1

Review of Random Signals

1.1 Probability Density, Mean, Variance

In this section, we present a short review of probability concepts. It is assumed that the reader has some familiarity with the subject on the level of Papoulis' book [1].

Let x be a *random variable* having probability density $p(x)$. Its *mean*, *variance*, and *second moment* are defined by the expectation values

$$m = E[x] = \int_{-\infty}^{\infty} xp(x)\,dx = \text{mean}$$

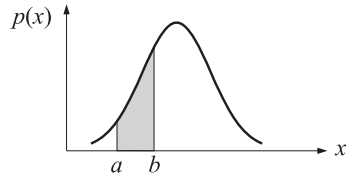
$$\sigma^2 = \text{Var}(x) = E[(x-m)^2] = \int_{-\infty}^{\infty} (x-m)^2 p(x)\,dx = \text{variance}$$

$$E[x^2] = \int_{-\infty}^{\infty} x^2 p(x)\,dx = \text{second moment}$$

These quantities are known as *second-order statistics* of the random variable x. Their importance is linked with the fact that most optimal filter design criteria require knowledge only of the second-order statistics and do not require more detailed knowledge, such as probability densities. It is necessary, then, to be able to extract such quantities from the actual measured data.

The probability that the random variable x will assume a value within an interval of values $[a,b]$ is given by

$$\text{Prob}[a \le x \le b] = \int_a^b p(x)\,dx = \text{shaded area}$$



The probability density is always normalized to unity by

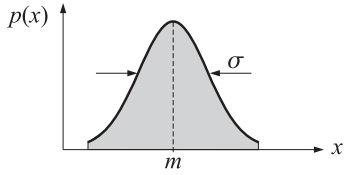
$$\int_{-\infty}^{\infty} p(x)\,dx = 1$$

which states that the probability of x taking a value somewhere within its range of variation is unity, that is, certainty. This property also implies

$$\sigma^2 = E\big[(x-m)^2\big] = E[x^2] - m^2$$

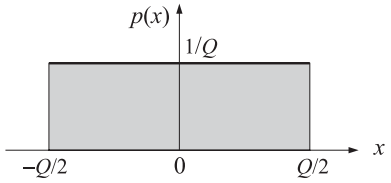
Example 1.1.1: Gaussian, or normal, distribution

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\big[-(x-m)^2/2\sigma^2\big]$$



Example 1.1.2: Uniform distribution

$$p(x) = \begin{cases} 1/Q, & \text{for } -Q/2 \le x \le Q/2 \\ 0, & \text{otherwise} \end{cases}$$



Its variance is $\sigma^2 = Q^2/12$. □

Both the gaussian and the uniform distributions will prove to be important examples. In typical signal processing problems of designing filters to remove or separate noise from signal, it is often assumed that the noise interference is gaussian. This assumption is justified on the grounds of the central limit theorem, provided that the noise arises from many different noise sources acting independently of each other.

The uniform distribution is also important. In digital signal processing applications. the quantization error arising from the signal quantization in the A/D converters, or the roundoff error arising from the finite accuracy of the internal arithmetic operations in digital filters, can often be assumed to be uniformly distributed.

Every computer provides system routines for the generation of random numbers. For example, the routines RANDU and GAUSS of the IBM Scientific Subroutine Package generate uniformly distributed random numbers over the interval $[0,1]$, and gaussian-distributed numbers, respectively. GAUSS calls RANDU twelve times, thus generating twelve independent uniformly distributed random numbers $x_1, x_2, \ldots, x_{12}$. Then, their sum $x = x_1 + x_2 + \cdots + x_{12}$, will be approximately gaussian, as guaranteed by the central limit theorem. It is interesting to note that the variance of x is unity, as it follows from the fact that the variance of each x_i, is $1/12$:

$$\sigma_x^2 = \sigma_{x_1}^2 + \sigma_{x_2}^2 + \cdots + \sigma_{x_{12}}^2 = \frac{1}{12} + \frac{1}{12} + \cdots + \frac{1}{12} = 1$$

The mean of x is $12/2 = 6$. By shifting and scaling x, one can obtain a gaussian-distributed random number of any desired mean and variance.

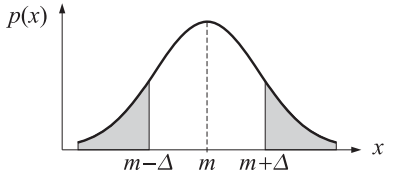
1.2 Chebyshev's Inequality

The variance σ^2 of a random variable x is a measure of the spread of the x-values about their mean. This intuitive interpretation of the variance is a direct consequence of Chebyshev's inequality, which states that the x-values tend to cluster about their mean in the sense that the probability of a value not occurring in the near vicinity of the mean is small; and it is smaller the smaller the variance.

More precisely, for any probability density $p(x)$ and any $\Delta > 0$, the probability that x will fall outside the interval of values $[m-\Delta, m+\Delta]$ is bounded by σ^2/Δ^2. Thus, for fixed Δ, as the variance σ^2 becomes smaller, the x-values tend to cluster more narrowly about the mean. In the extreme limiting case of a deterministic variable $x = m$, the density becomes infinitely narrow, $p(x) = \delta(x-m)$, and has zero variance.

$$\text{Prob}\big[|x-m| \ge \Delta\big] \le \frac{\sigma^2}{\Delta^2}$$

(Chebyshev's Inequality)



Chebyshev's inequality is especially important in proving asymptotic convergence results for sample estimates of parameters. For example, consider N independent samples $\{x_1, x_2, \ldots, x_N\}$ drawn from a gaussian probability distribution of mean m and variance σ^2. The sample estimate of the mean is

$$\hat{m} = \frac{1}{N}(x_1 + x_2 + \cdots + x_N) \tag{1.2.1}$$

Being a sum of N gaussian random variables, $\hat{m}$ will itself be a gaussian random variable. Its probability density is completely determined by the corresponding mean and variance. These are found as follows.

$$E[\hat{m}] = \frac{1}{N}\big(E[x_1] + E[x_2] + \cdots + E[x_N]\big) = \frac{1}{N}(m + m + \cdots + m) = m$$

Therefore, $\hat{m}$ is an *unbiased estimator* of m. However, the goodness of $\hat{m}$ as an estimator must be judged by how small its variance is—the smaller the better, by Chebyshev's inequality. By the assumption of independence, we have

$$\text{var}(\hat{m}) = E\big[(\hat{m}-m)^2\big] = \frac{1}{N^2}\big(\sigma_{x_1}^2 + \sigma_{x_2}^2 + \cdots + \sigma_{x_N}^2\big) = \frac{1}{N^2}(N\sigma^2) = \frac{\sigma^2}{N} \tag{1.2.2}$$

Thus, $\hat{m}$ is also a *consistent estimator* of m in the sense that its variance tends to zero as the number of samples N increases. The values of $\hat{m}$ will tend to cluster more and more closely about the true value of m as N becomes larger. Chebyshev's inequality implies that the probability of $\hat{m}$ falling outside any fixed neighborhood of m will tend to zero for large N. Equivalently, $\hat{m}$ will converge to m with probability one. This can also be seen from the probability density of $\hat{m}$, which is the gaussian

$$p(\hat{m}) = \frac{N^{1/2}}{(2\pi)^{1/2}\sigma}\exp\big[-\frac{N}{2\sigma^2}(\hat{m}-m)^2\big]$$

In the limit of large N, this density tends to the infinitely narrow delta function density $p(\hat{m}) = \delta(\hat{m} - m)$. In addition to the sample mean, we may also compute sample estimates of the variance σ^2 by

$$\hat{\sigma}^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{m})^2 \tag{1.2.3}$$

It is easily shown [2,3] that this estimator is slightly biased. But for large N, it is *asymptotically unbiased* and *consistent* as can be seen from its mean and variance:

$$E[\hat{\sigma}^2] = \frac{N-1}{N}\sigma^2, \quad \text{var}(\hat{\sigma}^2) = \frac{N-1}{N^2}2\sigma^4 \tag{1.2.4}$$

An unbiased and consistent estimator of σ^2 is the *standard deviation* defined by

$$s^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \hat{m})^2 \tag{1.2.5}$$

It has $E[s^2] = \sigma^2$ and $\text{var}(s^2) = 2\sigma^4/(N-1)$. In addition to the requirements of asymptotic unbiasedness and consistency, a good estimator of a parameter must also be judged in terms of its *efficiency* [2,3], which determines how closely the estimator meets its Cramér-Rao bound. This is discussed in Sec. 1.18. We will see there that the estimators (1.2.1) and (1.2.3)—being maximum likelihood estimators—are asymptotically efficient.

1.3 Joint and Conditional Densities, and Bayes' Rule

Next, we discuss random vectors. A pair of two different random variables $\mathbf{x} = (x_1, x_2)$ may be thought of as a vector-valued random variable. Its statistical description is more complicated than that of a single variable and requires knowledge of the joint probability density $p(x_1, x_2)$. The two random variables may or may not have any dependence on each other. It is possible, for example, that if x_2 assumes a particular value, then this fact may influence, or restrict, the possible values that x_1 can then assume.

A quantity that provides a measure for the degree of dependence of the two variables on each other is the conditional density $p(x_1|x_2)$ of x_1 given x_2; and $p(x_2|x_1)$ of x_2 given x_1. These are related by Bayes' rule

$$p(x_1, x_2) = p(x_1|x_2)\,p(x_2) = p(x_2|x_1)\,p(x_1)$$

More generally, Bayes' rule for two events A and B is

$$p(A, B) = p(A|B)\,p(B) = p(B|A)\,p(A)$$

The two random variables x_1 and x_2 are *independent* of each other if they do not condition each other in any way, that is, if

$$p(x_1|x_2) = p(x_1) \quad \text{or} \quad p(x_2|x_1) = p(x_2)$$

In other words, the occurrence of x_2 does not in any way influence the variable x_1. When two random variables are independent, their joint density factors into the product of single (marginal) densities:

$$p(x_1, x_2) = p(x_1)\,p(x_2)$$

The converse is also true. The *correlation* between x_1 and x_2 is defined by the expectation value

$$E[x_1 x_2] = \iint x_1 x_2 p(x_1, x_2)\, dx_1 dx_2$$

When x_1 and x_2 are independent, the correlation also factors as $E[x_1 x_2] = E[x_1]E[x_2]$.

Example 1.3.1: Suppose x_1 is related to x_2 by

$$x_1 = 5x_2 + v$$

where v is a zero-mean, unit-variance, gaussian random variable assumed to be independent of x_2. Determine the conditional density and conditional mean of x_1 given x_2.

Solution: The randomness of x_1 arises both from the randomness of x_2 and the randomness of v. But if x_2 takes on a particular value, then the randomness of x_1 will arise only from v. Identifying elemental probabilities we have

$$p(x_1|x_2)\,dx_1 = p(v)\,dv = (2\pi)^{-1/2}\exp\left(-\frac{1}{2}v^2\right)dv$$

But, $dx_1 = dv$ and $v = x_1 - 5x_2$. Therefore,

$$p(x_1|x_2) = (2\pi)^{-1/2}\exp\left[-\frac{1}{2}(x_1 - 5x_2)^2\right]$$

The conditional mean is the mean of x_1 with respect to the density $p(x_1|x_2)$. It is evident from the above gaussian expression that the conditional mean is $E[x_1|x_2] = 5x_2$. This can also be found directly as follows.

$$E[x_1|x_2] = E\left[(5x_2 + v)|x_2\right] = 5x_2 + E[v|x_2] = 5x_2$$

where we used the independence of v and x_2 to replace the conditional mean of v with its unconditional mean, which was given to be zero, that is, $E[v|x_2] = E[v] = 0$. □

The concept of a random vector generalizes to any dimension. A vector of N random variables

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}$$

requires knowledge of the joint density

$$p(\mathbf{x}) = p(x_1, x_2, \ldots, x_N) \tag{1.3.1}$$

for its complete statistical description. The second-order statistics of $\mathbf{x}$ are its mean, its *correlation matrix*, and its *covariance matrix*, defined by

$$\mathbf{m} = E[\mathbf{x}]\,, \quad R = E[\mathbf{x}\mathbf{x}^T]\,, \quad \Sigma = E\big[(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T\big] \tag{1.3.2}$$

where the superscript T denotes transposition, and the expectation operations are defined in terms of the joint density (1.3.1); for example,

$$E[\mathbf{x}] = \int \mathbf{x}\, p(\mathbf{x})\, d^N\mathbf{x}$$

where $d^N\mathbf{x} = dx_1 dx_2 \cdots dx_N$ denotes the corresponding N-dimensional volume element. The ijth matrix element of the correlation matrix R is the correlation between the ith random variable x_i with the jth random variable x_j, that is, $R_{ij} = E[x_i x_j]$. It is easily shown that the covariance and correlation matrices are related by

$$\Sigma = R - \mathbf{m}\mathbf{m}^T$$

When the mean is zero, R and Σ coincide. Both R and Σ are *symmetric positive semi-definite* matrices.

Example 1.3.2: The probability density of a gaussian random vector $\mathbf{x} = [x_1, x_2, \ldots, x_N]^T$ is completely specified by its mean $\mathbf{m}$ and covariance matrix Σ, that is,

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{N/2}(\det\Sigma)^{1/2}}\, \exp\big[-\frac{1}{2}(\mathbf{x}-\mathbf{m})^T\Sigma^{-1}(\mathbf{x}-\mathbf{m})\big]$$

Example 1.3.3: Under a linear transformation, a gaussian random vector remains gaussian. Let $\mathbf{x}$ be a gaussian random vector of dimension N, mean $\mathbf{m}_x$, and covariance Σ_x. Show that the linearly transformed vector

$$\boldsymbol{\xi} = B\mathbf{x} \qquad \text{where } B \text{ is a nonsingular } N\times N \text{ matrix}$$

is gaussian-distributed with mean and covariance given by

$$\mathbf{m}_\xi = B\mathbf{m}_x\,, \quad \Sigma_\xi = B\Sigma_x B^T \tag{1.3.3}$$

The relationships (1.3.3) are valid also for non-gaussian random vectors. They are easily derived as follows:

$$E[\boldsymbol{\xi}] = E[B\mathbf{x}] = BE[\mathbf{x}]\,, \quad E[\boldsymbol{\xi}\boldsymbol{\xi}^T] = E[B\mathbf{x}(B\mathbf{x})^T] = BE[\mathbf{x}\mathbf{x}^T]B^T$$

The probability density $p_\xi(\boldsymbol{\xi})$ is related to the density $p_x(\mathbf{x})$ by the requirement that, under the above change of variables, they both yield the same elemental probabilities:

$$p_\xi(\boldsymbol{\xi})\, d^N\boldsymbol{\xi} = p_x(\mathbf{x})\, d^N\mathbf{x} \tag{1.3.4}$$

Since the Jacobian of the transformation from $\mathbf{x}$ to $\boldsymbol{\xi}$ is $d^N\boldsymbol{\xi} = |\det B| d^N\mathbf{x}$, we obtain $p_\xi(\boldsymbol{\xi}) = p_x(\mathbf{x})/|\det B|$. Noting the invariance of the quadratic form

$$\begin{aligned}(\boldsymbol{\xi}-\mathbf{m}_\xi)^T\Sigma_\xi^{-1}(\boldsymbol{\xi}-\mathbf{m}_\xi) &= (\mathbf{x}-\mathbf{m}_x)^T B^T (B\Sigma_x B^T)^{-1} B(\mathbf{x}-\mathbf{m}_x)\\ &= (\mathbf{x}-\mathbf{m}_x)^T\Sigma_x^{-1}(\mathbf{x}-\mathbf{m}_x)\end{aligned}$$

and that $\det\Sigma_\xi = \det(B\Sigma_x B^T) = (\det B)^2 \det\Sigma_x$, we obtain

$$p_\xi(\boldsymbol{\xi}) = \frac{1}{(2\pi)^{N/2}(\det\Sigma_\xi)^{1/2}}\, \exp\big[-\frac{1}{2}(\boldsymbol{\xi}-\mathbf{m}_\xi)^T\Sigma_\xi^{-1}(\boldsymbol{\xi}-\mathbf{m}_\xi)\big]$$

Example 1.3.4: Consider two zero-mean random vectors $\mathbf{x}$ and $\mathbf{y}$ of dimensions N and M, respectively. Show that if they are *uncorrelated and jointly gaussian*, then they are also *independent* of each other. That $\mathbf{x}$ and $\mathbf{y}$ are jointly gaussian means that the $(N+M)$-dimensional joint vector $\mathbf{z} = \begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix}$ is zero-mean and gaussian, that is,

$$p(\mathbf{z}) = \frac{1}{(2\pi)^{(N+M)/2}(\det R_{zz})^{1/2}}\, \exp\big[-\frac{1}{2}\mathbf{z}^T R_{zz}^{-1}\mathbf{z}\big]$$

where the correlation (covariance) matrix R_{zz} is

$$R_{zz} = E\left[\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix}[\mathbf{x}^T, \mathbf{y}^T]\right] = \begin{bmatrix}E[\mathbf{x}\mathbf{x}^T] & E[\mathbf{x}\mathbf{y}^T]\\ E[\mathbf{y}\mathbf{x}^T] & E[\mathbf{y}\mathbf{y}^T]\end{bmatrix} = \begin{bmatrix}R_{xx} & R_{xy}\\ R_{yx} & R_{yy}\end{bmatrix}$$

If $\mathbf{x}$ and $\mathbf{y}$ are uncorrelated, that is, $R_{xy} = E[\mathbf{x}\mathbf{y}^T] = 0$, then the matrix R_{zz} becomes block diagonal and the quadratic form of the joint vector becomes the sum of the individual quadratic forms:

$$\mathbf{z}^T R_{zz}^{-1}\mathbf{z} = [\mathbf{x}^T, \mathbf{y}^T]\begin{bmatrix}R_{xx}^{-1} & 0\\ 0 & R_{yy}^{-1}\end{bmatrix}\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix} = \mathbf{x}^T R_{xx}^{-1}\mathbf{x} + \mathbf{y}^T R_{yy}^{-1}\mathbf{y}$$

Since $R_{xy} = 0$ also implies that $\det R_{zz} = (\det R_{xx})(\det R_{yy})$, it follows that the joint density $p(\mathbf{z}) = p(\mathbf{x},\mathbf{y})$ factors into the marginal densities:

$$p(\mathbf{x},\mathbf{y}) = p(\mathbf{x})\, p(\mathbf{y})$$

which shows the independence of $\mathbf{x}$ and $\mathbf{y}$.

