Random Signal Processing

DWIGHT F. MIX
UNIVERSITY OF ARKANSAS

PRENTICE HALL, Englewood Cliffs, New Jersey 07632

CHAPTER 8

Optimum Filtering

8.1 Mean-Square Estimation

Preview

Mean-square estimation seeks to minimize the mean-square error between the estimate and the quantity being estimated. That means we need to know second-order statistics—the means, variances, and covariances of the data—because the mean-square error depends only on these quantities. The mean and autocorrelation functions contain this information, and this is often available in random signal processing.

In Chapter 7 we introduced least-squares estimation. The terms "mean-square estimation" and "least-squares estimation" are similar, so you should be careful to avoid confusing them. Least-squares techniques use samples of the data to derive good estimates. Mean-square estimation is based on statistical averages.

Linear mean-square estimation is a special case of mean-square estimation. In order to show how it fits into the overall scheme, we begin with the simplest problem in mean-square estimation and gradually add complexity until we arrive at a general estimation problem. Then we will specialize this general problem to linear estimation and show how the orthogonality principle simplifies the problem.

Let $\hat{X}$ stand for an estimate of the random variable X. The error is the difference between the value of the random variable and our estimate:

$$e = X - \hat{X}$$

The estimate $\hat{X}$ is a number; X is a random variable. Hence the error e is a random variable. We square and then average this random variable to obtain the mean-square error:

$$E(e^2) = E[(X - \hat{X})^2] \tag{8.1.1}$$

The value of $\hat{X}$ that minimizes this expression is the minimum mean-square estimate of X.

Estimating the Random Variable X by a Constant

If $\hat{X} = c$, the error is $X - c$, and the mean-square error is

$$E(e^2) = E[(X - c)^2] = \int_{-\infty}^{\infty} (\alpha - c)^2 f_X(\alpha)\, d\alpha$$
$$= c^2 - 2cE(X) + E(X^2)$$

Differentiating with respect to c and setting the derivative to 0, we have

$$2c - 2E(X) = 0 \qquad \text{or} \qquad c = E(X) \tag{8.1.2}$$

Therefore we should use the average $E(X)$ to estimate X when we wish to minimize the mean-square error.

EXAMPLE 8.1.1. The random variable X is defined in Fig. 8.1.1. For the time being, ignore the other random variable in the diagram; we will use it later. The experiment consists of rolling a single die, and the experimental outcome determines a value for X (and the other random variable). Find a constant c to give the minimum mean-square estimate of X and calculate the resulting mean-square error.

SOLUTION: The average value of X gives the estimate.

$$c = E(X) = 0.5$$

The resulting mean-square error is

$$E(e^2) = E[(X - 0.5)^2] \approx 2.917$$

This illustrates a startling feature of mean-square estimation. The estimate need not be a possible value of X. Our estimate of 0.5 can never be correct, but the average square error will be less than that for any estimate that could possibly be correct. For example, if we fudge a little and choose as our estimate $c = 1$, we will be correct about one-sixth of the time, but the mean-square error increases to 3.167.

Here the experiment is performed with outcome ζ. This outcome ζ then determines a value of the random variable X, and it is this number we estimate. In this problem we make the estimate before performing the experiment (or

Y	4	1	0	1	4	9
X	−2	−1	0	1	2	3
ζ	⚀	⚁	⚂	⚃	⚄	⚅

Fig. 8.1.1. Definition of a random variable X.

at least before we have any information about the experimental outcome). In all of the following we perform the experiment first and provide partial information on which to base the estimate.

Estimating the Random Variable X given $Y(\zeta)$

Suppose that two random variables X and Y are defined on the same sample space. The experiment is performed and you are told the value of Y. The estimate $\hat{X}$ should incorporate this additional information, so $\hat{X}$ is a function of Y, $\hat{X} = g(Y)$. The mean-square error that we wish to minimize is given by

$$E(e^2) = E\{[X - g(Y)]^2\}$$

The function that minimizes this expression is

$$g(Y) = E(X|Y) \tag{8.1.3}$$

as shown below:

$$\begin{aligned} E\{[X - g(Y)]^2\} &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} [\alpha - g(\beta)]^2 f_{XY}(\alpha, \beta)\, d\alpha\, d\beta \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} [\alpha - g(\beta)]^2 f_X(\alpha|\beta) f_Y(\beta)\, d\alpha\, d\beta \\ &= \int_{-\infty}^{\infty} f_Y(\beta) \int_{-\infty}^{\infty} [\alpha - g(\beta)]^2 f_X(\alpha|\beta)\, d\alpha\, d\beta \end{aligned} \tag{8.1.4}$$

Let us define a new function $h(\beta)$ that is equal to the last integral in this expression:

$$h(\beta) = \int_{-\infty}^{\infty} [\alpha - g(\beta)]^2 f_X(\alpha|\beta)\, d\alpha \tag{8.1.5}$$

Now Eq. 8.1.4 can be written

$$E\{[X - g(Y)]^2\} = \int_{-\infty}^{\infty} f_Y(\beta)\, h(\beta)\, d\beta$$

We wish to choose $g(\beta)$ to minimize this expression. Note that $f_Y(\beta)$ is always positive because it is a probability density function. We have no control over $f_Y(\beta)$; this is the marginal pdf of the random variable Y. Hence we can operate only on $h(\beta)$. From Eq. 8.1.5 we see that $h(\beta)$ is also always positive. The product of a squared term times a density function is positive, so the integral of this positive expression is also positive. Therefore, if we choose $g(\beta)$ to minimize $h(\beta)$ for every value of β, this will minimize the mean-square error.

Expand Eq. 8.1.5 to obtain

$$h(\beta) = g^2(\beta) - 2g(\beta)\, E(X|\beta) + E(X^2|\beta)$$

Setting the derivative with respect to β equal to 0 gives

$$2g(\beta) - 2E(X|\beta) = 0$$

or

$$g(\beta) = E(X|\beta) \tag{8.1.6}$$

So to estimate X we should use the conditional expectation, the average value of the random variable X based on what we know about it. The estimate should be based on the given data $Y = \beta$, and the way to do this is to use the conditional expectation $E(X|Y = \beta)$.

This situation differs from the first (estimating X by a constant) in that the experiment is performed with outcome ζ, and we are allowed to observe the value of $Y(\zeta)$. If knowledge of Y allows us to determine ζ, then we can estimate $X(\zeta)$ exactly. That is, $E(X|\beta) = X$. Of course, this is the trivial case in which knowledge of Y gives us complete information about X. The other extreme is when knowledge of Y gives us no information about X, and $E(X|\beta) = E(X)$.

EXAMPLE 8.1.2. In Fig. 8.1.1 the random variable Y can take on one of four different numbers, 0, 1, 4, and 9. For each possible value of Y, find the best estimate of X, and calculate the resulting mean-square error.

SOLUTION: When $Y = 4$, X must be either -2 or 2 with equal probability. Therefore

$$g(4) = E(X|Y = 4) = 0$$

Similar reasoning applies to the other values of Y, giving

$$g(0) = E(X|Y = 0) = 0$$

$$g(1) = E(X|Y = 1) = 0$$

$$g(9) = E(X|Y = 9) = 3$$

Therefore we guess $\hat{X} = 0$ for every value of Y except for $Y = 9$. For that value of Y we guess $\hat{X} = 3$. This is shown in Fig. 8.1.2. The resulting mean-square error is given by

$$\begin{aligned} E(e^2) &= E[(X - \hat{X})^2] \\ &= (\tfrac{1}{6})[(-2)^2 + (-1)^2 + 0^2 + 1^2 + 2^2 + 0^2] = 1.667 \end{aligned}$$

Notice that this is less than the error in Example 8.1.1, as it should be. Here we use more knowledge to make the estimate.

The estimate determined by Eq. 8.1.6 is a nonlinear estimate because the values of $\hat{X}$ do not lie on a straight line through the origin. The following

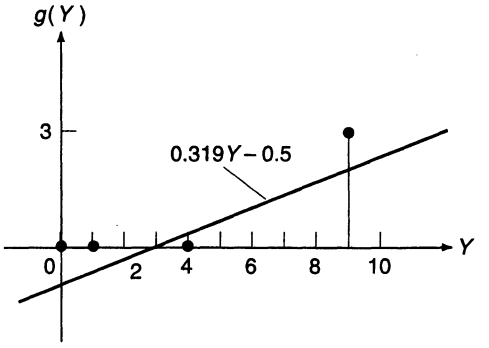


Fig. 8.1.2. Estimates for X.

example chooses the best straight-line fit, but this is not linear estimation because the straight line does not go through the origin.

EXAMPLE 8.1.3. Estimate the value of X given Y in Fig. 8.1.1 by points on a straight line,

$$\hat{X} = g(Y) = aY + b \tag{8.1.7}$$

SOLUTION: The only difference between this and the previous example is that we are restricting the form of $g(Y)$. It must be a straight line, so we need to solve only for the two constants, a and b. The mean-square error that we wish to minimize is given by

$$E(e^2) = E\{[X - (aY + b)]^2\} \tag{8.1.8}$$

Setting the partial derivative with respect to b equal to 0 gives

$$\frac{\partial}{\partial b} E(e^2) = 2b - 2E(X) + 2aE(Y) = 0$$

The value of b that minimizes Eq. 8.1.8 is therefore

$$b = m_X - am_Y \tag{8.1.9}$$

Using this value of b, we can write Eq. 8.1.8 as

$$\begin{aligned} E(e^2) &= E\{[(X - m_X) - a(Y - m_Y)]^2\} \\ &= \sigma_X^2 - 2a\mu_{11} + a^2\sigma_Y^2 \end{aligned}$$

where μ_{11} is the covariance between X and Y (see Eq. 4.2.2). Solving for a by setting the partial derivative with respect to a equal to 0 gives

$$a = \frac{\mu_{11}}{\sigma_Y^2} \tag{8.1.10}$$

For the random variables X and Y in Fig. 8.1.1 this gives

$$a = \frac{\mu_{11}}{\sigma_Y^2} = 0.319$$

$$b = m_X - am_Y = 0.5 - (0.319)(3.17) = -0.5$$

so that $g(Y) = 0.319Y - 0.5$. This function is plotted as the straight line in Fig. 8.1.2.

The resulting mean-square error is

$$E(e^2) = E[(X - aY - b)^2] = 1.99$$

Note that this error compares rather well with the optimum mean-square error of 1.67 from Example 8.1.2.

Review

This completes our introduction to mean-square estimation. In Section 8.3 we introduce linear mean-square estimation, which is nothing more than an estimate of the form $\hat{X} = aY$ for one data point Y. For more than one data value, the linear estimate is of the form $(\hat{X}) = \Sigma_{i=1}^{N} a_i Y_i$.

Mean-square estimation is a statistical procedure, which means that we need some statistical knowledge about the quantity we are estimating before we can derive the estimate. This is opposed to least-squares estimation, where we use the data alone to form the estimate. For mean-square estimation we need only second-order information, the means and correlations. We may be given more information, as in Example 8.1.2 where we had complete knowledge, but we use only the means and correlations in deriving our estimate.

8.2 Vector Spaces

Preview

Ask any engineer to define a vector, and she will likely answer, "a directed magnitude." This is a good answer; it is just wrong. Geometric vectors are directed magnitudes, but there are other kinds of vectors besides geometric vectors, and these other vectors don't have "magnitude and direction." Vectors, like anything else, are defined by listing properties. We list the important properties of geometric vectors in our definition below, and anything else that satisfies these properties is called a set of vectors. It turns out that many other things satisfy this definition, including random variables. This is important to us, since we will be using properties of vectors in our discussion of estimation theory.

In this section we define vector space, dot products, norms, metrics, and orthogonality. Then, in subsequent sections we will use these concepts to discuss mean-

square estimation procedures. This will lead eventually to Wiener and Kalman filtering.

DEFINITION 8.2.1. A vector space is a set $V = \{v_i\}$ together with a field of scalars $A = \{a_i\}$ that has the following two operations and seven properties.

1. We can add two vectors together and obtain a third vector. Thus we have a mechanism for combining two vectors to obtain a third.
2. We can multiply a vector by a scalar and obtain another vector. This mechanism combines a scalar with a vector to obtain another vector.

Using these two operations, vector addition and scalar multiplication, the following properties must hold for all $v_i \in V$ and all $a_i \in A$.

(1) $v_1 + v_2 = v_2 + v_1$
(2) $(v_1 + v_2) + v_3 = v_1 + (v_2 + v_3)$
(3) $a_1(v_1 + v_2) = a_1v_1 + a_1v_2$
(4) $(a_1 + a_2)v_i = a_1v_i + a_2v_i$
(5) $a_1(a_2v_i) = (a_1a_2)v_i$
(6) $1 \cdot v_i = v_i$
(7) There exists a unique vector v_0, called the zero vector, such that for all vectors v_i we have $0 \cdot v_i = v_0$, where 0 is the number 0.

Recall that a field is modeled after the real number system with the two operations addition and multiplication. In fact, we will have no need for any other field, so when we say "scalar" you can substitute "real number."

These are the properties that define a vector space. Any set of objects that collectively satisfies all of these properties is called a vector space. Notice that we have not defined a vector. Instead we defined a vector space, a set of vectors. A vector space is a set together with two functions, a mapping from $V \times V \rightarrow V$ that defines vector addition, and a mapping from $A \times V \rightarrow$ V that defines scalar multiplication. (The notation $V \times V$ and $A \times V$ stands for the set cross product.) Any set of objects with these two functions that satisfies the seven properties is a vector space. Furthermore, these vectors are equally as legitimate as geometric vectors, even though they may not have "magnitude" and "direction."

Ordinary geometric vectors satisfy these properties, and therefore form a vector space. To show this, let $v_1, v_2, \ldots$ be a set of ordinary geometric vectors. These are the familiar "directed magnitudes" of geometry. Let $a_1, a_2, \ldots$ be ordinary numbers (from the real number system). We will call these numbers "scalars," but they are just ordinary numbers from the field of real numbers (see Section 3.1). We can add two vectors together in any order (property 1). We can add three vectors by first adding $v_1 + v_2$ and then adding v_3, or we can add $v_2 + v_3$ before adding v_1, and we get the same result either way (property 2). Continuing in this way, you can see that geometric vectors satisfy each property listed.

But geometric vectors are not the only objects that satisfy Definition 8.2.1. All 3×2 matrices satisfy these properties. We don't usually call a matrix a vector, but all matrices of the same dimension form a vector space.

