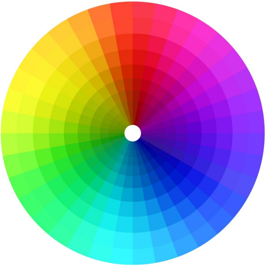
UPPSALA UNIVERSITET

Parametric methods for Rational Spectra



Per Mattsson

Systems and Control
Department of Information Technology
Uppsala University

2019-09-13

Summary from last lecture

Refined non-parametric methods

Blackman-Tukey: $$\hat{\phi}_{\mathrm{BT}}(\omega) = \sum_{k=-(M-1)}^{M-1} w(k)\hat{r}(k)e^{-i\omega k}$$

Welch: $$\hat{\phi}_{\mathrm{W}}(\omega) = \frac{1}{S}\sum_{j=1}^{S}\frac{1}{MP}\left|\sum_{t=1}^{M} v(t)y_j(t)e^{-i\omega t}\right|^2$$

- Windowed Correlogram/Periodogram (smoothing)
- Window functions (Bartlett, Hamming, Chebyshev, Kaiser)
- Bias/Variance or Resolution(main)/Leakage(side) trade-off
- Estimating continuous/line spectra

Today: Parametric methods for rational (continuous) spectra

Parametric methods

- Model the data/PSD using fewer parameters (cf. Periodogram has one for each ω, i.e. N parameters)
- If the model is poor, the result will be biased.
- Some prior knowledge needed.

Parametric methods

- Model the data/PSD using fewer parameters (cf. Periodogram has one for each ω, i.e. N parameters)
- If the model is poor, the result will be biased.
- Some prior knowledge needed.

Occam's razor or Principle of Parsimony

If two theories explain the data equally well, then the simpler theory is preferable.

Parametric methods

- Model the data/PSD using fewer parameters (cf. Periodogram has one for each ω, i.e. N parameters)
- If the model is poor, the result will be biased.
- Some prior knowledge needed.

Occam's razor or Principle of Parsimony

If two theories explain the data equally well, then the simpler theory is preferable.

- "Keep it as simple as possible, but no simpler" – Albert Einstein.

Parametric methods

- Model the data/PSD using fewer parameters (cf. Periodogram has one for each ω, i.e. N parameters)
- If the model is poor, the result will be biased.
- Some prior knowledge needed.

Occam's razor or Principle of Parsimony

If two theories explain the data equally well, then the simpler theory is preferable.

- "Keep it as simple as possible, but no simpler" – Albert Einstein.
- "With four parameters I can fit an elephant " – John von Neumann.

Parametric methods

- Model the data/PSD using fewer parameters (cf. Periodogram has one for each ω, i.e. N parameters)
- If the model is poor, the result will be biased.
- Some prior knowledge needed.

Occam's razor or Principle of Parsimony

If two theories explain the data equally well, then the simpler theory is preferable.

- "Keep it as simple as possible, but no simpler" – Albert Einstein.
- "With four parameters I can fit an elephant, and with five I can make him wiggle his trunk." – John von Neumann.

Parametric methods

- Model the data/PSD using fewer parameters (cf. Periodogram has one for each ω, i.e. N parameters)
- If the model is poor, the result will be biased.
- Some prior knowledge needed.

Occam's razor or Principle of Parsimony

If two theories explain the data equally well, then the simpler theory is preferable.

- "Keep it as simple as possible, but no simpler" – Albert Einstein.
- "With four parameters I can fit an elephant, and with five I can make him wiggle his trunk." – John von Neumann.
- Bias/variance trade-off

Parametric methods

- Model the data/PSD using fewer parameters (cf. Periodogram has one for each ω, i.e. N parameters)
- If the model is poor, the result will be biased.
- Some prior knowledge needed.

Occam's razor or Principle of Parsimony

If two theories explain the data equally well, then the simpler theory is preferable.

- "Keep it as simple as possible, but no simpler" – Albert Einstein.
- "With four parameters I can fit an elephant, and with five I can make him wiggle his trunk." – John von Neumann.
- Bias/variance trade-off
- Overfitting.

Parametric methods

- Model the data/PSD using fewer parameters (cf. Periodogram has one for each ω, i.e. N parameters)
- If the model is poor, the result will be biased.
- Some prior knowledge needed.

Occam's razor or Principle of Parsimony

If two theories explain the data equally well, then the simpler theory is preferable.

- "Keep it as simple as possible, but no simpler" – Albert Einstein.
- "With four parameters I can fit an elephant, and with five I can make him wiggle his trunk." – John von Neumann.
- Bias/variance trade-off
- Overfitting.
- Model order selection is important, but difficult (more in L7)

UPPSALA UNIVERSITET

Signals with rational spectra

UPPSALA UNIVERSITET

Rational spectra

Rational PSD

$$\phi(\omega) = \frac{\sum_{k=-m}^{m} \gamma_k e^{-i\omega k}}{\sum_{k=-n}^{n} \rho_k e^{-i\omega k}}$$

where $\gamma_{-k} = \gamma_k^*$ and $\rho_{-k} = \rho_k^*$, can approximate any **continuous** PSD for **sufficiently large** m, n

Rational spectra

Rational PSD

$$\phi(\omega) = \frac{\sum_{k=-m}^{m} \gamma_k e^{-i\omega k}}{\sum_{k=-n}^{n} \rho_k e^{-i\omega k}}$$

where $\gamma_{-k} = \gamma_k^*$ and $\rho_{-k} = \rho_k^*$, can approximate any **continuous** PSD for **sufficiently large** m, n

Spectral factorization theorem says we can write a rational $\phi(\omega)$ as

$$\phi(\omega) = \left| \frac{B(\omega)}{A(\omega)} \right|^2 \sigma^2$$

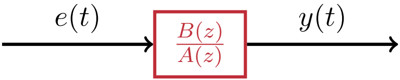
for some positive scalar σ^2, where

$$A(z) = 1 + a_1 z^{-1} + \ldots + a_n z^{-n}$$
$$B(z) = 1 + b_1 z^{-1} + \ldots + b_m z^{-m}$$

and $A(\omega) = A(z)|_{z=e^{i\omega}}$.

Signal modeling interpretation

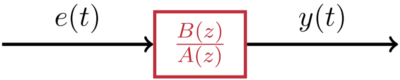
White noise $e(t)$ with variance σ^2 filtered through a linear filter:



- $y(t) = \frac{B(z)}{A(z)} e(t)$.

Signal modeling interpretation

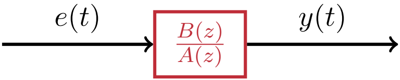
White noise $e(t)$ with variance σ^2 filtered through a linear filter:



- $y(t) = \frac{B(z)}{A(z)} e(t)$.
- Input spectrum: $\phi_e(\omega) =$

Signal modeling interpretation

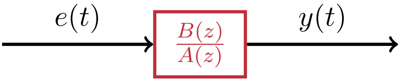
White noise $e(t)$ with variance σ^2 filtered through a linear filter:



- $y(t) = \frac{B(z)}{A(z)} e(t)$.
- Input spectrum: $\phi_e(\omega) = \sigma^2$.

Signal modeling interpretation

White noise $e(t)$ with variance σ^2 filtered through a linear filter:



- $y(t) = \frac{B(z)}{A(z)} e(t)$.
- Input spectrum: $\phi_e(\omega) = \sigma^2$.
- Output spectrum:

$$\phi_y(\omega) =$$

Signal modeling interpretation

White noise $e(t)$ with variance σ^2 filtered through a linear filter:

$$e(t) \longrightarrow \boxed{\frac{B(z)}{A(z)}} \longrightarrow y(t)$$

- $y(t) = \frac{B(z)}{A(z)} e(t)$.
- Input spectrum: $\phi_e(\omega) = \sigma^2$.
- Output spectrum:

$$\phi_y(\omega) = \left| \frac{B(\omega)}{A(\omega)} \right|^2 \sigma^2$$

Signal modeling interpretation

White noise $e(t)$ with variance σ^2 filtered through a linear filter:

$$e(t) \longrightarrow \boxed{\frac{B(z)}{A(z)}} \longrightarrow y(t)$$

- $y(t) = \frac{B(z)}{A(z)} e(t)$.
- Input spectrum: $\phi_e(\omega) = \sigma^2$.
- Output spectrum:

$$\phi_y(\omega) = \left| \frac{B(\omega)}{A(\omega)} \right|^2 \sigma^2$$

- Conclusion: A rational spectrum can be modelled as the spectrum of white noise filtered through a linear filter.

