University of Duisburg-Essen
Faculty of Business Administration and Economics
Chair of Econometrics

Winter Term 2019/2020

Dr. Yannick Hoga    Thilo Reinschlüssel

# Multivariate Time Series Analysis

## Solution Exercise Sheet 4

## 1 Exercise 1: Implied Models for Components

Consider the VAR(1) model $z_t = \phi_0 + \phi_1 z_{t-1} + a_t$ from the Exercise Sheet 3 again:

$$\phi_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \phi_1 = \begin{pmatrix} 0.75 & 0 \\ -0.25 & 0.5 \end{pmatrix}, \quad \Sigma_a = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} .$$

a) Write down the model using lag operator notation. Then rearrange the equation such that all parts based on $z_t$ are on the left-hand side and the remainder is on the right-hand side.

*Solution:*

$$\text{Model:} z_t = \phi_0 + \phi_1 z_{t-1} + a_t$$
$$\text{Lag notation: } z_t = \phi_0 + \phi_1 \,\mathrm{L}\, z_t + a_t$$

$$\Leftrightarrow z_t - \phi_1 \,\mathrm{L}\, Z_t = \phi_0 + a_t$$

b) By factoring out $z_t$ on the left, we obtain the lag polynomial $\phi(\mathrm{L})$. Compute its adjoint matrix by hand.

*Hint: Treat the lag operator as some scalar. The adjoint matrix can be computed like the inverse matrix but **without** the scaling by* $\dfrac{1}{\det(\phi(\mathrm{L}))}$.

*Solution:*

$$
\begin{aligned}
\underbrace{(I - \phi_1 \mathrm{L})}_{=:\phi\,\mathrm{L}} z_t &= \phi_0 + a_t \\
\Leftrightarrow \phi(\mathrm{L}) &= \begin{pmatrix} 1 - 0.75\,\mathrm{L} & 0 \\ -0.25\,\mathrm{L} & 1 - 0.5\,\mathrm{L} \end{pmatrix} \\
\Leftrightarrow \phi^{\text{adj}} &= \begin{pmatrix} 1 - 0.5\,\mathrm{L} & 0 \\ 0.25\,\mathrm{L} & 1 - 0.75\,\mathrm{L} \end{pmatrix}
\end{aligned}
$$

c) Pre-multiply the model equation you got in part a) with the adjoint matrix you computed in part b).

*Hint: You are supposed to end up with a diagonal matrix.*

*Solution:*

$$
\begin{aligned}
\begin{pmatrix} 1 - 0.5\,\mathrm{L} & 0 \\ 0.25\,\mathrm{L} & 1 - 0.75\,\mathrm{L} \end{pmatrix} \begin{pmatrix} 1 - 0.75\,\mathrm{L} & 0 \\ -0.25\,\mathrm{L} & 1 - 0.5\,\mathrm{L} \end{pmatrix} z_t &= \begin{pmatrix} 1 - 0.75\,\mathrm{L} & 0 \\ 0.25\,\mathrm{L} & 1 - 0.75\,\mathrm{L} \end{pmatrix} \cdot (\phi_0 + a_t) \\
\Leftrightarrow \begin{pmatrix} (1 - 0.5\,\mathrm{L})(1 - 0.75\,\mathrm{L}) & 0 \\ (0.25\,\mathrm{L})(1 - 0.75\,\mathrm{L}) + (1 - 0.75\,\mathrm{L})(-0.25\,\mathrm{L}) & (1 - 0.75\,\mathrm{L})(1 - 0.5\,\mathrm{L}) \end{pmatrix} z_t &= \begin{pmatrix} (1 - 0.5\,\mathrm{L}) \cdot 1 \\ 0.25\,\mathrm{L} \cdot 1 + (1 - 0.75\,\mathrm{L}) \cdot 0 \end{pmatrix} + \begin{pmatrix} (1 - 0.5\,\mathrm{L}) \cdot a_{1,t} \\ (0.25\,\mathrm{L})a_{1,t} + 1(1 - 0.75\,\mathrm{L})a_{2,t} \end{pmatrix} \\
\Leftrightarrow \begin{pmatrix} z_{1,t} - 1.25 z_{1,t-1} + 0.375 z_{1,t-2} \\ z_{2,t} - 1.25 z_{2,t-1} + 0.375 z_{2,t-2} \end{pmatrix} &= \begin{pmatrix} 0.5 \\ 0.25 \end{pmatrix} + \begin{pmatrix} a_{1,t} - 0.5 a_{1,t-1} \\ 0 + 0.25 a_{1,t-1} + a_{2,t} - 0.75 a_{2,t-1} \end{pmatrix}
\end{aligned}
$$

