Parametric Spectral Estimation

ARMA Spectral Estimation

Spectral density is modelled as

\[ S_{\text{ARMA}}(\omega) = \frac{\sigma^2}{|\sum_{k=0}^{N-1} b_k e^{-jkw}|^2} \]

\[ \frac{1}{|\sum_{k=0}^{M-1} a_k e^{-jkw}|^2} \]

Notation from statistics literature on ARMA processes, \( M = 8, N = P \)

Indirect spectral estimation: compute \( b_k, k = 0, \ldots, 8 \) and \( a_k, k = 1, \ldots, P \) from the estimated autocorrelation values.
only need (theoretically) autocorrelations for lag values \( m = 0, 1, \ldots, p + q \) to determine \( b_k, k = 0, 1, \ldots, q \) and \( a_k, k = 1, \ldots, p \)

(note: \( r_{xx}[-m] = r_{xx}^* [m] \))

ARMA spectral estimation works well

BUT the ARMA model parameters are nonlinearly related to the autocorrelation values

issue of computational complexity
\[ S_{AR}(w) = \frac{\sigma^2}{1 + \sum_{k=0}^{p} a_k e^{-jkw}} \]

- AR model parameters are computed from the estimated/measured autocorrelation values via a linear system of equations of the form

\[ \begin{bmatrix} R_{xx} \end{bmatrix}_p \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}_p = -r \]

where \( R_{xx} \) is a \( p \times p \) matrix and \( a \) is a \( p \times 1 \) vector.
\( \mathbf{R}_{xx} \) is a Toeplitz-Hermitian matrix which is formed from auto correlations for lag values 0, \( \frac{1}{2}, \ldots, p \):

\[
\begin{bmatrix}
    r_{xx}[0] & r_{xx}^*[1] & r_{xx}^*[2] & \cdots & r_{xx}^*[p-1] \\
    r_{xx}[1] & r_{xx}[0] & r_{xx}^*[1] & \cdots & r_{xx}^*[p-2] \\
    r_{xx}[2] & r_{xx}[1] & r_{xx}[0] & \cdots & r_{xx}^*[p-3] \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    r_{xx}[p-1] & r_{xx}[p-2] & \cdots & \cdots & r_{xx}[0]
\end{bmatrix}
\]

- **Toeplitz**: constant along any diagonal
- **Hermitian**: conjugate-symmetric about main diagonal
Due to Toeplitz structure of $R_{xx}$, $a = -r$ can be solved with low complexity. In fact, there is no need to even form the matrix $R_{xx}$.

Thus, AR spectral estimation is used more in practice than ARMA spectral estimation, even though ARMA generally performs better than AR. Can increase the value of $p$ to make AR perform almost as well as ARMA.
So, the steps are:

1. Estimate $\hat{r}_{xx}[m]$ for lag values $m=0, 1, \ldots, P$

   $$\hat{r}_{xx}[m] = \frac{1}{N-m-1} \sum_{n=0}^{N-m-1} x_n^* x[n+m]$$

   or $N \leftarrow n=0, m=0, 1, \ldots, P$

2. Solve $R_{xx} \hat{a} = -\mathbf{r}$ for the AR model parameters via the Levinson-Durbin algorithm

   $\hat{a}_k, k=1, \ldots, P$

3. Plug the AR model parameters into the AR spectral estimate:

   $$S_{xx}(\omega) = \frac{\mathbf{a}^2}{|1 + \sum_{k=1}^{P} a_k e^{-j2\pi k \omega}|^2}$$

   (talk about $\sigma^2$ later)
Relationship between AR model parameters $a_k, k=1, \ldots, p$ and autocorrelation values $r_{xx}[m], m=0, 1, \ldots, p$

Observation: consider passing white noise $\nu[n]$ through an all-pole filter

\[ H(z) = \frac{1}{1 + \sum_{k=1}^{p} a_k z^{-k}} \rightarrow x[n] \]

\[ r_{xx}[m] = \sigma_w^2 \delta[m] \]

\[ x[n] = -\sum_{k=1}^{p} a_k x[n-k] + \nu[n] \quad \{ \text{input: } \nu[n] \} \]

\[ H(\omega) = \lvert H(z) \rvert \bigg|_{z=e^{j\omega}} = \frac{1}{1 + \sum_{k=1}^{p} a_k e^{-j\omega k}} \]

\[ \text{output: } x[n] \]
Input/Output Relationship for passing DT random process thru LTI filter

\[ x[n] \rightarrow h[n] \rightarrow y[n] \]

\[ r_{xx}[m] \xrightarrow{DTFT} S_x^x(\omega) \]
\[ r_{yy}[m] \xrightarrow{DTFT} S_y^y(\omega) \]

- Easy to prove: \[ S_y^y(\omega) = |H(\omega)|^2 S_x^x(\omega) \]

- These relationships hold whether dealing with deterministic or stochastic auto correlation

\[ r_{yx}[m] = h[m] * r_{xx}[m] \]
\[ r_{yy}[m] = r_{xx}[m] * r_{hh}[m] \]
Again, for AR process, input is $y(n)$ and output is $x(n)$.

