Time Series Analysis

Henrik Madsen

hm@imm.dtu.dk

Informatics and Mathematical Modelling
Technical University of Denmark
DK-2800 Kgs. Lyngby

Outline of the lecture

Chapter 9 – Multivariate time series

Transfer function models with ARMA input

$$
\begin{aligned}
Y_t &= \frac{\omega(B)}{\delta(B)} B^b X_t + \frac{\theta_\varepsilon(B)}{\varphi_\varepsilon(B)} \varepsilon_t \\
X_t &= \frac{\theta_\eta(B)}{\varphi_\eta(B)} \eta_t
\end{aligned}
$$

we require $\{\varepsilon_t\}$ and $\{\eta_t\}$ to be mutually uncorrelated.

Transfer function models with ARMA input

$$\begin{aligned} Y_t &= \frac{\omega(B)}{\delta(B)} B^b X_t + \frac{\theta_\varepsilon(B)}{\varphi_\varepsilon(B)} \varepsilon_t \\ X_t &= \frac{\theta_\eta(B)}{\varphi_\eta(B)} \eta_t \end{aligned}$$

From the above we get

$$\begin{aligned} \delta(B)\varphi_\varepsilon(B) Y_t &= \varphi_\varepsilon(B)\omega(B) B^b X_t + \delta(B)\theta_\varepsilon(B)\varepsilon_t \\ \varphi_\eta(B) X_t &= \theta_\eta(B)\eta_t \end{aligned}$$

The term including X_t on the RHS is moved to the LHS

Transfer function models with ARMA input

$$
\begin{aligned}
Y_t &= \frac{\omega(B)}{\delta(B)} B^b X_t + \frac{\theta_\varepsilon(B)}{\varphi_\varepsilon(B)} \varepsilon_t \\
X_t &= \frac{\theta_\eta(B)}{\varphi_\eta(B)} \eta_t
\end{aligned}
$$

$$
\begin{aligned}
\delta(B)\varphi_\varepsilon(B)Y_t - \varphi_\varepsilon(B)\omega(B)B^b X_t &= \delta(B)\theta_\varepsilon(B)\varepsilon_t \\
\varphi_\eta(B)X_t &= \theta_\eta(B)\eta_t
\end{aligned}
$$

Which can be written in matrix notation

Transfer function models with ARMA input

$$\begin{aligned} Y_t &= \frac{\omega(B)}{\delta(B)} B^b X_t + \frac{\theta_\varepsilon(B)}{\varphi_\varepsilon(B)} \varepsilon_t \\ X_t &= \frac{\theta_\eta(B)}{\varphi_\eta(B)} \eta_t \end{aligned}$$

$$\begin{bmatrix} \delta(B)\varphi_\varepsilon(B) & -\varphi_\varepsilon(B)\omega(B)B^b \\ 0 & \varphi_\eta(B) \end{bmatrix} \begin{bmatrix} Y_t \\ X_t \end{bmatrix} = \begin{bmatrix} \delta(B)\theta_\varepsilon(B) & 0 \\ 0 & \theta_\eta(B) \end{bmatrix} \begin{bmatrix} \varepsilon_t \\ \eta_t \end{bmatrix}$$

Transfer function models with ARMA input

$$\begin{aligned} Y_t &= \frac{\omega(B)}{\delta(B)} B^b X_t + \frac{\theta_\varepsilon(B)}{\varphi_\varepsilon(B)} \varepsilon_t \\ X_t &= \frac{\theta_\eta(B)}{\varphi_\eta(B)} \eta_t \end{aligned}$$

$$\begin{bmatrix} \delta(B)\varphi_\varepsilon(B) & -\varphi_\varepsilon(B)\omega(B)B^b \\ 0 & \varphi_\eta(B) \end{bmatrix} \begin{bmatrix} Y_t \\ X_t \end{bmatrix} = \begin{bmatrix} \delta(B)\theta_\varepsilon(B) & 0 \\ 0 & \theta_\eta(B) \end{bmatrix} \begin{bmatrix} \varepsilon_t \\ \eta_t \end{bmatrix}$$

For multivariate ARMA-models we replace the zeroes by polynomials in B, allow non-zero correlation between ε_t and η_t, and generalize to more dimensions

Multivariate ARMA models

- The model can be written

$$\boldsymbol{\phi}(B)(\boldsymbol{Y}_t - \boldsymbol{c}) = \boldsymbol{\theta}(B)\boldsymbol{\epsilon}_t$$

