Take-Home Exam Spring 2020

DUE DATE: Friday 11^{th}, June 2021 Before mid-night.

I pledge to the Honor Code and to obey all rules for taking and performing Take-home exam as specified by the course instructor.

Your Name: Tanapat I.

Your Student ID: 6104640914

There are Three questions.

Instruction:

- You have to submit (1) The R codes (YourID_Exam.R), (2) Your report with the empirical analysis.

- If you have any question related to this exam, please send me the e-mail at Wasin@econ.tu.ac.th with the Topic: EE435: Take-home exam questions.

Question1. (20 Marks)

Consider four components of U.S. monthly industrial production index from January 1947 -April 2021. The four components are durable consumer goods (IPD- CONGD), nondurable consumer goods (IPNCONGD), business equivalent (IPBUSEQ), and materials (IPMAT). The original data are available from the Federal Reserve Bank of St. Louis and are seasonally adjusted. Note that IPMAT starts at January 1939. You can download these data set by using the below codes:

Toolbox 1

```
getSymbols("IPDCONGD",src="FRED")
dim(IPDCONGD)
tail(IPDCONGD)
getSymbols("IPNCONGD",src="FRED")
dim(IPNCONGD)
getSymbols("IPBUSEQ",src="FRED")
dim(IPBUSEQ)
getSymbols("IPMAT",src="FRED")
```

```
dim(IPMAT)
IP = cbind(as.numeric(IPDCONGD),as.numeric(IPNCONGD),as.numeric(IPBUSEQ),as.numeric(IPMAT[-c
    (1:96)]))
dim(IP)
colnames(IP) <- c("IPD","IPN","IPB","IPM")
```

1.1 Construct the growth rate series $\mathbf{z}_t$ of the four industrial production index, i.e. take the first difference of the log data. Obtain time plots of $\mathbf{z}_t$. Comment on the time plot.

```
#1.1
getSymbols("IPDCONGD",src="FRED")
getSymbols("IPNCONGD",src="FRED")
```

```
## [1] "IPNCONGD"
```

```
getSymbols("IPBUSEQ",src="FRED")
```

```
## [1] "IPBUSEQ"
```

```
getSymbols("IPMAT",src="FRED")
```

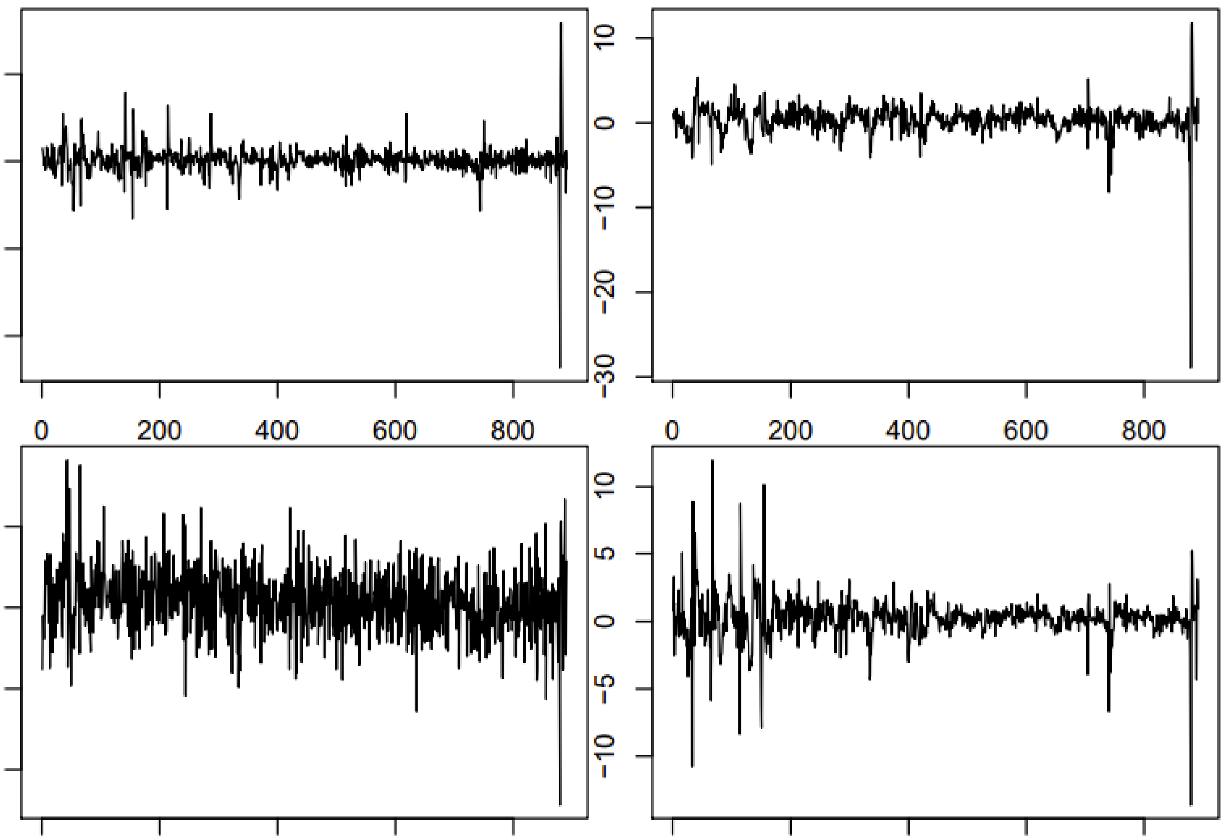
```
## [1] "IPMAT"
```

```
IP = cbind(as.numeric(IPDCONGD),as.numeric(IPNCONGD),as.numeric(IPBUSEQ),as.numeric(IPMAT[-c(1:96)]))
dim(IP)
```

```
## [1] 892   4
```

```
colnames(IP) <- c("IPD","IPN","IPB","IPM")
logIP=log(IP)
Zt=diff(logIP)*100
Zt
```

Zt represent the growth rate of all four series by first, binding all the four data into one. Then take log function to convert data into log series, and lastly take the difference of the log series and multiply by 100 to convert into growth series of all four industries.



The multiple time series show no sign of serial correlation between series and time frame. And the series seem to construct stationarity properties of constant mean and volatility.

1.2 Build a VAR model for $\mathbf{z}_t$, including simplification (or refine) the model by removing insignificant parameters with threshold of t-ratio 1.645 and perform model checking. Write down the final fitted model.

```
#1.2
VARorder(Zt) #with lowest possible bic criteria
```

```
## selected order: aic =  6
## selected order: bic =  2
## selected order: hq =  3
## Summary table:
##          p     AIC    BIC     HQ     M(p) p-value
##  [1,]    0 2.8153 2.8153 2.8153   0.0000  0.0000
##  [2,]    1 2.6470 2.7330 2.6798 178.2331  0.0000
##  [3,]    2 2.5125 2.6846 2.5783 147.9623  0.0000
##  [4,]    3 2.4570 2.7151 2.5556  79.0713  0.0000
##  [5,]    4 2.4563 2.8005 2.5879  31.4646  0.0117
##  [6,]    5 2.4045 2.8348 2.5689  75.1412  0.0000
##  [7,]    6 2.3832 2.8995 2.5805  48.7671  0.0000
##  [8,]    7 2.3984 3.0008 2.6287  17.5535  0.3507
##  [9,]    8 2.4168 3.1053 2.6799  14.8136  0.5383
## [10,]    9 2.4224 3.1970 2.7184  25.4492  0.0623
## [11,]   10 2.4268 3.2874 2.7557  26.3997  0.0487
## [12,]   11 2.4327 3.3793 2.7945  24.9845  0.0701
## [13,]   12 2.4258 3.4585 2.8205  35.4558  0.0034
## [14,]   13 2.4266 3.5454 2.8542  28.9289  0.0244
```

```
m.MTS=VAR(Zt,2)
```