d) The result of part c) should be a collection of two univariate ARMA(p,q) models. What is the lag order of both models?

*Solution:*

The lag order of both models is ARMA(2,1)

e) Simulate a trajectory with T = 1000 of the original VAR(1) model.

*Hint:'VARMAsim' on Slide 2–6.*

*Solution:*

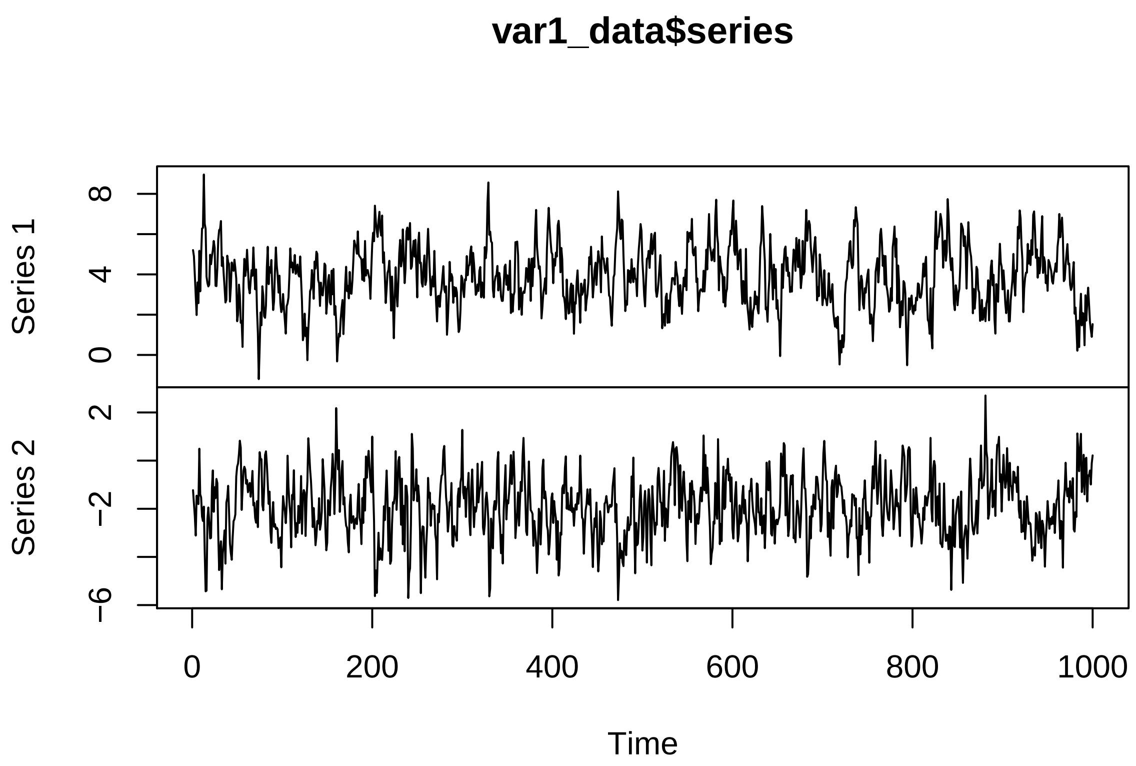
```r
# Preparation: Define matrices
phi_1 <- matrix(data = c(0.75, -0.25, 0, 0.5), nrow = 2)
phi_0 <- c(1, 0)
Sigma_a <- matrix(data = c(1, 0, 0, 1), nrow = 2)
```

```r
N <- 10^3 # length of trajectory
burn_in <- 250 # extra periods, are then cut away from the simulated
#to reduce the impact of starting point selection

set.seed(42) # set seed for reproducibility

var1_data <- VARMAsim(nobs = N, arlags = c(1), malags = NULL,
                      cnst = phi_0, phi = phi_1, skip = burn_in, sigma = Sigma_a)

plot.ts(var1_data$series)
```




f) Fit a VAR(1) model to the data, store the results as a variable and estimate the predictions' mean squared error for each variable in $z_t$