The set of real numbers satisfies these properties. Here the numbers play a dual role: They are both vector and scalar.

The set of all waveforms that can be generated in the laboratory satisfy these properties. Here, instead of $v_1, v_2, \ldots$, we have time functions $v_1(t)$, $v_2(t), \ldots$. We can add waveforms together to form another waveform. We can multiply by scalars to change the amplitude. And there is a zero vector, namely, $v_0(t) = 0$, for all time, which satisfies property 7. (These same statements apply equally to a set of discrete-time waveforms.) So you can see that waveforms (signals) are vectors.

Our interest in vectors stems from the fact that all random variables defined on the same sample space form a vector space. We can add two random variables and their sum is a third random variable. We can multiply a random variable by a scalar and the result is another random variable, and there is a zero random variable that assigns the number 0 to each experimental outcome. To illustrate, two random variables X and Y are defined on the sample space of die-toss outcomes in Fig. 8.2.1. We can add $X + Y$ and get another random variable. This means that we add the numbers for each experimental outcome. If the one-spot turns up, $X + Y = -2 + 4 = 2$. If the two-spot turns up, the sum is $1 - 1 = 0$. Continuing in this way, we can obtain a sum for each experimental outcome, which is another random variable. We can also multiply X by a scalar to obtain another random variable, and all seven properties are satisfied. So the set of all random variables that could be defined on the die-toss experiment forms a vector space. Notice that all random variables must be defined on the same sample space. Random variables defined on different sample spaces are not vectors in the same space.

So far, we have imposed little structure on our vector spaces. All we can do is multiply them by scalars and add two vectors together. We have said nothing about how to measure the length of a vector, or how to determine the distance between two vectors, or how to find the dot product of two vectors. In order to do geometry we must be able to do all three of these

Y	4	1	0	1	4	9
X	−2	−1	0	1	2	3
ζ						

Fig. 8.2.1. Random variables X and Y defined on the sample space of die-toss outcomes.

things. A vector space in which we can do geometry is called an *inner product* (or *dot product*) space. That is because once we have defined the dot product, a measure of length and distance is automatically imposed. So we begin by defining an inner product.

DEFINITION 8.2.2: Inner product. The symbol $\langle v_1|v_2\rangle$ denotes the operation of extracting a number from a pair of vectors. This is called the inner (or dot) product, and must satisfy the following four properties for all scalars and vectors.

(1) $\langle v_1|v_2\rangle = \langle v_2|v_1\rangle$
(2) $\langle v_1 + v_2|v_3\rangle \leq \langle v_1|v_3\rangle + \langle v_2|v_3\rangle$
(3) $\langle a_1 v_1|v_2\rangle = a_1\langle v_1|v_2\rangle$
(4) $\langle v_1|v_1\rangle \geq 0$, and $\langle v_1|v_1\rangle = 0$ if and only if $v_1 = v_0$ (the zero vector)

If the vector space $V = \{v_i\}$ satisfies these properties, it is called an inner product space, which means that we can do geometry in this space. The length of a vector v_1, denoted by $\|v_1\|$ and called the *norm,* is defined by

$$\|v_1\| = \langle v_1|v_1\rangle^{1/2} \tag{8.2.1}$$

The distance between two vectors is given in terms of the norm by

$$d(v_1, v_2) = \|v_1 - v_2\| \tag{8.2.2}$$

This gives us a hierarchy of measures: The inner product induces a norm (a measure of length), which, in turn, induces a measure of distance between two vectors. Incidentally, we can determine the "angle" θ between two vectors by the formula

$$\cos\theta = \frac{\langle v_1|v_2\rangle}{\|v_1\|\,\|v_2\|} \tag{8.2.3}$$

In the space $V = R^n$ over the field R of real numbers, the standard inner product of two vectors,

$$v_1 = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} \quad \text{and} \quad v_2 = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$$

is $\langle v_1|v_2\rangle = \sum_{i=1}^{n} a_i b_1$, which can be written in terms of matrices as $\langle v_1|v_2\rangle = v_1^t v_2$, where v_1^t is the transpose of the $(n \times 1)$ matrix v_1. The norm induced by this inner product is

$$\|v_1\| = \langle v_1|v_1\rangle^{1/2} = \left(\sum_{k=1}^{n} a_k^2\right)^{1/2}$$

and the metric (distance between two vectors) is

$$d(v_1, v_2) = \|v_1 - v_2\| = \left[\sum_{k=1}^{a}(a_k - b_k)^2\right]^{1/2}$$

These are the usual ways to measure length and distance, and they should be familiar from Euclidean geometry.

EXAMPLE 8.2.1. Find the inner product, the length, and the distance between the following vectors x and y. Use the usual dot product for Euclidean vectors.

$$x = \begin{bmatrix} 3 \\ -1 \\ 2 \\ 1 \end{bmatrix} \qquad y = \begin{bmatrix} 2 \\ 1 \\ -1 \\ 0 \end{bmatrix}$$

SOLUTION: The dot product is $\langle x|y\rangle = x^t y = 3$. The length of each vector is

$$\|x\| = [3 \cdot 3 + (-1)(-1) + 2 \cdot 2 + 1 \cdot 1]^{1/2} = \sqrt{15}$$
$$\|y\| = [2 \cdot 2 + 1 \cdot 1 + (-1)(-1) + 0 \cdot 0]^{1/2} = \sqrt{6}$$

The distance between them is

$$d(x, y) = \|x - y\| = [1^2 + (-2)^2 + 3^2 + 1^2]^{1/2} = \sqrt{15}$$

For another example, in the space of all continuous-time energy signals, the usual inner product is defined by

$$\langle v_1(t)|v_2(t)\rangle = \int_{-\infty}^{\infty} v_1(t)\, v_2(t)\, dt$$

This induces a norm given by

$$\|v(t)\| = \left[\int_{-\infty}^{\infty} v^2(t)\, dt\right]^{1/2}$$

And the metric is

$$d(v_1, v_2) = \left[\int_{-\infty}^{\infty} [v_1(t) - v_2(t)]^2\, dt\right]^{1/2}$$

EXAMPLE 8.2.2. Find the inner product, the norm of each waveform, and the distance between them for (see Fig. 8.2.2)

$$v_1(t) = e^{-t}u(t)$$
$$v_2(t) = u(t) - u(t-1)$$

SOLUTION:

$$\langle v_1|v_2\rangle = \int_0^1 e^{-t}\, dt = 1 - e^{-1} = 0.632$$

$$\|v_1\| = \left(\int_0^{\infty} e^{-2t}\, dt\right)^{1/2} = \sqrt{0.5}$$

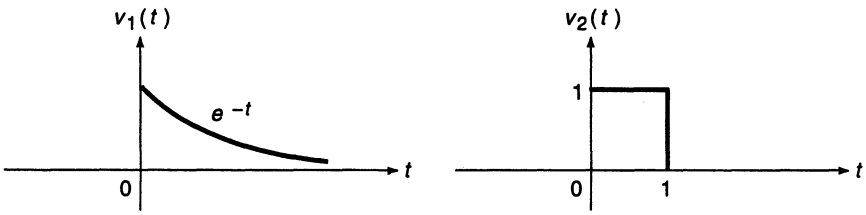


Fig. 8.2.2

$$\|v_2\| = \left(\int_0^1 1\,dt\right)^{1/2} = 1$$

$$d(v_1, v_2) = \left[\int_0^1 (1 - e^{-t})^2\,dt + \int_1^\infty e^{-2t}\,dt\right]^{1/2}$$
$$= (0.168 + 0.0677)^{1/2} = 0.484$$

Similar definitions define the usual inner product, norm, and the metric for discrete time energy signals.

$$\langle v_1(n) | v_2(n) \rangle = \sum_{n=-\infty}^{\infty} v_1(n) v_2(n)$$

$$\|v_1(n)\| = \left[\sum_{n=-\infty}^{\infty} v_1^2(n)\right]^{1/2}$$

$$d(v_1, v_2) = \left[\sum_{n=-\infty}^{\infty} [v_1(n) - v_2(n)]^2\right]^{1/2}$$

Our primary interest is in random variables. The usual dot product for random variables is simply the correlation:

$$\langle X | Y \rangle = E(XY) \tag{8.2.4}$$

The norm induced by this dot product is

$$\|X\| = [E(X^2)]^{1/2} \tag{8.2.5}$$

which is the root-mean-square (rms) value of the random variable X. This is the standard deviation if the mean of X is 0. The distance between two random variables X and Y is given by

$$d(X, Y) = \|X - Y\| = [E(X - Y)^2]^{1/2} \tag{8.2.6}$$

EXAMPLE 8.2.3. Find the inner product, the norm of each vector, and the distance between the random variables X and Y in Fig. 8.2.1.

SOLUTION: The inner product is

$$E(XY) = \sum_{i=1}^{6} \alpha_i \beta_i P[X = \alpha_i, Y = \beta_i]$$
$$= (\tfrac{1}{6})[(-2)(4) + (-1)(1) + 0 \cdot 0 + 1 \cdot 1 + 2 \cdot 4 + 3 \cdot 9] = 4.5$$

The norm of each random variable is

$$\|X\| = [E(X^2)]^{1/2} = \left(\sum_{i=1}^{6} \alpha_i^2 P[X = \alpha_i] \right)^{1/2}$$
$$= [(\tfrac{1}{6})(4 + 1 + 0 + 1 + 4 + 9)]^{1/2} = 1.78$$
$$\|Y\| = [(\tfrac{1}{6})(16 + 1 + 0 + 1 + 16 + 81)]^{1/2} = 4.378$$

The distance between them is

$$d(X, Y) = [E(X - Y)^2]^{1/2} = \left(\sum_{i=1}^{6} (\alpha_i - \beta_i)^2 P[X = \alpha_i, Y = \beta_i] \right)^{1/2}$$
$$= \{(\tfrac{1}{6})[(-2 - 4)^2 + (-1 - 1)^2 + 0 + 0 + (2 - 4)^2 + (3 - 9)^2]\}^{1/2}$$
$$= 3.65$$

Orthogonality

Two vectors are orthogonal if their inner product is 0. We are accustomed to geometric vectors that have the additional property that orthogonal vectors are 90° apart. Since the vectors we will use do not have an apparent "direction," you should not look for "right angles" when using orthogonality. For random variables, orthogonality simply means $E(XY) = 0$.

Notice that being uncorrelated and orthogonal are not the same thing. Uncorrelated means $E(XY) = E(X)\,E(Y)$. Orthogonal means $E(XY) = 0$. Only if the mean of either X or Y is 0 do these two terms mean the same thing.

The Cauchy–Buniakovshy–Schwartz (CBS) Inequality

If V is an inner-product space, which means we can do geometry, then the CBS inequality holds for any valid inner product:

$$|\langle u|v\rangle|^2 \leq \langle u|u\rangle\langle v|v\rangle \tag{8.2.7}$$

The proof follows from the properties of the inner product as follows: First, if $v = v_0$, the zero vector, then we have equality by property 4. If $v \neq v_0$, then we start with

$$0 \leq \langle u - \lambda v|u - \lambda v\rangle = \langle u|u\rangle - 2\lambda\langle u|v\rangle + \lambda^2\langle v|v\rangle \tag{8.2.8}$$

which holds for all λ. Let $\lambda = \langle u|v\rangle/\langle v|v\rangle$. Then Eq. 8.2.8 gives

$$0 \le \langle u|u\rangle - 2\frac{\langle u|v\rangle^2}{\langle v|v\rangle} + \frac{\langle u|v\rangle}{\langle v|v\rangle} = \langle u|u\rangle - \frac{\langle u|v\rangle^2}{\langle v|v\rangle}$$

which gives the CBS inequality in Eq. 8.2.7.

Here is an example of template matching using the CBS inequality. In order to store an image in computer memory, we sample the gray-scale values (how dark the image is) in small blocks called pixels. Thus, an $N \times M$ image has N rows and M columns of values, which may be stored and manipulated in a computer. Typical sizes are 128×128 or larger, but we will simplify this by using a 2×3 image array. Let

$$u = \begin{bmatrix} 3 & 1 & -2 \\ 1 & 2 & -3 \end{bmatrix} \qquad v = \begin{bmatrix} 1 & 3 & -3 \\ 2 & 1 & -2 \end{bmatrix}$$

The usual inner product for matrices multiplies corresponding pixels and sums the products. That is,

$$\langle u|v\rangle = \sum_i \sum_j u_{ij} v_{ij}$$

For the particular u and v above, the inner product is 22. We define the energy in image u as

$$E_u = \langle u|u\rangle = \|u\|^2$$

This gives

$$E_u = 28 \qquad E_v = 28$$

Thus the CBS inequality is satisfied because $|\langle u|v\rangle|^2$ is less than $\langle u|u\rangle\langle v|v\rangle$.

This example illustrates that for equal energy images, the maximum response (or correlation, or inner product) is given when the two images are identical, and this maximum response is determined by the CBS inequality.

Review

Notice that we did not define a vector in Definition 8.2.1. Instead, we defined a vector space, a set of objects that collectively satisfies the properties listed there. This means that we cannot determine that an object v by itself is or is not a vector. The entire set must satisfy all the properties of Definition 8.2.1. There are many sets besides geometric vectors that satisfy these properties, including all random variables defined on the same sample space.

The inner product, norm (length), and metric (distance) impose structure on a vector space. For random variables the usual inner product is $E(XY)$, which means the norm is the rms value, and the metric is the rms value of the difference between X and Y. See Eqs. 8.2.4, 8.2.5, and 8.2.6.

8.3 Linear Mean-Square Estimation

Preview

The concept of linearity applies to functions. When we say that a function is linear, we mean that the functional relationship between the domain and codomain is linear. When we say that an estimate is linear, we mean that the relationship between the data and the estimate is linear. When we say that a system is linear, we mean that the functional relationship between the input and output is linear; that is, the input–output function that describes the system is linear.

In this section we introduce the two-part test for linearity, and show how this applies to mean-square estimation. Then we introduce the idea of setting the error orthogonal to the data, which is the basic tenet in linear estimation.

A function is linear if it is additive and homogeneous. A function $f: X \rightarrow Y$ is said to be *additive* if

$$f(x_1 + x_2) = f(x_1) + f(x_2) \tag{8.3.1}$$

for all elements x_i in X. The addition on the left is in X, while the addition on the right is in Y.

