Eco 5316 Time Series Econometrics

Lecture 16 Vector Autoregression (VAR) Models

Motivation

- theoretical models are developed to shed light on interactions and dynamic relationship between variables
  - income, consumption
  - interest rates, investment
  - interest rates, inflation, output gap
  - interest rates, exchange rates
  - return of individual stocks, stock market index
- following similar goals, we will now move from univariate times series models to multivariate time series models

Intervention Analysis

- example: effect of installing metal detectors at airports (starting in January 1973) on number of aircraft hijackings

$$y_t = \phi_0 + \phi_1 y_{t-1} + \omega x_t + \varepsilon_t$$

  where $x_t = 0$ for $t < 1973M1$ and $x_t = 1$ for $t \geq 1973M1$
- this model can be rewritten as

$$y_t = \frac{\phi_0}{1-\phi_1 L} + \frac{\omega_0}{1-\phi_1 L} x_t + \frac{1}{1-\phi_1 L} \varepsilon_t$$

  or

$$y_t = \frac{\phi_0}{1-\phi_1} + \omega_0 \sum_{i=0}^{\infty} \phi_1^i x_{t-i} + \sum_{i=0}^{\infty} \phi_1^i \varepsilon_{t-i}$$

- immediate impact is given by ω_0, long term effect is $\omega_0/(1-\phi_1)$

Intervention Analysis

alternative ways how to model interventions

- **pulse function** - temporary intervention: $x_t = 1$ if $t = t_I$, and $x_t = 0$ otherwise
- **pure jump function** - if the intervention is permanent, implemented fast: $x_t = 1$ if $t \geq t_I$, and $x_t = 0$ otherwise
- **prolonged impulse function** - intervention in place for a limited time: $x_t = 1$ if $t \in [t_I, t_I + D]$, and $x_t = 0$ otherwise
- **gradually changing function** - intervention phased in, implemented gradually over time: for example $x_t = \min\{(t - t_I + 1)/4, 1\}$ if $t \geq t_I$, and $x_t = 0$ otherwise

Autoregressive Distributed Lag (ARDL) model

- going beyond deterministic $0/1$ dummy variable and allowing for some general exogenous variable x_t, which has effect of y_t that is distributed over time yields an **autoregressive distributed lag model**, ARDL(p, r)

$$\phi(L)y_t = \delta(L)x_t + \varepsilon_t$$

  where $\phi(L)$ is lag polynomial of order p, $\delta(L)$ is lag polynomial of order r
- immediate effect is δ_0, long-run effect cumulative effect is $\frac{\delta(1)}{\theta(1)}$
- crucial assumption: $\{x_t\}$ is exogenous, evolves independently of $\{y_t\}$

Autoregressive Distributed Lag (ARDL) model

autoregressive distributed lag model nests

- static regression model: $\phi(L) = 1$, $\delta(L) = \delta_0$
- autoregressive model AR(p): $\phi(L) = 1 - \phi_1 - \ldots - \phi_p$, $\delta(L) \equiv 0$
- distributed lag DL(r): $\phi(L) = 1$, $\delta(L) = \delta_0 + \delta_1 L + \ldots + \delta_r L^r$

Sims' Critique

- intervention analysis and ARDL model assume that there is no feedback from $\{y_t\}$ to $\{x_t\}$, thus $\{x_t\}$ is truly exogenous
- but such feedback is likely to exist for some policies - some policy variables are set with specific reference to the state of other variables in the system (e.g. Fed setting interest rate)
- Sims (1980): the proper way is then to estimate multivariate models in unrestricted reduced form, treating *all* variables as endogenous

Bivariate Structural VAR(1) Model

- as with ARMA models, we will assume that times series are weakly stationary
- suppose that weakly stationary time series $\{y_{1,t}\}$, $\{y_{2,t}\}$ follow

$$y_{1,t} = c_{0,1} - b_{0,12} y_{2,t} + b_{1,11} y_{1,t-1} + b_{1,12} y_{2,t-1} + \varepsilon_{1,t}$$
$$y_{2,t} = c_{0,2} - b_{0,21} y_{1,t} + b_{1,21} y_{1,t-1} + b_{1,22} y_{2,t-1} + \varepsilon_{2,t}$$

  or equivalently

$$\boldsymbol{B}_0 \mathbf{y}_t = \boldsymbol{c}_0 + \boldsymbol{B}_1 \boldsymbol{y}_{t-1} + \boldsymbol{\varepsilon}_t$$

  where

$$\boldsymbol{B}_0 = \begin{pmatrix} 1 & b_{0,12} \\ b_{0,21} & 1 \end{pmatrix} \quad \boldsymbol{B}_1 = \begin{pmatrix} b_{1,11} & b_{1,12} \\ b_{1,21} & b_{1,22} \end{pmatrix} \quad \boldsymbol{c}_0 = \begin{pmatrix} c_{0,1} \\ c_{0,2} \end{pmatrix} \quad \boldsymbol{\varepsilon}_t = \begin{pmatrix} \varepsilon_{1,t} \\ \varepsilon_{2,t} \end{pmatrix}$$

  and

$$E(\boldsymbol{\varepsilon}_t) = \mathbf{0} \qquad var(\boldsymbol{\varepsilon}_t) = \begin{pmatrix} \sigma^2_{\varepsilon_1} & 0 \\ 0 & \sigma^2_{\varepsilon_2} \end{pmatrix}$$

- can't estimate this by OLS since $y_{1,t}$ has a contemporaneous effect on $y_{2,t}$ and $y_{2,t}$ has a contemporaneous effect on $y_{1,t}$ - **endogeneity problem** - if regressors and error terms are correlated, OLS estimates are biased

Bivariate Reduced Form VAR(1)

- suppose that $\boldsymbol{y}_t = (y_{1,t}, y_{2,t})'$ follows

$$\boldsymbol{B}_0 \boldsymbol{y}_t = \boldsymbol{c}_0 + \boldsymbol{B}_1 \boldsymbol{y}_{t-1} + \varepsilon_t$$

  with $E(\varepsilon_t) = \mathbf{0}$, $var(\varepsilon_t) = \mathbf{\Sigma}_\varepsilon$