- The individual time series may have been transformed and differenced
- The variance-covariance matrix of the multivariate white noise process $\{\boldsymbol{\epsilon}_t\}$ is denoted $\boldsymbol{\Sigma}$.
- The matrices $\boldsymbol{\phi}(B)$ and $\boldsymbol{\theta}(B)$ has elements which are polynomials in the backshift operator
- The diagonal elements has leading terms of unity
- The off-diagonal elements have leading terms of zero (i.e. they normally start in B)

Air pollution in cities NO and NO_2

$$\begin{bmatrix} X_{1,t} \\ X_{2,t} \end{bmatrix} = \begin{bmatrix} 0.9 & -0.1 \\ 0.4 & 0.8 \end{bmatrix} \begin{bmatrix} X_{1,t-1} \\ X_{2,t-1} \end{bmatrix} + \begin{bmatrix} \xi_{1,t} \\ \xi_{2,t} \end{bmatrix} \qquad \mathbf{\Sigma} = \begin{bmatrix} 30 & 21 \\ 21 & 23 \end{bmatrix}$$

Or

$$\begin{aligned} X_{1,t} - 0.9X_{1,t} + 0.1X_{2,t-1} &= \xi_{1,t} \\ -0.4X_{1,t-1} + X_{2,t} - 0.8X_{2,t-1} &= \xi_{2,t} \end{aligned}$$

the LHS can be written using a matrix for which the elements are polynomials i B

Air pollution in cities NO and NO_2

$$\begin{bmatrix} X_{1,t} \\ X_{2,t} \end{bmatrix} = \begin{bmatrix} 0.9 & -0.1 \\ 0.4 & 0.8 \end{bmatrix} \begin{bmatrix} X_{1,t-1} \\ X_{2,t-1} \end{bmatrix} + \begin{bmatrix} \xi_{1,t} \\ \xi_{2,t} \end{bmatrix} \qquad \boldsymbol{\Sigma} = \begin{bmatrix} 30 & 21 \\ 21 & 23 \end{bmatrix}$$

Formulation using the backshift operator:

$$\begin{bmatrix} 1-0.9B & 0.1B \\ -0.4B & 1-0.8B \end{bmatrix} \boldsymbol{X}_t = \boldsymbol{\xi}_t \quad \text{or} \quad \phi(B)\boldsymbol{X}_t = \boldsymbol{\xi}_t$$

Air pollution in cities NO and NO_2

$$\begin{bmatrix} X_{1,t} \\ X_{2,t} \end{bmatrix} = \begin{bmatrix} 0.9 & -0.1 \\ 0.4 & 0.8 \end{bmatrix} \begin{bmatrix} X_{1,t-1} \\ X_{2,t-1} \end{bmatrix} + \begin{bmatrix} \xi_{1,t} \\ \xi_{2,t} \end{bmatrix} \qquad \boldsymbol{\Sigma} = \begin{bmatrix} 30 & 21 \\ 21 & 23 \end{bmatrix}$$

Formulation using the backshift operator:

$$\begin{bmatrix} 1-0.9B & 0.1B \\ -0.4B & 1-0.8B \end{bmatrix} \boldsymbol{X}_t = \boldsymbol{\xi}_t \quad \text{or} \quad \phi(B)\boldsymbol{X}_t = \boldsymbol{\xi}_t$$

Alternative formulation:

$$\boldsymbol{X}_t - \begin{bmatrix} 0.9 & -0.1 \\ 0.4 & 0.8 \end{bmatrix} \boldsymbol{X}_{t-1} = \boldsymbol{\xi}_t \quad \text{or} \quad \boldsymbol{X}_t - \phi_1 \boldsymbol{X}_{t-1} = \boldsymbol{\xi}_t$$

Stationarity and Invertability

The multivariate ARMA process

$$\boldsymbol{\phi}(B)(\boldsymbol{Y}_t - \boldsymbol{c}) = \boldsymbol{\theta}(B)\boldsymbol{\epsilon}_t$$

is stationary if

$$\det(\boldsymbol{\phi}(z^{-1})) = 0 \;\Rightarrow\; |z| < 1$$

is invertible if

$$\det(\boldsymbol{\theta}(z^{-1})) = 0 \;\Rightarrow\; |z| < 1$$

Two formulations (centered data)

Matrices with polynomials in B as elements:

$$\boldsymbol{\phi}(B)\boldsymbol{Y}_t = \boldsymbol{\theta}(B)\boldsymbol{\epsilon}_t$$

