Is annual growth rate of industrial production index, as a leading indicator, a reliable prediction indicator for unemployment rate in Taiwan?

Yu-Ting Huang yh23469

May 2, 2021

Contents

1 Abstract

Our main purpose of this paper is to examine whether annual growth rate of industrial production index, as a leading indicator, is a reliable prediction indicator for unemployment rate in Taiwan. The data is collected on National Statistic from January 2001 to December 2020. First, we utilize the VAR model to straightforwardly understand their relationship. Then we predict unemployment rate by Pseudo out-of-sample and compare RMSFE between AR model and ADL model. Eventually, the results present ADL model have better prediction power, therefore, annual growth rate of industrial production index is a valid prediction indicator for unemployment rate in Taiwan.

2 Introduction

The unemployment rate is an important economic indicator and can also be used as a manifestation of human resources. If the unemployment rate is too high, it may indicate idle labor force and lead to inefficiency in the market, in the same time, people may decrease purchase power then influence production market. Therefore, the government should always concern about this issue to prevent social unrest.

As a leading indicator, the industrial production index is a statistically weighted index of the output of the manufacturing, housing, mining and water, electricity and gas industries to measure the real total output of the four industries, which are very sensitive to economic changes and are one of the important indicators to observe economic fluctuations.

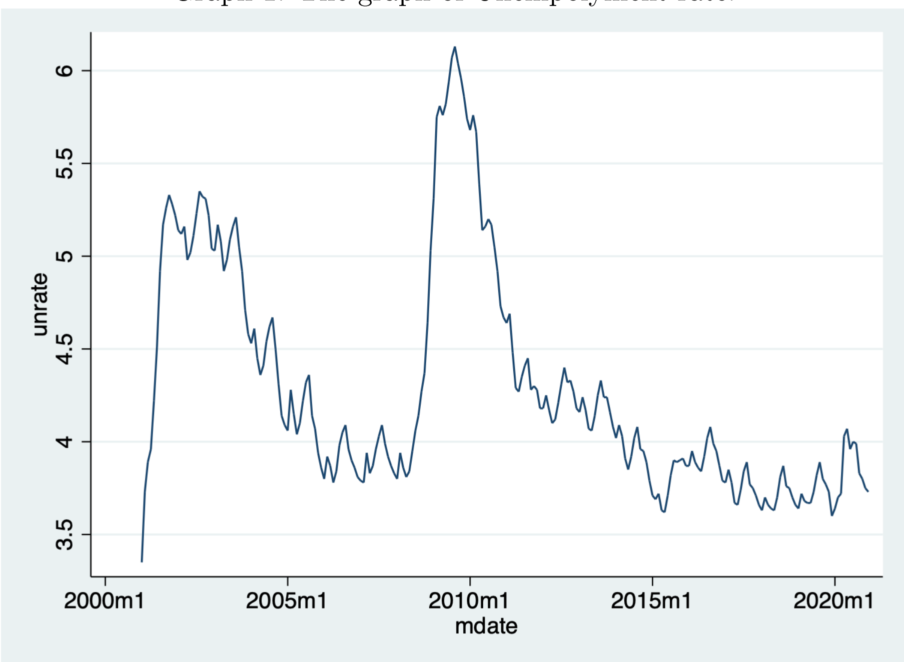
Moreover, annual variation in industrial production provides insight into the state of the economic cycle, because we may expect that the production of consumer durables and capital goods are likely to reduce during an economic downturn. Therefore, it is a leading indicator of Gross Domestic Product (GDP) growth and economic performance due to its sensitivity to consumer demand and interest rates, although the industrial sector only accounts for a portion of an economy's total output.

3 Data

This article uses 240 monthly data on unempolyment rates ($unrate_t$) and annul growth rate of industrial production index (ip_g_t) from January 2001 to December 2020. The data are collected from National Statistics, R.O.C(Taiwan). Note that we chose annual growth rate of industrial production index rather than industrial production index in the beginning because the annul growth rate is more stationary and there is no obvious trend. In this paper, we utilize the Augmented dickey-Fuller test with drift option to test whether there is a unit root in the time series.

3.1 Unempolyment rate

Graph 1. The graph of Unempolyment rate.



Overall, there were two periods of unusual unemployment rates (2000-2003 and 2008-2010), those high unemployment rates can be attributed to the dot-com bubble and the financial tsunami crisis. The line comes back to normal value after two economics crisis and there is no obvious trend. The next step is that we use ADF test to confirm whether there is a unit root. Depends on our consequence, we reject the null hypothesis under 0.05 significant level. Therefore, there is no unit root in this variable and we do not adjust our data with further transformation.