We say that $f: X \rightarrow Y$ is *homogeneous* if

$$f(ax) = af(x) \tag{8.3.2}$$

for all scalars a and all elements x in X. The multiplication on the left is in X, while the multiplication on the right is in Y.

EXAMPLE 8.3.1. Let $y = 2x + 3$. Test this relationship for additivity and homogeneity.

SOLUTION: Try $x_1 = 5$ and $x_2 = 6$ in Eq. 8.3.1. Then we have on the left,

$$x_1 + x_2 = 5 + 6 = 11$$

so

$$f(x_1 + x_2) = 2 \cdot 11 + 3 = 25$$

The right side of Eq. 8.3.1 gives

$$f(x_1) = 2x_1 + 3 = 13$$
$$f(x_2) = 2x_2 + 3 = 15$$

giving

$$f(x_1) + f(x_2) = 28$$

Since we have found at least one case where Eq. 8.3.1 is not satisfied, the relationship $f(x) = 2x + 3$ is not additive (and therefore not linear).

To test for homogeneity, let $x = 5$, $a = 2$ in Eq. 8.3.2. Then on the left we have

$$ax = 10 \qquad \text{and} \qquad f(ax) = 2 \cdot 10 + 3 = 23$$

The right side of Eq. 8.3.2 becomes

$$af(x) = 2(2x + 3) = 2 \cdot 13 = 26$$

Now we have found one case where the function is not homogeneous, so it also cannot be linear for this reason.

As this example shows, linearity is a very restrictive condition. Not only must the relationship be a straight line, it must also pass through the origin. If in the example above the constant had been 0, giving a function of the form $y = ax$, then and only then would it be linear.

Testing a function for linearity can be reduced to an algorithm, called the *two-part test for linearity*. This algorithm, which implements Eqs. 8.3.1 and 8.3.2 in a systematic way, is illustrated in Fig. 8.3.1. Part 1 of the test accomplishes the left side of Eqs. 8.3.1 and 8.3.2, performing the addition and multiplication operations in the domain of f. Part 2 does the right side of these equations, thus performing addition and multiplication in the codomain of f. The function f is linear if $y_1 = y_2$ for all possible scalars a_1, a_2, and for all values in the domain x_1 and x_2.

EXAMPLE 8.3.2. Test $y = 2x + 3$ for linearity using the two-part test in Fig. 8.3.1.

SOLUTION: For arbitrary a_1, a_2, x_1, and x_2, part 1 gives

$$y_1 = f(a_1x_1 + a_2x_2) = 2(a_1x_1 + a_2x_2) + 3$$

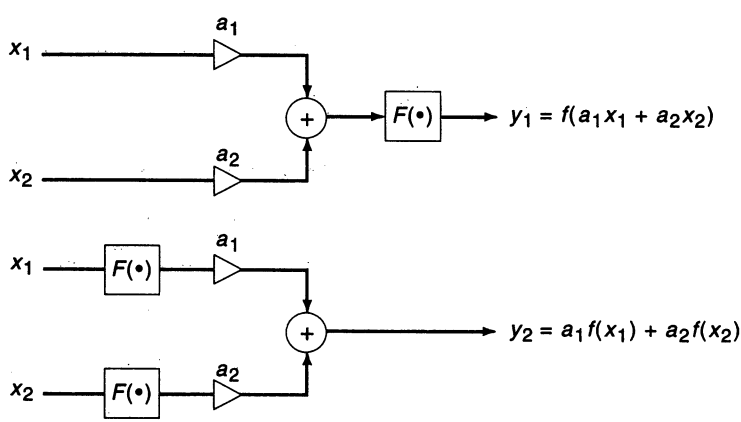


Fig. 8.3.1. The two-part test for linearity of f.

Part 2 gives

$$y_2 = a_1 f(x_1) + a_2 f(x_2) = a_1(2x_1 + 3) + a_2(2x_2 + 3)$$

The two terms y_1 and y_2 are not equal for all values of a_1, a_2, x_1, and x_2, so the function is not linear.

An *estimator* is a function, a relation between the data and the estimate. A filter, like a function, has an input and an output, so if we identify the filter input as data and the output as the estimate, the filter serves as the estimator. If the system is linear, then it is a linear estimator. A finite-impulse response (FIR) filter has the form

$$y(n) = \sum_{i=0}^{p} h_i x(n - i) \tag{8.3.3}$$

where $x(n)$ is the input, the h_i terms are the filter taps, and $y(n)$ is the output. To show linearity, apply the two-part test as follows. Let a_1 and a_2 be arbitrary constants, and let $x_1(n)$ and $x_2(n)$ be two input sequences. Part 1 of the test gives

$$y_1(n) = \sum_{i=0}^{p} h_i[a_1 x_1(n - i) + a_2 x_2(n - i)]$$

Part 2 gives

$$y_2(n) = \sum_{i=0}^{p} h_i a_1 x_1(n - i) + \sum_{i=0}^{p} h_i a_2 x_2(n - i)$$

The two terms in $y_2(n)$ can be combined to give the form $y_1(n)$, proving linearity.

A linear estimator has the form

$$\hat{d} = h_0 x(n) + h_1 x(n - 1) + \cdots + h_p x(n - p) \tag{8.3.4}$$

where $x(i)$ is the data, the h_i's are constants, and $\hat{d}$ is our estimate of the desired quantity d. Compare Eqs. 8.3.3 and 8.3.4 to see that the FIR filter shown in Fig. 8.3.2 will serve as one type of estimator for us. The other type is an IIR filter, which we will discuss later.

The usual paradigm has the filter input $x(n)$ equal to signal plus noise:

$$x(n) = s(n) + w(n) \tag{8.3.5}$$

The parameter that we wish to estimate can be a future value of the signal, $s(n + k)$, a past value of the signal, $s(n - k)$, or the present value of the signal, $s(n)$. This gives

Extrapolation: $d(n) = s(n + k)$
Interpolation: $d(n) = s(n - k)$
Smoothing: $d(n) = s(n)$

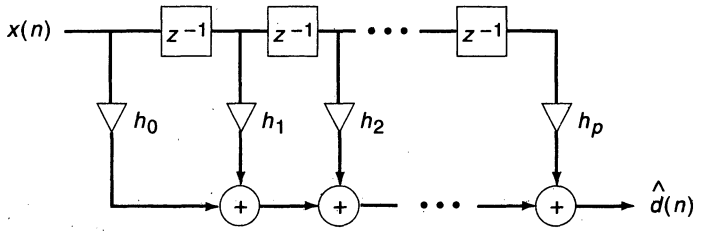


Fig. 8.3.2

where $d(n)$ is the desired parameter, the quantity to be estimated. The filter output in Fig. 8.3.2 is therefore $y(n) = \hat{d}(n)$ in each case. That is, regardless of whether we wish to predict a future value of the signal, estimate the present value, or interpolate a past value, the filter output can serve as the estimate of $d(n)$. The question to be answered in this chapter is: How do we select the filter coefficients in these filters to minimize the mean-square error? The answer lies in the orthogonality principle.

Set the Error Orthogonal to the Data

In Section 8.1 we established the general procedure for deriving a mean-square estimator: Find an expression for the squared error, take its expected value, and differentiate the expected value with respect to the parameters we seek in order to minimize the mean-square error. This leads to expressions for the unknown parameters such as Eqs. 8.1.9 and 8.1.10. We have a similar problem here, so we could follow the same procedure to arrive at expressions for the filter coefficients h_i. Less computation is required, however, by viewing random variables as vectors with the inner product between X and Y defined as $E(XY)$.

To give you an intuitive concept of minimizing error by setting the error orthogonal to the data, look at Fig. 8.3.3. The data is the vector x, the parameter to estimate is d, and the estimate is ax. You can see that the length of the error vector is minimum if the error is orthogonal to the data x. This is accomplished by choosing a so that the length of ax makes the error $d - ax$ a minimum. We will now apply this concept to several representative problems, and give a better explanation of why it is so after the examples.

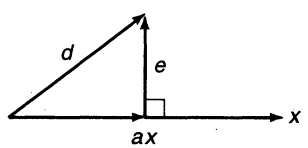


Fig. 8.3.3. The orthogonality principle.

Single Observation

Suppose that we are given one observation $x(n)$ and wish to estimate $s(n)$, where

$$x(n) = s(n) + w(n)$$
$$d(n) = s(n)$$

Set the error $e(n) = d(n) - \hat{d}(n)$ orthogonal to the data $x(n)$.

$$E\{[d(n) - \hat{d}(n)]x(n)\} = 0 \tag{8.3.6}$$

The estimate $\hat{d}(n)$ is $y(n) = h_0 x(n)$, giving

$$E\{[d(n) - h_0 x(n)]x(n)\} = 0$$

Since $d(n) = s(n)$, this gives

$$h_0 R_{XX}(0) = R_{SX}(0)$$

giving

$$h_0 = \frac{R_{SX}(0)}{R_{XX}(0)} \tag{8.3.7}$$

This optimum solution agrees with Eqs. 8.1.9 and 8.1.10. Both means are 0 here, giving $b = 0$ in Eq. 8.1.9. The value of a in Eq. 8.1.10 is the same as h_0 in Eq. 8.3.7.

EXAMPLE 8.3.3. Find the optimum h_0 and mean-square error in estimating $s(n)$ if the data is $x(n) = s(n) + w(n)$. The noise $w(n)$ is white Gaussian noise with zero mean and unit variance. The signal, which also has zero mean and is independent of the noise, has autocorrelation function given by

$$R_{SS}(n) = 0.9^{|n|}$$

SOLUTION: We need to find $R_{XX}(0)$ and $R_{SX}(0)$ to plug into Eq. 8.3.7. Since $x(n) = s(n) + w(n)$,

$$\begin{aligned} E[x^2(n)] = R_{XX}(0) &= E\{[s(n) + w(n)]^2\} \\ &= E[s^2(n) + E[s(n)w(n)] + E[w(n)s(n)] + E[w^2(n)] \\ &= R_{SS}(0) + R_{SW}(0) + R_{WS}(0) + R_{WW}(0) \end{aligned}$$

But the cross-correlation terms are 0, since $s(n)$ and $w(n)$ are zero-mean and independent. Since $w(n)$ is white with unit variance, $R_{WW}(n) = \delta(n)$. This gives

$$R_{XX}(0) = R_{SS}(0) + R_{WW}(0) = 0.9^0 + 1 = 2$$

The cross correlation $R_{SX}(0)$ is given by

$$E[s(n)x(n)] = E\{s(n)[s(n) + w(n)]\} = R_{SS}(0) = 1$$

Therefore

$$h_0 = \frac{R_{SX}(0)}{R_{XX}(0)} = \frac{1}{2}$$

The mean-square error is given by

$$\begin{aligned} E(e^2) &= E\{[s(n) - \hat{s}(n)]^2\} = E\{[s(n) - h_0x(n)]^2\} \\ &= E[s^2(n) - 2h_0s(n)x(n) + h_0^2x^2(n)] \\ &= R_{SS}(0) - 2h_0R_{SX}(0) + h_0^2R_{XX}(0) = 1 \end{aligned}$$

We will derive a simpler expression for the mean-square error in Eq. 8.3.11.

Multiple Observations

Suppose that we are given two observations, $x(n)$ and $x(n-1)$. Then our estimate is given by

$$y(n) = h_0x(n) + h_1x(n-1)$$

The error is

$$e(n) = d(n) - y(n) = d(n) - h_0x(n) - h_1x(n-1)$$

Therefore, setting the error orthogonal to the data gives two equations,

$$E\{[d(n) - h_0x(n) - h_1x(n-1)]x(n)\} = 0 \qquad (8.3.8a)$$

$$E\{[d(n) - h_0x(n) - h_1x(n-1)]x(n-1)\} = 0 \qquad (8.3.8b)$$

Taking expected values and transferring terms gives

$$\begin{aligned} h_0R_{XX}(0) + h_1R_{XX}(-1) &= R_{DX}(0) \\ h_0R_{XX}(1) + h_1R_{XX}(0) &= R_{DX}(1) \end{aligned}$$

These two equations in two unknowns allow us to solve for the filter coefficients to produce the optimum linear mean-square estimate. In matrix form they are given by

$$\begin{bmatrix} R_{XX}(0) & R_{XX}(-1) \\ R_{XX}(1) & R_{XX}(0) \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \end{bmatrix} = \begin{bmatrix} R_{DX}(0) \\ R_{DX}(1) \end{bmatrix} \qquad (8.3.9)$$

EXAMPLE 8.3.4. For the signal in Example 8.3.3, suppose that we are given two observations. Find the optimum linear estimate and the mean-square error.

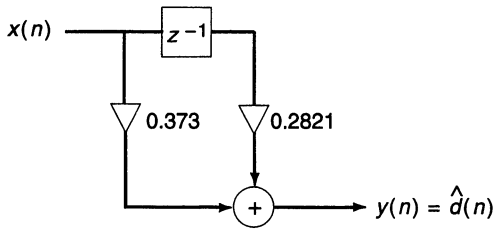


Fig. 8.3.4. The optimum filter.

SOLUTION: The quantities that go into Eq. 8.3.9 are

$$\begin{bmatrix} 2 & 0.9 \\ 0.9 & 2 \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0.9 \end{bmatrix}$$

giving $h_0 = 0.3730$ and $h_1 = 0.2821$. This first-order filter is shown in Fig. 8.3.4. The mean-square error is

$$E(e^2) = E\{[s(n) - h_0x(n) - h_1x(n-1)]^2\} = 0.373$$

There is always some confusion about the sign on the arguments in the expressions for the correlation functions. For example, how do we know just where to put $R_{XX}(1)$ and $R_{XX}(-1)$ in Eq. 8.3.9? The answer is that it really does not matter as long as you are consistent. The convention adopted in this text is that $E[x(i)x(j)] = R_{XX}(i-j)$. We subtract the second argument from the first. This convention gives the following results:

$$E[x(n)y(n-1)] = R_{XY}(1)$$
$$E[x(n-2)y(n)] = R_{XY}(-2)$$
$$E[x(n-2)y(n-3)] = R_{XY}(1)$$

The Projection Theorem

For more than two data values we can write Eq. 8.3.8 as

$$E\{[d(n) - h_0x(n) - \cdots - h_px(n-p)]x(n-i)\} = 0, \qquad i = 0, \cdots, p \tag{8.3.10}$$

This is really $p + 1$ equations, one for each i. The *projection theorem* says that the mean-square error is minimum if the coefficients h_i are chosen to make the error orthogonal to the data $x(n)$. We previously demonstrated this with geometric vectors, but it is time for a better explanation. We can show that this is true by taking the customary approach to minimization problems, i.e., by setting the derivatives of the mean-square error with respect to the parameters equal to 0.

$$\frac{\partial}{\partial h_i} E(e^2) = E\{2[d(n) - h_0x(n) - \cdots - h_px(n-p)][-x(n-i)]\} = 0$$

This gives Eq. 8.3.10, so we conclude that setting the error orthogonal to the data produces the optimum linear estimator.