*Solution:*

```r
var1_fit <- VAR(x = var1_data$series, p = c(1), include.mean = TRUE)
```

```
## Constant term:
## Estimates:  0.9776701 -0.03392965
```

```
## Std.Error:  0.09103255 0.08841188
## AR coefficient matrix
## AR( 1 )-matrix
##        [,1]     [,2]
## [1,]  0.747 -0.00125
## [2,] -0.226  0.51979
## standard error
##        [,1]   [,2]
## [1,] 0.0222 0.0260
## [2,] 0.0215 0.0253
##
## Residuals cov-mtx:
##            [,1]       [,2]
## [1,] 1.02729516 0.01620576
## [2,] 0.01620576 0.96899845
##
## det(SSE) =  0.9951848
## AIC =  0.003173153
## BIC =  0.02280417
## HQ  =  0.01063431
```

```r
mse_var <- colMeans(var1_fit$residuals^2) # MSEs of two sequences of residuals (a_1, a_
mse_var
```

```
## [1] 1.0272952 0.9689984
```

As we can see the estimation of a VAR(1) comes pretty close to the "true" (simulated) values of $\phi_0$ and $\phi_1$

```r
mse_var <- colMeans(var1_fit$residuals^2) # MSEs of two sequences of residuals
mse_var
```

```
## [1] 1.0272952 0.9689984
```

The MSEs are as we would expected, given $\Sigma_a$.

g) Repeat the task by fitting the two ARMA(p,q) models from b) to the data. Again compute the mean squared error for $z_{1,t}$ and $z_{2,t}$ each.

   *Hint:'arima'.*

*Solution:*

```
z1_fit <- arima(x = var1_data$series[,1], order = c(2,0,1), include.mean = TRUE)
z2_fit <- arima(x = var1_data$series[,2], order = c(2,0,1), include.mean = TRUE,
                optim.control = list(maxit = 10^3))

mse_arma <- c( mean(z1_fit$residuals^2), mean(z2_fit$residuals^2) ) # (a_1, a_2)

mse_arma
```

```
## [1] 1.025297 1.074352
```

h) Compare the MSEs of the VAR(1) estimates and the ARMA(p,q) estimates. Did the VAR(1) and the univariate ARMA(p,q) models perform similarly? If not, provide an intuition why.

*Solution:*

```
mse_arma / mse_var # element-wise ratio of MSEs
```

```
## [1] 0.998055 1.108725
```

$z_{1,t}$ is predicted similarly well by both models, but $z_{2,t}$ is predicted much betterby the VAR(1).\
Reson: $z_{1,t}$ is a genuine AR(1) process independent of $a_{2,t}$, whereas $z_{2,t}$ depends on $\underline{a_{2,t} \text{ and } a_{1,t}}$ through $z_{1,t}$. But a univariate model allows only to estimate the aggregated innovation sequence when the VAR estimated $k = 2$ sequences.

i) How can you manipulate the $\Sigma_a$ a matrix to equalise the MSEs of both the VAR(1) and the ARMA(p,q) models?

Two options:

- Option 1: $a_{1,t} \overset{!}{=} a_{2,t} =: \tilde{a}_t$

$$\Rightarrow \Sigma_a = \begin{pmatrix} \sigma_a^2 & \sigma_a^2 \\ \sigma_a^2 & \sigma_a^2 \end{pmatrix} = \sigma_a^2 \cdot \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

- Option 2: Let $a_{2,t}$ dominate $a_{1,t}$ by a larger varaince to mariginalise $a_{1,t}$.

$$\overset{\text{example}}{\Rightarrow} \Sigma_a = \begin{pmatrix} 1 & 0 \\ 0 & 10 \end{pmatrix}$$

## 2 Exercise 2: Least Squares Estimation

a) Again use your simulated time series from exercise 1. Regress $z_{1,t}$ on $z_{1,t-1}$ and $z_{2,t-1}$, then repeat with $z_{2,t}$ as dependent variable (meaning you estimate each row of the VAR(1) specification separately). How similar are the coefficients to those obtained from the VAR(1) regression?