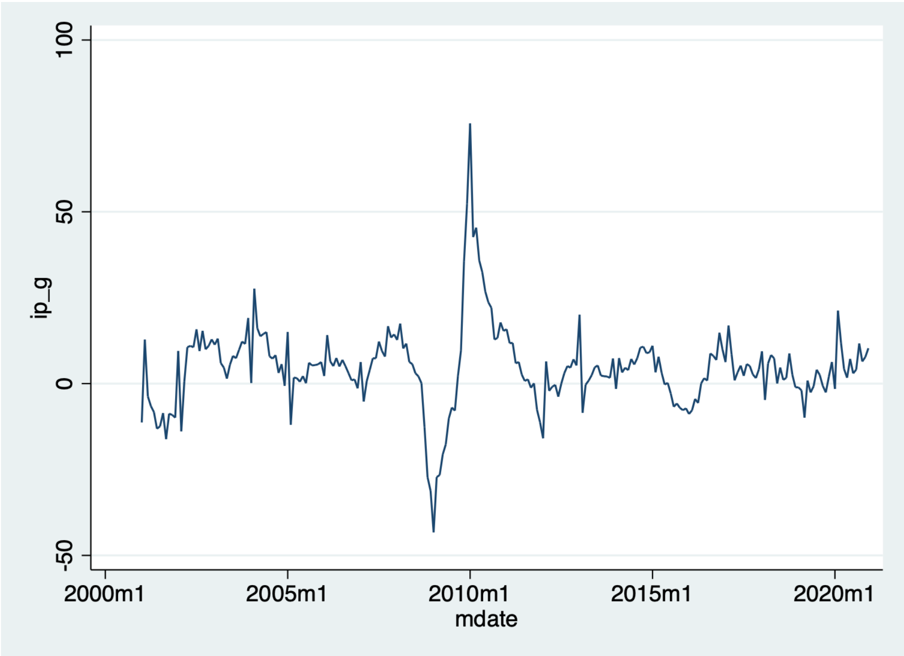
```
. dfuller unrate, drift

Dickey-Fuller test for unit root                     Number of obs   =       239

                               ----------- Z(t) has t-distribution -----------
                  Test         1% Critical       5% Critical      10% Critical
               Statistic           Value             Value             Value
------------------------------------------------------------------------------
 Z(t)             -1.679            -2.342            -1.651            -1.285
------------------------------------------------------------------------------
p-value for Z(t) = 0.0472
```

3.2 Annual growth rate of industrial production index

Graph 2. The graph of Annual growth rate of industrial production index.



Although the annual growth rate of industrial production index was relatively stable, it also had fluctuated greatly from 2008 to 2010, which was due to the financial tsunami as mentioned above. However, it still presented a white noise pattern, on the whole. Therefore, we firstly assume that the time series was stable, from the results of ADF test, it verifies our assumption that there is no unit root in our main independent variable.

```
. dfuller ip_g, drift

Dickey-Fuller test for unit root                     Number of obs   =       239

                               ----------- Z(t) has t-distribution -----------
                  Test         1% Critical       5% Critical      10% Critical
               Statistic           Value             Value             Value
------------------------------------------------------------------------------
 Z(t)             -5.310            -2.342            -1.651            -1.285
------------------------------------------------------------------------------
p-value for Z(t) = 0.0000
```

4 Model and Results

4.1 VAR model

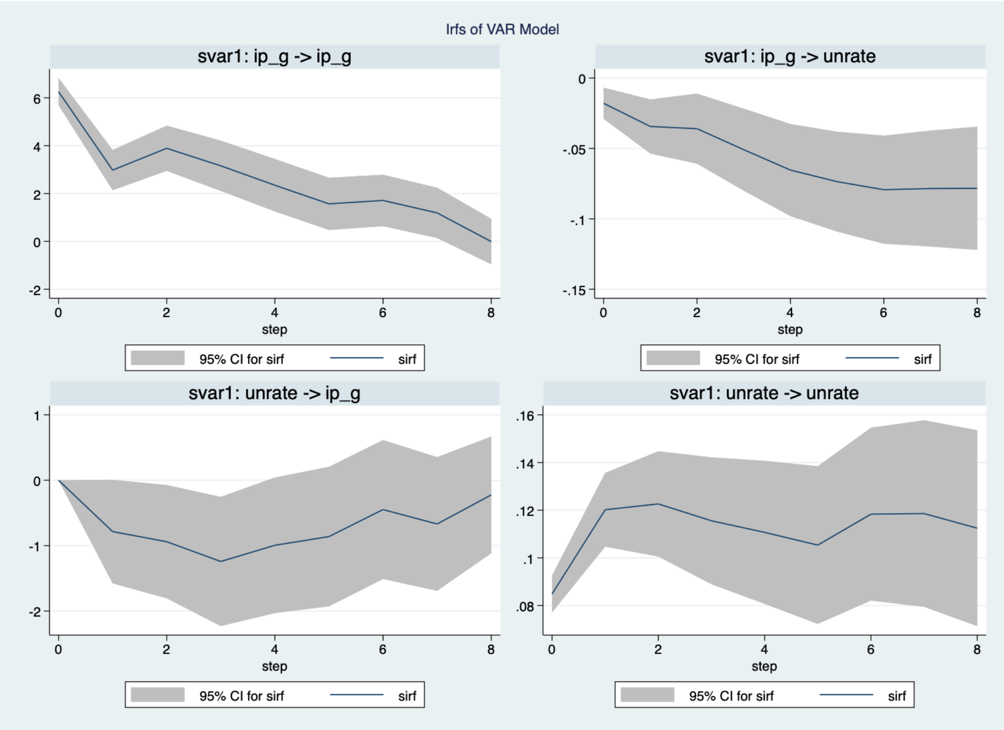
In this section, we would like to use VAR model with short run restrictions to graph the IRFs to examine how the shock influence those variables. In the paper, the order of this model is industrial production annual growth rate index first, then unemployment rate. The structural VAR model by matrix could be written as,