Without B, but with matrices as coefficients:

$$\boldsymbol{Y}_t - \boldsymbol{\phi}_1\boldsymbol{Y}_{t-1} - \ldots - \boldsymbol{\phi}_p\boldsymbol{Y}_{t-p} = \boldsymbol{\epsilon}_t - \boldsymbol{\theta}_1\boldsymbol{\epsilon}_{t-1} - \ldots - \boldsymbol{\theta}_q\boldsymbol{\epsilon}_{t-q}$$

Auto Covariance Matrix Functions

$$\boldsymbol{\Gamma}_k = E[(\boldsymbol{Y}_{t-k} - \boldsymbol{\mu}_Y)(\boldsymbol{Y}_t - \boldsymbol{\mu}_Y)^T] = \boldsymbol{\Gamma}_{-k}^T$$

Example for bivariate case $\boldsymbol{Y}_t = (Y_{1,t}\ Y_{2,t})^T$:

$$\boldsymbol{\Gamma}_k = \begin{bmatrix} \gamma_{11}(k) & \gamma_{12}(k) \\ \gamma_{21}(k) & \gamma_{22}(k) \end{bmatrix} = \begin{bmatrix} \gamma_{11}(k) & \gamma_{12}(k) \\ \gamma_{12}(-k) & \gamma_{22}(k) \end{bmatrix}$$

And therefore we will plot autocovariance or autocorrelation functions for $k = 0, 1, 2, \ldots$ and one of each pair of cross-covariance or cross-correlation functions for $k = 0, \pm 1, \pm 2, \ldots$

The Theoretical Autocovariance Matrix Functions

Using the matrix coefficients $\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_p$ and $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_q$, together with $\boldsymbol{\Sigma}$, the theoretical $\boldsymbol{\Gamma}_k$ can be calculated:

Pure Autoregressive Models: $\boldsymbol{\Gamma}_k$ is found from a multivariate version of Theorem 5.10 in the book, which leads to the Yule-Walker equations

Pure Moving Average Models: $\boldsymbol{\Gamma}_k$ is found from a multivariate version of (5.65) in the book

Autoregressive Moving Average Models: $\boldsymbol{\Gamma}_k$ is found multivariate versions of (5.100) and (5.101) in the book

- Examples can be found in the book – note the VAR(1)!

Identification using Autocovariance Matrix Functions

Sample Correlation Matrix Function; $\boldsymbol{R}_k$ near zero for pure moving average processes of order q when $k > q$

Sample Partial Correlation Matrix Function; $\boldsymbol{S}_k$ near zero for pure autoregressive processes of order p when $k > p$

Sample q-conditioned Partial Correlation Matrix Function; $\boldsymbol{S}_k(q)$ near zero for autoregressive moving average processes of order (p, q) when $k > p$ – can be used for univariate processes also.

Identification using (multivariate) prewhitening

- Fit univariate models to each individual series
- Investigate the residuals as a multivariate time series
- The cross correlations can then be compared with $\pm 2/\sqrt{N}$

