ### 4-variable model

         Amat = diag(5)
         # Identification Conditions

         Amat[2,1]   = NA;
         Amat[3,1]   = NA; Amat[3,2]   = NA;
         Amat[4,1]   = NA; Amat[4,2]   = NA; Amat[4,3]   = NA;
         Amat[5,1]   = NA; Amat[5,2]   = NA; Amat[5,3]   = NA; Amat[5,4]   = NA;

         Bmat = diag(5)
         diag(Bmat) = NA

         Amat;Bmat
```

```
                              1    0    0    0    0
                              NA   1    0    0    0
A matrix: 5 Œ 5 of type dbl   NA   NA   1    0    0
                              NA   NA   NA   1    0
                              NA   NA   NA   NA   1
```

```
                              NA   0    0    0    0
                              0    NA   0    0    0
A matrix: 5 Œ 5 of type dbl   0    0    NA   0    0
                              0    0    0    NA   0
                              0    0    0    0    NA
```

3.2 Estimate $\hat{A}, \hat{B}$

- Recall the set up of the model:

$$D(L)Y_t = Be_t$$

we have:

$$Y_t = \Phi_1 Y_{t-1} + \cdots + \Phi_p Y_{t-p} + \varepsilon_t$$

1. impose zero-restrictions on A, B matrix
2. apply cholesky decomposition on $\Sigma_\varepsilon = CC'$, then utilize $C = (I - D_0)^{-1}B = A^{-1}B$

```
In [20]: C.Prime <- chol(Sigma.OLS)
```

```
In [21]: C <- t(C.Prime)
         C
```

```
                              0.270402858    0.000000000   0.00000000   0.0000000   0.000000
                              0.000444225    0.005294214   0.00000000   0.0000000   0.000000
A matrix: 5 Œ 5 of type dbl  -0.001085620    0.015109160   0.10008968   0.0000000   0.000000
                             -0.382576612    0.198650455   0.08203991   0.8186867   0.000000
                             -0.302664459    1.459251317   0.50109428   0.1572838   1.796946
```

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ a_{21} & 1 & 0 & 0 & 0 \\ a_{31} & a_{32} & 1 & 0 & 0 \\ a_{41} & a_{42} & a_{43} & 1 & 0 \\ a_{41} & a_{42} & a_{43} & a_{44} & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} \sigma_1 & 0 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 & 0 \\ 0 & 0 & \sigma_3 & 0 & 0 \\ 0 & 0 & 0 & \sigma_4 & 0 \\ 0 & 0 & 0 & 0 & \sigma_5 \end{bmatrix}$$

3.2.1 Note: Solving system of linear equations

Since B, C known, and $AC = B$. Thus $A = BC^{-1}$

```
In [23]: B0 <- diag(diag(C), ncol = 5, nrow = 5)
         B0
```

```
                             0.2704029   0.000000000   0.0000000   0.0000000   0.000000
                             0.0000000   0.005294214   0.0000000   0.0000000   0.000000
   A matrix: 5 Œ 5 of type dbl   0.0000000   0.000000000   0.1000897   0.0000000   0.000000
                             0.0000000   0.000000000   0.0000000   0.8186867   0.000000
                             0.0000000   0.000000000   0.0000000   0.0000000   1.796946
```

```
In [24]: A0 <- B0 %*% solve(C)
         A0
```

```
                             1.000000000    -0.000000000000005022074     0.000000    0.00000000000000000000
                             -0.001642826   0.999999999999999888978      0.000000    -0.0000000000000000000
   A matrix: 5 Œ 5 of type dbl   0.008703287    -2.853900520084375003194     1.000000    0.00000000000000000004
                             1.469348039    -35.182941941053485379598    -0.819664   1.00000000000000000000
                             1.246264156    -254.584179619649546566507   -4.848982   -0.1921172546035310456
```

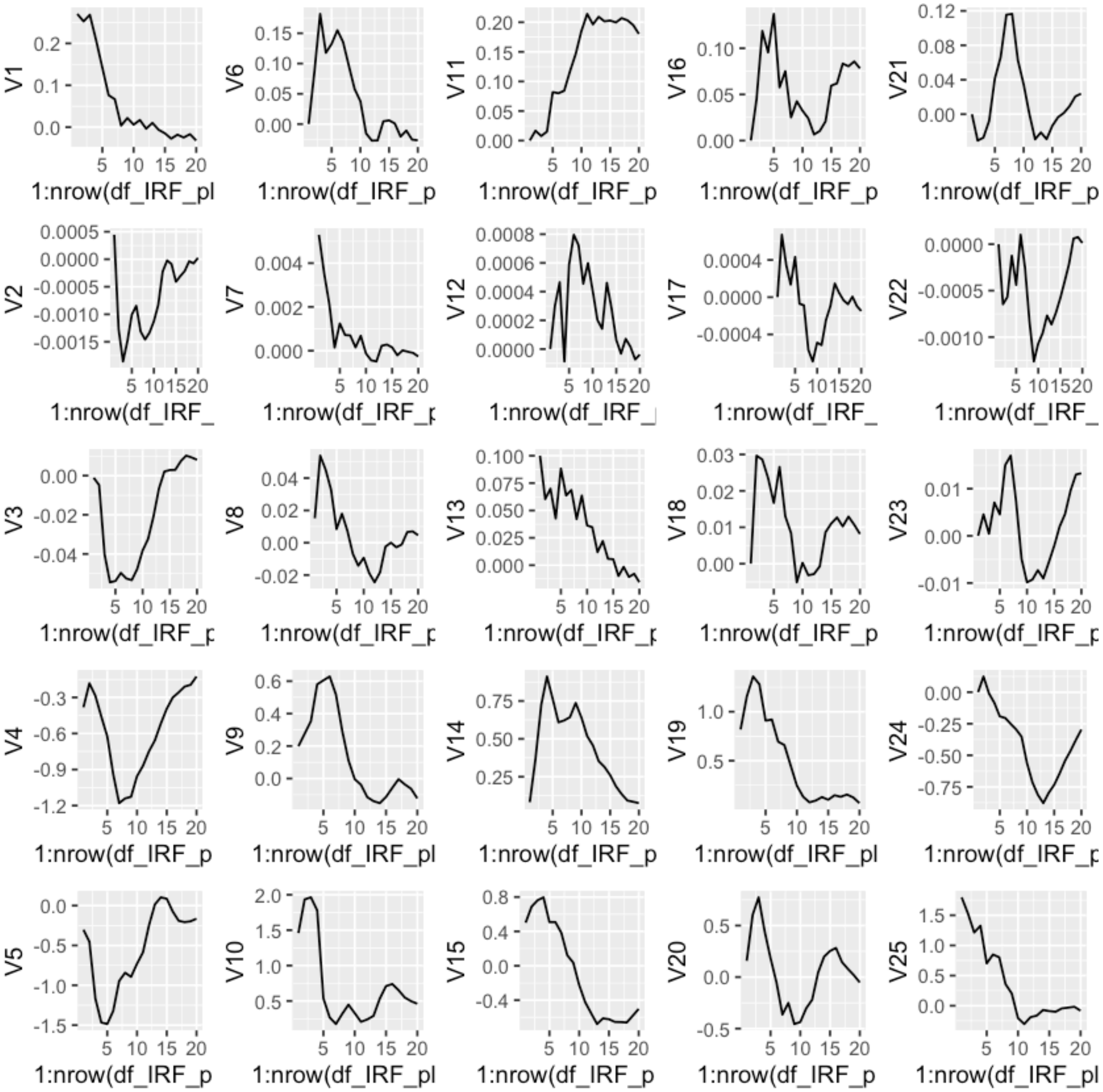
```
In [25]: SVAR_AB_est <- list("A0.svar" = A0, "B0.svar" = B0)
```

3.3 IRF (without Bootstrap C.I.)

```
In [26]: ### IRF
         SVAR_AB_IRF <- VAR.svarirf.AB(By, VAR.P, Amat, Bmat, h = hrz, CONST, SVAR_AB_est = SVA


         # 5*5time series
         df_IRF_plot <- matrix(NA, hrz+1, 25) #%>% as.tibble() ## hrz+1
         #dim(df_IRF_plot)
         h <- 0 # hIRF
         for(period in SVAR_AB_IRF){
           k <- 0 # k5*525coldfcolumns
           h <- h+1 # hIRF
           for(j in 1:5){
             for(i in 1:5){
               k <- k+1 # k5*525coldfcolumns
               df_IRF_plot[h,k] <- period[i,j]
             }
           }
         }
         df_IRF_plot <- df_IRF_plot %>% as_tibble()
```

```
In [27]: #output entire table
         IRF_TABLE <- df_IRF_plot[,c(5,10,15,20,25)] %>% select(mp=1,exp=2,hs=3,hd=4,hp=5)
         write.table(IRF_TABLE, file = "result/table/IRF_TABLE.csv", sep = ",", row.names = FA
In [28]: p1 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V1))
         p2 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V2))
         p3 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V3))
         p4 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V4))
         p5 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V5))
         p6 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V6))
         p7 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V7))
         p8 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V8))
         p9 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V9))
         p10 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V10))
         p11 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V11))
         p12 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V12))
         p13 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V13))
         p14 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V14))
         p15 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V15))
         p16 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V16))
         p17 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V17))
         p18 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V18))
         p19 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V19))
         p20 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V20))
         p21 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V21))
         p22 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V22))
         p23 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V23))
         p24 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V24))
         p25 <- ggplot(df_IRF_plot) + geom_line(aes(x = 1:nrow(df_IRF_plot), y = V25))
```

```
In [29]: multiplot(p1,p2,p3,p4,p5,
                   p6,p7,p8,p9,p10,
                   p11,p12,p13,p14,p15,
                   p16,p17,p18,p19,p20,
                   p21,p22,p23,p24,p25,
                   cols = 5)
```



3.4 IRF (Bootstrap C.I.)

Steps:

1. T
2. T5 periodTpseudo time series
3. pseudo time seriesSVARIRF

   a. $\hat{A}, \hat{B}$ matrixIRF

   b. residualpseudo time series $se(\hat{A}), se(\hat{B})$ bootstrap statistics

   c. 1000

4. IRF2.5% quantile97.5%quantileIRF95% C.I.

arrary

1. rowlag
2. columnshock1y1,..., shock5y5
3. pageBootstrap

4 Bootstrap C.I.

- http://www.eviews.com/help/helpintro.html#page/content%2Fmodels-Solving_the_Model.html%23ww100137

```
In [30]: #```R
         lower = 0.025                                        #  95% CI
         upper = 1-lower
         kk = ncol(By)
         ddY = VAR.ddY(By, VAR.P)
         ddX = VAR.ddX(By, VAR.P)

         # dim(ddY); dim(ddX)

         T   = nrow(ddY)
         T.total= nrow(By)
         Ik  = diag(rep(1, kk))
         # 16 coef if 4 variables; 55 coef if 5 variables
         Coef = t(VAR.EbyE(ddY, ddX, CONST)$ddA)              # Step 1
         # residuals
         U    = VAR.EbyE(ddY, ddX, CONST)$ddU
         BSigma.u = VAR.ddSigma.OLS(ddY, ddX, CONST)
         if(CONST == TRUE){
           const = Coef[, ncol(Coef)]
           Coef.noc= Coef[,-ncol(Coef)]                      #  const
         }else{
           const = matrix(0, kk, 1)
           Coef.noc = Coef
         }

         Theta.unit= VAR.Theta(Coef, h, BSigma.u, CONST)$unit #  Theta.unit
         Theta.std = VAR.Theta(Coef, h, BSigma.u, CONST)$std  #  Theta.std

         # dm.U <- U-mean(U)
         dm.U <- U

         N = 2000 #
         Theta.unit.sim = vector("list", N)
```

```
Theta.std.sim  = vector("list", N)

# check dimension
print("check dimensionality")
dim(ddX); dim(Coef.noc); dim(dm.U)


# NIRF
df_IRF.sim <- array(NA, c(hrz+1,kk^2,N)) #dimensions are: Time Period, Number of shoc
counter <- 1
while(TRUE){

  #cat("Now, there are ", counter-1, " sets of resamples.\n")
  Y.sim = matrix(0, nrow = T.total, ncol = kk)          # Y.sim = 0 #pseudo time seri
  Y.sim[c(1:VAR.P),] = By[c(1:VAR.P), ] #initial values

  boot.number = sample(c(1:T), replace = TRUE)      # Step 3
  U.sim = dm.U[boot.number,]

    # predicted values given the above initial values
    last.y= c(t(By[VAR.P:1,]))
    for(ii in 1:T){
         last.y = last.y[1:(kk*VAR.P)]
         Y.sim[ii+VAR.P, ] = Coef.noc %*% last.y + const + U.sim[ii,]      # Step 4
         last.y = c(Y.sim[ii+VAR.P,], last.y)
      }

#   Y.sim[-c(1:VAR.P),] <- matrix(const, nrow = T.total-VAR.P, ncol = kk, byrow = T) +


  #`Y.sim` is the pseudo time series
  # Step 5 SVAR

  ### SVAR.sim Start ###

    Y_pseudo     = VAR.Y(Y.sim, VAR.P)        #  Y
    X_pseudo     = VAR.X(Y.sim, VAR.P)        #  X
    Coef.OLS_pseudo    = VAR.OLS(Y_pseudo, X_pseudo, CONST)
    Sigma.OLS_pseudo   = VAR.Sigma.OLS(Y_pseudo, X_pseudo, Coef.OLS_pseudo, CONST)
    C.Prime_pseudo <- chol(Sigma.OLS_pseudo)
    C_pseudo <- t(C.Prime_pseudo)
    B0_pseudo <- diag(diag(C_pseudo), ncol = 5, nrow = 5)
    A0_pseudo <- B0_pseudo %*% solve(C_pseudo)
    SVAR_AB_est.sim <- list("A0.svar" = A0_pseudo, "B0.svar" = B0_pseudo)
    SVAR_AB_IRF.sim <- VAR.svarirf.AB(Y.sim, VAR.P, Amat, Bmat, h = hrz, CONST, SVAR_A

  # 5*5time series
  df_IRF_plot.sim <- matrix(NA, hrz+1, kk^2) #%>% as.tibble()
  # df_IRF.sim <- array(1:(120*25*N), c(120,25,N))
  # df_IRF.sim[2,1,1] # slicing

  h <- 0 # hIRF
  for(period in SVAR_AB_IRF.sim){
    k <- 0 # k5*525coldfcolumns
    h <- h+1 # hIRF
    for(j in 1:kk){
      for(i in 1:kk){
        k <- k+1 # k5*525coldfcolumns
        df_IRF_plot.sim[h,k] <- period[i,j]
      }
    }
  }
  # IRF append`df_IRF.sim`
  df_IRF.sim[,,counter] <- df_IRF_plot.sim
  ### SVAR.sim Ends ###
  if(counter>=N){
    break
  }
  counter <- counter+1
}
#```
```

```
[1] "check dimensionality"
```

1. 108 2. 35
1. 5 2. 35
1. 108 2. 5

```
In [31]: #```R
         # Save
         saveRDS(df_IRF.sim, file = "df_IRF.sim_0219_m1_hrz20.rds")
         #```
```

```
In [32]: df_IRF.sim <- read_rds("df_IRF.sim_0219_m1_hrz20.rds")
```

```
In [33]: #
         head(df_IRF.sim[,,1000])
         print(sum(is.na(df_IRF.sim)))
```

```
                              0.2141688   0.0005679190   0.004757537  -0.116877233  -0.18978718   0.00000
                              0.2472991   0.0002391613   0.009536948   0.065284665   0.03808164   0.05919
A matrix: 6 Œ 25 of type dbl  0.2812568  -0.0005868802  -0.005143707   0.140767813  -0.21868877   0.15185
                              0.2509795  -0.0006489705  -0.018689167  -0.008108773  -0.49738464   0.08132
                              0.1986898  -0.0006296184  -0.021024461  -0.204626514  -0.62553016   0.11072
                              0.1187769  -0.0009761512  -0.027221806  -0.469805133  -1.07081540   0.16541
```

```
[1] 0
```

4.0.1 IRF & Bootstrap C.I.

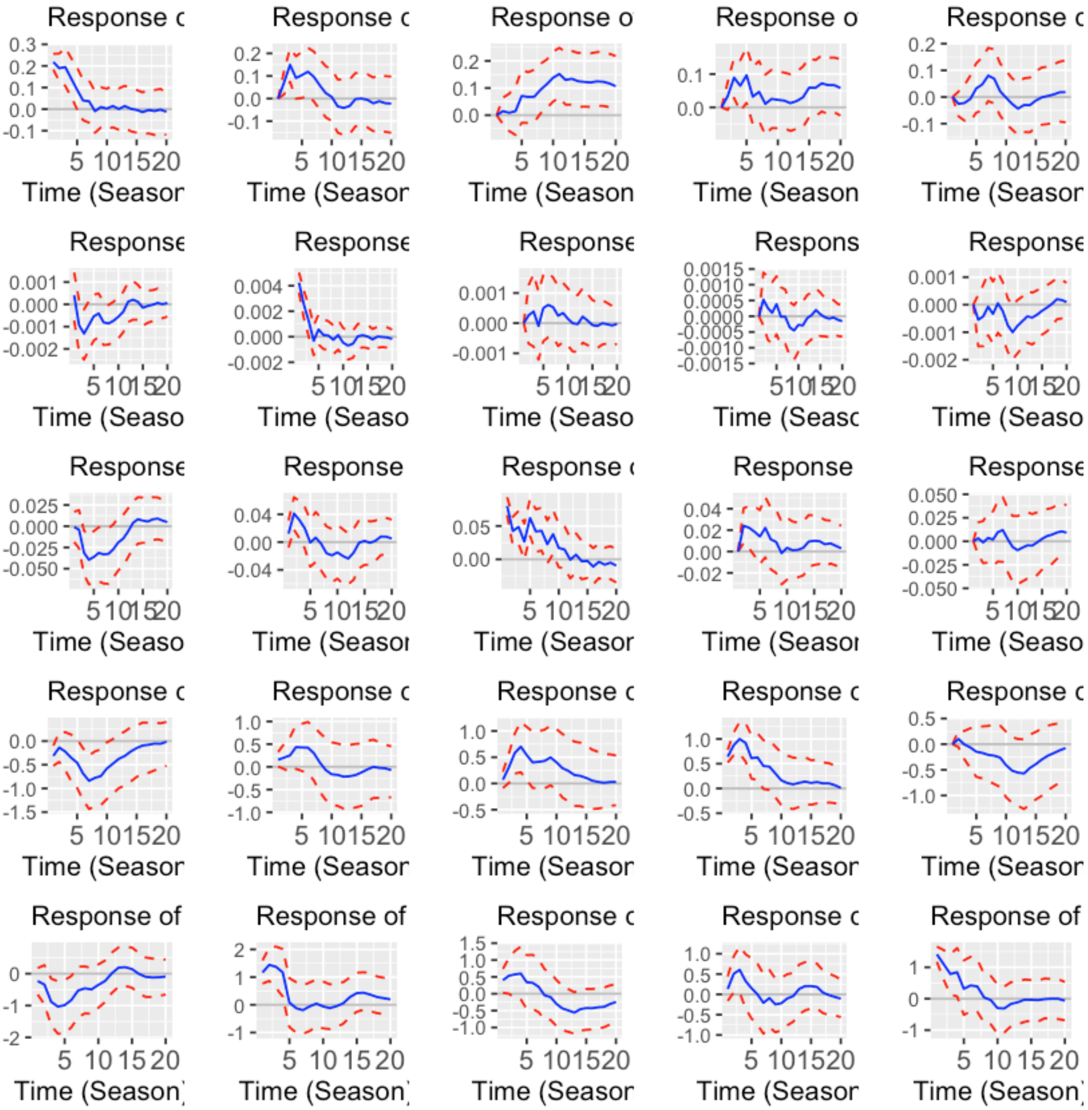
```
In [34]: df_IRF_plot.BS.L <- matrix(NA, nrow = hrz+1, ncol = 25)
         df_IRF_plot.BS.U <- matrix(NA, nrow = hrz+1, ncol = 25)
         df_IRF_plot.BS.Median <- matrix(NA, nrow = hrz+1, ncol = 25)
         df_IRF_plot.BS.Mean <- matrix(NA, nrow = hrz+1, ncol = 25)
         for(col in 1:25){
           for(row in 1:(hrz+1) ){
             df_IRF_plot.BS.L[row,col] <- quantile(df_IRF.sim[row,col,], probs = 0.025)
             df_IRF_plot.BS.U[row,col] <- quantile(df_IRF.sim[row,col,], probs = 0.975)
             df_IRF_plot.BS.Median[row,col] <- quantile(df_IRF.sim[row,col,], probs = 0.5)
             df_IRF_plot.BS.Mean[row,col] <- mean(df_IRF.sim[row,col,])
           }
         }