- premultiply by $\boldsymbol{B}_0^{-1}$ to obtain

$$\boldsymbol{y}_t = \boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{y}_{t-1} + \boldsymbol{e}_t$$

  where $\boldsymbol{c} = \boldsymbol{B}_0^{-1} \boldsymbol{c}_0$, $\boldsymbol{A}_1 = \boldsymbol{B}_0^{-1} \boldsymbol{B}_1$, $\boldsymbol{e}_t = \boldsymbol{B}_0^{-1} \varepsilon_t$, in addition also $E(\boldsymbol{e}_t) = \mathbf{0}$ and $var(\boldsymbol{e}_t) = \mathbf{\Sigma}_e = \boldsymbol{B}_0^{-1} \mathbf{\Sigma}_\varepsilon \boldsymbol{B}_0^{-1\prime}$

  this system can now be estimated equation by equation using standard OLS

- even though the innovations $\boldsymbol{e}_t$ may be contemporaneously correlated, OLS is efficient and equivalent to GLS since all equations have identical regressors
- on structural vs reduced form models: https://en.wikipedia.org/wiki/Reduced_form

Example: Bivariate Structural and Reduced Form VAR(1)

- suppose that $\boldsymbol{y}_t = (y_{1,t}, y_{2,t})'$ follows a structural VAR(1)

$$\boldsymbol{B}_0 \boldsymbol{y}_t = \boldsymbol{c}_0 + \boldsymbol{B}_1 \boldsymbol{y}_{t-1} + \boldsymbol{\varepsilon}_t$$

  with

$$\boldsymbol{B}_0 = \begin{pmatrix} 1 & 0 \\ -.5 & 1 \end{pmatrix} \quad \boldsymbol{B}_1 = \begin{pmatrix} .6 & .2 \\ -.1 & .5 \end{pmatrix} \quad \boldsymbol{c}_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \boldsymbol{\Sigma}_\varepsilon = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

- then we have

$$\boldsymbol{B}_0^{-1} = \begin{pmatrix} 1 & 0 \\ .5 & 1 \end{pmatrix}$$

  and thus the associated reduced form VAR(1) model for $\boldsymbol{y}_t$ is

$$\boldsymbol{y}_t = \boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{y}_{t-1} + \boldsymbol{e}_t$$

  with

$$\boldsymbol{A}_1 = \begin{pmatrix} .6 & .2 \\ .2 & .6 \end{pmatrix} \quad \boldsymbol{c} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \boldsymbol{\Sigma}_e = \begin{pmatrix} 2 & 1 \\ 1 & 2.5 \end{pmatrix}$$

- a simulated path of $\boldsymbol{y}_t$ with 100 observations is below in panel (i) - note that $\{y_{1,t}\}$ and $\{y_{2,t}\}$ tend to move in the same direction

Example: Bivariate Structural and Reduced Form VAR(1)

- alternatively, suppose that $\boldsymbol{y}_t = (y_{1,t}, y_{2,t})'$ follows a structural VAR(1)

$$\boldsymbol{B}_0 \boldsymbol{y}_t = \boldsymbol{c}_0 + \boldsymbol{B}_1 \boldsymbol{y}_{t-1} + \boldsymbol{\varepsilon}_t$$

with

$$\boldsymbol{B}_0 = \begin{pmatrix} 1 & 0 \\ .5 & 1 \end{pmatrix} \quad \boldsymbol{B}_1 = \begin{pmatrix} .6 & -.2 \\ .1 & .5 \end{pmatrix} \quad \boldsymbol{c}_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \boldsymbol{\Sigma}_\varepsilon = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

- then we have

$$\boldsymbol{B}_0^{-1} = \begin{pmatrix} 1 & 0 \\ -.5 & 1 \end{pmatrix}$$

and thus the associated reduced form VAR(1) model for $\boldsymbol{y}_t$ is

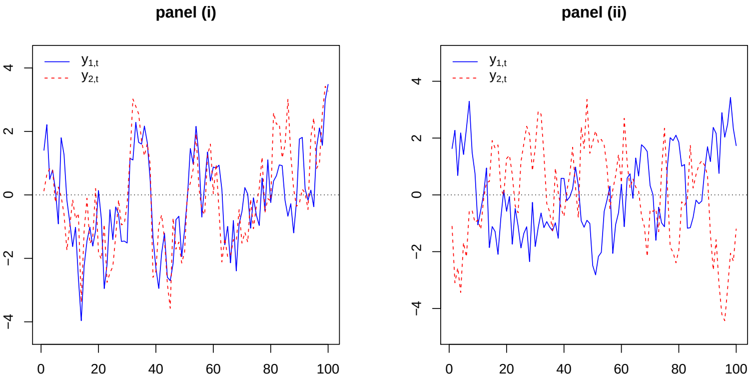
$$\boldsymbol{y}_t = \boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{y}_{t-1} + \boldsymbol{e}_t$$

with

$$\boldsymbol{A}_1 = \begin{pmatrix} .6 & -.2 \\ -.2 & .6 \end{pmatrix} \quad \boldsymbol{c} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \boldsymbol{\Sigma}_e = \begin{pmatrix} 2 & -1 \\ -1 & 2.5 \end{pmatrix}$$

- a simulated path of $\boldsymbol{y}_t$ with 100 observations is below in panel (ii) - note that $\{y_{1,t}\}$ and $\{y_{2,t}\}$ tend to move in opposite directions

Example: Bivariate Structural and Reduced Form VAR(1)



General Reduced Form VAR(p) Model

- multivariate order p VAR model: suppose that $\boldsymbol{y}_t = (y_{1,t}, y_{2,t}, \ldots, y_{k,t})'$ and that

$$\boldsymbol{y}_t = \boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{y}_{t-1} + \ldots + \boldsymbol{A}_p \boldsymbol{y}_{t-p} + \boldsymbol{e}_t$$

  where $\boldsymbol{c}$, $\boldsymbol{y}_t$, $\boldsymbol{e}_t$ are $k \times 1$ vectors, and $\boldsymbol{A}_i$ are $k \times k$ matrices
- using the lag operator we can write

$$\boldsymbol{A}(L)\boldsymbol{y}_t = \boldsymbol{c} + \boldsymbol{e}_t$$

  where $\boldsymbol{A}(L)$ is now a matrix polynomial in lag operator
- so VAR(p) vs AR(p) are kind of like Ice Ice Baby vs Under Pressure: https://www.youtube.com/watch?v=6TLo4Z_LWu4