$$\text{VAR} = \begin{bmatrix} 1 & -\beta_{12} \\ -\beta_{21} & 1 \end{bmatrix} \begin{bmatrix} ip\_g_t \\ un_t \end{bmatrix} = \begin{bmatrix} \gamma_{11} & \gamma_{12} \\ \gamma_{21} & \gamma_{22} \end{bmatrix} \begin{bmatrix} ip\_g_{t-1} \\ un_{t-1} \end{bmatrix} + \begin{bmatrix} \varepsilon_{1\mathrm{t}} \\ \varepsilon_{2\mathrm{t}} \end{bmatrix}$$

where we define that $\varepsilon_{1\mathrm{t}}$ is supply shock and $\varepsilon_{2\mathrm{t}}$ is demand shock. Finally, according to our model $Bx_t = \Gamma_0 + \Gamma_1 x_{t-1} + \varepsilon_t$, we could have the reduced form model by rewritten our equation, $x_t = B^{-1}\Gamma_0 + B^{-1}\Gamma_1 x_{t-1} + B^{-1}\varepsilon_t$, then use Cholesky decomposition method with lower triangular matrix (β_{12}=0) to identify our shocks.

In the beginning, we have to set short run restriction to meet our assumption for identification requirement. We may expect demand shock is zero contemporaneous effect on annual growth rate of industrial production, in other words, industrial production annual growth rate of industrial production may not be reflected immediately by unemployment rate.

Graph3. IRFs in short run restriction



According to the graph1, we could find that the supply shock has positive effect on annual growth rate of industrial production but decrease in the long run. However, supply shock has negative effect on unemployment rate in either short run or long run, which is fitted our expectation. On the other hand, demand shock has negative effect in the short run but converge to zero in the long run. Lastly, demand shock has slightly positive and stable effect on unemployment rate. Note that even we use different lags (8/12/16/24), they all have the same direction and tendency.

4.2 Pseudo out-of-sample forecasting

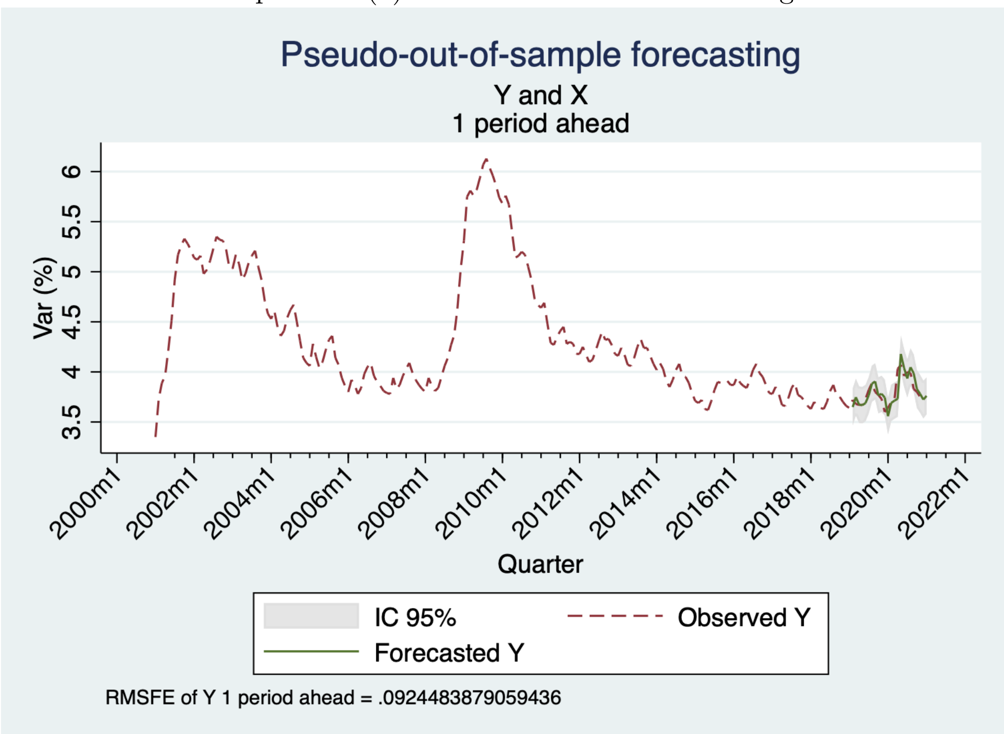
In this section, our goal is to find out which model has the best power of prediction. Therefore, we set in-sample from 2001m01 to 2018m12, and out-of-sample is from 2019m01 to 2020m12. In the beginning, we use AR model as our benchmark and baseline, then we also choose ADL model, which adding our leading indicator, annual growth rate of industrial production, to forecast unemployment rate. Finally, the criteria from AIC/BIC/RMSFE could help us decide which model is our "best" model.