         df_IRF_plot.BS.L <- df_IRF_plot.BS.L %>% as_tibble()
         df_IRF_plot.BS.U <- df_IRF_plot.BS.U %>% as_tibble()
         df_IRF_plot.BS.Median <- df_IRF_plot.BS.Median %>% as_tibble()
         df_IRF_plot.BS.Mean <- df_IRF_plot.BS.Mean %>% as_tibble()
```

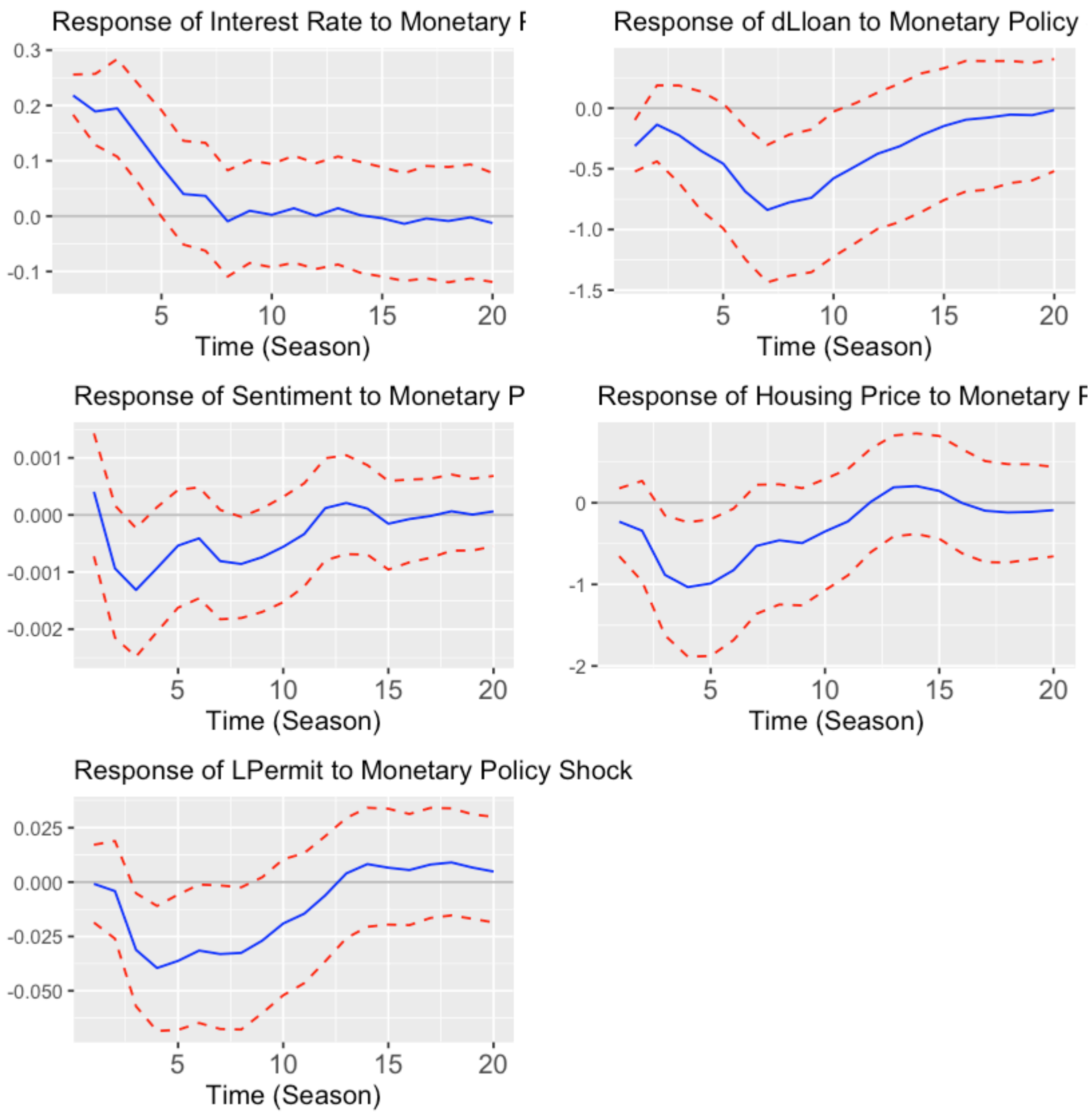
```
In [35]: ind <- 0
         for(i in 1:5){
           for(j in 1:5){
             ind <- ind+1
             nam <- paste("shock", j, "y", i, sep = '')
             assign(nam, bind_cols(df_IRF_plot.BS.L[ind], df_IRF_plot.BS.U[ind],
                                   df_IRF_plot.BS.Median[ind], df_IRF_plot.BS.Mean[ind],
                                   df_IRF_plot[ind]))
             #
             evalStr <- paste0("colnames(", nam, ") <- c('Lower', 'Upper', 'Median', 'Mean', '
             eval(parse(text=evalStr))
             #
             evalStr <- paste0("p", ind, " <- ", "ggplot(",nam,") +geom_hline(yintercept=0, co
             eval(parse(text=evalStr))
           }
         }
```

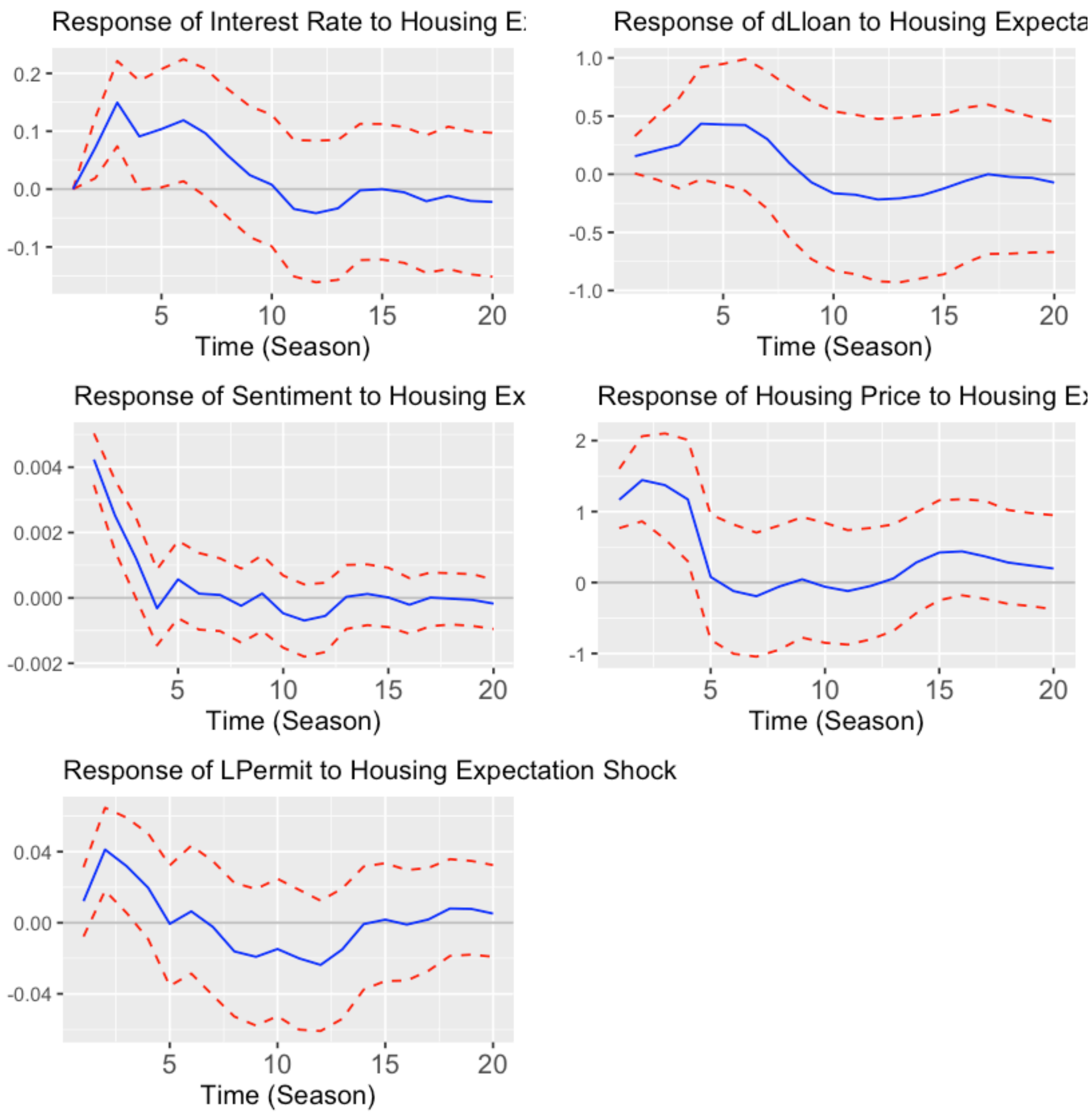
```
New names:
* V1 -> V1...1
* V1 -> V1...2
* V1 -> V1...3
* V1 -> V1...4
* V1 -> V1...5

New names:
* V2 -> V2...1
* V2 -> V2...2
* V2 -> V2...3
* V2 -> V2...4
* V2 -> V2...5

New names:
* V3 -> V3...1
* V3 -> V3...2
* V3 -> V3...3
* V3 -> V3...4
* V3 -> V3...5

New names:
* V4 -> V4...1
* V4 -> V4...2
* V4 -> V4...3
* V4 -> V4...4
* V4 -> V4...5

New names:
* V5 -> V5...1
* V5 -> V5...2
* V5 -> V5...3
* V5 -> V5...4
* V5 -> V5...5

New names:
* V6 -> V6...1
* V6 -> V6...2
* V6 -> V6...3
* V6 -> V6...4
* V6 -> V6...5

New names:
* V7 -> V7...1
* V7 -> V7...2
* V7 -> V7...3
* V7 -> V7...4
* V7 -> V7...5

New names:
* V8 -> V8...1
* V8 -> V8...2
* V8 -> V8...3
* V8 -> V8...4
* V8 -> V8...5

New names:
* V9 -> V9...1
* V9 -> V9...2
* V9 -> V9...3
* V9 -> V9...4
* V9 -> V9...5

New names:
* V10 -> V10...1
* V10 -> V10...2
* V10 -> V10...3
* V10 -> V10...4
* V10 -> V10...5

New names:
* V11 -> V11...1
* V11 -> V11...2
* V11 -> V11...3
* V11 -> V11...4
* V11 -> V11...5

New names:
* V12 -> V12...1
* V12 -> V12...2
* V12 -> V12...3
* V12 -> V12...4
* V12 -> V12...5

New names:
* V13 -> V13...1
* V13 -> V13...2
* V13 -> V13...3
* V13 -> V13...4
* V13 -> V13...5

New names:
* V14 -> V14...1
* V14 -> V14...2
* V14 -> V14...3
* V14 -> V14...4
* V14 -> V14...5

New names:
* V15 -> V15...1
* V15 -> V15...2
* V15 -> V15...3
* V15 -> V15...4
* V15 -> V15...5

New names:
* V16 -> V16...1
* V16 -> V16...2
* V16 -> V16...3
* V16 -> V16...4
* V16 -> V16...5

New names:
* V17 -> V17...1
* V17 -> V17...2
* V17 -> V17...3
* V17 -> V17...4
* V17 -> V17...5

New names:
* V18 -> V18...1
* V18 -> V18...2
* V18 -> V18...3
* V18 -> V18...4
* V18 -> V18...5

New names:
* V19 -> V19...1
* V19 -> V19...2
* V19 -> V19...3
* V19 -> V19...4
* V19 -> V19...5

New names:
* V20 -> V20...1
* V20 -> V20...2
* V20 -> V20...3
* V20 -> V20...4
* V20 -> V20...5

New names:
* V21 -> V21...1
* V21 -> V21...2
* V21 -> V21...3
* V21 -> V21...4
* V21 -> V21...5

New names:
* V22 -> V22...1
* V22 -> V22...2
* V22 -> V22...3
* V22 -> V22...4
* V22 -> V22...5

New names:
* V23 -> V23...1
* V23 -> V23...2
* V23 -> V23...3
* V23 -> V23...4
* V23 -> V23...5

New names:
* V24 -> V24...1
* V24 -> V24...2
* V24 -> V24...3
* V24 -> V24...4
* V24 -> V24...5

New names:
* V25 -> V25...1
* V25 -> V25...2
* V25 -> V25...3
* V25 -> V25...4
* V25 -> V25...5
```

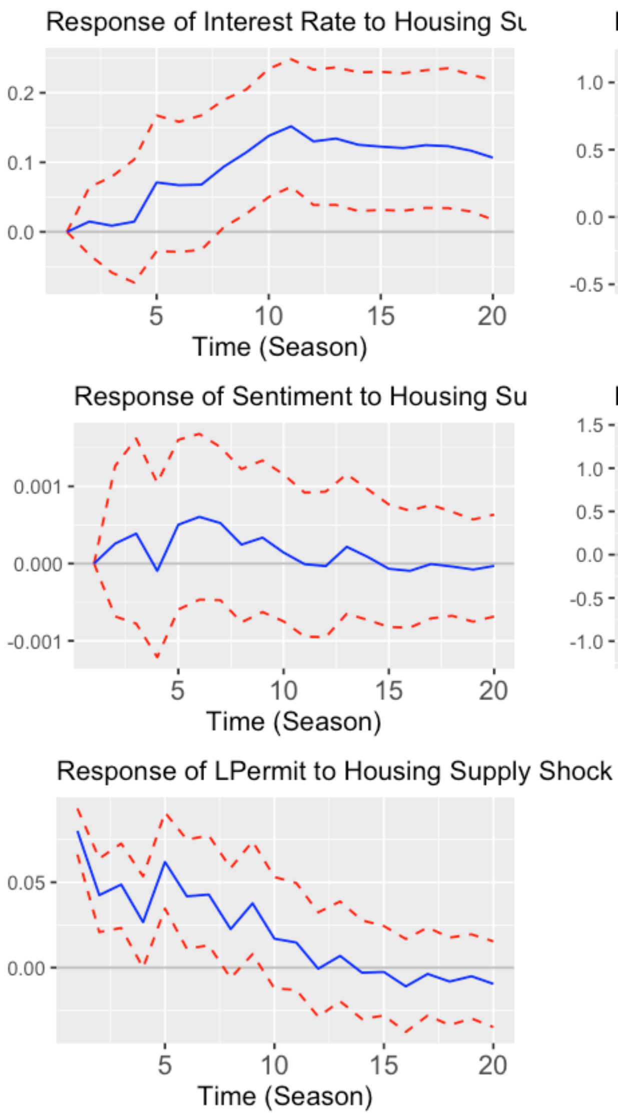
```
In [36]: Text_Size_Theme = theme(
           axis.title.x = element_text(size = 12),
           axis.text.x = element_text(size = 12),
           axis.title.y = element_text(size = 12),
           axis.title = element_text(size = 12),
           plot.title = element_text(size=12))

         ## shock1: mp
         p1 <- p1+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Interest Rate to Monetary Policy Shock')+Text_Size_Theme
         p2 <- p2+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Sentiment to Monetary Policy Shock')+Text_Size_Theme
         p3 <- p3+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of LPermit to Monetary Policy Shock')+Text_Size_Theme
         p4 <- p4+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of dLloan to Monetary Policy Shock')+Text_Size_Theme
         p5 <- p5+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Housing Price to Monetary Policy Shock')+Text_Size_Theme

         ## shock2: exp
         p6 <- p6+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Interest Rate to Housing Expectation Shock')+Text_Size_Th
         p7 <- p7+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Sentiment to Housing Expectation Shock')+Text_Size_Theme
         p8 <- p8+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of LPermit to Housing Expectation Shock')+Text_Size_Theme
         p9 <- p9+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of dLloan to Housing Expectation Shock')+Text_Size_Theme
         p10 <- p10+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Housing Price to Housing Expectation Shock')+Text_Size_Th

         ## shock3: supply
         p11 <- p11+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Interest Rate to Housing Supply Shock')+Text_Size_Theme
         p12 <- p12+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Sentiment to Housing Supply Shock')+Text_Size_Theme
         p13 <- p13+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of LPermit to Housing Supply Shock')+Text_Size_Theme
         p14 <- p14+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of dLloan to Housing Supply Shock')+Text_Size_Theme
         p15 <- p15+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Housing Price to Housing Supply Shock')+Text_Size_Theme




         ## shock4: demand
         p16 <- p16+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Interest Rate to Housing Demand Shock')+Text_Size_Theme
         p17 <- p17+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Sentiment to Housing Demand Shock')+Text_Size_Theme
         p18 <- p18+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of LPermit to Housing Demand Shock')+Text_Size_Theme
         p19 <- p19+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of dLloan to Housing Demand Shock')+Text_Size_Theme
         p20 <- p20+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Housing Price to Housing Demand Shock')+Text_Size_Theme