The VARorder function suggest the lag order 2, as the lowest possible BIC value, constructing the vector auto-regressive model using the Zt data with lag order 2.

```
## Estimates:  0.1724828 0.1811122 0.1757223 0.1275675
## Std.Error:  0.1133758 0.02803105 0.05863395 0.05433982
## AR coefficient matrix
## AR( 1 )-matrix
##         [,1]    [,2]     [,3]   [,4]
## [1,] 0.11876  0.2193 -0.10721 0.3384
## [2,] 0.02955 -0.1678  0.00916 0.0162
## [3,] 0.00339  0.0539  0.12272 0.1800
## [4,] 0.01713  0.1435  0.02771 0.2112
## standard error
##        [,1]   [,2]   [,3]   [,4]
## [1,] 0.0473 0.1467 0.0928 0.0821
## [2,] 0.0117 0.0363 0.0230 0.0203
## [3,] 0.0245 0.0759 0.0480 0.0425
## [4,] 0.0227 0.0703 0.0445 0.0394
## AR( 2 )-matrix
##           [,1]    [,2]     [,3]     [,4]
## [1,] -0.157177  0.1887 -0.19773  0.10530
## [2,]  0.000419 -0.0449 -0.00132  0.00082
## [3,] -0.142158  0.0250  0.18277  0.10314
## [4,]  0.010632  0.1338  0.02122 -0.06006
## standard error
##        [,1]   [,2]   [,3]   [,4]
## [1,] 0.0471 0.1461 0.0924 0.0825
## [2,] 0.0116 0.0361 0.0229 0.0204
## [3,] 0.0243 0.0756 0.0478 0.0427
## [4,] 0.0226 0.0700 0.0443 0.0395
##
## Residuals cov-mtx:
##            [,1]      [,2]      [,3]      [,4]
## [1,] 10.0294942 0.8338041 3.6930427 2.3401480
## [2,]  0.8338041 0.6130805 0.4406453 0.3140822
## [3,]  3.6930427 0.4406453 2.6824829 1.2567987
## [4,]  2.3401480 0.3140822 1.2567987 2.3039604
##
## det(SSE) =  11.45125
## AIC =  2.509928
## BIC =  2.682044
## HQ  =  2.575709
```

$$\begin{bmatrix} IPD \\ IPN \\ IPB \\ IPM \end{bmatrix} = \begin{bmatrix} 0.17248 \\ 0.18111 \\ 0.17572 \\ 0.12757 \end{bmatrix} + \begin{bmatrix} 0.11876 & 0.2193 & -0.10721 & 0.3384 \\ 0.02955 & -0.1678 & 0.00916 & 0.0162 \\ 0.00339 & 0.0539 & 0.1272 & 0.1800 \\ 0.01713 & 0.1435 & 0.02771 & 0.2112 \end{bmatrix} \begin{bmatrix} IPD_{t-1} \\ IPN_{t-1} \\ IPB_{t-1} \\ IPM_{t-1} \end{bmatrix}$$

$$+ \begin{bmatrix} -0.15718 & 0.1887 & -0.49773 & 0.10530 \\ 0.00042 & -0.0449 & -0.00132 & 0.00082 \\ -0.14216 & 0.0250 & 0.18277 & 0.10314 \\ 0.01063 & 0.1338 & 0.02122 & -0.06006 \end{bmatrix} \begin{bmatrix} IPD_{t-2} \\ IPN_{t-2} \\ IPB_{t-2} \\ IPM_{t-2} \end{bmatrix}$$

VAR model

```r
m2.MTS=refVAR(m.MTS, thres=1.645)
```

```
## Constant term:
## Estimates:  0.2152919 0.1773043 0.1842434 0.124001
## Std.Error:  0.109649 0.02686661 0.05687657 0.05346031
## AR coefficient matrix
## AR( 1 )-matrix
##        [,1]   [,2]  [,3]  [,4]
## [1,] 0.1023  0.000 0.000 0.347
## [2,] 0.0348 -0.155 0.000 0.000
## [3,] 0.0000  0.000 0.134 0.185
## [4,] 0.0000  0.178 0.000 0.232
## standard error
##         [,1]   [,2]   [,3]   [,4]
## [1,] 0.03733 0.0000 0.0000 0.0783
## [2,] 0.00844 0.0351 0.0000 0.0000
## [3,] 0.00000 0.0000 0.0378 0.0413
## [4,] 0.00000 0.0672 0.0000 0.0337
## AR( 2 )-matrix
##        [,1]  [,2]   [,3]  [,4]
## [1,] -0.129 0.000 -0.175 0.000
## [2,]  0.000 0.000  0.000 0.000
## [3,] -0.139 0.000  0.182 0.102
## [4,]  0.000 0.144  0.000 0.000
## standard error
##        [,1]   [,2]   [,3]   [,4]
## [1,] 0.0454 0.0000 0.0860 0.0000
## [2,] 0.0000 0.0000 0.0000 0.0000
## [3,] 0.0238 0.0000 0.0464 0.0425
## [4,] 0.0000 0.0654 0.0000 0.0000
##
## Residuals cov-mtx:
##           [,1]     [,2]     [,3]     [,4]
## [1,] 10.096231 0.827661 3.701401 2.322786
## [2,]  0.827661 0.614838 0.439733 0.313845
## [3,]  3.701401 0.439733 2.684327 1.256237
## [4,]  2.322786 0.313845 1.256237 2.315673
##
## det(SSE) =  11.75217
## AIC =  2.495463
## BIC =  2.570764
## HQ  =  2.524243
```

Refine VAR model

$$\begin{bmatrix} IPD \\ IPN \\ IPB \\ IPM \end{bmatrix} = \begin{bmatrix} 0.21529 \\ 0.17730 \\ 0.18424 \\ 0.12400 \end{bmatrix} + \begin{bmatrix} 0.1023 & 0 & 0 & 0.347 \\ 0.0348 & -0.155 & 0 & 0 \\ 0 & 0 & 0.134 & 0.185 \\ 0 & 0.178 & 0 & 0.232 \end{bmatrix} \begin{bmatrix} IPD_{t-1} \\ IPN_{t-1} \\ IPB_{t-1} \\ IPM_{t-1} \end{bmatrix} + \begin{bmatrix} -0.129 & 0 & -0.175 & 0 \\ 0 & 0 & 0 & 0 \\ -0.139 & 0 & 0.182 & 0.102 \\ 0 & 0.144 & 0 & 0 \end{bmatrix} \begin{bmatrix} IPD_{t-2} \\ IPN_{t-2} \\ IPB_{t-2} \\ IPM_{t-2} \end{bmatrix}$$

The residuals cov-matrix are equal between VAR model and Refine-VAR model.

```
## Ljung-Box Statistics:
##          m       Q(m)     df    p-value
##  [1,]   1.00     5.18   16.00     0.99
##  [2,]   2.00    12.10   32.00     1.00
##  [3,]   3.00    70.97   48.00     0.02
##  [4,]   4.00    99.82   64.00     0.00
##  [5,]   5.00   151.76   80.00     0.00
##  [6,]   6.00   196.00   96.00     0.00
##  [7,]   7.00   223.41  112.00     0.00
##  [8,]   8.00   249.96  128.00     0.00
##  [9,]   9.00   274.49  144.00     0.00
## [10,]  10.00   303.57  160.00     0.00
## [11,]  11.00   327.36  176.00     0.00
## [12,]  12.00   363.26  192.00     0.00
## [13,]  13.00   384.39  208.00     0.00
## [14,]  14.00   410.56  224.00     0.00
## [15,]  15.00   426.89  240.00     0.00
## [16,]  16.00   455.62  256.00     0.00
## [17,]  17.00   467.16  272.00     0.00
## [18,]  18.00   503.24  288.00     0.00
## [19,]  19.00   513.87  304.00     0.00
## [20,]  20.00   519.35  320.00     0.00
## [21,]  21.00   535.30  336.00     0.00
## [22,]  22.00   563.89  352.00     0.00
```

With lag 2's p-value, the calculated p-value is greater than 0.05 at 95% confidence interval, we cannot reject null hypothesis at 95% CI. Thus, the model does not contain serial correlation problem, the model is adequate and appropriate to use.