Lag Selection

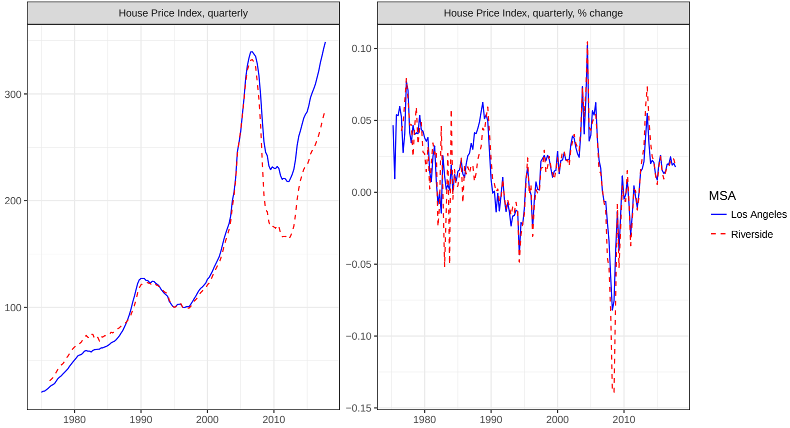
- VAR(p) has $k + pk^2$ parameters
- each additional lag introduces additional k^2 parameters, model thus become overparameterized quickly, and a lot of parameters will be insignificant
- information criteria - choose p to minimize AIC, SIC

Example

suppose we want to analyze joint dynamics of house prices in Los Angeles and Riverside, two MSAs about 60 miles apart

```r
library(tidyquant)
library(ggplot2)
library(ggfortify)

theme_set(theme_bw())

# obtain data on house price index for Los Angeles MSA and for Riverside MSA
hpi.tbl <-
    tq_get(c("ATNHPIUS31084Q","ATNHPIUS40140Q"), get = "economic.data",
           from  = "1940-01-01", to = "2017-12-31") %>%
    group_by(symbol) %>%
    rename(y = price) %>%
    mutate(dly = log(y) - lag(log(y)),
           msa = case_when(symbol == "ATNHPIUS31084Q" ~ "LA",
                           symbol == "ATNHPIUS40140Q" ~ "RI")) %>%
    ungroup() %>%
    select(msa, date, y, dly)
```

Example


House Price Index, quarterly
House Price Index, quarterly, % change
300
200
100
1980
1990
2000
2010
0.10
0.05
0.00
−0.05
−0.10
−0.15
MSA
Los Angeles
Riverside

Example

```r
# convert log change in house price index in Los Angeles MSA and for Riverside MSA into ts
library(timetk)
hpi.ts <-
    hpi.tbl %>%
    select(msa, date, dly) %>%
    spread(msa, dly) %>%
    filter(date >= "1976-07-01" & date <= "2012-10-01") %>%
    tk_ts(select = c("LA","RI"), start = 1976.5, frequency = 4)

library(vars)
VARselect(hpi.ts, lag.max = 8, type = "const")
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      3      1      1      3
##
## $criteria
##                  1          2          3          4          5          6
## AIC(n) -1.686e+01 -1.681e+01 -1.689e+01 -1.686e+01 -1.685e+01 -1.685e+01
## HQ(n)  -1.681e+01 -1.672e+01 -1.677e+01 -1.670e+01 -1.666e+01 -1.662e+01
## SC(n)  -1.673e+01 -1.660e+01 -1.659e+01 -1.647e+01 -1.638e+01 -1.630e+01
## FPE(n)  4.771e-08  5.001e-08  4.622e-08  4.785e-08  4.835e-08  4.830e-08
##                  7          8
## AIC(n) -1.680e+01 -1.676e+01
## HQ(n)  -1.655e+01 -1.647e+01
## SC(n)  -1.617e+01 -1.604e+01
## FPE(n)  5.046e-08  5.266e-08
```

Example

```r
var1 <- VAR(hpi.ts, p = 1, type = "const")
var1
```

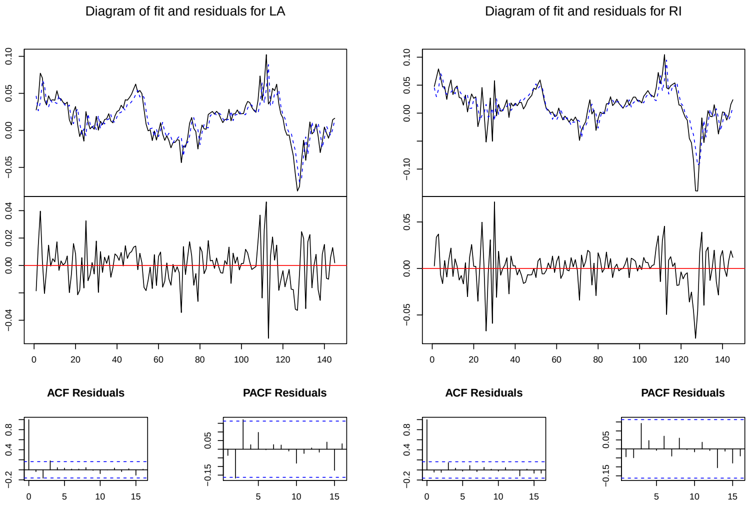
```
##
## VAR Estimation Results:
## =======================
##
## Estimated coefficients for equation LA:
## =======================================
## Call:
## LA = LA.l1 + RI.l1 + const
##
##       LA.l1       RI.l1       const
## 0.799352441 0.046017707 0.002210855
##
##
## Estimated coefficients for equation RI:
## =======================================
## Call:
## RI = LA.l1 + RI.l1 + const
##
##        LA.l1        RI.l1        const
##  0.680431636  0.257964563 -0.001699308
```

more detailed results (standard errors, t-statistics, . . .) can be obtained using

```r
summary(var1)
```

Example

```
plot(var1, nc = 4, lag.acf = 16, lag.pacf = 16)
```



Granger Causality

- test whether lags of one variable enter the equation for another variable
- variable j does not **Granger cause** variable i if all coefficients on lags of variable j in the equation for variable i are zero, that is

$$a_{1,ij} = a_{2,ij} = \ldots = a_{p,ij} = 0$$

- we thus test $H_0 : a_{1,ij} = \ldots = a_{p,ij} = 0$ against $H_A : \exists \ell \in \{1, \ldots, p\}$ such that $a_{\ell,ij} \neq 0$ using F-statistic and reject null if the statistic exceeds the critical value at the chosen level

- in addition: if the innovation to $y_{i,t}$ and the innovation to $y_{j,t}$ are correlated we say there is **instantaneous causality**

