ECE 251A: Digital Signal Processing I
Random Process Modeling and PSD Estimation

Florian Meyer

Electrical and Computer Engineering Department
University of California San Diego

UC San Diego
JACOBS SCHOOL OF ENGINEERING

Modeling and PSD Estimation

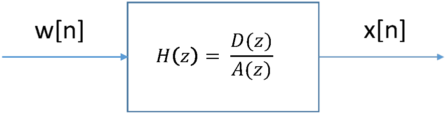

Figure: $x[n]$ is output of a LTI system driven by white noise ($E(w[n]) = 0$ and $E(w[n+m]w^*[n]) = \sigma_w^2 \delta[m]$.)

Benefits: $R_{xx}(e^{j\omega}) = \sigma_w^2 \, |H(e^{j\omega})|^2$ and problem reduces to finding parameters of $H(z)$, which hopefully is characterized by a few parameters.

Estimating few parameters from N samples is a more manageable problem.

Modeling Options

1. $H(z) = D(z)$. Process is referred to as Moving Average (MA) process. Will have limited value beyond the Fourier techniques for PSD estimation
2. $H(z) = \frac{1}{A(z)}$. Process is referred to as AutoRegressive (AR) process. Will be useful beyond the Fourier techniques for PSD estimation.
3. $H(z) = \frac{D(z)}{A(z)}$. Process is referred to as AutoRegressive Moving Average(ARMA) process. Will explore briefly.

Moving Average (MA) Modeling ($H(z) = D(z)$)

$R(z) = \sigma_w^2 D(z) D^*(\frac{1}{z^*})$ or $R(e^{j\omega}) = \sigma_w^2 |D(e^{j\omega})|^2$, where

$$D(z) = 1 + d_1 z^{-1} + \ldots + d_Q z^{-Q} = 1 + \sum_{l=1}^{Q} d_l z^{-l}$$

In the time domain

$$x[n] = w[n] + \sum_{l=1}^{Q} d_l w[n-l].$$

The output is a weighted average of the current and past Q input samples and hence the terminology moving average.

Based on the model, the autocorrelation sequence is of finite duration, i.e. $r_{xx}[m]$ is non-zero only for lag values $-Q \leqslant m \leqslant Q$.

Example of a MA process

$x[n] = w[n] + .9w[n-1] + \ldots + .1w[n-9]$, where $w[n]$ is a zero mean, variance one, i.i.d. Gaussian sequence. $H(z) = 1 + .9z^{-1} + \ldots + .1z^{-9}$.

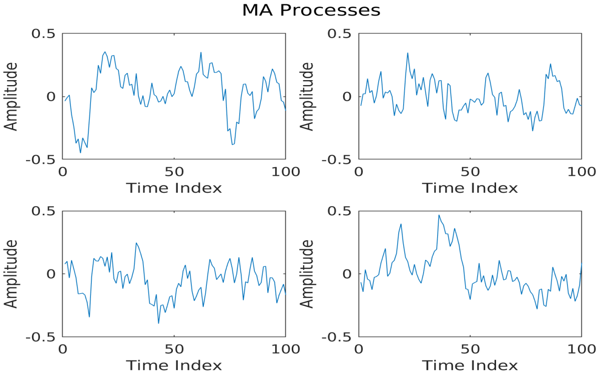

Figure: Four realizations of the MA Process

MA Modeling and PSD Estimation

Estimating $D(z)$ from $r[m]$ which is obtained from data.

1. Need model order Q. which can be estimated or assumed known. We will assume known for simplicity.
2. Compute $R(z) = \sum_{m=-Q}^{Q} r[m] z^{-m}$
3. Factor $R(z)$ to obtain $\breve{D}(z) = \breve{d}_0 + \breve{d}_1 z^{-1} + \ldots + \breve{d}_Q z^{-Q}$. Can choose the minimum phase solution if there are no other constraints. Scale the factor $\breve{D}(z)$ to set $d_0 = 1$ and obtain $D(z)$ as $\breve{D}(z) = \breve{d}_0 D(z)$. This will give us $\sigma_w^2 = \breve{d}_0^2$.

Once $D(z)$ has been found, PSD estimate is $\sigma_w^2 D(z) D^*(\frac{1}{z^*})$. If PSD is the goal, can be computed directly using the FFT of the autocorrelation sequence in step 2. MA models not very useful for PSD estimation but ok if modeling is the objective

AutoRegressive (AR) or All-Pole Modeling ($H(z) = \frac{1}{A(z)}$)

$R(z) = \frac{\sigma_w^2}{A(z)A^*(\frac{1}{z^*})}$ or $R(e^{j\omega}) = \frac{\sigma_w^2}{|A(e^{j\omega})|^2}$, where

$$A(z) = 1 + a_1^* z^{-1} + \ldots + a_P^* z^{-P} = 1 + \sum_{l=1}^{P} a_l^* z^{-l}$$

In the time domain

$$x[n] = -a_1^* x[n-1] - \ldots - a_P^* x[n-P] + w[n] = -\sum_{l=1}^{P} a_l^* x[n-l] + w[n]$$

$x[n]$ depends on past values of $x[n]$ in a linear manner leading to the Autoregressive terminology.