Example 1.3.5: Given a random vector $\mathbf{x}$ with mean $\mathbf{m}$ and covariance Σ, show that the best choice of a *deterministic* vector $\hat{\mathbf{x}}$ which minimizes the quantity

$$R_{ee} = E[\mathbf{e}\mathbf{e}^T] = \text{minimum}\,, \quad \text{where } \mathbf{e} = \mathbf{x} - \hat{\mathbf{x}},$$

is the mean $\mathbf{m}$ itself, that is, $\hat{\mathbf{x}} = \mathbf{m}$. Also show that for this optimal choice of $\hat{\mathbf{x}}$, the actual minimum value of the quantity R_{ee} is the covariance Σ. This property is easily shown by working with the deviation of $\hat{\mathbf{x}}$ from the mean $\mathbf{m}$, that is, let

$$\hat{\mathbf{x}} = \mathbf{m} + \boldsymbol{\Delta}$$

Then, the quantity R_{ee} becomes

$$\begin{aligned}R_{ee} &= E[\mathbf{e}\mathbf{e}^T] = E\big[(\mathbf{x}-\mathbf{m}-\boldsymbol{\Delta})(\mathbf{x}-\mathbf{m}-\boldsymbol{\Delta})^T\big]\\ &= E\big[(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T\big] - \boldsymbol{\Delta}\, E[\mathbf{x}^T - \mathbf{m}^T] - E[\mathbf{x}-\mathbf{m}]\,\boldsymbol{\Delta} + \boldsymbol{\Delta}\boldsymbol{\Delta}^T\\ &= \Sigma + \boldsymbol{\Delta}\boldsymbol{\Delta}^T\end{aligned}$$

where we used the fact that $E[\mathbf{x}-\mathbf{m}] = E[\mathbf{x}] - \mathbf{m} = 0$. Since the matrix $\boldsymbol{\Delta}\boldsymbol{\Delta}^T$ is nonnegative-definite, it follows that R_{ee}, will be minimized when $\boldsymbol{\Delta} = 0$, and in this case the minimum value will be $R_{ee}^{\min} = \Sigma$.

Since R_{ee} is a matrix, the sense in which it is minimized must be clarified. The statement that R_{ee} is greater than $R_{ee}^{\min}$ means that the difference $R_{ee} - R_{ee}^{\min}$ is a positive semi-definite (and symmetric) matrix, and therefore we have for the scalar quantities: $\mathbf{a}^T R_{ee}\mathbf{a} \geq \mathbf{a}^T R_{ee}^{\min}\mathbf{a}$ for any vector $\mathbf{a}$. □

1.4 Correlation Canceling and Optimum Estimation

The concept of correlation canceling plays a central role in the development of many optimum signal processing algorithms, because a correlation canceler is also the best linear processor for estimating one signal from another.

Consider two zero-mean random vectors $\mathbf{x}$ and $\mathbf{y}$ of dimensions N and M, respectively. If $\mathbf{x}$ and $\mathbf{y}$ are correlated with each other in the sense that $R_{xy} = E[\mathbf{x}\mathbf{y}^T] \neq 0$, then we may remove such correlations by means of a linear transformation of the form

$$\mathbf{e} = \mathbf{x} - H\mathbf{y} \tag{1.4.1}$$

where the $N \times M$ matrix H must be suitably chosen such that the new pair of vectors $\mathbf{e}, \mathbf{y}$ are no longer correlated with each other, that is, we require

$$R_{ey} = E[\mathbf{e}\mathbf{y}^T] = 0 \tag{1.4.2}$$

Using Eq. (1.4.1), we obtain

$$R_{ey} = E[\mathbf{e}\mathbf{y}^T] = E\big[(\mathbf{x} - H\mathbf{y})\mathbf{y}^T\big] = E[\mathbf{x}\mathbf{y}^T] - HE[\mathbf{y}\mathbf{y}^T] = R_{xy} - HR_{yy}$$

Then, the condition $R_{ey} = 0$ immediately implies that

$$\boxed{H = R_{xy}R_{yy}^{-1} = E[\mathbf{x}\mathbf{y}^T]E[\mathbf{y}\mathbf{y}^T]^{-1}} \tag{1.4.3}$$

Using $R_{ey} = 0$, the covariance matrix of the resulting vector $\mathbf{e}$ is easily found to be

$$R_{ee} = E[\mathbf{e}\mathbf{e}^T] = E\big[\mathbf{e}(\mathbf{x}^T - \mathbf{y}^T H)\big] = R_{ex} - R_{ey}H^T = R_{ex} = E\big[(\mathbf{x} - H\mathbf{y})\mathbf{x}^T\big], \quad \text{or,}$$

$$R_{ee} = R_{xx} - HR_{yx} = R_{xx} - R_{xy}R_{yy}^{-1}R_{yx} \tag{1.4.4}$$

The vector

$$\hat{\mathbf{x}} = H\mathbf{y} = R_{xy}R_{yy}^{-1}\mathbf{y} = E[\mathbf{x}\mathbf{y}^T]E[\mathbf{y}\mathbf{y}^T]^{-1}\mathbf{y} \tag{1.4.5}$$

obtained by linearly processing the vector $\mathbf{y}$ by the matrix H is called the *linear regression*, or *orthogonal projection*, of $\mathbf{x}$ on the vector $\mathbf{y}$. In a sense to be made precise later, $\hat{\mathbf{x}}$ also represents the best "copy," or *estimate*, of $\mathbf{x}$ that can be made on the basis of the vector $\mathbf{y}$. Thus, the vector $\mathbf{e} = \mathbf{x} - H\mathbf{y} = \mathbf{x} - \hat{\mathbf{x}}$ may be thought of as the *estimation error*.

Actually, it is better to think of $\hat{\mathbf{x}} = H\mathbf{y}$ not as an estimate of $\mathbf{x}$ but rather as an estimate of *that part* of $\mathbf{x}$ which is correlated with $\mathbf{y}$. Indeed, suppose that $\mathbf{x}$ consists of two parts

$$\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2$$

such that $\mathbf{x}_1$ is correlated with $\mathbf{y}$, but $\mathbf{x}_2$ is not, that is, $R_{x_2y} = E[\mathbf{x}_2\mathbf{y}^T] = 0$. Then,

$$R_{xy} = E[\mathbf{x}\mathbf{y}^T] = E\big[(\mathbf{x}_1 + \mathbf{x}_2)\mathbf{y}^T\big] = R_{x_1y} + R_{x_2y} = R_{x_1y}$$

and therefore,

$$\hat{\mathbf{x}} = R_{xy}R_{yy}^{-1}\mathbf{y} = R_{x_1y}R_{yy}^{-1}\mathbf{y} = \hat{\mathbf{x}}_1$$

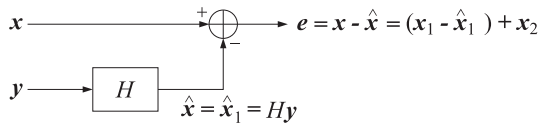


Fig. 1.4.1 Correlation canceler.

The vector $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}} = \mathbf{x}_1 + \mathbf{x}_2 - \hat{\mathbf{x}}_1 = (\mathbf{x}_1 - \hat{\mathbf{x}}_1) + \mathbf{x}_2$ consists of the estimation error $(\mathbf{x}_1 - \hat{\mathbf{x}}_1)$ of the $\mathbf{x}_1$-part plus the $\mathbf{x}_2$-part. Both of these terms are separately uncorrelated from $\mathbf{y}$. These operations are summarized in block diagram form in Fig. 1.4.1.

The most important feature of this arrangement is the *correlation cancellation* property which may be summarized as follows: If $\mathbf{x}$ has a part $\mathbf{x}_1$ which is correlated with $\mathbf{y}$, then this part will tend to be canceled as much as possible from the output $\mathbf{e}$. The linear processor H accomplishes this by converting $\mathbf{y}$ into the *best possible copy* $\hat{\mathbf{x}}_1$ of $\mathbf{x}_1$ and then proceeds to cancel it from the output. The output vector $\mathbf{e}$ is no longer correlated with $\mathbf{y}$. The part $\mathbf{x}_2$ of $\mathbf{x}$ which is uncorrelated with $\mathbf{y}$ remains entirely unaffected. It cannot be estimated in terms of $\mathbf{y}$.

The correlation canceler may also be thought of as an *optimal signal separator*. Indeed, the output of the processor H is essentially the $\mathbf{x}_1$ component of $\mathbf{x}$, whereas the output $\mathbf{e}$ is essentially the $\mathbf{x}_2$ component. The separation of $\mathbf{x}$ into $\mathbf{x}_1$ and $\mathbf{x}_2$ is optimal in the sense that the $\mathbf{x}_1$ component of $\mathbf{x}$ is removed as much as possible from $\mathbf{e}$.

Next, we discuss the *best linear estimator property* of the correlation canceler. The choice $H = R_{xy}R_{yy}^{-1}$, which guarantees correlation cancellation, is also the choice that gives the *best estimate* of $\mathbf{x}$ as a *linear* function of $\mathbf{y}$ in the form $\hat{\mathbf{x}} = H\mathbf{y}$. It is the best estimate in the sense that it produces the lowest *mean-square* estimation error. To see this, express the covariance matrix of the estimation error in terms of H, as follows:

$$R_{ee} = E[\mathbf{e}\mathbf{e}^T] = E\big[(\mathbf{x} - H\mathbf{y})(\mathbf{x}^T - \mathbf{y}^T H^T)\big] = R_{xx} - HR_{yx} - R_{xy}H^T + HR_{yy}H^T \tag{1.4.6}$$

Minimizing this expression with respect to H yields the optimum choice of H:

$$H_{\text{opt}} = R_{xy}R_{yy}^{-1}$$

with the minimum value for R_{ee} given by:

$$R_{ee}^{\min} = R_{xx} - R_{xy}R_{yy}^{-1}R_{yx}$$

Any other value will result in a larger value for R_{ee}. An alternative way to see this is to consider a deviation ΔH of H from its optimal value, that is, in (1.4.5) replace H by

$$H = H_{\text{opt}} + \Delta H = R_{xy}R_{yy}^{-1} + \Delta H$$

Then Eq. (1.4.6) may be expressed in terms of ΔH as follows:

$$R_{ee} = R_{ee}^{\min} + \Delta H\, R_{yy}\, \Delta H^T$$

Since R_{yy} is positive definite, the second term always represents a nonnegative contribution above the minimum value $R_{ee}^{\min}$, so that $(R_{ee} - R_{ee}^{\min})$ is positive semi-definite. In summary, there are three useful ways to think of the correlation canceler:

1. Optimal estimator of $\mathbf{x}$ from $\mathbf{y}$.
2. Optimal canceler of that part of $\mathbf{x}$ which is correlated with $\mathbf{y}$.
3. Optimal signal separator

The point of view is determined by the application. The first view is typified by Kalman filtering, channel equalization, and linear prediction applications. The second view is taken in echo canceling, noise canceling, and sidelobe canceling applications. The third view is useful in the adaptive line enhancer, which is a method of adaptively separating a signal into its broadband and narrowband components. All of these applications are considered later on.

Example 1.4.1: If $\mathbf{x}$ and $\mathbf{y}$ are *jointly gaussian*, show that the linear estimate $\hat{\mathbf{x}} = H\mathbf{y}$ is also the *conditional mean* $E[\mathbf{x}|\mathbf{y}]$ of the vector $\mathbf{x}$ given the vector $\mathbf{y}$. The conditional mean is defined in terms of the conditional density $p(\mathbf{x}|\mathbf{y})$ of $\mathbf{x}$ given $\mathbf{y}$ as follows:

$$E[\mathbf{x}|\mathbf{y}] = \int \mathbf{x}\, p(\mathbf{x}|\mathbf{y})\, d^N\mathbf{x}$$

Instead of computing this integral, we will use the results of Examples 1.3.3 and 1.3.4. The transformation from the jointly gaussian pair $(\mathbf{x},\mathbf{y})$ to the uncorrelated pair $(\mathbf{e},\mathbf{y})$ is linear:

$$\begin{bmatrix} \mathbf{e} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} I_N & -H \\ 0 & I_M \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}$$

where I_N and I_M are the unit matrices of dimensions N and M, respectively. Therefore, Example 1.3.3 implies that the transformed pair $(\mathbf{e},\mathbf{y})$ is also jointly gaussian. Furthermore, since $\mathbf{e}$ and $\mathbf{y}$ are uncorrelated, it follows from Example 1.3.4 that they must be independent of each other. The conditional mean of $\mathbf{x}$ can be computed by writing

$$\mathbf{x} = \hat{\mathbf{x}} + \mathbf{e} = H\mathbf{y} + \mathbf{e}$$

and noting that if $\mathbf{y}$ is given, then $H\mathbf{y}$ is no longer random. Therefore,

$$E[\mathbf{x}|\mathbf{y}] = E\big[(H\mathbf{y} + \mathbf{e})|\mathbf{y}\big] = H\mathbf{y} + E[\mathbf{e}|\mathbf{y}]$$

Since $\mathbf{e}$ and $\mathbf{y}$ are independent, the conditional mean $E[\mathbf{e}|\mathbf{y}]$ is the same as the unconditional mean $E[\mathbf{e}]$, which is zero by the zero-mean assumption. Thus,

$$E[\mathbf{x}|\mathbf{y}] = H\mathbf{y} = R_{xy}R_{yy}^{-1}\mathbf{y} \qquad \text{(jointly gaussian } \mathbf{x} \text{ and } \mathbf{y}\text{)} \tag{1.4.7}$$

Example 1.4.2: Show that the conditional mean $E[\mathbf{x}|\mathbf{y}]$ is the best *unrestricted* (i.e., not necessarily linear) estimate of $\mathbf{x}$ in the *mean-square* sense. The best linear estimate was obtained by seeking the best linear function of $\mathbf{y}$ that minimized the error criterion (1.4.6), that is, we required a priori that the estimate was to be of the form $\hat{\mathbf{x}} = H\mathbf{y}$. Here, our task is more general: find the most general function of $\mathbf{y}$, $\hat{\mathbf{x}} = \hat{\mathbf{x}}(\mathbf{y})$, which gives the best estimate of $\mathbf{x}$, in the sense of producing the lowest mean-squared estimation error $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}(\mathbf{y})$,

$$R_{ee} = E[\mathbf{e}\mathbf{e}^T] = E\big[(\mathbf{x} - \hat{\mathbf{x}}(\mathbf{y}))\,(\mathbf{x}^T - \hat{\mathbf{x}}(\mathbf{y})^T)\big] = \min$$

The functional dependence of $\hat{\mathbf{x}}(\mathbf{y})$ on $\mathbf{y}$ is not required to be linear a priori. Using $p(\mathbf{x},\mathbf{y}) = p(\mathbf{x}|\mathbf{y})\,p(\mathbf{y})$, the above expectation may be written as

$$\begin{aligned} R_{ee} &= \int (\mathbf{x} - \hat{\mathbf{x}}(\mathbf{y}))\,(\mathbf{x}^T - \hat{\mathbf{x}}(\mathbf{y})^T)\, p(\mathbf{x},\mathbf{y})\, d^N\mathbf{x}\, d^M\mathbf{y} \\ &= \int p(\mathbf{y})\, d^M\mathbf{y} \left[\int (\mathbf{x} - \hat{\mathbf{x}}(\mathbf{y}))\,(\mathbf{x}^T - \hat{\mathbf{x}}(\mathbf{y})^T)\, p(\mathbf{x}|\mathbf{y})\, d^N\mathbf{x} \right] \end{aligned}$$

Since $p(\mathbf{y})$ is nonnegative for all $\mathbf{y}$, it follows that R_{ee} will be minimized when the quantity

$$\int (\mathbf{x} - \hat{\mathbf{x}}(\mathbf{y}))\,(\mathbf{x}^T - \hat{\mathbf{x}}(\mathbf{y})^T)\, p(\mathbf{x}|\mathbf{y})\, d^N\mathbf{x}$$

is minimized with respect to $\hat{\mathbf{x}}$. But we know from Example 1.3.5 that this quantity is minimized when $\hat{\mathbf{x}}$ is chosen to be the corresponding mean; here, this is the mean with respect to the density $p(\mathbf{x}|\mathbf{y})$. Thus,

$$\hat{\mathbf{x}}(\mathbf{y}) = E[\mathbf{x}|\mathbf{y}] \tag{1.4.8}$$

To summarize, we have seen that

$$\hat{\mathbf{x}} = H\mathbf{y} = R_{xy}R_{yy}^{-1}\mathbf{y} = \text{best } \textit{linear} \text{ mean-square estimate of } \mathbf{x}$$

$$\hat{\mathbf{x}} = E[\mathbf{x}|\mathbf{y}] = \text{best } \textit{unrestricted} \text{ mean-square estimate of } \mathbf{x}$$

and Example 1.4.1 shows that the two are equal in the case of jointly gaussian vectors $\mathbf{x}$ and $\mathbf{y}$.

The concept of correlation canceling and its application to signal estimation problems will be discussed in more detail in Chap. 11. The adaptive implementation of the correlation canceler will be discussed in Chap. 16. In a typical signal processing application, the processor H would represent a *linear filtering operation* and the vectors $\mathbf{x}$ and $\mathbf{y}$ would be *blocks of signal samples*. The design of such processors requires knowledge of the quantities $R_{xy} = E[\mathbf{x}\mathbf{y}^T]$ and $R_{yy} = E[\mathbf{y}\mathbf{y}^T]$. How does one determine these? Basically, applications fall into two classes:

1. Both $\mathbf{x}$ and $\mathbf{y}$ are available for processing and the objective is to cancel the correlations that may exist between them.

2. Only the signal $\mathbf{y}$ is available for processing and the objective is to estimate the signal $\mathbf{x}$ on the basis of $\mathbf{y}$.

In the first class of applications, there exist two basic design approaches:

a. *Block processing* (off-line) methods. The required correlations R_{xy} and R_{yy} are computed on the basis of two actual blocks of signal samples $\mathbf{x}$ and $\mathbf{y}$ by replacing statistical averages by time averages.

b. *Adaptive processing* (on-line) methods. The quantities R_{xy} and R_{yy} are "learned" gradually as the data $\mathbf{x}$ and $\mathbf{y}$ become available in real time. The processor H is continually updated in response to the incoming data, until it reaches its optimal value.

Both methods are *data adaptive*. The first is adaptive on a *block-by-block* basis, whereas the second on a *sample-by-sample* basis. Both methods depend heavily on the assumption of *stationarity*. In block processing methods, the replacement of ensemble averages by time averages is justified by the assumption of ergodicity, which requires stationarity. The requirement of stationarity can place serious limitations on the allowed length of the signal blocks $\mathbf{x}$ and $\mathbf{y}$.

Similarly, in adaptive processing methods, convergence to the optimal value of the processor H again requires stationarity. Adaptive methods offer, however, the possibility of tracking nonstationary changes of the environment, as long as such changes occur slowly enough to allow convergence between changes. Thus, the issue of the speed of convergence of adaptation algorithms is an important one.

In the second class of applications where $\mathbf{x}$ is not available for processing, one must have a specific model of the relationship between $\mathbf{x}$ and $\mathbf{y}$ from which R_{xy} and R_{yy} may be calculated. This is, for example, what is done in Kalman filtering.