Granger Causality

```r
causality(var1, cause = "LA")
```

```
## $Granger
##
##  Granger causality H0: LA do not Granger-cause RI
##
## data:  VAR object var1
## F-Test = 24.285, df1 = 1, df2 = 284, p-value = 1.414e-06
##
##
## $Instant
##
##  H0: No instantaneous causality between: LA and RI
##
## data:  VAR object var1
## Chi-squared = 47.995, df = 1, p-value = 4.272e-12
```

```r
causality(var1, cause = "RI")
```

```
## $Granger
##
##  Granger causality H0: RI do not Granger-cause LA
##
## data:  VAR object var1
## F-Test = 0.30863, df1 = 1, df2 = 284, p-value = 0.579
##
##
## $Instant
##
##  H0: No instantaneous causality between: RI and LA
##
## data:  VAR object var1
## Chi-squared = 47.995, df = 1, p-value = 4.272e-12
```

Granger Causality

as the results of the Granger causality test for VAR(1) show

- we reject that $\Delta \log p_H^{LA}$ does not Granger cause $\Delta \log p_H^{RI}$ since the p-value in this case is 1.414×10^{-6}
- we can not reject that $\Delta \log p_H^{RI}$ does not Granger cause $\Delta \log p_H^{LA}$ since the p-value in this case is 0.579

similar conclusion can be made for the VAR(3) model suggested by the AIC

Granger Causality

```r
varp <- VAR(hpi.ts, ic = "AIC", lag.max = 8, type = "const")
causality(varp, cause = "LA")
```

```
## $Granger
##
##  Granger causality H0: LA do not Granger-cause RI
##
## data:  VAR object varp
## F-Test = 6.2946, df1 = 3, df2 = 272, p-value = 0.0003832
##
##
## $Instant
##
##  H0: No instantaneous causality between: LA and RI
##
## data:  VAR object varp
## Chi-squared = 49.615, df = 1, p-value = 1.871e-12
```

```r
causality(varp, cause = "RI")
```

```
## $Granger
##
##  Granger causality H0: RI do not Granger-cause LA
##
## data:  VAR object varp
## F-Test = 1.0448, df1 = 3, df2 = 272, p-value = 0.3732
##
##
## $Instant
##
##  H0: No instantaneous causality between: RI and LA
##
## data:  VAR object varp
## Chi-squared = 49.615, df = 1, p-value = 1.871e-12
```

Granger Causality

based on the results of the Granger causality test we thus remove the lags of $\Delta \log p^{RI}_{H,t}$ from the equation for $\Delta \log p^{LA}_{H,t}$

```
# define a  matrix with restictions
mat.r <- matrix(1, nrow = 2, ncol = 7)
mat.r[1, c(2,4,6)] <- 0

# estimate a restricted VAR
varp.r <- restrict(varp, method = "manual", resmat = mat.r)
varp.r
```

```
##
## VAR Estimation Results:
## =======================
##
## Estimated coefficients for equation LA:
## =======================================
## Call:
## LA = LA.l1 + LA.l2 + LA.l3 + const
##
##       LA.l1       LA.l2       LA.l3       const
##  0.80487880 -0.13866197  0.22324929  0.00134873
##
##
## Estimated coefficients for equation RI:
## =======================================
## Call:
## RI = LA.l1 + RI.l1 + LA.l2 + RI.l2 + LA.l3 + RI.l3 + const
##
##        LA.l1        RI.l1        LA.l2        RI.l2        LA.l3        RI.l3
##  0.682079767  0.201665973 -0.112882054  0.040154024  0.016615096  0.121347370
##        const
## -0.001814528
```

Forecasting

recall: under quadratic loss function, the forecast is the conditional mean

for simplicity, we will analyze the VAR(1) case but the method can be generalized to VAR(p) model in a straightforward way

- one step ahead forecast

$$\boldsymbol{\mu}_{t+1|t} = E_t \boldsymbol{y}_{t+1} = \boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{y}_t$$

  and forecast error

$$\boldsymbol{y}_{t+1} - \boldsymbol{\mu}_{t+1|t} = \left(\boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{y}_t + \boldsymbol{e}_{t+1}\right) - \left(\boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{y}_t\right) = \boldsymbol{e}_{t+1}$$

- two step ahead

$$\boldsymbol{\mu}_{t+2|t} = \boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{\mu}_{t+1|t}$$

  and forecast error

$$\boldsymbol{y}_{t+2} - \boldsymbol{\mu}_{t+2|t} = \left(\boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{y}_{t+1} + \boldsymbol{e}_{t+2}\right) - \left(\boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{\mu}_{t+1|t}\right) = \boldsymbol{A}_1 \boldsymbol{e}_{t+1} + \boldsymbol{e}_{t+2}$$

- in general, h step ahead forecast

$$\boldsymbol{\mu}_{t+h|t} = \boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{\mu}_{t+h-1|t}$$

  and forecast error

$$\boldsymbol{y}_{t+h} - \boldsymbol{\mu}_{t+h|t} = \boldsymbol{A}_1^{h-1} \boldsymbol{e}_{t+1} + \ldots + \boldsymbol{A}_1 \boldsymbol{e}_{t+h-1} + \boldsymbol{e}_{t+h}$$