         ## shock5: sp
         p21 <- p21+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Interest Rate to Residual Shock')+Text_Size_Theme
         p22 <- p22+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Sentiment to Residual Shock')+Text_Size_Theme
         p23 <- p23+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of LPermit to Residual Shock')+Text_Size_Theme
         p24 <- p24+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of dLloan to Residual Shock')+Text_Size_Theme
         p25 <- p25+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Housing Price to Residual Shock')+Text_Size_Theme
```

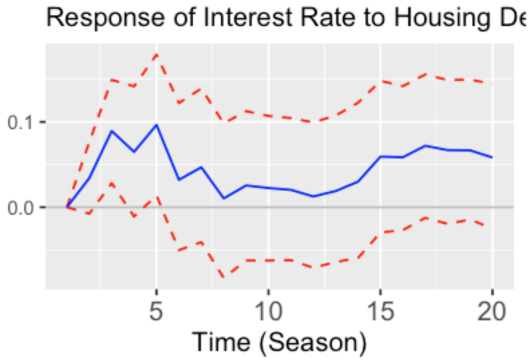
```
In [37]: multiplot(p1,p2,p3,p4,p5,
                   p6,p7,p8,p9,p10,
                   p11,p12,p13,p14,p15,
                   p16,p17,p18,p19,p20,
                   p21,p22,p23,p24,p25,
                   cols = 5)
         # For shock 1
         multiplot(p1,p2,p3,p4,p5,
                   cols = 2)
         # For shock 2
         multiplot(p6,p7,p8,p9,p10,
                   cols = 2)
         # For shock 3
         multiplot(p11,p12,p13,p14,p15,
                   cols = 2)
         # For shock 4
         multiplot(p16,p17,p18,p19,p20,
                   cols = 2)
         # For shock 5
         multiplot(p21,p22,p23,p24,p25,
                   cols = 2)
```

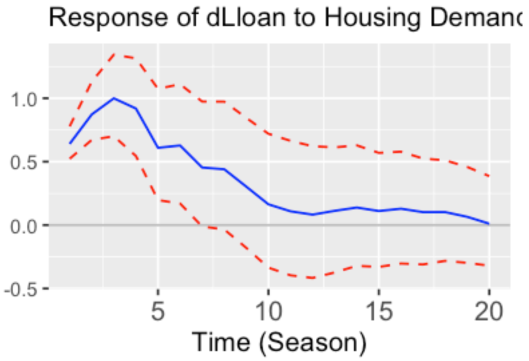

Response c
0.3
0.2
0.1
0.0
-0.1
5 101520
Time (Seasor
Response c
0.2
0.1
0.0
-0.1
5 101520
Time (Seasor
Response o
0.2
0.1
0.0
5 101520
Time (Season
Response o
0.1
0.0
5 101520
Time (Season
Response c
0.2
0.1
0.0
-0.1
5 101520
Time (Seasor
Response
0.001
0.000
-0.001
-0.002
5101520
Time (Seaso
Response
0.004
0.002
0.000
-0.002
5101520
Time (Seaso
Response
0.001
0.000
-0.001
5101520
Time (Seaso
Respons
0.0015
0.0010
0.0005
0.0000
-0.0005
-0.0010
-0.0015
5101520
Time (Seasc
Response
0.001
0.000
-0.001
-0.002
5101520
Time (Seaso
Response
0.025
0.000
-0.025
-0.050
5101520
Time (Seaso
Response
0.04
0.00
-0.04
5 101520
Time (Seasoı
Response (
0.05
0.00
5 101520
Time (Seasor
Response
0.04
0.02
0.00
-0.02
5 101520
Time (Seasoı
Response
0.050
0.025
0.000
-0.025
-0.050
5101520
Time (Seaso
Response c
0.0
-0.5
-1.0
-1.5
5 101520
Time (Seasor
Response c
1.0
0.5
0.0
-0.5
-1.0
5 101520
Time (Seasor
Response c
1.0
0.5
0.0
-0.5
5 101520
Time (Seasor
Response c
1.0
0.5
0.0
-0.5
5 101520
Time (Seasor
Response c
0.5
0.0
-0.5
-1.0
5 101520
Time (Seasor
Response of
0
-1
-2
5 10 15 20
Time (Season)
Response of
2
1
0
-1
5 10 15 20
Time (Season)
Response c
1.5
1.0
0.5
0.0
-0.5
-1.0
5 101520
Time (Seasor
Response c
1.0
0.5
0.0
-0.5
-1.0
5 101520
Time (Seasor
Response of
1
0
-1
5 10 15 20
Time (Season)

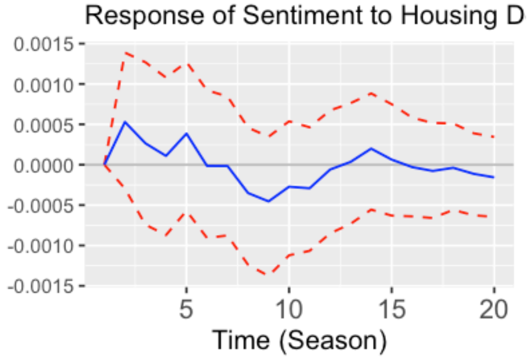

Response of Interest Rate to Monetary P
0.3
0.2
0.1
0.0
-0.1
5
10
15
20
Time (Season)
Response of dLloan to Monetary Policy
0.0
-0.5
-1.0
-1.5
5
10
15
20
Time (Season)
Response of Sentiment to Monetary P
0.001
0.000
-0.001
-0.002
5
10
15
20
Time (Season)
Response of Housing Price to Monetary P
0
-1
-2
5
10
15
20
Time (Season)
Response of LPermit to Monetary Policy Shock
0.025
0.000
-0.025
-0.050
5
10
15
20
Time (Season)

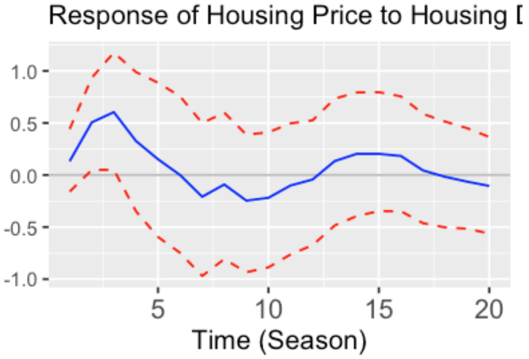

Response of Interest Rate to Housing E:
0.2
0.1
0.0
-0.1
5
10
15
20
Time (Season)
Response of dLloan to Housing Expecta
1.0
0.5
0.0
-0.5
-1.0
5
10
15
20
Time (Season)
Response of Sentiment to Housing Ex
0.004
0.002
0.000
-0.002
5
10
15
20
Time (Season)
Response of Housing Price to Housing E:
2
1
0
-1
5
10
15
20
Time (Season)
Response of LPermit to Housing Expectation Shock
0.04
0.00
-0.04
5
10
15
20
Time (Season)

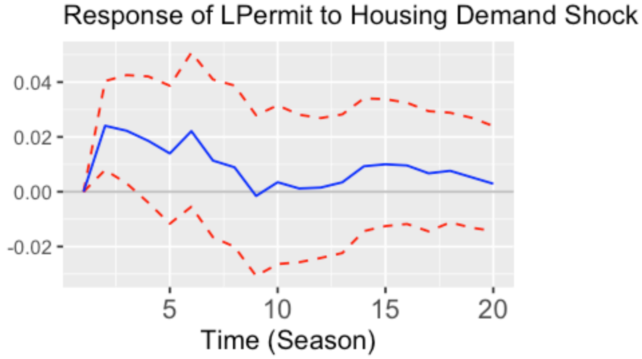

Response of Interest Rate to Housing Su
0.2
0.1
0.0
5
10
15
20
Time (Season)
Response of dLloan to Housing Supply :
1.0
0.5
0.0
-0.5
5
10
15
20
Time (Season)
Response of Sentiment to Housing Su
0.001
0.000
-0.001
5
10
15
20
Time (Season)
Response of Housing Price to Housing S
1.5
1.0
0.5
0.0
-0.5
-1.0
5
10
15
20
Time (Season)
Response of LPermit to Housing Supply Shock
0.05
0.00
5
10
15
20
Time (Season)

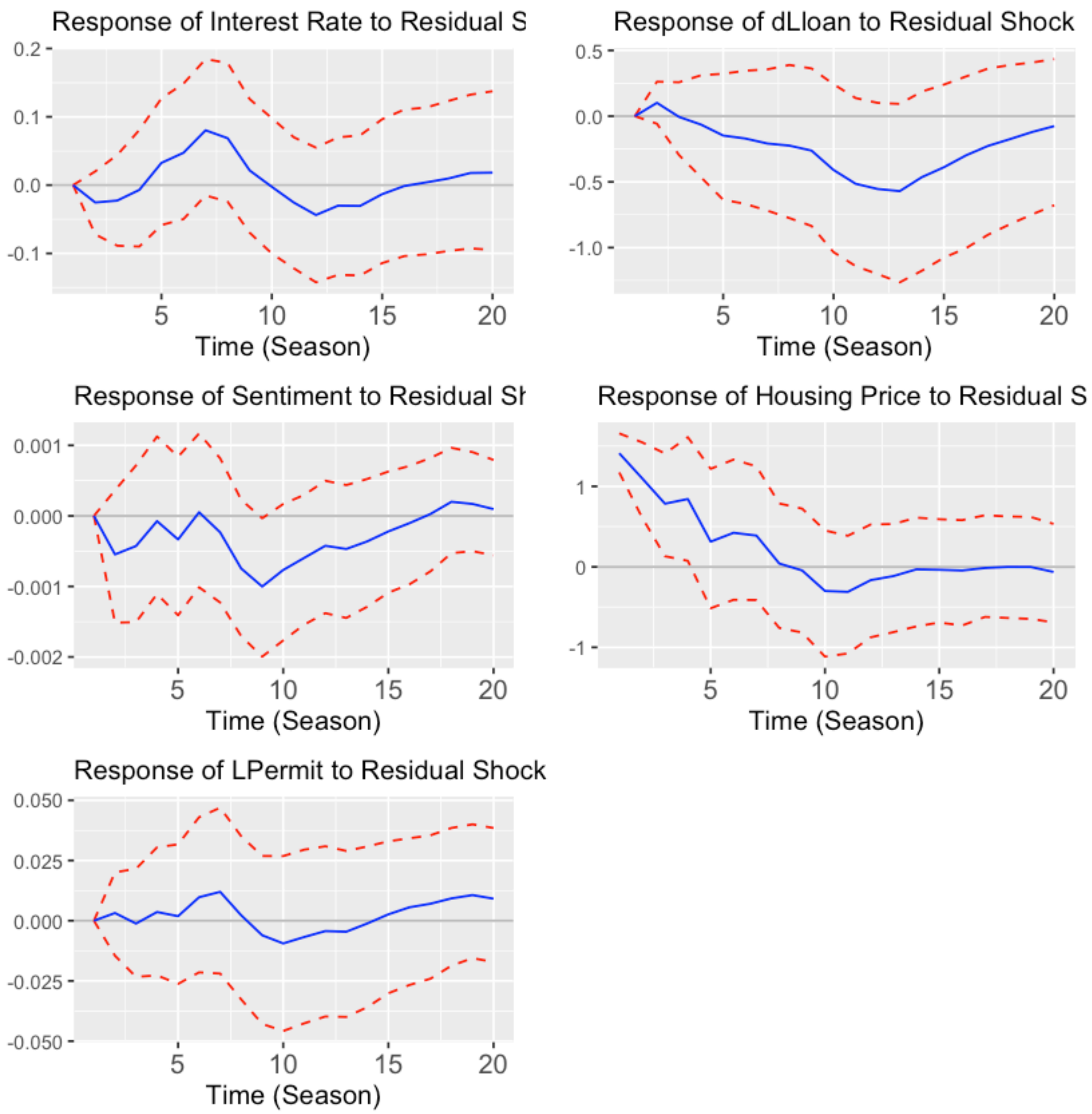

Response of Interest Rate to Housing De
0.1
0.0
5
10
15
20
Time (Season)


Response of dLloan to Housing Demand
1.0
0.5
0.0
-0.5
5
10
15
20
Time (Season)


Response of Sentiment to Housing D
0.0015
0.0010
0.0005
0.0000
-0.0005
-0.0010
-0.0015
5
10
15
20
Time (Season)

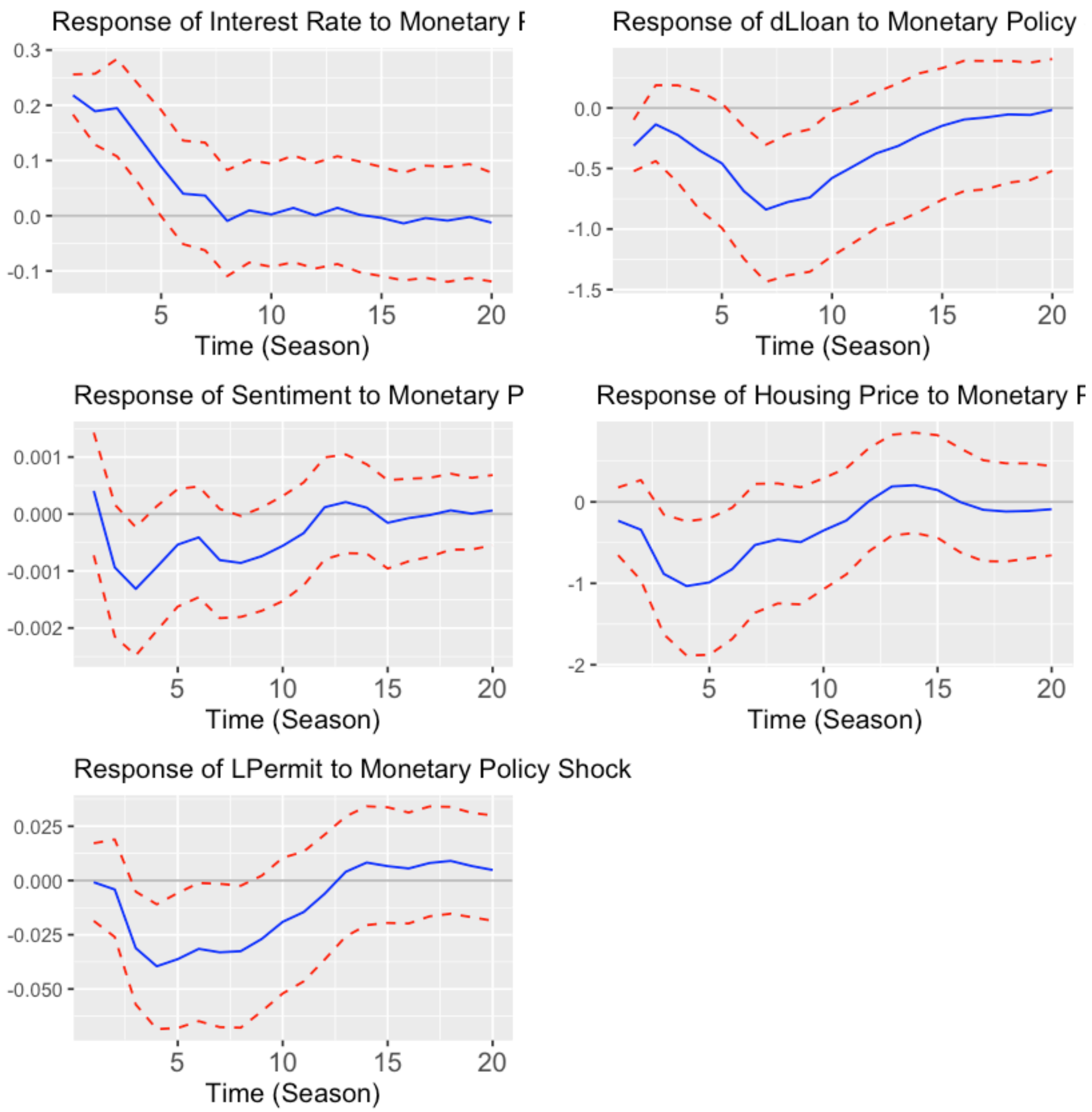

Response of Housing Price to Housing D
1.0
0.5
0.0
-0.5
-1.0
5
10
15
20
Time (Season)

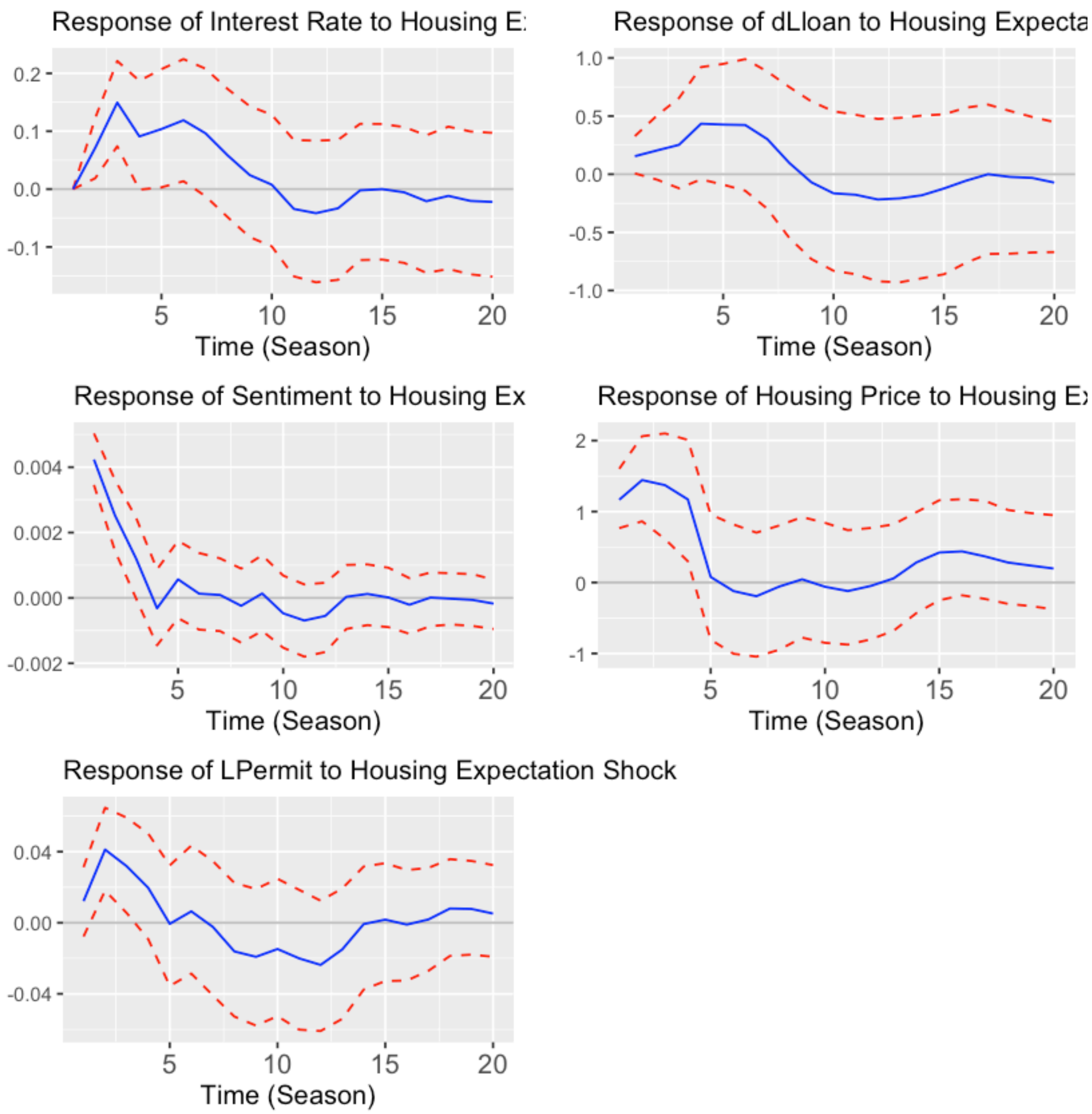

Response of LPermit to Housing Demand Shock
0.04
0.02
0.00
-0.02
5
10
15
20
Time (Season)



```
In [38]: # save plot
         # shock 1
         ggsave(filename = "result/figure/0219_m1_IRF_shock1.png",
                plot = multiplot(p1,p2,p3,p4,p5, cols = 2),
                width = 30, height = 20, units = "cm",
                device = "png")

         # shock 2
         ggsave(filename = "result/figure/0219_m1_IRF_shock2.png",
                plot = multiplot(p6,p7,p8,p9,p10, cols = 2),
                width = 30, height = 20, units = "cm",
                device = "png")
```

```
# shock 3
ggsave(filename = "result/figure/0219_m1_IRF_shock3.png",
       plot = multiplot(p11,p12,p13,p14,p15, cols = 2),
       width = 30, height = 20, units = "cm",
       device = "png")

# shock 4
ggsave(filename = "result/figure/0219_m1_IRF_shock4.png",
       plot = multiplot(p16,p17,p18,p19,p20, cols = 2),
       width = 30, height = 20, units = "cm",
       device = "png")

# shock 5
ggsave(filename = "result/figure/0219_m1_IRF_shock5.png",
       plot = multiplot(p21,p22,p23,p24,p25, cols = 2),
       width = 30, height = 20, units = "cm",
       device = "png")
```


Response of Interest Rate to Monetary F
0.3
0.2
0.1
0.0
-0.1
5
10
15
20
Time (Season)
Response of dLloan to Monetary Policy
0.0
-0.5
-1.0
-1.5
5
10
15
20
Time (Season)
Response of Sentiment to Monetary P
0.001
0.000
-0.001
-0.002
5
10
15
20
Time (Season)
Response of Housing Price to Monetary F
0
-1
-2
5
10
15
20
Time (Season)
Response of LPermit to Monetary Policy Shock
0.025
0.000
-0.025
-0.050
5
10
15
20
Time (Season)

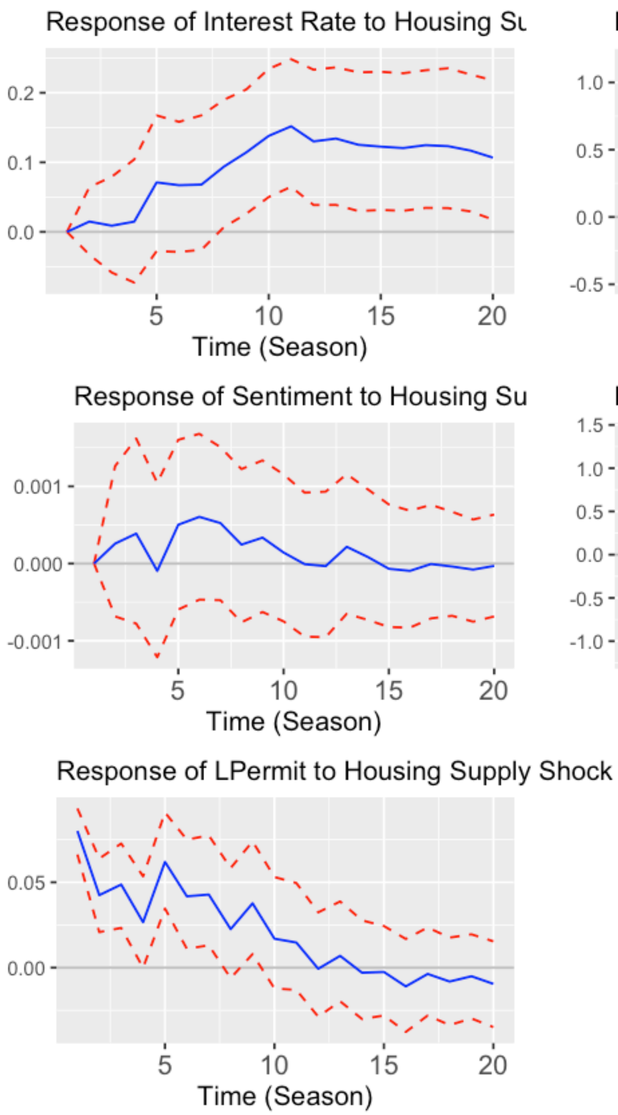

Response of Interest Rate to Housing E:

Response of dLloan to Housing Expecta

Response of Sentiment to Housing Ex

Response of Housing Price to Housing E:

Response of LPermit to Housing Expectation Shock



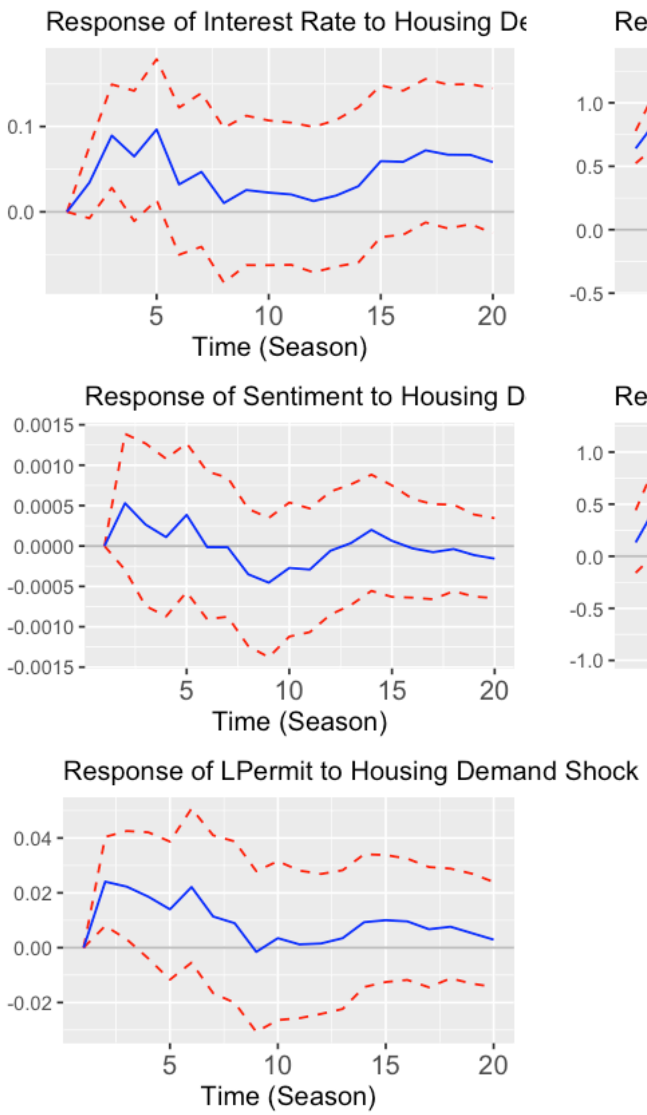
Response of Interest Rate to Housing Su

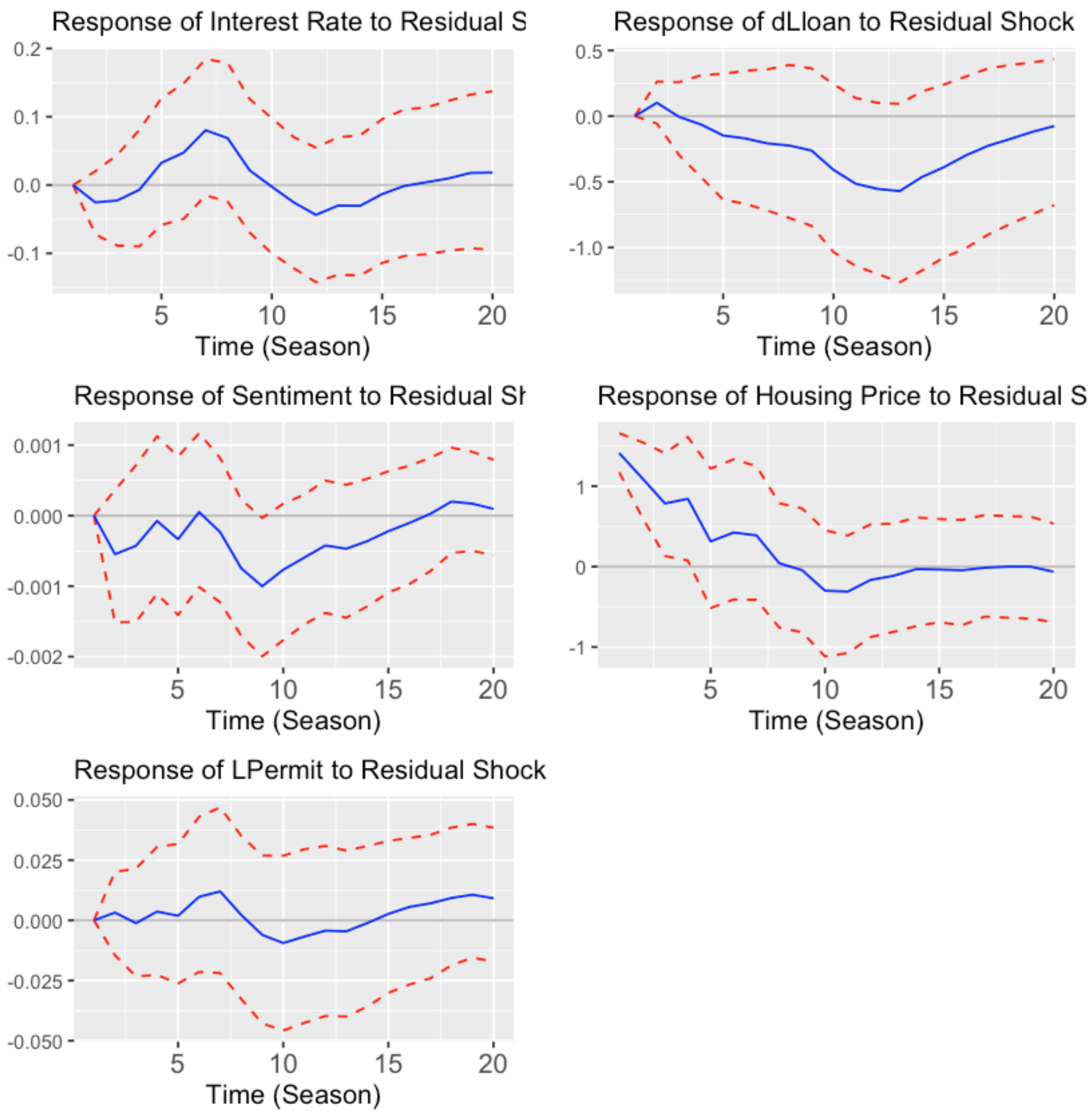
Response of dLloan to Housing Supply

Response of Sentiment to Housing Su

Response of Housing Price to Housing

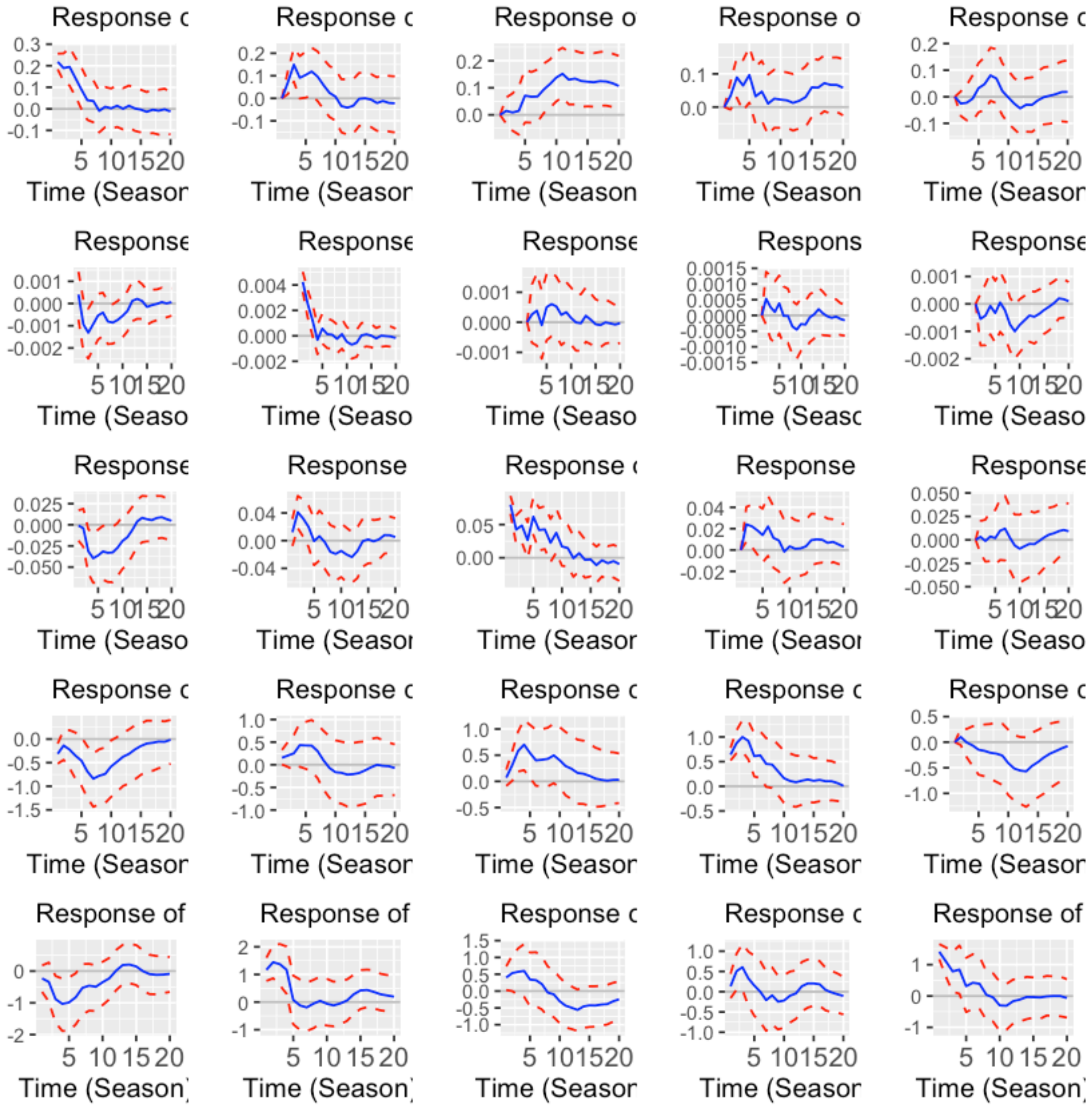
Response of LPermit to Housing Supply Shock


Response of Interest Rate to Housing De
0.1
0.0
5
10
15
20
Time (Season)
Response of dLloan to Housing Demanc
1.0
0.5
0.0
-0.5
5
10
15
20
Time (Season)
Response of Sentiment to Housing D
0.0015
0.0010
0.0005
0.0000
-0.0005
-0.0010
-0.0015
5
10
15
20
Time (Season)
Response of Housing Price to Housing I
1.0
0.5
0.0
-0.5
-1.0
5
10
15
20
Time (Season)
Response of LPermit to Housing Demand Shock
0.04
0.02
0.00
-0.02
5
10
15
20
Time (Season)



4.0.2 Saving all IRF

