

Spurius Granger-okság szimultán DAG szerkezetből

Balázs Frigyes, Gedeon Anna, Nagy Balázs

2025-12-30

Könyvtárak betöltése

```
library(mvtnorm)
library(vars)

## A szükséges csomag betöltődik: MASS
## A szükséges csomag betöltődik: strucchange
## A szükséges csomag betöltődik: zoo
##
## Kapcsolódás csomaghoz: 'zoo'
## The following objects are masked from 'package:base':
##
##   as.Date, as.Date.numeric
## A szükséges csomag betöltődik: sandwich
## A szükséges csomag betöltődik: urca
## A szükséges csomag betöltődik: lmtest
library(ggplot2)
```

Kétváltozós adatgeneráló folyamatok

Az adatgeneráló folyamat

$$X_t = (I - B_0)^{-1} B_1 X_{t-1} + E_t$$

Ahol a B_0 és B_1 mátrixokat különbözőképp variáljuk a következő szimulációk során. $E_t \sim N(0, \Sigma)$

Szimulációkhoz szükséges függvények

```
generate_sim_dag <- function(n, B0, B1, sigma, d = dim(B0)[1]){
  # őrai GY11_6-ből, átírva
  X <- matrix(0, nrow=n, ncol=d)
  E <- rmvnorm(n, sigma = sigma)
  for (t in 2:n){
    X[t,] <- solve(diag(d) - B0) %*% (B1 %*% X[t-1,] + E[t,])
  }
  res = as.data.frame(X)
  colnames(res) = paste0("x", 1:d)
  res
}
```

```

}

check_granger <- function(x, sig = 0.05){
  # órai GY11_6-ból, átírva
  varfit <- VAR(x, p = 1, type="const")

  tab <- summary(varfit$varresult$x1)$coefficients
  pval <- tab[grepl("x2", rownames(tab)), "Pr(>|t|)"]
  res <- list()
  res$pval <- pval
  res$reject <- pval < sig

  res
}

MC_sim <- function(n, B0, B1, sigma, nsim = 10){
  MC_res <- sapply(rep(n, nsim),
                    function(x){generate_sim_dag(x, B0, B1, sigma) |>
                                check_granger()})

  res <- list()
  res$reject <- sum(unlist(as.data.frame(MC_res)[2,])) / nsim
  res$list <- MC_res
  res
}

B1 <- diag(0.9,2)
B0 <- matrix(0,2,2)
sigma <- diag(1,2)

# MC_sim(200, B0, matrix(c(0.9,0,0.3,0.9),2,2), matrix(c(1,0.8,0.8,1),2,2), 100)
# generate_sim_dag(50, B0, matrix(c(0.9,0,0.3,0.9),2,2), sigma = matrix(c(1,0.8,0.8,1),2,2)) |> check_granger()
# generate_sim_dag(80, matrix(c(0,0,1,0),2), diag(0.9,2), diag(1,2)) |> VAR() |> summary()
# generate_sim_dag(80, matrix(c(0,0,2,0),2), matrix(c(0.9,0,-1,0.9),2), diag(1,2)) |> VAR() |> summary()
# generate_sim_dag(80, matrix(c(0,0,0,0),2), matrix(c(0.9,0,0,0.9),2), matrix(c(1,-.9,-.9,1),2)) |> VAR()

```

Mintaelemszám és azonnali kapcsolat erőssége

Azt vizsgáljuk, hogy mekkora elemszámok mellett milyen arányban érzékeljük az azonnali kapcsolatot Granger okságként.