Book uses the notation $A(z) = 1 + a_1 z^{-1} + \ldots + a_P z^{-P} = 1 + \sum_{l=1}^{P} a_l z^{-l}$, (No complex conjugate on the coefficients a_l)but that complicates the equations and connections to optimal filtering in the later chapter.

An Example

$x[n] = ax[n-1] + w[n]$, where $w[n]$ is a zero mean, variance one, i.i.d. Gaussian sequence. $H(z) = \frac{1}{1-az^{-1}}$.

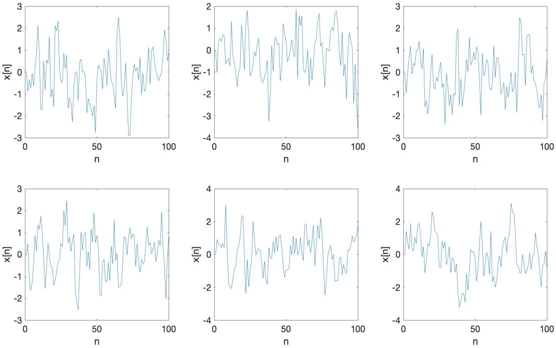

Figure: Six realizations for $a = 0.5$

Another Example with $a = -0.9$

$x[n] = ax[n-1] + w[n]$, where $w[n]$ is a zero mean, variance one, i.i.d. Gaussian sequence. $H(z) = \frac{1}{1-az^{-1}}$.

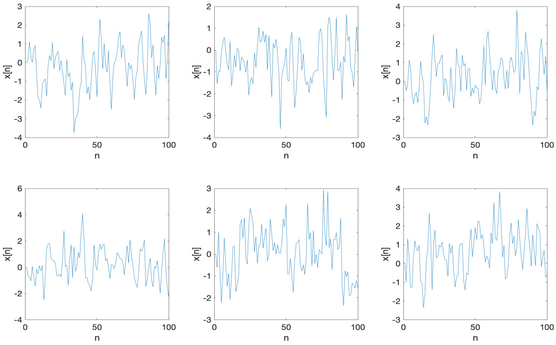

Figure: Six realizations for $a = -0.9$

AR Model Features

Based on the model, the autocorrelation sequence is of infinite duration.

$$H(z) = \frac{1}{A(z)} = \sum_{l=0}^{\infty} h[l]z^{-l} \text{ and } r[m] = \sigma_w^2(h[m] * h^*[-m])$$

AR modeling provides a way to extend the auto-correlation sequence!

Useful fact: $x[n] = -\sum_{l=1}^{P} a_l^* x[n-l] + w[n] = \sum_{k=0}^{\infty} h[k]w[n-k]$. Since $w[n]$ is a zero mean white noise sequence, i.e. $E(w[n]w^*[n-m]) = \sigma_w^2 \delta[m]$, we have

$$E\big(x[n]w^*[n+l]\big) = 0, l > 0.$$

Interpretation: Output $x[n]$ uncorrelated with future input $w[n+l], l > 0$.

Another useful fact is

$$E(x[n]w^*[n]) = E\left(\left(-\sum_{l=1}^{P} a_l^* x[n-l] + w[n]\right) w^*[n]\right) = E(w[n]w^*[n]) = \sigma_w^2.$$

AR Model Parameter Estimation

AR model parameters are $\mathbf{a}_P$ and σ_w^2 where $\mathbf{a}_P = [a_1, a_2, .., a_P]^T$.

$x[n] = -\sum_{l=1}^{P} a_l^* x[n-l] + w[n]$ can be rearranged as
$x[n] + \sum_{l=1}^{P} a_l^* x[n-l] = w[n]$.