```
In [39]: # For all IRF
         ggsave(filename = "result/figure/imp.png",
                plot = multiplot(p1,p2,p3,p4,p5,
                                 p6,p7,p8,p9,p10,
                                 p11,p12,p13,p14,p15,
                                 p16,p17,p18,p19,p20,
                                 p21,p22,p23,p24,p25,
                                 cols = 5),
                width = 15*2, height = 10*2, units = "cm",
                device = "png")
```



4.0.3 shock sign

```
In [40]: print(shock_sign)
```

```
[1] -1
```

```
In [41]: df_IRF_plot.BS.L_negative <- (df_IRF_plot.BS.L*shock_sign) %>% as_tibble()
         df_IRF_plot.BS.U_negative <- (df_IRF_plot.BS.U*shock_sign) %>% as_tibble()
         df_IRF_plot.BS.Median_negative <- (df_IRF_plot.BS.Median*shock_sign) %>% as_tibble()
         df_IRF_plot.BS.Mean_negative <- (df_IRF_plot.BS.Mean*shock_sign) %>% as_tibble()
         df_IRF_plot_negative <- (df_IRF_plot*shock_sign) %>% as_tibble()
```

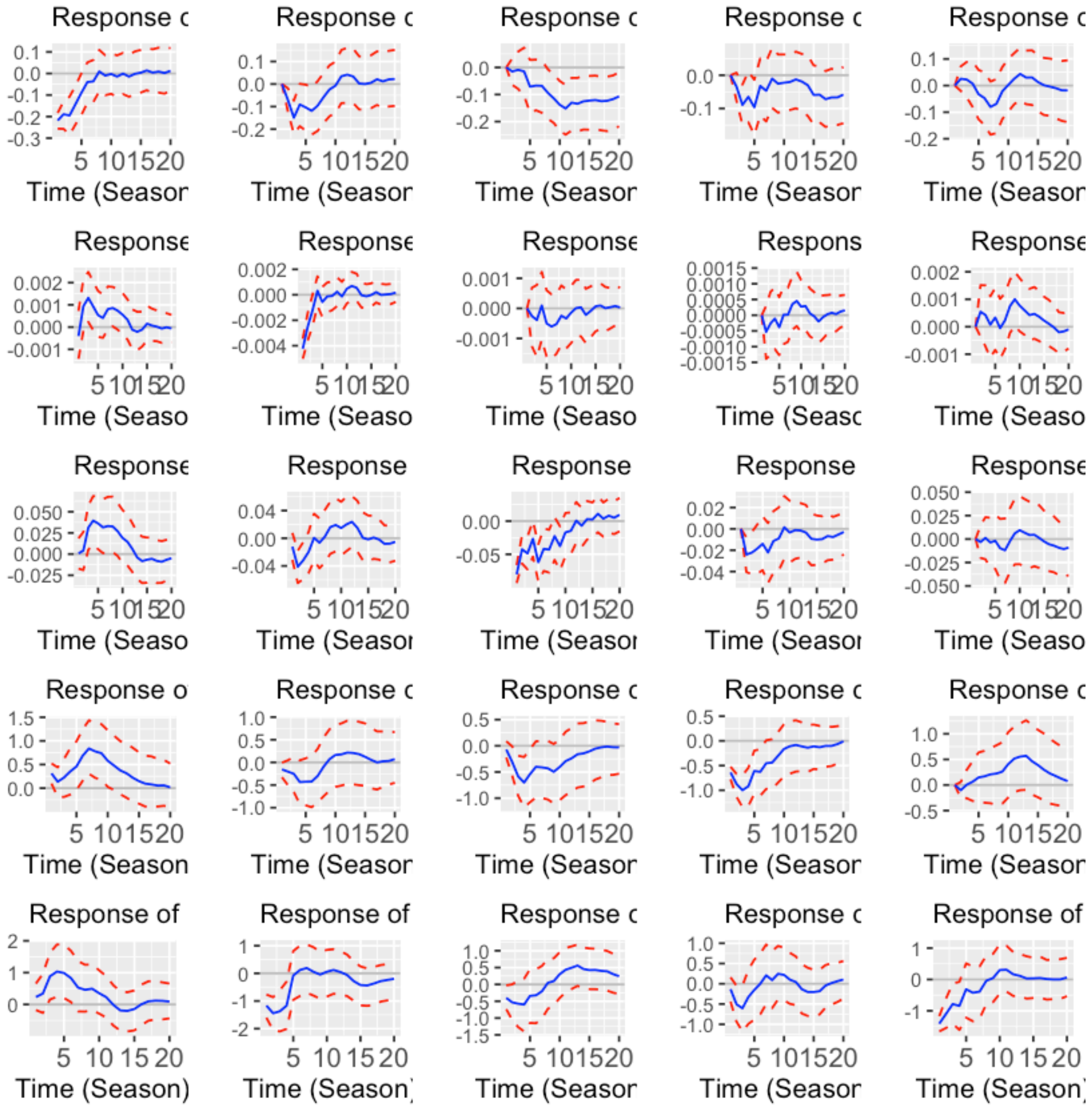
```
ind <- 0
for(i in 1:5){
  for(j in 1:5){
    ind <- ind+1
    nam <- paste("shock", j, "y", i,"_negative", sep = '')
    assign(nam, bind_cols(df_IRF_plot.BS.L_negative[ind], df_IRF_plot.BS.U_negative[in
                          df_IRF_plot.BS.Median_negative[ind], df_IRF_plot.BS.Mean_neg
                          df_IRF_plot_negative[ind]))
    #
    evalStr <- paste0("colnames(", nam, ") <- c('Lower', 'Upper', 'Median', 'Mean', '
    eval(parse(text=evalStr))
    #
    evalStr <- paste0("p", ind, " <- ", "ggplot(",nam,") +geom_hline(yintercept=0, col
    eval(parse(text=evalStr))
  }
}
```

```
New names:
* V1 -> V1...1
* V1 -> V1...2
* V1 -> V1...3
* V1 -> V1...4
* V1 -> V1...5

New names:
* V2 -> V2...1
* V2 -> V2...2
* V2 -> V2...3
* V2 -> V2...4
* V2 -> V2...5

New names:
* V3 -> V3...1
* V3 -> V3...2
* V3 -> V3...3
* V3 -> V3...4
* V3 -> V3...5

New names:
* V4 -> V4...1
* V4 -> V4...2
* V4 -> V4...3
* V4 -> V4...4
* V4 -> V4...5

New names:
* V5 -> V5...1
* V5 -> V5...2
* V5 -> V5...3
* V5 -> V5...4
* V5 -> V5...5

New names:
* V6 -> V6...1
* V6 -> V6...2
* V6 -> V6...3
* V6 -> V6...4
* V6 -> V6...5

New names:
* V7 -> V7...1
* V7 -> V7...2
* V7 -> V7...3
* V7 -> V7...4
* V7 -> V7...5

New names:
* V8 -> V8...1
* V8 -> V8...2
* V8 -> V8...3
* V8 -> V8...4
* V8 -> V8...5

New names:
* V9 -> V9...1
* V9 -> V9...2
* V9 -> V9...3
* V9 -> V9...4
* V9 -> V9...5

New names:
* V10 -> V10...1
* V10 -> V10...2
* V10 -> V10...3
* V10 -> V10...4
* V10 -> V10...5

New names:
* V11 -> V11...1
* V11 -> V11...2
* V11 -> V11...3
* V11 -> V11...4
* V11 -> V11...5

New names:
* V12 -> V12...1
* V12 -> V12...2
* V12 -> V12...3
* V12 -> V12...4
* V12 -> V12...5

New names:
* V13 -> V13...1
* V13 -> V13...2
* V13 -> V13...3
* V13 -> V13...4
* V13 -> V13...5

New names:
* V14 -> V14...1
* V14 -> V14...2
* V14 -> V14...3
* V14 -> V14...4
* V14 -> V14...5

New names:
* V15 -> V15...1
* V15 -> V15...2
* V15 -> V15...3
* V15 -> V15...4
* V15 -> V15...5

New names:
* V16 -> V16...1
* V16 -> V16...2
* V16 -> V16...3
* V16 -> V16...4
* V16 -> V16...5

New names:
* V17 -> V17...1
* V17 -> V17...2
* V17 -> V17...3
* V17 -> V17...4
* V17 -> V17...5

New names:
* V18 -> V18...1
* V18 -> V18...2
* V18 -> V18...3
* V18 -> V18...4
* V18 -> V18...5

New names:
```

```
* V19 -> V19...1
* V19 -> V19...2
* V19 -> V19...3
* V19 -> V19...4
* V19 -> V19...5

New names:
* V20 -> V20...1
* V20 -> V20...2
* V20 -> V20...3
* V20 -> V20...4
* V20 -> V20...5

New names:
* V21 -> V21...1
* V21 -> V21...2
* V21 -> V21...3
* V21 -> V21...4
* V21 -> V21...5

New names:
* V22 -> V22...1
* V22 -> V22...2
* V22 -> V22...3
* V22 -> V22...4
* V22 -> V22...5

New names:
* V23 -> V23...1
* V23 -> V23...2
* V23 -> V23...3
* V23 -> V23...4
* V23 -> V23...5

New names:
* V24 -> V24...1
* V24 -> V24...2
* V24 -> V24...3
* V24 -> V24...4
* V24 -> V24...5

New names:
* V25 -> V25...1
* V25 -> V25...2
* V25 -> V25...3
* V25 -> V25...4
* V25 -> V25...5
```

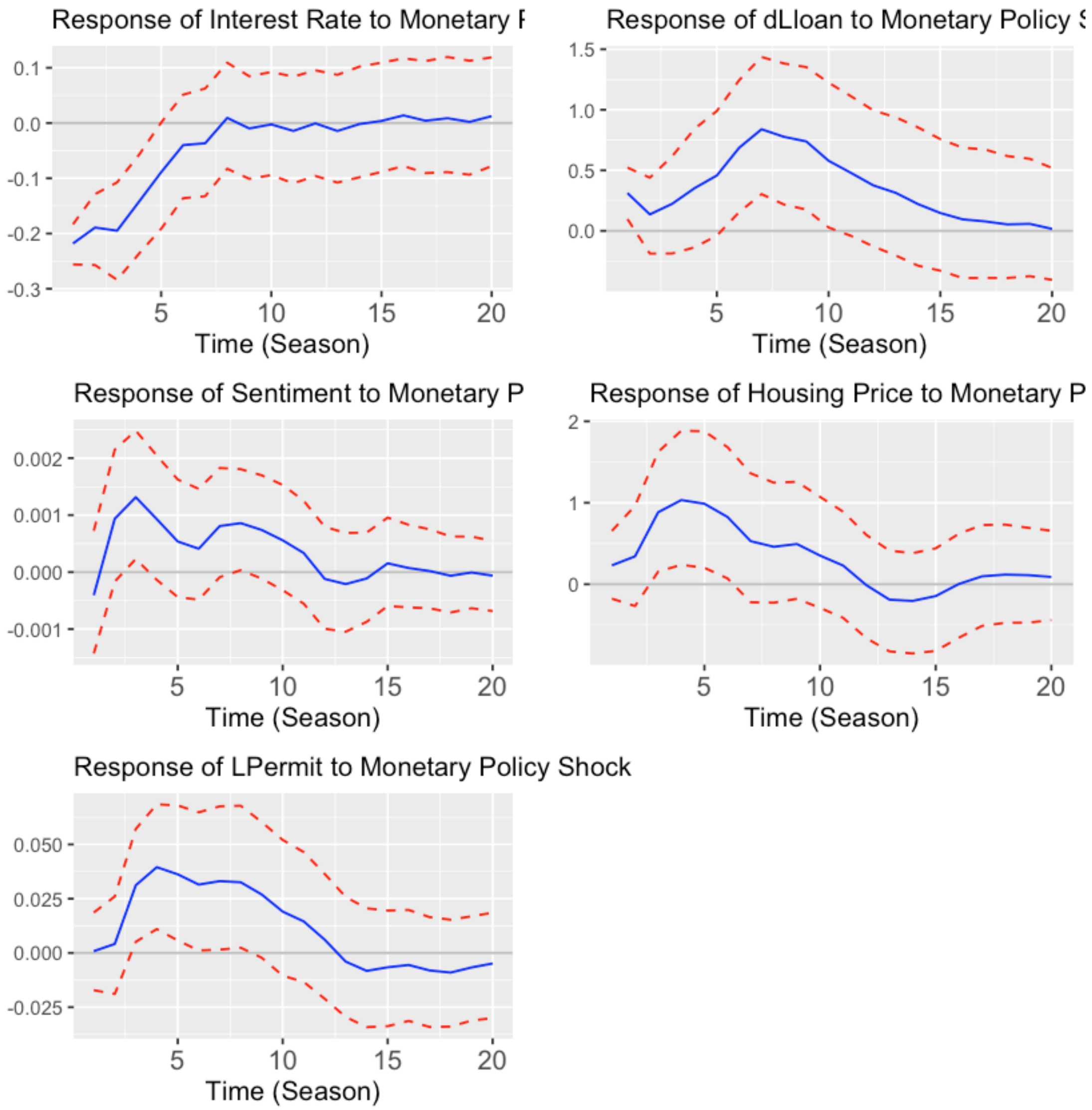
```
In [42]: ## shock1: mp
         p1 <- p1+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Interest Rate to Monetary Policy Shock')+Text_Size_Theme
         p2 <- p2+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Sentiment to Monetary Policy Shock')+Text_Size_Theme
         p3 <- p3+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of LPermit to Monetary Policy Shock')+Text_Size_Theme
         p4 <- p4+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of dLloan to Monetary Policy Shock')+Text_Size_Theme
         p5 <- p5+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Housing Price to Monetary Policy Shock')+Text_Size_Theme

         ## shock2: exp
         p6 <- p6+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Interest Rate to Housing Expectation Shock')+Text_Size_T
         p7 <- p7+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Sentiment to Housing Expectation Shock')+Text_Size_Theme
         p8 <- p8+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of LPermit to Housing Expectation Shock')+Text_Size_Theme
         p9 <- p9+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of dLloan to Housing Expectation Shock')+Text_Size_Theme
         p10 <- p10+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Housing Price to Housing Expectation Shock')+Text_Size_T

         ## shock3: supply
         p11 <- p11+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Interest Rate to Housing Supply Shock')+Text_Size_Theme
         p12 <- p12+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Sentiment to Housing Supply Shock')+Text_Size_Theme
         p13 <- p13+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of LPermit to Housing Supply Shock')+Text_Size_Theme
         p14 <- p14+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of dLloan to Housing Supply Shock')+Text_Size_Theme
         p15 <- p15+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Housing Price to Housing Supply Shock')+Text_Size_Theme




         ## shock4: demand
         p16 <- p16+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Interest Rate to Housing Demand Shock')+Text_Size_Theme
         p17 <- p17+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Sentiment to Housing Demand Shock')+Text_Size_Theme
         p18 <- p18+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of LPermit to Housing Demand Shock')+Text_Size_Theme
         p19 <- p19+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of dLloan to Housing Demand Shock')+Text_Size_Theme
         p20 <- p20+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Housing Price to Housing Demand Shock')+Text_Size_Theme