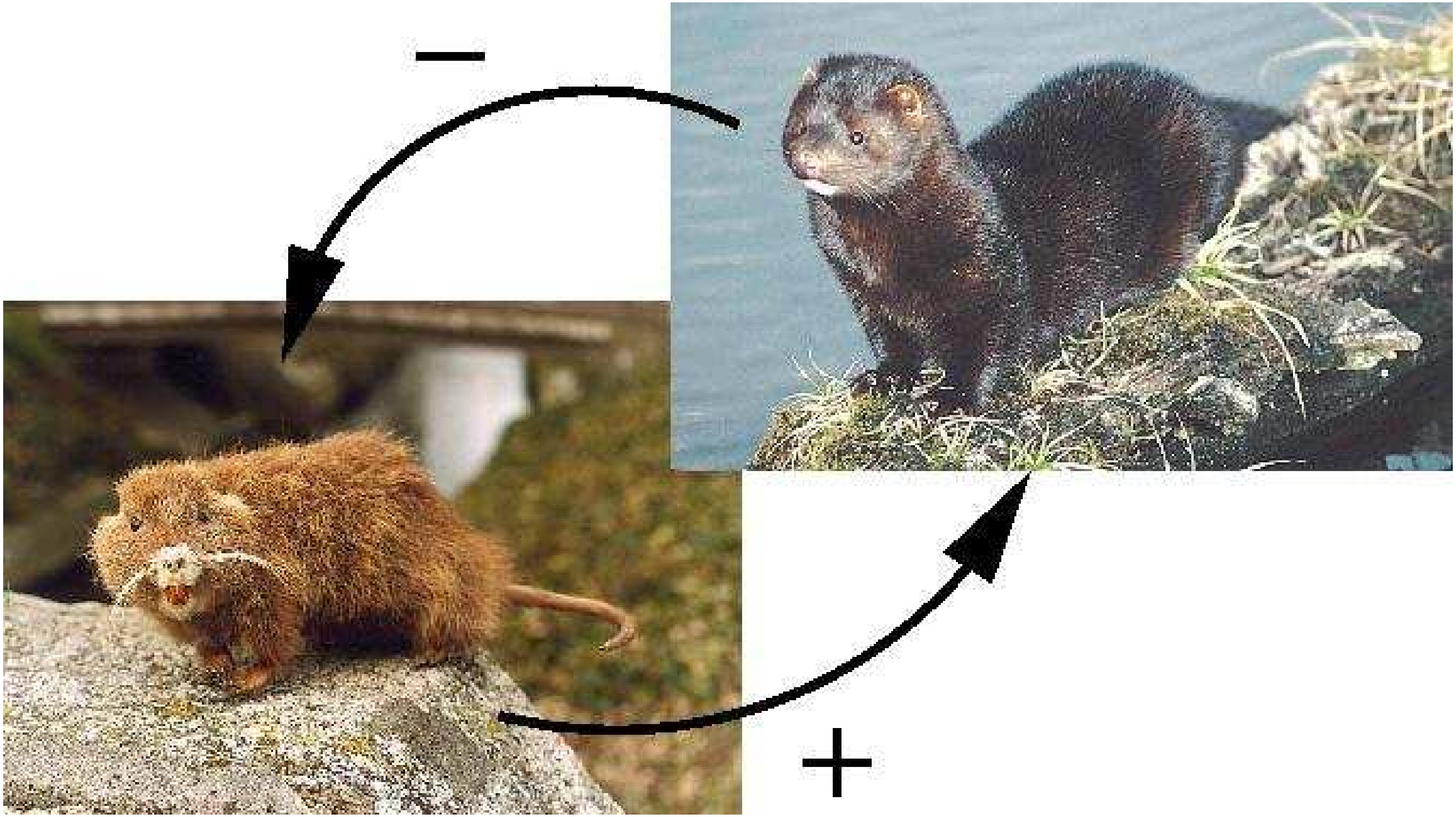
This is **not** the same form of prewhitening as in Chapter 8

The multivariate model $\boldsymbol{\phi}(B)Y_t = \boldsymbol{\theta}(B)\epsilon_t$ is equivalent to