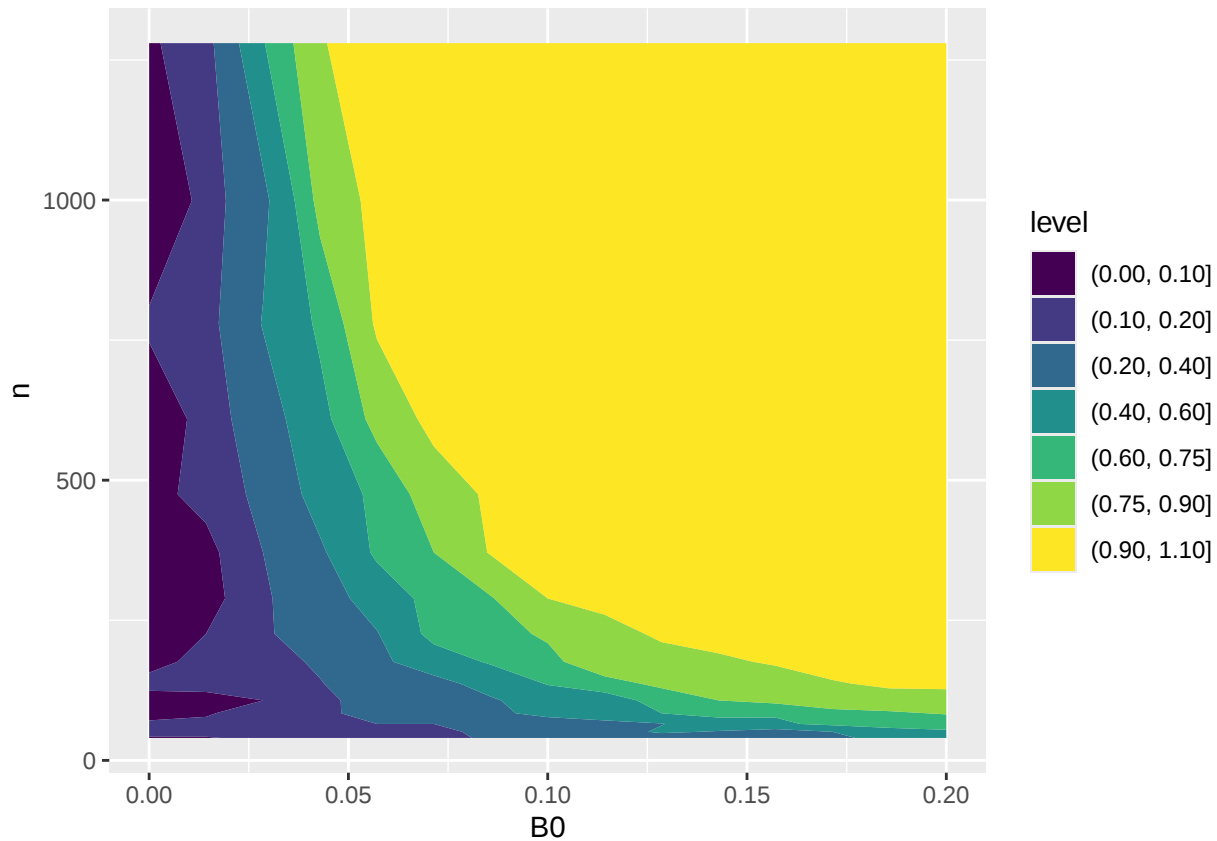
```

nsim = 100
B0s <- seq(0, 0.2, length.out = 15)
n_vec <- floor(10*2^seq(2,7, length.out = 15))
params <- expand.grid(B0s, n_vec)
res <- list()
for(i in 1:(nrow(params))){
  if(i%100==0) print(i)
  rej <- MC_sim(params[i,2], (function(x){matrix(c(0,0,x,0),2,2)})(params[i,1]),
                B1, sigma, nsim)$reject
  res[[i]] <- data.frame(B0 = params[i,1], n = params[i,2], reject = rej)
}

## [1] 100
## [1] 200

```

```
res_df <- do.call(rbind, res)
ggplot(res_df, aes(x=B0, y=n, z= reject))+
  geom_contour_filled(breaks = c(0,0.1,0.2,0.4,0.6,0.75,0.9, 1.1))
```