Here is the simpler expression for the mean-square error we promised. From Eq. 8.3.10 we can see that the error is orthogonal to any linear combination of the data, giving

$$\langle (d - \hat{d}) | [a_0 x(n) + \cdots + a_p x(n-p)] \rangle = 0 \qquad \text{for any } a_0, \ldots, a_p$$

Since $\hat{d}$ is itself a linear combination of the data, this gives $\langle (d - \hat{d}) | \hat{d} \rangle = 0$; that is, the error is orthogonal to the estimate. This can be used to simplify the expression for the minimum mean-square error:

$$E(e^2) = E[(d - \hat{d})^2] = E[(d - \hat{d})d] - E[(d - \hat{d})\hat{d}]$$

But this last term is 0, giving

$$E(e^2) = E[(d - \hat{d})^2] = E[(d - \hat{d})d] \tag{8.3.11}$$

Thus, to calculate the mean-square error we find the inner product of the error with the desired quantity d. (Note that Eq. 8.3.11 is valid only if the optimum estimate is used. In other words, if some estimate is used that does not make the error orthogonal to any linear combination of the data, then our argument is not valid, and the more general formula must be used to calculate the mean-square error.)

EXAMPLE 8.3.5. Suppose that the input signal $x(n)$ from our previous example represents the flight path of an enemy aircraft, and suppose that we wish to predict its future value, $x(n + 1)$. Thus the input signal $x(n)$ is the sum of signal $s(n)$ plus noise $x(n)$, and we wish to (a) find the optimum first-order FIR filter and resulting mean-square error; (b) find the optimum second-order FIR filter and resulting mean-square error.

SOLUTION: (a) Here $d(n) = x(n + 1)$. Setting the error orthogonal to the data $x(n)$ and $x(n - 1)$ gives

$$E\{[x(n+1) - h_0 x(n) - h_1 x(n-1)]x(n)\} = 0$$
$$E\{[x(n+1) - h_0 x(n) - h_1 x(n-1)]x(n-1)\} = 0$$

Taking the expected values and transferring terms gives

$$h_0 R_{XX}(0) + h_1 R_{XX}(-1) = R_{XX}(1)$$
$$h_0 R_{XX}(1) + h_1 R_{XX}(0) = R_{XX}(2)$$

Putting this in matrix form and substituting values gives

$$\begin{bmatrix} 2 & 0.9 \\ 0.9 & 2 \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \end{bmatrix} = \begin{bmatrix} 0.9 \\ 0.81 \end{bmatrix}$$

Solving these two equations gives $h_0 = 0.3357$ and $h_1 = 0.2539$.

The mean-square error is given by $E[(d - \hat{d})d]$ or

$$\begin{aligned} E(e^2) &= E\{[x(n+1) - h_0x(n) - h_1x(n-1)]x(n+1)\} \\ &= R_{XX}(0) - 0.3357R_{XX}(1) - 0.2539R_{XX}(2) = 1.4922 \end{aligned}$$

(b) To find the optimum second-order filter we set the error orthogonal to the data for three observations, resulting in

$$\begin{aligned} h_0R_{XX}(0) + h_1R_{XX}(-1) + h_2R_{XX}(-2) &= R_{XX}(1) \\ h_0R_{XX}(1) + h_1R_{XX}(0) + h_2R_{XX}(-1) &= R_{XX}(2) \\ h_0R_{XX}(2) + h_1R_{XX}(1) + h_2R_{XX}(0) &= R_{XX}(3) \end{aligned}$$

This in matrix form with values gives

$$\begin{bmatrix} 2 & 0.9 & 0.81 \\ 0.9 & 2 & 0.9 \\ 0.81 & 0.9 & 2 \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ h_2 \end{bmatrix} = \begin{bmatrix} 0.9 \\ 0.81 \\ 0.729 \end{bmatrix}$$

which has the solution $h_0 = 0.297$, $h_1 = 0.202$, and $h_2 = 0.153$ as shown in Fig. 8.3.5. The mean-square error is

$$\begin{aligned} E(e^2) &= E[(d - \hat{d})d] \\ &= E\{[x(n+1) - h_0x(n) - h_1x(n-1) - h_2x(n-2)]x(n+1)\} \\ &= R_{XX}(0) - 0.297R_{XX}(1) - 0.202R_{XX}(2) - 0.153R_{XX}(3) \\ &= 1.4575 \end{aligned}$$

Notice that the addition of one more term h_2 decreases the mean-square error from 1.4922 to 1.4575.

Review

We did not discuss continuous-time systems in this section because most FIR filters are discrete-time systems, although we had an example of a FIR continuous-time system in Section 6.6. In any event, we know how to design discrete-time FIR systems, and so our discussion was confined to that subject. We design them for mean-square estimation by setting the error orthogonal to the data and solving

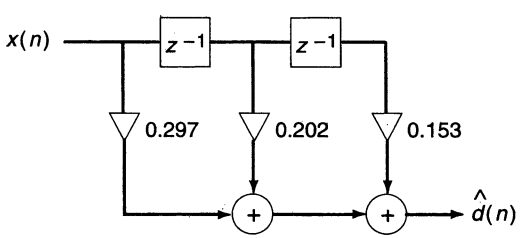


Fig. 8.3.5

the resulting simultaneous equations for the filter coefficients h_i. The mean-square error is found by taking the inner product between the error and $d(n)$, the quantity we are trying to estimate.

8.4 Spectral Factorization

Preview

In this section we introduce the concepts of minimum-phase systems, the innovations process, the Paley–Wiener theorem, and spectral factorization. Each of these concepts applies equally to continuous-time and discrete-time systems, but we will present only the discrete-time part of these concepts. We present all of these in this section because they are connected by the following sequence of facts.

A causal minimum-phase system is stable and has a causal stable inverse. The innovations filter that we will introduce for the stochastic processes of interest are minimum-phase. Spectral factorization comes into all this because the innovations filter is one of the factors derived by spectral factorization, and the Paley–Wiener theorem defines the conditions under which this is true.

If this seems horribly complicated, do not give up, for we will use all this in presenting the causal IIR Wiener filter in the next section.

DEFINITION 8.4.1 A minimum-phase polynomial has all its zeros inside the unit circle in the Argand diagram. Conversely, a maximum-phase polynomial has all its zeros outside the unit circle. A polynomial with some zeros inside or on the unit circle, while others are outside the unit circle, is neither minimum- nor maximum-phase.

DEFINITION 8.4.2. A minimum-phase system is a causal linear system with rational transfer function

$$H(z) = \frac{B(z)}{A(z)}$$

where both $A(z)$ and $B(z)$ are minimum-phase polynomials. A maximum-phase system is one where both $A(z)$ and $B(z)$ are maximum-phase polynomials.

EXAMPLE 8.4.1. The pole-zero plots for two systems H_1 and H_2 are shown in Fig. 8.4.1, where

$$H_1(z) = \frac{1}{1 - 0.5z^{-1}} = \frac{z}{z - 0.5}$$

$$H_2(z) = \frac{z - 2}{z^2 - 0.3z - 0.4} = \frac{z - 2}{(z + 0.5)(z - 0.8)}$$

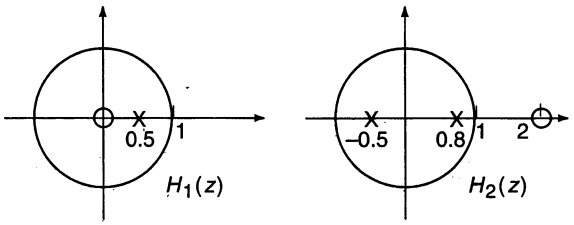


Fig. 8.4.1. Pole-zero locations for $H_1(z)$ and $H_2(z)$.

The first system is minimum-phase because all poles and zeros are inside the unit circle. The second system is not minimum-phase because the zero is outside the unit circle. Recall that all poles must be inside the unit circle for the system to be stable. Therefore, each of the systems above is stable.

Let us pause to remind you about the relation between pole location and stability. The transfer function can be put in the form

$$H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z + b_2 z^2 + \cdots + b_m z^m}{a_0 + a_1 z + a_2 z^2 + \cdots + a_n z^n}$$
$$= C \frac{(z - z_1)(z - z_2) \cdots (z - z_m)}{(z - p_1)(z - p_2) \cdots (z - p_n)}$$

where C is a constant, $z_1, z_2, \ldots, z_m$ are the zeros, and $p_1, p_2, \ldots, p_n$ are the poles of the transfer function $H(z)$. Collectively, the poles and zeros are called *singularities.* A system is stable if all its poles are inside the unit circle because such poles give rise to exponentially decaying terms in the source-free response. The partial fraction expansion of $H(z)$ has terms of the form

$$\frac{C_i z}{z - p_i}$$

The corresponding time functions are of the form $C_i(p_i)^k$, where k is time. As time increases, these terms decay to 0 because $|p_i| < 1$. Hence the unit circle serves as the boundary between stability and instability.

We can change a nonminimum-phase system into one with minimum phase while maintaining the same magnitude response $|H(\omega)|$ as follows: If no poles or zeros are on the unit circle, move all singularities from outside the unit circle to their conjugate reciprocal locations inside the unit circle. This alters only the phase. To relocate a zero at z_0, multiply $H(z)$ by

$$\frac{1 - z_0^* z}{z - z_0} = \frac{z^{-1} - z_0^*}{1 - z_0 z^{-1}} = (-z_0^*) \frac{z - 1/z_0^*}{z - z_0} \tag{8.4.1}$$

where z_0^* is the complex conjugate of z_0. To relocate a pole, multiply $H(z)$ by the reciprocal of this expression.

EXAMPLE 8.4.2. If we multiply $H_2(z)$ from Example 8.4.1 by Eq. 8.4.1 to move the zero inside the unit circle, we obtain

$$H_3(z) = H_2(z) \cdot (-2)\left(\frac{z - 0.5}{z - 2}\right) = -2\frac{z - 0.5}{(z + 0.5)(z - 0.8)}$$

The pole-zero plot for H_3 is shown in Fig. 8.4.2. Notice that $|H_2(\omega)| = |H_3(\omega)|$ for all ω. Only the phase has changed. In fact, the phase of $H_3(z)$ has magnitude less than or equal to the phase for $H_2(z)$ for all ω. This is the reason for the name "minimum phase."

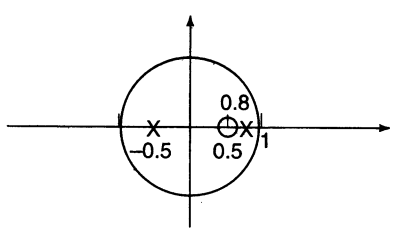
Minimum-phase systems are important because they are stable and their inverses are also stable. For example, $H_1(z)$ in Example 8.4.1 represents a causal stable system with impulse response

$$h_1(n) = 0.5^n u(n)$$

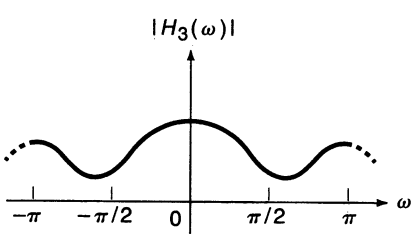
The inverse system is

$$H_1^{-1}(z) = 1 - 0.5z^{-1} \leftrightarrow \delta(n) - 0.5\,\delta(n - 1)$$

which is also stable and causal.



(a)



(b)

Fig. 8.4.2. Pole-zero (a) location and (b) magnitude response for $H_3(\omega)$.

In Sections 6.4 and 6.5 we illustrated the fact that when white noise with variance σ^2 is supplied to a system with rational transfer function $H(z)$, the output has an autocorrelation function given by

$$R_{YY}(l) = \sigma^2 r_{hh}(l)$$

and a power spectrum given by

$$G_{YY}(z) = \sigma^2 |H(z)|^2 = \sigma^2 H(z)H(z^{-1})$$

If $H(z)$ is a minimum-phase causal system, then $H(z^{-1})$ is a maximum-phase anticausal system. This gives us a unique way to represent the autocorrelation function of such a signal, i.e., in terms of a minimum-phase causal system function $H(z)$. Processes that can be represented this way are called *regular,* and they satisfy the Paley–Wiener condition.

The Paley–Wiener condition is a property of Fourier transforms that allow us to state that when the power spectrum of a process y satisfies the condition

$$\int_{-\pi}^{\pi} |\ln G_{YY}(\omega)| \, d\omega < \infty \tag{8.4.2}$$

then the spectral density can be factored as

$$G_{YY}(z) = K|H(z)|^2 = K H_c(z)H_a(z) \tag{8.4.3}$$

where K is a constant determined by the input power, $H_c(z)$ is a minimum-phase causal system function, and $H_a(z)$ is a maximum-phase anticausal system function. The subscript on H_c stands for "causal," while the a on H_a stands for "anticausal." H_c and H_a are related to each other by

$$H_a(z) = H_c(z^{-1}) \tag{8.4.4}$$

With this notation we are now in a position to describe the innovations representation and the innovations process for a stochastic process y. Figure 8.4.3 shows two systems, $H_c(z)$ and $H_c^{-1}(z)$, which means simply that they are reciprocal functions:

$$H_c^{-1}(z) = \frac{1}{H_c(z)}$$

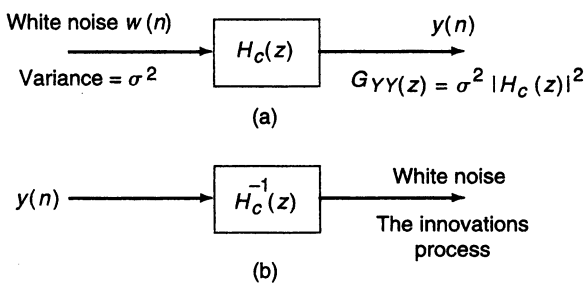


Fig. 8.4.3. The innovations representation (a) and the innovations process (b).