Example 1.4.3: As an example of the relationship that might exist between $\mathbf{x}$ and $\mathbf{y}$, let

$$y_n = xc_n + v_n\,, \quad n = 1, 2, \ldots, M$$

where x and v_n are zero-mean, unit-variance, random variables, and c_n are known coefficients. It is further assumed that v_n are mutually uncorrelated, and also uncorrelated with x, so that $E[v_n v_m] = \delta_{nm}$, $E[xv_n] = 0$. We would like to determine the optimal linear estimate (1.4.5) of x, and the corresponding estimation error (1.4.4). In obvious matrix notation we have $\mathbf{y} = \mathbf{c}\,x + \mathbf{v}$, with $E[x\mathbf{v}] = 0$ and $E[\mathbf{v}\mathbf{v}^T] = I$, where I is the $M\times M$ unit matrix. We find

$$E[x\mathbf{y}^T] = E[x(x\mathbf{c} + \mathbf{v})^T] = \mathbf{c}^T E[x^2] + E[x\mathbf{v}^T] = \mathbf{c}^T$$

$$E[\mathbf{y}\mathbf{y}^T] = E[(x\mathbf{c} + \mathbf{v})(x\mathbf{c} + \mathbf{v})^T] = \mathbf{c}\mathbf{c}^T E[x^2] + E[\mathbf{v}\mathbf{v}^T] = \mathbf{c}\mathbf{c}^T + I$$

and therefore, $H = E[x\mathbf{y}^T]E[\mathbf{y}\mathbf{y}^T]^{-1} = \mathbf{c}^T(I + \mathbf{c}\mathbf{c}^T)^{-1}$. Using the matrix inversion lemma we may write $(I + \mathbf{c}\mathbf{c}^T)^{-1} = I - \mathbf{c}(1 + \mathbf{c}^T\mathbf{c})^{-1}\mathbf{c}^T$, so that

$$H = \mathbf{c}^T[I - \mathbf{c}(1 + \mathbf{c}^T\mathbf{c})^{-1}\mathbf{c}^T] = (1 + \mathbf{c}^T\mathbf{c})^{-1}\mathbf{c}^T$$

The optimal estimate of x is then

$$\hat{x} = H\mathbf{y} = (1 + \mathbf{c}^T\mathbf{c})^{-1}\mathbf{c}^T\mathbf{y} \tag{1.4.9}$$

The corresponding estimation error is computed by

$$E[e^2] = R_{ee} = R_{xx} - HR_{yy} = 1 - (1 + \mathbf{c}^T\mathbf{c})^{-1}\mathbf{c}^T\mathbf{c} = (1 + \mathbf{c}^T\mathbf{c})^{-1}$$

1.5 Regression Lemma

The regression lemma is a key result in the derivation of the Kalman filter. The optimum estimate and estimation error of a (zero-mean) random vector $\mathbf{x}$ based on a (zero-mean) vector of observations $\mathbf{y}_1$ are given by

$$\hat{\mathbf{x}}_1 = R_{xy_1}R_{y_1y_1}^{-1}\mathbf{y}_1 = E[\mathbf{x}\mathbf{y}_1^T]E[\mathbf{y}_1\mathbf{y}_1^T]^{-1}\mathbf{y}_1$$

$$\mathbf{e}_1 = \mathbf{x} - \hat{\mathbf{x}}_1$$

$$R_{e_1e_1} = E[\mathbf{e}_1\mathbf{e}_1^T] = R_{xx} - R_{xy_1}R_{y_1y_1}^{-1}R_{y_1x}$$

If the observation set is enlarged by adjoining to it a new set of observations $\mathbf{y}_2$, so that the enlarged observation vector is $\mathbf{y} = \begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\end{bmatrix}$, the corresponding estimate of $\mathbf{x}$ will given by,

$$\hat{\mathbf{x}} = R_{xy}R_{yy}^{-1}\mathbf{y} = [R_{xy_1}, R_{xy_2}]\begin{bmatrix}R_{y_1y_1} & R_{y_1y_2}\\ R_{y_2y_1} & R_{y_2y_2}\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\end{bmatrix}$$

The *regression lemma* states that $\hat{\mathbf{x}}$ can be obtained by the following alternative expression of updating $\hat{\mathbf{x}}_1$ by the addition of a correction term,

$$\boxed{\hat{\mathbf{x}} = \hat{\mathbf{x}}_1 + R_{x\varepsilon_2}R_{\varepsilon_2\varepsilon_2}^{-1}\boldsymbol{\varepsilon}_2} \quad \text{(regression lemma)} \tag{1.5.1}$$

where $\boldsymbol{\varepsilon}_2$ is the innovations residual obtained by removing from $\mathbf{y}_2$ that part which is predictable from $\mathbf{y}_1$, that is,

$$\boldsymbol{\varepsilon}_2 = \mathbf{y}_2 - \hat{\mathbf{y}}_{2/1} = \mathbf{y}_2 - R_{y_2y_1}R_{y_1y_1}^{-1}\mathbf{y}_1$$

The improvement in using more observations is quantified by the following result, which shows that the mean-square error is reduced:

$$\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}} \quad \Rightarrow \quad R_{ee} = R_{e_1e_1} - R_{x\varepsilon_2}R_{\varepsilon_2\varepsilon_2}^{-1}R_{\varepsilon_2x} \tag{1.5.2}$$

where we defined,

$$R_{x\varepsilon_2} = R_{\varepsilon_2x}^T = E[\mathbf{x}\boldsymbol{\varepsilon}_2^T]\,, \quad R_{\varepsilon_2\varepsilon_2} = E[\boldsymbol{\varepsilon}_2\boldsymbol{\varepsilon}_2^T]$$

The proof of Eq. (1.5.1) is straightforward and is left as an exercise. As a hint, the following property may be used,

$$\begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\end{bmatrix} = \begin{bmatrix}I & 0\\ H & I\end{bmatrix}\begin{bmatrix}\mathbf{y}_1\\ \boldsymbol{\varepsilon}_2\end{bmatrix}, \quad \begin{bmatrix}R_{y_1y_1} & R_{y_1y_2}\\ R_{y_2y_1} & R_{y_2y_2}\end{bmatrix} = \begin{bmatrix}I & 0\\ H & I\end{bmatrix}\begin{bmatrix}R_{y_1y_1} & 0\\ 0 & R_{\varepsilon_2\varepsilon_2}\end{bmatrix}\begin{bmatrix}I & 0\\ H & I\end{bmatrix}^T$$

where $H = R_{y_2y_1}R_{y_1y_1}^{-1}$. A special case of this lemma is discussed next.

1.6 Gram-Schmidt Orthogonalization

In the previous section, we saw that any random vector $\mathbf{x}$ may be decomposed relative to another vector $\mathbf{y}$ into two parts, $\mathbf{x} = \hat{\mathbf{x}} + \mathbf{e}$, one part which is correlated with $\mathbf{y}$, and one which is not. These two parts are uncorrelated with each other since $R_{e\hat{x}} = E[\mathbf{e}\hat{\mathbf{x}}^T] = E[\mathbf{e}\mathbf{y}^T H^T] = E[\mathbf{e}\mathbf{y}^T]H^T = 0$. In a sense, they are orthogonal to each other. In this section, we will briefly develop such a geometrical interpretation.

The usefulness of the geometrical approach is threefold: First, it provides a very simple and intuitive framework in which to formulate and understand signal estimation problems. Second, through the Gram-Schmidt orthogonalization process, it provides the basis for making *signal models*, which find themselves in a variety of signal processing applications, such as speech synthesis, data compression, and modern methods of spectrum estimation. Third, again through the Gram-Schmidt construction, by decorrelating the given set of observations it provides the most convenient basis to work

with, containing no redundancies. Linear estimates expressed in the decorrelated basis become computationally efficient.

Geometrical ideas may be introduced by thinking of the space of random variables under consideration as a *linear vector space* [7]. For example, in the previous section we dealt with the multicomponent random variables $\mathbf{x}$ and $\mathbf{y}$ consisting, say, of the random variables $\{x_1, x_2, \ldots, x_N\}$ and $\{y_1, y_2, \ldots, y_M\}$, respectively. In this case, the space of random variables under consideration is the set

$$\{x_1, x_2, \ldots, x_N,\ y_1, y_2, \ldots, y_M\} \tag{1.6.1}$$

Since any linear combination of random variables from this set is itself a random variable, the above set may be enlarged by adjoining to it all such possible linear combinations. This is the linear vector space *generated* or spanned by the given set of random variables. The next step is to convert this vector space into an *inner-product space* (a Hilbert space) by defining an inner product between any two random variables u and v as follows:

$$(u, v) = E[uv] \tag{1.6.2}$$

With this definition of an inner product, "orthogonal" means "uncorrelated." The *distance* between u and v is defined by the norm $\|u - v\|$ induced by the above inner product:

$$\|u - v\|^2 = E\big[(u - v)^2\big] \tag{1.6.3}$$

Mutually orthogonal (i.e., uncorrelated) random variables may be used to define *orthogonal bases*. Consider, for example, M mutually orthogonal random variables $\{\epsilon_1, \epsilon_2, \ldots, \epsilon_M\}$, such that

$$(\epsilon_i, \epsilon_j) = E[\epsilon_i \epsilon_j] = 0\,, \quad \text{if } i \neq j \tag{1.6.4}$$

and let $Y = \{\epsilon_1, \epsilon_2, \ldots, \epsilon_M\}$ be the linear subspace *spanned* by these M random variables. Without loss of generality, we may assume that the ϵ_is are linearly independent; therefore, they form a linearly independent and orthogonal basis for the subspace Y.

One of the standard results on linear vector spaces is the *orthogonal decomposition theorem* [8], which in our context may be stated as follows: Any random variable x may be decomposed uniquely, with respect to a subspace Y, into two mutually orthogonal parts. One part is *parallel* to the subspace Y (i.e., it lies in it), and the other is *perpendicular* to it. That is,

$$x = \hat{x} + e \qquad \text{with } \hat{x} \in Y \text{ and } e \perp Y \tag{1.6.5}$$

The component $\hat{x}$ is called the *orthogonal projection* of x onto the subspace Y. This decomposition is depicted in Fig. 1.6.1. The orthogonality condition $e \perp Y$ means that e must be orthogonal to every vector in Y; or equivalently, to every basis vector ϵ_i,

$$(e, \epsilon_i) = E[e\epsilon_i] = 0\,, \quad i = 1, 2, \ldots, M \tag{1.6.6}$$

Since the component $\hat{x}$ lies in Y, it may be expanded in terms of the orthogonal basis in the form

$$\hat{x} = \sum_{i=1}^{M} a_i \epsilon_i$$

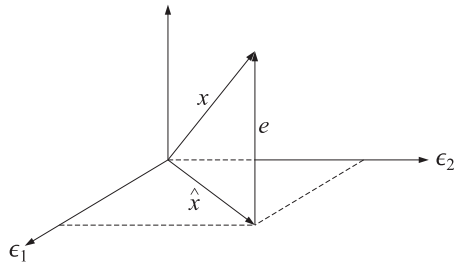


Fig. 1.6.1 Orthogonal decomposition with respect to $Y = \{\epsilon_1, \epsilon_2\}$.

The coefficients a_i can be determined using the orthogonality equations (1.6.6), as follows,

$$(x, \epsilon_i) = (\hat{x} + e, \epsilon_i) = (\hat{x}, \epsilon_i) + (e, \epsilon_i) = (\hat{x}, \epsilon_i)$$

$$= \left(\sum_{j=1}^{M} a_j \epsilon_j,\ \epsilon_i \right) = \sum_{j=1}^{M} a_j (\epsilon_j, \epsilon_i) = a_i (\epsilon_i, \epsilon_i)$$

where in the last equality we used Eq. (1.6.4). Thus, $a_i = (x, \epsilon_i)\,(\epsilon_i, \epsilon_i)^{-1}$. or, $a_i = E[x\epsilon_i]E[\epsilon_i \epsilon_i]^{-1}$, and we can write Eq. (1.6.5) as

$$x = \hat{x} + e = \sum_{i=1}^{M} E[x\epsilon_i]E[\epsilon_i \epsilon_i]^{-1} \epsilon_i + e \tag{1.6.7}$$

Eq. (1.6.7) may also be written in a compact matrix form by introducing the M-vector,

$$\boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_M \end{bmatrix}$$

the corresponding cross-correlation M-vector,

$$E[x\boldsymbol{\epsilon}] = \begin{bmatrix} E[x\epsilon_1] \\ E[x\epsilon_2] \\ \vdots \\ E[x\epsilon_M] \end{bmatrix}$$

and the correlation matrix $R_{\epsilon\epsilon} = E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T]$, which is *diagonal* because of Eq. (1.6.4):

$$R_{\epsilon\epsilon} = E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T] = \operatorname{diag}\{E[\epsilon_1^2], E[\epsilon_2^2], \ldots, E[\epsilon_M^2]\}$$

Then, Eq. (1.6.7) may be written as

$$x = \hat{x} + e = E[x\boldsymbol{\epsilon}^T]E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T]^{-1}\boldsymbol{\epsilon} + e \tag{1.6.8}$$

The orthogonality equations (1.6.6) can be written as

$$R_{e\epsilon} = E[e\boldsymbol{\epsilon}^T] = 0 \tag{1.6.9}$$

Equations (1.6.8) and (1.6.9) represent the unique orthogonal decomposition of any random variable x relative to a linear subspace Y of random variables. If one has a collection of N random variables $\{x_1, x_2, \ldots, x_N\}$, then each one may be orthogonally decomposed with respect to the same subspace Y, giving $x_i = \hat{x}_i + e_i$, $i = 1, 2, \ldots, N$. These may be grouped together into a compact matrix form as

$$\mathbf{x} = \hat{\mathbf{x}} + \mathbf{e} = E[\mathbf{x}\boldsymbol{\epsilon}^T]E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T]^{-1}\boldsymbol{\epsilon} + \mathbf{e} \tag{1.6.10}$$

where $\mathbf{x}$ stands for the column N-vector $\mathbf{x} = [x_1, x_2, \ldots, x_N]^T$, and so on. This is identical to the correlation canceler decomposition of the previous section.

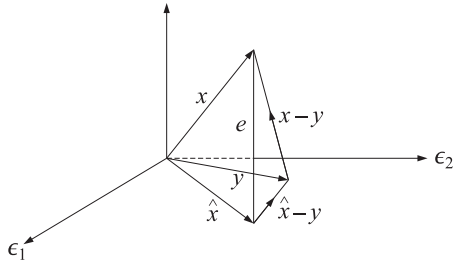
Next, we briefly discuss the *orthogonal projection theorem*. In Sec. 1.4, we noted the best linear estimator property of the correlation canceler decomposition. The same result may be understood geometrically by means of the orthogonal projection theorem, which states: The orthogonal projection $\hat{x}$ of a vector x onto a linear subspace Y is that vector in Y that lies closest to x with respect to the distance induced by the inner product of the vector space.

The theorem is a simple consequence of the orthogonal decomposition theorem and the Pythagorean theorem. Indeed, let $x = \hat{x} + e$ be the unique orthogonal decomposition of x with respect to Y, so that $\hat{x} \in Y$ and $e \perp Y$ and let y be an arbitrary vector in Y; noting that $(\hat{x} - y) \in Y$ and therefore $e \perp (\hat{x} - y)$, we have

$$\|x - y\|^2 = \|(\hat{x} - y) + e\|^2 = \|\hat{x} - y\|^2 + \|e\|^2$$

or, in terms of Eq. (1.6.3),

$$E[(x - y)^2] = E[(\hat{x} - y)^2] + E[e^2]$$



Since the vector y varies over the subspace Y, it follows that the above quantity will be minimized when $y = \hat{x}$. In summary, $\hat{x}$ represents the best approximation of x that can be made as a linear function of the random variables in Y in the minimum mean-square sense.

Above, we developed the orthogonal decomposition of a random variable relative to a linear subspace Y which was generated by means of an orthogonal basis $\epsilon_1, \epsilon_2, \ldots, \epsilon_M$. In practice, the subspace Y is almost always defined by means of a nonorthogonal basis, such as a collection of random variables

$$Y = \{y_1, y_2, \ldots, y_M\}$$

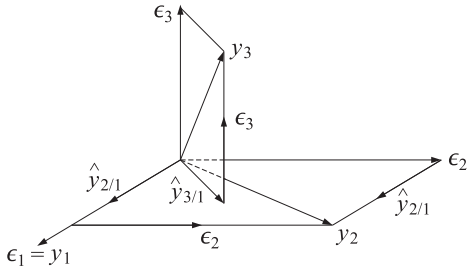
which may be mutually correlated. The subspace Y is defined again as the linear *span* of this basis. The *Gram-Schmidt orthogonalization* process is a recursive procedure of generating an orthogonal basis $\{\epsilon_1, \epsilon_2, \ldots, \epsilon_M\}$ from $\{y_1, y_2, \ldots, y_M\}$.

The basic idea of the method is this: Initialize the procedure by selecting $\epsilon_1 = y_1$. Next, consider y_2 and decompose it relative to ϵ_1. Then, the component of y_2 which is perpendicular to ϵ_1 is selected as ϵ_2, so that $(\epsilon_1, \epsilon_2) = 0$. Next, take y_3 and decompose it relative to the subspace spanned by $\{\epsilon_1, \epsilon_2\}$ and take the corresponding perpendicular component to be ϵ_3, and so on. For example, the first three steps of the procedure are

$$\epsilon_1 = y_1$$

$$\epsilon_2 = y_2 - E[y_2\epsilon_1]E[\epsilon_1\epsilon_1]^{-1}\epsilon_1$$

$$\epsilon_3 = y_3 - E[y_3\epsilon_1]E[\epsilon_1\epsilon_1]^{-1}\epsilon_1 - E[y_3\epsilon_2]E[\epsilon_2\epsilon_2]^{-1}\epsilon_2$$



At the nth iteration step

$$\epsilon_n = y_n - \sum_{i=1}^{n-1} E[y_n\epsilon_i]E[\epsilon_i\epsilon_i]^{-1}\epsilon_i\,, \quad n = 2, 3, \ldots, M \tag{1.6.11}$$

The basis $\{\epsilon_1, \epsilon_2, \ldots, \epsilon_M\}$ generated in this way is orthogonal by construction. The Gram-Schmidt process may be understood in terms of the hierarchy of subspaces:

$$\begin{aligned} Y_1 &= \{\epsilon_1\} = \{y_1\} \\ Y_2 &= \{\epsilon_1, \epsilon_2\} = \{y_1, y_2\} \\ Y_3 &= \{\epsilon_1, \epsilon_2, \epsilon_3\} = \{y_1, y_2, y_3\} \\ &\vdots \\ Y_n &= \{\epsilon_1, \epsilon_2, \ldots, \epsilon_n\} = \{y_1, y_2, \ldots, y_n\} \end{aligned}$$

for $n = 1, 2, \ldots, M$, where each is a subspace of the next one and differs from the next by the addition of one more basis vector. The second term in Eq. (1.6.11) may be recognized now as the component of y_n parallel to the subspace Y_{n-1}. We may denote this as

$$\hat{y}_{n/n-1} = \sum_{i=1}^{n-1} E[y_n\epsilon_i]E[\epsilon_i\epsilon_i]^{-1}\epsilon_i \tag{1.6.12}$$

Then, Eq. (1.6.11) may be written as

$$\epsilon_n = y_n - \hat{y}_{n/n-1} \quad \text{or} \quad y_n = \hat{y}_{n/n-1} + \epsilon_n \tag{1.6.13}$$

which represents the orthogonal decomposition of y_n relative to the subspace Y_{n-1}. Since, the term $\hat{y}_{n/n-1}$ already lies in Y_{n-1}, we have the direct sum decomposition

$$Y_n = Y_{n-1} \oplus \{y_n\} = Y_{n-1} \oplus \{\epsilon_n\}$$

Introducing the notation

$$b_{ni} = E[y_n \epsilon_i] E[\epsilon_i \epsilon_i]^{-1}, \quad 1 \le i \le n-1 \tag{1.6.14}$$

and $b_{nn} = 1$, we may write Eq. (1.6.13) in the form

$$y_n = \sum_{i=1}^{n} b_{ni} \epsilon_i = \epsilon_n + \sum_{i=1}^{n-1} b_{ni} \epsilon_i = \epsilon_n + \hat{y}_{n/n-1} \tag{1.6.15}$$

for $1 \le n \le M$. And in matrix form,

$$\mathbf{y} = B\boldsymbol{\epsilon}, \quad \text{where} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{bmatrix}, \quad \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_M \end{bmatrix} \tag{1.6.16}$$

and B is a *lower-triangular* matrix with matrix elements given by (1.6.14). Its main diagonal is unity. For example, for $M = 4$ we have

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ b_{21} & 1 & 0 & 0 \\ b_{31} & b_{32} & 1 & 0 \\ b_{41} & b_{42} & b_{43} & 1 \end{bmatrix} \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \end{bmatrix}$$

Both the matrix B and its inverse B^{-1} are unit lower-triangular matrices. The information contained in the two bases $\mathbf{y}$ and $\boldsymbol{\epsilon}$ is the same. Going from the basis $\mathbf{y}$ to the basis $\boldsymbol{\epsilon}$ removes all the redundant correlations that may exist in $\mathbf{y}$ and "distills" the essential information contained in $\mathbf{y}$ to its most basic form. Because the basis $\boldsymbol{\epsilon}$ is uncorrelated, every basis vector ϵ_i, $i = 1, 2, \ldots, M$ will represent something different, or new. Therefore, the random variables ϵ_i are sometimes called the *innovations*, and the representation (1.6.16) of $\mathbf{y}$ in terms of $\boldsymbol{\epsilon}$, the *innovations representation*.

