ELEG-636: Statistical Signal Processing

Gonzalo R. Arce

Department of Electrical and Computer Engineering
University of Delaware

Spring 2010
Course Objectives & Structure

Objective: Given a discrete time sequence \( \{x(n)\} \), develop

- Statistical and spectral signal representation
- Filtering, prediction, and system identification algorithms
- Optimization methods that are
  - Statistical
  - Adaptive

Course Structure:

- Weekly lectures [notes: www.ece.udel.edu/~arce]
- Periodic homework (theory & Matlab implementations) [15%]
- Midterm & Final examinations [85%]

Textbook:

- Haykin, Adaptive Filter Theory.
Course Objectives & Structure

- Broad Applications in Communications, Imaging, Sensors.
- Emerging application in
  - Brain-imaging techniques
  - Brain-machine interfaces,
  - Implantable devices.
- Neurofeedback presents real-time physiological signals from MRIs in a visual or auditory form to provide information about brain activity. These signals are used to train the patient to alter neural activity in a desired direction.
- Traditionally, feedback using EEGs or other mechanisms has not focused on the brain because the resolution is not good enough.
Problem Statement
Produce an estimate of a desired process statistically related to a set of observations

Historical Notes: The linear filtering problem was solved by
- Andrey Kolmogorov for discrete time – his 1938 paper "established the basic theorems for smoothing and predicting stationary stochastic processes"
- Norbert Wiener in 1941 for continuous time – not published until the 1949 paper *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*
System restrictions and considerations:

- Filter is linear
- Filter is discrete time
- Filter is finite impulse response (FIR)
- The process is WSS
- Statistical optimization is employed
For the discrete time case

\[ x(n) \rightarrow Z^{-1} \rightarrow Z^{-1} \rightarrow \cdots \rightarrow Z^{-1} \rightarrow \]

\[ w_0^* \quad w_1^* \quad \cdots \quad w_{M-1}^* \]

\[ \vdots \]

\[ \hat{d}(n) \quad + \quad d(n) \quad e(n) \]

- The filter impulse response is finite and given by
  \[ h_k = \begin{cases} 
  w_k^* & \text{for } k = 0, 1, \ldots, M - 1 \\
  0 & \text{otherwise}
  \end{cases} \]

- The output \( \hat{d}(n) \) is an estimate of the desired signal \( d(n) \)
  - \( x(n) \) and \( d(n) \) are statistically related \( \Rightarrow \) \( \hat{d}(n) \) and \( d(n) \) are statistically related
In convolution and vector form

\[
\hat{d}(n) = \sum_{k=0}^{M-1} w_k^* x(n - k) = \mathbf{w}^H \mathbf{x}(n)
\]

where

\[
\mathbf{w} = [w_0, w_1, \cdots, w_{M-1}]^T \quad \text{[filter coefficient vector]}
\]

\[
\mathbf{x} = [x(n), x(n - 1), \cdots, x(n - M + 1)]^T \quad \text{[observation vector]}
\]

The error can now be written as

\[
e(n) = d(n) - \hat{d}(n) = d(n) - \mathbf{w}^H \mathbf{x}(n)
\]

**Question:** Under what criteria should the error be minimized?

**Selected Criteria:** Mean squared-error (MSE)

\[
J(\mathbf{w}) = E\{e(n)e^*(n)\} \quad (*)
\]

**Result:** The \( \mathbf{w} \) that minimizes \( J(\mathbf{w}) \) is the optimal (Wiener) filter
Utilizing $e(n) = d(n) - w^H x(n)$ in (*) and expanding,

$$J(w) = E\{e(n)e^*(n)\}$$
$$= E\{(d(n) - w^H x(n))(d^*(n) - x^H(n)w)\}$$
$$= E\{|d(n)|^2 - d(n)x^H(n)w - w^H x(n)d^*(n)$$
$$+ w^H x(n)x^H(n)w\}$$
$$= E\{|d(n)|^2\} - E\{d(n)x^H(n)\}w - w^H E\{x(n)d^*(n)\}$$
$$+ w^H E\{x(n)x^H(n)\}w \quad (** \text{)}$$

Let

$$R = E\{x(n)x^H(n)\} \quad \text{[autocorrelation of } x(n)\text{]}$$
$$p = E\{x(n)d^*(n)\} \quad \text{[cross correlation between } x(n) \text{ and } d(n)\text{]}$$