The first system, $H_c(z)$, has white noise $w(n)$ with variance σ^2 as its input. The output process has the spectrum given by Eq. 8.4.3, and is called the *innovations representation.*

Since the second system is the inverse of $H_c(z)$, it "undoes" what $H_c(z)$ does. Thus, if we supply the inverse filter with a signal $y(n)$ that has spectrum $G_{YY}(z)$, the output will be white noise. This white noise derived from $y(n)$ is called the *innovations process.* (This is not just any white noise: It is derived from y.)

Wiener coined the term "innovations" because it represents the unpredictable part of $y(n)$. Notice that we have used the term "innovations" in two different (and opposite) ways here. You should not confuse the innovations representation in Fig. 8.4.3a with the innovations process that is the output of the system in Fig. 8.4.3b. Also notice that we have discussed three filters, all different. They are $H_c(z)$, $H_c^{-1}(z)$, and $H_a(z) = H_c(z^{-1})$.

Given a regular process with spectrum $G_{YY}(z)$, the procedure for finding the innovations representation is called *spectral factorization.* This is not difficult, but you must be careful to derive minimum-phase systems from $G_{YY}(z)$; otherwise the representation will not be the desired innovations representation. Here are some examples.

EXAMPLE 8.4.3. Find the innovations representation for a stochastic process with spectrum

$$G_{XX}(z) = \frac{-2}{z - 2.5 + z^{-1}}$$

SOLUTION: Factoring the denominator, we can write

$$G_{XX}(z) = \frac{-2}{(1 - 0.5z^{-1})(z - 2)}$$

Now if we divide top and bottom by -2, we get

$$G_{XX}(z) = \frac{1}{(1 - 0.5z^{-1})(1 - 0.5z)}$$

This gives $K = 1$ in Eq. 8.4.3 and

$$H_c(z) = \frac{1}{1 - 0.5z^{-1}}$$

$$H_c^{-1}(z) = 1 - 0.5z^{-1}$$

and

$$H_a(z) = \frac{1}{1 - 0.5z}$$

EXAMPLE 8.4.4. Find the innovations model for a process with an autocorrelation function given by

$$R_{YY}(l) = 4(0.9)^{|l|}$$

SOLUTION: A first-order impulse response $h(n) = a^n u(n)$ has a corresponding second-order impulse response found by correlating $h(n)$ with itself. The results give

$$r_{hh}(l) = \left[\frac{1}{1-a^2}\right] a^{|l|} = \frac{1}{0.19}(0.9)^{|l|} \qquad \text{since} \qquad a = 0.9$$

Since $R_{YY}(l) = \sigma^2 r_{hh}(l)$, we see that $\sigma^2 = 4(0.19) = 0.76$.

The spectrum corresponding to this type of autocorrelation function is given by

$$a^{|l|} \leftrightarrow \frac{1-a^2}{(1-az^{-1})(1-az)} \tag{8.4.5}$$

(This is a good formula to memorize.) Therefore $G_{YY}(z)$ is

$$G_{YY}(z) = \frac{4(1-0.9^2)}{(1-0.9z^{-1})(1-0.9z)}$$
$$= \frac{0.76}{(1-0.9z^{-1})(1-0.9z)}$$

Therefore $K = 0.76$ and

$$H_c(z) = \frac{1}{1-0.9z^{-1}}$$
$$H_c^{-1}(z) = 1 - 0.9z^{-1}$$

and

$$H_a(z) = \frac{1}{1-0.9z}$$

While we are preparing for the next section, there is one more topic we need to discuss. Figure 8.4.4 shows a cascade of two systems, h_1 and h_2. The input signal $x(n)$ drives h_1 to produce $v(n)$, which drives h_2 to produce $d(n)$. Thus there is a cause-and-effect relationship between these signals, so

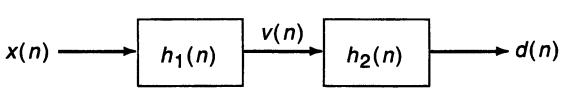


Fig. 8.4.4. A cascade of two systems.

if we know the correlation between $x(n)$ and $d(n)$, we should be able to find the correlation between $v(n)$ and $d(n)$. This is the relationship we will need in the next section.

Since R_{DX} is known, and x and v are related by convolution, we can find R_{DV} in terms of R_{DX} as follows. Write

$$v(n) = \sum_{k=-\infty}^{\infty} h_1(n-k)x(k)$$

Multiply this expression by $d(n-l)$ and take the expected value to get

$$R_{VD}(l) = E[v(n)d(n-l)] = E\left\{\left[\sum_{k=-\infty}^{\infty} h_1(n-k)x(k)\right] d(n-l)\right\}$$

$$= \sum_{k=-\infty}^{\infty} h_1(n-k)R_{XD}(k-n+l)$$

Change the variable. Let $\lambda = n - k$.

$$R_{VD}(l) = \sum_{\lambda=-\infty}^{\infty} h_1(\lambda)R_{XD}(l-\lambda) = h_1(l) * R_{XD}(l)$$

Reversing the subscripts gives

$$R_{DV}(l) = h_1(-l) * R_{DX}(l) \tag{8.4.6}$$

This relationship between R_{DV} and R_{DX} seems rather strange because $h_1(-l)$ is the time-reversed or anticausal version of $h_1(l)$, but this is the relation we will need in the next section.

Review

A signal with a specified spectrum $G_{YY}(z)$ can be generated by supplying a unique filter $H_c(z)$ with white noise. Provided Eq. 8.4.2 is satisfied, the output signal $y(n)$ then has the spectrum

$$G_{YY}(z) = K|H_c(z)|^2 = KH_c(z)H_c(z^{-1})$$

From this we obtained three filters of interest (at least they will be of interest in the next section). They are $H_c(z)$, $H_c^{-1}(z)$, and $H_a(z) = H_c(z^{-1})$. These are the filters we will need to define the optimum IIR Wiener filter.

Physically realizable (causal) minimum-phase systems are stable and have stable inverses. Thus, if $H_c(z)$ is minimum-phase, we are assured that its inverse $H_c^{-1}(z)$ exists and is stable. This is why the concept of minimum-phase systems is important to us. Spectral factorization is the procedure for decomposing a given spectrum into components K, H_c, and H_a so that H_c is minimum phase. These are the skills we will need in the next section.

8.5 IIR Wiener Filters

Preview

In this section we describe how to derive the optimum IIR filter to perform mean-square estimation, so we combine the two subjects of system theory and mean-square estimation. This marriage of the two disciplines is called *Wiener filtering,* after Norbert Wiener (1894–1964), who originated the concept.

In the preface to his book, *Statistical Theory of Communication,* Prof. Y. W. Lee explains that he used Weiner's new work, *The Extrapolation, Interpolation and Smoothing of Stationary Time Series,* to teach the first course in this discipline at MIT in 1947. His course notes evolved into his book published by Wiley in 1960. Wiener's work was often referred to as the "yellow peril" because of its yellow cover and its difficult mathematics. The main source of difficulty was the system theory in the document, not the mean-square estimation. Since that time others have discovered easier ways to explain the necessary system theory, and it is this theory that we now explore in this section.

Wiener's original work was for continuous-time systems, since that was all they had at the time. The digital computer was essentially unknown in 1939 (the copyright date on Weiner's document), for John von Neumann did not invent the stored program concept until about 1944. We are leaving out the continuous-time system derivation for two reasons. The discrete-time system theory, which we will present, parallels Wiener's work, and most applications now are to discrete-time systems.

A linear system output $y(n) = h(n) * x(n)$ serves as the estimate $\hat{d}(n)$ for some desired part of the input $x(n)$. The error is $e(n) = d(n) - y(n)$ and the data are $x(n)$, $-\infty < n < \infty$. We first assume that the data are available for all time, which includes the future. This will lead to physically unrealizable filters and serve as a precursor for the more complex realizable filters.

Abiding by the orthogonality principle, we set the error orthogonal to the data, giving

$$E\{[d(n) - y(n)]x(l)\} = 0, \qquad -\infty < l < \infty$$

Substituting $y(n) = \sum_{k=-\infty}^{\infty} h(k)\, x(n-k)$ and taking expected values gives

$$\sum_{k=-\infty}^{\infty} h(k) R_{XX}(n-k-l) = R_{DX}(n-l), \qquad -\infty < l < \infty$$

The number of variables can be reduced if we set $\lambda = n - l$.

$$\sum_{k=-\infty}^{\infty} h(k) R_{XX}(\lambda - k) = R_{DX}(\lambda), \qquad -\infty < \lambda < \infty \tag{8.5.1}$$

This is called the *Wiener–Hopf equation.* Since the term on the left is the convolution summation, we can write it in the form $h(l) * R_{XX}(l) = R_{DX}(l)$.

The mean-square error, found by the inner product of the error with $d(n)$, is given by

$$E(e^2) = R_{DD}(0) - \sum_{k=-\infty}^{\infty} R_{DX}(k)h(k) \tag{8.5.2}$$

Solving for the optimum filter coefficients from Eq. 8.5.1 is easy. That is the reason we begin with noncausal filters. Taking z transforms and solving for $H(z)$ gives

$$H(z)G_{XX}(z) = G_{DX}(z)$$

or

$$H(z) = \frac{G_{DX}(z)}{G_{XX}(z)} \tag{8.5.3}$$

where $G_{XX}(z)$ and $G_{DX}(z)$ are the z transforms of $R_{XX}(n)$ and $R_{DX}(n)$, respectively.

A common situation has the input signal $x(n)$ equal to the sum of signal plus zero-mean independent noise, and the desired quantity is the signal. When the signal is a first-order low-pass process and the noise is white, we have

$$x(n) = s(n) + w(n) \qquad d(n) = s(n)$$

$$R_{SS}(l) = Aa^{|l|} \qquad R_{WW}(l) = B\delta(l)$$

Solving for the spectral densities to substitute into Eq. 8.5.3 gives (see Eq. 8.4.5)

$$\begin{aligned} G_{XX}(z) &= G_{SS}(z) + G_{WW}(z) \\ &= \frac{A(1-a^2)}{(1-az^{-1})(1-az)} + B, \qquad |a| < |z| < \left|\frac{1}{a}\right| \end{aligned} \tag{8.5.4}$$

$$G_{DX}(z) = G_{SS}(z) = \frac{A(1-a^2)}{(1-az^{-1})(1-az)}, \qquad |a| < |z| < \left|\frac{1}{a}\right| \tag{8.5.5}$$

We may now substitute these quantities into Eq. 8.5.3 and solve for $H(z)$ to obtain the optimum filter. From $H(z)$ we can find $h(n)$. Here is an example.

EXAMPLE 8.5.1. Let $A = 1$, $B = 1$, and $a = 0.9$ in the specification above. Find the optimum filter $h(n)$.

SOLUTION: Substituting into Eqs. 8.5.4 and 8.5.5 gives

$$G_{XX}(z) = \frac{z^2 - (2/a)z + 1}{(z - a)(z - 1/a)}$$

$$G_{DX}(z) = \frac{z(a - 1/a)}{(z - a)(z - 1/a)}$$

Hence

$$H(z) = \frac{G_{DX}(z)}{G_{XX}(z)} = \frac{z(a - 1/a)}{z^2 - (2/a)z + 1} = \frac{-0.2111z}{z^2 - 2.222z + 1}$$

Using partial fraction expansion gives

$$\frac{H(z)}{z} = \frac{-0.21111}{(z - 0.62679)(z - 1.59543)}$$

$$= \frac{0.21794}{(z - 0.62679)} - \frac{0.21794}{(z - 1.59543)}$$

Taking the inverse z transform gives the impulse response:

$$h(n) = 0.21794[(0.62679)^n u(n) + (1.59543)^n u(-n - 1)]$$

$$= 0.21794(0.62679)^{|n|}$$

This function is plotted in Fig. 8.5.1. A system with this impulse response is physically unrealizable, meaning the output occurs before the input is applied.

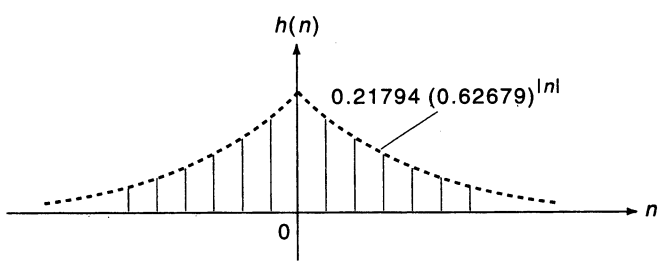


Fig. 8.5.1

The Optimum Physically Realizable Filter

Now it is time to derive the optimum realizable filter for mean-square estimation. One thought that comes to mind upon looking at Fig. 8.5.1 is why not discard the anticipatory part of the impulse response and keep the realizable part? Would that be optimum? If so, we can derive the optimum filter by first assuming that the signal is present for all time and deriving the optimum nonrealizable filter. Then we simply discard the part that gives us trouble and keep the rest, and hope this is optimum. In order to find out how well this works, we first need to go through some complicated math (the yellow peril) and compare.

First, here is what we are aiming at. We break the optimum filter $h(n)$ into two parts, $h_1(n)$ and $h_2(n)$, as shown in Fig. 8.5.2. Now we want to show that if $h_1(n)$ is the whitening filter from Section 8.4,

$$h_1(n) \leftrightarrow H_c^{-1}(z) \tag{8.5.6}$$

and if $h_2(n)$ is given by

$$h_2(n) \leftrightarrow H_2(z) = \frac{1}{k}\left[\frac{G_{DX}(z)}{H_c(z^{-1})}\right]_+ \tag{8.5.7}$$

then the optimum filter is $h(n) = h_1(n) * h_2(n)$, as shown in Fig. 8.5.2.

Recall that we use the double-headed arrow to indicate transforms (in this case, the z transform), and that the power spectral density for the data $x(n)$ is given by

$$G_{XX}(z) = KH_c(z)H_c(z^{-1}) \tag{8.5.8}$$

(see Section 8.4). The filter $H_c(z)$ is the physically realizable minimum-phase component of G_{XX}, K is a constant related to the power in the signal, and $H_c(z^{-1})$ is the maximum-phase component of the spectrum. The subscript outside the right bracket in Eq. 8.5.7 means that we will use the physically realizable part of this expression and discard the anticipatory part.