Since the correlation matrix $R_{\epsilon\epsilon} = E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T]$ is diagonal, the transformation (1.6.16) corresponds to an LU (lower-upper) *Cholesky factorization* of the correlation matrix of $\mathbf{y}$, that is,

$$R_{yy} = E[\mathbf{y}\mathbf{y}^T] = BE[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T]B^T = BR_{\epsilon\epsilon}B^T \tag{1.6.17}$$

We note also the invariance of the projected vector $\hat{\mathbf{x}}$ of Eq. (1.6.10) under such linear change of basis:

$$\hat{\mathbf{x}} = E[\mathbf{x}\boldsymbol{\epsilon}^T] E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T]^{-1} \boldsymbol{\epsilon} = E[\mathbf{x}\mathbf{y}^T] E[\mathbf{y}\mathbf{y}^T]^{-1} \mathbf{y} \tag{1.6.18}$$

This shows the equivalence of the orthogonal decompositions (1.6.10) to the correlation canceler decompositions (1.4.1). The computational efficiency of the $\boldsymbol{\epsilon}$ basis over the $\mathbf{y}$ basis is evident from the fact that the covariance matrix $E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T]$ is diagonal, and

therefore, its inverse is trivially computed. We may also apply the property (1.6.18) to $\mathbf{y}$ itself. Defining the vectors

$$\boldsymbol{\epsilon}_{n-1} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_{n-1} \end{bmatrix} \qquad \mathbf{y}_{n-1} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n-1} \end{bmatrix}$$

we may write the projection $\hat{y}_{n/n-1}$ of y_n on the subspace Y_{n-1} given by Eq. (1.6.12) as follows:

$$\hat{y}_{n/n-1} = E[y_n \boldsymbol{\epsilon}_{n-1}^T] E[\boldsymbol{\epsilon}_{n-1} \boldsymbol{\epsilon}_{n-1}^T]^{-1} \boldsymbol{\epsilon}_{n-1} = E[y_n \mathbf{y}_{n-1}^T] E[\mathbf{y}_{n-1} \mathbf{y}_{n-1}^T]^{-1} \mathbf{y}_{n-1} \tag{1.6.19}$$

Eq. (1.6.13) is then written as

$$\epsilon_n = y_n - \hat{y}_{n/n-1} = y_n - E[y_n \mathbf{y}_{n-1}^T] E[\mathbf{y}_{n-1} \mathbf{y}_{n-1}^T]^{-1} \mathbf{y}_{n-1} \tag{1.6.20}$$

which provides a construction of ϵ_n directly in terms of the y_ns. We note that the quantity $\hat{y}_{n/n-1}$ is also the *best linear estimate* of y_n that can be made on the basis of the *previous* y_ns, $Y_{n-1} = \{y_1, y_2, \ldots, y_{n-1}\}$. If the index n represents the time index, as it does for random signals, then $\hat{y}_{n/n-1}$ is the best *linear prediction* of y_n on the basis of its past; and ϵ_n is the corresponding prediction error.

The Gram-Schmidt process was started with the first element y_1 of $\mathbf{y}$ and proceeded forward to y_M. The process can just as well be started with y_M and proceed backward to y_1 (see Problem 1.15). It may be interpreted as *backward prediction*, or postdiction, and leads to the UL (rather than LU) factorization of the covariance matrix R_{yy}. In Sec. 1.8, we study the properties of such forward and backward orthogonalization procedures in some detail.

Example 1.6.1: Consider the three zero-mean random variables $\{y_1, y_2, y_3\}$ and let $R_{ij} = E[y_i y_j]$ for $i, j = 1, 2, 3$, denote their correlation matrix. Then, the explicit construction indicated in Eq. (1.6.20) can be carried out as follows. The required vectors $\mathbf{y}_{n-1}$ are:

$$\mathbf{y}_1 = [y_1], \quad \mathbf{y}_2 = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

and hence

$$E[y_2 \mathbf{y}_1^T] = E[y_2 y_1] = R_{21}$$

$$E[\mathbf{y}_1 \mathbf{y}_1^T] = E[y_1 y_1] = R_{11}$$

$$E[y_3 \mathbf{y}_2^T] = E[y_3 [y_1, y_2]] = [R_{31}, R_{32}]$$

$$E[\mathbf{y}_2 \mathbf{y}_2^T] = E\left[\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} [y_1, y_2] \right] = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}$$

Therefore, Eq. (1.6.20) becomes

$$\epsilon_1 = y_1$$

$$\epsilon_2 = y_2 - \hat{y}_{2/1} = y_2 - R_{21} R_{11}^{-1} y_1$$

$$\epsilon_3 = y_3 - \hat{y}_{3/2} = y_3 - [R_{31}, R_{32}] \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}^{-1} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

Example 1.6.2: The zero-mean random vector $\mathbf{y} = [y_1, y_2, y_3]^T$ has covariance matrix

$$R_{yy} = \begin{bmatrix} 1 & -1 & 1 \\ -1 & 3 & 3 \\ 1 & 3 & 12 \end{bmatrix}$$

Determine the innovations representation of $\mathbf{y}$ in two ways: using the Gram- Schmidt construction and using the results of Example 1.6.1.

Solution: Starting with $\epsilon_1 = y_1$, we find $E[y_2\epsilon_1] = R_{21} = -1$ and $E[\epsilon_1^2] = R_{11} = 1$. Therefore,

$$\epsilon_2 = y_2 - E[y_2\epsilon_1]E[\epsilon_1^2]^{-1}\epsilon_1 = y_2 + \epsilon_1 = y_2 + y_1$$

with a mean-square value $E[\epsilon_2^2] = E[y_2^2] + 2E[y_2y_1] + E[y_1^2] = 3 - 2 + 1 = 2$. Similarly, we find $E[y_3\epsilon_1] = R_{31} = 1$ and

$$E[y_3\epsilon_2] = E[y_3(y_2 + y_1)] = R_{32} + R_{31} = 3 + 1 = 4$$

Thus,

$$\epsilon_3 = y_3 - E[y_3\epsilon_1]E[\epsilon_1\epsilon_1]^{-1}\epsilon_1 - E[y_3\epsilon_2]E[\epsilon_2\epsilon_2]^{-1}\epsilon_2 = y_3 - \epsilon_1 - 2\epsilon_2$$

or,

$$\epsilon_3 = y_3 - y_1 - 2(y_2 + y_1) = y_3 - 2y_2 - 3y_1$$

Solving for the ys and writing the answer in matrix form we have

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 1 & 2 & 1 \end{bmatrix} \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{bmatrix} = B\boldsymbol{\epsilon}$$

The last row determines $E[\epsilon_3^2]$. Using the mutual orthogonality of the ϵ_is, we have

$$E[y_3^2] = E[(\epsilon_3 + 2\epsilon_2 + \epsilon_1)^2] = E[\epsilon_3^2] + 4E[\epsilon_2^2] + E[\epsilon_1^2] \quad \Rightarrow \quad 12 = E[\epsilon_3^2] + 8 + 1$$

which gives $E[\epsilon_3^2] = 3$. Using the results of Example 1.6.1, we have

$$\epsilon_3 = y_3 - [R_{31}, R_{32}] \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}^{-1} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = y_3 - [1, 3] \begin{bmatrix} 1 & -1 \\ -1 & 3 \end{bmatrix}^{-1} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

The indicated matrix operations are computed easily and lead to the same expression for ϵ_3 found above. □

The innovations representation Eq. (1.6.16) and the Cholesky factorization (1.6.17) are also very useful for the purpose of simulating a random vector having a prescribed covariance matrix. The procedure is as follows: given $R = E[\mathbf{y}\mathbf{y}^T]$, find its Cholesky factor B and the diagonal matrix $R_{\epsilon\epsilon}$; then, using any standard random number generator, generate M independent random numbers $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, \ldots, \epsilon_M]^T$ of mean zero and variances equal to the diagonal entries of $R_{\epsilon\epsilon}$, and perform the matrix operation $\mathbf{y} = B\boldsymbol{\epsilon}$ to obtain a realization of the random vector $\mathbf{y}$.

Conversely, if a number of independent realizations of $\mathbf{y}$ are available, $\{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N\}$, we may form an estimate of the covariance matrix by the following expression, referred to as the *sample covariance matrix*

$$\hat{R} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{y}_n\mathbf{y}_n^T \tag{1.6.21}$$

Example 1.6.3: In typical array processing applications, a linear array of, say, M equally spaced sensors measures the incident radiation field. This field may consist of a number of plane waves incident from different angles on the array plus background noise. The objective is to determine the number, angles of arrival, and strengths of the incident plane waves from measurements of the field at the sensor elements. At each time instant, the measurements at the M sensors may be assembled into the M-dimensional random vector $\mathbf{y}$, called an instantaneous *snapshot*. Thus, the correlation matrix $R = E[\mathbf{y}\mathbf{y}^T]$ measures the correlations that exist among sensors, that is, *spatial correlations*. In Chap. 14, we will consider methods of extracting the angle-of-arrival information from the covariance matrix R. Most of these methods require an estimate of the covariance matrix, which is typically given by Eq. (1.6.21) on the basis of N snapshots. □

How good an estimate of R is $\hat{R}$? First, note that it is an *unbiased* estimate:

$$E[\hat{R}] = \frac{1}{N}\sum_{n=1}^{N} E[\mathbf{y}_n\mathbf{y}_n^T] = \frac{1}{N}(NR) = R$$

Second, we show that it is *consistent*. The correlation between the various matrix elements of $\hat{R}$ is obtained as follows:

$$E[\hat{R}_{ij}\hat{R}_{kl}] = \frac{1}{N^2}\sum_{n=1}^{N}\sum_{m=1}^{N} E[y_{ni}y_{nj}y_{mk}y_{ml}]$$

where y_{ni} is the ith component of the nth vector $\mathbf{y}_n$. To get a simple expression for the covariance of $\hat{R}$, we will assume that $\mathbf{y}_n$, $n = 1, 2, \ldots, N$ are independent zero-mean gaussian random vectors of covariance matrix R. This implies that [4,5]

$$E[y_{ni}y_{nj}y_{mk}y_{ml}] = R_{ij}R_{kl} + \delta_{nm}(R_{ik}R_{jl} + R_{il}R_{jk})$$

It follows that

$$E[\hat{R}_{ij}\hat{R}_{kl}] = R_{ij}R_{jk} + \frac{1}{N}(R_{ik}R_{jl} + R_{il}R_{jk}) \tag{1.6.22}$$

Writing $\Delta R = \hat{R} - E[\hat{R}] = \hat{R} - R$, we obtain for the covariance

$$E[\Delta R_{ij}\Delta R_{kl}] = \frac{1}{N}(R_{ik}R_{jl} + R_{il}R_{jk}) \tag{1.6.23}$$

Thus, $\hat{R}$ is a consistent estimator. The result of Eq. (1.6.23) is typical of the asymptotic results that are available in the statistical literature [4,5]. It will be used in Chap. 14 to obtain asymptotic results for linear prediction parameters and for the eigenstructure methods of spectrum estimation.

The sample covariance matrix (1.6.21) may also be written in an *adaptive*, or recursive form,

$$\hat{R}_N = \frac{1}{N}\sum_{n=1}^{N} \mathbf{y}_n\mathbf{y}_n^T = \frac{1}{N}\left[\sum_{n=1}^{N-1} \mathbf{y}_n\mathbf{y}_n^T + \mathbf{y}_N\mathbf{y}_N^T\right] = \frac{1}{N}\left[(N-1)\hat{R}_{N-1} + \mathbf{y}_N\mathbf{y}_N^T\right]$$

where we wrote $\hat{R}_N$ to explicitly indicate the dependence on N. A more intuitive way of writing this recursion is in the "predictor/corrector" form

$$\hat{R}_N = \hat{R}_{N-1} + \frac{1}{N}(\mathbf{y}_N\mathbf{y}_N^T - \hat{R}_{N-1}) \tag{1.6.24}$$

The term $\hat{R}_{N-1}$ may be thought of as a prediction of R based on $N-1$ observations, the Nth observation $\mathbf{y}_N\mathbf{y}_N^T$ may be thought of as an instantaneous estimate of R, and the term in the parenthesis as the prediction error that is used to correct the prediction. The function **sampcov** takes as input the old matrix $\hat{R}_{N-1}$, and the new observation $\mathbf{y}_N$, and outputs the updated matrix $\hat{R}_N$, overwriting the old one.

Example 1.6.4: Consider the 3×3 random vector $\mathbf{y}$ defined in Example 1.6.2. Using the innovations representation of $\mathbf{y}$, generate $N = 200$ independent vectors $\mathbf{y}_n$, $n = 1, 2, \ldots, N$ and then compute the estimated sample covariance matrix (1.6.21) and compare it with the theoretical R. Compute the sample covariance matrix $\hat{R}$ recursively and plot its matrix elements as functions of the iteration number N.

Solution: Generate N independent 3-vectors $\boldsymbol{\epsilon}_n$, and compute $\mathbf{y}_n = B\boldsymbol{\epsilon}_n$. The estimated and theoretical covariance matrices are

$$\hat{R} = \begin{bmatrix} 0.995 & -1.090 & 0.880 \\ -1.090 & 3.102 & 2.858 \\ 0.880 & 2.858 & 11.457 \end{bmatrix}, \quad R = \begin{bmatrix} 1 & -1 & 1 \\ -1 & 3 & 3 \\ 1 & 3 & 12 \end{bmatrix}$$

Can we claim that this is a good estimate of R? Yes, because the deviations from R are consistent with the expected deviations given by Eq. (1.6.23). The standard deviation of the ijth matrix element is

$$\delta R_{ij} = \sqrt{E\big[(\Delta R_{ij})^2\big]} = \sqrt{(R_{ii}R_{jj} + R_{ij}^2)/N}$$

The estimated values $\hat{R}_{ij}$ fall within the intervals $R_{ij} - \delta R_{ij} \le \hat{R}_{ij} \le R_{ij} + \delta R_{ij}$, as can be verified by inspecting the matrices

$$R - \delta R = \begin{bmatrix} 0.901 & -1.146 & 0.754 \\ -1.146 & 2.691 & 2.534 \\ 0.754 & 2.534 & 10.857 \end{bmatrix}, \quad R + \delta R = \begin{bmatrix} 1.099 & -0.854 & 1.246 \\ -0.854 & 3.309 & 3.466 \\ 1.246 & 3.466 & 13.143 \end{bmatrix}$$

The recursive computation Eq. (1.6.24), implemented by successive calls to the function **sampcov**, is shown in Fig. 1.6.2, where only the matrix elements R_{11}, R_{12}, and R_{22} are plotted versus N. Such graphs give us a better idea of how fast the sample estimate $\hat{R}_N$ converges to the theoretical R. □

1.7 Partial Correlations

A concept intimately connected to the Gram-Schmidt orthogonalization is that of the partial correlation. It plays a central role in linear prediction applications.