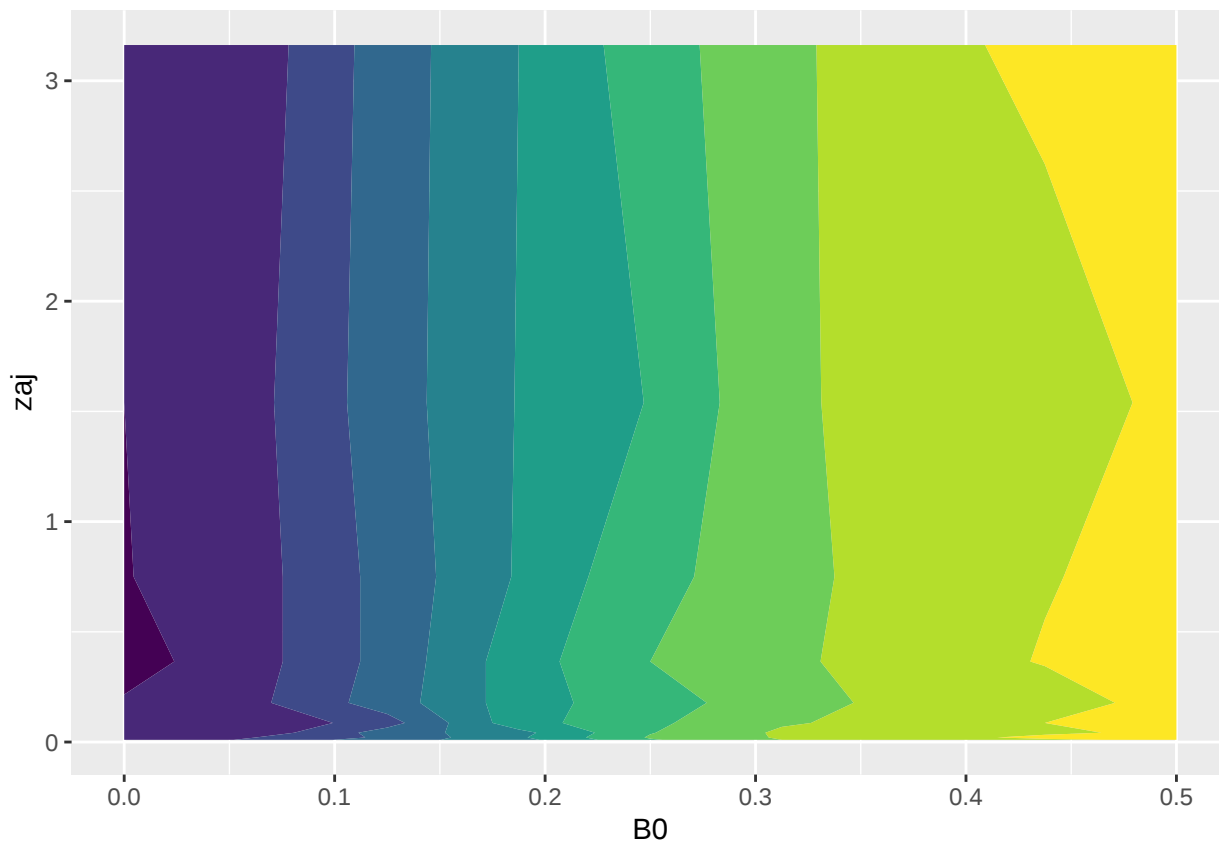


Kapcsolat erősségének és zaj méretének kapcsolata

Előzőhöz hasonlóan, de $n = 50$, és a zaj (E_t) méretén változtatunk

```
nsim = 200
B0s <- seq(0, 0.5, length.out = 9)
zaj <- 10^(seq(-2, 0.5, length.out = 9))
params <- expand.grid(B0s, zaj)
res <- list()
for(i in 1:nrow(params)){
  if(i%100==0) print(i)
  rej <- MC_sim(50, (function(x){matrix(c(0,0,x,0),2,2)})(params[i,1]),
    B1, sigma*params[i,2], nsim)$reject
  res[[i]] <- data.frame(B0 = params[i,1], zaj = params[i,2], reject = rej)
}

res_df <- do.call(rbind, res)
ggplot(res_df, aes(x=B0, y=zaj, z= reject))+
  geom_contour_filled(breaks = seq(0,1.01, by=0.1), show.legend = F)
```