Then (**) can be compactly expressed as

$$J(w) = \sigma_d^2 - p^H w - w^H p + w^H R w$$

where we have assumed $x(n)$ & $d(n)$ are zero mean, WSS.
The MSE criteria as a function of the filter weight vector $\mathbf{w}$

$$J(\mathbf{w}) = \sigma_d^2 - \mathbf{p}^H \mathbf{w} - \mathbf{w}^H \mathbf{p} + \mathbf{w}^H \mathbf{R} \mathbf{w}$$

**Observation:** The error is a quadratic function of $\mathbf{w}$

**Consequences:** The error is an $M$–dimensional bowl–shaped function of $\mathbf{w}$ with a unique minimum

**Result:** The optimal weight vector, $\mathbf{w}_0$, is determined by differentiating $J(\mathbf{w})$ and setting the result to zero

$$\nabla_{\mathbf{w}} J(\mathbf{w}) |_{\mathbf{w} = \mathbf{w}_0} = 0$$

- A closed form solution exists
Example

Consider a two dimensional case, i.e., a $M = 2$ tap filter. Plot the error surface and error contours.

**Figure 5.6** Error-performance surface of the two-tap transversal filter described in the numerical example.

**Figure 5.7** Contour plots of the error-performance surface depicted in Fig. 5.6.
Aside (Matrix Differentiation): For complex data,

\[ w_k = a_k + jb_k, \quad k = 0, 1, \ldots, M - 1 \]

the gradient, with respect to \( w_k \), is

\[
\nabla_k (J) = \frac{\partial J}{\partial a_k} + j \frac{\partial J}{\partial b_k}, \quad k = 0, 1, \ldots, M - 1
\]

The complete gradient is thus given by

\[
\nabla_w (J) = \begin{bmatrix}
\nabla_0 (J) \\
\nabla_1 (J) \\
\vdots \\
\nabla_{M-1} (J)
\end{bmatrix} = \begin{bmatrix}
\frac{\partial J}{\partial a_0} + j \frac{\partial J}{\partial b_0} \\
\frac{\partial J}{\partial a_1} + j \frac{\partial J}{\partial b_1} \\
\vdots \\
\frac{\partial J}{\partial a_{M-1}} + j \frac{\partial J}{\partial b_{M-1}}
\end{bmatrix}
\]
Example

Let \( c \) and \( w \) be \( M \times 1 \) complex vectors. For \( g = c^H w \), find \( \nabla_w(g) \)

Note

\[
g = c^H w = \sum_{k=0}^{M-1} c_k^* w_k = \sum_{k=0}^{M-1} c_k^* (a_k + jb_k)
\]

Thus

\[
\nabla_k(g) = \frac{\partial g}{\partial a_k} + j \frac{\partial g}{\partial b_k} = c_k^* + j(jc_k^*) = 0, \quad k = 0, 1, \ldots, M - 1
\]

Result: For \( g = c^H w \)

\[
\nabla_w(g) = \begin{bmatrix}
\nabla_0(g) \\
\nabla_1(g) \\
\vdots \\
\nabla_{M-1}(g)
\end{bmatrix} = \begin{bmatrix}
0 \\
0 \\
\vdots \\
0
\end{bmatrix} = 0
\]
Example

Now suppose $g = w^H c$. Find $\nabla_w(g)$

In this case,

$$g = w^H c = \sum_{k=0}^{M-1} w^*_k c_k = \sum_{k=0}^{M-1} c_k(a_k - jb_k)$$

and

$$\nabla_k(g) = \frac{\partial g}{\partial a_k} + j \frac{\partial g}{\partial b_k}$$

$$= c_k + j(-jc_k) = 2c_k, \quad k = 0, 1, \ldots, M-1$$

Result: For $g = w^H c$

$$\nabla_w(g) = \begin{bmatrix} \nabla_0(g) \\ \nabla_1(g) \\ \vdots \\ \nabla_{M-1}(g) \end{bmatrix} = \begin{bmatrix} 2c_0 \\ 2c_1 \\ \vdots \\ 2c_{M-1} \end{bmatrix} = 2c$$
Example

Lastly, suppose \( g = w^H Qw \). Find \( \nabla_w (g) \)