Consider the Gram-Schmidt orthogonalization of a random vector $\mathbf{y}$ in the form $\mathbf{y} = B\boldsymbol{\epsilon}$, where B is a unit lower-triangular matrix, and $\boldsymbol{\epsilon}$ is a vector of mutually uncorrelated components. Inverting, we have

$$\boldsymbol{\epsilon} = A\mathbf{y} \tag{1.7.1}$$

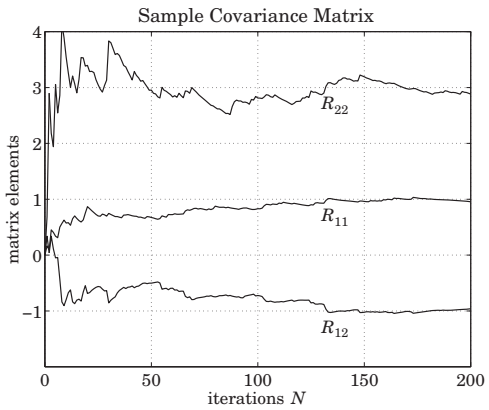


Fig. 1.6.2 Recursive computation of the sample covariance matrix.

where $A = B^{-1}$. Now, suppose the vector $\mathbf{y}$ is arbitrarily subdivided into three subvectors as follows:

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_0 \\ \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}$$

where $\mathbf{y}_0, \mathbf{y}_1, \mathbf{y}_2$ do not necessarily have the same dimension. Then, the matrix equation (1.7.1) may also be decomposed in a block-compatible form:

$$\begin{bmatrix} \boldsymbol{\epsilon}_0 \\ \boldsymbol{\epsilon}_1 \\ \boldsymbol{\epsilon}_2 \end{bmatrix} = \begin{bmatrix} A_{00} & 0 & 0 \\ A_{11} & A_{10} & 0 \\ A_{22} & A_{21} & A_{20} \end{bmatrix} \begin{bmatrix} \mathbf{y}_0 \\ \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix} \tag{1.7.2}$$

where A_{00}, A_{10}, A_{20} are unit lower-triangular matrices. Since $\mathbf{y}$ has components that are generally correlated with each other, it follows that $\mathbf{y}_0$ will be correlated with $\mathbf{y}_1$, and $\mathbf{y}_1$ will be correlated with $\mathbf{y}_2$. Thus, through the intermediate action of $\mathbf{y}_1$, the vector $\mathbf{y}_0$ will be indirectly coupled with the vector $\mathbf{y}_2$. The question we would like to ask is this: Suppose the effect of the intermediate vector $\mathbf{y}_1$ were to be removed, then what would be the correlation that is left between $\mathbf{y}_0$ and $\mathbf{y}_2$? This is the *partial correlation*. It represents the "true" or "direct" influence of $\mathbf{y}_0$ on $\mathbf{y}_2$, when the indirect influence via $\mathbf{y}_1$ is removed. To remove the effect of $\mathbf{y}_1$, we project both $\mathbf{y}_0$ and $\mathbf{y}_2$ on the subspace spanned by $\mathbf{y}_1$ and then subtract these parts from both, that is, let

$$\mathbf{e}_0 = \mathbf{y}_0 - (\text{projection of } \mathbf{y}_0 \text{ on } \mathbf{y}_1)$$

$$\mathbf{e}_2 = \mathbf{y}_2 - (\text{projection of } \mathbf{y}_2 \text{ on } \mathbf{y}_1)$$

or,

$$\begin{aligned} \mathbf{e}_0 &= \mathbf{y}_0 - R_{01}R_{11}^{-1}\mathbf{y}_1 \\ \mathbf{e}_2 &= \mathbf{y}_2 - R_{21}R_{11}^{-1}\mathbf{y}_1 \end{aligned} \tag{1.7.3}$$

where we defined $R_{ij} = E[\mathbf{y}_i\mathbf{y}_j^T]$, for $i,j = 0,1,2$. We define the *partial correlation* (PARCOR) *coefficient* between $\mathbf{y}_0$ and $\mathbf{y}_2$, with the effect of the intermediate $\mathbf{y}_1$ removed, as follows:

$$\Gamma = E[\mathbf{e}_2\mathbf{e}_0^T]E[\mathbf{e}_0\mathbf{e}_0^T]^{-1} \tag{1.7.4}$$

Then, Γ may be expressed in terms of the entries of the matrix A as follows:

$$\Gamma = -A_{20}^{-1}A_{22} \tag{1.7.5}$$

To prove this result, we consider the last equation of (1.7.2):

$$\boldsymbol{\epsilon}_2 = A_{22}\mathbf{y}_0 + A_{21}\mathbf{y}_1 + A_{20}\mathbf{y}_2 \tag{1.7.6}$$

By construction, $\boldsymbol{\epsilon}_2$ is orthogonal to $\mathbf{y}_1$, so that $E[\boldsymbol{\epsilon}_2\mathbf{y}_1^T] = 0$. Thus we obtain the relationship:

$$\begin{aligned} E[\boldsymbol{\epsilon}_2\mathbf{y}_1^T] &= A_{22}E[\mathbf{y}_0\mathbf{y}_1^T] + A_{21}E[\mathbf{y}_1\mathbf{y}_1^T] + A_{20}E[\mathbf{y}_2\mathbf{y}_1^T] \\ &= A_{22}R_{01} + A_{21}R_{11} + A_{20}R_{21} = 0 \end{aligned} \tag{1.7.7}$$

Using Eqs. (1.7.3) and (1.7.7), we may express $\boldsymbol{\epsilon}_2$ in terms of $\mathbf{e}_0$ and $\mathbf{e}_2$, as follows:

$$\begin{aligned} \boldsymbol{\epsilon}_2 &= A_{22}(\mathbf{e}_0 + R_{01}R_{11}^{-1}\mathbf{y}_1) + A_{21}\mathbf{y}_1 + A_{20}(\mathbf{e}_2 + R_{21}R_{11}^{-1}\mathbf{y}_1) \\ &= A_{22}\mathbf{e}_0 + A_{20}\mathbf{e}_2 + (A_{22}R_{01} + A_{21}R_{11} + A_{20}R_{21})R_{11}^{-1}\mathbf{y}_1 \\ &= A_{22}\mathbf{e}_0 + A_{20}\mathbf{e}_2 \end{aligned} \tag{1.7.8}$$

Now, by construction, $\boldsymbol{\epsilon}_2$ is orthogonal to both $\mathbf{y}_0$ and $\mathbf{y}_1$, and hence also to $\mathbf{e}_0$, that is, $E[\boldsymbol{\epsilon}_2\mathbf{e}_0^T] = 0$. Using Eq. (1.7.8) we obtain

$$E[\boldsymbol{\epsilon}_2\mathbf{e}_0^T] = A_{22}E[\mathbf{e}_0\mathbf{e}_0^T] + A_{20}E[\mathbf{e}_2\mathbf{e}_0^T] = 0$$

from which (1.7.5) follows. It is interesting also to note that (1.7.8) may be written as

$$\boldsymbol{\epsilon}_2 = A_{20}\mathbf{e}$$

where $\mathbf{e} = \mathbf{e}_2 - \Gamma\mathbf{e}_0$ is the orthogonal complement of $\mathbf{e}_2$ relative to $\mathbf{e}_0$.

Example 1.7.1: An important special case of Eq. (1.7.5) is when $\mathbf{y}_0$ and $\mathbf{y}_2$ are selected as the first and last components of $\mathbf{y}$, and therefore $\mathbf{y}_1$ consists of all the intermediate components. For example, suppose $\mathbf{y} = [y_0, y_1, y_2, y_3, y_4]^T$. Then, the decomposition (1.7.2) can be written as follows:

$$\left[\begin{array}{c} \epsilon_0 \\ \hline \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \hline \epsilon_4 \end{array}\right] = \left[\begin{array}{c|cccc} 1 & 0 & 0 & 0 & 0 \\ \hline a_{11} & 1 & 0 & 0 & 0 \\ a_{22} & a_{21} & 1 & 0 & 0 \\ a_{33} & a_{32} & a_{31} & 1 & 0 \\ \hline a_{44} & a_{43} & a_{42} & a_{41} & 1 \end{array}\right] \left[\begin{array}{c} y_0 \\ \hline y_1 \\ y_2 \\ y_3 \\ \hline y_4 \end{array}\right] \tag{1.7.9}$$

where $\mathbf{y}_0, \mathbf{y}_1, \mathbf{y}_2$ are chosen as the vectors

$$\mathbf{y}_0 = [y_0], \quad \mathbf{y}_1 = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}, \quad \mathbf{y}_2 = [y_4]$$

The matrices A_{20} and A_{22} are in this case the scalars $A_{20} = [1]$ and $A_{22} = [a_{44}]$. Therefore, the corresponding PARCOR coefficient (1.7.5) is

$$\Gamma = -a_{44}$$

Clearly, the first column $[1, a_{11}, a_{22}, a_{33}, a_{44}]$ of A contains all the lower order PARCOR coefficients, that is, the quantity

$$\gamma_p = -a_{pp}, \quad p = 1,2,3,4$$

represents the partial correlation coefficient between y_0 and y_p, with the effect of all the intermediate variables $y_1, y_2, \ldots, y_{p-1}$ removed. □

We note the backward indexing of the entries of the matrix A in Eqs. (1.7.2) and (1.7.9). It corresponds to writing ϵ_n in a convolutional form

$$\epsilon_n = \sum_{i=0}^{n} a_{ni}y_{n-i} = \sum_{i=0}^{n} a_{n,n-i}y_i = y_n + a_{n1}y_{n-1} + a_{n2}y_{n-2} + \cdots + a_{nn}y_0 \tag{1.7.10}$$

and conforms to standard notation in linear prediction applications. Comparing (1.7.10) with (1.6.13), we note that the projection of y_n onto the subspace Y_{n-1} may also be expressed directly in terms of the correlated basis $Y_{n-1} = \{y_0, y_1, \ldots, y_{n-1}\}$ as follows:

$$\hat{y}_{n/n-1} = -[a_{n1}y_{n-1} + a_{n2}y_{n-2} + \cdots + a_{nn}y_0] \tag{1.7.11}$$

An alternative expression was given in Eq. (1.6.19). Writing Eq. (1.7.10) in vector form, we have

$$\epsilon_n = [a_{nn}, \ldots, a_{n1}, 1]\begin{bmatrix} y_0 \\ \vdots \\ y_{n-1} \\ y_n \end{bmatrix} = [1, a_{n1}, \ldots, a_{nn}]\begin{bmatrix} y_n \\ y_{n-1} \\ \vdots \\ y_0 \end{bmatrix} \tag{1.7.12}$$

Thus, there are two possible definitions for the data vector $\mathbf{y}$ and corresponding weight vector $\mathbf{a}$. According to the first definition—which is what we used in Eqs. (1.7.1) and (1.7.9)—the vector $\mathbf{y}$ is indexed from the lowest to the highest index and the vector $\mathbf{a}$ is indexed in the reverse way. According to the second definition, $\mathbf{y}$ and $\mathbf{a}$ are exactly the *reverse*, or upside-down, versions of the first definition, namely, $\mathbf{y}$ is indexed backward from high to low, whereas $\mathbf{a}$ is indexed forward. If we use the second definition and write Eq. (1.7.12) in matrix form, we obtain the reverse of Eq. (1.7.9), that is

$$\boldsymbol{\epsilon}_{\text{rev}} = \begin{bmatrix} \epsilon_4 \\ \epsilon_3 \\ \epsilon_2 \\ \epsilon_1 \\ \epsilon_0 \end{bmatrix} = \begin{bmatrix} 1 & a_{41} & a_{42} & a_{43} & a_{44} \\ 0 & 1 & a_{31} & a_{32} & a_{33} \\ 0 & 0 & 1 & a_{21} & a_{22} \\ 0 & 0 & 0 & 1 & a_{11} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_4 \\ y_3 \\ y_2 \\ y_1 \\ y_0 \end{bmatrix} = U\mathbf{y}_{\text{rev}} \tag{1.7.13}$$

Thus, the transformation between the correlated and decorrelated bases is now by means of a unit *upper-triangular* matrix U. It corresponds to the UL (rather than LU) factorization of the covariance matrix of the reversed vector $\mathbf{y}_{\text{rev}}$. Writing $R_{\text{rev}} = E[\mathbf{y}_{\text{rev}}\mathbf{y}_{\text{rev}}^T]$ and $D_{\text{rev}} = E[\boldsymbol{\epsilon}_{\text{rev}}\boldsymbol{\epsilon}_{\text{rev}}^T]$, it follows from Eq. (1.7.13) that

$$D_{\text{rev}} = UR_{\text{rev}}U^T \tag{1.7.14}$$

The precise connection between the original basis and its reverse, and between their respective Cholesky factorizations, can be seen as follows. The operation of reversing a vector is equivalent to a linear transformation by the so-called *reversing* matrix J, consisting of ones along its antidiagonal and zeros everywhere else; for example, in the 5×5 case of Example 1.7.1,

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

The reversed vectors will be $\mathbf{y}_{\rm rev} = J\mathbf{y}$ and $\boldsymbol{\epsilon}_{\rm rev} = J\boldsymbol{\epsilon}$. Using the property $J = J^T$, it follows that $R_{\rm rev} = JR_{yy}J$ and $D_{\rm rev} = JR_{\epsilon\epsilon}J$. Comparing Eq. (1.7.9) and Eq. (1.7.13) and using the property $J^2 = I$, we find,

$$\boldsymbol{\epsilon}_{\rm rev} = J\boldsymbol{\epsilon} = JA\mathbf{y} = (JAJ)(J\mathbf{y}) = (JAJ)\mathbf{y}_{\rm rev}\,, \quad \text{or,}$$

$$U = JAJ \tag{1.7.15}$$

Note that J acting on a matrix from the left reverses each column, whereas acting from the right, it reverses each row. Thus, U is obtained from A by reversing all its columns and then all its rows. Regardless of the choice of the vector $\mathbf{y}$, the Gram-Schmidt construction proceeds from the lowest to the highest index of $\mathbf{y}$, and therefore, it can be interpreted as predicting the present from the past. But whether this process leads to LU or UL factorization depends on whether $\mathbf{y}$ or its reverse is used as the basis. Of course, the choice of basis does not affect the computation of linear estimates. As we saw in Eq. (1.6.18), linear estimates are invariant under any linear change of basis; in particular,

$$\hat{x} = E[x\mathbf{y}^T]E[\mathbf{y}\mathbf{y}^T]^{-1}\mathbf{y} = E[x\mathbf{y}_{\rm rev}^T]E[\mathbf{y}_{\rm rev}\mathbf{y}_{\rm rev}^T]^{-1}\mathbf{y}_{\rm rev}$$

In this book, we use both representations $\mathbf{y}$ and $\mathbf{y}_{\rm rev}$, whichever is the most convenient depending on the context and application. For example, in discussing the classical Wiener filtering problem and Kalman filtering in Chap. 11, we find the basis $\mathbf{y}$ more natural. On the other hand, the basis $\mathbf{y}_{\rm rev}$ is more appropriate for discussing the lattice and direct-form realizations of FIR Wiener filters.

The ideas discussed in the last three sections are basic in the development of optimum signal processing algorithms, and will be pursued further in subsequent chapters. However, taking a brief look ahead, we point out how some of these concepts fit into the signal processing context:

1. The correlation canceling/orthogonal decompositions of Eqs. (1.4.1) and (1.6.10) for the basis of optimum Wiener and Kalman filtering.
2. The Gram-Schmidt process expressed by Eqs. (1.6.13) and (1.6.20) forms the basis of linear prediction and is also used in the development of the Kalman filter.
3. The representation $\mathbf{y} = B\boldsymbol{\epsilon}$ may be thought of as a signal model for synthesizing $\mathbf{y}$ by processing the uncorrelated (white noise) vector $\boldsymbol{\epsilon}$ through the linear filter B. The lower-triangular nature of B is equivalent to causality. Such signal models have a very broad range of applications, among which are speech synthesis and modern methods of spectrum estimation.

4. The inverse representation $\boldsymbol{\epsilon} = A\mathbf{y}$ of Eqs. (1.7.1) and (1.7.10) corresponds to the analysis filters of linear prediction. The PARCOR coefficients will turn out to be the reflection coefficients of the lattice filter realizations of linear prediction.
5. The Cholesky factorization (1.6.17) is the matrix analog of the spectral factorization theorem. It not only facilitates the solution of optimum Wiener filtering problems, but also the making of signal models of the type of Eq. (1.6.16).

1.8 Forward/Backward Prediction and LU/UL Factorization

The Gram-Schmidt orthogonalization procedure discussed in the previous sections was a *forward* procedure in the sense that the successive orthogonalization of the components of a random vector $\mathbf{y}$ proceeded forward from the first component to the last. It was given a linear prediction interpretation, that is, at each orthogonalization step, a prediction of the present component of $\mathbf{y}$ is made in terms of all the past ones. The procedure was seen to be mathematically equivalent to the LU Cholesky factorization of the covariance matrix $R = E[\mathbf{y}\mathbf{y}^T]$ (or, the UL factorization with respect to the reversed basis). We remarked in Sec. 1.6 (see also Problem 1.15) that if the Gram-Schmidt construction is started at the other end of the random vector $\mathbf{y}$ then the UL factorization of R is obtained (equivalently, the LU factorization in the reversed basis).

In this section, we discuss in detail such forward and backward Gram-Schmidt constructions and their relationship to *forward* and *backward* linear prediction and to LU and UL Cholesky factorizations, and show how to realize linear estimators in the forward and backward orthogonal bases.

Our main objective is to gain further insight into the properties of the basis of observations $\mathbf{y}$ and to provide a preliminary introduction to a large number of concepts and methods that have become standard tools in modern signal processing practice, namely, Levinson's and Schur's algorithms; fast Cholesky factorizations; lattice filters for linear prediction; lattice realizations of FIR Wiener filters; and fast recursive least squares adaptive algorithms. Although these concepts are fully developed in Chapters 12 and 16, we would like to show in this preliminary discussion how far one can go toward these goals *without* making any assumptions about any structural properties of the covariance matrix R, such as Toeplitz and stationarity properties, or the so-called *shift-invariance* property of adaptive least squares problems.

Forward/Backward Normal Equations

Let $\mathbf{y} = [y_a, \ldots, y_b]^T$ be a random vector whose first and last components are y_a and y_b. Let $\hat{y}_b$ be the best linear estimate of y_b based on the *rest* of the vector $\mathbf{y}$, that is,

$$\hat{y}_b = E[y_b\bar{\mathbf{y}}^T]E[\bar{\mathbf{y}}\bar{\mathbf{y}}^T]^{-1}\bar{\mathbf{y}} \tag{1.8.1}$$

where $\bar{\mathbf{y}}$ is the upper part of $\mathbf{y}$, namely,

$$\mathbf{y} = \begin{bmatrix} y_a \\ \vdots \\ y_b \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{y}} \\ y_b \end{bmatrix} \tag{1.8.2}$$

Similarly, let $\hat{y}_a$ be the best estimate of y_a based on the rest of $\mathbf{y}$, namely,

$$\hat{y}_a = E[y_a\tilde{\mathbf{y}}^T]E[\tilde{\mathbf{y}}\tilde{\mathbf{y}}^T]^{-1}\tilde{\mathbf{y}} \tag{1.8.3}$$

where $\tilde{\mathbf{y}}$ is the lower part of $\mathbf{y}$, that is,

$$\mathbf{y} = \begin{bmatrix} y_a \\ \vdots \\ y_b \end{bmatrix} = \begin{bmatrix} y_a \\ \tilde{\mathbf{y}} \end{bmatrix} \tag{1.8.4}$$

The decompositions (1.8.2) and (1.8.4) imply analogous decompositions of the covariance matrix $R = E[\mathbf{y}\mathbf{y}^T]$ as follows

$$R = \begin{bmatrix} \bar{R} & \mathbf{r}_b \\ \mathbf{r}_b^T & \rho_b \end{bmatrix} = \begin{bmatrix} \rho_a & \mathbf{r}_a^T \\ \mathbf{r}_a & \tilde{R} \end{bmatrix} \tag{1.8.5}$$

where

$$\begin{aligned} \tilde{R} &= E[\tilde{\mathbf{y}}\tilde{\mathbf{y}}^T]\,, \quad \mathbf{r}_a = E[y_a\tilde{\mathbf{y}}]\,, \quad \rho_a = E[y_a^2] \\ \bar{R} &= E[\bar{\mathbf{y}}\,\bar{\mathbf{y}}^T]\,, \quad \mathbf{r}_b = E[y_b\bar{\mathbf{y}}]\,, \quad \rho_b = E[y_b^2] \end{aligned} \tag{1.8.6}$$

We will refer to $\hat{y}_a$ and $\hat{y}_b$ as the forward and backward predictors, respectively. Since we have not yet introduced any notion of time in our discussion of random vectors, we will employ the terms forward and backward as convenient ways of referring to the above two estimates. In the present section, the basis $\mathbf{y}$ will be chosen according to the reversed-basis convention. As discussed in Sec. 1.7, LU becomes UL factorization in the reversed basis. By the same token, UL becomes LU factorization. Therefore, the term forward will be associated with UL and the term backward with LU factorization. The motivation for the choice of basis arises from the time series case, where the consistent usage of these two terms requires that $\mathbf{y}$ be reverse-indexed from high to low indices. For example, a typical choice of $\mathbf{y}$, relevant in the context of Mth order FIR Wiener filtering problems, is

$$\mathbf{y} = \begin{bmatrix} y_n \\ y_{n-1} \\ \vdots \\ y_{n-M} \end{bmatrix}$$

where n represents the time index. Therefore, estimating the first element, y_n, from the rest of $\mathbf{y}$ will be equivalent to prediction, and estimating the last element, y_{n-M}, from the rest of $\mathbf{y}$ will be equivalent to postdiction. Next, we introduce the forward and backward prediction coefficients by

$$\mathbf{a} = \begin{bmatrix} 1 \\ \boldsymbol{\alpha} \end{bmatrix}\,, \quad \mathbf{b} = \begin{bmatrix} \boldsymbol{\beta} \\ 1 \end{bmatrix}\,, \quad \text{where} \quad \boldsymbol{\alpha} = -\tilde{R}^{-1}\mathbf{r}_a\,, \quad \boldsymbol{\beta} = -\bar{R}^{-1}\mathbf{r}_b \tag{1.8.7}$$