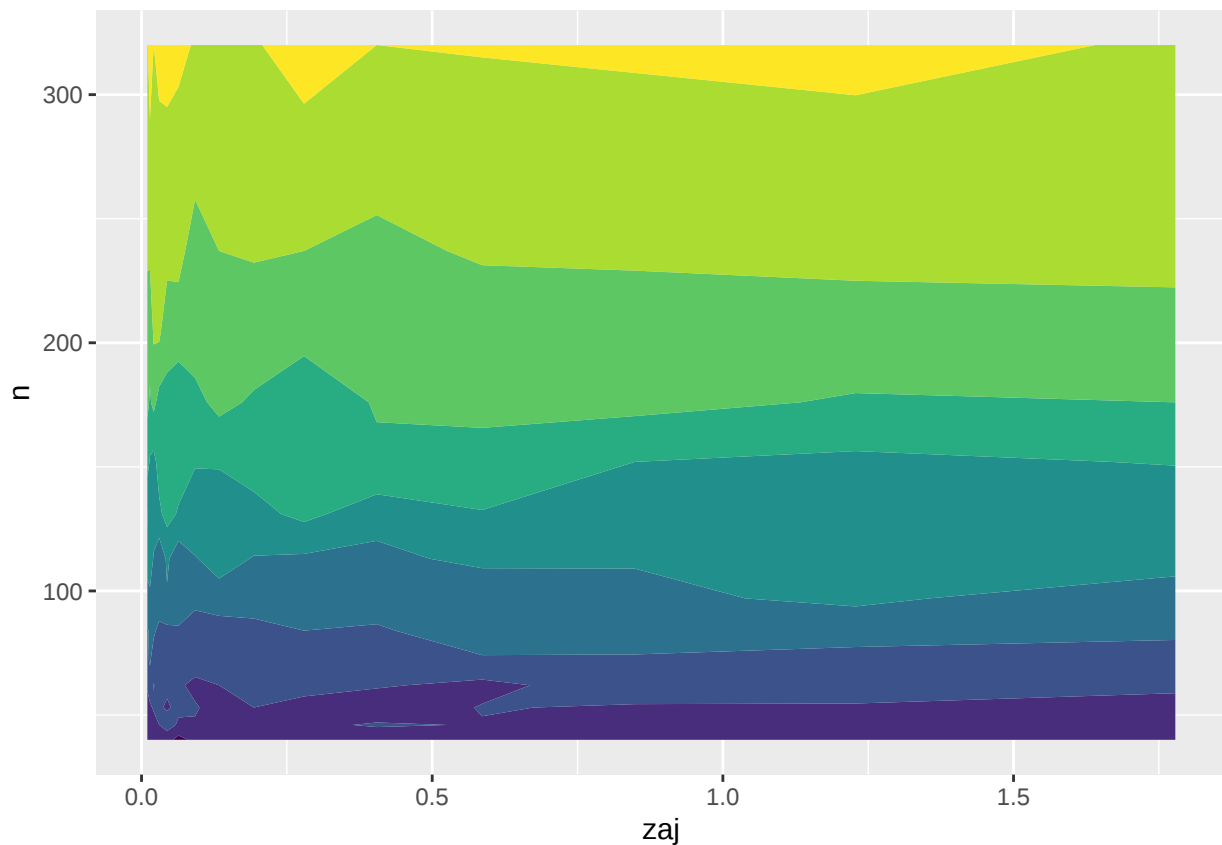
Elemszám és zaj méretének kapcsolata fix kapcsolaterősség mellett

```
nsim = 200

n_vec <- floor(10*2^seq(2,5, length.out = 15))
zaj <- 10^(seq(-2, 0.25, length.out = 15))
params <- expand.grid(n_vec, zaj)
res <- list()
for(i in 1:(nrow(params))){
  if(i%%100==0) print(i)
  rej <- MC_sim(params[i,1], (function(x){matrix(c(0,0,x,0),2,2)})(0.1),
    B1, sigma*params[i,2], nsim)$reject
  res[[i]] <- data.frame(n = params[i,1], zaj = params[i,2], reject = rej)
}

## [1] 100
## [1] 200

res_df <- do.call(rbind, res)
ggplot(res_df, aes(y=n, x=zaj,z= reject))+
  geom_contour_filled(breaks = seq(-0.01,2.01, by=0.1), show.legend = F)
```