         ## shock5: sp
         p21 <- p21+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Interest Rate to Residual Shock')+Text_Size_Theme
         p22 <- p22+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Sentiment to Residual Shock')+Text_Size_Theme
         p23 <- p23+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of LPermit to Residual Shock')+Text_Size_Theme
         p24 <- p24+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of dLloan to Residual Shock')+Text_Size_Theme
         p25 <- p25+labs(x = 'Time (Season)',
                 y = '',
                 title = 'Response of Housing Price to Residual Shock')+Text_Size_Theme
```

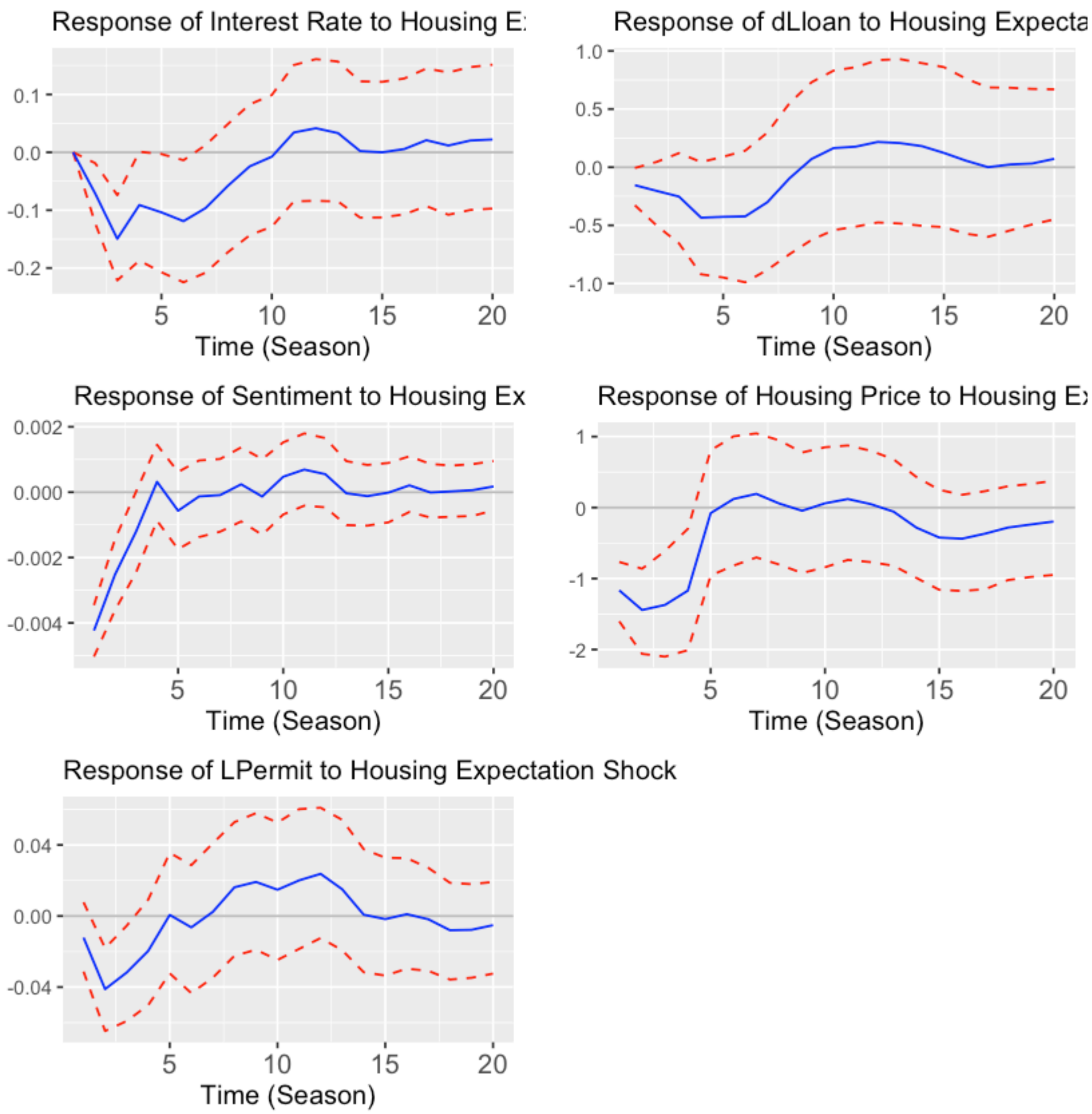
```
In [43]: multiplot(p1,p2,p3,p4,p5,
                   p6,p7,p8,p9,p10,
                   p11,p12,p13,p14,p15,
                   p16,p17,p18,p19,p20,
                   p21,p22,p23,p24,p25,
                   cols = 5)
```

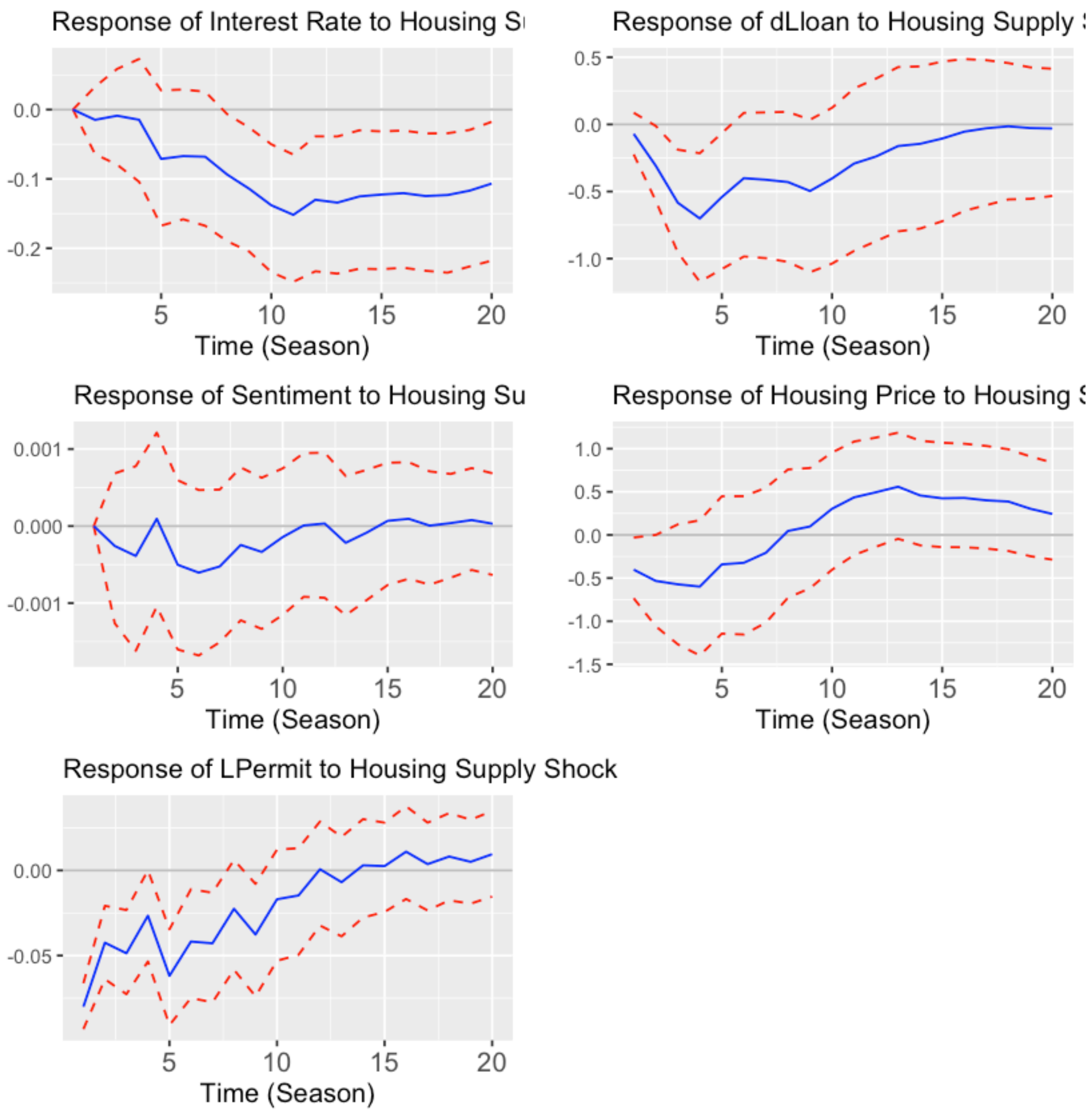


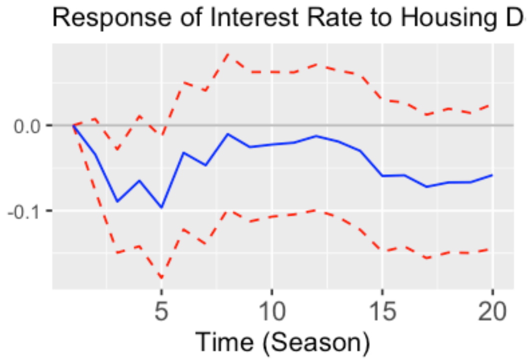
```
In [44]: # For shock 1
         multiplot(p1,p2,p3,p4,p5,
                   cols = 2)
         # For shock 2
         multiplot(p6,p7,p8,p9,p10,
                   cols = 2)
         # For shock 3
         multiplot(p11,p12,p13,p14,p15,
                   cols = 2)
         # For shock 4
```

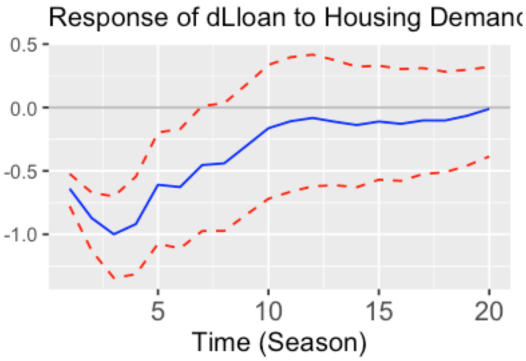
```
multiplot(p16,p17,p18,p19,p20,
          cols = 2)
# For shock 5
multiplot(p21,p22,p23,p24,p25,
          cols = 2)
```

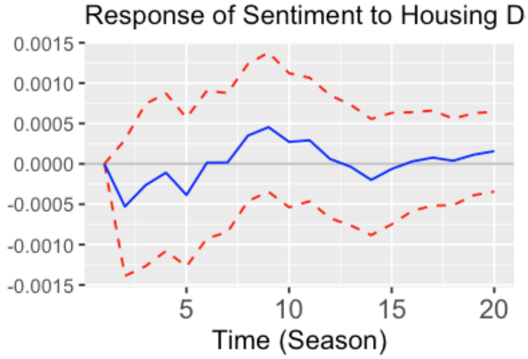


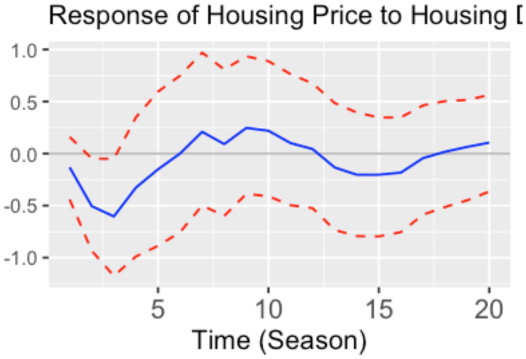

Response of Interest Rate to Housing E:
0.1
0.0
-0.1
-0.2
5
10
15
20
Time (Season)
Response of dLloan to Housing Expecta
1.0
0.5
0.0
-0.5
-1.0
5
10
15
20
Time (Season)
Response of Sentiment to Housing Ex
0.002
0.000
-0.002
-0.004
5
10
15
20
Time (Season)
Response of Housing Price to Housing E:
1
0
-1
-2
5
10
15
20
Time (Season)
Response of LPermit to Housing Expectation Shock
0.04
0.00
-0.04
5
10
15
20
Time (Season)

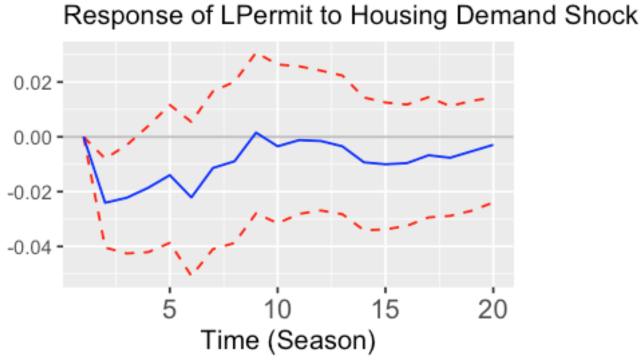

Response of Interest Rate to Housing S
0.0
-0.1
-0.2
5
10
15
20
Time (Season)
Response of dLloan to Housing Supply
0.5
0.0
-0.5
-1.0
5
10
15
20
Time (Season)
Response of Sentiment to Housing Su
0.001
0.000
-0.001
5
10
15
20
Time (Season)
Response of Housing Price to Housing
1.0
0.5
0.0
-0.5
-1.0
-1.5
5
10
15
20
Time (Season)
Response of LPermit to Housing Supply Shock
0.00
-0.05
5
10
15
20
Time (Season)

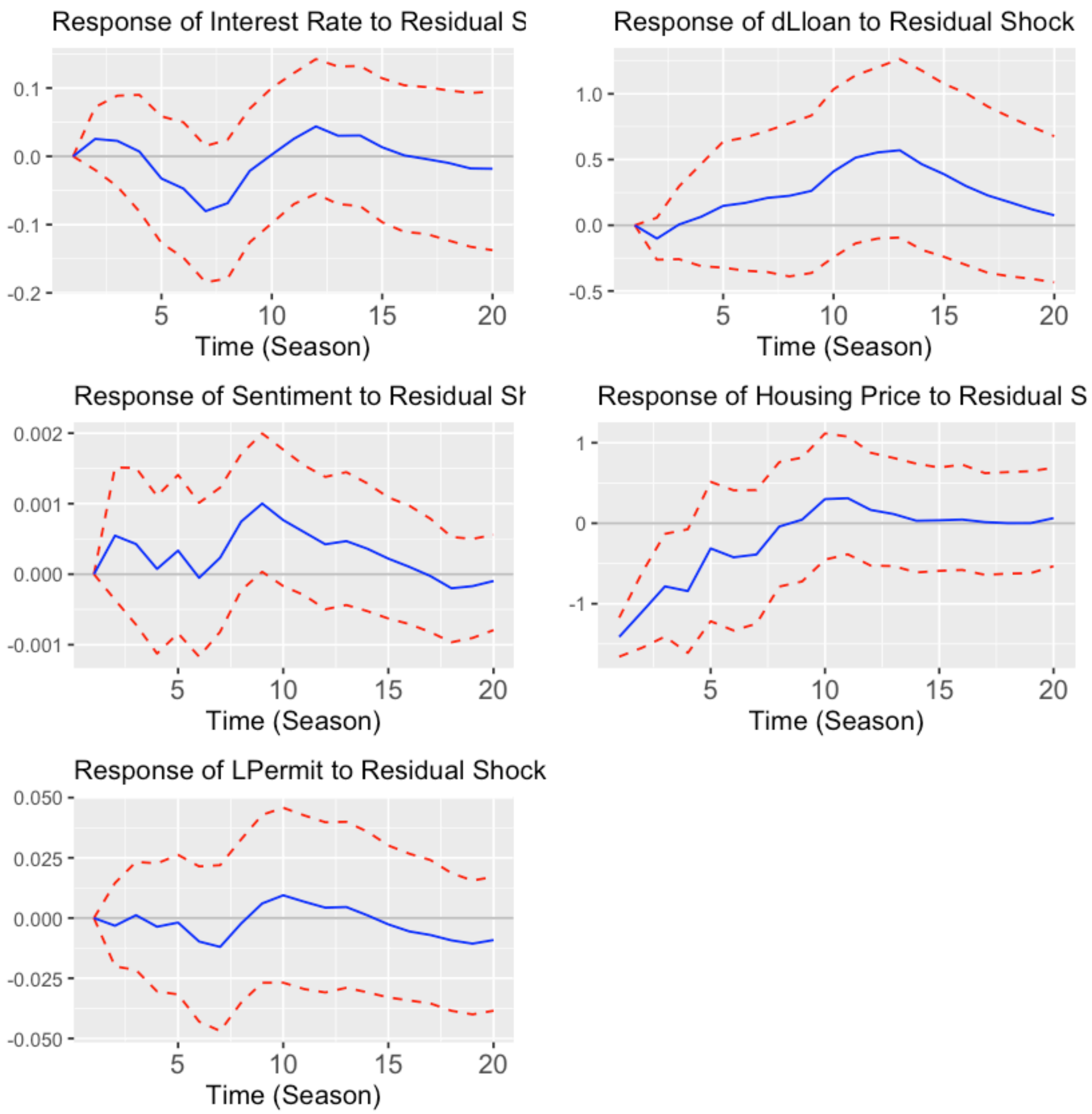

Response of Interest Rate to Housing D


Response of dLloan to Housing Demanc


Response of Sentiment to Housing D

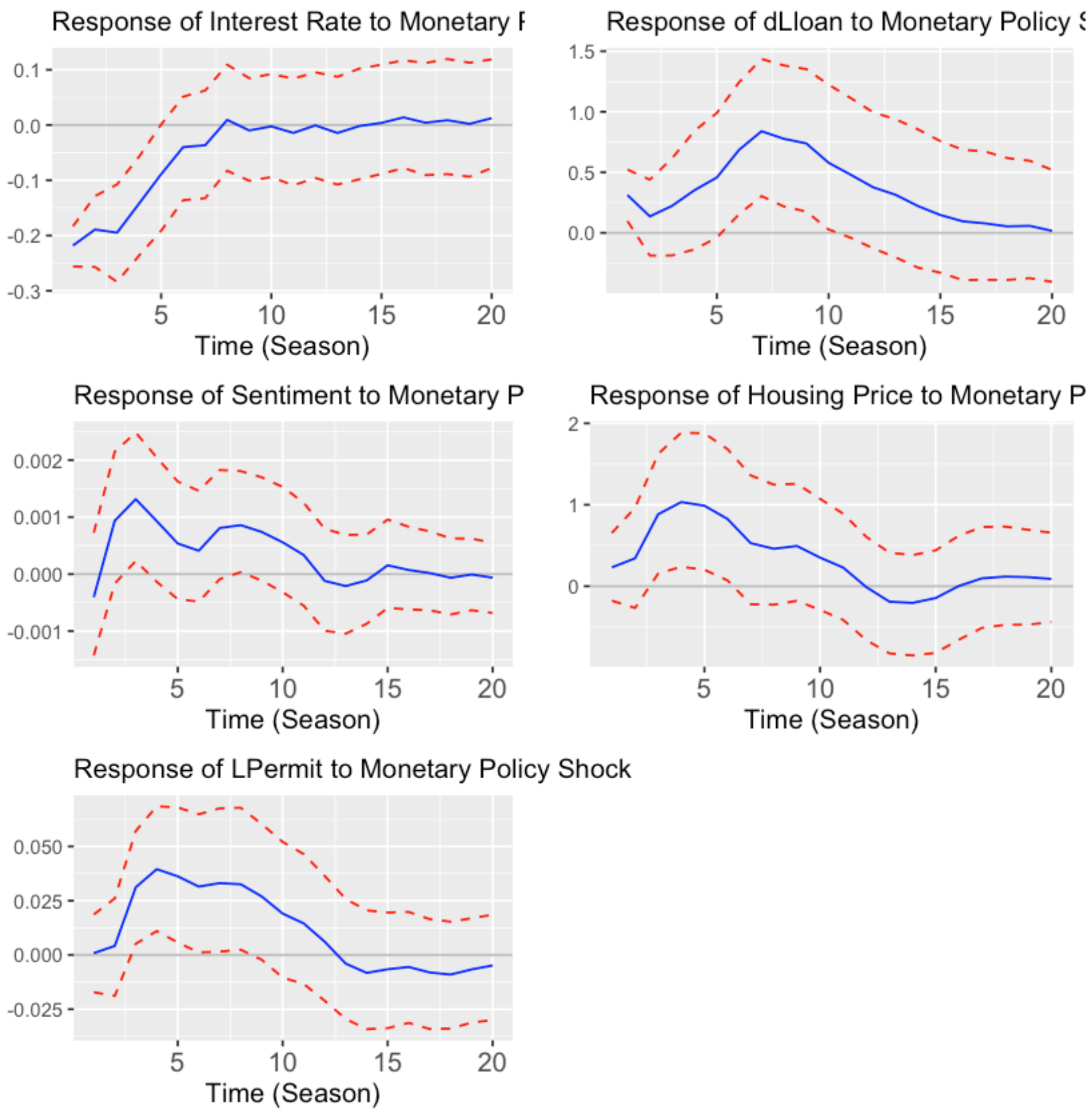

Response of Housing Price to Housing I

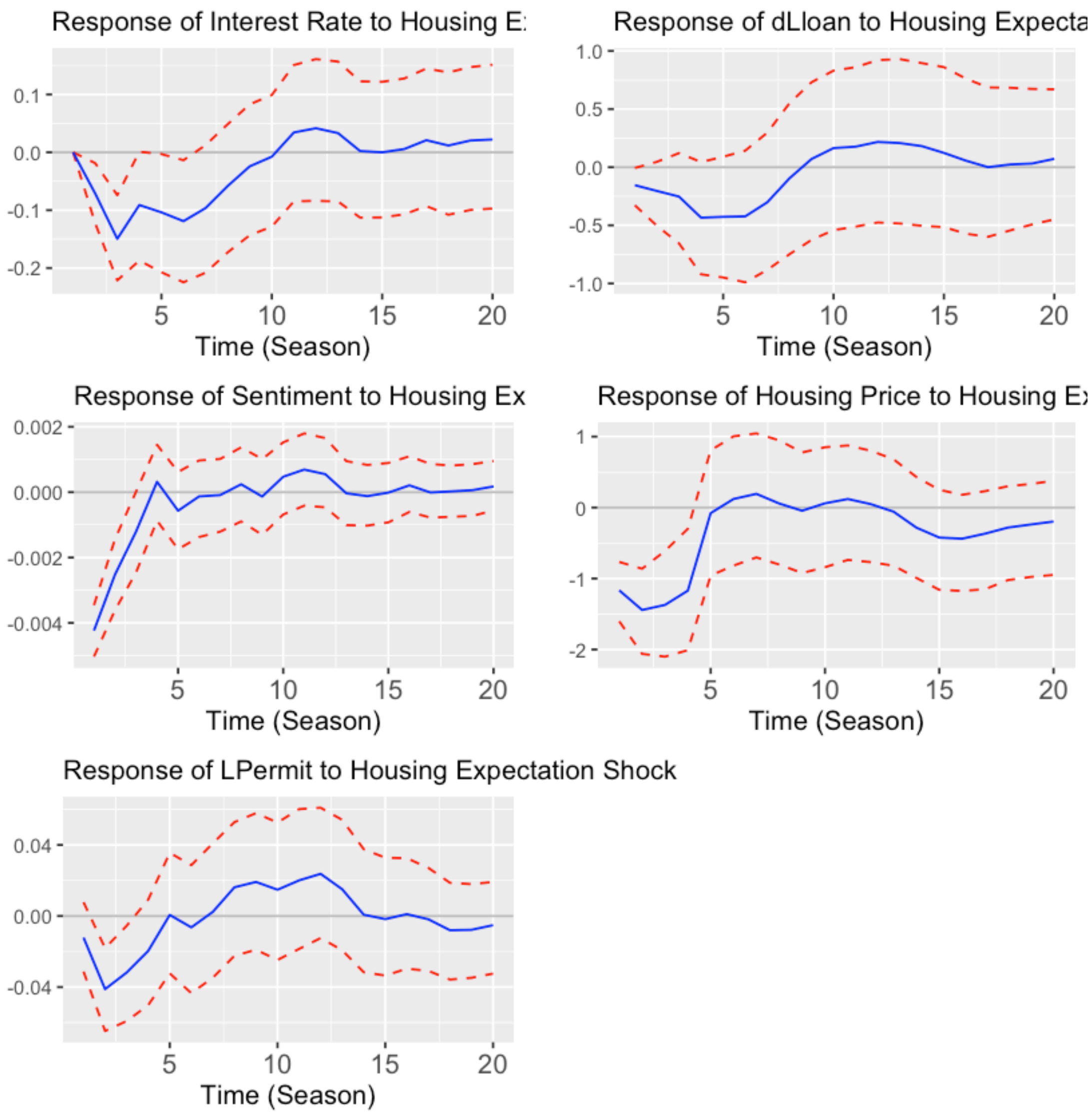

Response of LPermit to Housing Demand Shock



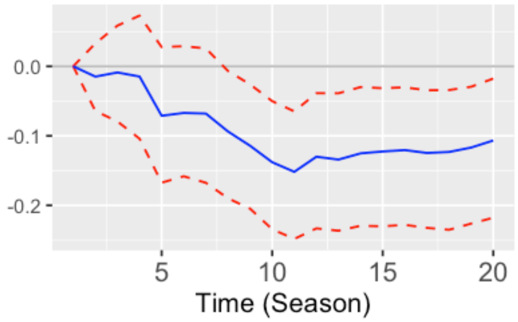
```
In [45]: # save plot
         # shock 1
         ggsave(filename = "result/figure/0219_m1_IRF_shock1_negative.png",
                plot = multiplot(p1,p2,p3,p4,p5, cols = 2),
                width = 30, height = 20, units = "cm",
                device = "png")

         # shock 2
         ggsave(filename = "result/figure/0219_m1_IRF_shock2_negative.png",
                plot = multiplot(p6,p7,p8,p9,p10, cols = 2),
                width = 30, height = 20, units = "cm",
                device = "png")
```

```
# shock 3
ggsave(filename = "result/figure/0219_m1_IRF_shock3_negative.png",
       plot = multiplot(p11,p12,p13,p14,p15, cols = 2),
       width = 30, height = 20, units = "cm",
       device = "png")

# shock 4
ggsave(filename = "result/figure/0219_m1_IRF_shock4_negative.png",
       plot = multiplot(p16,p17,p18,p19,p20, cols = 2),
       width = 30, height = 20, units = "cm",
       device = "png")