In this notation, the predictors (1.8.1) and (1.8.3) are written as

$$\hat{y}_a = -\boldsymbol{\alpha}^T\tilde{\mathbf{y}}\,, \quad \hat{y}_b = -\boldsymbol{\beta}^T\tilde{\mathbf{y}} \tag{1.8.8}$$

The corresponding prediction errors are

$$e_a = y_a - \hat{y}_a = y_a + \boldsymbol{\alpha}^T\tilde{\mathbf{y}} = \mathbf{a}^T\mathbf{y}\,, \quad e_b = y_b - \hat{y}_b = y_b + \boldsymbol{\beta}^T\tilde{\mathbf{y}} = \mathbf{b}^T\mathbf{y} \tag{1.8.9}$$

with mean square values

$$\begin{aligned} E_a &= E[e_a^2] = E\big[\,(\mathbf{a}^T\mathbf{y})\,(\mathbf{y}^T\mathbf{a})\,\big] = \mathbf{a}^T R\mathbf{a} \\ E_b &= E[e_b^2] = E\big[\,(\mathbf{b}^T\mathbf{y})\,(\mathbf{y}^T\mathbf{b})\,\big] = \mathbf{b}^T R\mathbf{b} \end{aligned} \tag{1.8.10}$$

Because the estimation errors are orthogonal to the observations that make up the estimates, that is, $E[e_b\tilde{\mathbf{y}}] = 0$ and $E[e_a\tilde{\mathbf{y}}] = 0$, it follows that $E[\hat{y}_a e_a] = 0$ and $E[\hat{y}_b e_b] = 0$. Therefore, we can write $E[e_a^2] = E[y_a e_a]$ and $E[e_b^2] = E[y_b e_b]$. Thus, the minimized values of the prediction errors (1.8.10) can be written as

$$\begin{aligned} E_a &= E[y_a e_a] = E\big[y_a\,(y_a + \boldsymbol{\alpha}^T\tilde{\mathbf{y}})\,\big] = \rho_a + \boldsymbol{\alpha}^T\mathbf{r}_a = \rho_a - \mathbf{r}_a^T\tilde{R}^{-1}\mathbf{r}_a \\ E_b &= E[y_b e_b] = E\big[y_b\,(y_b + \boldsymbol{\beta}^T\bar{\mathbf{y}})\,\big] = \rho_b + \boldsymbol{\beta}^T\mathbf{r}_b = \rho_b - \mathbf{r}_b^T\bar{R}^{-1}\mathbf{r}_b \end{aligned} \tag{1.8.11}$$

By construction, the mean square estimation errors are positive quantities. This also follows from the positivity of the covariance matrix R. With respect to the block decompositions (1.8.5), it is easily shown that a necessary and sufficient condition for R to be positive definite is that $\bar{R}$ be positive definite and $\rho_b - \mathbf{r}_b^T\bar{R}^{-1}\mathbf{r}_b > 0$; alternatively, that $\tilde{R}$ be positive definite and $\rho_a - \mathbf{r}_a^T\tilde{R}^{-1}\mathbf{r}_a > 0$.

Equations (1.8.7) and (1.8.11) may be combined now into the more compact forms, referred to as the forward and backward *normal equations* of linear prediction,

$$R\mathbf{a} = E_a\mathbf{u}\,, \quad R\mathbf{b} = E_b\mathbf{v}\,, \qquad \text{where} \quad \mathbf{u} = \begin{bmatrix} 1 \\ \mathbf{0} \end{bmatrix}\,, \quad \mathbf{v} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} \tag{1.8.12}$$

For example,

$$R\mathbf{b} = \begin{bmatrix} \bar{R} & \mathbf{r}_b \\ \mathbf{r}_b^T & \rho_b \end{bmatrix}\begin{bmatrix} \boldsymbol{\beta} \\ 1 \end{bmatrix} = \begin{bmatrix} \bar{R}\boldsymbol{\beta} + \mathbf{r}_b \\ \mathbf{r}_b^T\boldsymbol{\beta} + \rho_b \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ E_b \end{bmatrix} = E_b\mathbf{v}$$

and similarly,

$$R\mathbf{a} = \begin{bmatrix} \rho_a & \mathbf{r}_a^T \\ \mathbf{r}_a & \tilde{R} \end{bmatrix}\begin{bmatrix} 1 \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} \rho_a + \mathbf{r}_a^T\boldsymbol{\alpha} \\ \mathbf{r}_a + \tilde{R}\boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} E_a \\ \mathbf{0} \end{bmatrix} = E_a\mathbf{u}$$

Backward Prediction and LU Factorization

Next, we discuss the connection of the forward and backward predictors to the Gram-Schmidt procedure and to the Cholesky factorizations of the covariance matrix R. Consider an arbitrary unit lower triangular matrix $\bar{L}$ of the same dimension as $\bar{R}$ and form the larger unit lower triangular matrix whose bottom row is $\mathbf{b}^T = [\boldsymbol{\beta}^T, 1]$

$$L = \begin{bmatrix} \bar{L} & \mathbf{0} \\ \boldsymbol{\beta}^T & 1 \end{bmatrix} \tag{1.8.13}$$

Then, it follows from Eq. (1.8.12) that

$$LRL^T = \begin{bmatrix} \bar{L}\bar{R}\bar{L}^T & \mathbf{0} \\ \mathbf{0}^T & E_b \end{bmatrix} \tag{1.8.14}$$

Indeed, we have

$$LRL^T = \begin{bmatrix} \bar{L} & \mathbf{0} \\ \boldsymbol{\beta}^T & 1 \end{bmatrix} \begin{bmatrix} \bar{R} & \mathbf{r}_b \\ \mathbf{r}_b^T & \rho_b \end{bmatrix} L^T = \begin{bmatrix} \bar{L}\bar{R} & \bar{L}\mathbf{r}_b \\ \boldsymbol{\beta}^T\bar{R} + \mathbf{r}_b^T & \boldsymbol{\beta}^T\mathbf{r}_b + \rho_b \end{bmatrix} L^T = \begin{bmatrix} \bar{L}\bar{R} & \bar{L}\mathbf{r}_b \\ \mathbf{0}^T & E_b \end{bmatrix} L^T$$

$$= \begin{bmatrix} \bar{L}\bar{R}\bar{L}^T & \bar{L}\mathbf{r}_b + \bar{L}\bar{R}\boldsymbol{\beta} \\ \mathbf{0}^T & E_b \end{bmatrix} = \begin{bmatrix} \bar{L}\bar{R}\bar{L}^T & \mathbf{0} \\ \mathbf{0}^T & E_b \end{bmatrix}$$

Defining the transformed random vector $\mathbf{e}_b = L\mathbf{y}$, we have

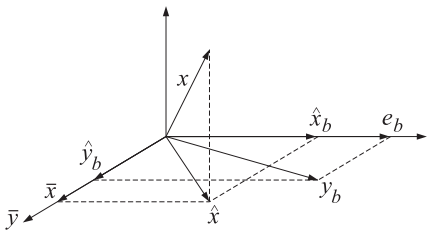
$$\mathbf{e}_b = L\mathbf{y} = \begin{bmatrix} \bar{L} & \mathbf{0} \\ \boldsymbol{\beta}^T & 1 \end{bmatrix} \begin{bmatrix} \bar{\mathbf{y}} \\ y_b \end{bmatrix} = \begin{bmatrix} \bar{L}\bar{\mathbf{y}} \\ \boldsymbol{\beta}^T\bar{\mathbf{y}} + y_b \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{e}}_b \\ e_b \end{bmatrix} \tag{1.8.15}$$

where $\bar{\mathbf{e}}_b = \bar{L}\bar{\mathbf{y}}$. It follows that LRL^T is the covariance matrix of the transformed vector $\mathbf{e}_b$. The significance of Eq. (1.8.14) is that by replacing the $\mathbf{y}$ basis by $\mathbf{e}_b$ we have achieved partial decorrelation of the random vector $\mathbf{y}$. The new basis $\mathbf{e}_b$ is better to work with because it contains less redundancy than $\mathbf{y}$. For example, choosing $\bar{L}$ to be the identity matrix, $\bar{L} = \bar{I}$, Eqs. (1.8.14) and (1.8.15) become

$$LRL^T = \begin{bmatrix} \bar{R} & \mathbf{0} \\ \mathbf{0}^T & E_b \end{bmatrix}, \quad \mathbf{e}_b = \begin{bmatrix} \bar{\mathbf{y}} \\ e_b \end{bmatrix} \tag{1.8.16}$$

This represents the direct sum decomposition of the subspace spanned by $\mathbf{y}$ into the subspace spanned by $\bar{\mathbf{y}}$ and an orthogonal part spanned by e_b, that is,

$$\{\mathbf{y}\} = \{\bar{\mathbf{y}}, y_b\} = \{\bar{\mathbf{y}}\} \oplus \{e_b\}$$



The advantage of the new basis may be appreciated by considering the estimation of a random variable x in terms of $\mathbf{y}$. The estimate $\hat{x}$ may be expressed either in the $\mathbf{y}$ basis, or in the new basis $\mathbf{e}_b$ by

$$\hat{x} = E[x\mathbf{y}^T]E[\mathbf{y}\mathbf{y}^T]^{-1}\mathbf{y} = E[x\,\mathbf{e}_b^T]E[\mathbf{e}_b\mathbf{e}_b^T]^{-1}\mathbf{e}_b$$

Using the orthogonality between $\bar{\mathbf{y}}$ and e_b, or the block-diagonal property of the covariance matrix of $\mathbf{e}_b$ given by Eq. (1.8.16), we find

$$\hat{x} = E[x\bar{\mathbf{y}}^T]E[\bar{\mathbf{y}}\bar{\mathbf{y}}^T]^{-1}\bar{\mathbf{y}} + E[xe_b]E[e_b^2]^{-1}e_b = \bar{x} + \hat{x}_b$$

The two terms in $\hat{x}$ are recognized as the estimates of x based on the two orthogonal parts of the $\mathbf{y}$ basis. The first term still requires the computation of a matrix inverse, namely, $\bar{R}^{-1} = E[\bar{\mathbf{y}}\bar{\mathbf{y}}^T]^{-1}$, but the order of the matrix is reduced by one as compared with the original covariance matrix R. The same order-reduction procedure can now be applied to $\bar{R}$ itself, thereby reducing its order by one. And so on, by repeating the order-reduction procedure, the original matrix R can be completely diagonalized. This process is equivalent to performing Gram-Schmidt orthogonalization on $\mathbf{y}$ starting with y_a and ending with y_b. It is also equivalent to choosing $\bar{L}$ to correspond to the LU Cholesky factorization of $\bar{R}$. Then, the matrix L will correspond to the LU factorization of R. Indeed, if $\bar{L}$ is such that $\bar{L}\bar{R}\bar{L}^T = \bar{D}_b$, that is, a diagonal matrix, then

$$LRL^T = \begin{bmatrix} \bar{L}\bar{R}\bar{L}^T & \mathbf{0} \\ \mathbf{0}^T & E_b \end{bmatrix} = \begin{bmatrix} \bar{D}_b & \mathbf{0} \\ \mathbf{0}^T & E_b \end{bmatrix} = D_b \tag{1.8.17}$$

will itself be diagonal. The basis $\mathbf{e}_b = L\mathbf{y}$ will be completely decorrelated, having diagonal covariance matrix $E[\mathbf{e}_b\mathbf{e}_b^T] = D_b$. Thus, by successively solving backward prediction problems of lower and lower order we eventually orthogonalize the original basis $\mathbf{y}$ and obtain the LU factorization of its covariance matrix. By construction, the bottom row of L is the backward predictor $\mathbf{b}^T$. Similarly, the bottom row of $\bar{L}$ will be the backward predictor of order one less, and so on. In other words, the rows of L are simply the *backward predictors* of successive orders. The overall construction of L is illustrated by the following example.

Example 1.8.1: The random vector $\mathbf{y} = [y_a, y_c, y_b]^T$ has covariance matrix

$$R = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 3 & 2 \\ 0 & 2 & 3 \end{bmatrix}$$

By successively solving backward prediction problems of lower and lower order construct the LU factorization of R.

Solution: The backward prediction coefficients for predicting y_b are given by Eq. (1.8.7):

$$\boldsymbol{\beta} = -\bar{R}^{-1}\mathbf{r}_b = -\begin{bmatrix} 1 & 1 \\ 1 & 3 \end{bmatrix}^{-1}\begin{bmatrix} 0 \\ 2 \end{bmatrix} = -\frac{1}{2}\begin{bmatrix} 3 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} 0 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

Thus, $\mathbf{b}^T = [\boldsymbol{\beta}^T, 1] = [1, -1, 1]$. The estimation error is given by Eq. (1.8.11):

$$E_b = \rho_b + \boldsymbol{\beta}^T\mathbf{r}_b = 3 + [1, -1]\begin{bmatrix} 0 \\ 2 \end{bmatrix} = 1$$

Repeating the procedure on $\bar{R} = \begin{bmatrix} 1 & 1 \\ 1 & 3 \end{bmatrix}$, we find for the corresponding backward prediction coefficients, satisfying $\bar{R}\bar{\mathbf{b}} = \bar{E}_b\bar{\mathbf{v}}$, $\bar{\mathbf{v}} = [0, 1]^T$

$$\bar{\boldsymbol{\beta}} = -[1]^{-1}[1] = [-1], \quad \bar{\mathbf{b}}^T = [\bar{\boldsymbol{\beta}}^T, 1] = [-1, 1]$$

and $\bar{E}_b = \bar{\rho}_b + \bar{\boldsymbol{\beta}}^T \bar{\mathbf{r}}_b = 3 - 1\times 1 = 2$. The rows of L are the backward predictor coefficients, and the diagonal entries of D_b are the E_b. Thus,

$$L = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 1 & -1 & 1 \end{bmatrix}, \quad D_b = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

It is easily verified that $LRL^T = D_b$. Note that the first entry of D_b is always equal to ρ_a. Next, we obtain the same results by carrying out the Gram-Schmidt construction starting at y_a and ending with y_b. Starting with $\epsilon_1 = y_a$ and $E[\epsilon_1^2] = 1$, define

$$\epsilon_2 = y_c - E[y_c\epsilon_1]E[\epsilon_1^2]^{-1}\epsilon_1 = y_c - y_a$$

having $E[\epsilon_2^2] = E[y_c^2] - 2E[y_c y_a] + E[y_a^2] = 2$. Thus, the $\bar{\mathbf{e}}_b$ portion of the Gram-Schmidt construction will be

$$\bar{\mathbf{e}}_b = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} y_a \\ y_c \end{bmatrix} = \bar{L}\bar{\mathbf{y}}$$

The last step of the Gram-Schmidt construction is

$$e_b = y_b - E[y_b\epsilon_1]E[\epsilon_1^2]^{-1}\epsilon_1 - E[y_b\epsilon_2]E[\epsilon_2^2]^{-1}\epsilon_2 = y_b - (y_c - y_a) = y_a - y_c + y_b$$

giving for the last row of L, $\mathbf{b}^T = [1, -1, 1]$. In the above step, we used

$$E[y_b\epsilon_2] = E[y_b(y_c - y_a)] = E[y_b y_c] - E[y_b y_a] = 2 - 0 = 2$$

and $E[y_b\epsilon_1] = E[y_b y_a] = 0$. □

Linear Estimation in the Backward Basis

Equation (1.8.17) may be written in the form

$$R = L^{-1} D_b L^{-T} \tag{1.8.18}$$

where L^{-T} is the inverse of the transpose of L. Thus, L^{-1} and L^{-T} correspond to the conventional LU Cholesky factors of R. The computational advantage of this form becomes immediately obvious when we consider the inverse of R,

$$R^{-1} = L^T D_b^{-1} L \tag{1.8.19}$$

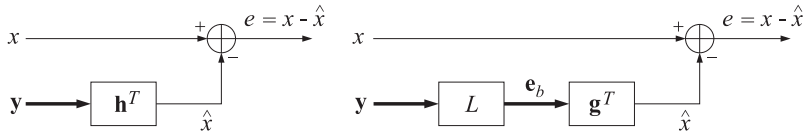
which shows that R^{-1} can be computed without any matrix inversion (the inverse of the diagonal matrix D_b is trivial). The design of linear estimators is simplified considerably in the $\mathbf{e}_b$ basis. The estimate of x is

$$\hat{x} = \mathbf{h}^T \mathbf{y} \tag{1.8.20}$$

where $\mathbf{h} = E[\mathbf{y}\mathbf{y}^T]^{-1}E[x\mathbf{y}] \equiv R^{-1}\mathbf{r}$. Writing $\mathbf{y} = L^{-1}\mathbf{e}_b$ and defining a new vector of estimation weights by $\mathbf{g} = L^{-T}\mathbf{h}$, we can rewrite Eq. (1.8.20) as

$$\hat{x} = \mathbf{h}^T \mathbf{y} = \mathbf{g}^T \mathbf{e}_b \tag{1.8.21}$$

The block diagram representations of the two realizations are shown below:



There are three major advantages of the representation of Eq. (1.8.21) over Eq. (1.8.20). First, to get the estimate $\hat{x}$ using (1.8.20), the processor has to linearly combine a lot of *redundant* information because the $\mathbf{y}$ basis is correlated, whereas the processor (1.8.21) linearly combines only the *non-redundant* part of the same information. This has important implications for the adaptive implementations of such processors. An adaptive processor that uses the representation (1.8.20) will tend to be slow in learning the statistics of the data vector $\mathbf{y}$ because it has to process all the redundancies in the data. Moreover, the more the redundancies, or equivalently, the higher the correlations in the data $\mathbf{y}$, the slower the speed of adaptation. On the other hand, an adaptive processor based on (1.8.21) should adapt very quickly. The preprocessing operation, $\mathbf{e}_b = L\mathbf{y}$, that decorrelates the data vector $\mathbf{y}$ can also be implemented adaptively. In time series applications, it is conveniently realized by means of a *lattice structure*. In adaptive array applications, it gives rise to the so-called *Gram-Schmidt preprocessor* implementations.