The idea behind the derivation to follow is that if $h_2(n)$ is chosen to make the error orthogonal to its input data $v(n)$, and if $h_1(n)$ is the whitening filter $H_c^{-1}(z)$, then the overall filter $h(n)$ will make the error orthogonal to the input data $x(n)$, meaning that $h(n)$ is optimum. Notice that the $v(n)$ is the innovations process for the input $x(n)$. Wiener used the term innovations because

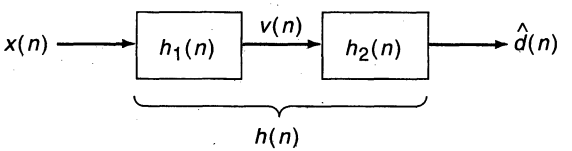


Fig. 8.5.2. The optimum filter $h(n)$ broken into the cascade of two systems, $h_1(n)$ and $h_2(n)$.

$v(n)$ contains the "new information" in $x(n)$. The signal can be split into two parts, the predictable part and the unpredictable part. (This is called the *Wold decomposition.*) The predictable part is that which can be predicted with complete accuracy based on the past history of $x(n)$. The innovations process contains the difference between the actual value of $x(n + 1)$ and the predicted value $v(n + 1)$. But before getting into the details, let us work an example to make certain we understand what each of the terms in Eqs. 8.5.6 and 8.5.7 stand for.

EXAMPLE 8.5.2. Find the optimum physically realizable filter if

$$x(n) = s(n) + w(n) \qquad d(n) = s(n)$$
$$R_{SS}(l) = 0.9^{|l|} \qquad R_{WW}(l) = \delta(l)$$

where the noise is zero-mean and independent of the signal, as in Example 8.5.1.

SOLUTION: We need to find $G_{DX}(z)$ and $G_{XX}(z)$ and identify the terms K, $H_c(z)$, and $H_c(z^{-1})$ to substitute into Eqs. 8.5.6 and 8.5.7.

$$R_{DX}(l) = E[d(n)x(n - l)] = E\{s(n)[s(n - l) + w(n - l)]\} = R_{SS}(l)$$

since the noise and signal are independent with zero mean. Thus

$$G_{DX}(z) = G_{SS}(z) = \frac{1 - 0.9^2}{(1 - 0.9z^{-1})(1 - 0.9z)}$$

We use the same reasoning to find R_{XX}.

$$R_{XX}(l) = R_{SS}(l) + R_{WW}(l)$$

so

$$G_{XX}(l) = G_{SS}(l) + G_{WW}(l) = \frac{0.19}{(1 - 0.9z^{-1})(1 - 0.9z)} + 1$$
$$= 1.43589 \frac{(1 - 0.62679z^{-1})(1 - 0.62679z)}{(1 - 0.9z^{-1})(1 - 0.9z)}$$

From this we can identify the parameters in Eqs. 8.5.6 and 8.5.7 as

$$K = 1.43589$$
$$H_c(z) = \frac{(1 - 0.62679z^{-1})}{(1 - 0.9z^{-1})}$$
$$H_a(z) = H_c(z^{-1}) = \frac{(1 - 0.62679z)}{(1 - 0.9z)}$$

Therefore $H_2(z)$ in Eq. 8.5.7 is given by

$$H_2(z) = \frac{1}{K}\left[\frac{G_{DX}(z)}{H_c(z^{-1})}\right]_+ = 0.69643\left[\frac{0.19}{(1-0.9z^{-1})(1-0.62679z)}\right]_+$$

Expanding in partial fractions to separate the causal part gives

$$\frac{0.13232}{(1-0.9z^{-1})(1-0.62679z)} = \frac{0.30357}{1-0.9z^{-1}} + \frac{B}{1-0.62679z}$$

so

$$H_2(z) = \frac{0.30357}{1-0.9z^{-1}}$$

Multiplying by $H_1 = H_c^{-1}$ gives the final answer:

$$H(z) = \left(\frac{1-0.9z^{-1}}{1-0.62679z^{-1}}\right)\left(\frac{0.30357}{1-0.9z^{-1}}\right) = \frac{0.30357}{1-0.62679z^{-1}}$$

Hence

$$h(n) = 0.30357(0.62679)^n u(n)$$

This filter is shown in Fig. 8.5.3. Notice that the only difference between the realizable part of the optimum filter in Example 8.5.1 and this filter is in the gain. The realizable part of that filter had a gain of 0.21794, while here the gain is 0.30357. Is this important? Well, it can be. If you are trying to estimate $d(n)$, it is all-important. If you are trying to separate signal from noise, however, as in an ordinary AM radio, the gain is adjustable by the operator. It therefore assumes little importance. In presenting this theory we can only assume that the gain is very important, for we must be prepared to estimate the desired quantity $d(n)$ as accurately as possible.

Now to derive the optimum filter. We want to show that when the input signal $x(n)$ has a spectrum that can be decomposed into the components K and H_c according to Eq. 8.5.8,

$$G_{XX}(z) = KH_c(z)H_a(z) \qquad \text{(repeated) (8.5.8)}$$

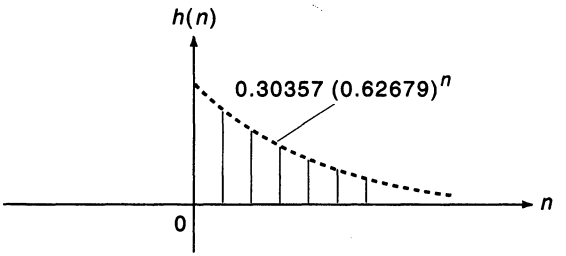


Fig. 8.5.3. The physically realizable optimum filter.

where $H_a(z) = H_c(z^{-1})$, then the optimum filter is given in Fig. 8.5.2, where h_1 and h_2 are given in Eqs. 8.5.6 and 8.5.7.

$$h_1(n) \leftrightarrow H_c^{-1}(z) \qquad \text{(repeated) (8.5.6)}$$

$$h_2(n) \leftrightarrow H_2(z) = \frac{1}{k}\left[\frac{G_{DX}(z)}{H_c(z^{-1})}\right]_+ \qquad \text{(repeated) (8.5.7)}$$

We start with the estimate $\hat{d}(n)$ given by

$$\hat{d}(n) = \sum_{k=-\infty}^{n} h(n-k)x(k) \tag{8.5.9}$$

Notice that the upper limit on this summation is n. This is because the data $x(k)$ are available for the infinite past, but none of the future values of $x(k)$ are available. This is the only difference between the assumptions here and those for the anticipatory system in Eq. 8.5.3. The error is

$$e(n) = d(n) - \hat{d}(n) \tag{8.5.10}$$

and the data are $x(n-i)$ for $i = 0, 1, 2, \ldots$. We set the error orthogonal to the data to obtain

$$E[e(n)x(n-i)] = E\left\{\left[d(n) - \sum_{k=-\infty}^{n} h(n-k)x(k)\right]x(n-i)\right\} = 0$$

or

$$R_{DX}(i) = \sum_{k=-\infty}^{n} h(n-k)R_{XX}(k-n+i), \qquad i = 0, 1, 2, \ldots$$

We can simplify this expression if we let $l = n - k$.

$$R_{DX}(i) = \sum_{l=0}^{\infty} h(l)R_{XX}(i-l), \qquad i = 0, 1, 2, \ldots \tag{8.5.11}$$

This is the Wiener–Hopf equation. It would be the convolution of $h(l)$ with $R_{XX}(l)$ except for the troublesome restriction on the values of i. We now wish to show that $h(n)$ in Fig. 8.5.2 satisfies this equation.

Let us first consider $h_2(n)$ with white-noise input $v(n)$, where h_2 is specified by Eq. 8.5.7. From the Wiener–Hopf equation with $x = v$,

$$R_{DV}(i) = \sum_{l=0}^{\infty} R_{VV}(i-l)h_2(l), \qquad i = 0, 1, 2, \ldots$$

but since v is white noise with variance σ_V^2, we have

$$R_{VV}(l) = \sigma_V^2 \delta(l)$$

so

$$h_2(l) = \frac{1}{\sigma_v^2} R_{DV}(l), \qquad l \geq 0$$

so now we need to find $R_{DV}(l)$. We derived Eq. 8.4.6 in the previous section specifically for this occasion. That formula gave the relationship as

$$R_{DV}(l) = h_1(-l) * R_{DX}(l) \qquad \text{(repeated) (8.4.6)}$$

But $h_1(-l)$ is the time-reversed or anticausal version of $h_1(l)$, so its transform is $H_1(z^{-1})$. This means that the transforms of the terms in Eq. 8.4.6 are

$$G_{DV}(z) = G_{DX}(z)H_1(z^{-1})$$

Since $H_1(z) = H_c^{-1}(z) = 1/H_c(z)$, then $H_1(z^{-1}) = 1/H_c(z^{-1})$. This gives

$$G_{DV}(z) = \frac{G_{DX}(z)}{H_c(z^{-1})}$$

since $h_2(l) = (1/\sigma_v^2)R_{DV}(l)$, $l \geq 0$,

$$H_2(z) = \frac{1}{K}[G_{DV}(z)]_+ = \frac{1}{K}\left[\frac{G_{DX}(z)}{H_c(z^{-1})}\right]_+$$

where $K = \sigma_v^2$ and the $+$ is necessary because of the condition $l \geq 0$ on h_2.

This completes our derivation of Eq. 8.6.7. Now we choose for h_1 the innovations filter h_c^{-1} and show that this, combined with h_2, makes the overall filter optimum, i.e., it makes the error orthogonal to the data. Since $v(n) = h_1(n) * x(n)$,

$$x(n) = v(n) * h_1^{-1}(n) = \sum_{k=-\infty}^{\infty} h_1^{-1}(n-k)\, v(k)$$

Multiply this by the error and take the inner product:

$$\begin{aligned}\langle x|e\rangle &= E[x(n-i)e(n)] \\ &= \sum_{k=-\infty}^{\infty} h_1^{-1}(n-i-k)\, E[v(k)\, e(n)], \qquad i = 0, 1, 2, \ldots\end{aligned}$$

But $v(n)$ is orthogonal to the error because we chose h_2 to make it so. Therefore it follows that $x(n)$ is orthogonal to $e(n)$, meaning that $h(n) = h_1(n) * h_2(n)$ is optimal.

The mean-square error resulting from the filter that satisfies this equation is found by the inner product of the error with $d(n)$:

$$E[e^2] = R_{DD}(0) - \sum_{l=0}^{\infty} h(l)\, R_{DX}(l) \qquad (8.5.12)$$

EXAMPLE 8.5.3. Find the mean square error in Example 8.5.2.

SOLUTION

$$R_{DD}(0) = R_{SS}(0) = 1$$
$$R_{DX}(l) = R_{SX}(l) = R_{SS}(l) = 0.9^l \qquad l \geq 0$$

Therefore

$$E(e^2) = 1 - \sum_{l=0}^{\infty} h(l)\, R_{SS}(l) = 1 - \sum_{l=0}^{\infty} 0.30357(0.62679)^l(0.9)^l$$
$$= 1 - 0.30357 \sum_{l=0}^{\infty} (0.56411)^l$$

From Eq. 5.1.6 we get

$$E(e^2) = 1 - 0.30357(2.29416) = 0.30357$$

Review

In order to find the filter derived in this section we need to know the input spectrum $G_{XX}(z)$ and the cross spectrum between the input and the desired signal $G_{DX}(z)$. Given these quantities (or their transforms), we first decompose $G_{XX}(z)$ into $H_c(z)$ and $H_a(z) = H_c(z^{-1})$ by

$$G_{XX}(z) = KH_c(z)H_c(z^{-1}) \qquad \text{(repeated) (8.5.8)}$$

then the optimum physically realizable filter is the cascade of $h_1(n)$ and $h_2(n)$ given by

$$h_1(n) \leftrightarrow H_c^{-1}(z) \qquad \text{(repeated) (8.5.6)}$$

$$h_2(n) \leftrightarrow H_2(z) = \frac{1}{k}\left[\frac{G_{DX}(z)}{H_c(z^{-1})}\right]_+ \qquad \text{(repeated) (8.5.7)}$$

8.6 Recursive Filtering

Preview

The finite-impulse response filter in Section 8.3 is optimum for a fixed-length signal. That is, if the input signal is of length p, then a filter of length p with appropriate coefficients $h_0, h_1, \ldots, h_p$ gives the optimum estimate of $d(n)$ at its output. But in real life, p increases as time increases. For example, in a radar tracking system the input signal has a beginning, and the number of terms in the input signal increases with time. Neither the fixed-length FIR filter nor the IIR filter (which also has fixed length) is appropriate for this situation. What we need is a different FIR filter at each time instant, the difference being that one more filter tap is added for each new input signal term. That is what recursive filtering accomplishes.

In this section we will consider only the simplest recursive filtering problem, which is for a first-order signal with additive white noise. We do this to illustrate the essential features of recursive filtering without too many complications. The extension of these concepts to more realistic applications should not be too difficult when you need to do so.

Here we illustrate the features of recursive filtering in the following way. Figure 8.6.1 shows a sequence of optimum filters. The minimum mean-square-error linear estimate for $d(n)$ when only one value of $x(n)$ is available for use is $\hat{d}(n) = h_0^0 x(n)$. If two data values $x(n)$ and $x(n-1)$ are available, then the optimum estimate uses a linear combination given by $\hat{d}(n) = h_0^1 x(n) + h_1^1 x(n-1)$. In general, when the present value plus p past values of the data are available, the optimum estimate is

$$\hat{d}(n) = h_0^p x(n) + h_1^p x(n-1) + \cdots + h_p^p x(n-p) \qquad (8.6.1)$$

The superscript p indicates simply that the filter tap values vary from one filter to another. That is, $h_i^p \neq h_i^{p+1}$ for every i. Of course, in all this we are assuming proper choice for the coefficients in each filter.

As the order of the filter increases, meaning the estimate is based on more data, the error should decrease. Here is an example to illustrate these ideas.

EXAMPLE 8.6.1. Suppose we are presented with signal plus zero-mean independent noise and we wish to estimate the signal.

$$x(n) = s(n) + w(n)$$
$$d(n) = s(n)$$

where

$$R_{ss}(l) = 0.9^{|l|} \quad \text{and} \quad R_{ww}(l) = \delta(l)$$

(See Examples 8.3.3 and 8.3.4.) Find the optimum filter and the mean-square error for each case in Fig. 8.6.1.