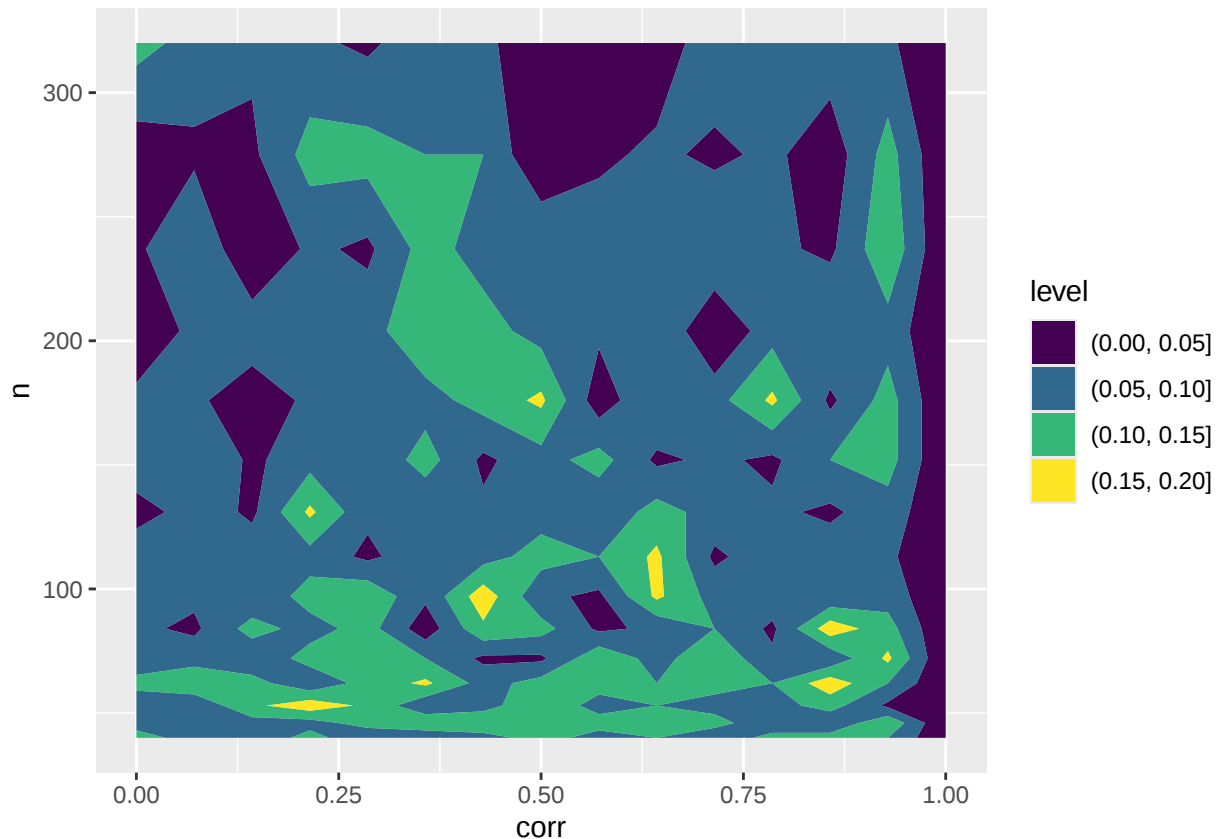
Zajok korrelációján keresztül van kapcsolat

Próbáltunk olyat is, hogy nincsen tényleges ok-okozati kapcsolat, de a zajok erősen korrelálnak. Ennél, mivel a VAR erre korrigál, így nem torzítja az együttthatók becsléseit.

```
nsim = 50
corr <- seq(0,1, length.out = 15)
n_vec <- floor(10*2^seq(2,5, length.out = 15))
params <- expand.grid(n_vec, corr)
res <- list()
for(i in 1:nrow(params)){
  if(i%100==0) print(i)
  rej <- MC_sim(params[i,1], B0,
                B1, (function(x){matrix(c(1,x,x,1),2,2)})(params[i,2]), nsim)$reject
  res[[i]] <- data.frame(n = params[i,1], corr = params[i,2], reject = rej)
}
```

```
## [1] 100
## [1] 200
```

```
res_df <- do.call(rbind, res)
ggplot(res_df, aes(y=n, x=corr,z= reject))+
  geom_contour_filled(breaks = seq(0,0.2, by=0.05), show.legend = T)
```