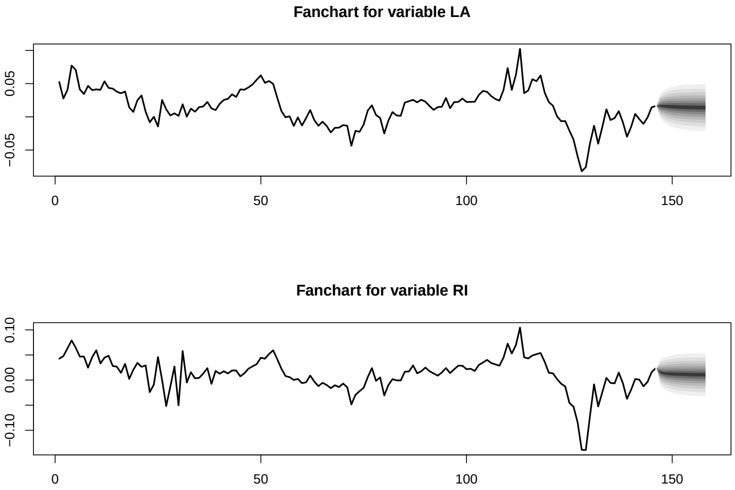
Forecasting

```
# construct 1 to 12 quarter ahead forecast and its 90% confidence interval
var1.f <- predict(var1, n.ahead = 12, ci = 0.9)
var1.f
```

```
## $LA
##              fcst        lower      upper         CI
##  [1,] 0.01634432 -0.008226675 0.04091532 0.02457100
##  [2,] 0.01599298 -0.016179902 0.04816586 0.03217288
##  [3,] 0.01561348 -0.021092418 0.05231938 0.03670590
##  [4,] 0.01527368 -0.024398088 0.05494544 0.03967176
##  [5,] 0.01498077 -0.026711930 0.05667347 0.04169270
##  [6,] 0.01473050 -0.028370017 0.05783102 0.04310052
##  [7,] 0.01451712 -0.029577321 0.05861156 0.04409444
##  [8,] 0.01433527 -0.030466940 0.05913749 0.04480221
##  [9,] 0.01418032 -0.031128807 0.05948946 0.04530913
## [10,] 0.01404830 -0.031625332 0.05972193 0.04567363
## [11,] 0.01393580 -0.032000641 0.05987225 0.04593645
## [12,] 0.01383995 -0.032286359 0.05996627 0.04612631
##
## $RI
##              fcst       lower      upper         CI
##  [1,] 0.01558637 -0.01835478 0.04952751 0.03394115
##  [2,] 0.01344262 -0.02795927 0.05484451 0.04140189
##  [3,] 0.01265054 -0.03339539 0.05869647 0.04604593
##  [4,] 0.01218799 -0.03697150 0.06134747 0.04915949
##  [5,] 0.01183745 -0.03946773 0.06314264 0.05130518
##  [6,] 0.01154773 -0.04126127 0.06435672 0.05280900
##  [7,] 0.01130270 -0.04257189 0.06517728 0.05387458
##  [8,] 0.01109429 -0.04354091 0.06572950 0.05463520
##  [9,] 0.01091680 -0.04426405 0.06609765 0.05518085
## [10,] 0.01076558 -0.04480805 0.06633921 0.05557363
## [11,] 0.01063674 -0.04522033 0.06649380 0.05585706
## [12,] 0.01052696 -0.04553498 0.06658889 0.05606194
```

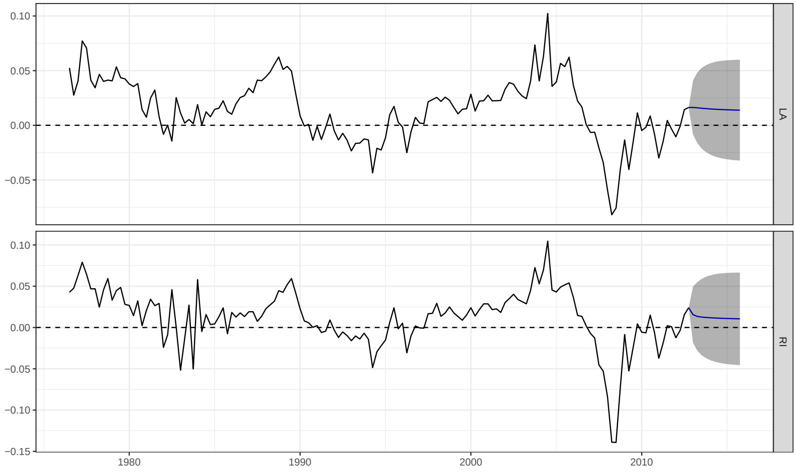
Forecasting

```r
# fanchart - by default the step is 0.1
fanchart(var1.f, lwd = 2)
```



Forecasting

```r
library(ggfortify)
autoplot(var1.f) + geom_hline(yintercept = 0, linetype = "dashed") + theme_bw()
```



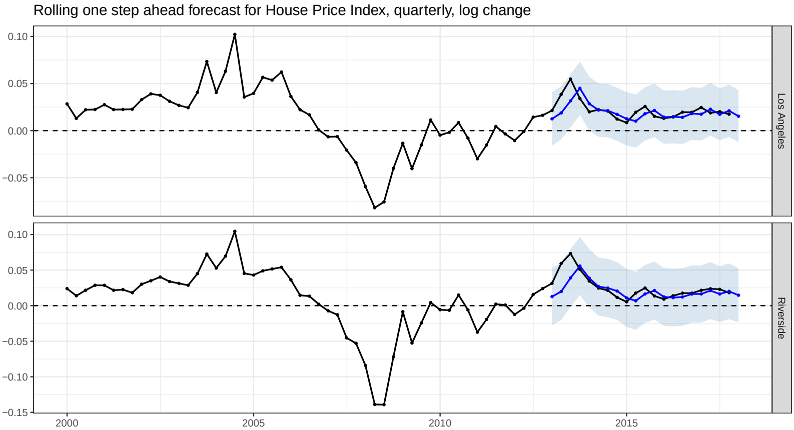
Forecasting

```r
library(tibbletime)
library(broom)

# estimate rolling VAR with window size = window.length
window.length <- nrow(hpi.ts)
# create rolling VAR function with rollify from tibbletime package
roll_VAR <- rollify(function(LA, RI) {
                        x <- cbind(LA, RI)
                        VAR(x, ic = "AIC", lag.max = 8, type = "const")
                        },
                    window = window.length, unlist = FALSE)


# estimate rolling VAR model, create 1 period ahead rolling forecasts
results <-
    hpi.tbl %>%
    dplyr::select(msa, date, dly) %>%
    spread(msa, dly) %>%
    filter(date >= "1976-07-01") %>%
    as_tbl_time(index = date) %>%
    mutate(VAR.model = roll_VAR(LA,RI)) %>%
    filter(!is.na(VAR.model)) %>%
    mutate(VAR.coefs = map(VAR.model, (. %$% map(varresult, tidy, conf.int = TRUE) %>%
                                            map(as.tibble) %>%
                                            bind_rows(.id = "msa"))),
           VAR.f = map(VAR.model, (. %>% predict(n.ahead = 1) %$%
                                        fcst %>%
                                        map(as.tibble) %>%
                                        bind_rows(.id = "msa"))))
```