Poles, zeros and the spectrum

Linear filter:

$$y(t) = \frac{B(z)}{A(z)} e(t) = \sum_{k=0}^{\infty} h_k e(t-k), \quad h_0 = 1 \text{ in these slides}$$

Poles, zeros and the spectrum

Linear filter:

$$y(t) = \frac{B(z)}{A(z)} e(t) = \sum_{k=0}^{\infty} h_k e(t-k), \quad h_0 = 1 \text{ in these slides}$$

- Poles: Zeros of $A(z)$. Zeros: Zeros of $B(z)$.

Poles, zeros and the spectrum

Linear filter:

$$y(t) = \frac{B(z)}{A(z)}e(t) = \sum_{k=0}^{\infty} h_k e(t-k), \quad h_0 = 1 \text{ in these slides}$$

- Poles: Zeros of $A(z)$. Zeros: Zeros of $B(z)$.
- Stable if all poles lies inside the unit circle.

Poles, zeros and the spectrum

Linear filter:

$$y(t) = \frac{B(z)}{A(z)} e(t) = \sum_{k=0}^{\infty} h_k e(t-k), \quad h_0 = 1 \text{ in these slides}$$

- Poles: Zeros of $A(z)$. Zeros: Zeros of $B(z)$.
- Stable if all poles lies inside the unit circle.

Spectrum:

$$\phi_y(z) = \frac{B(z)B^*(1/z^*)}{A(z)A^*(1/z^*)}\sigma^2 \xRightarrow{z=e^{i\omega}} \phi_y(\omega) = \left|\frac{B(\omega)}{A(\omega)}\right|^2 \sigma^2$$

per.mattsson@it.uu.se

Poles, zeros and the spectrum

Linear filter:

$$y(t) = \frac{B(z)}{A(z)}e(t) = \sum_{k=0}^{\infty} h_k e(t-k), \quad h_0 = 1 \text{ in these slides}$$

- Poles: Zeros of $A(z)$. Zeros: Zeros of $B(z)$.
- Stable if all poles lies inside the unit circle.

Spectrum:

$$\phi_y(z) = \frac{B(z)B^*(1/z^*)}{A(z)A^*(1/z^*)}\sigma^2 \xRightarrow{z=e^{i\omega}} \phi_y(\omega) = \left|\frac{B(\omega)}{A(\omega)}\right|^2 \sigma^2$$

- $\phi(\omega)$ is finite for all $\omega \iff$

UPPSALA UNIVERSITET

Poles, zeros and the spectrum

Linear filter:

$$y(t) = \frac{B(z)}{A(z)} e(t) = \sum_{k=0}^{\infty} h_k e(t-k), \quad h_0 = 1 \text{ in these slides}$$

- Poles: Zeros of $A(z)$. Zeros: Zeros of $B(z)$.
- Stable if all poles lies inside the unit circle.

Spectrum:

$$\phi_y(z) = \frac{B(z)B^*(1/z^*)}{A(z)A^*(1/z^*)}\sigma^2 \xrightarrow{z=e^{i\omega}} \phi_y(\omega) = \left|\frac{B(\omega)}{A(\omega)}\right|^2 \sigma^2$$

- $\phi(\omega)$ is finite for all $\omega \iff$ No poles on the unit circle.

Poles, zeros and the spectrum

Linear filter:

$$y(t) = \frac{B(z)}{A(z)}e(t) = \sum_{k=0}^{\infty} h_k e(t-k), \quad h_0 = 1 \text{ in these slides}$$

- Poles: Zeros of $A(z)$. Zeros: Zeros of $B(z)$.
- Stable if all poles lies inside the unit circle.

Spectrum:

$$\phi_y(z) = \frac{B(z)B^*(1/z^*)}{A(z)A^*(1/z^*)}\sigma^2 \xRightarrow{z=e^{i\omega}} \phi_y(\omega) = \left|\frac{B(\omega)}{A(\omega)}\right|^2 \sigma^2$$

- $\phi(\omega)$ is finite for all $\omega \iff$ No poles on the unit circle.
- Can always choose $A(z)$ and $B(z)$ so that the poles and zeros lie inside the unit circle. (If $\phi(\omega)$ is finite)

UPPSALA UNIVERSITET

Poles, zeros and the spectrum

Linear filter:

$$y(t) = \frac{B(z)}{A(z)} e(t) = \sum_{k=0}^{\infty} h_k e(t-k), \quad h_0 = 1 \text{ in these slides}$$

- Poles: Zeros of $A(z)$. Zeros: Zeros of $B(z)$.
- Stable if all poles lies inside the unit circle.

Spectrum:

$$\phi_y(z) = \frac{B(z)B^*(1/z^*)}{A(z)A^*(1/z^*)}\sigma^2 \xRightarrow{z=e^{i\omega}} \phi_y(\omega) = \left|\frac{B(\omega)}{A(\omega)}\right|^2 \sigma^2$$

- $\phi(\omega)$ is finite for all $\omega \iff$ No poles on the unit circle.
- Can always choose $A(z)$ and $B(z)$ so that the poles and zeros lie inside the unit circle. (If $\phi(\omega)$ is finite)
- Pole at $z_1 \approx e^{i\omega_1}$ (near unit circle)

per.mattsson@it.uu.se

Poles, zeros and the spectrum

Linear filter:

$$y(t) = \frac{B(z)}{A(z)} e(t) = \sum_{k=0}^{\infty} h_k e(t-k), \quad h_0 = 1 \text{ in these slides}$$

- Poles: Zeros of $A(z)$. Zeros: Zeros of $B(z)$.
- Stable if all poles lies inside the unit circle.

Spectrum:

$$\phi_y(z) = \frac{B(z)B^*(1/z^*)}{A(z)A^*(1/z^*)}\sigma^2 \xrightarrow{z=e^{i\omega}} \phi_y(\omega) = \left|\frac{B(\omega)}{A(\omega)}\right|^2 \sigma^2$$

- $\phi(\omega)$ is finite for all $\omega \iff$ No poles on the unit circle.
- Can always choose $A(z)$ and $B(z)$ so that the poles and zeros lie inside the unit circle. (If $\phi(\omega)$ is finite)
- Pole at $z_1 \approx e^{i\omega_1}$ (near unit circle) $\implies \phi(\omega_1)$ large.

Poles, zeros and the spectrum

Linear filter:

$$y(t) = \frac{B(z)}{A(z)} e(t) = \sum_{k=0}^{\infty} h_k e(t-k), \quad h_0 = 1 \text{ in these slides}$$

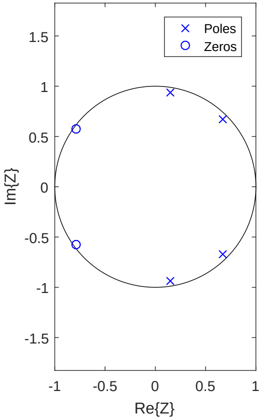
- Poles: Zeros of $A(z)$. Zeros: Zeros of $B(z)$.
- Stable if all poles lies inside the unit circle.