Second, the computation of $\mathbf{g}$ can be done efficiently without any matrix inversion. Given the LU factors of R as in Eq. (1.8.19) and the cross correlation vector $\mathbf{r}$, we may compute $\mathbf{g}$ by

$$\mathbf{g} = L^{-T}\mathbf{h} = L^{-T}R^{-1}\mathbf{r} = L^{-T}(L^T D_b^{-1} L)\mathbf{r} = D_b^{-1}L\mathbf{r} \tag{1.8.22}$$

If so desired, the original weights $\mathbf{h}$ may be recovered from $\mathbf{g}$ by

$$\mathbf{h} = L^T \mathbf{g} \tag{1.8.23}$$

The third advantage of the form Eq. (1.8.21) is that any lower-order portion of the weight vector $\mathbf{g}$ is already optimal for that order. Thus, the order of the estimator can be increased without having to redesign the lower-order portions of it. Recognizing that $L\mathbf{r} = LE[x\mathbf{y}] = E[x\,\mathbf{e}_b]$, we write Eq. (1.8.22) as

$$\mathbf{g} = D_b^{-1}E[x\,\mathbf{e}_b] = \begin{bmatrix} \bar{D}_b^{-1}E[x\,\bar{\mathbf{e}}_b] \\ E_b^{-1}E[xe_b] \end{bmatrix} \equiv \begin{bmatrix} \bar{\mathbf{g}} \\ g \end{bmatrix}$$

where we used the diagonal nature of D_b given in Eq. (1.8.17) and the decomposition (1.8.15). The estimate (1.8.21) can be written as

$$\hat{x} = \mathbf{g}^T\mathbf{e}_b = [\bar{\mathbf{g}}^T, g]\begin{bmatrix} \bar{\mathbf{e}}_b \\ e_b \end{bmatrix} = \bar{\mathbf{g}}^T\bar{\mathbf{e}}_b + ge_b \equiv \bar{x} + \hat{x}_b \tag{1.8.24}$$

It is clear that the two terms

$$\bar{x} = \bar{\mathbf{g}}^T\bar{\mathbf{e}}_b = E[x\,\bar{\mathbf{e}}_b^T]\bar{D}_b^{-1}\bar{\mathbf{e}}_b\,, \quad \hat{x}_b = ge_b = E[xe_b]E[e_b^2]^{-1}e_b \tag{1.8.25}$$

are the optimal estimates of x based on the two orthogonal parts of the subspace of observations, namely,

$$\{\mathbf{y}\} = \{\bar{\mathbf{y}}\} \oplus \{e_b\}\,, \quad \text{or,} \quad \{\mathbf{e}_b\} = \{\bar{\mathbf{e}}_b\} \oplus \{e_b\}$$

The first term, $\bar{x}$, is the same estimate of x based on $\bar{\mathbf{y}}$ that we considered earlier but now it is expressed in the diagonal basis $\bar{\mathbf{e}}_b = \bar{L}\bar{\mathbf{y}}$. The second term, $\hat{x}_b$, represents the improvement in that estimate that arises by taking into account one more observation, namely, y_b. It represents that part of x that cannot be estimated from $\bar{\mathbf{y}}$. And, it is computable only from that part of the new observation y_b that *cannot* be predicted from $\bar{\mathbf{y}}$, that is, e_b. The degree of improvement of $\hat{x}$ over $\bar{x}$, as measured by the mean-square estimation errors, can be computed explicitly in this basis. To see this, denote the estimation errors based on $\mathbf{y}$ and $\bar{\mathbf{y}}$ by

$$e = x - \hat{x} = x - \mathbf{g}^T\mathbf{e}_b\,, \quad \bar{e} = x - \bar{x} = x - \bar{\mathbf{g}}^T\bar{\mathbf{e}}_b$$

Then, Eq. (1.8.24) implies $e = x - \hat{x} = (x - \bar{x}) - \hat{x}_b$, or

$$e = \bar{e} - ge_b \tag{1.8.26}$$

Because e and $\mathbf{y}$, or $\mathbf{e}_b$, are orthogonal, we have $E[\hat{x}e] = 0$, which implies that

$$\mathcal{E} = E[e^2] = E[xe] = E\big[x(x - \mathbf{g}^T\mathbf{e}_b)\big] = E[x^2] - \mathbf{g}^T E[x\,\mathbf{e}_b]$$

Similarly, $\bar{\mathcal{E}} = E[\bar{e}^2] = E[x^2] - \bar{\mathbf{g}}^T E[x\,\bar{\mathbf{e}}_b]$. It follows that

$$\mathcal{E} = \bar{\mathcal{E}} - gE[xe_b] = \bar{\mathcal{E}} - g^2E_b \tag{1.8.27}$$

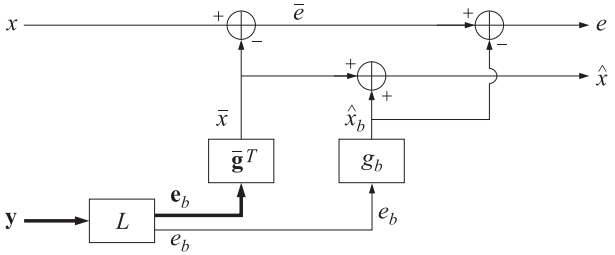
where we used $g = E[xe_b]E_b^{-1}$. The subtracted term represents the improvement obtained by including one more observation in the estimate. It follows from the above discussion that the lower-order portion $\bar{\mathbf{g}}$ of $\mathbf{g}$ is already optimal. This is not so in the $\mathbf{y}$ basis, that is, the lower-order portion of $\mathbf{h}$ is not equal to the lower-order optimal weights $\bar{\mathbf{h}} = \bar{R}^{-1}\bar{\mathbf{r}}$, where $\bar{\mathbf{r}} = E[x\bar{\mathbf{y}}]$. The explicit relationship between the two may be found as follows. Inserting the block decomposition Eq. (1.8.13) of L into Eq. (1.8.19) and using the lower-order result $\bar{R}^{-1} = \bar{L}^T\bar{D}_b^{-1}\bar{L}$, we may derive the following order-updating expression for R^{-1}

$$R^{-1} = \begin{bmatrix} \bar{R}^{-1} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \frac{1}{E_b}\mathbf{b}\mathbf{b}^T \tag{1.8.28}$$

Noting that $\bar{\mathbf{r}}$ is the lower-order part of $\mathbf{r}$, $\mathbf{r} = [\bar{\mathbf{r}}^T, r_b]^T$, where $r_b = E[xy_b]$, we obtain the following order-updating equation for the optimal $\mathbf{h}$

$$\mathbf{h} = R^{-1}\mathbf{r} = \begin{bmatrix} \bar{R}^{-1} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix}\begin{bmatrix} \bar{\mathbf{r}} \\ r_b \end{bmatrix} + \frac{1}{E_b}(\mathbf{b}\mathbf{b}^T)\mathbf{r} = \begin{bmatrix} \bar{\mathbf{h}} \\ 0 \end{bmatrix} + c_b\mathbf{b} \tag{1.8.29}$$

where $c_b = (\mathbf{b}^T\mathbf{r})/E_b = (\boldsymbol{\beta}^T\bar{\mathbf{r}} + r_b)/E_b$. A block diagram realization that takes into account the order-recursive construction of the estimate (1.8.24) and estimation error (1.8.26) is shown below.



In Chap. 12, we discuss in greater detail the design procedure given by Eq. (1.8.22) and show how to realize Eqs. (1.8.21), or (1.8.24) and (1.8.26), by means of a *lattice structure*. In Chap. 16, we discuss the corresponding adaptive versions, leading to the so-called *adaptive lattice filters* for linear prediction and Wiener filtering, such as the gradient lattice and RLS lattice.

Forward Prediction and UL Factorization

Next, we turn our attention to the forward predictors defined in Eq. (1.8.12). They lead to UL (rather than LU) factorization of the covariance matrix. Considering an arbitrary unit upper-triangular matrix $\tilde{U}$ of the same dimension as $\tilde{R}$, we may form the larger unit upper-triangular matrix whose top row is the forward predictor $\mathbf{a}^T = [1, \boldsymbol{\alpha}^T]$

$$U = \begin{bmatrix} 1 & \boldsymbol{\alpha}^T \\ \mathbf{0} & \tilde{U} \end{bmatrix} \tag{1.8.30}$$

Then, it follows from Eq. (1.8.12) that

$$URU^T = \begin{bmatrix} E_a & \mathbf{0}^T \\ \mathbf{0} & \tilde{U}\tilde{R}\tilde{U}^T \end{bmatrix} \tag{1.8.31}$$

It follows that URU^T is the covariance matrix of the transformed vector

$$\mathbf{e}_a = U\mathbf{y} = \begin{bmatrix} 1 & \boldsymbol{\alpha}^T \\ \mathbf{0} & \tilde{U} \end{bmatrix}\begin{bmatrix} y_a \\ \tilde{\mathbf{y}} \end{bmatrix} = \begin{bmatrix} y_a + \boldsymbol{\alpha}^T\tilde{\mathbf{y}} \\ \tilde{U}\tilde{\mathbf{y}} \end{bmatrix} = \begin{bmatrix} e_a \\ \tilde{\mathbf{e}}_a \end{bmatrix} \tag{1.8.32}$$

Choosing $\tilde{U}$ to correspond to the UL factor of $\tilde{R}$, that is, $\tilde{U}\tilde{R}\tilde{U}^T = \tilde{D}_a$, where $\tilde{D}_a$ is diagonal, then Eq. (1.8.31) implies that U will correspond to the UL factor of R:

$$URU^T = \begin{bmatrix} E_a & \mathbf{0}^T \\ \mathbf{0} & \tilde{D}_a \end{bmatrix} = D_a \tag{1.8.33}$$

This is equivalent to Eq. (1.7.14). The basis $\mathbf{e}_a = U\mathbf{y}$ is completely decorrelated, with covariance matrix $E[\mathbf{e}_a\mathbf{e}_a^T] = D_a$. It is equivalent to Eq. (1.7.13). The rows of U are the *forward* predictors of successive orders. And therefore, the UL factorization of R is equivalent to performing the Gram-Schmidt construction starting at the endpoint y_b and proceeding to y_a. The following example illustrates the method.

Example 1.8.2: By successively solving forward prediction problems of lower and lower order, construct the UL factorization of the covariance matrix R of Example 1.8.1.

Solution: Using Eq. (1.8.7), we find

$$\boldsymbol{\alpha} = -\tilde{R}^{-1}\mathbf{r}_a = -\begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}^{-1}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = -\frac{1}{5}\begin{bmatrix} 3 & -2 \\ -2 & 3 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} -3/5 \\ 2/5 \end{bmatrix}$$

Thus, $\mathbf{a}^T = [1, \boldsymbol{\alpha}^T] = [1, -3/5, 2/5]$. The estimation error is

$$E_a = \rho_a + \boldsymbol{\alpha}^T\mathbf{r}_a = 1 + [-3/5, 2/5]\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{2}{5}$$

Repeating the procedure on $\tilde{R} = \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}$, we find the corresponding forward prediction coefficients, satisfying $\tilde{R}\tilde{\mathbf{a}} = \tilde{E}_a\tilde{\mathbf{u}}$, where $\tilde{\mathbf{u}} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$,

$$\tilde{\boldsymbol{\alpha}} = -[3]^{-1}[2] = -\frac{2}{3}, \quad \tilde{\mathbf{a}}^T = [1, \tilde{\boldsymbol{\alpha}}^T] = [1, -2/3]$$

and $\tilde{E}_a = \tilde{\rho}_a + \tilde{\boldsymbol{\alpha}}^T\tilde{\mathbf{r}}_a = 3 - (2/3)\times 2 = 5/3$. The rows of U are the forward predictor coefficients and the diagonal entries of D_a are the E_as:

$$U = \begin{bmatrix} 1 & -3/5 & 2/5 \\ 0 & 1 & -2/3 \\ 0 & 0 & 1 \end{bmatrix}, \quad D_a = \begin{bmatrix} 2/5 & 0 & 0 \\ 0 & 5/3 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

It is easily verified that $URU^T = D_a$. Note that the last entry of D_a is always equal to ρ_b. □

Equation (1.8.33) can be used to compute the inverse of R:

$$R^{-1} = U^T D_a^{-1} U \tag{1.8.34}$$

Using the lower-order result $\tilde{R}^{-1} = \tilde{U}^T\tilde{D}_a^{-1}\tilde{U}$ and the decomposition (1.8.30), we find the following order-updating equation for R^{-1}, analogous to Eq. (1.8.28):

$$R^{-1} = \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \tilde{R}^{-1} \end{bmatrix} + \frac{1}{E_a}\mathbf{a}\mathbf{a}^T \tag{1.8.35}$$

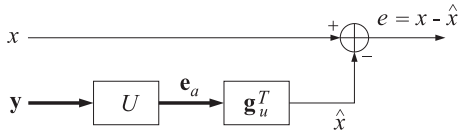
Denoting $\tilde{\mathbf{r}} = E[x\tilde{\mathbf{y}}]$ and $r_a = E[xy_a]$, we obtain the alternative order-update equation for $\mathbf{h}$, analogous to Eq. (1.8.29):

$$\mathbf{h} = R^{-1}\mathbf{r} = \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \tilde{R}^{-1} \end{bmatrix}\begin{bmatrix} r_a \\ \tilde{\mathbf{r}} \end{bmatrix} + \frac{1}{E_a}(\mathbf{a}^T\mathbf{r})\mathbf{a} = \begin{bmatrix} 0 \\ \tilde{\mathbf{h}} \end{bmatrix} + c_a\mathbf{a} \tag{1.8.36}$$

where $c_a = (\mathbf{a}^T\mathbf{r})/E_a = (r_a + \boldsymbol{\alpha}^T\tilde{\mathbf{r}})/E_a$, and $\tilde{\mathbf{h}} = \tilde{R}^{-1}\tilde{\mathbf{r}}$ is the lower-order optimal estimator for estimating x from $\tilde{\mathbf{y}}$. By analogy with Eq. (1.8.21), we could also choose to express the estimates in the $\mathbf{e}_a$ basis

$$\hat{x} = \mathbf{h}^T\mathbf{y} = \mathbf{h}^T U^{-1}\mathbf{e}_a = \mathbf{g}_u^T\mathbf{e}_a \tag{1.8.37}$$

where $\mathbf{g}_u = U^{-T}\mathbf{h}$. A realization is shown below.



The most important part of the realizations based on the diagonal bases $\mathbf{e}_a$ or $\mathbf{e}_a$ is the preprocessing part that decorrelates the $\mathbf{y}$ basis, namely, $\mathbf{e}_b = L\mathbf{y}$, or $\mathbf{e}_a = U\mathbf{y}$. We will see in Chapters 12 and 16 that this part can be done efficiently using the Levinson recursion and the *lattice structures* of linear prediction. The LU representation, based on the backward predictors, $\mathbf{e}_b = L\mathbf{y}$, is preferred because it is somewhat more conveniently realized in terms of the lattice structure than the UL representation $\mathbf{e}_a = U\mathbf{y}$.

Order Updates

So far, we studied the problems of forward and backward prediction separately from each other. Next, we would like to consider the two problems together and show how to construct the solution of the pair of equations (1.8.12) from the solution of a similar pair of lower order. This construction is the essence behind Levinson's algorithm for solving the linear prediction problem, both in the stationary and in the adaptive least squares cases. Consider the following pair of lower-order forward and backward predictors, defined in terms of the block decompositions (1.8.5) of R:

$$\tilde{R}\tilde{\mathbf{a}} = \tilde{E}_a\tilde{\mathbf{u}}, \quad \tilde{R}\tilde{\mathbf{b}} = \tilde{E}_b\tilde{\mathbf{v}} \tag{1.8.38}$$

where $\tilde{\mathbf{u}}$ and $\tilde{\mathbf{v}}$ are unit vectors of dimension one less than those of Eq. (1.8.12). They are related to $\mathbf{u}$ and $\mathbf{v}$ through the decompositions

$$\mathbf{u} = \begin{bmatrix} \tilde{\mathbf{u}} \\ 0 \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} 0 \\ \tilde{\mathbf{v}} \end{bmatrix} \tag{1.8.39}$$

The basic result we would like to show is that the solution of the pair (1.8.12) may be constructed from the solution of the pair (1.8.38) by

$$\begin{aligned} \mathbf{a} &= \begin{bmatrix} \tilde{\mathbf{a}} \\ 0 \end{bmatrix} - \gamma_b \begin{bmatrix} 0 \\ \tilde{\mathbf{b}} \end{bmatrix} \\ \mathbf{b} &= \begin{bmatrix} 0 \\ \tilde{\mathbf{b}} \end{bmatrix} - \gamma_a \begin{bmatrix} \tilde{\mathbf{a}} \\ 0 \end{bmatrix} \end{aligned} \tag{1.8.40}$$

This result is motivated by Eq. (1.8.39), which shows that the right-hand sides of Eqs. (1.8.38) are already part of the right-hand sides of Eq. (1.8.12), and therefore, the solutions of Eq. (1.8.38) may appear as part of the solutions of (1.8.12). The prediction errors are updated by

$$E_a = (1 - \gamma_a\gamma_b)\tilde{E}_a, \quad E_b = (1 - \gamma_a\gamma_b)\tilde{E}_b \tag{1.8.41}$$

where

$$\gamma_b = \frac{\Delta_a}{\tilde{E}_b}, \quad \gamma_a = \frac{\Delta_b}{\tilde{E}_a} \tag{1.8.42}$$

The γs are known as the *reflection* or *PARCOR* coefficients. The quantities Δ_a and Δ_b are defined by

$$\Delta_a = \tilde{\mathbf{a}}^T\mathbf{r}_b, \quad \Delta_b = \tilde{\mathbf{b}}^T\mathbf{r}_a \tag{1.8.43}$$

The two Δs are equal, $\Delta_a = \Delta_b$, as seen from the following considerations. Using the decompositions (1.8.5), we find

$$R\begin{bmatrix} \tilde{\mathbf{a}} \\ 0 \end{bmatrix} = \begin{bmatrix} \tilde{R} & \mathbf{r}_b \\ \mathbf{r}_b^T & \rho_b \end{bmatrix}\begin{bmatrix} \tilde{\mathbf{a}} \\ 0 \end{bmatrix} = \begin{bmatrix} \tilde{R}\tilde{\mathbf{a}} \\ \mathbf{r}_b^T\tilde{\mathbf{a}} \end{bmatrix} = \begin{bmatrix} \tilde{E}_a\tilde{\mathbf{u}} \\ \Delta_a \end{bmatrix}$$

$$R\begin{bmatrix} 0 \\ \tilde{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} \rho_a & \mathbf{r}_a^T \\ \mathbf{r}_a & \tilde{R} \end{bmatrix}\begin{bmatrix} 0 \\ \tilde{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_a^T\tilde{\mathbf{b}} \\ \tilde{R}\tilde{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} \Delta_b \\ \tilde{E}_b\tilde{\mathbf{v}} \end{bmatrix}$$

They may be written more conveniently as

$$R\begin{bmatrix}\bar{\mathbf{a}}\\0\end{bmatrix}=\begin{bmatrix}\bar{E}_a\bar{\mathbf{u}}\\\Delta_a\end{bmatrix}=\bar{E}_a\begin{bmatrix}\bar{\mathbf{u}}\\0\end{bmatrix}+\Delta_a\begin{bmatrix}\mathbf{0}\\1\end{bmatrix}=\bar{E}_a\mathbf{u}+\Delta_a\mathbf{v} \tag{1.8.44a}$$

$$R\begin{bmatrix}0\\\tilde{\mathbf{b}}\end{bmatrix}=\begin{bmatrix}\Delta_b\\\tilde{E}_b\tilde{\mathbf{v}}\end{bmatrix}=\Delta_b\begin{bmatrix}1\\\mathbf{0}\end{bmatrix}+\tilde{E}_b\begin{bmatrix}0\\\tilde{\mathbf{v}}\end{bmatrix}=\Delta_b\mathbf{u}+\tilde{E}_b\mathbf{v} \tag{1.8.44b}$$