1.3 Obtain the plots of impulse response function of the fitted model. Interpret the plots.

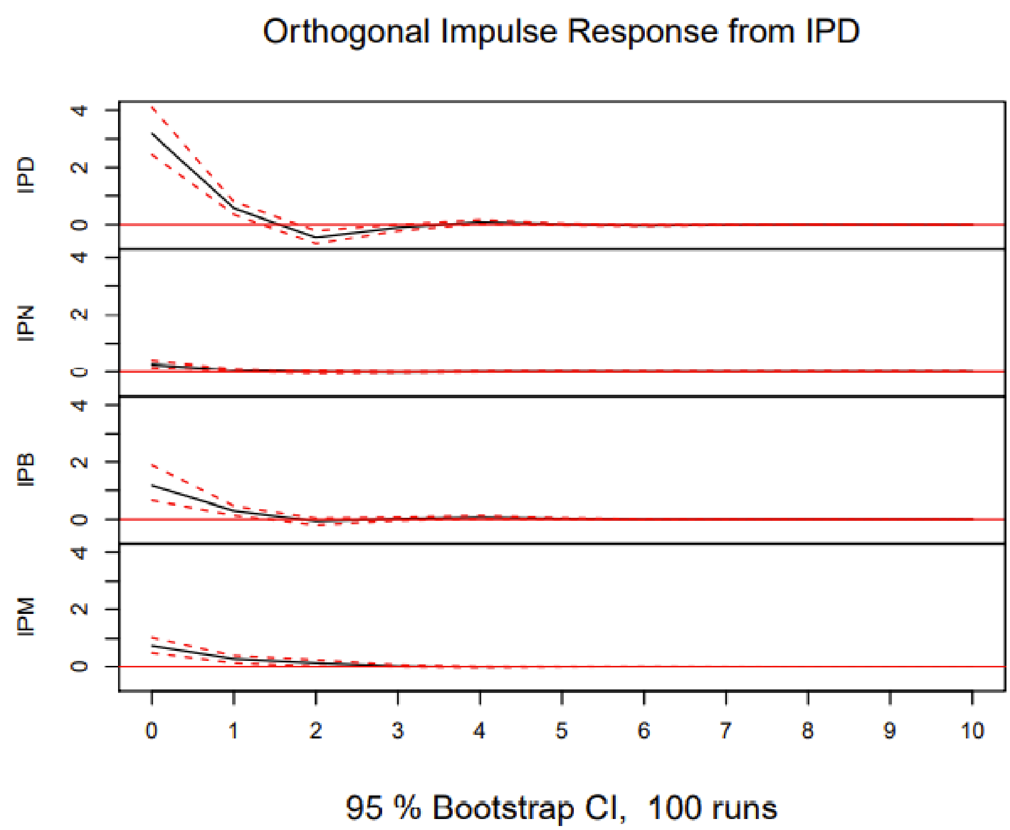
```
detach(package:MTS, unload = TRUE)
require(vars)
```

```
VARselect(Zt, lag.max = 10)
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n) 
##      6      3      2      6 
## 
## $criteria
##                1         2         3         4         5         6         7
## AIC(n)  2.649961  2.516158  2.460867  2.460460  2.408936  2.388695  2.404697
## HQ(n)   2.691465  2.590866  2.568778  2.601575  2.583254  2.596217  2.645422
## SC(n)   2.758498  2.711525  2.743063  2.829486  2.864792  2.931380  3.034212
## FPE(n) 14.153494 12.380973 11.715059 11.710427 11.122525 10.899925 11.076126
##                8         9        10
## AIC(n)  2.422751  2.429609  2.433897
## HQ(n)   2.696680  2.736741  2.774233
## SC(n)   3.139096  3.232783  3.323901
## FPE(n) 11.278421 11.356682 11.406304
```

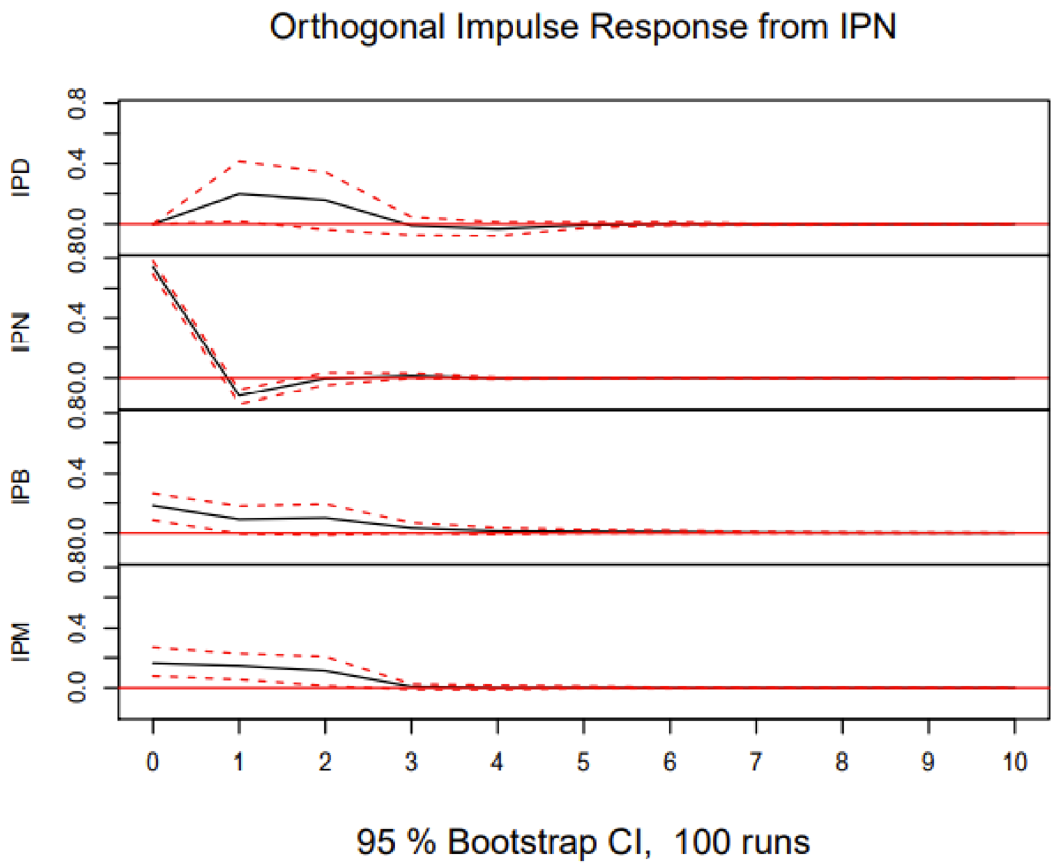
The VARselect function suggest SC(n) criteria at lag 2, then construct

```
m2.vars=VAR(Zt, p = 2)
summary(m2.vars)
```

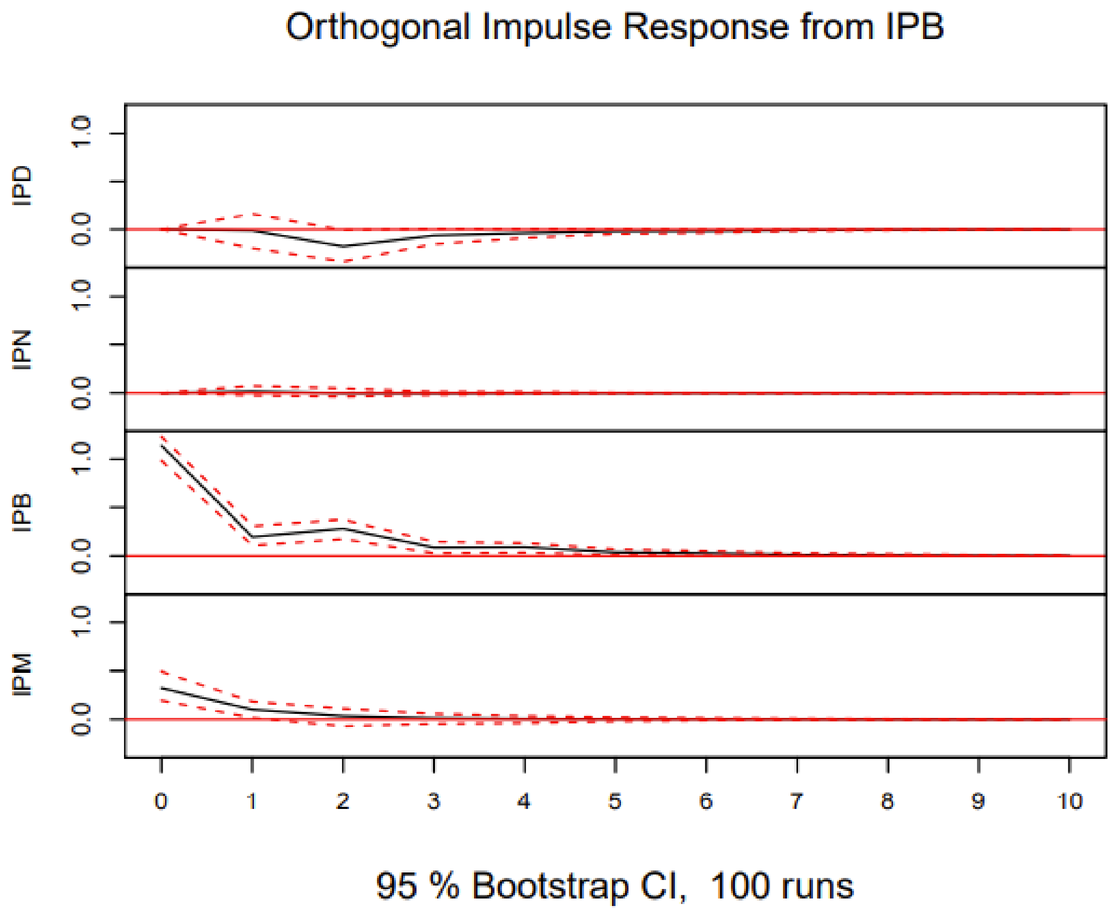


The impulse response from IPD affect IPD industry the most according to the IRF graph above