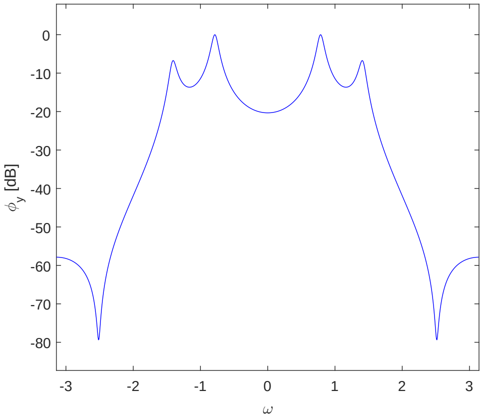
Spectrum:

$$\phi_y(z) = \frac{B(z)B^*(1/z^*)}{A(z)A^*(1/z^*)}\sigma^2 \xRightarrow{z=e^{i\omega}} \phi_y(\omega) = \left|\frac{B(\omega)}{A(\omega)}\right|^2 \sigma^2$$

- $\phi(\omega)$ is finite for all $\omega \iff$ No poles on the unit circle.
- Can always choose $A(z)$ and $B(z)$ so that the poles and zeros lie inside the unit circle. (If $\phi(\omega)$ is finite)
- Pole at $z_1 \approx e^{i\omega_1}$ (near unit circle) $\implies \phi(\omega_1)$ large.
- Zero at $z_1 \approx e^{i\omega_1}$ (near unit circle) $\implies \phi(\omega_1)$ small.

Example


Poles
Zeros
1.5
1
0.5
0
-0.5
-1
-1.5
Im{Z}
-1
-0.5
0
0.5
1
Re{Z}


0
-10
-20
-30
-40
-50
-60
-70
-80
ϕ_y [dB]
-3
-2
-1
0
1
2
3
ω

AR(MA) (1)

Autoregressive moving average (pole-zero model)

$$\text{ARMA: } A(z)y(t) = B(z)e(t)$$

Special cases

$$\text{AR: } A(z)y(t) = e(t)$$

$$\text{MA: } y(t) = B(z)e(t)$$

Estimate parameters: $\{a_i\}_{i=1}^{n}, \{b_i\}_{i=1}^{m}, \sigma^2 \implies$ PSD estimate

per.mattsson@it.uu.se

AR(MA) (1)

Autoregressive moving average (pole-zero model)

$$\text{ARMA: } A(z)y(t) = B(z)e(t)$$

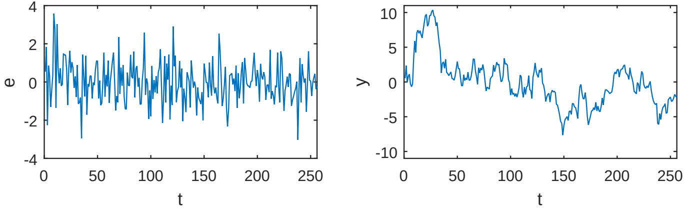
Special cases

$$\text{AR: } A(z)y(t) = e(t)$$

$$\text{MA: } y(t) = B(z)e(t)$$

Estimate parameters: $\{a_i\}_{i=1}^{n}, \{b_i\}_{i=1}^{m}, \sigma^2 \implies$ PSD estimate

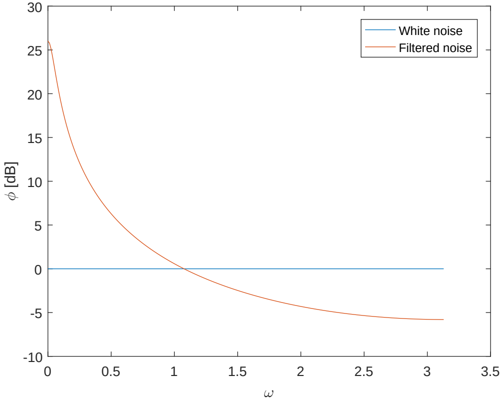
AR Ex.: `N=256; e=randn(N,1); y=filter(1,[1 -0.95],e);`



AR(MA) (2)

```
N=256; e=randn(N,1); y=filter(1,[1 -0.95],e);
```

Single real-valued pole ($A(z) = 0$) at $z = 0.95$, i.e. close to unit circle at $\omega = 0$



Covariance structure

ARMA can be written out as (with $b_0 = 1$)

$$y(t) + \sum_{i=1}^{n} a_i y(t-i) = \sum_{j=0}^{m} b_j e(t-j)$$

Covariance structure

ARMA can be written out as (with $b_0 = 1$)

$$y(t) + \sum_{i=1}^{n} a_i y(t-i) = \sum_{j=0}^{m} b_j e(t-j)$$

Multiplying by $y^*(t-k)$ and taking the expectation gives

Covariance structure

ARMA can be written out as (with $b_0 = 1$)

$$y(t) + \sum_{i=1}^{n} a_i y(t-i) = \sum_{j=0}^{m} b_j e(t-j)$$

Multiplying by $y^*(t-k)$ and taking the expectation gives

$$r(k) + \sum_{i=1}^{n} a_i r(k-i) = \sigma^2 \sum_{j=0}^{m} b_j h^*_{j-k}$$

Covariance structure

ARMA can be written out as (with $b_0 = 1$)

$$y(t) + \sum_{i=1}^{n} a_i y(t-i) = \sum_{j=0}^{m} b_j e(t-j)$$

Multiplying by $y^*(t-k)$ and taking the expectation gives

$$r(k) + \sum_{i=1}^{n} a_i r(k-i) = \sigma^2 \sum_{j=0}^{m} b_j h_{j-k}^*$$

But $h_s = 0$ for $s < 0$ (causality), which gives

Covariance structure

ARMA can be written out as (with $b_0 = 1$)

$$y(t) + \sum_{i=1}^{n} a_i y(t-i) = \sum_{j=0}^{m} b_j e(t-j)$$

Multiplying by $y^*(t-k)$ and taking the expectation gives

$$r(k) + \sum_{i=1}^{n} a_i r(k-i) = \sigma^2 \sum_{j=0}^{m} b_j h^*_{j-k}$$

But $h_s = 0$ for $s < 0$ (causality), which gives

Covariance structure

$$r(k) + \sum_{i=1}^{n} a_i r(k-i) = 0, \quad \text{for } k > m$$

per.mattsson@it.uu.se

Methods for AR-models

AR: Yule-Walker equations

AR-model: $A(z)y(t) = e(t)$. Here $m = 0$, and $B(z) = b_0 = 1$, so

$$r(0) + \sum_{i=1}^{n} a_i r(-i) =$$

per.mattsson@it.uu.se

AR: Yule-Walker equations

AR-model: $A(z)y(t) = e(t)$. Here $m = 0$, and $B(z) = b_0 = 1$, so

$$r(0) + \sum_{i=1}^{n} a_i r(-i) = \sigma^2 b_0 h_0 =$$

per.mattsson@it.uu.se

UPPSALA UNIVERSITET

AR: Yule-Walker equations

AR-model: $A(z)y(t) = e(t)$. Here $m = 0$, and $B(z) = b_0 = 1$, so

$$r(0) + \sum_{i=1}^{n} a_i r(-i) = \sigma^2 b_0 h_0 = \sigma^2,$$

AR: Yule-Walker equations

AR-model: $A(z)y(t) = e(t)$. Here $m = 0$, and $B(z) = b_0 = 1$, so

$$r(0) + \sum_{i=1}^{n} a_i r(-i) = \sigma^2 b_0 h_0 = \sigma^2,$$

$$r(k) + \sum_{i=1}^{n} a_i r(k-i) = 0, \quad k = 1, \dots, n$$

These are called the **Yule-Walker equations**.

AR: Yule-Walker equations

AR-model: $A(z)y(t) = e(t)$. Here $m = 0$, and $B(z) = b_0 = 1$, so

$$
\begin{aligned}
r(0) + \sum_{i=1}^{n} a_i r(-i) &= \sigma^2 b_0 h_0 = \sigma^2, \\
r(k) + \sum_{i=1}^{n} a_i r(k-i) &= 0, \quad k = 1, \dots, n
\end{aligned}
$$

These are called the **Yule-Walker equations**.
On matrix form:

$$
\begin{bmatrix} r(0) & r(-1) & \cdots & r(-n) \\ r(1) & r(0) & \cdots & r(-n+1) \\ \vdots & & \ddots & \vdots \\ r(n) & r(n-1) & \cdots & r(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \sigma^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}
$$

Yule-Walker method

- Want to find an estimate of $\theta = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix}^\top$.

