EE538 DSPI

Module 29

Outline

- Unconstrained Least Squares AR Spectral Estimation: Sect. 12.3.4
    - Refinement: backward prediction – Sect. 11.2.2
- Two-Step ARMA Spectral Estimation – Sect. 12.3.8 – see demo ARMA2stepest.m

- consider $m = p+1$ with $x[n]$ an AR(p) process
- $$a_{p+1}(p+1) = K_{p+1} = \frac{-\left[r_{xx}[p+1] + \sum_{k=1}^{p} a_p(k)\, r_{xx}(p+1-k)\right]}{E_p}$$
- thus, for $m > p$, if $x[n]$ is AR(p):

$$a_m(k) = a_p(k), \quad k = 1, 2, \ldots, p$$

$$a_m(k) = 0, \quad k = p+1, \ldots, m$$

- $\mathcal{E}_m^{min}$ monotonically decreases until $\mathcal{E}_p^{min} = \sigma_w^2$ and $\mathcal{E}_m^{min} = \sigma_w^2$ for $m > p$

- Unconstrained Least Squares (LS) Method for Estimating AR Model Parameters
- showed that AR spectral estimation is inherently related to linear prediction
- consider replacing expected value of prediction error by a time-averaged estimate of prediction error

$$\hat{\mathcal{E}} = \frac{1}{N-p} \sum_{n=p}^{N-1} \left| x[n] + \sum_{k=1}^{p} a_k x[n-k] \right|^2$$

- recall: $x[n] \neq 0$ for $0 \leq n \leq N-1$
- rationale for limits on summation:
- <u>lower limit:</u> $n=p$. require p nonzero past values to predict $x[n]$ or have larger than "normal" error
- <u>upper limit:</u> $n=N-1$. nonsensical to predict 0 from p past values $N-1 < n < N+p-1$

- assume $x[n]$ is real-valued for sake of simplicity
- take partial wrt a_ℓ, $\ell = 1, \ldots, p$ and set equal to 0 to obtain p equations in p unknowns
- $$\frac{\partial}{\partial a_\ell}\hat{\varepsilon} = \frac{1}{N-p}\sum_{n=p}^{N-1} 2\left\{x[n] + \sum_{k=1}^{p} a_k x[n-k]\right\} x[n-\ell] = 0$$
- rearranging:

$$\sum_{k=1}^{p} a_k \frac{1}{N-p}\sum_{n=p}^{N-1} x[n-k]\,x[n-\ell] = -\frac{1}{N-p}\sum_{n=p}^{N-1} x[n]\,x[n-\ell]$$

- in matrix form: $\underset{p\times p}{\underline{\hat{R}}^{LS}}\ \underset{p\times 1}{\underline{\hat{a}}} = -\underset{p\times 1}{\underline{\hat{r}}^{LS}}$

- where:

$$\hat{R}^{LS}_{kl} = \frac{1}{N-p}\sum_{n=p}^{N-1} x[n-k]\,x^*[n-l]$$

$$k = 1, \ldots, p$$
$$l = 1, \ldots, p$$

$$\underline{\hat{r}}^{LS}_{l} = \frac{1}{N-p}\sum_{n=p}^{N-1} x[n]\,x^*[n-l]$$

- Note: $\underline{\hat{R}}^{LS}$ is <u>not</u> Toeplitz $\Rightarrow$ can't use Levinson-Durbin algorithm

- P+M Text Terminology:
- <u>Yule-Walker Method:</u> solve $\underline{R}\,\underline{a} = -\underline{r}$ where $\underline{R}$ is Toeplitz-Hermitian formed from biased time-avg'd estimates of autocorrelation sequence
- <u>Unconstrained LS:</u> solve $\underline{\hat{R}}^{LS}\underline{\hat{a}} = -\underline{\hat{r}}^{LS}$ where $\underline{\hat{R}}^{LS}$ has elements as prescribed previously - not Toeplitz!