SOLUTION: From Examples 8.3.3 and 8.3.4,

$$h_0^0 = 0.5 \qquad E[e_0^2] = 0.5$$
$$h_0^1 = 0.3730 \qquad h_1^1 = 0.2821 \qquad E[e_1^2] = 0.3730$$

For three data values,

$$\hat{d}(n) = h_0^2 x(n) + h_1^2 x(n-1) + h_2^2 x(n-2)$$

and the error is

$$e_2(n) = d(n) - h_0^2 x(n) - h_1^2 x(n-1) - h_2^2 x(n-2)$$

Setting the error $e_2(n)$ orthogonal to the data, which are $x(n)$, $x(n-1)$,

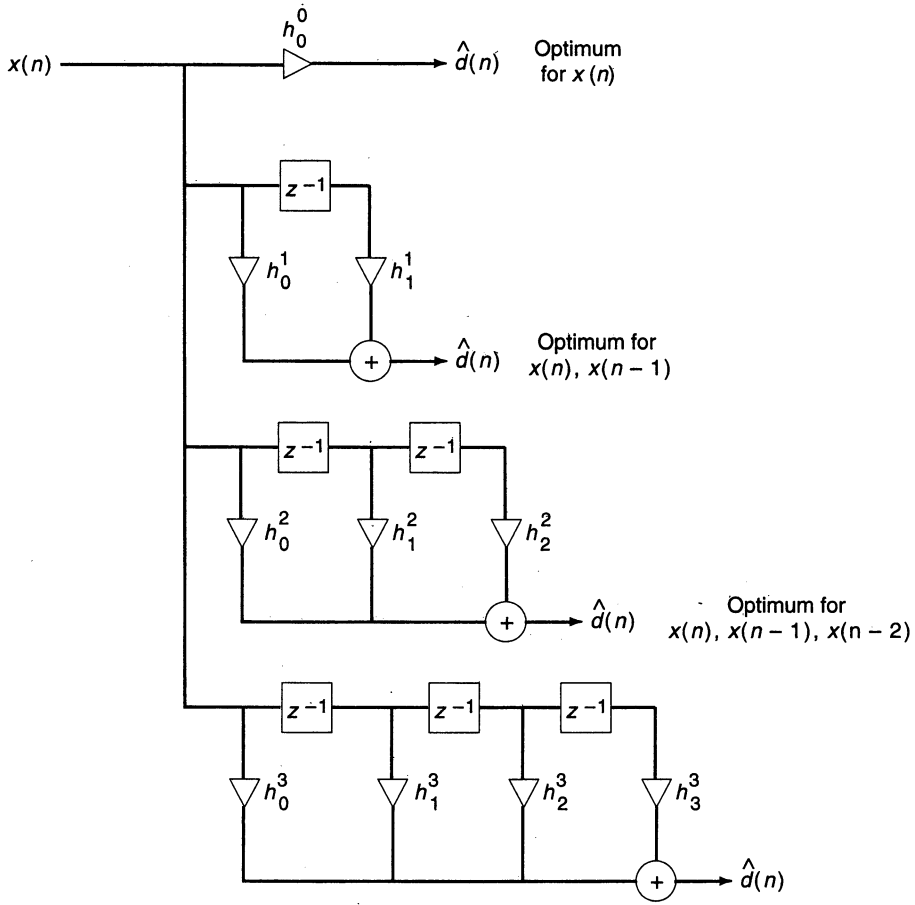


Fig. 8.6.1. Optimum filters for data of length exactly 1, 2, 3, or 4.

and $x(n-2)$, gives three equations in three unknowns:

$$\begin{bmatrix} R_{XX}(0) & R_{XX}(-1) & R_{XX}(-2) \\ R_{XX}(1) & R_{XX}(0) & R_{XX}(-1) \\ R_{XX}(2) & R_{XX}(1) & R_{XX}(0) \end{bmatrix} \begin{bmatrix} h_0^2 \\ h_1^2 \\ h_2^2 \end{bmatrix} = \begin{bmatrix} R_{DX}(0) \\ R_{DX}(1) \\ R_{DX}(2) \end{bmatrix}$$

or

$$\begin{bmatrix} 2 & 0.9 & 0.81 \\ 0.9 & 2 & 0.9 \\ 0.81 & 0.9 & 2 \end{bmatrix} \begin{bmatrix} h_0^2 \\ h_1^2 \\ h_2^2 \end{bmatrix} = \begin{bmatrix} 1.0 \\ 0.9 \\ 0.81 \end{bmatrix}$$

This gives

$$h_0^2 = 0.3298 \qquad h_1^2 = 0.2250 \qquad h_2^2 = 0.1702 \qquad E[e_2^2] = 0.3298$$

When we repeat this procedure for four observations, we get

$$h_0^3 = 0.3137 \qquad h_1^3 = 0.2037 \qquad h_2^3 = 0.1389$$
$$h_3^3 = 0.1051 \qquad E[e_3^2] = 0.3137$$

The results of this example plus the h^4 coefficients, along with the associated mean-square error, are displayed in Table 8.6.1. Notice that the mean-square error (mse) decreases as we use more data (longer filters) in our estimates. This is called *block processing*, because successive blocks of data are processed by successively larger systems. As each new data value is received, the processor must be redesigned with new coefficients h_i^p and one new delay added to the system. This begs the question of whether we can design an equivalent recursive filter, one with time-varying parameters, that produces the same estimate at each step. If so, it would need to be of the form

$$\hat{d}(n) = A_n \hat{d}(n-1) + K_n x(n) \tag{8.6.2}$$

where A_n and K_n are the time-varying coefficients to be determined. For this to be equivalent to Eq. 8.6.1, we start at $n = 0$ and set

$$\hat{d}(0) = h_0^0 x(0)$$

Then successive steps give

$$\begin{aligned}\hat{d}(1) &= A_1 \hat{d}(0) + K_1 x(1) \\ &= A_1[h_0^0 x(0)] + K_1 x(1) \\ \hat{d}(2) &= A_2 \hat{d}(1) + K_2 x(2) \\ &= A_2[A_1 h_0^0 x(0) + K_1 x(1)] + K_2 x(2) \\ &\quad \text{etc.}\end{aligned}$$

If we know the block-processing parameters h_i^p, we can use these equations to solve for the A_i and K_i parameters. But this is not what we want. We need to be able to find the A_i's and K_i's directly.

Table 8.6.1

	h_0	h_1	h_2	h_3	h_4	*mse*
h^0	0.5					0.5
h^1	0.373	0.2821				0.373
h^2	0.3298	0.2250	0.1702			0.3298
h^3	0.3137	0.2037	0.1389	0.1051		0.3137
h^4	0.3075	0.1955	0.1269	0.0866	0.0655	0.3075

The approach taken by R. E. Kalman in 1960 began with a general model for the signal based on the same idea as the innovations representation. If the signal could be generated by a system with known characteristics, this would allow him to find the optimum recursive estimates. These estimates turn out to be related to the parameters of the system that generated the signal in the first place. For instance, the signal in Example 8.6.1 has an exponential correlation function, and we know that the innovations representation for this signal is a first-order system described by any of the three descriptions,

$$\begin{aligned} &\text{Impulse response:} && h(n) = \alpha^n u(n) \\ &\text{Transfer function:} && H(z) = \frac{1}{1 - \alpha z^{-1}} \\ &\text{Difference equation:} && s(n) = \alpha\, s(n-1) + \eta(n) \end{aligned} \tag{8.6.3}$$

where the difference equation description has white-noise input $\eta(n)$ and output signal $s(n)$. The observations are in the form

$$x(n) = s(n) + w(n) \tag{8.6.4}$$

The system diagram in Fig. 8.6.2 combines these last two equations to give the signal model. The system driven by white noise $\eta(n)$ produces the signal $s(n)$. The noise $w(n)$, which is independent of the driving force noise $\eta(n)$, is added to $s(n)$ to produce the observation $x(n)$. [We are using two different noise terms here. The $\eta(n)$ noise is part of the innovations model. The $w(n)$ noise is our usual additive noise.]

The mean-square linear recursive estimate for $s(n)$ is given by Eq. 8.6.2. As we will show later, A_n is given in terms of the parameter α by

$$A_n = (1 - K_n)\alpha \tag{8.6.5}$$

giving

$$\hat{d}(n) = \alpha\, \hat{d}(n-1) + K_n[x(n) - \alpha\, \hat{d}(n-1)] \tag{8.6.6}$$

This allows us to express the recursive estimate as the sum of two terms given by

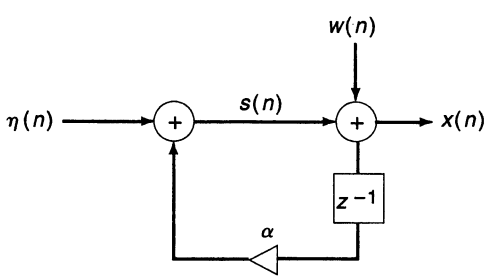


Fig. 8.6.2. The signal model.

Term 1: $\hat{d}_1(n) = \alpha\,\hat{d}(n-1)$ The forward prediction term

Term 2: $\hat{d}_2(n) = K_n[x(n) - \alpha\,\hat{d}(n-1)]$ The residual or correction term

where α is the parameter in Eq. 8.6.3, and K_n is a time-varying gain to be determined. We can derive a system diagram from this equation, just as we derived Fig. 8.6.2 from Eq. 8.6.3. This is the recursive filter shown in Fig. 8.6.3. For proper choice of K_n and initial estimate $\hat{s}(0)$, it will produce the same estimates as the sequence of block-processing units in Fig. 8.6.1.

The way to derive these proper values of K_n is to set the error orthogonal to the data and solve the resulting equations. That is a rather involved procedure, which we will get to shortly. In the meantime, let us present the procedure for evaluating the parameters and do an example.

The changing mean-square error determines the time-varying gain K_n as follows: Label this mean-square error as $\varepsilon(n)$, where

$$\varepsilon(n) = E[e^2(n)] = E\{[d(n) - A_n\,\hat{d}(n-1) - K_n x(n)]^2\} \quad (8.6.7a)$$

Since the estimate we are using is linear, this is also given by the inner product of the error with $d(n)$,

$$\varepsilon(n) = E[e(n)d(n)] \quad (8.6.7b)$$

Then K_n is given in terms of $\varepsilon(n)$ by

$$K_n = \frac{\varepsilon(n)}{\sigma_W^2} \quad (8.6.8)$$

You may notice we have traded one problem for another. Since K_n depends on $\varepsilon(n)$ in Eq. 8.6.8, and $\varepsilon(n)$ depends on K_n from Eq. 8.6.7a, it seems we are going around in circles. And we are, but we can escape from this loop by the fact that $\varepsilon(n)$ depends on the previous mean-square error $\varepsilon(n-1)$.

$$\varepsilon(n) = \left[\frac{\sigma_\eta^2 + \alpha^2\varepsilon(n-1)}{\sigma_\eta^2 + \sigma_W^2 + \alpha^2\varepsilon(n-1)}\right]\sigma_W^2 \quad (8.6.9)$$

with initial value given by

$$\varepsilon(0) = \frac{\sigma_S^2\sigma_W^2}{\sigma_S^2 + \sigma_W^2} \quad (8.6.10)$$

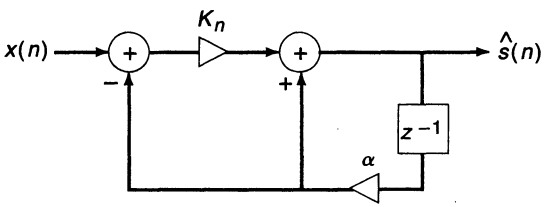


Fig. 8.6.3. The recursive filter. For proper choice of K_n and initial estimate $\hat{s}(0)$ it is exactly equivalent to the sequence of block processors in Fig. 8.6.1.

We will derive these equations later.

EXAMPLE 8.6.2. For the conditions in Example 8.6.1, calculate the first three Kalman estimates.

SOLUTION: The initial error term is given by

$$\varepsilon(0) = \frac{\sigma_S^2 \sigma_W^2}{\sigma_S^2 + \sigma_W^2} = \frac{1 \cdot 1}{1 + 1} = 0.5$$

Equation 8.6.8 gives

$$K_0 = \frac{0.5}{1} = 0.5$$

so

$$\hat{s}(0) = K_0\, x(0) = 0.5\, x(0)$$

For $n = 1$, the variance σ_η^2 is related to the variance σ_S^2 of the signal by

$$\sigma_\eta^2 = (1 - \alpha^2)\sigma_S^2 = (1 - \alpha^2)R_{SS}(0) = 1 - 0.81 = 0.19$$

Plugging this into Eq. 8.6.9 with $n = 1$ gives

$$\varepsilon(n) = \left[\frac{\sigma_\eta^2 + \alpha^2 \varepsilon(n-1)}{\sigma_\eta^2 + \sigma_W^2 + \alpha^2 \varepsilon(n-1)}\right] \sigma_W^2$$

$$= \left[\frac{0.19 + 0.81(0.5)}{0.19 + 1 + 0.81(0.5)}\right] = 0.3730$$

Then, from Eq. 8.6.8, $K_1 = 0.3730$. Therefore,

$$\hat{s}(1) = \alpha\hat{s}(0) + K_1[x(1) - \alpha\hat{s}(0)] = 0.373\, x(1) + 0.2821\, x(0)$$

which agrees with the coefficients in Example 8.6.1.

For $n = 2$,

$$\varepsilon(2) = \left[\frac{0.19 + 0.81(0.373)}{0.19 + 1 + 0.81(0.373)}\right] = 0.3298$$

$$K_2 = \frac{\varepsilon(2)}{\sigma_W^2} = 0.3298$$

$$\hat{s}(2) = \alpha\hat{s}(1) + K_2[x(2) - \alpha\hat{s}(1)]$$

$$= 0.3298\, x(2) + 0.225\, x(1) + 0.1702\, x(0)$$

which agrees again with the values in Table 8.6.1. As you can see, these recursive estimates are equal to those obtained by block processing.

Derivation of the Kalman Filter

We wish to show now that the recursion formulas 8.6.6, 8.6.8, and 8.6.9 can be derived by setting the error orthogonal to the data. The data are

$$x(n) = s(n) + w(n) \qquad n = 0, 1, 2, \ldots$$

The estimate is

$$\hat{s}(n) = A_n\hat{s}(n-1) + K_n x(n)$$

and the error is

$$\varepsilon(n) = s(n) - \hat{s}(n)$$

We break the following analysis into two parts, $l = 0$ and $l > 0$.