Yule-Walker method

- Want to find an estimate of $\theta = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix}^\top$.
- Idea: Replace all $r(k)$ with **standard biased estimate** $\hat{r}(k)$.

Yule-Walker method

- Want to find an estimate of $\theta = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix}^\top$.
- Idea: Replace all $r(k)$ with **standard biased estimate** $\hat{r}(k)$.
- If we remove the first row we get

$$\underbrace{\begin{bmatrix} \hat{r}(1) \\ \vdots \\ \hat{r}(n) \end{bmatrix}}_{\hat{r}_n} + \underbrace{\begin{bmatrix} \hat{r}(0) & \cdots & \hat{r}(-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(n-1) & \cdots & \hat{r}(0) \end{bmatrix}}_{\hat{R}_n} \underbrace{\begin{bmatrix} \hat{a}_1 \\ \vdots \\ \hat{a}_n \end{bmatrix}}_{\hat{\theta}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

UPPSALA UNIVERSITET

Yule-Walker method

- Want to find an estimate of $\theta = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix}^\top$.
- Idea: Replace all $r(k)$ with **standard biased estimate** $\hat{r}(k)$.
- If we remove the first row we get

$$\underbrace{\begin{bmatrix} \hat{r}(1) \\ \vdots \\ \hat{r}(n) \end{bmatrix}}_{\hat{r}_n} + \underbrace{\begin{bmatrix} \hat{r}(0) & \cdots & \hat{r}(-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(n-1) & \cdots & \hat{r}(0) \end{bmatrix}}_{\hat{R}_n} \underbrace{\begin{bmatrix} \hat{a}_1 \\ \vdots \\ \hat{a}_n \end{bmatrix}}_{\hat{\theta}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Solution: $\hat{\theta} =$

UPPSALA UNIVERSITET

Yule-Walker method

- Want to find an estimate of $\theta = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix}^\top$.
- Idea: Replace all $r(k)$ with **standard biased estimate** $\hat{r}(k)$.
- If we remove the first row we get

$$\underbrace{\begin{bmatrix} \hat{r}(1) \\ \vdots \\ \hat{r}(n) \end{bmatrix}}_{\hat{r}_n} + \underbrace{\begin{bmatrix} \hat{r}(0) & \cdots & \hat{r}(-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(n-1) & \cdots & \hat{r}(0) \end{bmatrix}}_{\hat{R}_n} \underbrace{\begin{bmatrix} \hat{a}_1 \\ \vdots \\ \hat{a}_n \end{bmatrix}}_{\hat{\theta}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Solution: $\hat{\theta} = -\hat{R}_n^{-1}\hat{r}_n$.

Yule-Walker method

- Want to find an estimate of $\theta = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix}^\top$.
- Idea: Replace all $r(k)$ with **standard biased estimate** $\hat{r}(k)$.
- If we remove the first row we get

$$\underbrace{\begin{bmatrix} \hat{r}(1) \\ \vdots \\ \hat{r}(n) \end{bmatrix}}_{\hat{r}_n} + \underbrace{\begin{bmatrix} \hat{r}(0) & \cdots & \hat{r}(-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(n-1) & \cdots & \hat{r}(0) \end{bmatrix}}_{\hat{R}_n} \underbrace{\begin{bmatrix} \hat{a}_1 \\ \vdots \\ \hat{a}_n \end{bmatrix}}_{\hat{\theta}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Solution: $\hat{\theta} = -\hat{R}_n^{-1}\hat{r}_n$.
- Using $\hat{\theta}$, the first row now gives

$$\hat{\sigma}^2 =$$

Yule-Walker method

- Want to find an estimate of $\theta = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix}^\top$.
- Idea: Replace all $r(k)$ with **standard biased estimate** $\hat{r}(k)$.
- If we remove the first row we get

$$\underbrace{\begin{bmatrix} \hat{r}(1) \\ \vdots \\ \hat{r}(n) \end{bmatrix}}_{\hat{r}_n} + \underbrace{\begin{bmatrix} \hat{r}(0) & \cdots & \hat{r}(-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(n-1) & \cdots & \hat{r}(0) \end{bmatrix}}_{\hat{R}_n} \underbrace{\begin{bmatrix} \hat{a}_1 \\ \vdots \\ \hat{a}_n \end{bmatrix}}_{\hat{\theta}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Solution: $\hat{\theta} = -\hat{R}_n^{-1}\hat{r}_n$.
- Using $\hat{\theta}$, the first row now gives

$$\hat{\sigma}^2 = \hat{r}(0) + \begin{bmatrix} \hat{r}(-1) & \cdots & \hat{r}(-n) \end{bmatrix} \hat{\theta}.$$

AR as linear regression

The AR-model $A(z)y(t) = e(t)$ can be written as

$$e(t) = y(t) + \sum_{i=1}^{n} a_i y(t-i) =$$

AR as linear regression

The AR-model $A(z)y(t) = e(t)$ can be written as

$$e(t) = y(t) + \sum_{i=1}^{n} a_i y(t-i) = y(t) - \varphi^\top(t)\theta,$$

where

$$\varphi(t) = \begin{bmatrix} -y(t-1) \\ \vdots \\ -y(t-n) \end{bmatrix}, \quad \theta = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}.$$

AR as linear regression

The AR-model $A(z)y(t) = e(t)$ can be written as

$$e(t) = y(t) + \sum_{i=1}^{n} a_i y(t-i) = y(t) - \varphi^\top(t)\theta,$$

where

$$\varphi(t) = \begin{bmatrix} -y(t-1) \\ \vdots \\ -y(t-n) \end{bmatrix}, \quad \theta = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}.$$

- Prediction error: If we predict $y(t)$ using $\hat{y}(t) = \varphi^\top(t)\theta$, then $e(t) = y(t) - \hat{y}(t)$ is the prediction error.

AR as linear regression

The AR-model $A(z)y(t) = e(t)$ can be written as

$$e(t) = y(t) + \sum_{i=1}^{n} a_i y(t-i) = y(t) - \varphi^\top(t)\theta,$$

where

$$\varphi(t) = \begin{bmatrix} -y(t-1) \\ \vdots \\ -y(t-n) \end{bmatrix}, \quad \theta = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}.$$

- Prediction error: If we predict $y(t)$ using $\hat{y}(t) = \varphi^\top(t)\theta$, then $e(t) = y(t) - \hat{y}(t)$ is the prediction error.
- Note: In the book $\varphi^\top(t)$ is defined without the minus sign, so we have $e(t) = y(t) + \varphi^\top(t)\theta$.

AR: The least squares method

- Idea: Estimate θ by minimizing the prediction errors

$$\hat{\theta} = \arg\min_{\theta} \sum_{t=1}^{N} |e(t)|^2 =$$

UPPSALA UNIVERSITET

AR: The least squares method

- Idea: Estimate θ by minimizing the prediction errors

$$\hat{\theta} = \arg\min_{\theta} \sum_{t=1}^{N} |e(t)|^2 = \arg\min_{\theta} \sum_{t=1}^{N} |y - \varphi^\top(t)\theta|^2$$

$$=$$

UPPSALA UNIVERSITET

AR: The least squares method

- Idea: Estimate θ by minimizing the prediction errors

$$\hat{\theta} = \arg\min_{\theta} \sum_{t=1}^{N} |e(t)|^2 = \arg\min_{\theta} \sum_{t=1}^{N} |y - \varphi^\top(t)\theta|^2$$
$$= \arg\min_{\theta} \|y - \Phi\theta\|_2^2,$$

where

$$y = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix}, \quad \Phi = \begin{bmatrix} \varphi^\top(1) \\ \vdots \\ \varphi^\top(N) \end{bmatrix}.$$