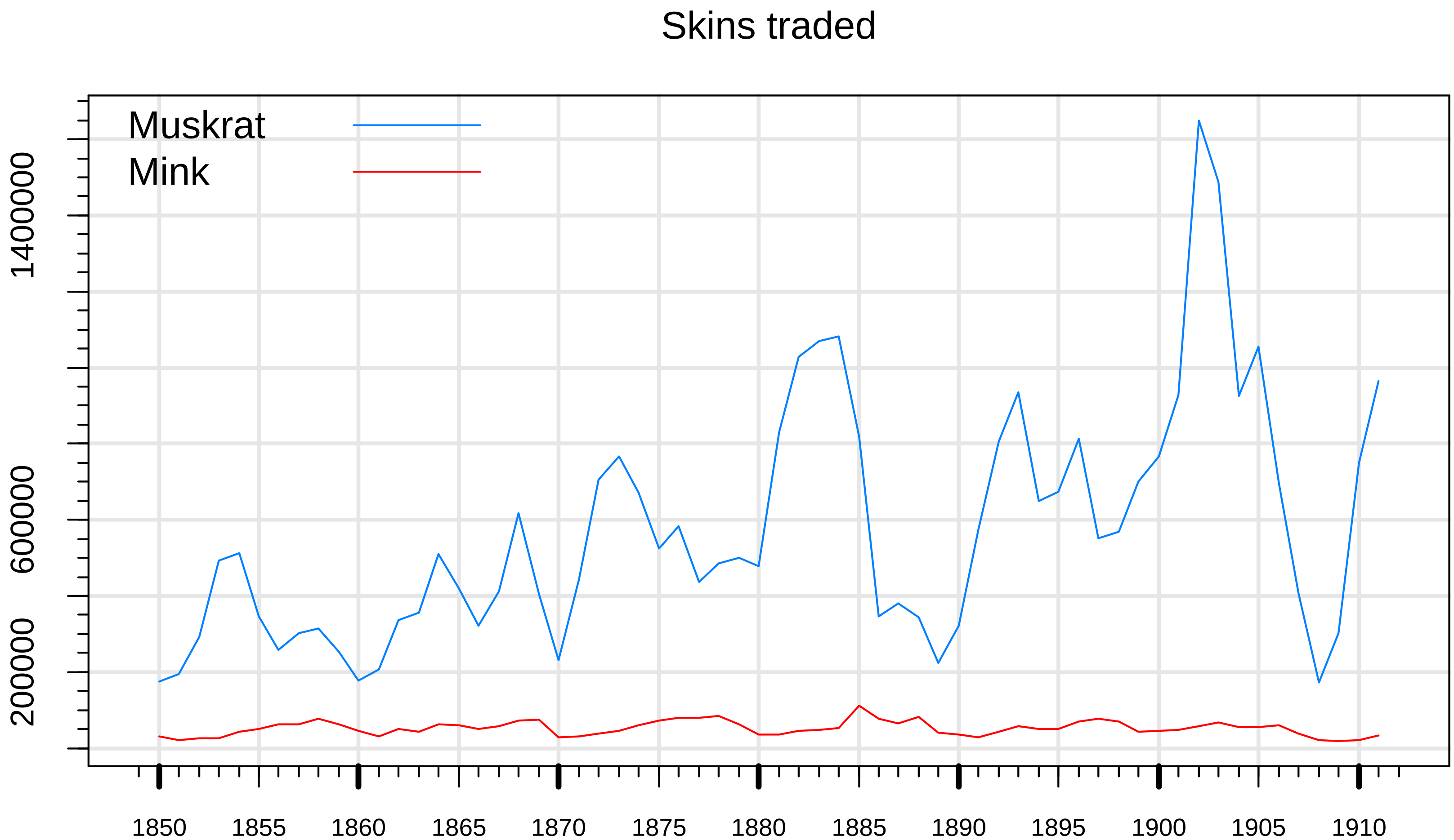
$$\text{diag}(\det(\boldsymbol{\phi}(B)))Y_t = \text{adj}(\boldsymbol{\phi}(B))\boldsymbol{\theta}(B)\epsilon_t$$

Therefore the corresponding univariate models will have much higher order, so although this approach is often used in the literature: Don't use this approach!