# shock 5
ggsave(filename = "result/figure/0219_m1_IRF_shock5_negative.png",
       plot = multiplot(p21,p22,p23,p24,p25, cols = 2),
       width = 30, height = 20, units = "cm",
       device = "png")
```

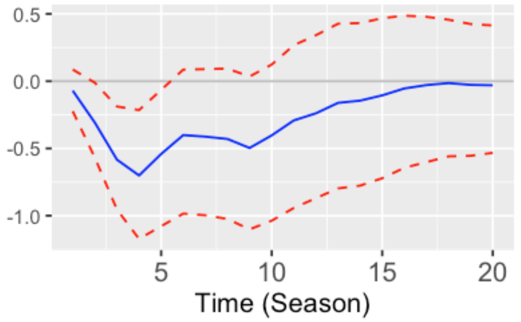

Response of Interest Rate to Monetary F
0.1
0.0
-0.1
-0.2
-0.3
5
10
15
20
Time (Season)
Response of dLloan to Monetary Policy S
1.5
1.0
0.5
0.0
5
10
15
20
Time (Season)
Response of Sentiment to Monetary P
0.002
0.001
0.000
-0.001
5
10
15
20
Time (Season)
Response of Housing Price to Monetary P
2
1
0
5
10
15
20
Time (Season)
Response of LPermit to Monetary Policy Shock
0.050
0.025
0.000
-0.025
5
10
15
20
Time (Season)


Response of Interest Rate to Housing E
0.1
0.0
-0.1
-0.2
5
10
15
20
Time (Season)
Response of dLloan to Housing Expecta
1.0
0.5
0.0
-0.5
-1.0
5
10
15
20
Time (Season)
Response of Sentiment to Housing Ex
0.002
0.000
-0.002
-0.004
5
10
15
20
Time (Season)
Response of Housing Price to Housing E
1
0
-1
-2
5
10
15
20
Time (Season)
Response of LPermit to Housing Expectation Shock
0.04
0.00
-0.04
5
10
15
20
Time (Season)

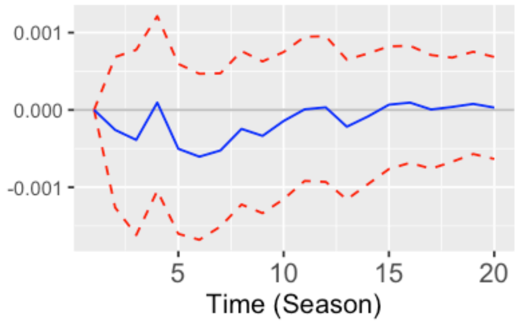
Response of Interest Rate to Housing Sı



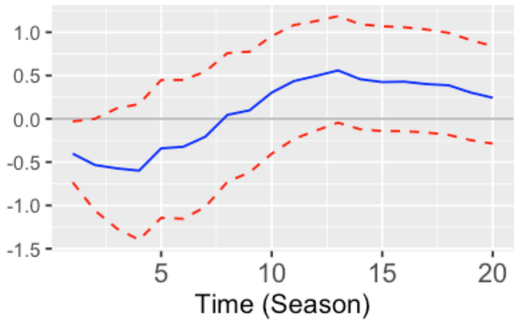
Response of dLloan to Housing Supply :



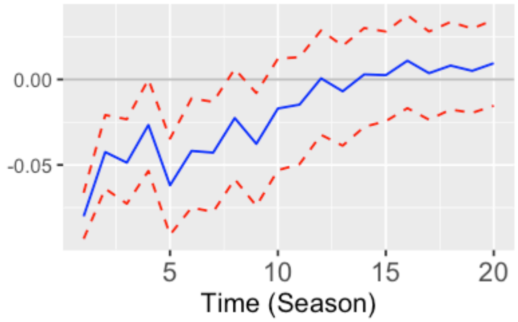
Response of Sentiment to Housing Su

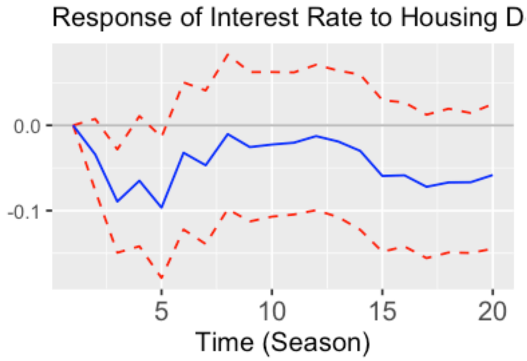


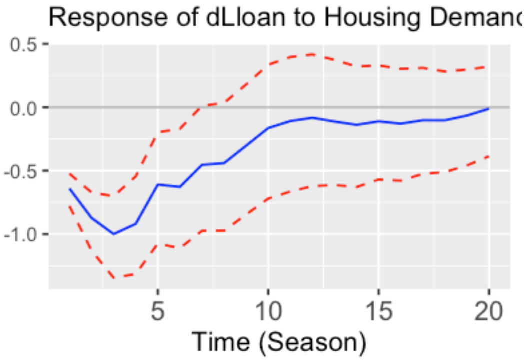
Response of Housing Price to Housing S

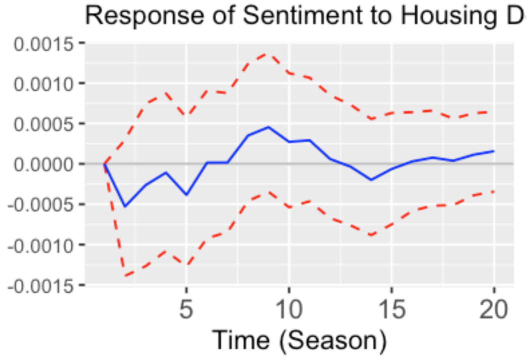


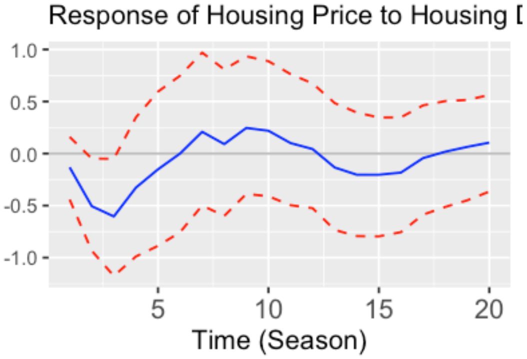
Response of LPermit to Housing Supply Shock

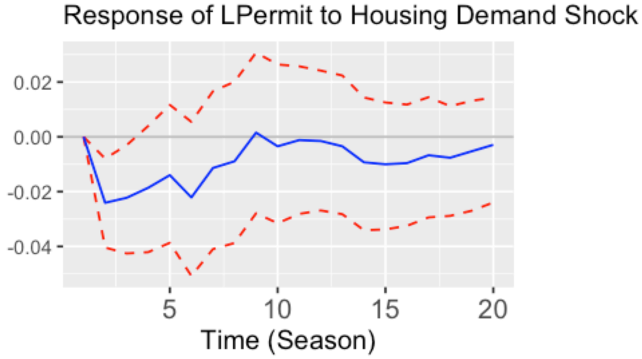


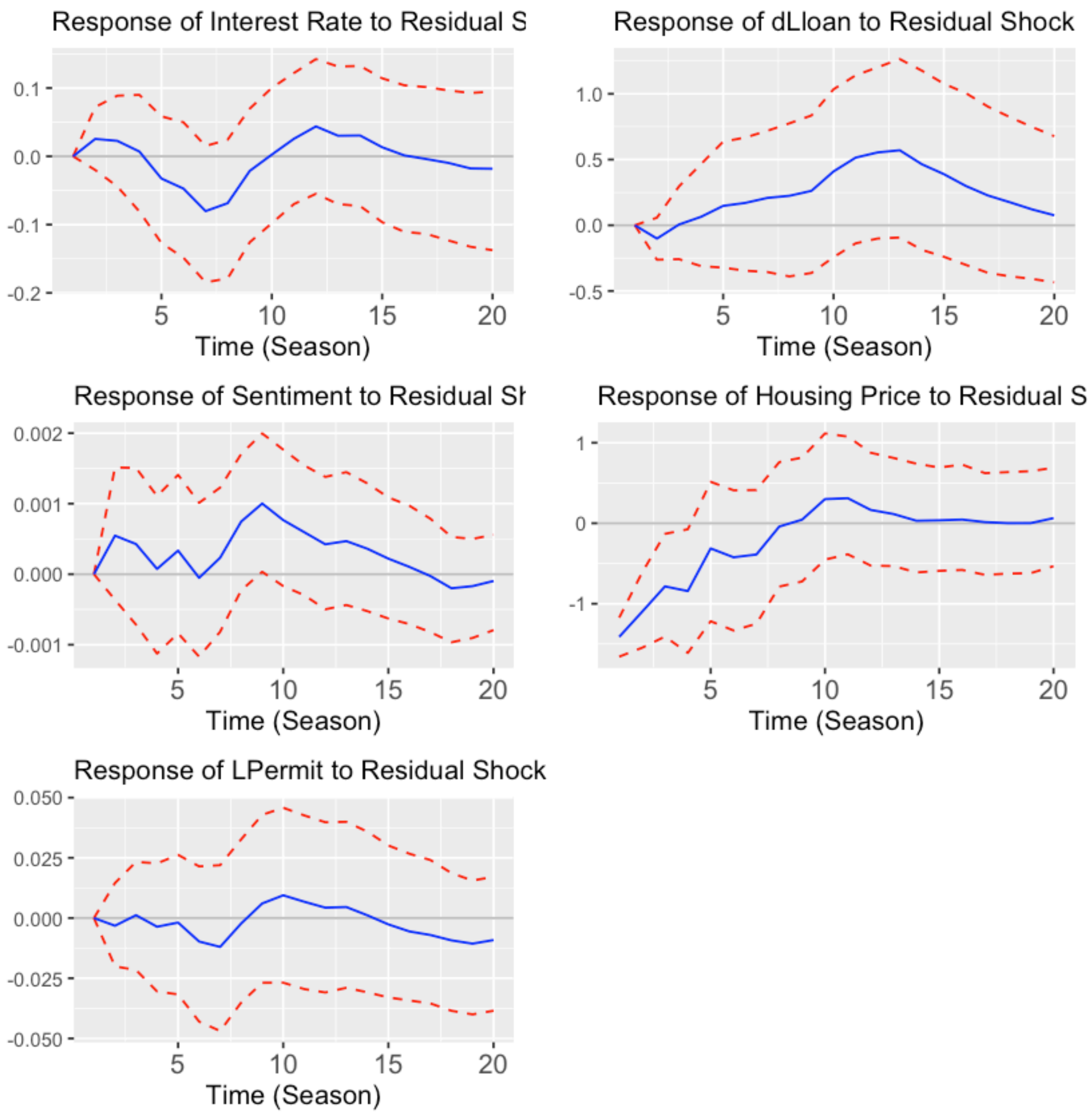

Response of Interest Rate to Housing D


Response of dLloan to Housing Demand


Response of Sentiment to Housing D


Response of Housing Price to Housing D


Response of LPermit to Housing Demand Shock



4.1 Variance Decomposition

```
In [46]: # `ddTheta` IRF ()
         # m
         SVAR_AB_VarDecomp <- VAR.svardecomp.AB(m = 5, By, VAR.P,
                                               AMat, BMat, h=(hrz+1),
                                               Const=TRUE, ddTheta = SVAR_AB_IRF)
         # head(SVAR_AB_VarDecomp*100)
         # tail(SVAR_AB_VarDecomp*100)
         (SVAR_AB_VarDecomp*100)
```

A matrix: 20 Œ 5 of type dbl

h=1	1.599858	37.18937	4.385282	0.4320429	56.39344
h=2	2.316156	45.67997	5.634016	3.0285947	43.34127
h=3	7.989155	46.94600	6.294991	4.7693250	34.00053
h=4	13.299399	45.03495	6.770022	4.1526396	30.74299
h=5	18.853358	41.30597	6.883715	3.8318558	29.12510
h=6	22.370682	38.17730	7.071972	3.5302057	28.84984
h=7	23.689622	36.32726	7.116281	3.7139886	29.15285
h=8	24.936748	35.61240	6.963412	3.7801441	28.70730
h=9	26.193792	34.98162	6.742617	4.1928626	27.88911
h=10	26.924141	34.44031	6.712728	4.5746108	27.34821
h=11	27.278225	33.90744	7.029447	4.7159749	27.06891
h=12	27.094738	33.62618	7.683179	4.7751874	26.82072
h=13	26.722538	33.36541	8.681520	4.7143830	26.51615
h=14	26.297295	33.47574	9.418352	4.7274140	26.08120
h=15	25.724070	33.89080	10.101359	4.7696512	25.51412
h=16	25.115997	34.30733	10.820613	4.8344525	24.92161
h=17	24.688936	34.55370	11.551101	4.7840111	24.42226
h=18	24.361412	34.61750	12.297563	4.7159898	24.00754
h=19	24.118545	34.68388	12.857251	4.6535687	23.68676
h=20	23.919991	34.76675	13.253594	4.6099126	23.44976

```
In [47]: # output entire table
         VD_TABLE <- (SVAR_AB_VarDecomp*100) %>% as.tibble %>% select(mp=1,exp=2,hs=3,hd=4,hp=
         write.table(VD_TABLE, file = "result/table/VD_TABLE.csv", sep = ",", row.names = FALS
```

```
In [48]: # table
         SVAR_VD <- (SVAR_AB_VarDecomp*100) %>% as.tibble()
         SVAR_VD <- SVAR_VD %>% filter(row_number(V1) %in% c(1,2,4,8,12,16,20)) %>%
             mutate(period = c(1,2,4,8,12,16,20)) %>%
             select(period = period,
                    mp = V1,
                    exp = V2,
                    hs = V3,
                    hd = V4,
                    sp = V5) %>%
             mutate(period = as.character(period),
                    mp = round(mp, digits = 2),
                    exp = round(exp, digits = 2),
                    hs = round(hs, digits = 2),
                    hd = round(hd, digits = 2),
                    sp = round(sp, digits = 2))
         SVAR_VD
```

A tibble: 7 Œ 6

period <chr>	mp <dbl>	exp <dbl>	hs <dbl>	hd <dbl>	sp <dbl>
1	1.60	37.19	4.39	0.43	56.39
2	2.32	45.68	5.63	3.03	43.34
4	13.30	45.03	6.77	4.15	30.74
8	24.94	35.61	6.96	3.78	28.71
12	27.28	33.91	7.03	4.72	27.07
16	26.30	33.48	9.42	4.73	26.08
20	23.92	34.77	13.25	4.61	23.45

```
In [49]: library(xtable)
```

```
In [50]: tab_VD <- xtable(SVAR_VD, caption= "", align=c("c","c","c","c","c","c","c"))
         print(tab_VD, include.rownames=FALSE)
```

```
% latex table generated in R 3.6.2 by xtable 1.8-4 package
% Fri Oct 15 00:34:10 2021
\begin{table}[ht]
\centering
\begin{tabular}{cccccc}
  \hline
period & mp & exp & hs & hd & sp \\
  \hline
1 & 1.60 & 37.19 & 4.39 & 0.43 & 56.39 \\
  2 & 2.32 & 45.68 & 5.63 & 3.03 & 43.34 \\
  4 & 13.30 & 45.03 & 6.77 & 4.15 & 30.74 \\
  8 & 24.94 & 35.61 & 6.96 & 3.78 & 28.71 \\
  12 & 27.28 & 33.91 & 7.03 & 4.72 & 27.07 \\
  16 & 26.30 & 33.48 & 9.42 & 4.73 & 26.08 \\
  20 & 23.92 & 34.77 & 13.25 & 4.61 & 23.45 \\
   \hline
\end{tabular}
\caption{}
\end{table}
```

```
In [51]: print(tab_VD, file="result/table/VD_0219_m1.tex",
               include.rownames=FALSE,
               append=T, table.placement = "h",
               caption.placement="bottom", hline.after=seq(from=-1,to=nrow(tab_VD),by=1))
```

4.2 Historical Decomposition

- Quandt-Andrews unkonown break point test
- break points: 2003Q3 2006Q4 2009Q2 2013Q3 2016Q2

1. 1991Q1-2020Q3
2. subsample 1: 1991Q1-2003Q3 ()

3. subsample 2: 2003Q3-2006Q4 ()
4. subsample 3: 2006Q4-2009Q2 ()
5. subsample 4: 2009Q2-2013Q3 ()
6. subsample 5: 2013Q3-2016Q2 ()
7. subsample 6: 2016Q2-2020Q3 ()

```
In [52]: nrow(By)
```

115

```
In [53]: if(hrz<nrow(By)){
              SVAR_AB_IRF <- VAR.svarirf.AB(By, VAR.P, Amat, Bmat, h = nrow(By), CONST, SVAR_AB_
         }
```

```
In [54]: SVAR_AB_HistDecomp <- VAR.svarhist.AB(By, VAR.P, Amat, Bmat, CONST)
         dim(SVAR_AB_HistDecomp)
```

1. 115 2. 25

```
In [55]: #----- Base Project  -----#
         SVAR_AB_Hist.c0 = VAR.baseproject(By, VAR.P, CONST)
         head(SVAR_AB_Hist.c0)
         dim(SVAR_AB_Hist.c0)
         dim(By)
```

A matrix: 6 Œ 5 of type dbl

6.3733	0.01314865	9.867342	25.93350	14.264570
7.7187	0.01737129	10.115044	30.67660	12.322986
6.9243	0.01875118	10.020159	36.25574	11.155613
6.2483	0.01988772	9.906583	32.59566	7.839785
7.2170	0.01940172	9.810550	27.92726	4.962686
6.7253	0.01760358	9.940783	23.11461	-1.023725

1. 115 2. 5
1. 115 2. 5

```
In [56]: #
         head(By-SVAR_AB_Hist.c0, 10)
         # VAR.Plag0
```

A matrix: 10 Œ 5 of type dbl

R	Sent	LPermit	dLloan	dLhp
0.00000000	0.000000000	0.000000000	0.000000	0.0000000
0.00000000	0.000000000	0.000000000	0.000000	0.0000000
0.00000000	0.000000000	0.000000000	0.000000	0.0000000
0.00000000	0.000000000	0.000000000	0.000000	0.0000000
0.00000000	0.000000000	0.000000000	0.000000	0.0000000
0.00000000	0.000000000	0.000000000	0.000000	0.0000000
0.00000000	0.000000000	0.000000000	0.000000	0.0000000
0.01141845	0.002499533	-0.007586097	1.050678	-0.3862832
-0.03637027	0.004828356	-0.050348847	2.794793	1.4508424
-0.32278865	0.003878028	0.032943436	4.435096	2.7793703

```
In [57]: #
         ## shock1: monetary policy shock
         head(SVAR_AB_HistDecomp[,c(1,6,11,16,21)], 10)
```

```
                              NA            NA              NA              NA            NA
                              NA            NA              NA              NA            NA
                              NA            NA              NA              NA            NA
                              NA            NA              NA              NA            NA
A matrix: 10 Œ 5 of type dbl  NA            NA              NA              NA            NA
                              NA            NA              NA              NA            NA
                              NA            NA              NA              NA            NA
                              0.01141845    0.00001875854   -0.00004584307  -0.01615528   -0.01278078
                              -0.14831627   -0.00031460433  0.00043365968   0.21723384    0.15878008
                              -0.64521052   -0.00016917821  0.00319318803   0.81394690    0.78636029
```

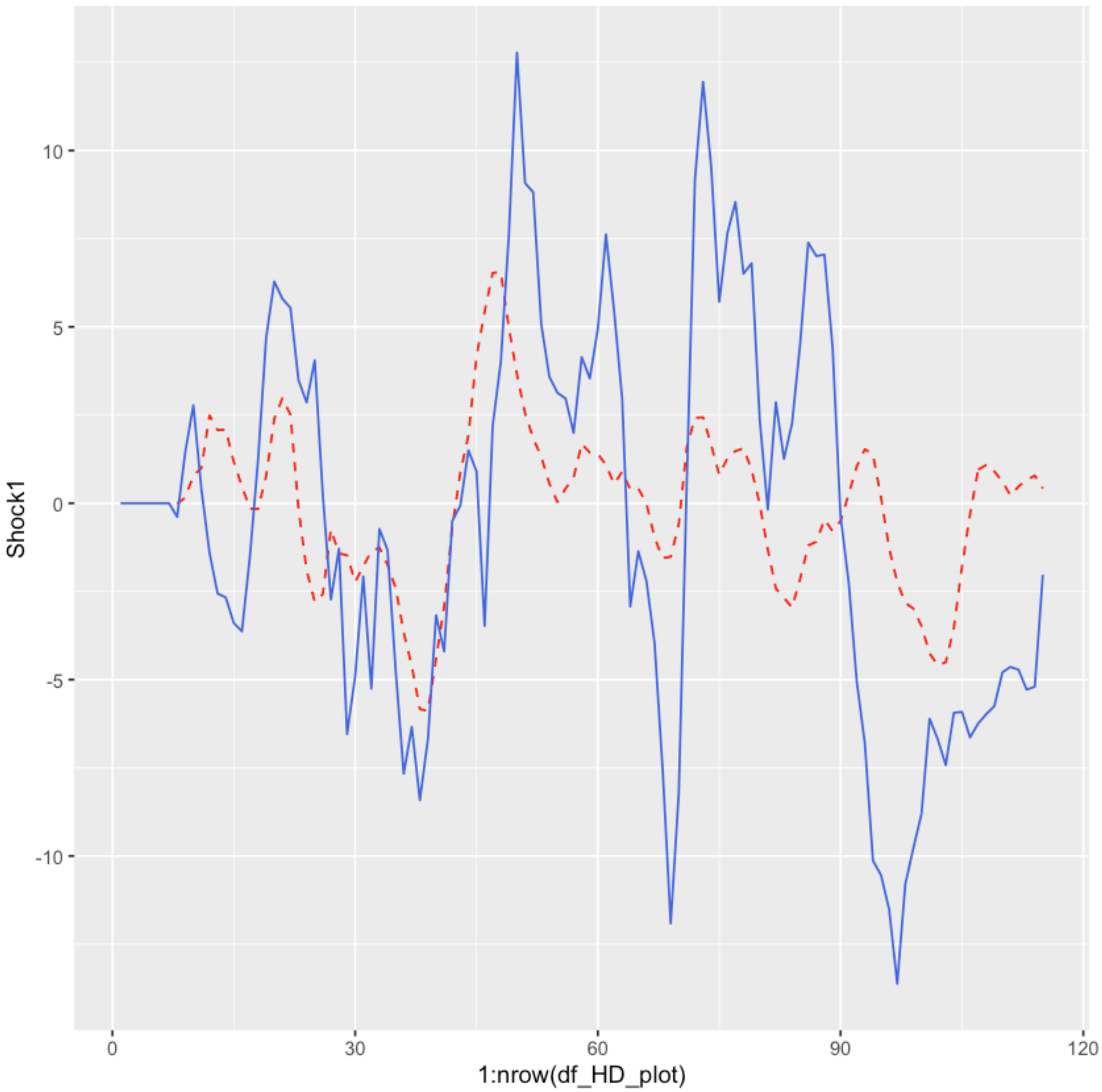
```
In [58]: ##baseline hp shock1  hp
         tail(cbind((By-SVAR_AB_Hist.c0)[,5],
              (SVAR_AB_HistDecomp[,c(1,6,11,16,21)])[,5]))
```

```
                             [110,] | -4.793248   0.6144680
                             [111,] | -4.639120   0.2197710
A matrix: 6 Œ 2 of type dbl  [112,] | -4.722549   0.4810390
                             [113,] | -5.281133   0.6558706
                             [114,] | -5.200768   0.7851396
                             [115,] | -2.025775   0.4209959
```

4.2.1 monetary policy shock

```
In [59]: df_HD_plot <- bind_cols((By-SVAR_AB_Hist.c0)[,5],
                                 (SVAR_AB_HistDecomp[,c(1,6,11,16,21)])[,5])
         colnames(df_HD_plot) <- c("BaseLine", "Shock1")
         ggplot(df_HD_plot)+
           geom_line(aes(x = 1:nrow(df_HD_plot), y = Shock1), col = 'red', linetype = "dashed"
           geom_line(aes(x = 1:nrow(df_HD_plot), y = BaseLine), col = 'royalblue')
```

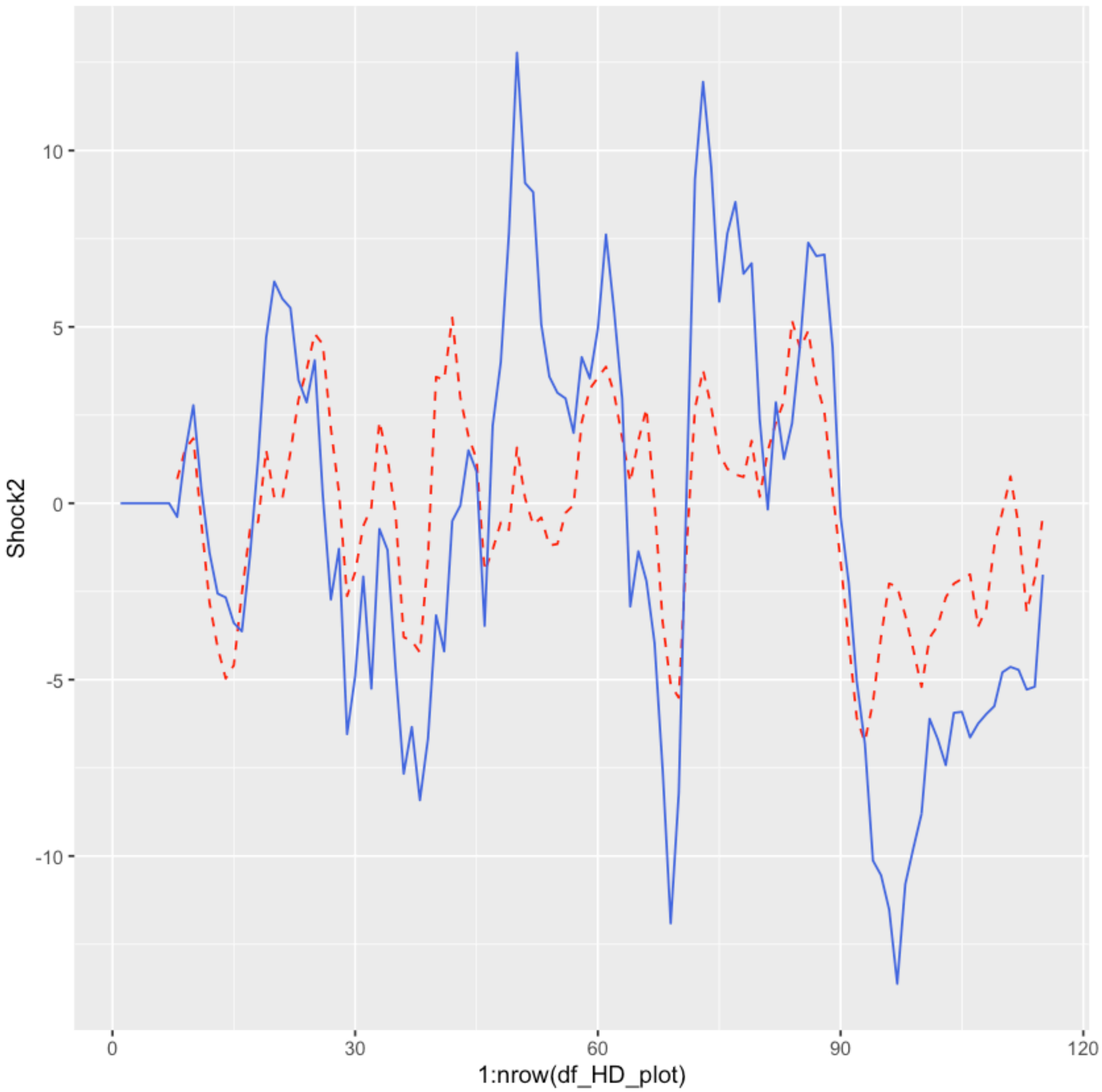
```
New names:
* NA -> ...1
* NA -> ...2
```



4.2.2 expectation shock