AR: The least squares method

- Idea: Estimate θ by minimizing the prediction errors

$$\hat{\theta} = \arg\min_{\theta} \sum_{t=1}^{N} |e(t)|^2 = \arg\min_{\theta} \sum_{t=1}^{N} |y - \varphi^\top(t)\theta|^2$$
$$= \arg\min_{\theta} \|y - \Phi\theta\|_2^2,$$

where

$$y = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix}, \quad \Phi = \begin{bmatrix} \varphi^\top(1) \\ \vdots \\ \varphi^\top(N) \end{bmatrix}.$$

- Solution:

$$\hat{\theta} =$$

per.mattsson@it.uu.se

AR: The least squares method

- Idea: Estimate θ by minimizing the prediction errors

$$\hat{\theta} = \arg\min_{\theta} \sum_{t=1}^{N} |e(t)|^2 = \arg\min_{\theta} \sum_{t=1}^{N} |y - \varphi^\top(t)\theta|^2$$
$$= \arg\min_{\theta} \|y - \Phi\theta\|_2^2,$$

where

$$y = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix}, \quad \Phi = \begin{bmatrix} \varphi^\top(1) \\ \vdots \\ \varphi^\top(N) \end{bmatrix}.$$

- Solution:

$$\hat{\theta} = (\Phi^*\Phi)^{-1}\Phi^* y.$$

per.mattsson@it.uu.se

AR: The least squares method

- Idea: Estimate θ by minimizing the prediction errors

$$\hat{\theta} = \arg\min_{\theta} \sum_{t=1}^{N} |e(t)|^2 = \arg\min_{\theta} \sum_{t=1}^{N} |y - \varphi^\top(t)\theta|^2$$
$$= \arg\min_{\theta} \|y - \Phi\theta\|_2^2,$$

where

$$y = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix}, \quad \Phi = \begin{bmatrix} \varphi^\top(1) \\ \vdots \\ \varphi^\top(N) \end{bmatrix}.$$

- Solution:

$$\hat{\theta} = (\Phi^*\Phi)^{-1}\Phi^* y.$$

- Again, the book uses $Y = -\Phi$, so $\hat{\theta} = -(Y^*Y)^{-1}Y^*y$.

What about negative t?

What about negative t?

- If, for example, $n = 3$ then

$$\varphi^\top(1) = \begin{bmatrix} y(0) & y(-1) & y(-2) \end{bmatrix}$$
$$\varphi^\top(2) = \begin{bmatrix} y(1) & y(0) & y(-1) \end{bmatrix}$$
$$\varphi^\top(3) = \begin{bmatrix} y(2) & y(1) & y(0) \end{bmatrix}$$

What about negative t?

- If, for example, $n = 3$ then

$$\begin{aligned}
\varphi^\top(1) &= \begin{bmatrix} y(0) & y(-1) & y(-2) \end{bmatrix} \\
\varphi^\top(2) &= \begin{bmatrix} y(1) & y(0) & y(-1) \end{bmatrix} \\
\varphi^\top(3) &= \begin{bmatrix} y(2) & y(1) & y(0) \end{bmatrix}
\end{aligned}$$

- We have only measured $y(t)$ for $t \geq 1$. How to handle this?

What about negative t?

- If, for example, $n = 3$ then

$$\varphi^\top(1) = \begin{bmatrix} y(0) & y(-1) & y(-2) \end{bmatrix}$$
$$\varphi^\top(2) = \begin{bmatrix} y(1) & y(0) & y(-1) \end{bmatrix}$$
$$\varphi^\top(3) = \begin{bmatrix} y(2) & y(1) & y(0) \end{bmatrix}$$

- We have only measured $y(t)$ for $t \geq 1$. How to handle this?
- Option 1: Let $y(t) = 0$ for $t \leq 0$.

What about negative t?

- If, for example, $n = 3$ then

$$\varphi^\top(1) = \begin{bmatrix} y(0) & y(-1) & y(-2) \end{bmatrix}$$
$$\varphi^\top(2) = \begin{bmatrix} y(1) & y(0) & y(-1) \end{bmatrix}$$
$$\varphi^\top(3) = \begin{bmatrix} y(2) & y(1) & y(0) \end{bmatrix}$$

- We have only measured $y(t)$ for $t \geq 1$. How to handle this?
- Option 1: Let $y(t) = 0$ for $t \leq 0$.
- Option 2: Skip the first n rows in Y and Φ.

What about negative t?

- If, for example, $n = 3$ then

$$\begin{aligned}
\varphi^\top(1) &= \begin{bmatrix} y(0) & y(-1) & y(-2) \end{bmatrix} \\
\varphi^\top(2) &= \begin{bmatrix} y(1) & y(0) & y(-1) \end{bmatrix} \\
\varphi^\top(3) &= \begin{bmatrix} y(2) & y(1) & y(0) \end{bmatrix}
\end{aligned}$$

- We have only measured $y(t)$ for $t \geq 1$. How to handle this?
- Option 1: Let $y(t) = 0$ for $t \leq 0$.
- Option 2: Skip the first n rows in Y and Φ.
- Could also add rows corresponding to e.g. $t = N+1, \ldots, N+n$ and assume that $y(t) = 0$ at these rows.

AR: The least squares method

$$
y = \left[\begin{array}{c}
y(1) \\
y(2) \\
\vdots \\
\hline
y(n+1) \\
y(n+2) \\
\vdots \\
y(N) \\
\hline
0 \\
0 \\
\vdots \\
0
\end{array}\right], \quad
Y = \left[\begin{array}{cccc}
0 & 0 & \ldots & 0 \\
y(1) & 0 & & \vdots \\
\vdots & \ddots & \ddots & 0 \\
\hline
y(n) & y(n-1) & \cdots & y(1) \\
y(n+1) & y(n) & \cdots & y(2) \\
\vdots & & & \vdots \\
y(N-1) & y(N-2) & \cdots & y(N-n) \\
\hline
y(N) & y(N-1) & \cdots & y(N-n+1) \\
0 & y(N) & & \\
\vdots & \ddots & \ddots & \vdots \\
0 & \ldots & 0 & y(N)
\end{array}\right]
$$

Note: With notation used in these slides $\Phi = -Y$.

AR: The least squares method

$$
y = \left[\begin{array}{c} y(1) \\ y(2) \\ \vdots \\ \hline y(n+1) \\ y(n+2) \\ \vdots \\ y(N) \\ \hline 0 \\ 0 \\ \vdots \\ 0 \end{array}\right], \quad Y = \left[\begin{array}{cccc} 0 & 0 & \dots & 0 \\ y(1) & 0 & & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \hline y(n) & y(n-1) & \cdots & y(1) \\ y(n+1) & y(n) & \cdots & y(2) \\ \vdots & & & \vdots \\ y(N-1) & y(N-2) & \cdots & y(N-n) \\ \hline y(N) & y(N-1) & \cdots & y(N-n+1) \\ 0 & y(N) & & \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & y(N) \end{array}\right]
$$

Note: With notation used in these slides $\Phi = -Y$.

- Autocorrelation method: Use full matrices. Equiv. to YW.

AR: The least squares method

$$
y = \left[\begin{array}{c} y(1) \\ y(2) \\ \vdots \\ \hline y(n+1) \\ y(n+2) \\ \vdots \\ y(N) \\ \hline 0 \\ 0 \\ \vdots \\ 0 \end{array}\right], \quad Y = \left[\begin{array}{cccc} 0 & 0 & \dots & 0 \\ y(1) & 0 & & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \hline y(n) & y(n-1) & \cdots & y(1) \\ y(n+1) & y(n) & \cdots & y(2) \\ \vdots & & & \vdots \\ y(N-1) & y(N-2) & \cdots & y(N-n) \\ \hline y(N) & y(N-1) & \cdots & y(N-n+1) \\ 0 & y(N) & & \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & y(N) \end{array}\right]
$$

Note: With notation used in these slides $\Phi = -Y$.

- Autocorrelation method: Use full matrices. Equiv. to YW.
- LS method: Only keep middle part. Approx YW.