In this case,

\[
g = \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} w_i^* w_j q_{i,j}
\]

\[
= \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} (a_i - jb_i)(a_j + jb_j) q_{i,j}
\]

\[
\Rightarrow \nabla_k (g) = \frac{\partial g}{\partial a_k} + j \frac{\partial g}{\partial b_k}
\]

\[
= 2 \sum_{j=0}^{M-1} (a_j + jb_j)q_{k,j} + 0
\]

\[
= 2 \sum_{j=0}^{M-1} w_j q_{k,j}
\]
Result: For \( g = w^H Qw \)

\[
\nabla_w(g) = \begin{bmatrix}
\nabla_0(g) \\
\nabla_1(g) \\
\vdots \\
\nabla_{M-1}(g)
\end{bmatrix} = 2 \begin{bmatrix}
\sum_{i=0}^{M-1} q_{0,i} w_i \\
\sum_{i=0}^{M-1} q_{1,i} w_i \\
\vdots \\
\sum_{i=0}^{M-1} q_{M-1,i} w_i
\end{bmatrix} = 2Qw
\]

**Observation:** Differentiation result depends on matrix ordering
Returning to the MSE performance criteria

\[ J(w) = \sigma_d^2 - p^H w - w^H p + w^H R w \]

**Approach:** Minimize error by differentiating with respect to \( w \) and set result to 0

\[
\nabla_w (J) = 0 - 0 - 2p + 2Rw
\]

\[ = 0 \]

\[ \Rightarrow Rw_0 = p \quad [\text{normal equation}] \]

**Result:** The Wiener filter coefficients are defined by

\[ w_0 = R^{-1} p \]

**Question:** Does \( R^{-1} \) always exist? Recall \( R \) is positive semi-definite, and usually positive definite
Orthogonality Principle

Consider again the normal equation that defines the optimal solution

\[ \mathbf{Rw}_0 = \mathbf{p} \]
\[ \Rightarrow E\{\mathbf{x}(n)\mathbf{x}^H(n)\}\mathbf{w}_0 = E\{\mathbf{x}(n)d^*(n)\} \]

Rearranging

\[ E\{\mathbf{x}(n)d^*(n)\} - E\{\mathbf{x}(n)\mathbf{x}^H(n)\}\mathbf{w}_0 = 0 \]
\[ E\{\mathbf{x}(n)[d^*(n) - \mathbf{x}^H(n)\mathbf{w}_0]\} = 0 \]
\[ E\{\mathbf{x}(n)e_0^*(n)\} = 0 \]

Note: \( e_0^*(n) \) is the error when the optimal weights are used, i.e.,

\[ e_0^*(n) = d^*(n) - \mathbf{x}^H(n)\mathbf{w}_0 \]
Thus

\[ E\{x(n)e_0^*(n)\} = E \begin{bmatrix} x(n)e_0^*(n) \\ x(n-1)e_0^*(n) \\ \vdots \\ x(n-M+1)e_0^*(n) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \]

**Orthogonality Principle**

A necessary and sufficient condition for a filter to be optimal is that the estimate error, \( e^*(n) \), be orthogonal to each input sample in \( x(n) \).

**Interpretation:** The observations samples and error are orthogonal and contain no mutual “information”
Objective: Determine the minimum MSE

Approach: Use the optimal weights $w_0 = R^{-1}p$ in the MSE expression

$$J(w) = \sigma_d^2 - p^H w - w^H p + w^H R w$$

$$\Rightarrow J_{\text{min}} = \sigma_d^2 - p^H w_0 - w_0^H p + w_0^H R (R^{-1} p)$$

$$= \sigma_d^2 - p^H w_0 - w_0^H p + w_0^H p$$

$$= \sigma_d^2 - p^H w_0$$

Result:

$$J_{\text{min}} = \sigma_d^2 - p^H R^{-1} p$$

where the substitution $w_0 = R^{-1}p$ has been employed
**Objective:** Consider the excess MSE introduced by using a weighted vector that is *not* optimal.

\[ J(w) - J_{\text{min}} = (\sigma^2_d - p^Hw - w^Hp + w^HRw) - (\sigma^2_d - p^Hw_0 - w_0^Hp + w_0^HRw_0) \]