Forecasting

```
# extract 1 period ahead rolling forecasts
tbl.f.1.rol <-
    bind_rows(
        # actual data
        hpi.tbl %>%
            dplyr::select(date, msa, dly) %>%
            rename(value = dly) %>%
            mutate(key = "actual"),
        # forecasts
        results %>%
            dplyr::select(date, VAR.f) %>%
            unnest(VAR.f) %>%
            rename(value = fcst) %>%
            mutate(key = "forecast",
                   date = date %m+% months(3))
    ) %>%
    arrange(date, msa)

# plot the 1 period ahead rolling forecasts
tbl.f.1.rol %>%
    dplyr::filter(date >= "2000-01-01") %>%
    mutate(msa.f = factor(msa, labels = c("Los Angeles","Riverside"))) %>%
    ggplot(aes(x = date, y = value, col = key, group = key)) +
        geom_ribbon(aes(ymin = lower, ymax = upper), color = NA, fill = "steelblue", alpha = 0.2) +
        geom_line(size = 0.7) +
        geom_point(size = 0.7) +
        geom_hline(yintercept = 0, linetype = "dashed") +
        scale_color_manual(values = c("black","blue")) +
        labs(x = "", y = "",
             title = "Rolling one step ahead forecast for House Price Index, quarterly, log change") +
        facet_grid(msa.f ~ ., scales = "free_y") +
        theme(legend.position = "none")
```

Forecasting



Innovations Accounting

two important tools used to examine relationships among economic variables

- **impulse-response analysis**: the goal is to track the response of a variable y_i to a one time shock $\varepsilon_{j,t}$
- **forecast error variance decomposition**: the goal is to find the fraction of the overall fluctuations in y_i that is due to shock $\varepsilon_{j,t}$

as with forecasting, we will analyze the VAR(1) case but the method can be generalized for VAR(p) model

Impulse-Response Functions (IRF)

- goal: obtain **response of y_i over time to a one time increase in $\varepsilon_{j,t}$**
- IRFs are constructed using vector moving average (VMA) representation
- consider a reduced form VAR(1)

  $$\boldsymbol{y}_t = \boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{y}_{t-1} + \boldsymbol{e}_t$$

  by repeated substitutions

  $$\boldsymbol{y}_t = \boldsymbol{c} + \boldsymbol{A}_1 \left(\boldsymbol{c} + \boldsymbol{A}_1 \boldsymbol{y}_{t-2} + \boldsymbol{e}_{t-1} \right) + \boldsymbol{e}_t = \ldots = \boldsymbol{\mu} + \sum_{h=0}^{\infty} \boldsymbol{A}_1^h \boldsymbol{e}_{t-h}$$

  where $\boldsymbol{\mu} = \sum_{h=0}^{\infty} \boldsymbol{A}_1^h \boldsymbol{c}$ is the long run average of $\boldsymbol{y}$
- since $\boldsymbol{e}_t = \boldsymbol{B}_0^{-1} \varepsilon_t$ we can define $\boldsymbol{\Psi}_h = \boldsymbol{A}_1^h \boldsymbol{B}_0^{-1}$ and write

  $$\boldsymbol{y}_t - \boldsymbol{\mu} = \sum_{h=0}^{\infty} \boldsymbol{\Psi}_h \boldsymbol{\varepsilon}_{t-h}$$

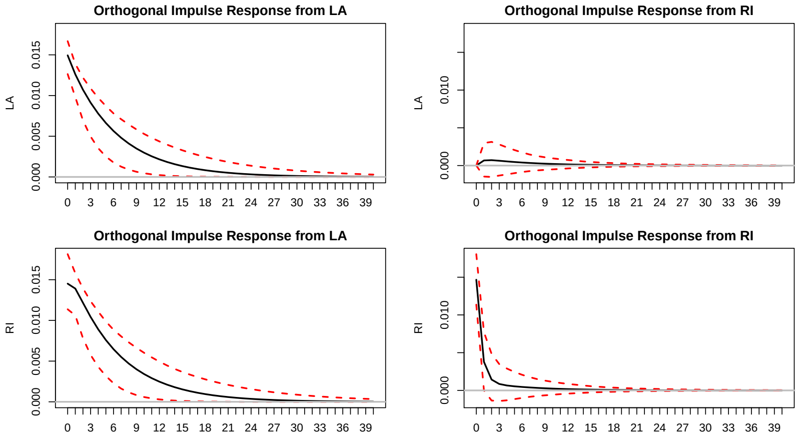
- $\psi_{h,ij}$ i.e. row i column j element of matrix $\boldsymbol{\Psi}_h$ shows
  - the impact of a unit increase in $\varepsilon_{j,t-h}$ on $y_{i,t}$
  - the impact of a unit increase in $\varepsilon_{j,t}$ on $y_{i,t+h}$
- a plot of $\psi_{h,ij}$ as function of h is the **impulse-response function** plot
- confidence intervals for IRFs - parameters of the VAR model are unknown, estimated, there is thus uncertainty regarding the response of $\boldsymbol{y}$ to changes in ε

Impulse-Response Functions (IRF)

- the IRF components $\Psi_{h,ij}$ show $\frac{\partial y_{i,t+h}}{\partial \varepsilon_{j,t}}$ that is, how y_i responds over time to a one time increase in ε_j
- when constructing the IRF the size of the one time shock in ε_j is taken to be one standard deviation σ_{ϵ_j}
- the plot of the IRF for y_i shows the effect ε_j of as deviations of y_i from its long run equilibrium μ_i
- because all variables in VAR are weakly stationary, deviations eventually converge to 0 as the system converges back to the long run equilibrium given by $\boldsymbol{\mu}$

Impulse-Response Functions (IRF)

```
var1.irfs <- irf(var1, n.ahead = 40)
par(mfcol = c(2,2), cex = 0.8, mar = c(3,4,2,2))
plot(var1.irfs, plot.type = "single", lwd = 2)
```



Impulse-Response Functions (IRF)

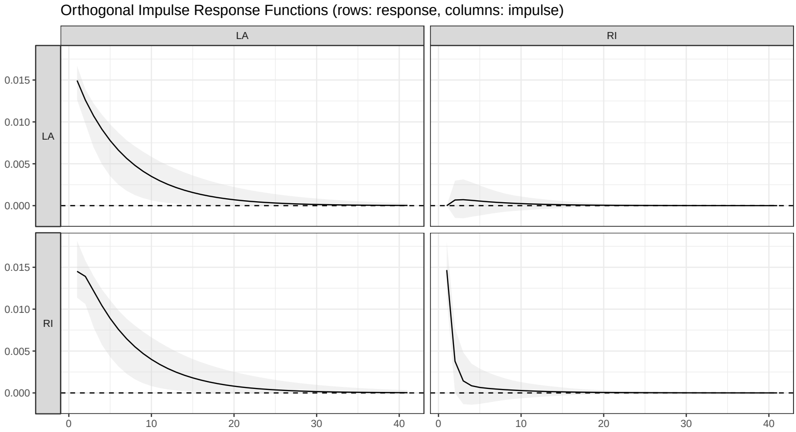
IRFs for VAR(1) model of house prices in Los Angeles and Riverside show that