- Unconstrained LS generally performs better than Yule-Walker
- especially true when the actual spectrum has closely-spaced sharp spectral peaks
- See Figures 12.6 & 12.7 in P&M Text
- See demo - matlab file at course web site ⇒ YWvsULS.m

- Refinement to Unconstrained LS
- motivation: $\underline{\hat{R}}^{LS} \xrightarrow{N \to \infty} \underline{R}$
- where: $\underline{R}$ is Toeplitz-Hermitian
- $\underline{R}$ satisfies: $\underline{\tilde{I}}\,\underline{R}^*\underline{\tilde{I}} = \underline{R}$
- $\underline{\tilde{I}}$: reverse permutation matrix

$$\underline{\tilde{I}} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 0 & & & \cdot^{\cdot^{\cdot}} & 0 \\ \vdots & & & & \vdots \\ 0 & 1 & & & \\ 1 & & & & 0 \end{bmatrix}$$

one's along anti-diagonal & zeros elsewhere

- action of $\underset{\sim}{\tilde{I}}$ on a vector:

$$\underset{\sim}{\tilde{I}} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \underset{\sim}{\tilde{I}} = \begin{bmatrix} 3 & 2 & 1 \end{bmatrix}$$

- consider 3x3 example Toeplitz matrix

$$\underset{\sim}{R} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 1 & 2 \\ 3 & 2 & 1 \end{bmatrix}; \quad \underset{\sim}{\tilde{I}} \underset{\sim}{R} = \begin{bmatrix} 3 & 2 & 1 \\ 2 & 1 & 2 \\ 1 & 2 & 3 \end{bmatrix}$$

$$(\tilde{\underline{I}}\,\underline{R})\,\tilde{\underline{I}} = \begin{bmatrix}1 & 2 & 3\\ 2 & 1 & 2\\ 3 & 2 & 1\end{bmatrix} = \underline{R} \quad \checkmark$$

- viewer can verify w complex example

$$\tilde{\underline{\underline{I}}}\,\underline{R}^{*}\,\tilde{\underline{\underline{I}}} = \underline{R}$$

- conjugate centro-Hermitian property
- since $\hat{\underline{R}}^{LS}$ is aysmptotically Toeplitz,

replace $\hat{\underline{R}}^{LS}$ by

$$\hat{\underline{R}}^{fb} = \frac{1}{2}\left\{\hat{\underline{R}}^{LS} + \tilde{\underline{\underline{I}}}\,\hat{R}^{LS*}\,\tilde{\underline{\underline{I}}}\right\}$$

- $\underline{\hat{R}}^{fb} = \underline{\tilde{I}}\,\underline{\hat{R}}^{fb*}\,\underline{\tilde{I}}$
- to prove this, need to use $\underline{\tilde{I}}\,\underline{\tilde{I}} = I$
- show by example:

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

- $\underline{\tilde{I}}\,\underline{\hat{R}}^{fb*}\,\underline{\tilde{I}} = \underline{\tilde{I}}\left\{\frac{1}{2}\underline{\hat{R}}^{LS*} + \frac{1}{2}\underline{\tilde{I}}\,\underline{\hat{R}}^{LS}\,\underline{\tilde{I}}\right\}\underline{\tilde{I}}$

$$= \frac{1}{2}\left\{\underline{\tilde{I}}\,\underline{\hat{R}}^{LS*}\,\underline{\tilde{I}} + \underline{\hat{R}}^{LS}\right\} = \underline{\hat{R}}^{fb}$$

- viewer can see Sect. 11.2.2 for relationship to backward prediction
- note: no computation involved in forming $\underline{\tilde{I}}\,\hat{R}^{LS*}\,\underline{\tilde{I}}$
  - just simple reordering of matrix elements plus conjugation
- thus, formula (12.3.7) in P&M text is deceptive
-

- again: Unconstrained LS generally outperforms Yule-Walker
  - offers better resolution when power spectrum has closely-spaced sharp spectral peaks
- computational trade-off: can't use Levinson-Durbin algorithm to solve

$$\underline{\hat{R}}^{fb}\,\underline{\hat{a}} = \underline{\hat{r}}^{fb}$$

- However, if we do backward avg. as well as forward-avg. (very little added comp.)

- exploiting this "centro-Hermitian" property, Larry Marple developed a method for solving $\underline{\hat{R}}^{fb}\,\underline{\hat{a}} = \underline{\bar{\hat{r}}}^{fb}$ that is similar in complexity to the Levinson-Durbin algorithm
- Reference: "Digital Spectral Analysis" by Larry Marple, Prentice-Hall, ~1991.
- backward avg. offers almost negligible performance improvement ⇒ mostly useful for computational reduction

- ARMA Spectral Estimation
- more generally applicable than AR
- power spectrum modeled as rationale:

$$S_{xx}(\omega) = \sigma_w^2 \frac{\left|1+\sum_{k=1}^{q} b_k e^{-jk\omega}\right|^2}{\left|1+\sum_{k=1}^{p} a_k e^{-jk\omega}\right|^2}$$