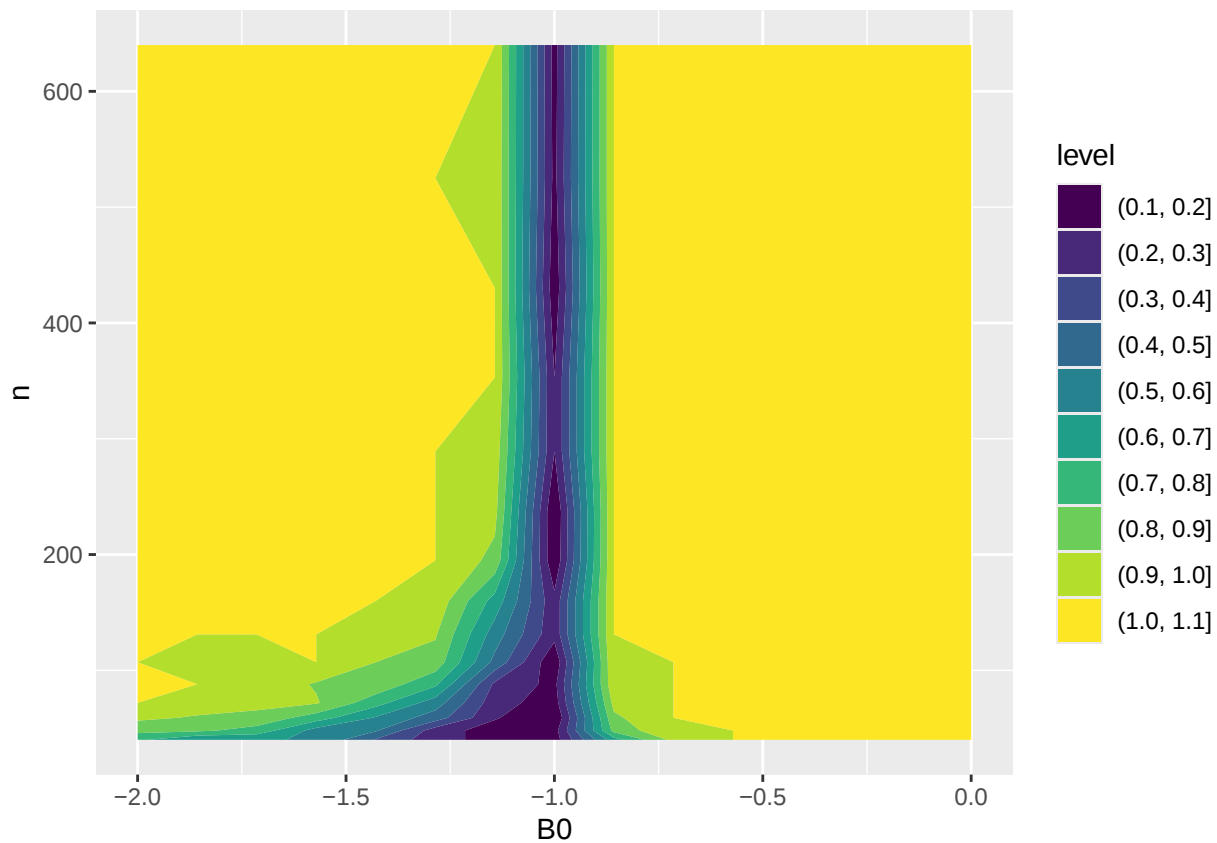
Azonnali és késleltetett ok-okozati kapcsolatok hatása VAR együttható becslésre

Ha van azonnali és késleltetett hatásunk is, és ezek ellentétes előjelűek, akkor ezek a hatások képesek egymást gyengíteni, annak ellenére, hogy mind a kettő fennáll, olyan, mintha semmilyen hatás nem lenne.

```
nsim = 100
B1 <- matrix(c(1,0,1,1),2)
B0s <- seq(-2,0, length.out = 15)
n_vec <- floor(10*2^seq(2,6, length.out = 15))
params <- expand.grid(n_vec, B0s)
res <- list()
for(i in 1:(nrow(params))) {
  if(i%%100==0) print(i)
  rej <- MC_sim(params[i,1], (function(x){matrix(c(0,0,x,0),2)})(params[i,2]),
    B1, sigma, nsim)$reject
  res[[i]] <- data.frame(n = params[i,1], B0 = params[i,2], reject = rej)
}

## [1] 100
## [1] 200

res_df <- do.call(rbind, res)
ggplot(res_df, aes(y=n, x=B0, z= reject))+
  geom_contour_filled(breaks = seq(0,1.1, by=0.1), show.legend = T)
```