Using the fact that

\[ p = Rw_0 \quad \text{and} \quad p^H = w_0^HR \]

yields

\[ J(w) - J_{\text{min}} = -p^Hw - w^Hp + w^HRw + p^Hw_0 + w_0^Hp - w_0^HRw_0 \]

\[ = -w_0^HRw - w^HRw_0 + w^HRw + w_0^HRw_0 \]

\[ + w_0^HRw_0 - w_0^HRw_0 \]

\[ = -w_0^HRw - w^HRw_0 + w^HRw + w_0^HRw_0 \]

\[ = (w - w_0)^HR(w - w_0) \]

\[ \Rightarrow J(w) = J_{\text{min}} + (w - w_0)^HR(w - w_0) \]
Finally, using the eigenvalue and vector representation $R = QQ^H$

$$J(w) = J_{\text{min}} + (w - w_0)^H QQ^H (w - w_0)$$

or defining the eigenvector transformed difference

$$v = Q^H (w - w_0) \quad (*)$$

$$\Rightarrow J(w) = J_{\text{min}} + v^H \Omega v$$

$$= J_{\text{min}} + \sum_{k=1}^{M} \lambda_k v_k v_k^*$$

Result:

$$J(w) = J_{\text{min}} + \sum_{k=1}^{M} \lambda_k |v_k|^2$$

Note: (*) shows that $v_k$ is the difference $(w - w_0)$ projected onto eigenvector $q_k$
Example
Consider the following system

\[ v_1(n) \rightarrow \text{AR system} \rightarrow d(n) \rightarrow \text{Communication channel} \rightarrow x(n) \rightarrow u(n) + v_2(n) \rightarrow u(n) \rightarrow \text{FIR filter} \rightarrow \hat{d}(n) \]

Objective: Determine the optimal filter for a given system and channel
Specific Objective

Determine the optimal order two filter weights, $w_0$, for

$$H_1(z) = \frac{1}{1 + 0.8458z^{-1}} \quad \text{[AR process]}$$
$$H_2(z) = \frac{1}{1 - 0.9458z^{-1}} \quad \text{[communication channel]}$$

where $v_1(n)$ and $v_2(n)$ zero mean white noise processes with $\sigma_1^2 = 0.27$ and $\sigma_2^2 = 0.1$

**Note:** To determine $w_0$, we need:

- $R_u$ — auto-correlation of the received signal
- $p$ — the cross correlation between received signal $u(n)$ and the desired signal $d(n)$
Procedure: Consider $R_u$ first

Since $u(n) = x(n) + v_2(n)$, where $v_2(n)$ is white with $\sigma_2^2 = 0.1$ and is uncorrelated with $x(n)$

$$R_u = R_x + R_{v_2} = R_x + \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$$
Next, consider the relation between $x(n)$ and $v_1(n)$

$$X(z) = H_1(z) H_2(z) V_1(z)$$

where for the given systems

$$H_1(z) H_2(z) = \frac{1}{(1 + 0.8458z^{-1})(1 - 0.9458z^{-1})}$$

Converting to the time domain, we see $x(n)$ is an order 2 AR process

$$x(n) - 0.1x(n-1) - 0.8x(n-2) = v_1(n)$$

$$x(n) + a_1 x(n-1) + a_2 x(n-2) = v_1(n)$$
Since $x(n)$ is a real valued order two AR process, the Yule-Walker equations are given by

$$
\begin{bmatrix}
  r(0) & r(1) \\
  r^*(1) & r(0)
\end{bmatrix}
\begin{bmatrix}
  -a_1 \\
  -a_2
\end{bmatrix}
= 
\begin{bmatrix}
  r^*(1) \\
  r^*(2)
\end{bmatrix}
$$

$$
\begin{bmatrix}
  r(0) & r(1) \\
  r(1) & r(0)
\end{bmatrix}
\begin{bmatrix}
  -a_1 \\
  -a_2
\end{bmatrix}
= 
\begin{bmatrix}
  r(1) \\
  r(2)
\end{bmatrix}
$$

Solving for the coefficients:

$$
-a_1 = \frac{r(1)[r(0) - r(2)]}{r^2(0) - r^2(1)}
$$

$$
-a_2 = \frac{r(0)r(2) - r^2(1)}{r^2(0) - r^2(1)}
$$

**Question:** What are the known and unknown terms in this system?
**Note:** We must solve the system to obtain the unknown $r(\cdot)$ values