4.2.1 AR model Selection

According to the results from STATA, we using the Pseudo-out-of forecasting to estimate our prediction model. By different criteria, we could get the different "best" model. AR(2) model minimize the BIC (-425.145), AR(7) model minimize the AIC (-440.305), and AR(9) model have the smallest RMSFE (0.09245). Although we lead to have fewer lag in our model, in order to have the best prediction power, we finally decide to choose the AR(9) model with one period ahead which have the minimum RMSFE.

The graph4 shows the results of forecasting, the red line is our observed unemployment rate (Y) and the green line is forecasted unemployment rate, and we also have the confidence interval in the period of out-of-sample(2019m01-2020m12). The forecasted Y compare with observed Y, it almost captures the tendency/direction of unemployment rate.

Graph4. AR(9) model with POOS forecasting.

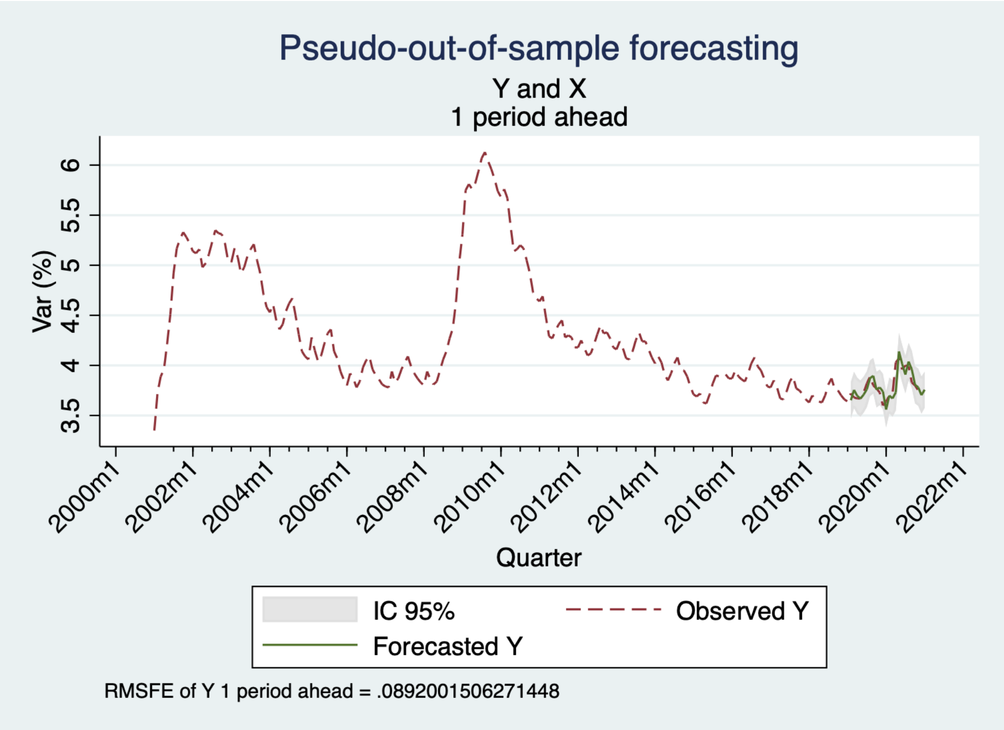


4.2.2 ADL model Selection

For the purpose to compare our model, we also estimate the ADL model by POOS forecasting. We utilize annual growth rate of industrial production to help us predict unemployment rate. Depends on the results of STATA, ADL(2,1) model minimize the BIC (-428.334), ADL(10,3) model minimize the AIC (-446.791), and ADL(9,4) model have the smallest RMSFE (0.08920).