UPPSALA UNIVERSITET

Methods for ARMA-models

ARMA-models

- Sharp peaks captured by AR-part, and deep valleys by MA-part:
$$A(q)y(t) = B(q)e(t).$$

ARMA-models

- Sharp peaks captured by AR-part, and deep valleys by MA-part:
$$A(q)y(t) = B(q)e(t).$$
- Much harder to estimate an ARMA-model!

ARMA-models

- Sharp peaks captured by AR-part, and deep valleys by MA-part:
$$A(q)y(t) = B(q)e(t).$$
- Much harder to estimate an ARMA-model!
- Here two methods that often works will be discussed.

ARMA-models

- Sharp peaks captured by AR-part, and deep valleys by MA-part:
$$A(q)y(t) = B(q)e(t).$$
- Much harder to estimate an ARMA-model!
- Here two methods that often works will be discussed.
- Many more estimators in the literature.

The two-stage LS method

We can write the ARMA-model as

$$e(t) = y(t) - \varphi^\top(t)\theta$$

with

$$\varphi(t) = \qquad\qquad \theta =$$

The two-stage LS method

We can write the ARMA-model as

$$e(t) = y(t) - \varphi^\top(t)\theta$$

with

$$\varphi(t) = \begin{bmatrix} -y(t-1) \\ \vdots \\ -y(t-n) \\ -e(t-1) \\ \vdots \\ -e(t-m) \end{bmatrix}, \quad \theta = \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_m \end{bmatrix}.$$

The two-stage LS method

We can write the ARMA-model as

$$e(t) = y(t) - \varphi^\top(t)\theta$$

with

$$\varphi(t) = \begin{bmatrix} -y(t-1) \\ \vdots \\ -y(t-n) \\ -e(t-1) \\ \vdots \\ -e(t-m) \end{bmatrix}, \quad \theta = \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_m \end{bmatrix}.$$

- Idea: Minimize $\sum_{t=1}^{N} |e(t)|^2 = \|y - \Phi\theta\|_2^2$.

per.mattsson@it.uu.se

The two-stage LS method

We can write the ARMA-model as

$$e(t) = y(t) - \varphi^\top(t)\theta$$

with

$$\varphi(t) = \begin{bmatrix} -y(t-1) \\ \vdots \\ -y(t-n) \\ -e(t-1) \\ \vdots \\ -e(t-m) \end{bmatrix}, \quad \theta = \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_m \end{bmatrix}.$$

- Idea: Minimize $\sum_{t=1}^{N} |e(t)|^2 = \|y - \Phi\theta\|_2^2$.
- Problem: Can't create Φ since we do not have $e(t)$.

The two-stage LS method

We can write the ARMA-model as

$$e(t) = y(t) - \varphi^\top(t)\theta$$

with

$$\varphi(t) = \begin{bmatrix} -y(t-1) \\ \vdots \\ -y(t-n) \\ -e(t-1) \\ \vdots \\ -e(t-m) \end{bmatrix}, \quad \theta = \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_m \end{bmatrix}.$$

- Idea: Minimize $\sum_{t=1}^{N} |e(t)|^2 = \|y - \Phi\theta\|_2^2$.
- Problem: Can't create Φ since we do not have $e(t)$.
- Potential solution: First estimate $e(t)$, then use LS to estimate θ.

The two-stage LS method

- We can assume that the ARMA-model is minimum phase.
- Then the ARMA-model can be written as an infinite dimensional AR-model

$$e(t) = \frac{A(q)}{B(q)}y(t) = y(t) + \alpha_1 y(t-1) + \alpha_2 y(t-2) + \cdots ,$$

where $\alpha_k \to 0$ as $k \to \infty$.

The two-stage LS method

- We can assume that the ARMA-model is minimum phase.
- Then the ARMA-model can be written as an infinite dimensional AR-model

$$e(t) = \frac{A(q)}{B(q)} y(t) = y(t) + \alpha_1 y(t-1) + \alpha_2 y(t-2) + \cdots,$$

  where $\alpha_k \to 0$ as $k \to \infty$.
- For large enough K we should be able to approximate the ARMA-model with an AR-model of order K.

The two-stage LS method

- We can assume that the ARMA-model is minimum phase.
- Then the ARMA-model can be written as an infinite dimensional AR-model

$$e(t) = \frac{A(q)}{B(q)} y(t) = y(t) + \alpha_1 y(t-1) + \alpha_2 y(t-2) + \cdots ,$$

  where $\alpha_k \to 0$ as $k \to \infty$.
- For large enough K we should be able to approximate the ARMA-model with an AR-model of order K.
- Potential problems:
  - K should not be too large with respect to N (overfitting).

The two-stage LS method

- We can assume that the ARMA-model is minimum phase.
- Then the ARMA-model can be written as an infinite dimensional AR-model

$$e(t) = \frac{A(q)}{B(q)} y(t) = y(t) + \alpha_1 y(t-1) + \alpha_2 y(t-2) + \cdots ,$$

  where $\alpha_k \to 0$ as $k \to \infty$.
- For large enough K we should be able to approximate the ARMA-model with an AR-model of order K.
- Potential problems:
  - K should not be too large with respect to N (overfitting).
  - With **zeros close to the unit circle** (deep valleys in the PSD), K must be very large to keep the bias errors low.

The two-stage LS method

- **Step 1:** Choose a sufficiently large K, and estimate an AR-model of order K. Let

$$\hat{e}(t) = y(t) + \sum_{k=1}^{K} \hat{\alpha}_k y(t-k)$$

The two-stage LS method

- **Step 1:** Choose a sufficiently large K, and estimate an AR-model of order K. Let

$$\hat{e}(t) = y(t) + \sum_{k=1}^{K} \hat{\alpha}_k y(t-k)$$

- **Step 2:** Estimate the ARMA-model by minimizing

$$\|y - \hat{\Phi}\theta\|_2^2$$

where we get $\hat{\Phi}$ by replacing $e(t)$ with $\hat{e}(t)$.

The two-stage LS method

- **Step 1:** Choose a sufficiently large K, and estimate an AR-model of order K. Let

$$\hat{e}(t) = y(t) + \sum_{k=1}^{K} \hat{\alpha}_k y(t-k)$$

- **Step 2:** Estimate the ARMA-model by minimizing

$$\|y - \hat{\Phi}\theta\|_2^2$$

where we get $\hat{\Phi}$ by replacing $e(t)$ with $\hat{e}(t)$. Can use either YW-method or LS method for this.

The two-stage LS method

- **Step 1:** Choose a sufficiently large K, and estimate an AR-model of order K. Let

$$\hat{e}(t) = y(t) + \sum_{k=1}^{K} \hat{\alpha}_k y(t-k)$$

- **Step 2:** Estimate the ARMA-model by minimizing

$$\|y - \hat{\Phi}\theta\|_2^2$$

  where we get $\hat{\Phi}$ by replacing $e(t)$ with $\hat{e}(t)$. Can use either YW-method or LS method for this.
- The variance σ^2 is estimated by the sample variance of $\hat{e}(t)$.

per.mattsson@it.uu.se

UPPSALA UNIVERSITET

The two-stage LS method

- **Step 1:** Choose a sufficiently large K, and estimate an AR-model of order K. Let

$$\hat{e}(t) = y(t) + \sum_{k=1}^{K} \hat{\alpha}_k y(t-k)$$

- **Step 2:** Estimate the ARMA-model by minimizing

$$\|y - \hat{\Phi}\theta\|_2^2$$

  where we get $\hat{\Phi}$ by replacing $e(t)$ with $\hat{e}(t)$. Can use either YW-method or LS method for this.
- The variance σ^2 is estimated by the sample variance of $\hat{e}(t)$.
- Finally, the estimate of the spectrum is

$$\hat{\phi}(\omega) = \left| \frac{\hat{B}(\omega)}{\hat{A}(\omega)} \right|^2 \hat{\sigma}^2.$$

UPPSALA UNIVERSITET

Modified YW

- We want to estimate $A(q)$ and $B(q)$ in

$$A(q)y(k) = B(q)e(k)$$

UPPSALA UNIVERSITET

Modified YW

- We want to estimate $A(q)$ and $B(q)$ in

$$A(q)y(k) = B(q)e(k)$$

- From earlier in this lecture

$$r(k) + \sum_{i=1}^{n} a_i r(k-i) = 0, \quad k > m.$$

Modified YW

- We want to estimate $A(q)$ and $B(q)$ in

$$A(q)y(k) = B(q)e(k)$$

- From earlier in this lecture

$$r(k) + \sum_{i=1}^{n} a_i r(k-i) = 0, \quad k > m.$$

- Idea: Use these equations to estimate $A(q)$.