*Solution:*

```
# estimating the VAR regression by regression
z1_reg <- lm(var1_data$series[-1,1] ~ var1_data$series[-N,]) # row 1
z2_reg <- lm(var1_data$series[-1,2] ~ var1_data$series[-N,]) # row 2
ols_coef_matrix <- cbind(z1_reg$coefficients, z2_reg$coefficients) # putting the coeffi
# comparing the coefficients

ols_coef_matrix - var1_fit$coef # difference
```

```
##                                [,1]          [,2]
## (Intercept)            -2.453593e-14 -1.216388e-14
## var1_data$series[-N, ]1  8.437695e-15  4.357625e-15
## var1_data$series[-N, ]2  5.340563e-15  2.886580e-15
```

As we can see from the differences, the coefficients coincide pretty well.

b) Show that you can generally estimate a VAR(p) by row-wise separate regressions using the derivations starting from Slide 3–3.

*Hint: Make sure you understand how the trick in equation (3.3) works.*

*Solution:*

The trick:

$$\operatorname{vec}(ABC) = (C' \otimes A)\operatorname{vec}(B)$$
$$\Rightarrow X\beta = X\beta I_k \Rightarrow \operatorname{vec}(X\beta) = \operatorname{vec}(X\beta I_k) = (I_k \otimes X)\operatorname{vec}(\beta)$$

Here:

$Z = X\beta + A, \widehat{\beta} = \left(X'X\right)^{-1} X'Z$ and $\beta$ is a $(Kp+1) \times K$ matrix. If one wants to predict $z_j$ (column "j" in Z), one can rewrite the estimat to: $\operatorname{vec}\left(\widehat{\beta}\right) = \left(I_K \otimes \left(X'X\right)^{-1} X'\right)\operatorname{vec}(z)$. One is interested in column "j" of $Z$ and $\widehat{\beta}_1$ that means one has to look at the "$j^{th}$" row of matrices in $\left(I_K \otimes \left(X'X\right)^{-1} X'\right)$. This is done by inspecting $I_K(j,j)$. The result is just $\left(X'X\right)^{-1} X'$ since $I_K(j,l) = 0 \; \forall \; l \neq j$.

$$\mathrm{vec}\left(\widehat{\beta}\right) = \begin{pmatrix} \left(X'X\right)^{-1} X' & 0_{T-p} & \dots & & 0_{T-p} \\ 0_{T-p} & \left(X'X\right)^{-1} X' & \ddots & & \vdots \\ \vdots & & \left(X'X\right)^{-1} X' & & \vdots \\ \vdots & \ddots & \ddots & & 0_{T-p} \\ 0_{T-p} & \vdots & \vdots & 0_{T-p} & \left(X'X\right)^{-1} X' \end{pmatrix} \begin{pmatrix} z_{1,p+1} \\ \vdots \\ z_{1,T} \\ z_{2,p+1} \\ \vdots \\ z_{1,T} \\ \vdots \\ z_{j,p+1} \\ z_{j,T} \\ z_{j+1,p+1} \\ z_{j+1,T} \\ \vdots \\ z_{K,T} \end{pmatrix}$$

$$= \begin{pmatrix} \phi_{0,1} & \phi_{0,2} & \dots & \phi_{0,j} & \dots & \phi_{0,K} \\ \phi_{1,11} & \dots & \dots & \phi_{1,1j} & \dots & \phi_{0,1K} \\ \vdots & \dots & \dots & \vdots & \vdots & \vdots \\ \phi_{1,K1} & \vdots & \vdots & \phi_{1,Kj} & \dots & \phi_{0,KK} \\ \vdots & \dots & \dots & \vdots & \vdots & \vdots \\ \phi_{p,K1} & \dots & \dots & \phi_{p,Kj} & \dots & \phi_{p,KK} \end{pmatrix}$$

That's why $\widehat{\beta}_j = \left(X'X\right)^{-1} X'Z_j$.