The graph 5 presents the same logical as graph 4, however, we find that the RMSFE of ADL(9,4) model (0.08920) is less than RMSFE of AR(9) model (0.09245). This means that ADL(9,4) model could help us more precise and exact forecast unemployment rate, although it is hard to distinguish the difference from graph4 and graph5.

Graph5. ADR(9,4) model with POOS forecasting.



4.2.3 AR model vs ADL model

Table1. Comparing AR(9) and ADL(9,4) with h=1 to h=4 .

RMSFE	h=1	h=2	h=3	h=4
AR(9)	0.09245	0.09327	0.09516	0.09756
ADL(9,4)	0.08920	0.09011	0.09156	0.09378

Table1 shows the different ahead forecasting from h=1 to h=4 in order to further compare the power of prediction between AR(9) model and ADL(9,4) model. The consequence indicate that ADL(9,4) always have smaller RMSFE in different ahead forecast. This results verify our research question again, namely, annual growth rate of industrial production is a reliable indicator for unemployment rate in Taiwan.

5 Conclusion

In conclusion, according to our above results, first and foremost, annual growth rate of industrial production as a leading indicator, is a reliable indicator to help us more exact and precise predict unemployment rate via ADL model with Pseudo out-of-sample forecasting. In addition, autoregressive model as a benchmark also have an exceptional predict capability. Finally, we choose ADL(9,4) model and AR(9) as our predict model in this research. On the other hand, VAR model and graph of IRFs give us some practical points to examine how supply and demand shock influence unemployment rate and growth rate of industrial production.

As we know that if we choose different period to do the Pseudo out-of-sample forecasting, it may have different results. Fortunately, while the world is facing the covid-19 crisis, Taiwan's economy has not suffered a major shock and fluctuations during the period of 2019 to 2020. This is because the government and the whole citizen have consciously strengthened health management, and major companies is also adjusted their strategies quickly. Therefore, our model uses recent two year to make an evaluation in the Pseudo out-of-sample, which is representative and effective. In other words, there is no distortion of our model due to the epidemic.

6 Reference

Montgomery, A., Zarnowitz, V., Tsay, R., Tiao, G. (1998). Forecasting the U.S. Unemployment Rate. Journal of the American Statistical Association, 93(442), 478-493. doi:10.2307/2670094

Anindya Banerjee, Massimiliano Marcellino, Are there any reliable leading indicators for US inflation and GDP growth?, International Journal of Forecasting, Volume 22, Issue 1, 2006, Pages 137-151, ISSN 0169-2070, https://doi.org/10.1016/j.ijforecast.2005.03.005.