Hyvarinen második példa

Az adatgeneráló folyamat

$$X_t = (I - B_0)^{-1} B_1 X_{t-1} + E_t$$

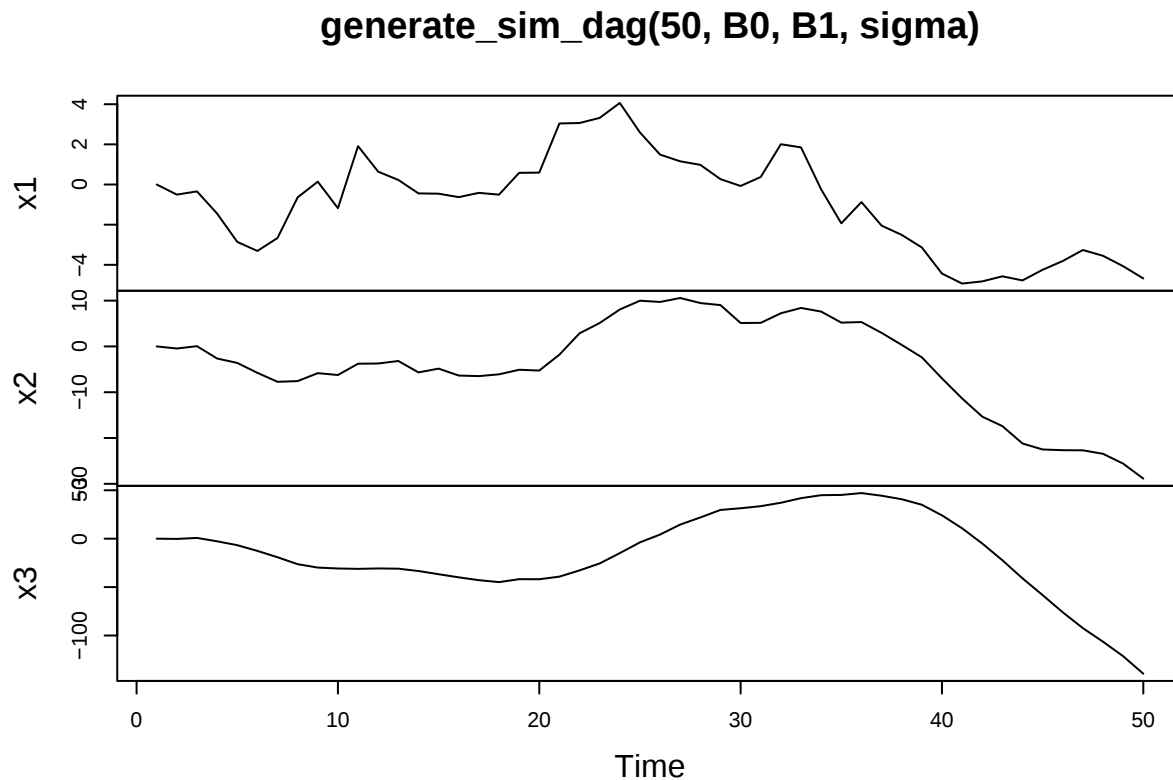
Ahol:

$$B_0 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0.9 & 0 & 0 \\ 0 & 0.9 & 0 \\ 0 & 0 & 0.9 \end{bmatrix}$$

```
B0 <- matrix(c(0,1,0,0,0,1,0,0,0),3)
B1 <- diag(0.9,3)
sigma<- diag(1,3)

generate_sim_dag(50, B0, B1, sigma) |> plot.ts()
```



Megvizsgáljuk azt is, hogy X_3 -at a teszt szerint Granger okozza-e X_1 közvetlenül. (Mivel közvetlenül nem okozza, így Spurious Granger okság)

```
nsim = 500
n_vec <- floor(10*2^seq(1,4, length.out = 15))
res <- list()
for(i in n_vec){
  mres <- c()
  for(j in 1:nsim){
    tres <- generate_sim_dag(i, matrix(c(0,1,0,0,0,1,0,0,0), 3, 3), B1 = diag(0.9,3),diag(1,3)) |> %>%
      summarise(pval = tres[["varresult"]][["x3"]][["coefficients"]][1,4])
    mres <- c(mres,pval)
  }
  res[[as.character(i)]] <- data.frame(N=i, pval = mean(mres))
}

res_df <- do.call(rbind, res)
ggplot(res_df, aes(y=pval, x=N))+
  geom_line() + ggtitle("Spurious Granger okság változó szignifikanciaszintje")
```

Spurious Granger okság változó szignifikanciaszintje