Modified YW

- We want to estimate $A(q)$ and $B(q)$ in

$$A(q)y(k) = B(q)e(k)$$

- From earlier in this lecture

$$r(k) + \sum_{i=1}^{n} a_i r(k-i) = 0, \quad k > m.$$

- Idea: Use these equations to estimate $A(q)$. We can then use $\hat{A}(q)$ to estimate $B(q)$.

UPPSALA UNIVERSITET

Estimating the AR-part using modified YW

- Replace $r(k)$ with estimates $\hat{r}(k)$ to get,

$$\hat{r}(k) + \sum_{i=1}^{n} a_i \hat{r}(k-i) = 0, \quad k > m.$$

per.mattsson@it.uu.se

Estimating the AR-part using modified YW

- Replace $r(k)$ with estimates $\hat{r}(k)$ to get,

$$\hat{r}(k) + \sum_{i=1}^{n} a_i \hat{r}(k-i) = 0, \quad k > m.$$

- or on matrix form for $k = m+1, \dots, m+M$,

$$\begin{bmatrix} \hat{r}(m+1) \\ \vdots \\ \hat{r}(m+M) \end{bmatrix} + \begin{bmatrix} \hat{r}(m) & \cdots & \hat{r}(m-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(m+M-1) & \cdots & \hat{r}(m-n+M) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

UPPSALA UNIVERSITET

Estimating the AR-part using modified YW

- Replace $r(k)$ with estimates $\hat{r}(k)$ to get,

$$\hat{r}(k) + \sum_{i=1}^{n} a_i \hat{r}(k-i) = 0, \quad k > m.$$

- or on matrix form for $k = m+1, \ldots, m+M$,

$$\begin{bmatrix} \hat{r}(m+1) \\ \vdots \\ \hat{r}(m+M) \end{bmatrix} + \begin{bmatrix} \hat{r}(m) & \cdots & \hat{r}(m-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(m+M-1) & \cdots & \hat{r}(m-n+M) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Can estimate $a_1, \ldots, a_n$ with least squares.

Estimating the AR-part using modified YW

- Replace $r(k)$ with estimates $\hat{r}(k)$ to get,

$$\hat{r}(k) + \sum_{i=1}^{n} a_i \hat{r}(k-i) = 0, \quad k > m.$$

- or on matrix form for $k = m+1, \ldots, m+M$,

$$\begin{bmatrix} \hat{r}(m+1) \\ \vdots \\ \hat{r}(m+M) \end{bmatrix} + \begin{bmatrix} \hat{r}(m) & \cdots & \hat{r}(m-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(m+M-1) & \cdots & \hat{r}(m-n+M) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Can estimate $a_1, \ldots, a_n$ with least squares.
- User choice: How large should M be?

Estimating the AR-part using modified YW

- Replace $r(k)$ with estimates $\hat{r}(k)$ to get,

$$\hat{r}(k) + \sum_{i=1}^{n} a_i \hat{r}(k-i) = 0, \quad k > m.$$

- or on matrix form for $k = m+1, \ldots, m+M$,

$$\begin{bmatrix} \hat{r}(m+1) \\ \vdots \\ \hat{r}(m+M) \end{bmatrix} + \begin{bmatrix} \hat{r}(m) & \cdots & \hat{r}(m-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(m+M-1) & \cdots & \hat{r}(m-n+M) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Can estimate $a_1, \ldots, a_n$ with least squares.
- User choice: How large should M be?
  - $M = n$ gives square matrix. $M > n$ overdetermined system.

Estimating the AR-part using modified YW

- Replace $r(k)$ with estimates $\hat{r}(k)$ to get,

$$\hat{r}(k) + \sum_{i=1}^{n} a_i \hat{r}(k-i) = 0, \quad k > m.$$

- or on matrix form for $k = m+1, \dots, m+M$,

$$\begin{bmatrix} \hat{r}(m+1) \\ \vdots \\ \hat{r}(m+M) \end{bmatrix} + \begin{bmatrix} \hat{r}(m) & \cdots & \hat{r}(m-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(m+M-1) & \cdots & \hat{r}(m-n+M) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Can estimate $a_1, \dots, a_n$ with least squares.
- User choice: How large should M be?
  - $M = n$ gives square matrix. $M > n$ overdetermined system.
  - Larger M needed when poles and zeros are closely spaced near the unit circle (narrowband spectrum).

Estimating the AR-part using modified YW

- Replace $r(k)$ with estimates $\hat{r}(k)$ to get,

$$\hat{r}(k) + \sum_{i=1}^{n} a_i \hat{r}(k-i) = 0, \quad k > m.$$

- or on matrix form for $k = m+1, \ldots, m+M$,

$$\begin{bmatrix} \hat{r}(m+1) \\ \vdots \\ \hat{r}(m+M) \end{bmatrix} + \begin{bmatrix} \hat{r}(m) & \cdots & \hat{r}(m-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(m+M-1) & \cdots & \hat{r}(m-n+M) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Can estimate $a_1, \ldots, a_n$ with least squares.
- User choice: How large should M be?
  - $M = n$ gives square matrix. $M > n$ overdetermined system.
  - Larger M needed when poles and zeros are closely spaced near the unit circle (narrowband spectrum).
  - When poles and zeros are both well inside the unit circle, $M = n$ typically works well.

The MA-part

- ARMA-model: $A(q)y(k) = B(q)e(k)$. Let $x(k) = B(q)e(k)$.

The MA-part

- ARMA-model: $A(q)y(k) = B(q)e(k)$. Let $x(k) = B(q)e(k)$.
- Note that

$$\phi_y(\omega) = \frac{1}{|A(\omega)|^2}\phi_x(\omega).$$

The MA-part

- ARMA-model: $A(q)y(k) = B(q)e(k)$. Let $x(k) = B(q)e(k)$.
- Note that
$$\phi_y(\omega) = \frac{1}{|A(\omega)|^2}\phi_x(\omega).$$
- Let γ_k be the ACS for $x(k)$, then $\gamma_k = 0$ for $|k| > m$, and

$$\phi_x(\omega) =$$

The MA-part

- ARMA-model: $A(q)y(k) = B(q)e(k)$. Let $x(k) = B(q)e(k)$.
- Note that
$$\phi_y(\omega) = \frac{1}{|A(\omega)|^2}\phi_x(\omega).$$
- Let γ_k be the ACS for $x(k)$, then $\gamma_k = 0$ for $|k| > m$, and

$$\phi_x(\omega) = |B(\omega)|^2\sigma^2 =$$

per.mattsson@it.uu.se

UPPSALA UNIVERSITET

The MA-part

- ARMA-model: $A(q)y(k) = B(q)e(k)$. Let $x(k) = B(q)e(k)$.
- Note that
$$\phi_y(\omega) = \frac{1}{|A(\omega)|^2}\phi_x(\omega).$$
- Let γ_k be the ACS for $x(k)$, then $\gamma_k = 0$ for $|k| > m$, and
$$\phi_x(\omega) = |B(\omega)|^2\sigma^2 = \sum_{k=-m}^{m} \gamma_k e^{-i\omega k}.$$

The MA-part

- ARMA-model: $A(q)y(k) = B(q)e(k)$. Let $x(k) = B(q)e(k)$.
- Note that
$$\phi_y(\omega) = \frac{1}{|A(\omega)|^2}\phi_x(\omega).$$
- Let γ_k be the ACS for $x(k)$, then $\gamma_k = 0$ for $|k| > m$, and
$$\phi_x(\omega) = |B(\omega)|^2\sigma^2 = \sum_{k=-m}^{m} \gamma_k e^{-i\omega k}.$$
- Finally,
$$\gamma_k =$$