$l = 0$. First, for $l = 0$, setting the error orthogonal to the data gives

$$\begin{aligned} E[e(n)x(n)] &= E\{e(n)[s(n) + w(n)]\} \\ &= E[e(n)s(n)] + E[e(n)w(n)] = 0 \end{aligned}$$

The first term is $\varepsilon(n)$, the mean-square error, because this term is the inner product of the error with the term to be estimated. Substituting this term and expanding the second term gives

$$\begin{aligned} \varepsilon(n) + E[e(n)w(n)] &= \varepsilon(n) + E\{[s(n) - A_n\hat{s}(n-1) - K_n x(n)]w(n)\} \\ &= \varepsilon(n) + E[s(n)w(n)] - A_n E[\hat{s}(n-1)w(n)] \\ &\quad - K_n E[x(n)w(n)] = 0 \end{aligned}$$

Since the noise is uncorrelated with $s(n)$, and also uncorrelated with past values of $x(n)$, the second and third terms are zero. This gives

$$\varepsilon(n) - K_n\sigma_W^2 = 0$$

or

$$K_n = \frac{\varepsilon(n)}{\sigma_W^2} \qquad \text{(repeated) (8.6.8)}$$

This gives a formula for calculating K_n if we can calculate $\varepsilon(n)$. This is provided by the second part of the derivation, for $l > 0$.

$l > 0$. Now when we set the error orthogonal to the data we get

$$\begin{aligned} E[e(n)x(n-l)] &= E\{[s(n) - A_n\hat{s}(n-1) - K_n x(n)]x(n-l)\} \\ &= E[s(n)x(n-l)] - A_n E[\hat{s}(n-1)x(n-l) \\ &\quad - K_n E[x(n)x(n-l)] = 0 \end{aligned}$$

The first term is $R_{ss}(l)$. So is the last term, though it may not be so obvious. The last term is $R_{xx}(l) = R_{ss}(l) + R_{ww}(l)$, but for $l > 0$ we have $R_{ww}(l) = 0$ [because $w(n)$ is white noise]. This gives

$$(1 - K_n)R_{ss}(l) - A_n\{[s(n-1) - e(n-1)]x(n-l)\} = 0$$

But $E[e(n-1)x(n-l)] = 0$ for $l > 0$, giving

$$\begin{aligned}(1 - K_n)R_{ss}(l) - A_n[s(n-1)x(n-l)] \\ = (1 - K_n)R_{ss}(l) - A_nR_{ss}(l-1) \\ = 0\end{aligned}$$

or

$$R_{ss}(l) = \frac{A_n}{1 - K_n} R_{ss}(l-1) \tag{8.6.11}$$

This first-order homogeneous difference equation has the solution

$$R_{ss}(l) = R_{ss}(0)\alpha^l, \qquad l > 0 \tag{8.6.12}$$

where

$$\alpha = \frac{A_n}{1 - K_n} \tag{8.6.13}$$

which gives us Eq. 8.6.5.

We have succeeded in deriving Eqs. 8.6.5 and 8.6.6. Now we need to show that the recursive error equation 8.6.9 is valid. The mean-square error at each step is the inner product of the error with the quantity we are estimating.

$$\begin{aligned}\varepsilon(n) = \langle s(n)|e(n)\rangle = E\{s(n)[s(n) - \alpha(1 - K_n)\hat{s}(n-1) - K_nx(n)]\} \\ = E[s^2(n)] - \alpha(1 - K_n)E[s(n)\hat{s}(n-1)] - K_nE[s(n)x(n)]\end{aligned}$$

The first and last expected values are $R_{ss}(0) = \sigma_s^2$. This gives

$$\begin{aligned}\varepsilon(n) = \sigma_S^2(1 - K_n) - \alpha(1 - K_n)E[s(n)\hat{s}(n-1)] \\ = (1 - K_n)(\sigma_S^2 - \alpha E\{[\alpha s(n-1) + \eta(n)]\hat{s}(n-1)\})\end{aligned}$$

The input noise $\eta(n)$ is uncorrelated, implying that it is uncorrelated with past values of the signal. Therefore,

$$\varepsilon(n) = (1 - K_n)[\sigma_S^2 - \alpha^2 E\{s(n-1)\hat{s}(n-1)\}] \tag{8.6.14}$$

From this we may now derive Eqs. 8.6.9 and 8.6.10. Initially, with $n = 0$, if we assume that there is no prior estimate $\hat{s}(n-1)$, we get

$$\varepsilon(n) = (1 - K_n)\sigma_S^2$$

Substituting $K_n = \varepsilon(n)/\sigma_W^2$ from Eq. 8.6.8 and performing a few manipulations gives Eq. 8.6.10.

For $n > 0$ we can use Eq. 8.6.14 to derive Eq. 8.6.9, but first let us display an expression for $\varepsilon(n-1)$:

$$\begin{aligned}\varepsilon(n-1) &= E[s(n-1)e(n-1)] = E\{s(n-1)[s(n-1)-\hat{s}(n-1)]\} \\ &= \sigma_S^2 - E[s(n-1)\hat{s}(n-1)]\end{aligned}$$

Solving for $E[s(n-1)\hat{s}(n-1)]$ and substituting into Eq. 8.6.14 gives

$$\varepsilon(n) = (1-K_n)\{\sigma_S^2 - \alpha^2[\sigma_S^2 - \varepsilon(n-1)]\} \tag{8.6.15}$$

This provides a recursive relationship between $\varepsilon(n)$ and $\varepsilon(n-1)$. Recall that $\sigma_\eta^2 = (1-\alpha^2)\sigma_S^2$ and that $K_n = \varepsilon(n)/\sigma_W^2$. Substituting these two quantities into Eq. 8.6.15 gives Eq. 8.6.9.

Review

In order to use the Kalman filter derived in this section, we need to know:

1. The signal has exponential autocorrelation function, meaning that a signal with identical second-order statistics can be generated by the system in Fig. 8.6.2. The parameters α and σ_η^2 must be known.
2. The additive noise $w(n)$ is white with known variance σ_W^2.

Then we perform the following steps.

Step 1: Set $n = 0$ and calculate the initial mean-square error:

$$\varepsilon(0) = \frac{\sigma_S^2\sigma_W^2}{\sigma_S^2 + \sigma_W^2}$$

Step 2: Calculate $K_n = \varepsilon(n)/\sigma_W^2$.

Step 3: Input the data $x(n)$ and calculate the estimate:

$$\hat{s}(n) = \alpha\hat{s}(n-1) + K_n[x(n) - \alpha\hat{s}(n-1)]$$

[For $n = 0$ assume $\hat{s}(n-1) = 0$, so $\hat{s}(0) = K_0x(0)$.]

Step 4: Let $n = n + 1$.

Step 5: Update the error:

$$\varepsilon(n) = \left[\frac{\sigma_\eta^2 + \alpha^2\varepsilon(n-1)}{\sigma_\eta^2 + \sigma_W^2 + \alpha^2\varepsilon(n-1)}\right]\sigma_W^2$$

where $\sigma_\eta^2 = (1-\alpha^2)\sigma_S^2$.

Step 6: Go to step 2.

8.7 Problems

8.1. Let X and Y be random variables defined on the $\{H, T\}$ sample space. Define vector addition and scalar multiplication by the rules

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 + y_1 \\ x_2 + y_2 \end{bmatrix} \qquad a \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} ax_1 \\ 0 \end{bmatrix}$$

Is this a vector space? Show why or why not.

8.2. Let A be the ternary field as in Problem 2.8b. Let $X = \{x, y, z\}$. Fill in the blanks in the following table for vector addition to satisfy the properties listed in Definition 8.2.1.

+	x	y	z
x	x	y	z
y	y		
z	z		

Example: $x + z = z$.

Also fill in the following table to define scalar multiplication.

A \ X	x	y	z
0			
1			
2			

8.3. Two random variables X and Y are defined on the sample space given in Fig. 8.7.1. There are three possible experimental outcomes ζ_1, ζ_2, and ζ_3, and they occur with equal probability.

(a) Find the mean-square estimate of X and the resulting mean-square error.

(b) Find the mean-square estimate of X given Y and the resulting mean-square error.

X	−2	0	1	
Y	1	0	1	
ζ	ζ_1	ζ_2	ζ_3	

Fig. 8.7.1

8.4. Repeat Problem 8.3 for the probabilities

$$P(\zeta_1) = P(\zeta_2) = \tfrac{1}{4} \qquad P(\zeta_3) = \tfrac{1}{2}$$

8.5. Let X be a random variable that assumes the values -2, -1, 0, 1, 2 with equal probability. Let $Y = X + N$, where N is another random variable, statistically independent from X, that assumes any of the values -1, 0, 1 with equal probability. Find the linear estimate of X given Y and the resulting mean-square error.

8.6. Let X and N be as in Problem 8.5, but now we perform the experiment as follows: A number is selected for X and recorded. A number N_1 is chosen and added to X to form $Y_1 = X + N_1$. A second number N_2, independent of N_1, is chosen and added to the same X to form $Y_2 = X + N_2$. If you are shown the two numbers Y_1 and Y_2, how can you estimate X with a linear estimator? Also find the resulting mean-square error and see that it is smaller than in Problem 8.5.

8.7. Repeat Problem 8.6 with three observations Y_1, Y_2, and Y_3.

8.8. Let N be a Gaussian random variable with zero mean and unit variance. Repeat Problem 8.5.

8.9. A continuous-time, stationary stochastic process $X(t)$ with mean 0 has correlation

$$R_{XX}(\tau) = \frac{\sin(\pi\tau)}{\pi\tau}$$

(a) One sample is taken at time t. It is $x(t) = 1.5$. Find the linear estimate of $x(t + \frac{1}{2})$ and the resulting mean-square error.
(b) Now suppose that you are given two values of x: $x(t) = 1.5$ and $x(t - \frac{1}{2}) = -0.5$. Repeat part (a).

8.10. Which of the following discrete-time systems are minimum-phase systems?

(a) $H_1(z) = \dfrac{z^2 + 1}{z^2 + 0.6z - 0.16}$

(b) $H_2(z) = \dfrac{z(z - 0.5)}{z^2 + 0.6z - 0.16}$

(c) $H_3(z) = \dfrac{z^2 + 1.7z + 0.6}{z^2 + 0.6z - 0.16}$

8.11. For each minimum-phase system in Problem 8.10, do the following:
(a) Find the impulse response of the system.
(b) Find the inverse system and its impulse response.
(c) Convolve the two impulse responses and show that the result is a delta function.

8.12. Factor the following complex spectral density functions into minimum- and maximum-phase components.

(a) $G_1(z) = \dfrac{-2.5}{z - 2.9 + z^{-1}}$

(b) $G_2(z) = \dfrac{z - 2.5 + z^{-1}}{z - 2.05 + z^{-1}}$

8.13. Find the innovations representation for each process in Problem 8.12.

8.14. Factor the following complex spectral density functions into minimum- and maximum-phase components.

(a) $G_1(z) = \dfrac{-4}{z - 4.25 + z^{-1}}$

(b) $G_2(z) = \dfrac{-5z^{-2}}{1 - 2.5z^{-2} + z^{-4}}$

8.15. Find the innovations representation for each process in Problem 8.14.

8.16. Suppose that we are presented with signal plus zero-mean independent noise and we wish to estimate the signal.

$$x(n) = s(n) + w(n)$$
$$d(n) = s(n)$$

where

$$R_{ss}(l) = 2(0.8)^{|l|} \qquad R_{ww}(l) = 0.5\delta(l)$$

Find the optimum realizable IIR Wiener filter and the resulting mean-square error.

8.17. For the conditions in Problem 8.16, let $d(n) = s(n + 1)$ (prediction) and repeat the problem.

8.18. For the conditions in Problem 8.16, let $d(n) = s(n - 1)$ (smoothing) and repeat the problem.

8.19. Suppose that we are presented with signal plus zero-mean independent noise and we wish to estimate the signal.

$$x(n) = s(n) + w(n)$$
$$d(n) = s(n)$$

where

$$R_{ss}(l) = 1.2(0.7)^{|l|} \qquad R_{ww}(l) = 0.2\delta(l)$$

Find the optimum realizable IIR Wiener filter and the resulting mean-square error.

8.20. For the conditions in Problem 8.19, let $d(n) = s(n + 1)$ (prediction) and repeat the problem.

8.21. For the conditions in Problem 8.19, let $d(n) = s(n - 1)$ (smoothing) and repeat the problem.

8.22. Suppose that we are presented with signal plus zero-mean independent noise and we wish to estimate the signal.

$$x(n) = s(n) + w(n)$$
$$d(n) = s(n)$$

where

$$R_{ss}(l) = 2(0.8)^{|l|} \qquad R_{ww}(l) = 0.5\delta(l)$$

Find the recursive (Kalman) estimate and the resulting mean-square error for $n = 1, 2, 3$, and 4.

8.23. For the conditions in Problem 8.22, let $d(n) = s(n + 1)$ (prediction) and repeat the problem.

8.24. For the conditions in Problem 8.22, let $d(n) = s(n - 1)$ (smoothing) and repeat the problem.

8.25. Suppose that we are presented with signal plus zero-mean independent noise and we wish to estimate the signal.

$$x(n) = s(n) + w(n)$$
$$d(n) = s(n)$$

where

$$R_{ss}(l) = 1.2(0.7)^{|l|} \qquad R_{ww}(l) = 0.2\delta(l)$$

Find the recursive (Kalman) estimate and the resulting mean-square error for $n = 1, 2, 3$, and 4.

8.26. For the conditions in Problem 8.16, let $d(n) = s(n + 1)$ (prediction) and repeat the problem.

8.27. For the conditions in Problem 8.16, let $d(n) = s(n - 1)$ (smoothing) and repeat the problem.

Further Reading

1. ATHANASIOS PAPOULIS, *Probability, Random Variables, and Stochastic Processes,* 3rd ed., McGraw-Hill, New York, 1991.
2. CHARLES W. THERRIEN, *Discrete Random Signals and Statistical Signal Processing,* Prentice Hall, Englewood Cliffs, NJ, 1992.
3. JAMES A. CADZOW, *Foundations of Digital Signal Processing and Data Analysis,* Macmillan, New York, 1987.