```
In [60]: df_HD_plot <- bind_cols((By-SVAR_AB_Hist.c0)[,5],
                                 (SVAR_AB_HistDecomp[,c(2,7,12,17,22)])[,5])
         colnames(df_HD_plot) <- c("BaseLine", "Shock2")
         ggplot(df_HD_plot)+
           geom_line(aes(x = 1:nrow(df_HD_plot), y = Shock2), col = 'red', linetype = "dashed")
           geom_line(aes(x = 1:nrow(df_HD_plot), y = BaseLine), col = 'royalblue')
```

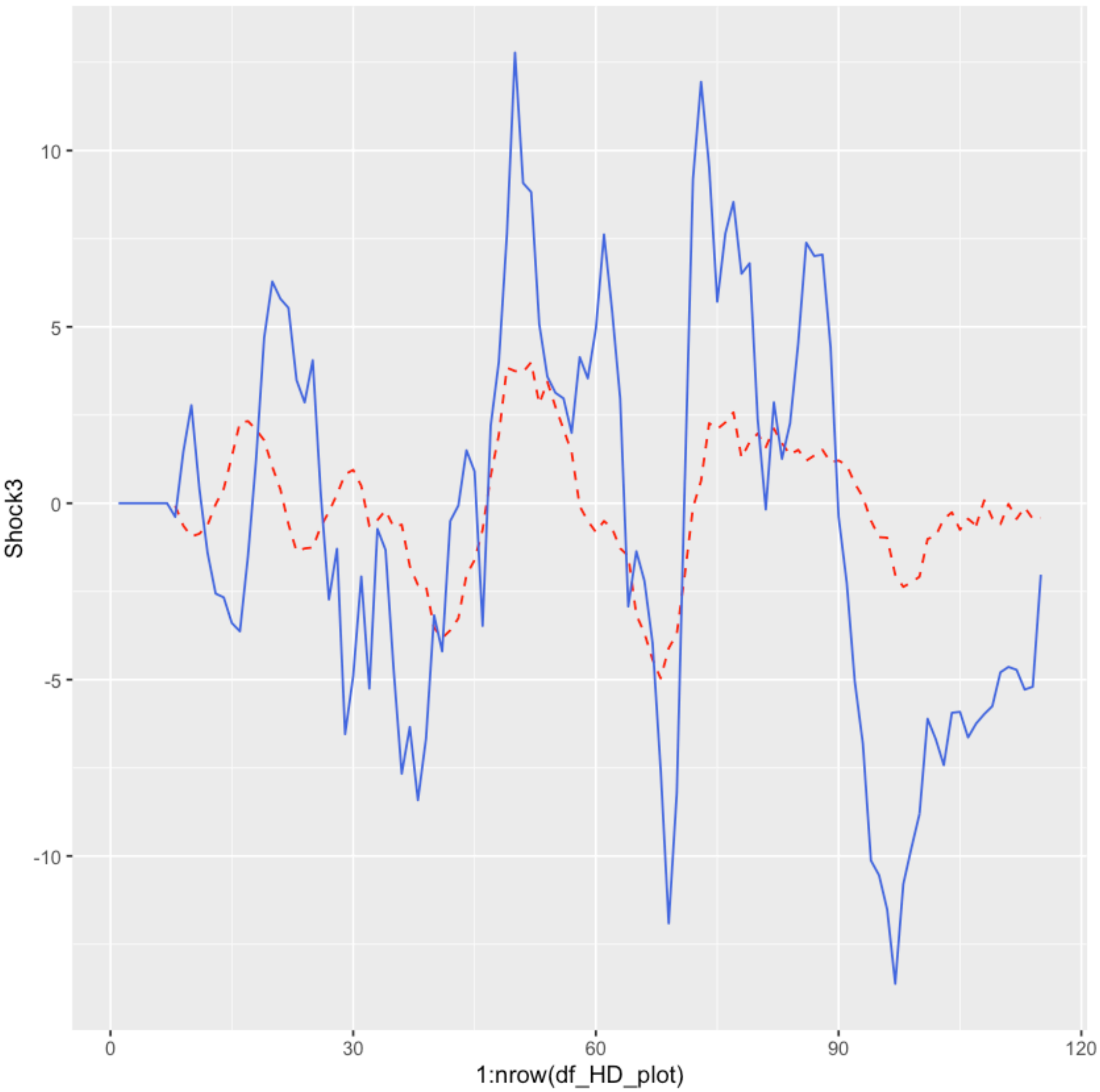
```
New names:
* NA -> ...1
* NA -> ...2
```



4.2.3 supply shock

```
In [61]: df_HD_plot <- bind_cols((By-SVAR_AB_Hist.c0)[,5],
                                 (SVAR_AB_HistDecomp[,c(3,8,13,18,23)])[,5])
         colnames(df_HD_plot) <- c("BaseLine", "Shock3")
         ggplot(df_HD_plot)+
           geom_line(aes(x = 1:nrow(df_HD_plot), y = Shock3), col = 'red', linetype = "dashed"
           geom_line(aes(x = 1:nrow(df_HD_plot), y = BaseLine), col = 'royalblue')
```

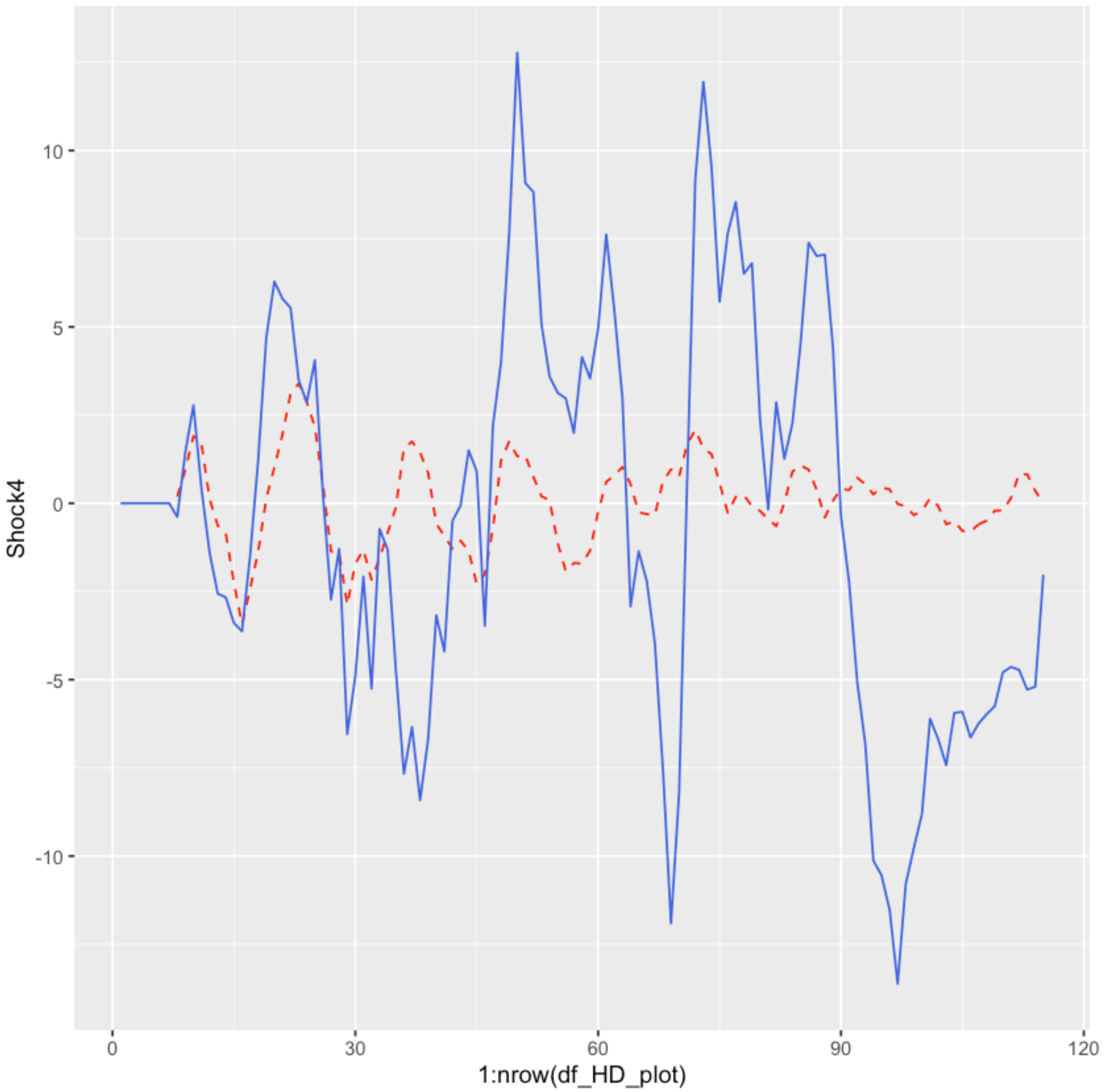
```
New names:
* NA -> ...1
* NA -> ...2
```



4.2.4 housing demand shock

```
In [62]: df_HD_plot <- bind_cols((By-SVAR_AB_Hist.c0)[,5],
                                 (SVAR_AB_HistDecomp[,c(4,9,14,19,24)])[,5])
         colnames(df_HD_plot) <- c("BaseLine", "Shock4")
         ggplot(df_HD_plot)+
           geom_line(aes(x = 1:nrow(df_HD_plot), y = Shock4), col = 'red', linetype = "dashed")
           geom_line(aes(x = 1:nrow(df_HD_plot), y = BaseLine), col = 'royalblue')
```

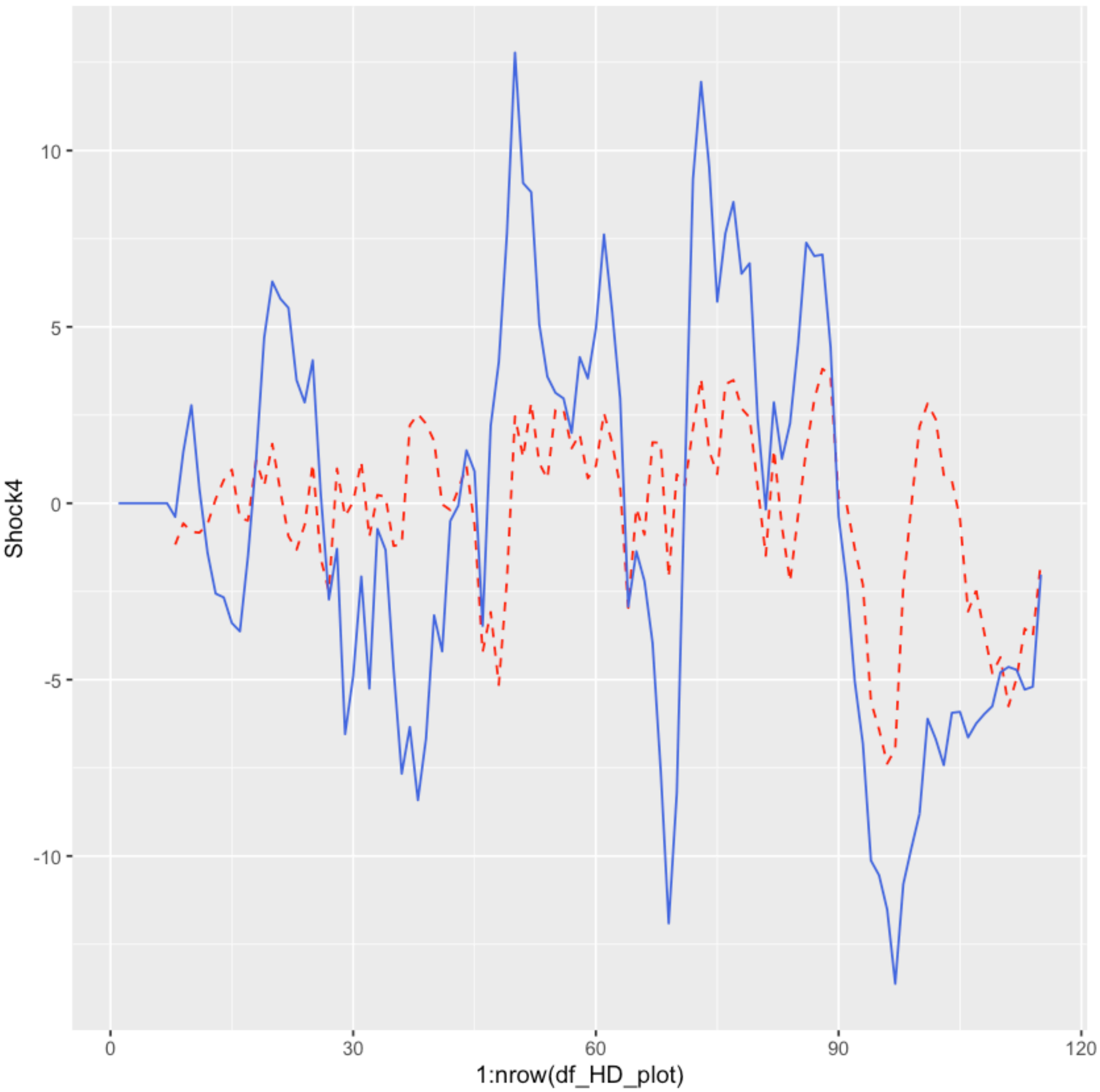
```
New names:
* NA -> ...1
* NA -> ...2
```



4.2.5 Residual shock

```
In [63]: df_HD_plot <- bind_cols((By-SVAR_AB_Hist.c0)[,5],
                                 (SVAR_AB_HistDecomp[,c(5,10,15,20,25)])[,5])
         colnames(df_HD_plot) <- c("BaseLine", "Shock4")
         ggplot(df_HD_plot)+
           geom_line(aes(x = 1:nrow(df_HD_plot), y = Shock4), col = 'red', linetype = "dashed"
           geom_line(aes(x = 1:nrow(df_HD_plot), y = BaseLine), col = 'royalblue')
```

```
New names:
* NA -> ...1
* NA -> ...2
```



4.2.6 Table

- 1992Q1

  1991Q1-2020Q3, diff(x, 4)1992Q1

```
In [64]: t_label <- c()
         year_label <- c()
         season_label <- c()
         year <- 1992
         for(q in 1:115){
             if(q%%4==0){
                 t_label <- c(t_label, paste0(year, "Q4", sep = ""))
                 year_label <- c(year_label, year)
                 season_label <- c(season_label, 4)
                 year <- year+1
             }else{
                 t_label <- c(t_label, paste0(year, "Q", q%%4, sep = ""))
                 year_label <- c(year_label, year)
                 season_label <- c(season_label, q%%4)
             }
         }
```

```
In [65]: df_HD <- bind_cols(t_label,
                            year_label,
                            season_label,
                            (By-SVAR_AB_Hist.c0)[,5],
                            SVAR_AB_HistDecomp[,21],
                            SVAR_AB_HistDecomp[,22],
                            SVAR_AB_HistDecomp[,23],
                            SVAR_AB_HistDecomp[,24],
                            SVAR_AB_HistDecomp[,25])
         colnames(df_HD) <- c("Time", "Year", "Season", "BaseLine", "mp", "exp", "hs", "hd", "s
         tail(df_HD)
```

```
New names:
* NA -> ...1
* NA -> ...2
* NA -> ...3
* NA -> ...4
* NA -> ...5
* ...
```

A tibble: 6 Œ 9

Time <chr>	Year <dbl>	Season <dbl>	BaseLine <dbl>	mp <dbl>	exp <dbl>	hs <dbl>	hd <dbl>
2019Q2	2019	2	-4.793248	0.6144680	-0.2409125	-0.59356579	-0.207934811
2019Q3	2019	3	-4.639120	0.2197710	0.7632198	-0.01844522	0.150153777
2019Q4	2019	4	-4.722549	0.4810390	-0.6313722	-0.44429988	0.831912276
2020Q1	2020	1	-5.281133	0.6558706	-3.0880884	-0.11607816	0.820332857
2020Q2	2020	2	-5.200768	0.7851396	-2.1252739	-0.44044565	0.350250199
2020Q3	2020	3	-2.025775	0.4209959	-0.3590478	-0.41418181	-0.006178081

```
In [66]: df_HD.table <- df_HD %>% summarise(Time = Time,
                             Year = Year,
                             Season = Season,
                             mp = mp/BaseLine*100,
                             exp = exp/BaseLine*100,
                             hs = hs/BaseLine*100,
                             hd = hd/BaseLine*100,
                             sp = sp/BaseLine*100) %>%
             drop_na()
```

```
In [67]: df_HD.table
```

A tibble: 108 Œ 8

Time	Year	Season	mp	exp	hs	hd	sp
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<d
1993Q4	1993	4	3.308656	-177.015005	18.948542	-49.025231	303
1994Q1	1994	1	10.943992	106.077987	-43.682641	65.892361	-39
1994Q2	1994	2	28.292750	66.293043	-33.868337	68.485734	-29
1994Q3	1994	3	258.589226	-149.933735	-223.522112	430.625509	-21
1994Q4	1994	4	-176.957447	200.233859	43.749444	-8.881384	41.
1995Q1	1995	1	-80.912314	159.661908	1.241708	24.686807	-4.
1995Q2	1995	2	-78.199927	186.107959	-15.435993	32.107922	-24
1995Q3	1995	3	-34.473569	135.291372	-38.982330	66.297651	-28
1995Q4	1995	4	-13.302156	70.037030	-63.027957	94.595245	11.
1996Q1	1996	1	10.425865	49.888299	-156.355184	163.103038	32.
1996Q2	1996	2	-13.198739	-42.375507	168.239184	-110.518857	97.
1996Q3	1996	3	16.570934	32.127117	37.955804	2.949832	10.
1996Q4	1996	4	37.957698	2.562573	16.463435	16.233979	26.
1997Q1	1997	1	51.304861	2.331296	6.848888	33.658666	5.8
1997Q2	1997	2	45.355589	26.139468	-10.680763	56.095143	-16
1997Q3	1997	3	-4.629960	84.218353	-39.104054	97.220119	-37
1997Q4	1997	4	-66.561229	132.307031	-44.818538	100.536053	-21
1998Q1	1998	1	-68.816391	118.549793	-30.955608	53.407466	27.
1998Q2	1998	2	-1087.271422	1899.664941	-278.455274	228.002615	-66
1998Q3	1998	3	27.168670	-77.746689	8.802550	50.194087	91.
1998Q4	1998	4	110.204167	-23.126964	-19.508284	108.570063	-76
1999Q1	1999	1	22.538639	40.221345	-12.532716	44.052489	5.7
1999Q2	1999	2	45.494679	39.829490	-19.310490	34.914211	-0.
1999Q3	1999	3	86.003675	31.093022	-23.645249	63.463143	-56
1999Q4	1999	4	25.820596	2.338328	12.318965	41.282907	18.
2000Q1	2000	1	172.927378	-319.085795	65.417039	213.478367	-32
2000Q2	2000	2	132.142728	-96.523816	15.302039	62.694967	-13
2000Q3	2000	3	50.799964	6.631011	14.290587	2.616765	25.
2000Q4	2000	4	47.754868	49.520153	7.912859	-20.320864	15.
2001Q1	2001	1	72.806231	61.486742	28.136863	-27.586518	-34
2013Q2	2013	2	-16.195693	66.187484	16.3446129	12.9118722	20.
2013Q3	2013	3	-15.585395	48.936647	19.3329080	5.3188377	41.
2013Q4	2013	4	-6.509104	36.649776	21.5024671	-5.7021060	54.
2014Q1	2014	1	-17.708796	8.267309	26.2182385	1.5845659	81.
2014Q2	2014	2	131.447250	431.536626	-316.0129022	-105.8915237	-41
2014Q3	2014	3	-11.715246	174.486425	-48.1332478	-16.7439692	2.1
2014Q4	2014	4	-20.657349	120.546559	-11.1235084	-14.2998391	25.
2015Q1	2015	1	-22.490489	99.187742	-2.5981037	-7.9204041	33.
2015Q2	2015	2	-13.619591	55.714693	4.8540277	-2.4949080	55.
2015Q3	2015	3	-1.527735	35.838478	9.1194084	-4.2271128	60.
2015Q4	2015	4	11.074638	19.770152	8.4880489	-3.4383299	64.
2016Q1	2016	1	16.418873	17.358056	14.9112220	0.1598825	51.
2016Q2	2016	2	26.160543	29.514701	21.9575878	0.7866500	21.
2016Q3	2016	3	30.570143	42.589063	22.9586910	3.4713393	0.4
2016Q4	2016	4	39.503143	59.131676	23.6338777	2.4039517	-24
2017Q1	2017	1	69.626389	62.380427	16.5993827	-2.3362992	-46
2017Q2	2017	2	68.714609	52.016828	13.8245322	1.1094348	-35
2017Q3	2017	3	60.725214	36.000347	6.0186359	8.0462370	-10
2017Q4	2017	4	59.384311	38.357273	4.3421579	8.5995624	-10
2018Q1	2018	1	30.460253	36.241255	12.7157146	13.2899035	7.2

```
In [68]: get_HD.table <- function(df_HD.table,
                                  year_start, season_start,
                                  year_end, season_end){
             HD_seq.temp <- df_HD.table %>%
                 filter( Year >= year_start & Year <= year_end) %>%
                 filter( !(Year==year_start & Season < season_start) ) %>%
                 filter( !(Year==year_end & Season > season_end) ) %>%
                 summarise(mp = median(mp),
                   exp = median(exp),
                   hs = median(hs),
                   hd = median(hd),
                   sp = median(sp))
             return(HD_seq.temp)
         }
```

```
In [69]: # all samples
         # 1991Q1-2020Q3
         HD_seq1 <- get_HD.table(df_HD.table, 1991, 1, 2020, 3)

         # subsample 1: 1991Q1-2003Q3 ()
         HD_seq2 <- get_HD.table(df_HD.table, 1991, 1, 2003, 3)

         # subsample 2: 2003Q3-2006Q4 ()
         HD_seq3 <- get_HD.table(df_HD.table, 2003, 3, 2006, 4)

         # subsample 3: 2006Q4-2009Q2 ()
         HD_seq4 <- get_HD.table(df_HD.table, 2006, 4, 2009, 2)

         # subsample 4: 2009Q2-2013Q3 ()
         HD_seq5 <- get_HD.table(df_HD.table, 2009, 2, 2013, 3)

         # subsample 5: 2013Q3-2016Q2 ()
         HD_seq6 <- get_HD.table(df_HD.table, 2013, 3, 2016, 2)

         # subsample 6: 2016Q2-2020Q3 ()
         HD_seq7 <- get_HD.table(df_HD.table, 2016, 2, 2020, 3)
```

```
In [70]: HD_seq <- bind_rows(HD_seq1, HD_seq2, HD_seq3, HD_seq4, HD_seq5, HD_seq6, HD_seq7)
         HD.table <- bind_cols(c("All Samples (1993Q4-2020Q3)",
                                 "Subsample 1 (1993Q4-2003Q3)",
                                 "Subsample 2 (2003Q3-2006Q4)",
                                 "Subsample 3 (2006Q4-2009Q2)",
                                 "Subsample 4 (2009Q2-2013Q3)",
                                 "Subsample 5 (2013Q3-2016Q2)",
                                 "Subsample 6 (2016Q2-2020Q3)"), HD_seq)
         colnames(HD.table) <- c("",
                                 "Montary Policy Shock",
                                 "Expectation Shock",
                                 "Demand Shock",
                                 "Supply Shock",
                                 "Residual Shock")
         HD.table
```

```
New names:
* NA -> ...1
```

A tibble: 7 Œ 6

	Montary Policy Shock	Expectation Shock	Demand Shoc
<chr>	<dbl>	<dbl>	<dbl>
All Samples (1993Q4-2020Q3)	16.61415521	31.881494	14.600905
Subsample 1 (1993Q4-2003Q3)	33.12522396	35.978303	7.380873
Subsample 2 (2003Q3-2006Q4)	28.35634146	-7.440279	46.801099
Subsample 3 (2006Q4-2009Q2)	12.76642861	44.469220	45.033616
Subsample 4 (2009Q2-2013Q3)	14.04880363	28.944058	27.717423
Subsample 5 (2013Q3-2016Q2)	-9.11217539	42.793212	8.803729
Subsample 6 (2016Q2-2020Q3)	-0.02322655	37.299264	9.762709