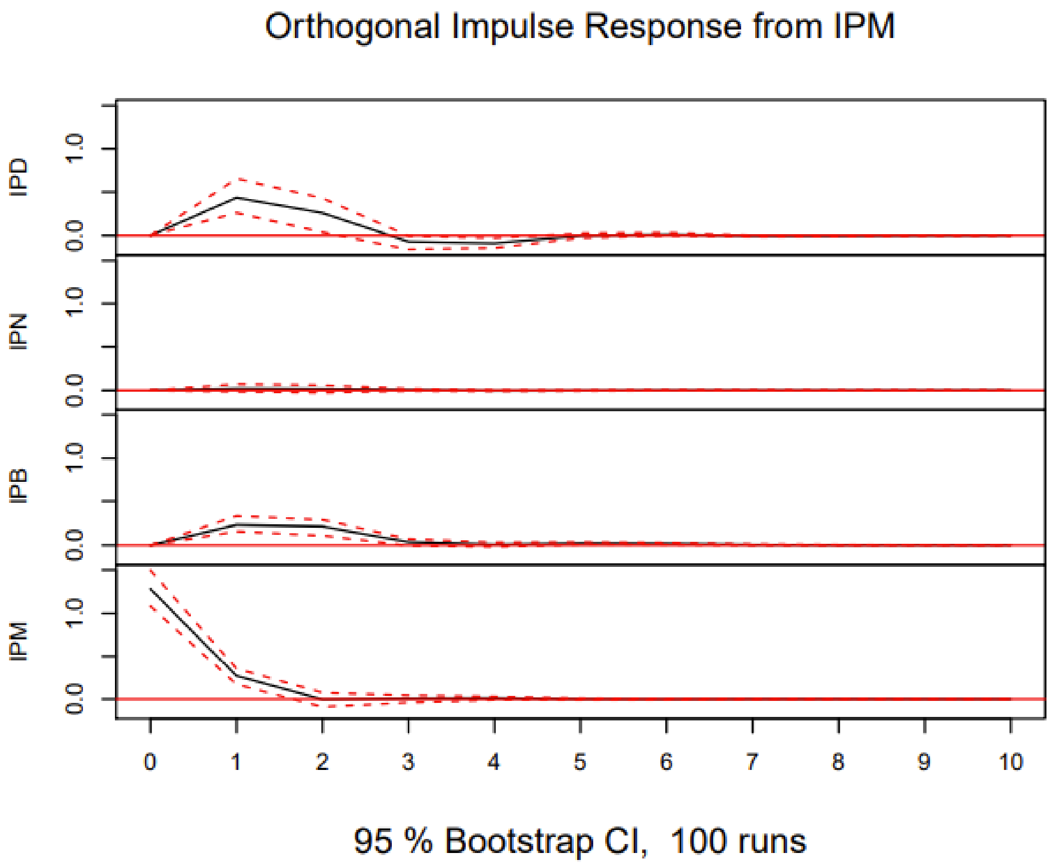
The effect heavily affects IPD and slowly faded by the 3rd period.



The impulse from IPN heavily affect IPN industry by the 1st period and slowly dried up by the 2nd period.



The impulse effect from IPB industry affect IPB the most, mostly positive. The effect slowly reduced after the 1st period and faded away by the 4th period. The impulse response function of IPD industries also shows slightly negative response from the impact from IPB industry during 1st to 2nd period and faded away by the 4th period.



The impact of IPM industry affect IPM itself the most, the impact slowly declined and eliminated after the 2nd period. The impulse response function of IPD and IPB also shows slight affected by the impact during 0 to 3rd period. However, the effect prolong to 5th period in IPD industry.

```r
#1.4
fevd(m2.vars, n.ahead=20)
```

```
## $IPD
##              IPD         IPN          IPB        IPM
##  [1,] 1.0000000 0.000000000 0.000000e+00 0.00000000
##  [2,] 0.9786677 0.003677477 1.547554e-05 0.01763932
##  [3,] 0.9681556 0.005825149 2.865465e-03 0.02315381
##  [4,] 0.9673749 0.005826039 3.233717e-03 0.02356535
##  [5,] 0.9664951 0.005913986 3.365580e-03 0.02422538
##  [6,] 0.9664566 0.005918078 3.400740e-03 0.02422457
##  [7,] 0.9664239 0.005917334 3.427742e-03 0.02423098
##  [8,] 0.9664153 0.005917546 3.434368e-03 0.02423283
##  [9,] 0.9664095 0.005918007 3.436407e-03 0.02423607
## [10,] 0.9664087 0.005918072 3.437077e-03 0.02423614
## [11,] 0.9664084 0.005918071 3.437398e-03 0.02423613
## [12,] 0.9664083 0.005918075 3.437499e-03 0.02423614
## [13,] 0.9664082 0.005918079 3.437531e-03 0.02423616
## [14,] 0.9664082 0.005918080 3.437542e-03 0.02423617
## [15,] 0.9664082 0.005918080 3.437546e-03 0.02423617
## [16,] 0.9664082 0.005918080 3.437547e-03 0.02423617
## [17,] 0.9664082 0.005918080 3.437548e-03 0.02423617
## [18,] 0.9664082 0.005918080 3.437548e-03 0.02423617
## [19,] 0.9664082 0.005918080 3.437548e-03 0.02423617
## [20,] 0.9664082 0.005918080 3.437548e-03 0.02423617
##
```

The forecast error variance decomposition with 20 lags ahead, the impact from IPD industry affect IPD itself the most according to the statistical data above.

```
## $IPN
##              IPD       IPN          IPB          IPM
##  [1,] 0.1130659 0.8869341 0.0000000000 0.0000000000
##  [2,] 0.1176617 0.8812807 0.0003848728 0.0006727082
##  [3,] 0.1176060 0.8808902 0.0003857537 0.0011180226
##  [4,] 0.1178793 0.8805521 0.0003980169 0.0011706388
##  [5,] 0.1178761 0.8805347 0.0003990007 0.0011902249
##  [6,] 0.1178942 0.8805099 0.0003989982 0.0011968518
##  [7,] 0.1178942 0.8805093 0.0003991068 0.0011974314
##  [8,] 0.1178956 0.8805076 0.0003991836 0.0011976850
##  [9,] 0.1178956 0.8805075 0.0003991979 0.0011977477
## [10,] 0.1178956 0.8805074 0.0003991997 0.0011977775
## [11,] 0.1178956 0.8805074 0.0003992011 0.0011977783
## [12,] 0.1178956 0.8805074 0.0003992018 0.0011977789
## [13,] 0.1178956 0.8805074 0.0003992021 0.0011977792
## [14,] 0.1178956 0.8805074 0.0003992021 0.0011977793
## [15,] 0.1178956 0.8805074 0.0003992021 0.0011977793
## [16,] 0.1178956 0.8805074 0.0003992021 0.0011977793
## [17,] 0.1178956 0.8805074 0.0003992021 0.0011977793
## [18,] 0.1178956 0.8805074 0.0003992021 0.0011977793
## [19,] 0.1178956 0.8805074 0.0003992021 0.0011977793
## [20,] 0.1178956 0.8805074 0.0003992021 0.0011977793
##
```

The impact from IPN industry affect IPN itself the most

```
## $IPB
##             IPD        IPN       IPB        IPM
##  [1,] 0.5069355 0.01224107 0.4808235 0.00000000
##  [2,] 0.5048449 0.01432290 0.4624710 0.01836117
##  [3,] 0.4828424 0.01696945 0.4679075 0.03228060
##  [4,] 0.4812822 0.01727732 0.4688451 0.03259533
##  [5,] 0.4808006 0.01725670 0.4694796 0.03246315
##  [6,] 0.4806182 0.01727671 0.4695372 0.03256786
##  [7,] 0.4804322 0.01729458 0.4696161 0.03265708
##  [8,] 0.4803961 0.01729859 0.4696392 0.03266607
##  [9,] 0.4803859 0.01729914 0.4696485 0.03266650
## [10,] 0.4803820 0.01729948 0.4696507 0.03266778
## [11,] 0.4803799 0.01729967 0.4696519 0.03266859
## [12,] 0.4803793 0.01729973 0.4696522 0.03266874
## [13,] 0.4803791 0.01729974 0.4696524 0.03266877
## [14,] 0.4803790 0.01729975 0.4696524 0.03266879
## [15,] 0.4803790 0.01729975 0.4696524 0.03266880
## [16,] 0.4803790 0.01729975 0.4696524 0.03266880
## [17,] 0.4803790 0.01729975 0.4696524 0.03266880
## [18,] 0.4803790 0.01729975 0.4696524 0.03266880
## [19,] 0.4803790 0.01729975 0.4696524 0.03266880
## [20,] 0.4803790 0.01729975 0.4696524 0.03266880
##
```