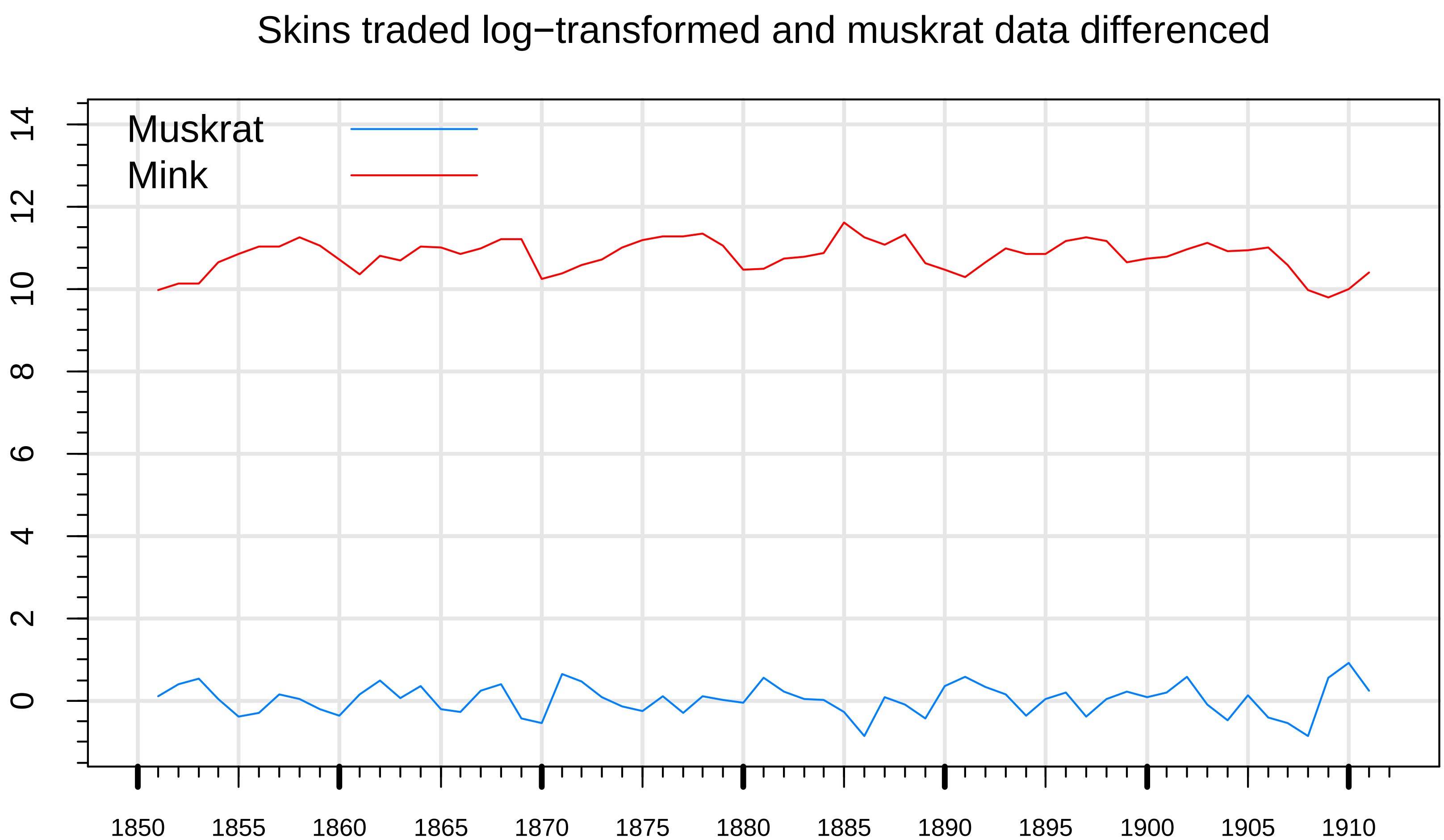
Example – Muskrat and Mink skins traded



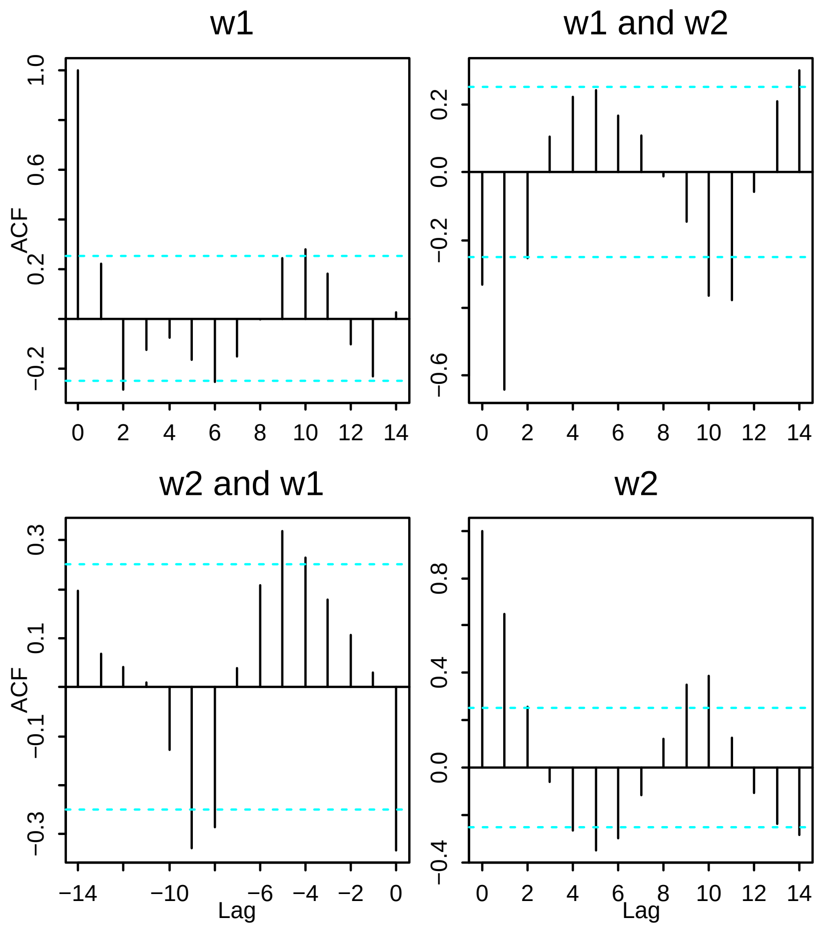
Raw data (maybe not exactly as in the paper)