```
In [71]: # table
         tab_HD <- xtable(HD.table, caption= "(%)", align=c("c","c","c","c","c","c", "c"))
         print(tab_HD, include.rownames=FALSE)
```

```
% latex table generated in R 3.6.2 by xtable 1.8-4 package
% Fri Oct 15 00:34:13 2021
\begin{table}[ht]
\centering
\begin{tabular}{cccccc}
  \hline
 & Montary Policy Shock & Expectation Shock & Demand Shock & Supply Shock & Residual Shock \\
  \hline
All Samples (1993Q4-2020Q3) & 16.61 & 31.88 & 14.60 & 6.64 & 19.29 \\
  Subsample 1 (1993Q4-2003Q3) & 33.13 & 35.98 & 7.38 & 42.67 & -9.15 \\
  Subsample 2 (2003Q3-2006Q4) & 28.36 & -7.44 & 46.80 & -1.13 & 20.66 \\
  Subsample 3 (2006Q4-2009Q2) & 12.77 & 44.47 & 45.03 & 7.96 & 17.60 \\
  Subsample 4 (2009Q2-2013Q3) & 14.05 & 28.94 & 27.72 & 7.67 & 26.36 \\
  Subsample 5 (2013Q3-2016Q2) & -9.11 & 42.79 & 8.80 & -3.83 & 46.57 \\
  Subsample 6 (2016Q2-2020Q3) & -0.02 & 37.30 & 9.76 & 2.94 & 42.91 \\
   \hline
\end{tabular}
\caption{(%)}
\end{table}
```

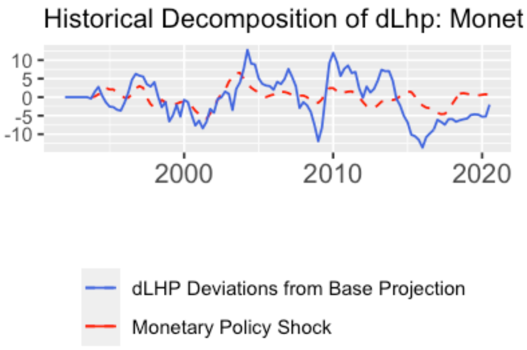
```
In [72]: print(tab_HD, include.rownames=FALSE,
               file="result/table/HD_0219_m1.tex",
               append=T, table.placement = "h",
               caption.placement="bottom", hline.after=seq(from=-1,to=nrow(tab_HD),by=1))
```

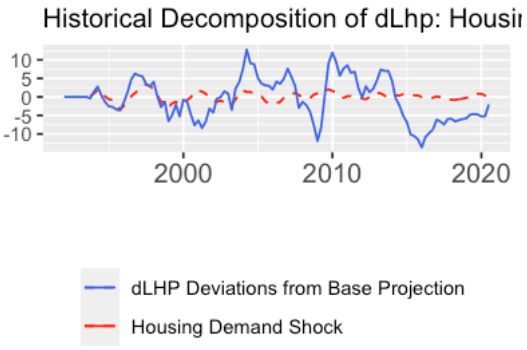
4.2.7 Save Plot

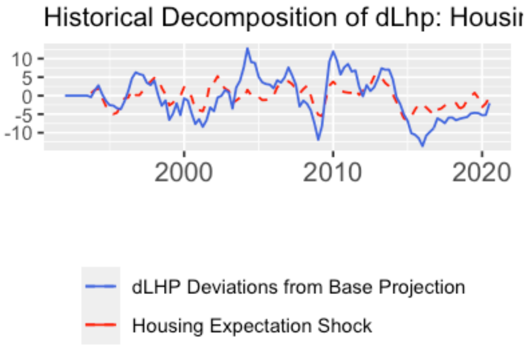
```
In [73]: xlab <- lubridate::yq(df_HD$Time)
```

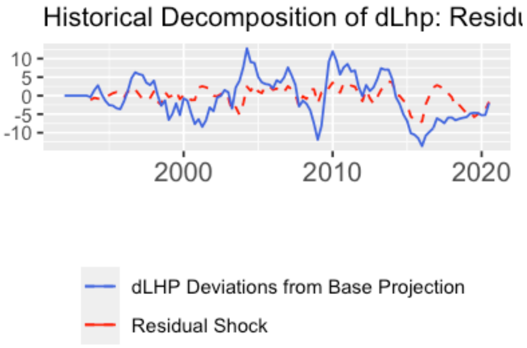
```
In [74]: # plot 1
         figure_HD.1 <- df_HD %>%
             ggplot()+
             geom_line(aes(x = xlab, y = mp, color = "Monetary Policy Shock"), linetype = "dash
             geom_line(aes(x = xlab, y = BaseLine, color = "dLHP Deviations from Base Projecti
             labs(x = '',
                  y = '',
                  title = 'Historical Decomposition of dLhp: Monetary Policy Shock')+
             Text_Size_Theme+
             scale_color_manual(values=c('royalblue','red'))+
             theme(legend.position="bottom",
                   legend.direction="vertical",
                   legend.title = element_blank())

         # plot 2
         figure_HD.2 <- df_HD %>%
             ggplot()+
             geom_line(aes(x = xlab, y = exp, color = "Housing Expectation Shock"), linetype =
             geom_line(aes(x = xlab, y = BaseLine, color = "dLHP Deviations from Base Projecti
             labs(x = '',
                  y = '',
                  title = 'Historical Decomposition of dLhp: Housing Expectation Shock')+
             Text_Size_Theme+
             scale_color_manual(values=c('royalblue','red'))+
             theme(legend.position="bottom",
                   legend.direction="vertical",
                   legend.title = element_blank())

         # plot 3
         figure_HD.3 <- df_HD %>%
             ggplot()+
             geom_line(aes(x = xlab, y = hs, color = "Housing Supply Shock"), linetype = "dashe
             geom_line(aes(x = xlab, y = BaseLine, color = "dLHP Deviations from Base Projecti
             labs(x = '',
                  y = '',
                  title = 'Historical Decomposition of dLhp: Housing Supply Shock')+
             Text_Size_Theme+
             scale_color_manual(values=c('royalblue','red'))+
             theme(legend.position="bottom",
                   legend.direction="vertical",
                   legend.title = element_blank())

         # plot 4
         figure_HD.4 <- df_HD %>%
             ggplot()+
             geom_line(aes(x = xlab, y = hd, color = "Housing Demand Shock"), linetype = "dashe
             geom_line(aes(x = xlab, y = BaseLine, color = "dLHP Deviations from Base Projectio
             labs(x = '',
                  y = '',
                  title = 'Historical Decomposition of dLhp: Housing Demand Shock')+
             Text_Size_Theme+
             scale_color_manual(values=c('royalblue','red'))+
             theme(legend.position="bottom",
                   legend.direction="vertical",
                   legend.title = element_blank())



         # plot 5
         figure_HD.5 <- df_HD %>%
             ggplot()+
             geom_line(aes(x = xlab, y = sp, color = "Residual Shock"), linetype = "dashed")+
             geom_line(aes(x = xlab, y = BaseLine, color = "dLHP Deviations from Base Projectio
             labs(x = '',
                  y = '',
                  title = 'Historical Decomposition of dLhp: Residual Shock')+
             Text_Size_Theme+
             scale_color_manual(values=c('royalblue','red'))+
             theme(legend.position="bottom",
                   legend.direction="vertical",
                   legend.title = element_blank())

In [75]: # save shock 1
         ggsave(filename = "result/figure/0219_m1_HD_shock1.png",
                plot = figure_HD.1,
                width = 15, height = 10, units = "cm",
                device = "png")

         # save shock 2
         ggsave(filename = "result/figure/0219_m1_HD_shock2.png",
                plot = figure_HD.2,
                width = 15, height = 10, units = "cm",
                device = "png")

         # save shock 3
         ggsave(filename = "result/figure/0219_m1_HD_shock3.png",
                plot = figure_HD.3,
                width = 15, height = 10, units = "cm",
                device = "png")

         # save shock 4
         ggsave(filename = "result/figure/0219_m1_HD_shock4.png",
                plot = figure_HD.4,
                width = 15, height = 10, units = "cm",
                device = "png")
         # save shock 5
         ggsave(filename = "result/figure/0219_m1_HD_shock5.png",
                plot = figure_HD.5,
                width = 15, height = 10, units = "cm",
                device = "png")
```

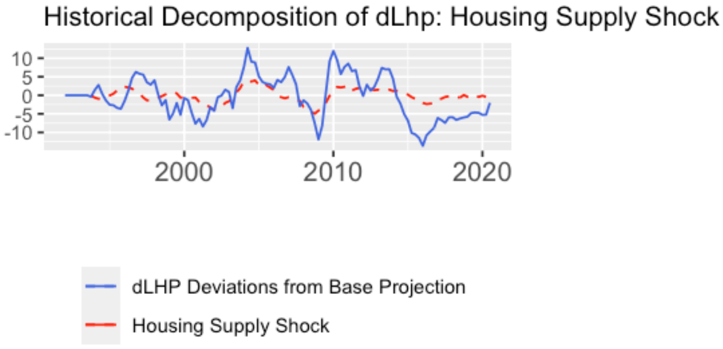
```
In [76]: # For hp
         multiplot(figure_HD.1,figure_HD.2,figure_HD.3,figure_HD.4,figure_HD.5,
                   cols = 2)
         ggsave(filename = "result/figure/HD.png",
                plot = multiplot(figure_HD.1,
                                 figure_HD.2,
                                 figure_HD.3,
                                 figure_HD.4,
                                 figure_HD.5,
                                 cols = 2),
                width = 15*2, height = 10*2, units = "cm",
                device = "png")
```

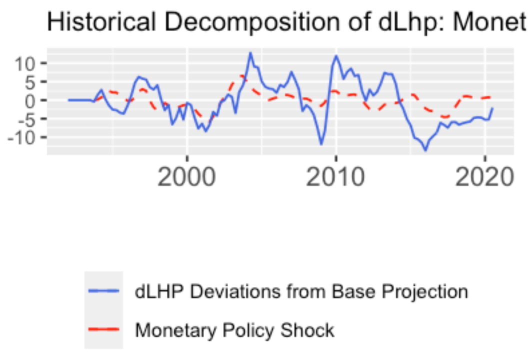

Historical Decomposition of dLhp: Monet
10
5
0
-5
-10
2000
2010
2020
dLHP Deviations from Base Projection
Monetary Policy Shock

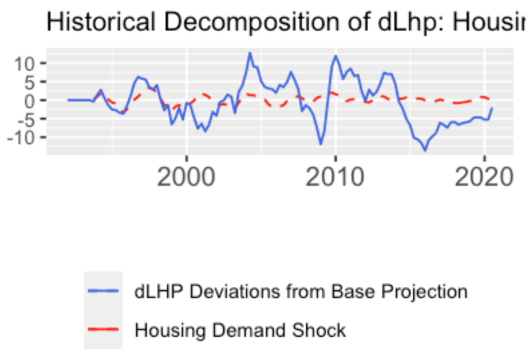

Historical Decomposition of dLhp: Housi
10
5
0
-5
-10
2000
2010
2020
dLHP Deviations from Base Projection
Housing Demand Shock

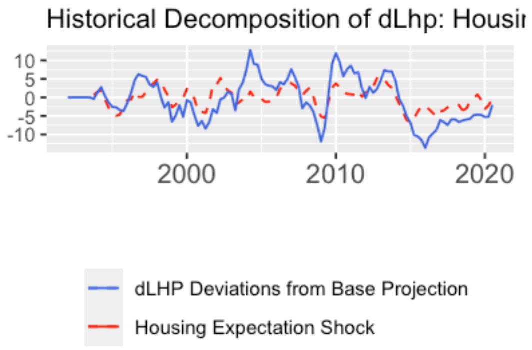

Historical Decomposition of dLhp: Housi
10
5
0
-5
-10
2000
2010
2020
dLHP Deviations from Base Projection
Housing Expectation Shock

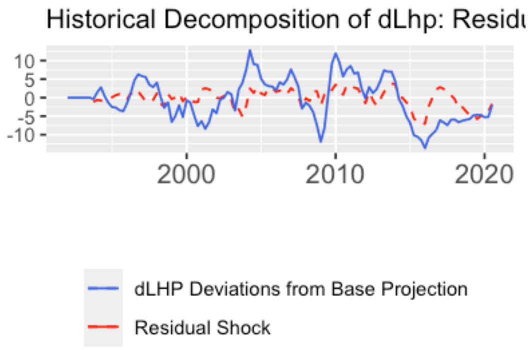

Historical Decomposition of dLhp: Resid
10
5
0
-5
-10
2000
2010
2020
dLHP Deviations from Base Projection
Residual Shock

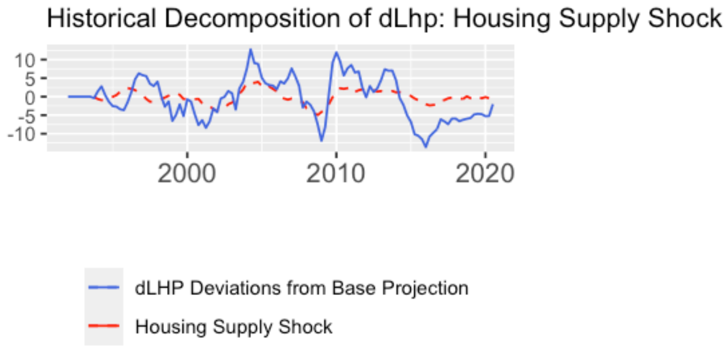

Historical Decomposition of dLhp: Housing Supply Shock
10
5
0
-5
-10
2000
2010
2020
dLHP Deviations from Base Projection
Housing Supply Shock


Historical Decomposition of dLhp: Monet

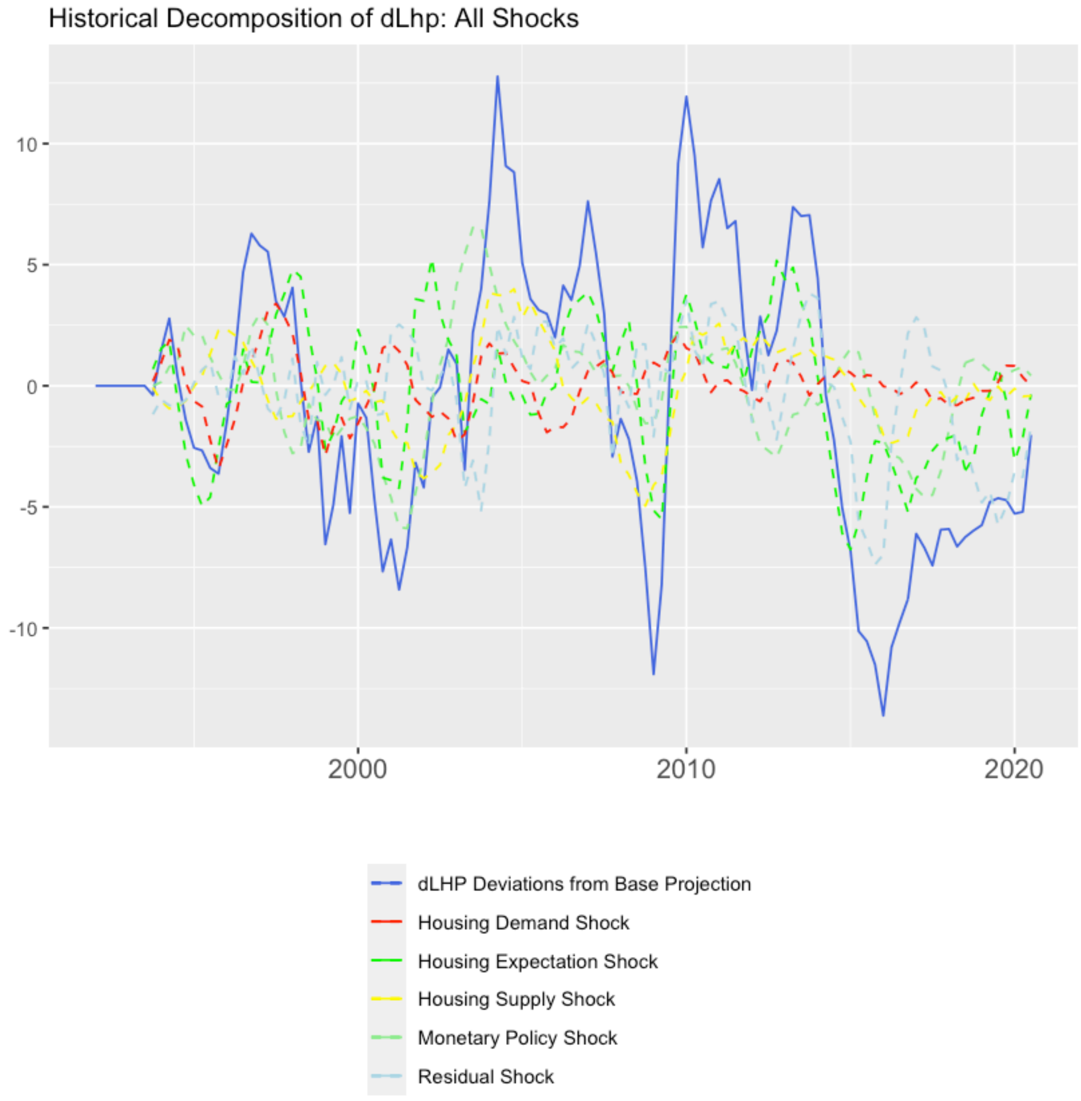

Historical Decomposition of dLhp: Housi


Historical Decomposition of dLhp: Housi

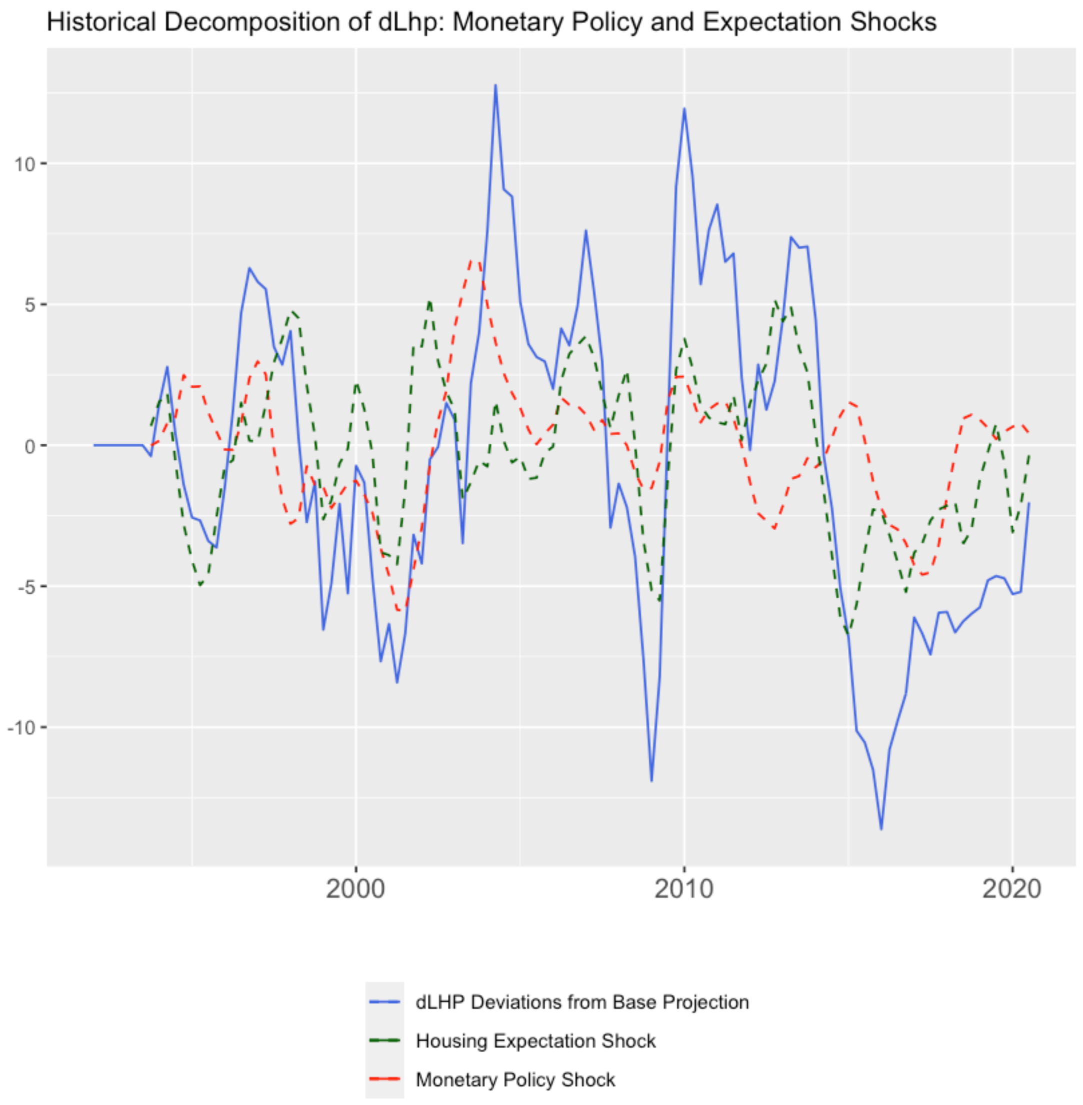

Historical Decomposition of dLhp: Resid


Historical Decomposition of dLhp: Housing Supply Shock

```
In [77]: # ALL
         figure_HD.6 <- df_HD %>%
             ggplot()+
             geom_line(aes(x = xlab, y = BaseLine, color = "dLHP Deviations from Base Projecti
             geom_line(aes(x = xlab, y = mp, color = "Monetary Policy Shock"), linetype = "das
             geom_line(aes(x = xlab, y = exp, color = "Housing Expectation Shock"), linetype =
             geom_line(aes(x = xlab, y = hs, color = "Housing Supply Shock"), linetype = "dashe
             geom_line(aes(x = xlab, y = hd, color = "Housing Demand Shock"), linetype = "dashe
             geom_line(aes(x = xlab, y = sp, color = "Residual Shock"), linetype = "dashed")+
             labs(x = '',
                  y = '',
                  title = 'Historical Decomposition of dLhp: All Shocks')+
             Text_Size_Theme+
             scale_color_manual(values=c('royalblue','red', 'green', 'yellow', 'lightgreen', 'l
             theme(legend.position="bottom",
                   legend.direction="vertical",
                   legend.title = element_blank())
         # save shock 6
         ggsave(filename = "result/figure/0219_m1_HD_shock6.png",
                plot = figure_HD.6,
                width = 15, height = 10, units = "cm",
                device = "png")
         figure_HD.6
```



```
In [78]: # ALL
         figure_HD.6 <- df_HD %>%
             ggplot()+
             geom_line(aes(x = xlab, y = BaseLine, color = "dLHP Deviations from Base Projecti
             geom_line(aes(x = xlab, y = mp, color = "Monetary Policy Shock"), linetype = "das
             geom_line(aes(x = xlab, y = exp, color = "Housing Expectation Shock"), linetype =
         #     geom_line(aes(x = xlab, y = sp, color = "Residual Shock"), linetype = "dashed")
             labs(x = '',
                  y = '',
                  title = 'Historical Decomposition of dLhp: Monetary Policy and Expectation S
             Text_Size_Theme+
             scale_color_manual(values=c('royalblue', 'darkgreen', 'red'))+
             theme(legend.position="bottom",
                   legend.direction="vertical",
                   legend.title = element_blank())
         # save shock 6
         ggsave(filename = "result/figure/0219_m1_HD_shock6.png",
                plot = figure_HD.6,
                width = 20, height = 15, units = "cm",
                device = "png")
         figure_HD.6
```



```
In [79]: getwd()
```

```
'/Users/Andy 1/google_drive/0_Preserved/Thesis/7_writing/model_playground/model/0_benchmark'
```