The impact of IPB heavily influence the IPD industry the most, followed by the IPB industry itself according to the FEVD function above.

```
## $IPM
##             IPD        IPN        IPB       IPM
##  [1,] 0.2369914 0.01140501 0.04501385 0.7065897
##  [2,] 0.2512518 0.01903374 0.04569678 0.6840177
##  [3,] 0.2554440 0.02378324 0.04554842 0.6752243
##  [4,] 0.2555947 0.02379966 0.04558903 0.6750166
##  [5,] 0.2556231 0.02379681 0.04559513 0.6749849
##  [6,] 0.2556249 0.02379765 0.04559845 0.6749790
##  [7,] 0.2556289 0.02379741 0.04560048 0.6749732
##  [8,] 0.2556296 0.02379737 0.04560115 0.6749719
##  [9,] 0.2556295 0.02379740 0.04560134 0.6749718
## [10,] 0.2556295 0.02379741 0.04560140 0.6749717
## [11,] 0.2556295 0.02379741 0.04560143 0.6749717
## [12,] 0.2556295 0.02379741 0.04560144 0.6749717
## [13,] 0.2556295 0.02379741 0.04560145 0.6749717
## [14,] 0.2556295 0.02379741 0.04560145 0.6749717
## [15,] 0.2556295 0.02379741 0.04560145 0.6749717
## [16,] 0.2556295 0.02379741 0.04560145 0.6749717
## [17,] 0.2556295 0.02379741 0.04560145 0.6749717
## [18,] 0.2556295 0.02379741 0.04560145 0.6749717
## [19,] 0.2556295 0.02379741 0.04560145 0.6749717
## [20,] 0.2556295 0.02379741 0.04560145 0.6749717
```

Similar to IPD and IPN, the impact of IPM industry affect itself the most.

1.5 Compute 1-step to 6-step ahead predictions of $\mathbf{z}_t$ at the forecast origin h = 879 (April 2020). Obtain 95 % interval forecasts for each component series.

```r
m2vars.forecast <- predict(m2.vars, n.ahead = 6, ci=0.95)
m2vars.forecast
```

```
## $IPD
##              fcst     lower    upper       CI
## [1,] -0.06150473 -6.300249 6.177239 6.238744
## [2,]  0.77907327 -5.624736 7.182883 6.403809
## [3,]  0.18461230 -6.318302 6.687526 6.502914
## [4,]  0.05341044 -6.455561 6.562382 6.508971
## [5,]  0.18430627 -6.330032 6.698645 6.514339
## [6,]  0.24660374 -6.267897 6.761105 6.514501
##
## $IPN
##              fcst     lower    upper       CI
## [1,] -0.05426783 -1.596737 1.488201 1.542469
## [2,]  0.15209749 -1.415492 1.719687 1.567589
## [3,]  0.19159580 -1.376365 1.759557 1.567961
## [4,]  0.15594667 -1.412572 1.724465 1.568519
## [5,]  0.15469714 -1.413842 1.723237 1.568540
## [6,]  0.16062462 -1.407939 1.729188 1.568563
##
## $IPB
##            fcst     lower    upper       CI
## [1,] 0.9027450 -2.323714 4.129204 3.226459
## [2,] 0.7194082 -2.619747 4.058564 3.339155
## [3,] 0.5411836 -2.877592 3.959959 3.418775
## [4,] 0.3508809 -3.073996 3.775757 3.424877
## [5,] 0.3737427 -3.059243 3.806729 3.432986
## [6,] 0.3594820 -3.075094 3.794058 3.434576
##
## $IPM
##            fcst     lower    upper       CI
## [1,] 0.3852687 -2.604896 3.375434 2.990165
## [2,] 0.3011154 -2.805045 3.407276 3.106160
## [3,] 0.2343799 -2.891949 3.360708 3.126329
## [4,] 0.2485357 -2.878334 3.375405 3.126869
## [5,] 0.2380792 -2.888995 3.365153 3.127074
## [6,] 0.2275148 -2.899574 3.354604 3.127089
```

If your last student id number is the even number, you have to use the series of Dogecoin from January 01, 2015 to May 24, 2021.

The data are available from Yahoo Finance.

2.1 propose the research paper to identify the optimal model for forecasting the log returns and its volatilities.

```
#2
da <- read.table("DOGE-USD.txt", fill=TRUE, header=TRUE)
D <- (da)
MOON=as.numeric(D[,6])
```

```
## Warning: NAs introduced by coercion
```

```
DD=log(MOON)
DDD=diff(DD)*100
```

```
Dt=na.omit(DDD)
```

To obtain the optimal model for forecasting log returns and volatilities of Dogecoin per US dollars. First, importing the data available from the Yahoo Finance since January 1st, 2015, to May 24th, 2021. Selecting the adjusted price in the 6th column of the data and convert the data into log price using log(x) command function. After obtaining log price of Doge-coin/US dollars, taking the difference between log price in a daily frequency to acquire the log return of Doge-coin. As illustrated in the figure above. With some NAs produced during the process of transforming the data, omitting NAs is require in order to procced to the finding optimal model for Dogecoin data, by using na.omit function.

After obtaining processed data, using partial auto-correlation function, auto regressive and auto arima function order to identify optimal lag for the model, the suggestion in AR and auto arima function are difference. However, auto arima function suggest the lag with lowest possible BIC value according to the BIC criteria at lag 3 with Autoregressive and Moving Average model at lag 3 with BIC value equal to 15976.33. Constructing ARMA model (3,0,3) using arima function for mean model.

$$DOGE_t = \underset{(0.1685)}{0.3246} \underset{(0.2863)}{-0.1365 DOGE_{t-1}} \underset{(0.1375)}{-0.6177\, DOGE_{t-2}} \underset{(0.2476)}{0.2899 DOGE_{t-3}} \underset{(0.2920)}{-0.1575 a_{t-1}}$$

$$\underset{(0.1342)}{-0.668 a_{t-2}} \underset{(0.2587)}{+0.2124 a_{t-3}} + a_t$$

Proceeding to performing T-test to identify the significant of mean with null hypothesis: mean is equal to zero, and alternative hypothesis: mean is significantly differed from zero. The calculated p-value = 0.03448, less than 0.05. Thus, we can reject null hypothesis in favor of alternative hypothesis at 95% confidence interval, mean of Dogecoin is significantly differ from zero. In addition, performing Ljung-Box test on both residuals and squared residuals of Dogecoin for linear and non-linear dependence of the data. The calculated p-value for Ljung-box test on residual = 0.0026, we can reject null hypothesis: the data contains linear dependence problem at 95% confidence interval. Therefore, the data does not contain linear dependence problem. On the contrary, Ljung-Box test on squared residuals of log return of Dogecoin, calculated p-value is less than 0.05, rejecting null hypothesis: data contains ARCH effect in favor of alternative hypothesis at 95% confidence interval. The log return series of Dogecoin contains ARCH effect.

```
G1=garchFit(~arma(3,3)+garch(1,1),data=Dt,trace=F)
```

```
G2=garchFit(~arma(3,3)+garch(1,1),data=Dt,cond.dist='std',trace=F)
```

```
G3=garchFit(~arma(2,2)+garch(1,1),data=Dt,cond.dist='std',trace=F)
```

```
G4=garchFit(~arma(1,1)+garch(1,1),data=Dt,cond.dist='std',trace=F)
```

```
G5=garchFit(~garch(1,1),data=Dt,cond.dist='std',trace=F)
```

Due to the occurrence of ARCH effect in the series, constructing GARCH model to eliminate ARCH effect on the series. Firstly, constructing ARMA (3,3) and GARCH (1,1) model represented by G1. The model is inadequate, and the distribution is unequal in the first model (G1). Furthermore, constructing ARMA (3,3) and GARCH (1,1) with condition distribution = std in attempt to create adequacy and normal distribution represented in G2. The second model could not construct the adequacy and the mean model is insignificant. Reducing ARMA lag to create significant in the mean model, G3 and G4 represents model with lower lag of ARMA. Nevertheless, G3 and G4 could not construct the significant mean model according to the Ljung-Box test on the residuals of both models. The fifth model constructed with pure GARCH and conditional distribution=std represented in G5. The model indicates significant means and variance model at 99% confidence interval, adequate and significant for forecasting mean and volatility of the Dogecoin log return series.

$$mean = \frac{-0.13885}{(0.05137)}$$

$$varaince = \frac{1.41763}{(0.12416)} + \frac{0.55619a_{t-1}}{(0.02888)}$$

Question3 (20 marks)

Consider the quarterly real GDP of United Kingdom, Canada, and the United States from the first quarter of 1980 to the second quarter of 2011. The data are available from the Federal Reserve Bank of St. Louis (FRED). Let $\mathbf{z}_t$ be the log GDP series. You can learn how to download the data from the Federal Reserve Bank of St. Louis .