Thus: \[ S'_{xx}(\omega) = |H(\omega)|^2 S'_{yy}(\omega) \]

\[ S'_{yy}(\omega) = \text{DTFT}\{r_{yy}[m]\} = \text{DTFT}\{\sigma_w^2 s[m]\} = \sigma_w^2 \]

\[ H(\omega) = \frac{1}{1 + \sum_{k=1}^{p} a_k e^{-j k \omega}} \]

Hence:

\[ S'_{xx}(\omega) = \frac{\sigma_w^2}{1 + \sum_{k=1}^{p} a_k e^{-j k \omega}}^2 = S'_{AR}(\omega) \]

running white noise thru an all-pole filter generates this spectral density.
\[ x(n) = - \sum_{k=1}^{p} a_k x(n-k) + u(n) \]

- Multiply both sides by \( x^*(n-m) \) and take expected value

\[
E\{ x(n) x^*(n-m) \} = - \sum_{k=1}^{p} a_k E\{ x(n-k) x^*(n-m) \} \\
+ E\{ u(n) x^*(n-m) \}
\]

\[ r_{xx}(m) = - \sum_{k=1}^{p} a_k r_{xx}(m-k) + E\{ u(n) x^*(n-m) \} \]

(for WSS \( x(n) \): \( E\{ x(n-k) x^*(n-m) \} = r_{xx}(m-k) \))

- Consider \( m \) to be strictly positive, since \( r_{xx}(-m) = r_{xx}(m) \)
\[ x[n-m] = -\sum_{k=1}^{p} a_k x[n-m-k] + \nu[n-m] \]

for \( m > 0 \): \( x[n-m] \) depends only on past values of \( \nu[n] \)

Since \( r_{xx}[m] = E\{x[n]x^*[n-m]\} = \sigma_w^2 \delta[m] \)

it follows that for \( m > 0 \):

\[ r_{xx}[m] = -\sum_{k=1}^{p} a_k r_{xx}[m-k] \quad m > 0 \]

Thus, we use \( r_{xx}[m], \quad m = 0, 1, \ldots, p \) to set up a system of \( p \) equations in \( p \) unknowns
\[
\begin{bmatrix}
\begin{array}{c}
\ldots \\
r_{xx}[1] \\
r_{xx}[2] \\
r_{xx}[p] \\
\end{array}
\end{bmatrix}
\begin{bmatrix}
\begin{array}{ccc}
r_{xx}[0] & r_{xx}[\ldots] & r_{xx}[1-p] \\
r_{xx}[1] & r_{xx}[0] & \ldots & r_{xx}[2-p] \\
\vdots & \vdots & \ddots & \vdots \\
r_{xx}[p-1] & r_{xx}[p-2] & \ldots & r_{xx}[0] \\
\end{array}
\end{bmatrix}
\begin{bmatrix}
a_1 \\
a_2 \\
\vdots \\
a_p \\
\end{bmatrix}
\]

\[r = -\mathbf{R}_{xx} \mathbf{a} \quad \text{or} \quad \mathbf{R}_{xx} \mathbf{a} = -\mathbf{r}\]

Using \(r_{xx}[-m] = r_{xx}^*[m]\), we observe Hermitian (conjugate) symmetry about the main diagonal, as well as Toeplitz structure, discussed previously.
For \( m = 0 \) equation, we have

\[
\begin{align*}
    r_{xx}[0] &= -\sum_{k=1}^{p} a_k r_{xx}[0-k] + E\left\{\nu[n] x^*[n]\right\} \\
    x[n] &= -\sum_{k=1}^{p} x[n-k] + \nu[n] \\
    \text{only contain} \\
    \nu[0-1], \ldots, \nu[n-p] \\
    \text{past values of } \nu[n]
\end{align*}
\]

Thus:

\[
\begin{align*}
    r_{xx}[0] &= -\sum_{k=1}^{p} a_k r_{xx}[0-k] + E\left\{\nu[n] x^*[n]\right\} \\
    r_{xx}[0] &= -\sum_{k=1}^{p} a_k r_{xx}^*[k] + \sigma_w^2 \\
    \Rightarrow \sigma_w^2 &= r_{xx}[0] + \sum_{k=1}^{p} a_k r_{xx}^*[k]
\end{align*}
\]