Olivier Blanchard and Danny Quah, (1989), The Dynamic Effects of Aggregate Demand and Supply Disturbances, American Economic Review, 79, (4), 655-73

```
cd "/Users/mac/OneDrive - The University of Texas at Austin/學習小札/2020 UTAustin/2021 Sp_Time Series/TS_Final Project"

insheet using "/Users/mac/OneDrive - The University of Texas at Austin/學習小札/2020 UTAustin/2021 Sp_Time Series/TS_Final Project/TSdata.csv", clear

save "/Users/mac/OneDrive - The University of Texas at Austin/學習小札/2020 UTAustin/2021 Sp_Time Series/TS_Final Project/TSdata.dta", replace

save "/Users/mac/OneDrive - The University of Texas at Austin/學習小札/2020 UTAustin/2021 Sp_Time Series/TS_Final Project/TSdata2.dta", replace

*****
cd "/Users/mac/OneDrive - The University of Texas at Austin/學習小札/2020 UTAustin/2021 Sp_Time Series/TS_Final Project"

use "/Users/mac/OneDrive - The University of Texas at Austin/學習小札/2020 UTAustin/2021 Sp_Time Series/TS_Final Project/TSdata2.dta", replace

gen mdate = m(1997m1) + _n -1
tsset mdate, m
drop if mdate>m(2020m12)
drop if mdate<m(2001m01)

tsset mdate

****ip_g****
tsline ip_g
graph export ip_g.png, replace
dfuller ip_g
dfuller ip_g, drift
ac ip_g, lag(30)
graph export ac_ip_g_30.png, replace
pac ip_g
graph export pac_ip_g.png, replace
*****
*****ip****
dfuller ip
dfuller ip, trend
dfuller ip, drift
*****

***unrate***
tsline unrate
graph export unrate.png, replace

dfuller unrate
dfuller unrate, trend
dfuller unrate, drift
dfuller d.unrate

ac unrate, lag(30)
graph export ac_un_30.png, replace
pac unrate
graph export pac_un.png, replace

gen dunrate = d.unrate
tsline dunrate

dfuller unrate
dfuller d.unrate

*VAR
****short run restriction*****
matrix A2 = (1,0 \ .,1)
matrix B2 = (.,0 \ 0,.)

svar ip_g unrate, lags(1/8) aeq(A2) beq(B2)

irf create var2irf, step(24) set(var2,replace)
```

```
irf graph sirf, irf(var2irf) yline(0,lcolor(black))

irf graph sirf, impulse(ip_g) response(ip_g)
irf graph sirf, impulse(ip_g) response(unrate)
irf graph sirf, impulse(unrate) response(unrate)
irf graph sirf, impulse(unrate) response(ip_g)


svar ip_g unrate, lags(1/12) aeq(A2) beq(B2)
irf create svar1, set(myGraph1, replace)
irf cgraph (svar1 ip_g ip_g sirf) (svar1 ip_g unrate sirf)  ///
           (svar1 unrate ip_g sirf) (svar1 unrate unrate sirf),  ///
                title ("Irfs of VAR Model", size(vsmall))
graph export irf_srr_12lag.png, replace

****long run restriction*****
matrix C = (.,0 \ .,.) //no 1's on the diagonal
svar ip_g unrate, lags(1/8) lreq(C)
irf create lr, set(lrirf1) step(40) replace
*view IRF (not cumulative )
irf graph sirf, yline(0,lcolor(black)) xlabel(0(4)40) byopts(yrescale) set(lrirf1)
graph export irf_longinshort.png, replace

*create cumulative irf graphs for responses of output variable
use lrirf1.irf, clear
sort irfname impulse response step
gen csirf = sirf
by irfname impulse: replace csirf = sum(sirf) if response== "ip_g"
order irfname impulse response step sirf csirf
save lrirf3.irf, replace

irf set lrirf3.irf
irf graph csirf, yline(0,lcolor(black)) noci xlabel(0(4)40) byopts(yrescale)

*create cumulative irf graphs for responses of all variables
by irfname impulse: replace csirf = sum(sirf) if response != "ip_g"
save lrirf3.irf, replace

irf set lrirf3.irf
irf graph csirf, yline(0,lcolor(black)) noci xlabel(0(4)40) byopts(yrescale)
graph export cirf.png, replace


*AR/MA/ADL
*****AR/MA model for un*****
forvalues i=1/4 {
forvalues j=1/4 {
arima unrate, arima(`i',0,`j') vce(robust) nolog
estat ic
}
}


forvalues i=1/4 {
arima unrate, arima(`i',0,0) vce(robust) nolog
estat ic
}


arima unrate, arima(9,0,0) vce(robust) nolog
estat ic
arima unrate, arima(7,0,0) vce(robust) nolog
estat ic
arima unrate, arima(2,0,0) vce(robust) nolog
estat ic


***we may not consider MA model, we use AR(2) for unrate as benchmark***

*favorite model = AR(2)
arima unrate, arima(2,0,0) vce(robust) nolog
predict e, resid
```

```
gen e_2 = e^2
summarize e_2
scalar rmse_ar1 = sqrt(r(mean))
**method one
ac e
pac e
corrgram e
*there is  no significant
**method two
reg e, noconstant
estat bgodfrey
*we can not reject null hypothesis (no serial correlation)
**method three
wntestq e
*we reject the null hypothesis, so no white noise

***in-sample predict for 2021m01***
tsappend, add(12)
qui arima unrate, arima(9,0,0) vce(robust)
predict yhat1, xb
predict fvar1, mse
gen upper1=yhat1 + 1.96*sqrt(fvar1)
gen lower1=yhat1 - 1.96*sqrt(fvar1)
twoway (rarea lower1 upper1 mdate, bcolor(gs14) legend(lab(1 "IC 95%"))) (line yhat1
mdate) (line unrate mdate), title(Forecast Profile: AR2)

graph export X.png, replace