3.1 Build a VAR model for $\mathbf{z}_t$, including simplification (or refine) the model by removing insignificant parameters with threshold of t-ratio 1.645 and perform model checking. Write down the final fitted model.

```
US=read.table("USGDP.txt", header=T)
CA=read.table("CAGDP.txt", header=T)
UK=read.table("UKGDP.txt", header=T)
X=cbind(as.numeric(US$GDPC1), as.numeric(CA$NAEXKP01CAQ189S), as.numeric(UK$CLVMNACSCAB1GQUK))
Z=log(X)
Zt=diffM(Z)*100
colnames(Zt)<-c('US' , 'CA', 'UK')

#finding optimal lag for the model with lowest possible BIC criteria
#simple VAR model
VARorder(Zt)
```

```
## selected order: aic =  1
## selected order: bic =  1
## selected order: hq =  1
## Summary table:
##          p     AIC     BIC      HQ     M(p) p-value
## [1,]     0 -3.0738 -3.0738 -3.0738   0.0000  0.0000
## [2,]     1 -4.0259 -3.8222 -3.9432 117.8334  0.0000
## [3,]     2 -4.0234 -3.6161 -3.8580  14.7895  0.0969
## [4,]     3 -3.9305 -3.3196 -3.6823   5.1831  0.8181
## [5,]     4 -3.9171 -3.1025 -3.5862  12.8644  0.1688
```

Using the suggestion from VARorder function by using lag 1 as the optimal lag with lowest possible BIC value to construct VAR model with lag =1.

```
G.MTS=VAR(Zt,1)
```

```
## Constant term:
## Estimates:  0.3067611 0.0616576 0.2249959
## Std.Error:  0.07752755 0.07398554 0.073608
## AR coefficient matrix
## AR( 1 )-matrix
```

```
##        [,1]   [,2]  [,3]
## [1,] 0.1475 0.1667 0.335
## [2,] 0.3732 0.2238 0.285
## [3,] 0.0839 0.0604 0.461
## standard error
##        [,1]   [,2]   [,3]
## [1,] 0.0964 0.0880 0.0854
## [2,] 0.0920 0.0839 0.0815
## [3,] 0.0916 0.0835 0.0811
##
## Residuals cov-mtx:
##           [,1]       [,2]       [,3]
## [1,] 0.3621711 0.14536356 0.05115430
## [2,] 0.1453636 0.32983392 0.01119834
## [3,] 0.0511543 0.01119834 0.32647630
##
## det(SSE) =  0.03135906
## AIC =  -3.318252
## BIC =  -3.114614
## HQ  =  -3.235525
```

VAR model:

$$\begin{bmatrix} US \\ CA \\ UK \end{bmatrix} = \begin{bmatrix} 0.30676 \\ 0.06166 \\ 0.22500 \end{bmatrix} + \begin{bmatrix} 0.1475 & 0.1667 & 0.335 \\ 0.3732 & 0.2238 & 0.285 \\ 0.0839 & 0.0604 & 0.461 \end{bmatrix} \begin{bmatrix} US_{t-1} \\ CA_{t-1} \\ UK_{t-1} \end{bmatrix}$$

Refine VAR model with threshold at 1.645

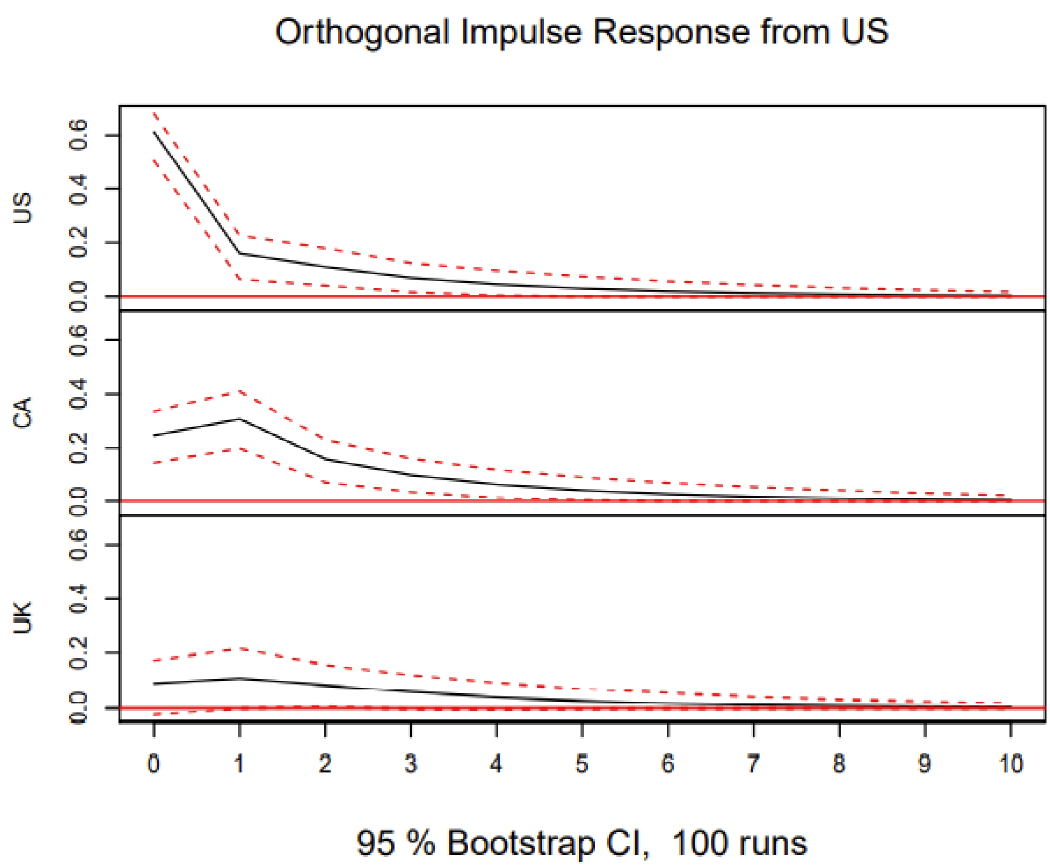
$$\begin{bmatrix} US \\ CA \\ UK \end{bmatrix} = \begin{bmatrix} 0.33811 \\ 0 \\ 0.28175 \end{bmatrix} + \begin{bmatrix} 0 & 0.239 & 0.378 \\ 0.393 & 0.236 & 0.302 \\ 0 & 0 & 0.526 \end{bmatrix} \begin{bmatrix} US_{t-1} \\ CA_{t-1} \\ UK_{t-1} \end{bmatrix}$$

```
## Ljung-Box Statistics:
##          m       Q(m)      df     p-value
##  [1,]   1.00      8.33     9.00      0.50
##  [2,]   2.00     13.48    18.00      0.76
##  [3,]   3.00     21.85    27.00      0.74
##  [4,]   4.00     35.85    36.00      0.48
##  [5,]   5.00     43.84    45.00      0.52
##  [6,]   6.00     54.32    54.00      0.46
##  [7,]   7.00     59.77    63.00      0.59
##  [8,]   8.00     71.52    72.00      0.49
##  [9,]   9.00     78.77    81.00      0.55
## [10,]  10.00     87.70    90.00      0.55
## [11,]  11.00     91.24    99.00      0.70
## [12,]  12.00    103.76   108.00      0.60
## [13,]  13.00    110.59   117.00      0.65
## [14,]  14.00    118.35   126.00      0.67
## [15,]  15.00    132.46   135.00      0.55
## [16,]  16.00    138.07   144.00      0.62
## [17,]  17.00    143.41   153.00      0.70
## [18,]  18.00    148.81   162.00      0.76
## [19,]  19.00    163.45   171.00      0.65
## [20,]  20.00    167.81   180.00      0.73
## [21,]  21.00    170.36   189.00      0.83
## [22,]  22.00    175.58   198.00      0.87
```