In vector form, with $\mathbf{x}_{P+1}[n] = [x[n], x[n-1], .., x[n-P]]^T$, we have

$$\left[1, \mathbf{a}_P^H\right] \mathbf{x}_{P+1}[n] = w[n] \;\text{ or }\; \mathbf{x}_{P+1}^H[n] \begin{bmatrix} 1 \\ \mathbf{a}_P \end{bmatrix} = w^*[n]$$

Multipling both sides by $\mathbf{x}_{P+1}[n]$ and taking expectations, we have

$$E\left(\mathbf{x}_{P+1}[n]\mathbf{x}_{P+1}^H[n] \begin{bmatrix} 1 \\ \mathbf{a}_P \end{bmatrix}\right) = E(\mathbf{x}_{P+1}[n] w^*[n]) \text{ or } \mathbf{R}_{P+1} \begin{bmatrix} 1 \\ \mathbf{a}_P \end{bmatrix} = \begin{bmatrix} \sigma_w^2 \\ \mathbf{0}_{P\times 1} \end{bmatrix}$$

$\mathbf{R}_{P+1}$ is the $(P+1)\times(P+1)$ Toeplitz autocorrelation matrix. The above system of equations are referred to as the Augmented Yule-Walker Equations.The last P equations can be used to solve for the AR model parameters $\mathbf{a}_P$ and the first equation for σ_w^2.

Toeplitz Autocorrelation Matrix $\mathbf{R}_{P+1}$

$$
\begin{aligned}
\mathbf{R}_{P+1} &= E(\mathbf{x}_{P+1}[n]\mathbf{x}_{P+1}^H[n]) = E\left(\begin{bmatrix} x[n] \\ x[n-1] \\ \vdots \\ x[n-P] \end{bmatrix} [x^*[n], x^*[n-1], \dots, x^*[n-P]]\right) \\
&= \begin{bmatrix} r[0] & r[1] & \dots & r[P] \\ r[-1] & r[0] & \dots & r[P-1] \\ \vdots & \vdots & \vdots & \vdots \\ r[-P] & r(-(P-1) & \dots & r[0] \end{bmatrix}
\end{aligned}
$$

1. $\mathbf{R}_{P+1} = \mathbf{R}_{P+1}^H$ (Hermitian Symmetry)
2. $\mathbf{R}_{P+1}$ is positive semidefinite, $\mathbf{a}^H\mathbf{R}_{P+1}\mathbf{a} \geqslant 0, \forall \mathbf{a}$. (usually positive definite)
3. $\mathbf{R}_{P+1}$ is Toeplitz, and defined by the first row. The Toeplitz structure allows for low complexity algorithms. Inversion which is usually $O(P^3)$ complexity can be reduced to $O(P^2)$ complexity.
4. Eigendecomposition: $\mathbf{R}_{P+1} = Q\Lambda Q^H$. Q is a $(P+1)\times(P+1)$ matrix containing the orthonormal eigenvectors, i.e. $Q = [q_1, q_2, \dots, q_{P+1}]$ and $QQ^H = Q^HQ = \mathbf{I}_M$. Λ is a diagonal matrix containing the non-negative eigenvalues, i.e. $\Lambda = \text{diag}(\lambda_1, \lambda_2, .., \lambda_{P+1})$ with $\lambda_i \geqslant 0$.

Centro-Hermitian Property of $\mathbf{R}_{P+1} = J\mathbf{R}^*_{P+1}J$

Define $J = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 1 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \end{bmatrix}$ and $\mathbf{y}[n] = J\mathbf{x}^*_{P+1}[n]$. Then

$E(\mathbf{y}[n]\mathbf{y}^H[n]) = JE(\mathbf{x}^*_{P+1}[n]\mathbf{x}^T_{P+1}[n])J = J(E(\mathbf{x}_{P+1}[n]\mathbf{x}^H_{P+1}[n]))^* J = J\mathbf{R}^*_{P+1}J.$

$$\mathbf{y}[n] = J\mathbf{x}^*_{P+1}[n] = J \begin{bmatrix} x^*[n] \\ x^*[n-1] \\ \vdots \\ x^*[n-P] \end{bmatrix} = \begin{bmatrix} x^*[n-P] \\ x^*[n-P+1] \\ \vdots \\ x^*[n] \end{bmatrix}.$$

$$\begin{aligned} E\left(\mathbf{y}[n]\mathbf{y}^H[n]\right) &= E\left(\begin{bmatrix} x^*[n-P] \\ x^*[n-P+1] \\ \vdots \\ x^*[n] \end{bmatrix} [x[n-P], x[n-P+1], \dots, x[n]]\right) \\ &= \begin{bmatrix} r[0] & r[1] & \dots & r[P] \\ r[-1] & r[0] & \dots & r[P-1] \\ \vdots & \vdots & \vdots & \vdots \\ r[-P] & r(-(P-1) & \dots & r[0] \end{bmatrix} = \mathbf{R}_{P+1} \end{aligned}$$