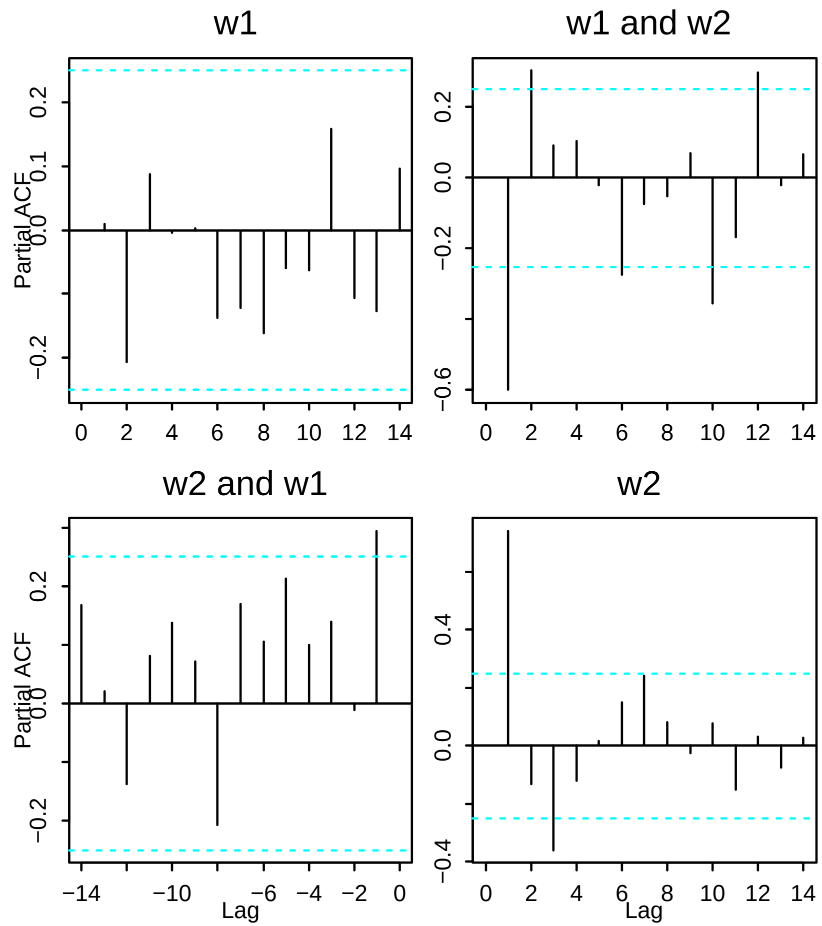
Stationary and Gaussian data



SACF

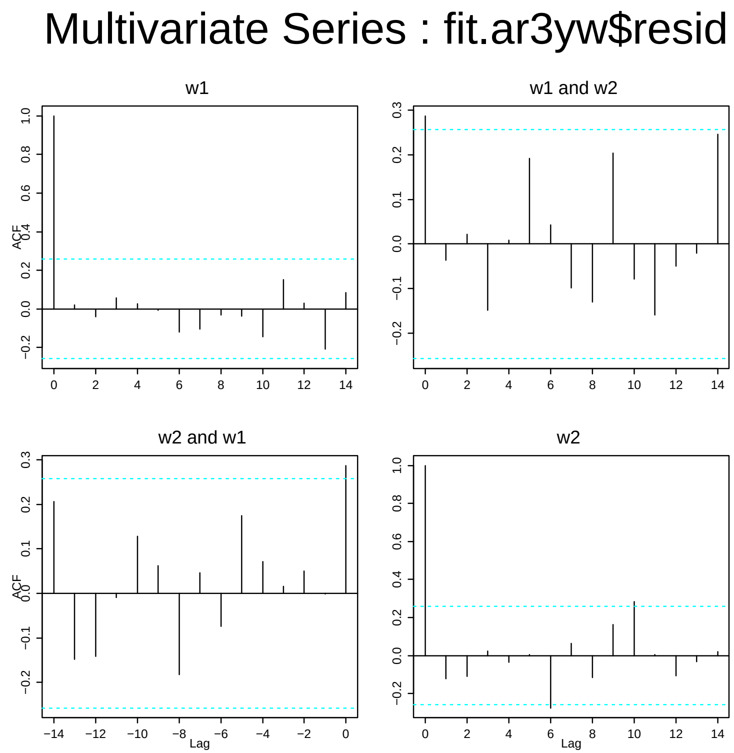

w1
w1 and w2
w2 and w1
w2
ACF
Lag

SPACF


w1
w1 and w2
w2 and w1
w2
Partial ACF
Lag

Yule-Walker Estimates

```
> fit.ar3yw <- ar(mmdata.tr, order=3)
> fit.ar3yw$ar[1,,] # lag 1
          [,1]       [,2]
[1,] 0.2307413 -0.7228413
[2,] 0.4172876  0.7417832
> fit.ar3yw$ar[2,,] # lag 2
           [,1]      [,2]
[1,] -0.1846027 0.2534646
[2,] -0.2145025 0.1920450
> fit.ar3yw$ar[3,,] # lag 3
           [,1]        [,2]
[1,] 0.08742842  0.09047402
[2,] 0.14015902 -0.36146002
> acf(fit.ar3yw$resid)
```



Maximum likelihood estimates

```
> ## Load module
> module(finmetrics)
S+FinMetrics Version 2.0.2 for Linux 2.4.21 : 2005
> ## Means:
> colMeans(mmdata.tr)
        w1      w2
 0.02792121 10.7961
> ## Center non-differences data:
> tmp.dat <- mmdata.tr
> tmp.dat[,2] <- tmp.dat[,2] - mean(tmp.dat[,2])
> colMeans(tmp.dat)
        w1            w2
 0.02792121 -1.514271e-15
```

Maximum likelihood estimates (cont'nd)

```
> mgarch(~ -1 + ar(3), ~ dvec(0,0), series=tmp.dat, armaType="full")
...[deleted]...
Convergence reached.

Call:

mgarch(formula.mean =  ~ -1 + ar(3), formula.var =  ~ dvec(0, 0),
       series = tmp.dat, armaType = "full")

Mean Equation:  ~ -1 + ar(3)

Conditional Variance Equation:  ~ dvec(0, 0)

Coefficients:
AR(1; 1, 1)  0.23213
.
.
.
```

Maximum likelihood estimates (cont'nd)

```
Coefficients:
AR(1; 1, 1)  0.23213
AR(1; 2, 1)  0.43642
AR(1; 1, 2) -0.73288