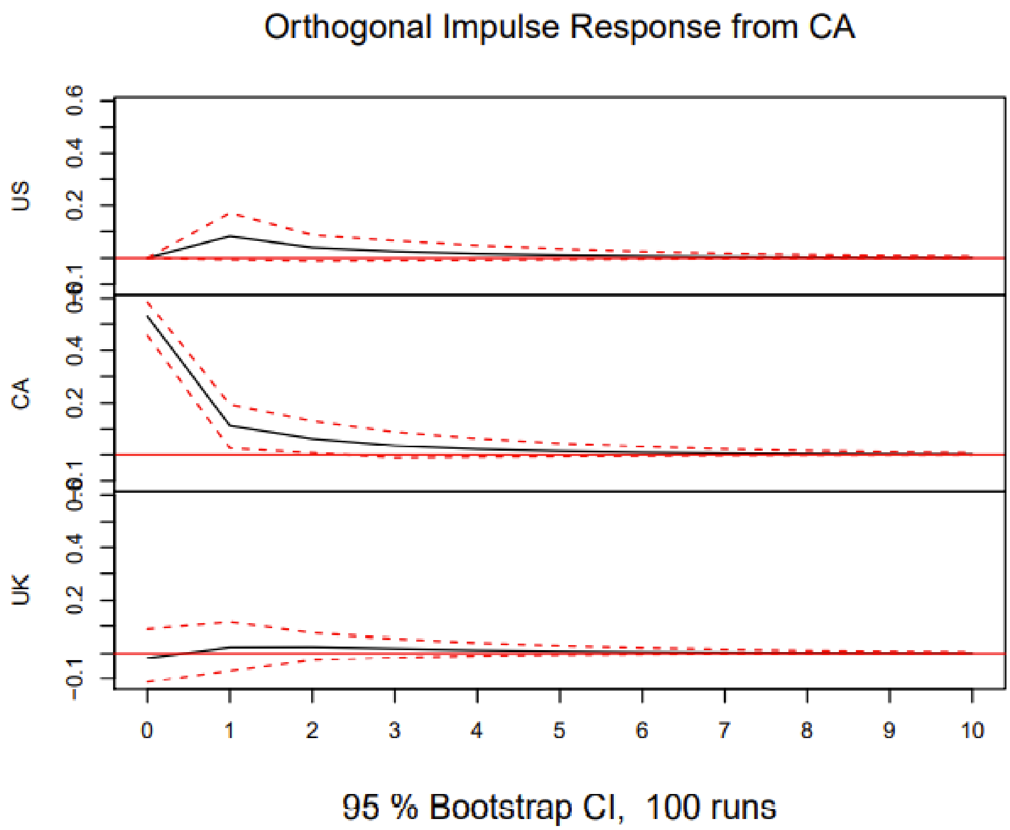
The Ljung-Box statistic at lag 1 indicates the significant of the model. We cannot reject null hypothesis at 95% confidence interval; thus, the model does not contain serial correlation problem and adequate to use.

3.2 Obtain the plots of impulse response function of the fitted model. Interpret the plots.

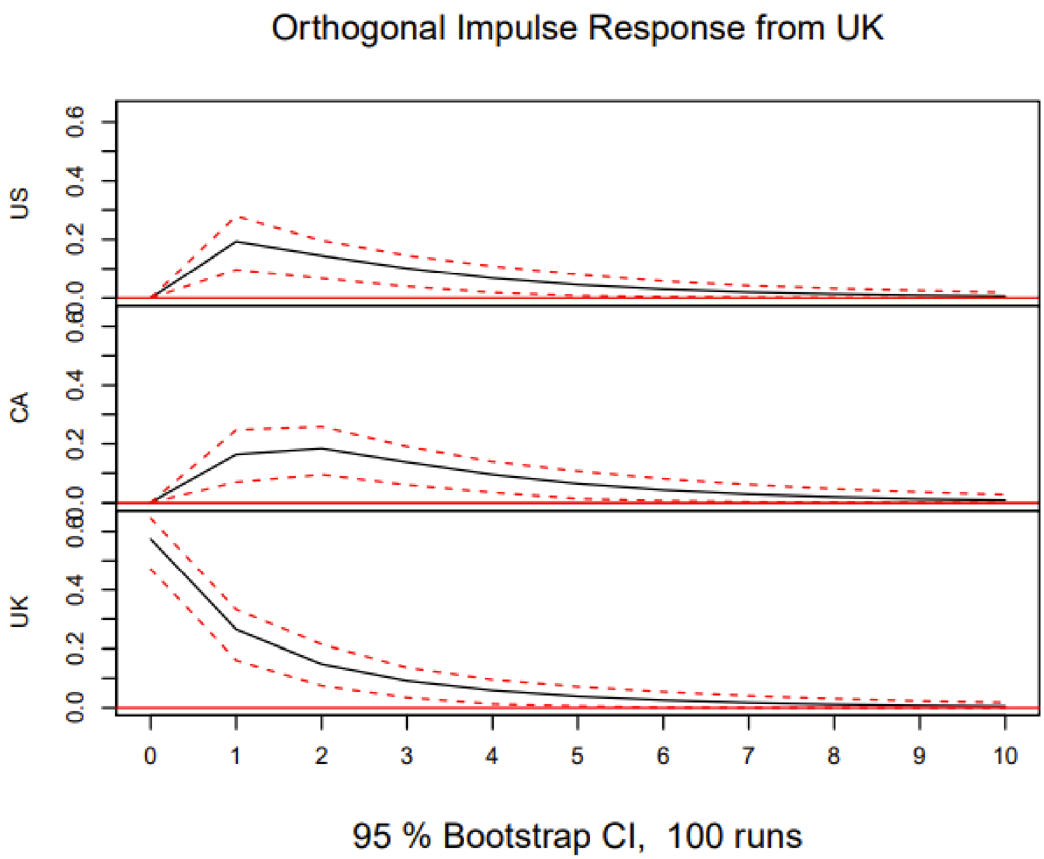
```
#3.2
IRF2=irf(fittedmodel)
plot(IRF2)
```



The impact on the changes in US real gross domestic product (GDP) affect US real GDP itself the most, followed by Canada and poses slightly effect on UK real GDP as well, mostly positive effects. The impulse response function indicates the impulse impact are mostly extended to 10th period after the initial impact at a declining rate after 2nd period.



The impact from changes in Canada real GDP affect Canada itself the most, similar to Orthogonal impulse response from US. However, the shocks slight impose impact on US and UK unlike Us. In addition, the impulse effect also extended to the 10th period with declining rate after 2nd period.



The impulse response from UK shows that the impact significantly impacts UK, at a larger scale than the first 2 IRF. Shock from changes in UK real GDP heavily influence US and Canada real GDP and the period of the impact has been prolonged to 10^{th} period as well but at a much slower declining rate than the other IRF.

3.3 Obtain the forecast error variance decomposition. Interpret the results.

```r
#3.3
fevd(fittedmodel, n.ahead = 20)
```

```
## $US
##             US         CA         UK
##  [1,] 1.0000000 0.00000000 0.00000000
##  [2,] 0.9014051 0.01522190 0.08337296
##  [3,] 0.8618029 0.01728981 0.12090727
##  [4,] 0.8441355 0.01794729 0.13791725
##  [5,] 0.8364629 0.01818767 0.14534944
##  [6,] 0.8331224 0.01828595 0.14859163
##  [7,] 0.8316649 0.01832783 0.15000730
##  [8,] 0.8310278 0.01834597 0.15062623
##  [9,] 0.8307491 0.01835389 0.15089704
## [10,] 0.8306271 0.01835735 0.15101559
## [11,] 0.8305736 0.01835886 0.15106751
## [12,] 0.8305502 0.01835953 0.15109024
## [13,] 0.8305400 0.01835982 0.15110019
## [14,] 0.8305355 0.01835994 0.15110455
## [15,] 0.8305335 0.01836000 0.15110646
## [16,] 0.8305327 0.01836002 0.15110730
## [17,] 0.8305323 0.01836004 0.15110767
## [18,] 0.8305321 0.01836004 0.15110783
## [19,] 0.8305321 0.01836004 0.15110790
## [20,] 0.8305320 0.01836004 0.15110793
##
```

```
## $CA
##             US        CA         UK
##  [1,] 0.1768895 0.8231105 0.00000000
##  [2,] 0.3263397 0.6174899 0.05617044
##  [3,] 0.3350550 0.5526210 0.11232403
##  [4,] 0.3348694 0.5256374 0.13949326
##  [5,] 0.3341671 0.5142020 0.15163093
##  [6,] 0.3337738 0.5092796 0.15694662
##  [7,] 0.3335888 0.5071426 0.15926863
##  [8,] 0.3335058 0.5062107 0.16028348
##  [9,] 0.3334692 0.5058034 0.16072739
## [10,] 0.3334532 0.5056251 0.16092168
## [11,] 0.3334461 0.5055471 0.16100675
## [12,] 0.3334430 0.5055130 0.16104400
## [13,] 0.3334417 0.5054980 0.16106031
## [14,] 0.3334411 0.5054915 0.16106745
## [15,] 0.3334408 0.5054886 0.16107058
## [16,] 0.3334407 0.5054873 0.16107195
## [17,] 0.3334407 0.5054868 0.16107255
## [18,] 0.3334407 0.5054865 0.16107282
## [19,] 0.3334406 0.5054864 0.16107293
## [20,] 0.3334406 0.5054864 0.16107298
##
```

```
## $UK
##              US           CA        UK
##  [1,] 0.02213088 0.0009828026 0.9768863
##  [2,] 0.04461130 0.0021202629 0.9532684
##  [3,] 0.05632526 0.0033360235 0.9403387
##  [4,] 0.06173350 0.0039872011 0.9342793
##  [5,] 0.06413333 0.0042902726 0.9315764
##  [6,] 0.06518672 0.0044255088 0.9303878
##  [7,] 0.06564791 0.0044850658 0.9298670
##  [8,] 0.06584977 0.0045111883 0.9296390
##  [9,] 0.06593814 0.0045226326 0.9295392
## [10,] 0.06597684 0.0045276449 0.9294955
## [11,] 0.06599378 0.0045298401 0.9294764
## [12,] 0.06600120 0.0045308014 0.9294680
## [13,] 0.06600445 0.0045312225 0.9294643
## [14,] 0.06600587 0.0045314069 0.9294627
## [15,] 0.06600650 0.0045314877 0.9294620
## [16,] 0.06600677 0.0045315230 0.9294617
## [17,] 0.06600689 0.0045315385 0.9294616
## [18,] 0.06600694 0.0045315453 0.9294615
## [19,] 0.06600696 0.0045315483 0.9294615
## [20,] 0.06600697 0.0045315496 0.9294615
```