Hence $\mathbf{R}_{P+1} = J\mathbf{R}^*_{P+1}J$,

Nested Toeplitz Structure of $\mathbf{R}_{P+1}$

$$
\begin{aligned}
\mathbf{R}_{P+1} &= \left[\begin{array}{c|cccc}
r[0] & r[1] & \dots & r[P] \\
\hline
r[-1] & r[0] & \dots & r[P-1] \\
\vdots & \vdots & \vdots & \vdots \\
r[-P] & r(-(P-1) & \dots & r[0]
\end{array}\right] \\
&= \left[\begin{array}{ccc|c}
r[0] & \dots & r[P-1] & r[P] \\
\vdots & \vdots & \vdots & \vdots \\
r[-P+1] & \dots & r[0] & r[1] \\
\hline
r[-P] & \dots & r[-1] & r[0]
\end{array}\right]
\end{aligned}
$$

Define $\mathbf{r} = [r[1], r[2], \dots, r[P]]^T$, then

$$
\mathbf{R}_{P+1} = \begin{bmatrix} r[0] & \mathbf{r}^T \\ \mathbf{r}^* & \mathbf{R}_P \end{bmatrix} = \begin{bmatrix} \mathbf{R}_P & J\mathbf{r} \\ (J\mathbf{r})^H & r[0] \end{bmatrix}
$$

AR Model Parameter Estimation Complexity

The AR model estimation equation (Extended Yule-Walker Equations)

$$\mathbf{R}_{P+1} \begin{bmatrix} 1 \\ \mathbf{a}_P \end{bmatrix} = \begin{bmatrix} \sigma_w^2 \\ \mathbf{0}_P \end{bmatrix}$$

Using the partition

$$\mathbf{R}_{P+1} = \begin{bmatrix} r[0] & \mathbf{r}^T \\ \mathbf{r}^* & \mathbf{R}_P \end{bmatrix}$$

The last P equations lead to $\mathbf{R}_P \mathbf{a}_P = -\mathbf{r}^*$ (Yule-Walker (YW) Equations), which can be used to solve for $\mathbf{a}_P$. The first equation leads to $r[0] + \mathbf{r}^T \mathbf{a}_P = \sigma_w^2$ which can be used to solve for σ_w^2 since $\mathbf{a}_P$ has already been obtained from the YW equations.

The AR model parameter estimation requires autocorrelation lags $r[0], r[1], .., r[P]$ (only $(P+1)$ lags).

AR Modeling Issues

1. Computational complexity (Matrix Inversion). Will develop a fast algorithm that exploits the Toeplitz structure.
2. What if the process is not AR? Will exploit the connection with linear prediction (LP) to provide some robustness arguments. This will lead to our aside on linear mean squared estimation in the next lecture.
3. Model order of the AR process? Important but not critical based on the LP connection.

The Linear Prediction framework will supersede the AR modeling formulation and make the algorithm development more meaningful and broadly applicable.

ARMA Modeling ($H(z) = \frac{D(z)}{A(z)}$)

$$x[n] = -\sum_{l=1}^{P} a_l^* x[n-l] + \sum_{k=0}^{Q} d_l w[n-k]$$

PSD is given by

$$R(z) = \sigma_w^2 \frac{D(z)D(\frac{1}{z^*})}{A(z)A(\frac{1}{z^*})} \text{ and } \sigma_w^2 D(z) D\left(\frac{1}{z^*}\right) = R(z)A(z)A\left(\frac{1}{z^*}\right)$$

If the AR parameters are found, then the data can be filtered by $A(z)$ to determine $R(z)A(z)A(\frac{1}{z^*})$ and $D(z)$ obtained by spectral-factorization.

Will assume model order P and Q are known, and will concentrate on finding $A(z)$ or $\mathbf{a} = [a_1, a_2, .., a_P]^T$.

Finding $A(z)$ or $\mathbf{a}$

Reminder: $x[n] = \sum_{m=0}^{\infty} h[m]w[n-m]$ and so $E(x[n]w^*[n+l]) = 0, l > 0$.
Reasoning: $x[n]$ depends only on $w[k], k \leqslant n$.

Alternate version useful for our derivation is
$E(x[n-m]w^*[n-k]) = 0, m \geqslant 0, k \geqslant 0$, and $m > k$.

Considering $m > Q$,

$$\begin{aligned}
r[m] &= E(x[n-m]^*x[n]) = E(x^*[n-m](-\sum_{l=1}^{P} a_l^* x[n-l] + \sum_{k=0}^{Q} d_k w[n-k])) \\
&= -\sum_{l=1}^{P} a_l^* E(x^*[n-m]x[n-l]) + \sum_{k=0}^{Q} d_l E(x^*[n-m]w[n-k])) \\
&= -\sum_{l=1}^{P} a_l^* r[m-l] + 0
\end{aligned}$$

If we use $m = Q+1, Q+2, .., Q+P$, we will have P equations which we can solve for $\mathbf{a}$.