AR(1; 2, 2)  0.74650
AR(2; 1, 1) -0.17461
AR(2; 2, 1) -0.25434
AR(2; 1, 2)  0.26512
AR(2; 2, 2)  0.21456
AR(3; 1, 1)  0.07007
AR(3; 2, 1)  0.17788
AR(3; 1, 2)  0.12217
AR(3; 2, 2) -0.41734
    A(1, 1)  0.06228
    A(2, 1)  0.01473
    A(2, 2)  0.06381
```

Model Validation

- For the individual residual series; all the methods from Chapter 6 in the book
- with the extension for the cross correlation as mentioned in Chapter 8 in the book

Forecasting

The model:

$$\boldsymbol{Y}_{t+\ell} = \boldsymbol{\phi}_1 \boldsymbol{Y}_{t+\ell-1} + \ldots + \boldsymbol{\phi}_p \boldsymbol{Y}_{t+\ell-p} + \boldsymbol{\epsilon}_{t+\ell} - \boldsymbol{\theta}_1 \boldsymbol{\epsilon}_{t+\ell-1} - \ldots - \boldsymbol{\theta}_q \boldsymbol{\epsilon}_{t+\ell-q}$$

1-step:

$$\hat{\boldsymbol{Y}}_{t+1|t} = \boldsymbol{\phi}_1 \boldsymbol{Y}_{t+1-1} + \ldots + \boldsymbol{\phi}_p \boldsymbol{Y}_{t+1-p} + \mathbf{0} - \boldsymbol{\theta}_1 \boldsymbol{\epsilon}_{t+1-1} - \ldots - \boldsymbol{\theta}_q \boldsymbol{\epsilon}_{t+1-q}$$

2-step:

$$\hat{\boldsymbol{Y}}_{t+2|t} = \boldsymbol{\phi}_1 \hat{\boldsymbol{Y}}_{t+2-1|t} + \ldots + \boldsymbol{\phi}_p \boldsymbol{Y}_{t+2-p} + \mathbf{0} - \boldsymbol{\theta}_1 \mathbf{0} - \ldots - \boldsymbol{\theta}_q \boldsymbol{\epsilon}_{t+2-q}$$

and so on . . . in S-PLUS:

```
> predict(fit.ml, 10) # fit.ml from mgarch() above
```

However, this does *not* calculate the variance-covariance matrix of the forecast errors – use the hint given in the text book.