The forecast error variance decomposition indicates that most of the shocks generated by each country impact itself the most according to the fevd function above.

3.4 Propose your co-integrating vector, and check it by using the Engle-Granger cointegration test.

```
#3.4
#install.packages('fUnitRoots')
library(quantmod)
USTEST=cbind(as.numeric(US$GDPC1))
UKTEST=cbind(as.numeric(UK$CLVMNACSCAB1GQUK))
CATEST=cbind(as.numeric(CA$NAEXKP01CAQ189S))
fitted <- lm(USTEST ~ UKTEST + CATEST)
summary(fitted)
```

```
##
## Call:
## lm(formula = USTEST ~ UKTEST + CATEST)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -396.82 -119.69   10.52  139.80  322.70
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.604e+03  7.025e+01 -22.834  < 2e-16 ***
## UKTEST       3.084e-02  2.605e-03  11.839  < 2e-16 ***
## CATEST       1.090e-08  2.431e-09   4.484 1.66e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 174.5 on 123 degrees of freedom
## Multiple R-squared:  0.9967, Adjusted R-squared:  0.9966
## F-statistic: 1.859e+04 on 2 and 123 DF,  p-value: < 2.2e-16
```

Then performing Engle-G

```
#engle-granger cointegration test

adf.test(UKTEST)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  UKTEST
## Dickey-Fuller = -1.7183, Lag order = 4, p-value = 0.6936
## alternative hypothesis: stationary
```

```
adf.test(diff(UKTEST))
```

```
## Warning in adf.test(diff(UKTEST)): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(UKTEST)
## Dickey-Fuller = -4.2135, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

```
adf.test(CATEST)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  CATEST
## Dickey-Fuller = -2.5166, Lag order = 4, p-value = 0.3619
## alternative hypothesis: stationary
```

```
adf.test(diff(CATEST))
```

```
## Warning in adf.test(diff(CATEST)): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(CATEST)
## Dickey-Fuller = -4.0709, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

```
adf.test(USTEST)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  USTEST
## Dickey-Fuller = -2.7836, Lag order = 4, p-value = 0.2509
## alternative hypothesis: stationary
```

```
adf.test(diff(USTEST))
```

```
## Warning in adf.test(diff(USTEST)): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(USTEST)
## Dickey-Fuller = -4.4039, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

```
error = residuals(fitted)
adf.test(error)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  error
## Dickey-Fuller = -2.6669, Lag order = 4, p-value = 0.2994
## alternative hypothesis: stationary
```

3.5 Build an ECM-VAR model (aka. VECM). Write down the fitted model and interpret the result.

```
VECM1 <- lm(DUK ~ DUKL1+DCAL1+DUSL1+EL1)
VECM2 <- lm(DUS ~ DUKL1+DCAL1+DUSL1+EL1)
VECM3 <- lm(DCA ~ DUKL1+DCAL1+DUSL1+EL1)
summary(VECM1)
```

```
##
## Call:
## lm(formula = DUK ~ DUKL1 + DCAL1 + DUSL1 + EL1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5487.3  -807.2    28.3  1041.4  3805.4
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.906e+02  2.026e+02   2.915  0.00425 **
## DUKL1        5.748e-01  8.079e-02   7.115  9.1e-11 ***
## DCAL1       -6.933e-08  7.636e-08  -0.908  0.36570
## DUSL1        3.772e+00  2.521e+00   1.497  0.13716
## EL1          2.443e+00  8.745e-01   2.793  0.00608 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1604 on 119 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.4709, Adjusted R-squared:  0.4532
## F-statistic: 26.48 on 4 and 119 DF,  p-value: 1.027e-15
```

$$DUK = 5.906e^{2} + 0.5748DUK_{t-1} - 6.933e^{-8}DCA_{t-1} + 3.772DUS_{t-1} + 2.443\hat{e}_{t-1} + \mathcal{E}_t$$

```r
summary(VECM2)
```

```
##
## Call:
## lm(formula = DUS ~ DUKL1 + DCAL1 + DUSL1 + EL1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -253.153  -33.513    2.067   46.046  136.263
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.414e+01  8.123e+00   4.202 5.13e-05 ***
## DUKL1       1.781e-02  3.239e-03   5.498 2.23e-07 ***
## DCAL1       1.769e-09  3.061e-09   0.578    0.564
## DUSL1       9.813e-02  1.011e-01   0.971    0.334
## EL1         2.336e-02  3.506e-02   0.666    0.507
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 64.3 on 119 degrees of freedom
##   (1 observation deleted due to missingness)

## Multiple R-squared:  0.3623, Adjusted R-squared:  0.3408
## F-statistic:  16.9 on 4 and 119 DF,  p-value: 5.354e-11
```

$$DUS = 34.14 + 0.01781DUK_{t-1} - 1.769e^{-9}DCA_{t-1} + 0.09813DUS_{t-1} + 0.02336\hat{e}_{t-1} + \mathcal{E}_t$$

```r
summary(VECM3)
```

```
##
## Call:
## lm(formula = DCA ~ DUKL1 + DCAL1 + DUSL1 + EL1)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -3.327e+09 -1.047e+09 -1.357e+08  1.083e+09  4.114e+09
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.541e+08  2.115e+08   1.202 0.231933
## DUKL1       3.747e+05  8.432e+04   4.444    2e-05 ***
## DCAL1       1.886e-01  7.970e-02   2.366 0.019583 *
## DUSL1       9.414e+06  2.631e+06   3.578 0.000502 ***
## EL1         1.554e+06  9.128e+05   1.702 0.091291 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.674e+09 on 119 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.5472, Adjusted R-squared:  0.532
## F-statistic: 35.96 on 4 and 119 DF,  p-value: < 2.2e-16
```

$$DCA = 2.541e^{8} + 3.747e^{5}DUK_{t-1} + 0.1886DCA_{t-1} + 9.414e^{6}DUS_{t-1} + 0.02336e^{6}\hat{e}_{t-1} + \mathcal{E}_t$$

$$DUK = 5.906e^{2} + 0.5748DUK_{t-1} - 6.933e^{-8}DCA_{t-1} + 3.772DUS_{t-1} + 2.443\hat{e}_{t-1} + \mathcal{E}_t$$

$$DUS = 34.14 + 0.01781DUK_{t-1} - 1.769e^{-9}DCA_{t-1} + 0.09813DUS_{t-1} + 0.02336\hat{e}_{t-1} + \mathcal{E}_t$$

$$DCA = 2.541e^{8} + 3.747e^{5}DUK_{t-1} + 0.1886DCA_{t-1} + 9.414e^{6}DUS_{t-1} + 0.02336e^{6}\hat{e}_{t-1} + \mathcal{E}_t$$