Noting $r(0) = \sigma^2_x$ and rearranging to solve for $r(1)$ and $r(2)$

$$r(1) = \frac{-a_1}{1 + a_2} \sigma^2_x$$

$$r(2) = \left(-a_2 + \frac{a_1^2}{1 + a_2}\right) \sigma^2_x$$

The Yule-Walker equations also stipulate

$$\sigma^2_{V_1} = r(0) + a_1 r(1) + a_2 r(2)$$

Next, utilize the determined $r(1), r(2)$ and given $a_1, a_2, \sigma^2_{V_1}$ values to determine $r(0) = \sigma^2_x$
Substituting (*) and (**) into (**), utilize \( r(0) = \sigma_x^2 \), and rearranging

\[
\sigma_v^2 = r(0) + a_1 r(1) + a_2 r(2)
\]

\[
= \sigma_x^2 + a_1 \left( -\frac{a_1}{1 + a_2} \right) \sigma_x^2 + a_2 \left( -a_2 + \frac{a_1^2}{1 + a_2} \right) \sigma_x^2
\]

\[
= \left( 1 + \frac{-a_1^2}{1 + a_2} - a_2^2 + \frac{a_1^2 a_2}{1 + a_2} \right) \sigma_x^2
\]

\[
\Rightarrow \sigma_x^2 = \left( \frac{1 + a_2}{1 - a_2} \right) \frac{\sigma_v^2}{(1 + a_2)^2 - a_1^2}
\]

Note: All correlation terms have been determined since \( r(0) = \sigma_x^2 \)
Using $a_1 = -0.1$, $a_2 = -0.8$, and $\sigma_{v_1}^2 = 0.27$,

$$
\sigma_x^2 = \left(\frac{1 + a_2}{1 - a_2}\right) \frac{\sigma_{v_1}^2}{(1 + a_2)^2 - a_1^2} = 1
$$

Thus $r(0) = 1$. Similarly

$$
r(1) = \frac{-a_1}{1 + a_2} \sigma_x^2 = 0.5
$$

and finally, $R_x$ is

$$
R_x = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}
$$
Recall the overall system

Putting the pieces of $R_u$ together

$$R_u = R_x + R_{v_2}$$

$$= \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix} + \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$$

$$= \begin{bmatrix} 1.1 & 0.5 \\ 0.5 & 1.1 \end{bmatrix}$$
Recall: The Wiener solution is given by $w_0 = R^{-1}p$

Note: $R$ is known, but $p$ is still to be determined

\[ p = E \left\{ \begin{bmatrix} d(n)x(n) \\ d(n)u(n-1) \end{bmatrix} \right\} \]

Recall

\[ X(z) = H_2(z)D(z) = \frac{D(z)}{1 - 0.9458z^{-1}} \]

or in the time domain

\[ x(n) - 0.9458x(n-1) = d(n) \]

Lastly, the observation is corrupted by additive noise

\[ u(n) = x(n) + v_2(n) \]
Thus

\[ E\{u(n)d(n)\} = E\{[x(n) + v_2(n)][x(n) - 0.9458x(n - 1)]\} \]
\[ = E\{x^2(n)\} + E\{x(n)v_2(n)\} - 0.9458E\{x(n)x(n - 1)\} \]
\[ - 0.9458E\{v_2(n)x(n - 1)\} \]
\[ = \sigma_x^2 + 0 - 0.9458r(1) - 0 \]
\[ = 1 - 0.9458 \left( \frac{1}{2} \right) \]
\[ = 0.5272 \]

Similarly,

\[ E\{u(n - 1)d(n)\} = E\{[x(n - 1) + v_2(n - 1)][x(n) - 0.9458x(n - 1)]\} \]
\[ = r(1) - 0.9458r(0) \]
\[ = -0.4458 \]
Thus

\[ p = \begin{bmatrix} 0.5272 \\ -0.4458 \end{bmatrix} \]

**Final Solution:**

\[ w_0 = R^{-1}p \]

\[ = \begin{bmatrix} 1.1 & 0.5 \\ 0.5 & 1.1 \end{bmatrix}^{-1} \begin{bmatrix} 0.5272 \\ -0.4458 \end{bmatrix} \]

\[ = \begin{bmatrix} 0.8360 \\ -0.7853 \end{bmatrix} \]

- The optimal filter weights are a function of the source signal statistics and the communications channel.

- **Question:** How do we optimized a filter of when the statistics are not known in closed form or *a priori*?