- $\Delta \log p_H^{LA}$ and $\Delta \log p_H^{LA}$ react to ε_{LA} in similar way and with similar magnitude - on impact the price increases by about 1.5% in both markets, this is followed by a gradual decline but the effects are significant even after 16 quarters (4 years)
- $\Delta \log p_H^{LA}$ increases only very little in response to ε_{RI} and this response is not statistically significant since the 95% confidence interval contains 0
- $\Delta \log p_H^{LA}$ increases more in response to ε_{RI} but this increase is short lived and the variable converges back within about 3 quarters
- the demand and supply shock in the Los Angeles market thus clearly dominate the dynamics and the variability of home prices in both markets, their transmission into the Riverside market is quite strong

Impulse-Response Functions (IRF)

```
# arrange IRF data into a tibble to be used with ggplot
var1.irfs.tbl <-
    var1.irfs[1:3] %>%
    modify_depth(2, as.tibble) %>%
    modify_depth(1, bind_rows, .id = "impulse") %>%
    map_df(bind_rows, .id = "key") %>%
    gather(response, value, -key, -impulse) %>%
    group_by(key, impulse, response) %>%
    mutate(lag = row_number()) %>%
    ungroup() %>%
    spread(key, value)

# plot IRFs using ggplot
ggplot(data = var1.irfs.tbl, aes(x = lag, y = irf)) +
    geom_ribbon(aes(x = lag, ymin = Lower, ymax = Upper), fill = "lightgray", alpha = .3) +
    geom_line() +
    geom_hline(yintercept = 0, linetype = "dashed") +
    labs(x = "", y = "",
         title = "Orthogonal Impulse Response Functions (rows: response, columns: impulse)") +
    facet_grid(response ~ impulse, switch = "y") +
    theme(strip.text.y = element_text(angle = 0))
```

Impulse-Response Functions (IRF)



Forecast Error Variance Decomposition (FEVD)

- goal: find **fraction of overall fluctuations in y_i that is due to shock $\varepsilon_{j,t}$**
- we obtained that the h step ahead forecast error of the VAR(1) model is

$$\boldsymbol{y}_{t+h} - \boldsymbol{\mu}_{t+h|t} = \boldsymbol{A}_1^{h-1}\boldsymbol{e}_{t+1} + \ldots + \boldsymbol{A}_1\boldsymbol{e}_{t+h-1} + \boldsymbol{e}_{t+h}$$

- like with IRFs, since $\boldsymbol{e}_t = \boldsymbol{B}_0^{-1}\boldsymbol{\varepsilon}_t$ we define $\boldsymbol{\Psi}_h = \boldsymbol{A}_1^h\boldsymbol{B}_0^{-1}$ and write

$$\boldsymbol{y}_{t+h} - \boldsymbol{\mu}_{t+h|t} = \boldsymbol{\Psi}_{h-1}\boldsymbol{\varepsilon}_{t+1} + \ldots + \boldsymbol{\Psi}_1\boldsymbol{\varepsilon}_{t+h-1} + \boldsymbol{\Psi}_0\boldsymbol{\varepsilon}_{t+h}$$

- thus for variable y_i the variance of h step ahead forecast error is given by

$$\sigma^2_{y_i,h} = var(y_{t+h} - \mu_{t+h|t}) = \sum_{j=1}^{k} \sigma^2_{\varepsilon_j} \sum_{\tau=0}^{h-1} \Psi^2_{\tau,ij}$$

  and the portion of $\sigma^2_{y_i,h}$ that is due to shock $\{\varepsilon_{j,t}\}$ is

$$\frac{\sigma^2_{\varepsilon_j} \sum_{\tau=0}^{h-1} \Psi^2_{\tau,ij}}{\sum_{j=1}^{k} \sigma^2_{\varepsilon_j} \sum_{\tau=0}^{h-1} \Psi^2_{\tau,ij}}$$

- this yields a decomposition of forecast error variance, and provides insight which shocks are behind the fluctuations of y_i

Forecast Error Variance Decomposition (FEVD)

```
var1.fevd <- fevd(var1, n.ahead = 40)
```

decomposition for $\Delta \log p_H^{LA}$

```
var1.fevd[[1]][c(1,4,8,40),]
```

```
##             LA          RI
## [1,] 1.0000000 0.000000000
## [2,] 0.9976450 0.002355005
## [3,] 0.9970820 0.002917960
## [4,] 0.9969241 0.003075916
```

thus at 1 quarter horizon fluctuations are *entirely* due to ε_{LA}, and at 40 quarters horizon (i.e. 10 years) more than 99% are due to ε_{LA} and less than 1% due to ε_{RI}

decomposition for $\Delta \log p_H^{RI}$

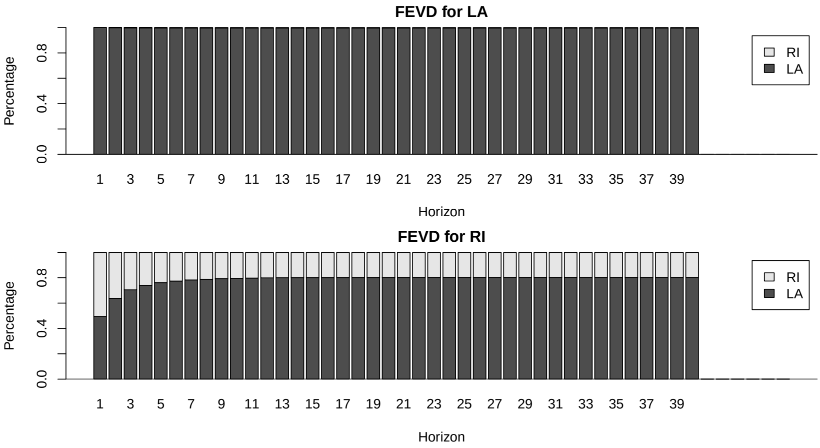
```
var1.fevd[[2]][c(1,4,8,40),]
```

```
##             LA        RI
## [1,] 0.4947726 0.5052274
## [2,] 0.7400088 0.2599912
## [3,] 0.7885300 0.2114700
## [4,] 0.8026250 0.1973750
```

thus at 1 quarter horizon fluctuations are about half and half due to ε_{LA} and ε_{LA}; at 40 quarters horizon about 80% are due to ε_{LA} and 20% due to ε_{RI}