***include ADL model***
***out-of-sample as A3Q3****

****AR model****

// Generate variables
forvalues i = 1/10 {
*di `i' `j'
qui gen f_Y_X1`i' = .       //generate space for forecasts
qui gen stdf_Y_X1`i' = .    //generate space for forecasts sd
qui gen sqdiff1`i' = .         //generate space for sum of square differences

***pseudo-out-of-sample (POOS) model to estimate

// First estimate the model with data you want to use to generate the model parameters
forvalues p = `=tm(2019m01)'/`=tm(2020m12)' {
//di `p'
qui regress Y L(1/`i').Y if mdate<`p'

qui predict Temp_Y_hat, xb  // save predicted values of the regression
qui predict Temp_Y_rmsfe, stdf  // save standard deviation of the error of the regression

qui replace f_Y_X1`i' = Temp_Y_hat if mdate==`p'+1 // use the predicted values as in-
sample forecast

qui replace stdf_Y_X1`i' = Temp_Y_rmsfe if mdate==`p'+1 // use the s.d. of error of the
previous regression, as in-sample s.d.

qui replace sqdiff1`i' = (Y - Temp_Y_hat)^2 if mdate==`p'+1 // compute squared forecast
errors

qui drop Temp_Y_hat Temp_Y_rmsfe
}

***compute the information criteria for the i lag
qui estat ic  //Akaike's and Schwarz's Bayesian information criteria
mat IC = r(S) //save results
scalar BIC`i' = el(IC, 1, 6)  //define a matrix for BIC
scalar AIC`i' = el(IC, 1, 5)  //define a matrix for AIC

***compute the RMSFE for the i lag
qui egen meansqdiff`i' = mean(sqdiff1`i')
qui scalar RMSFE`i' = sqrt(meansqdiff`i')
```

```
qui drop meansqdiff`i'
}


***find the best criteria between all lags
// generate space for the minimum values of the criteria
scalar minBIC = .
scalar minAIC = .
scalar minRMSFE = .

// find the minimum values of the criteria
forvalues i = 1/10 {
forvalues j =1/10 {
if BIC`i' < minBIC {
scalar minBIC = BIC`i'
local minBICij ",`i'"
}
if AIC`i' < minAIC {
scalar minAIC = AIC`i'
local minAICij ",`i'"
}
if RMSFE`i' < minRMSFE {
scalar minRMSFE = RMSFE`i'
local minRMSFEij ",`i'"
}
}
}

// display your results
di "min BIC lag"`minBICij' " : " minBIC
di "min AIC lag"`minAICij' " : " minAIC
di "min RMSFE lag"`minRMSFEij' " : " minRMSFE

-

//do a relative MSFE test for AR(?) and AR(?) models
local T1 = tm(2019m01)
local T2 = tm(2020m12)
local h = 1
qui total(sqdiff19) if mdate <= `T2' & mdate >= `T1' + `h'
local MSFEAR9 = e(b)[1,1]
qui total(sqdiff19_4) if mdate <= `T2' & mdate >= `T1' + `h'
local MSFEADL9_4 = e(b)[1,1]
di (1/(`T2' - `T1' - `h' + 1)*`MSFEAR9')/(1/(`T2' - `T1' - `h' + 1)*`MSFEADL9_4')



***ADL model***
// Generate variables
forvalues i = 1/10 {
forvalues j = 1/10 {
*di `i' `j'
qui gen f_Y_X1`i'_`j' = .      //generate space for forecasts
qui gen stdf_Y_X1`i'_`j' = .   //generate space for forecasts sd
qui gen sqdiff1`i'_`j' = .    //generate space for sum of square differences

***pseudo-out-of-sample (POOS) model to estimate

// First estimate the model with data you want to use to generate the model parameters
forvalues p = `=tm(2019m01)'/`=tm(2020m12)' {
//di `p'
qui regress Y L(1/`i').Y L(1/`j').X if mdate <`p'

qui predict Temp_Y_hat, xb  // save predicted values of the regression
qui predict Temp_Y_rmsfe, stdf  // save standard deviation of the error of the regression

qui replace f_Y_X1`i'_`j' = Temp_Y_hat if mdate==`p'+1 // use the predicted values as in-
sample forecast

qui replace stdf_Y_X1`i'_`j' = Temp_Y_rmsfe if mdate==`p'+1 // use the s.d. of error of
the previous regression, as in-sample s.d.
```

```
qui replace sqdiff1`i'_`j' = (Y - Temp_Y_hat)^2 if mdate==`p'+1 // compute squared
forecast errors

qui drop Temp_Y_hat Temp_Y_rmsfe
}

***compute the information criteria for the i lag
qui estat ic  //Akaike’s and Schwarz’s Bayesian information criteria
mat IC = r(S) //save results
scalar BIC`i'_`j' = el(IC, 1, 6)  //define a matrix for BIC
scalar AIC`i'_`j' = el(IC, 1, 5)  //define a matrix for AIC

***compute the RMSFE for the i lag
qui egen meansqdiff`i'_`j' = mean(sqdiff1`i'_`j')
qui scalar RMSFE`i'_`j' = sqrt(meansqdiff`i'_`j')

qui drop meansqdiff`i'_`j'
}
}

***find the best criteria between all lags
// generate space for the minimum values of the criteria
scalar minBIC = .
scalar minAIC = .
scalar minRMSFE = .

// find the minimum values of the criteria
forvalues i = 1/10 {
forvalues j =1/10 {
if BIC`i'_`j' < minBIC {
scalar minBIC = BIC`i'_`j'
local minBICij ",`i' , `j'"
}
if AIC`i'_`j' < minAIC {
scalar minAIC = AIC`i'_`j'
local minAICij ",`i' , `j'"
}
if RMSFE`i'_`j' < minRMSFE {
scalar minRMSFE = RMSFE`i'_`j'
local minRMSFEij ",`i' , `j'"
}
}
}

// display your results
di "min BIC lag"`minBICij' " : " minBIC
di "min AIC lag"`minAICij' " : " minAIC
di "min RMSFE lag"`minRMSFEij' " : " minRMSFE
```