UPPSALA UNIVERSITET

The MA-part

- ARMA-model: $A(q)y(k) = B(q)e(k)$. Let $x(k) = B(q)e(k)$.
- Note that
$$\phi_y(\omega) = \frac{1}{|A(\omega)|^2}\phi_x(\omega).$$
- Let γ_k be the ACS for $x(k)$, then $\gamma_k = 0$ for $|k| > m$, and
$$\phi_x(\omega) = |B(\omega)|^2\sigma^2 = \sum_{k=-m}^{m} \gamma_k e^{-i\omega k}.$$
- Finally,
$$\gamma_k = E\{x(t)x^*(t-k)\} =$$

The MA-part

- ARMA-model: $A(q)y(k) = B(q)e(k)$. Let $x(k) = B(q)e(k)$.
- Note that
$$\phi_y(\omega) = \frac{1}{|A(\omega)|^2}\phi_x(\omega).$$
- Let γ_k be the ACS for $x(k)$, then $\gamma_k = 0$ for $|k| > m$, and
$$\phi_x(\omega) = |B(\omega)|^2\sigma^2 = \sum_{k=-m}^{m} \gamma_k e^{-i\omega k}.$$
- Finally,
$$\gamma_k = E\{x(t)x^*(t-k)\} = E\{[B(q)e(t)][B(q)e(t-k)]^*\}$$
$$=$$

The MA-part

- ARMA-model: $A(q)y(k) = B(q)e(k)$. Let $x(k) = B(q)e(k)$.
- Note that
$$\phi_y(\omega) = \frac{1}{|A(\omega)|^2}\phi_x(\omega).$$
- Let γ_k be the ACS for $x(k)$, then $\gamma_k = 0$ for $|k| > m$, and
$$\phi_x(\omega) = |B(\omega)|^2\sigma^2 = \sum_{k=-m}^{m} \gamma_k e^{-i\omega k}.$$
- Finally,
$$\begin{aligned}\gamma_k &= E\{x(t)x^*(t-k)\} = E\{[B(q)e(t)][B(q)e(t-k)]^*\} \\ &= E\{[A(q)y(t)][A(q)y(t-k)]^*\} \\ &= \end{aligned}$$

The MA-part

- ARMA-model: $A(q)y(k) = B(q)e(k)$. Let $x(k) = B(q)e(k)$.
- Note that
$$\phi_y(\omega) = \frac{1}{|A(\omega)|^2}\phi_x(\omega).$$
- Let γ_k be the ACS for $x(k)$, then $\gamma_k = 0$ for $|k| > m$, and
$$\phi_x(\omega) = |B(\omega)|^2\sigma^2 = \sum_{k=-m}^{m} \gamma_k e^{-i\omega k}.$$
- Finally,
$$\begin{aligned}\gamma_k &= E\{x(t)x^*(t-k)\} = E\{[B(q)e(t)][B(q)e(t-k)]^*\} \\ &= E\{[A(q)y(t)][A(q)y(t-k)]^*\} \\ &= \sum_{j=0}^{n}\sum_{p=0}^{n} a_j a_p^* r(k+p-j), \quad a_0 = 1.\end{aligned}$$

Summary of modified YW

$$A(q)y(t) = B(q)e(t)$$

Summary of modified YW

$$A(q)y(t) = B(q)e(t)$$

- Compute estimate $\hat{r}(k)$ of $r(k)$.

per.mattsson@it.uu.se

Summary of modified YW

$$A(q)y(t) = B(q)e(t)$$

- Compute estimate $\hat{r}(k)$ of $r(k)$.
- Find an estimate $\hat{A}(q)$ of $A(q)$ from

$$\begin{bmatrix} \hat{r}(m+1) \\ \vdots \\ \hat{r}(m+M) \end{bmatrix} + \begin{bmatrix} \hat{r}(m) & \cdots & \hat{r}(m-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(m+M-1) & \cdots & \hat{r}(m-n+M) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

Summary of modified YW

$$A(q)y(t) = B(q)e(t)$$

- Compute estimate $\hat{r}(k)$ of $r(k)$.
- Find an estimate $\hat{A}(q)$ of $A(q)$ from

$$\begin{bmatrix} \hat{r}(m+1) \\ \vdots \\ \hat{r}(m+M) \end{bmatrix} + \begin{bmatrix} \hat{r}(m) & \cdots & \hat{r}(m-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(m+M-1) & \cdots & \hat{r}(m-n+M) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Estimate the ACS of $B(q)e(t)$ with

$$\hat{\gamma}_k = \sum_{j=0}^{n} \sum_{p=0}^{n} \hat{a}_j \hat{a}_p^* \hat{r}(k+p-j).$$

UPPSALA UNIVERSITET

Summary of modified YW

$$A(q)y(t) = B(q)e(t)$$

- Compute estimate $\hat{r}(k)$ of $r(k)$.
- Find an estimate $\hat{A}(q)$ of $A(q)$ from

$$\begin{bmatrix} \hat{r}(m+1) \\ \vdots \\ \hat{r}(m+M) \end{bmatrix} + \begin{bmatrix} \hat{r}(m) & \cdots & \hat{r}(m-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(m+M-1) & \cdots & \hat{r}(m-n+M) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Estimate the ACS of $B(q)e(t)$ with

$$\hat{\gamma}_k = \sum_{j=0}^{n} \sum_{p=0}^{n} \hat{a}_j \hat{a}_p^* \hat{r}(k+p-j).$$

- The estimate of the ARMA spectrum is then

$$\hat{\phi}(\omega) =$$

Summary of modified YW

$$A(q)y(t) = B(q)e(t)$$

- Compute estimate $\hat{r}(k)$ of $r(k)$.
- Find an estimate $\hat{A}(q)$ of $A(q)$ from

$$\begin{bmatrix} \hat{r}(m+1) \\ \vdots \\ \hat{r}(m+M) \end{bmatrix} + \begin{bmatrix} \hat{r}(m) & \cdots & \hat{r}(m-n+1) \\ \vdots & \ddots & \vdots \\ \hat{r}(m+M-1) & \cdots & \hat{r}(m-n+M) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Estimate the ACS of $B(q)e(t)$ with

$$\hat{\gamma}_k = \sum_{j=0}^{n} \sum_{p=0}^{n} \hat{a}_j \hat{a}_p^* \hat{r}(k+p-j).$$

- The estimate of the ARMA spectrum is then

$$\hat{\phi}(\omega) = \frac{\sum_{k=-m}^{m} \hat{\gamma}_k e^{-i\omega k}}{|\hat{A}(\omega)|^2}.$$

per.mattsson@it.uu.se

UPPSALA UNIVERSITET

Summing up

- Parametric approach
- Rational spectra – AR(MA) processes
- Pole/zero placement (intuition)
- Covariance structure for AR(MA)
- Yule-Walker for AR(MA)
- Least squares for AR(MA)

Method	Computational Burden	Accuracy	Guarantee $\hat{\phi}(\omega) \geq 0$?	Use for
AR: YW or LS	low	medium	Yes	Spectra with (narrow) peaks but no valley
MA: BT	low	low-medium	No	Broadband spectra possibly with valleys but no peaks
ARMA: MYW	low-medium	medium	No	Spectra with both peaks and (not too deep) valleys
ARMA: 2-Stage LS	medium-high	medium-high	Yes	As above

MATLAB

Useful MATLAB functions:

- `lsar(y,n)`
- `lsarma(y,n,m,K)`
- `yulewalker(y,n)`
- `mywarma(y,n,m,M)`
- `armase(b,a,sig2,L)`
- `argamse(gamma,a,L)`
- `zplane(b,a)`

Note: These methods assume that the mean value of y is 0. If this is not the case for your data set, start by removing the mean!