Forecast Error Variance Decomposition (FEVD)

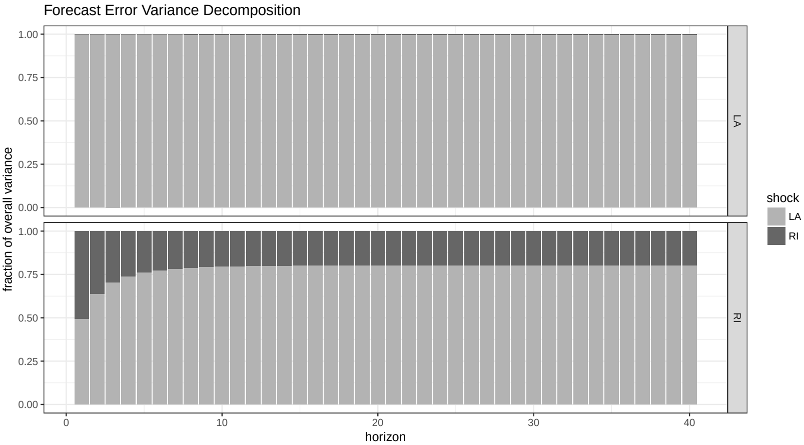
```r
plot(var1.fevd)
```



Forecast Error Variance Decomposition (FEVD)

```r
# arrange FEVD data into a tibble to be used with ggplot
var1.fevd.tbl <-
    var1.fevd %>%
    modify_depth(1, as.tibble) %>%
    map_df(bind_rows, .id = "variable") %>%
    gather(shock, value, -variable) %>%
    group_by(shock, variable) %>%
    mutate(horizon = row_number()) %>%
    ungroup()

# plot FEVD using ggplot
ggplot(data = var1.fevd.tbl, aes(x = horizon, y = value, fill = shock)) +
    geom_col(position = position_stack(reverse = TRUE)) +
    scale_fill_manual(values = c("gray70","gray40")) +
    labs(x = "horizon", y = "fraction of overall variance",
         title = "Forecast Error Variance Decomposition") +
    facet_grid(variable ~ .)
```

Forecast Error Variance Decomposition (FEVD)



Reduced Form Errors vs Structural Shocks

note that

- IRFs trace out the response of $\boldsymbol{y}_t$ to structural shocks ε_t, not reduced form errors $\boldsymbol{e}_t$
- FEVD gives the fraction of variance of $\boldsymbol{y}_t$ caused by different structural shocks ε_t, not reduced form errors $\boldsymbol{e}_t$
- since $\varepsilon_t = \boldsymbol{B}_0 \boldsymbol{e}_t$ to construct the IRFs and FEVD for a VAR(p) model we need to know $\boldsymbol{B}_0$, in addition to $\boldsymbol{A}_1, \ldots, \boldsymbol{A}_p$

Identification of Structural Shocks

Q: Is it possible to recover $\boldsymbol{c}_0$, $\{\boldsymbol{B}_i\}_{i=0}^{p}$ and $\boldsymbol{\Sigma}_\varepsilon$ from $\boldsymbol{c}$, $\{\boldsymbol{A}_i\}_{i=1}^{p}$ and $\boldsymbol{\Sigma}_e$?

A: Only if we are willing to impose additional restrictions.

Identification of Structural Shocks

Example: bivariate VAR(1)

- reduced form VAR(1) yields estimates of 9 parameters in $\boldsymbol{c}$, $\boldsymbol{A}_1$, $\boldsymbol{\Sigma}_e$

$$\begin{pmatrix} y_{1,t} \\ y_{2,t} \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} + \begin{pmatrix} a_{1,11} & a_{1,12} \\ a_{1,21} & a_{1,22} \end{pmatrix} \begin{pmatrix} y_{1,t-1} \\ y_{2,t-1} \end{pmatrix} + \begin{pmatrix} e_{1,t} \\ e_{2,t} \end{pmatrix} \qquad \boldsymbol{\Sigma}_e = \begin{pmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{pmatrix}$$

- we are trying to uncover $\boldsymbol{c}_0$, $\boldsymbol{B}_0$, $\boldsymbol{B}_1$, $\boldsymbol{\Sigma}_\varepsilon$ which contain 10 unknown values

$$\begin{pmatrix} 1 & b_{0,12} \\ b_{0,21} & 1 \end{pmatrix} \begin{pmatrix} y_{1,t} \\ y_{2,t} \end{pmatrix} = \begin{pmatrix} c_{0,1} \\ c_{0,2} \end{pmatrix} + \begin{pmatrix} b_{1,11} & b_{1,12} \\ b_{1,21} & b_{1,22} \end{pmatrix} \begin{pmatrix} y_{1,t-1} \\ y_{2,t-1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{1,t} \\ \varepsilon_{2,t} \end{pmatrix} \qquad \boldsymbol{\Sigma}_\epsilon = \begin{pmatrix} \sigma_{\varepsilon_1}^2 & 0 \\ 0 & \sigma_{\varepsilon_2}^2 \end{pmatrix}$$

- one additional restriction on parameters thus needs to be *imposed* in the VAR(1)
- one possible way to do this is the **Choleski decomposition**
  - impose $b_{0,12} = 0$ so that $y_{1,t}$ has contemporaneous effect on $y_{2,t}$, but $y_{2,t}$ does not have a contemporaneous effect on $y_{1,t}$
  - this also means that both $\varepsilon_{1,t}$ and $\varepsilon_{2,t}$ have a contemporaneous effect on $y_{2,t}$, but only $\varepsilon_{1,t}$ has an effect on $y_{1,t}$
  - this is how `vars` package constructs IRFs and FEVD shown on previous slides

Identification of Structural Shocks

general case, a VAR(p) model with k variables

- reduced form has $k + pk^2 + k(k+1)/2$ parameters
- structural form has $k + (p+1)k^2 + k$ parameters
- identification thus requires $k(k-1)/2$ additional restrictions
- Choleski decomposition: set elements of $\boldsymbol{B}_0$ above main diagonal equal zero
- ordering of variables in the VAR(p) model thus matters:
  $y_{i,t}$ is only affected by shocks $\varepsilon_{1,t}, \dots, \varepsilon_{i,t}$, remaining shocks $\varepsilon_{i+1,t}, \dots, \varepsilon_{k,t}$ have no contemporaneous effect on $y_{i,t}$ and will only affect $y_{i,t'}$ for $t' > t$ indirectly through their effect on $y_{i+1,t}, \dots, y_{k,t}$