## 3 Exercise 3: Maximum Likelihood Estimation

a) Let $\epsilon_1, \dots, \epsilon_T$ an i.i.d. sample from a normal distribution with unknown mean $\mu$ and variance $\sigma^2$. Find maximum likelihood estimators for $\mu$ and $\sigma^2$.

*Solution:*

$$L = \prod_{t=1}^{T} f\left(\epsilon_t; \mu, \sigma^2\right)$$

$$\text{standard normal} = \prod_{t=1}^{T} \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{1}{2}\left(\frac{\epsilon_t - \mu}{\sigma}\right)^2}$$

$$\overset{\log(\cdot)}{\Rightarrow} \text{ll} = \sum_{t=1}^{T} \left[ -\frac{1}{2} \log(2\pi\sigma^2) - \frac{1}{2} \left( \frac{\epsilon_t - \mu}{\sigma} \right)^2 \right]$$

$$= -\frac{T}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \cdot \sum_{t=1}^{T} (\epsilon_t - \mu)^2$$

FOCs :

$$\frac{\partial \, \text{ll}}{\partial \mu} = -\frac{1}{2\sigma^2} \cdot (-2) \cdot \sum_{t=1}^{T} (\epsilon_t - \mu) \stackrel{!}{=} 0$$

$$\Leftrightarrow 0 \stackrel{!}{=} \sum_{t=1}^{T} \epsilon_t - \sum_{t=1}^{T} \mu$$

$$\Leftrightarrow \sum_{t=1}^{T} \epsilon_t \stackrel{!}{=} T \cdot \mu$$

$$\Leftrightarrow \mu = \underline{\frac{1}{T} \sum_{t=1}^{T} \epsilon_t}$$

$$\frac{\partial \, \text{ll}}{\partial \sigma^2} = -\frac{T}{2} \cdot 2\pi \cdot \frac{1}{2\pi\sigma^2} - \frac{1}{2} \cdot (-1) \cdot \sum_{t=1}^{T} \frac{(\epsilon_t - \mu)^2}{\sigma^4} \stackrel{!}{=} 0$$

$$\Leftrightarrow 0 \stackrel{!}{=} -\frac{T}{2\sigma^2} + \frac{1}{2} \frac{1}{\sigma^4} \sum_{t=1}^{T} (\epsilon_t - \mu)^2$$

$$\Leftrightarrow sum_{t=1}^{T} (\epsilon_t - \mu)^2 = \frac{2T\sigma^4}{2\sigma^2}$$

$$\Leftrightarrow \sigma^2 = \underline{\frac{1}{T} \sum_{t=1}^{T} (\epsilon_t - \mu)^2}$$

b) Prove equation (3.12) in the lecture slides.

*Solution:*

In general, let $f(x, y, z)$ be a joint density.

$$f(x, y, z) = \underbrace{\frac{f(x, y, z)}{f(y, z)}}_{=: f(x)_{x|Y=y, Z=z}} \cdot f(y, z)$$

$$= f_{x|Y,Z}(x) \cdot f_{y|Z}(y) \cdot f_Z(z)$$

Here:

$$
\begin{aligned}
f_{z_{p+1:T}|z_{1:p}}(z_{p+1},\ldots,z_T) &= f_{z_T|z_{1:T-1}}(z_T)\cdot f_{z_{p+1:T}-1|z_{1:p}}(z_{p+1},\ldots,z_{T-1}) \\
&= f_{z_T|z_{1:T-1}}(z_T)\cdot f_{z_T|z_{1:T-2}}(z_{T-1})\cdot\ldots\cdot f_{z_{p+1}|z_{1:p}}(z_{p+1}) \\
&= \prod_{t=p+1}^{T} f_{z_t|z_{p:t-1}}(z_t) z_{p+1} - \mathbb{E}\left(z_{p+1}|z_p,\ldots,z_1\right)
\end{aligned}
$$