- again, for power spectrum estimation, only trying to "match" the power spectrum of a stationary random process

$$v[n] \longrightarrow \boxed{H(z) = \frac{\sum_{k=0}^{q} b_k z^{-k}}{1 + \sum_{k=1}^{p} a_k z^{-k}}} \longrightarrow x[n]$$

i.i.d.
for all n

- $E\{v[n]v^*[n-m]\} = \sigma_w^2 \delta[m]$ * *

- in contrast to AR model, there is a nonlinear relationship between $r_{xx}[m] = E\{x[n]x^*[n-m]\}$ and the MA model parameters $\{b_k, k=1,\ldots,p\}$

See Eqn. 12.3.4

- Yet, the AR parameters $\{a_k, k=1,\dots,p\}$ may be solved for via a linear system of equations using $r_{xx}[0], \dots, r_{xx}[p], \dots, r_{xx}[q+p]$
- let's examine $r_{xx}[m]$ for $m>q$

$$E\{x[n]\,x^*[n-m]\} =$$
$$E\left\{\left[-\sum_{k=1}^{p} a_k x[n-k] + \sum_{k=0}^{q} b_k v[n-k]\right] x^*[n-m]\right\}$$

- $r_{xx}[m] = -\sum_{k=1}^{p} a_k r_{xx}[m-k] +$
$\sum_{k=0}^{q} b_k E\left\{ v[n-k]\, x^*[n-m] \right\}$

$v[n], \ldots, v[n-q]$
contribute

$v[n-m]$,
$v[n-m-1]$,
$\vdots$
contribute

- for $m > q$, second term is zero. Thus:

$$r_{xx}[m] = -\sum_{k=1}^{p} a_k r_{xx}[m-k]$$

"pee": upper limit

$$\underbrace{\begin{bmatrix} r_{xx}[q] & r_{xx}[q-1] & \cdots & r_{xx}[q-p] \\ r_{xx}[q+1] & r_{xx}[q] & \cdots & r_{xx}[q-p+1] \\ \vdots & \vdots & \cdots & \vdots \\ r_{xx}[q+p-1] & & \cdots & r_{xx}[q] \end{bmatrix}}_{\text{Toeplitz} \Rightarrow \text{ but } \underline{\text{NOT}} \text{ symmetric}} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = - \begin{bmatrix} r_{xx}[q+1] \\ r_{xx}[q+2] \\ \vdots \\ r_{xx}[q+p] \end{bmatrix}$$

- Text states create overdetermined system of equations – compute LS error sol'n. – see p. 935 (12.3.46)

$$\nu[n] \longrightarrow \boxed{\dfrac{\sum_{k=0}^{q} b_k z^{-k}}{1+\sum_{k=1}^{p} a_k z^{-k}}} \xrightarrow{x[n]} \boxed{1+\sum_{k=1}^{p} \hat{a}_k z^{-k}} \xrightarrow{v[n]}$$

- if $\hat{a}_k = a_k$, then

$$\left.\begin{aligned} v[n] &= \sum_{k=0}^{q} b_k \, \nu[n-k] \\ \end{aligned}\right\} \text{MA}(q) \text{ Process}$$

- $$r_{vv}[m] = \sigma_w^2 \sum_{k=0}^{q} b_k \, b_{k-m}^*$$

- $v[n] = \sum_{k=0}^{q} b_k \, \nu[n-k]$

- $r_{vv}[m] = E\{v[n]\, v^*[n-m]\}$

$$= \sum_{k=0}^{q} \sum_{l=0}^{q} b_k \, b_l^* \underbrace{E\{\nu[n-k]\,\nu^*[n-m-l]\}}_{\sigma_w^2 \, \delta[k-m-l]}$$

$k = m+l \Rightarrow l = k-m$

$$= \sum_{k=0}^{q} b_k \, b_{k-m}^* \Rightarrow$$ deterministic autocorrelation of $\{b_0, b_1, \ldots, b_q\}$

- thus, $r_{vv}[m] \neq 0$ only
for $|m| \leq q$

- $$\hat{S}_{xx}^{ARMA}(\omega) = \frac{\sum_{m=-q}^{q} \hat{r}_{vv}[m] e^{-j\omega m}}{\left|1 + \sum_{k=1}^{p} \hat{a}_k e^{-jk\omega}\right|^2}$$

- See demo : ARMA2stepest.m