____

```
*************************記得改下面的數字跑之前然後上面的兩個也都要先跑 tm/total裡面/標題
*** relatiive MSFE test of AR(1) and ADL(3,1)
local T1 = tq(2006q2)
local T2 = tq(2020q4)
local h = 1
qui total(sqdiff1) if quarter <= `T2' & quarter >= `T1' + `h'
local MSFE_AR1 = e(b)[1,1]
qui total(sqdiff11_1) if quarter <= `T2' & quarter >= `T1' + `h'
local MSFE1_1 = e(b)[1,1]
qui total(sqdiff13_1) if quarter <= `T2' & quarter >= `T1' + `h'
local MSFE3_1 = e(b)[1,1]
di (1/(`T2' - `T1' - `h' + 1)*`MSFE_AR1')/(1/(`T2' - `T1' - `h' + 1)*`MSFE3_1')
// AR(1) fits better
*** relatiive MSFE test of AR(1) and ADL(1,1)
di (1/(`T2' - `T1' - `h' + 1)*`MSFE_AR1')/(1/(`T2' - `T1' - `h' + 1)*`MSFE1_1')
// AR(1) fits better

//do a relative MSFE test for ADL(1,1) and ADL(3,1) models
local T1 = tm(2019m01)
local T2 = tm(2020m12)
local h = 1
```

```
qui total(sqdiff11_1) if mdate <= `T2' & mdate >= `T1' + `h'
local MSFE1_1 = e(b)[1,1]
qui total(sqdiff13_1) if mdate <= `T2' & mdate >= `T1' + `h'
local MSFE3_1 = e(b)[1,1]
di (1/(`T2' - `T1' - `h' + 1)*`MSFE10_3')/(1/(`T2' - `T1' - `h' + 1)*`MSFE9_4')

*******graph*****
*** Now estimate the POOS model with the lags of the models that have the minimum
critiria values found above.
*****
***`p'+1 is for one period ahead forecast and you can add more periods changing the "+1"
to "+h"
*****
drop f_Y_X1 stdf_Y_X1 sqdiff1 meansqdiff1 fcastHigh fcastLow

qui gen f_Y_X1 = .
qui gen stdf_Y_X1 = .
qui gen sqdiff1 = .
forvalues p = `=tm(2019m01)'/`=tm(2020m12)' {

//qui regress Y L(1/1).Y L(1/1).X  if quarter<`p' arima unrate, arima(2,0,0) vce(robust)
qui regress Y L(1/9).Y if mdate<`p'

qui predict Temp_Y_hat, xb
qui predict Temp_Y_rmsfe, stdf

qui replace f_Y_X1 = Temp_Y_hat if mdate==`p'+2
qui replace stdf_Y_X1 = Temp_Y_rmsfe if mdate==`p'+2

qui replace sqdiff1 = (Y - Temp_Y_hat)^2 if mdate==`p'+2

qui drop Temp_Y_hat Temp_Y_rmsfe
}

***Compute the RMSFE for the POOS estimated model
qui egen meansqdiff1 = mean(sqdiff1)
qui scalar RMSFE_Y_X1 = sqrt(meansqdiff1)
di "RMSFE of Y and X 1 periods ahead =" %9.4f scalar(RMSFE_Y_X1)

***Compute the confidence intervals
qui gen fcastHigh = f_Y_X1 + 1.96*stdf_Y_X1
qui gen fcastLow = f_Y_X1 - 1.96*stdf_Y_X1

***graph
qui twoway (rarea fcastLow fcastHigh mdate, bcolor(gs14) legend(lab(1 "IC 95%"))) (tsline
Y f_Y_X1, lpattern(dash solid) legend(lab(2 "Observed Y") lab(3 "Forecasted Y"))),
title("Pseudo-out-of-sample forecasting") subtitle("Y and X" "2 period ahead")
ytitle("Var (%)") xtitle("Quarter") xlabel(#10, angle(45)) xmticks(#40) note("RMSFE of Y
2 period ahead = `=scalar(RMSFE_Y_X1)'") name(f_YX1_graph, replace)


graph export ar9_2h.png, replace
```