Noting that $\mathbf{d}^T\mathbf{u}$ and $\mathbf{d}^T\mathbf{v}$ are equal to the first and last components of a vector $\mathbf{d}$, we have $[0,\tilde{\mathbf{b}}^T]\mathbf{u}=0$ and $[0,\tilde{\mathbf{b}}^T]\mathbf{v}=1$ because the first and last components of $[0,\tilde{\mathbf{b}}^T]$ are zero and one, respectively. Similarly, $[\bar{\mathbf{a}}^T,0]\mathbf{u}=1$ and $[\bar{\mathbf{a}}^T,0]\mathbf{v}=0$. Thus, multiplying Eq. (1.8.44a) from the left by $[0,\tilde{\mathbf{b}}^T]$ and Eq. (1.8.44b) by $[\bar{\mathbf{a}}^T,0]$, we find

$$[0,\tilde{\mathbf{b}}^T]R\begin{bmatrix}\bar{\mathbf{a}}\\0\end{bmatrix}=\Delta_a\,,\quad[\bar{\mathbf{a}}^T,0]R\begin{bmatrix}0\\\tilde{\mathbf{b}}\end{bmatrix}=\Delta_b \tag{1.8.45}$$

The equality of the Δs follows now from the fact that R is a symmetric matrix. Thus,

$$\Delta_a=\Delta_b\equiv\Delta \tag{1.8.46}$$

An alternative proof, based on partial correlations, will be given later. Equations (1.8.40) and (1.8.41) follow now in a straightforward fashion from Eq. (1.8.44). Multiplying the first part of Eq. (1.8.40) by R and using Eqs. (1.8.12) and (1.8.44), we find

$$E_a\mathbf{u}=R\mathbf{a}=R\begin{bmatrix}\bar{\mathbf{a}}\\0\end{bmatrix}-\gamma_bR\begin{bmatrix}0\\\tilde{\mathbf{b}}\end{bmatrix}$$

or,

$$E_a\mathbf{u}=(\bar{E}_a\mathbf{u}+\Delta_a\mathbf{v})-\gamma_b(\Delta_b\mathbf{u}+\tilde{E}_b\mathbf{v})=(\bar{E}_a-\gamma_b\Delta_b)\mathbf{u}+(\Delta_b-\gamma_b\tilde{E}_b)\mathbf{v}$$

which implies the conditions

$$E_a=\bar{E}_a-\gamma_b\Delta_b\,,\quad\Delta_a-\gamma_b\tilde{E}_b=0 \tag{1.8.47}$$

Similarly, multiplying the second part of the Eq. (1.8.40) by R, we obtain

$$E_b\mathbf{v}=(\Delta_b\mathbf{u}+\tilde{E}_b\mathbf{v})-\gamma_a(\bar{E}_a\mathbf{u}+\Delta_b\mathbf{v})=(\Delta_b-\gamma_a\bar{E}_a)\mathbf{u}+(\tilde{E}_b-\gamma_a\Delta_a)\mathbf{v}$$

which implies

$$E_b=\tilde{E}_b-\gamma_a\Delta_a\,,\quad\Delta_b-\gamma_a\bar{E}_a=0 \tag{1.8.48}$$

Equations (1.8.41) and (1.8.42) follow now from (1.8.47) and (1.8.48). By analogy with Eq. (1.8.9), we may now define the prediction errors corresponding to the lower-order predictors $\bar{\mathbf{a}}$ and $\tilde{\mathbf{b}}$ by

$$\bar{e}_a=\bar{\mathbf{a}}^T\bar{\mathbf{y}}\,,\quad\tilde{e}_b=\tilde{\mathbf{b}}^T\tilde{\mathbf{y}} \tag{1.8.49}$$

Using Eqs. (1.8.9) and (1.8.40), we find the following updating equations for the prediction errors

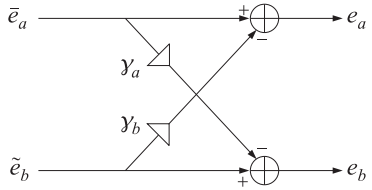
$$\mathbf{a}^T\mathbf{y}=[\bar{\mathbf{a}}^T,0]\begin{bmatrix}\bar{\mathbf{y}}\\y_b\end{bmatrix}-\gamma_b[0,\tilde{\mathbf{b}}^T]\begin{bmatrix}y_a\\\tilde{\mathbf{y}}\end{bmatrix}=\bar{\mathbf{a}}^T\bar{\mathbf{y}}-\gamma_b\tilde{\mathbf{b}}^T\tilde{\mathbf{y}}$$

$$\mathbf{b}^T\mathbf{y}=[0,\tilde{\mathbf{b}}^T]\begin{bmatrix}y_a\\\tilde{\mathbf{y}}\end{bmatrix}-\gamma_a[\bar{\mathbf{a}}^T,0]\begin{bmatrix}\bar{\mathbf{y}}\\y_b\end{bmatrix}=\tilde{\mathbf{b}}^T\tilde{\mathbf{y}}-\gamma_a\bar{\mathbf{a}}^T\bar{\mathbf{y}}$$

or,

$$e_a=\bar{e}_a-\gamma_b\tilde{e}_b\,,\quad e_b=\tilde{e}_b-\gamma_a\bar{e}_a \tag{1.8.50}$$

A lattice type realization ofEq. (1.8.50) is shown below. It forms the basis of the lattice structures of linear prediction discussed in Chapters 12 and 16.



The order-updating procedure is illustrated by the following example.

Example 1.8.3: Using Eq. (1.8.40), construct the forward and backward predictors **a** and **b** found previously in Examples 1.8.1 and 1.8.2.

Solution: The first part of Eq. (1.8.38), $\bar{R}\bar{\mathbf{a}}=\bar{E}_a\bar{\mathbf{u}}$ is solved as follows:

$$\begin{bmatrix}1&1\\1&3\end{bmatrix}\begin{bmatrix}1\\\bar{\alpha}\end{bmatrix}=\bar{E}_a\begin{bmatrix}1\\0\end{bmatrix}\quad\Rightarrow\quad\bar{\alpha}=-\frac{1}{3}\,,\quad\bar{E}_a=\frac{2}{3}$$

Therefore, $\bar{\mathbf{a}}=\begin{bmatrix}1\\-1/3\end{bmatrix}$. Similarly, $\tilde{R}\tilde{\mathbf{y}}=\tilde{E}_b\tilde{\mathbf{v}}$, is solved by

$$\begin{bmatrix}3&2\\2&3\end{bmatrix}\begin{bmatrix}\tilde{\beta}\\1\end{bmatrix}=\tilde{E}_b\begin{bmatrix}0\\1\end{bmatrix}\quad\Rightarrow\quad\tilde{\beta}=-\frac{2}{3}\,,\quad\tilde{E}_b=\frac{5}{3}$$

Hence, $\tilde{\mathbf{b}}=\begin{bmatrix}-2/3\\1\end{bmatrix}$. Next, we determine

$$\Delta=\bar{\mathbf{a}}^T\mathbf{r}_b=[1,-1/3]\begin{bmatrix}0\\2\end{bmatrix}=-\frac{2}{3}\,,\quad\gamma_b=\frac{\Delta}{\tilde{E}_b}=-\frac{2}{5}\,,\quad\gamma_a=\frac{\Delta}{\bar{E}_a}=-1$$

It follows from Eq. (1.8.40) that

$$\mathbf{a}=\begin{bmatrix}\bar{\mathbf{a}}\\0\end{bmatrix}-\gamma_b\begin{bmatrix}0\\\tilde{\mathbf{b}}\end{bmatrix}=\begin{bmatrix}1\\-1/3\\0\end{bmatrix}-\left(-\frac{2}{5}\right)\begin{bmatrix}0\\-2/3\\1\end{bmatrix}=\begin{bmatrix}1\\-3/5\\2/5\end{bmatrix}$$

$$\mathbf{b}=\begin{bmatrix}0\\\tilde{\mathbf{b}}\end{bmatrix}-\gamma_a\begin{bmatrix}\bar{\mathbf{a}}\\0\end{bmatrix}=\begin{bmatrix}0\\-2/3\\1\end{bmatrix}-(-1)\begin{bmatrix}1\\-1/3\\0\end{bmatrix}=\begin{bmatrix}1\\-1\\1\end{bmatrix}$$

and the prediction errors are found from Eq. (1.8.41)

$$E_a=\bar{E}_a(1-\gamma_a\gamma_b)=\frac{2}{3}(1-2/5)=\frac{2}{5}\,,\quad E_b=\tilde{E}_b(1-\gamma_a\gamma_b)=\frac{5}{3}(1-2/5)=1$$

Partial Correlation Interpretation

Next, we show that γ_a and γ_b are partial correlation coefficients in the sense of Sec. 1.7. Let $\mathbf{y}_c$ denote all the components of $\mathbf{y}$ that lie between y_a and y_b, so that

$$\mathbf{y} = \begin{bmatrix} y_a \\ \mathbf{y}_c \\ y_b \end{bmatrix}, \quad \bar{\mathbf{y}} = \begin{bmatrix} y_a \\ \mathbf{y}_c \end{bmatrix}, \quad \tilde{\mathbf{y}} = \begin{bmatrix} \mathbf{y}_c \\ y_b \end{bmatrix} \tag{1.8.51}$$

The forward predictor $\mathbf{a}$ was defined as the best estimator of y_a based on the rest of the vector $\mathbf{y}$. By the same token, $\bar{\mathbf{a}}$ is the best estimator of y_a based on the rest of $\bar{\mathbf{y}}$, that is, $\mathbf{y}_c$. Similarly, the backward predictor $\tilde{\mathbf{b}}$ defines the best estimator of y_b based on the rest of the vector $\tilde{\mathbf{y}}$; again, $\mathbf{y}_c$. Decomposing $\bar{\mathbf{a}}$ and $\tilde{\mathbf{b}}$ as

$$\bar{\mathbf{a}} = \begin{bmatrix} 1 \\ \bar{\boldsymbol{\alpha}} \end{bmatrix}, \quad \tilde{\mathbf{b}} = \begin{bmatrix} \tilde{\boldsymbol{\beta}} \\ 1 \end{bmatrix}$$

we may write the best estimates of y_a and y_b based on $\mathbf{y}_c$ as

$$\hat{y}_{a/c} = E[y_a \mathbf{y}_c^T] E[\mathbf{y}_c \mathbf{y}_c^T]^{-1} \mathbf{y}_c = -\bar{\boldsymbol{\alpha}}^T \mathbf{y}_c\,, \quad \hat{y}_{b/c} = E[y_b \mathbf{y}_c^T] E[\mathbf{y}_c \mathbf{y}_c^T]^{-1} \mathbf{y}_c = -\tilde{\boldsymbol{\beta}}^T \mathbf{y}_c$$

and the estimation errors

$$\bar{e}_a = \bar{\mathbf{a}}^T \bar{\mathbf{y}} = y_a - \hat{y}_{a/c}\,, \quad \tilde{e}_b = \tilde{\mathbf{b}}^T \tilde{\mathbf{y}} = y_b - \hat{y}_{b/c} \tag{1.8.52}$$

Thus, $\bar{e}_a$ and $\tilde{e}_b$ represent what is left of y_a and y_b after we project out their dependence on the intermediate vector $\mathbf{y}_c$. The direct influence of y_a on y_b, with the effect of $\mathbf{y}_c$ removed, is measured by the correlation $E[\bar{e}_a \tilde{e}_b]$. This correlation is equal to the quantity Δ defined in Eq. (1.8.46). This follows from Eq. (1.8.43)

$$\Delta_a = \bar{\mathbf{a}}^T \mathbf{r}_b = \bar{\mathbf{a}}^T E[y_b \bar{\mathbf{y}}] = E[y_b (\bar{\mathbf{a}}^T \bar{\mathbf{y}})] = E[y_b \bar{e}_a]$$

similarly,

$$\Delta_b = \tilde{\mathbf{b}}^T \mathbf{r}_a = \tilde{\mathbf{b}}^T E[y_a \tilde{\mathbf{y}}] = E[y_a (\tilde{\mathbf{b}}^T \tilde{\mathbf{y}})] = E[y_a \tilde{e}_b]$$

Now, because $\bar{e}_a$ is orthogonal to $\mathbf{y}_c$ and $\hat{y}_{b/c}$ is a linear combination of $\mathbf{y}_c$, it follows that $E[\hat{y}_{b/c} \bar{e}_a] = 0$. Similarly, because $\tilde{e}_b$ is orthogonal to $\mathbf{y}_c$ and $\hat{y}_{a/c}$ is linearly related to $\mathbf{y}_c$, it follows that $E[\hat{y}_{a/c} \tilde{e}_b] = 0$. Thus,

$$\Delta_a = E[y_b \bar{e}_a] = E[(y_b - \hat{y}_{b/c}) \bar{e}_a] = E[\tilde{e}_b \bar{e}_a]$$

$$\Delta_b = E[y_a \tilde{e}_b] = E[(y_a - \hat{y}_{a/c}) \tilde{e}_b] = E[\bar{e}_a \tilde{e}_b]$$

Therefore, Δ_a and Δ_b are equal

$$\Delta_a = \Delta_b = E[\bar{e}_a \tilde{e}_b] \tag{1.8.53}$$

This is an alternative proof of Eq. (1.8.46). It follows that γ_a and γ_b are normalized PARCOR coefficients in the sense of Sec. 1.7:

$$\gamma_b = \frac{E[\bar{e}_a \tilde{e}_b]}{E[\tilde{e}_b^2]}\,, \quad \gamma_a = \frac{E[\tilde{e}_b \bar{e}_a]}{E[\bar{e}_a^2]} \tag{1.8.54}$$

Using the Schwarz inequality for the inner product between two random variables, namely, $|E[uv]|^2 \le E[u^2] E[v^2]$, we find the inequality

$$0 \le \gamma_a \gamma_b = \frac{E[\bar{e}_a \tilde{e}_b]^2}{E[\tilde{e}_b^2] E[\bar{e}_a^2]} \le 1 \tag{1.8.55}$$

This inequality also follows from Eq. (1.8.41) and the fact that E_a and $\bar{E}_a$ are positive quantities, both being mean square errors.

Example 1.8.4: For Example 1.8.1, compute the estimates $\hat{y}_{a/c}$ and $\hat{y}_{b/c}$ directly and compare them with the results of Example 1.8.3.

Solution: From the matrix elements of R we have $E[y_a y_b] = 1$, $E[y_b y_c] = 2$, and $E[y_c^2] = 3$. Thus,

$$\hat{y}_{a/c} = E[y_a y_c] E[y_c^2]^{-1} y_c = \frac{1}{3} y_c\,, \quad \hat{y}_{b/c} = E[y_b y_c] E[y_c^2]^{-1} y_c = \frac{2}{3} y_c$$

The corresponding errors will be

$$\bar{e}_a = y_a - \frac{1}{3} y_c = [1, -1/3]\bar{\mathbf{y}}\,, \quad \tilde{e}_b = y_b - \frac{2}{3} y_c = [-2/3, 1]\tilde{\mathbf{y}}$$

The results are identical to those of Example 1.8.3. □

Conventional Cholesky Factorizations

Equation (1.8.18) shows that the conventional Cholesky factor of R is given by the inverse matrix L^{-1}. A direct construction of the conventional Cholesky factor that avoids the computation of this inverse is as follows. Define

$$G_b = E[\mathbf{y} \mathbf{e}_b^T] \tag{1.8.56}$$

If we use $\mathbf{e}_b = L\mathbf{y}$ and $E[\mathbf{e}_b \mathbf{e}_b^T] = D_b$, it follows that

$$L G_b = L E[\mathbf{y} \mathbf{e}_b^T] = E[\mathbf{e}_b \mathbf{e}_b^T] = D_b$$

or,

$$G_b = L^{-1} D_b \tag{1.8.57}$$

Thus, G_b is a *lower-triangular* matrix. Its main diagonal consists of the diagonal entries of D_b. Solving for $L^{-1} = G_b D_b^{-1}$ and inserting in Eq. (1.8.18), we find the conventional LU factorization of R:

$$R = (G_b D_b^{-1}) D_b (D_b^{-1} G_b^T) = G_b D_b^{-1} G_b^T \tag{1.8.58}$$

Similarly, the conventional UL factorization of R is obtained from Eq. (1.8.33) by defining the *upper-triangular* matrix

$$G_a = E[\mathbf{y} \mathbf{e}_a^T] \tag{1.8.59}$$

Using $\mathbf{e}_a = U\mathbf{y}$ and $E[\mathbf{e}_a \mathbf{e}_a^T] = D_a$, we find

$$U G_a = D_a \quad \Rightarrow \quad G_a = U^{-1} D_a \tag{1.8.60}$$

which yields the conventional UL factorization of R:

$$R = U^{-1}D_aU^{-T} = G_aD_a^{-1}G_a^T$$

The columns of the matrices G_a and G_b will be referred to as the forward and backward *gapped* functions. This terminology will be justified in Chap. 12. The decomposition of G_b into its columns can be done order-recursively using the decomposition (1.8.15). We have

$$G_b = E[\mathbf{y}[\bar{\mathbf{e}}_b^T, e_b]] \equiv [\bar{G}_b, \mathbf{g}_b] \tag{1.8.61}$$

where $\bar{G}_b = E[\mathbf{y}\bar{\mathbf{e}}_b^T]$ and $\mathbf{g}_b = E[\mathbf{y}e_b]$. Similarly, using Eq. (1.8.23) we find

$$G_a = E[\mathbf{y}[e_a, \tilde{\mathbf{e}}_a^T]] \equiv [\mathbf{g}_a, \tilde{G}_a] \tag{1.8.62}$$

where $\tilde{G}_a = E[\mathbf{y}\tilde{\mathbf{e}}_a^T]$ and $\mathbf{g}_a = E[\mathbf{y}e_a]$. Motivated by the lattice recursions (1.8.50), we are led to define the lower-order gapped functions

$$\tilde{\mathbf{g}}_b = E[\mathbf{y}\tilde{e}_b], \quad \bar{\mathbf{g}}_a = E[\mathbf{y}\bar{e}_a]$$

It follows that the gapped functions $\mathbf{g}_a = E[\mathbf{y}e_a]$ and $\mathbf{g}_b = E[\mathbf{y}e_b]$ can be constructed order-recursively by the lattice-type equations

$$\begin{aligned} \mathbf{g}_a &= \bar{\mathbf{g}}_a - \gamma_b\tilde{\mathbf{g}}_b \\ \mathbf{g}_b &= \tilde{\mathbf{g}}_b - \gamma_a\bar{\mathbf{g}}_a \end{aligned} \tag{1.8.63}$$

The proof is straightforward. For example, $E[\mathbf{y}e_a] = E[\mathbf{y}(\bar{e}_a - \gamma_b\tilde{e}_b)]$. In Chap. 12 we will see that these equations are equivalent to the celebrated *Schur algorithm* for solving the linear prediction problem. In recent years, the Schur algorithm has emerged as an important signal processing tool because it admits efficient fixed-point and parallel processor implementations. Equations (1.8.63) are mathematically equivalent to the Levinson-type recursions (1.8.40). In fact, Eq. (1.8.40) can be derived from Eq. (1.8.63) as follows. Using $e_a = \mathbf{a}^T\mathbf{y}$ and $e_b = \mathbf{b}^T\mathbf{y}$, it follows that

$$\mathbf{g}_a = E[\mathbf{y}e_a] = E[\mathbf{y}(\mathbf{y}^T\mathbf{a})] = R\mathbf{a}, \quad \mathbf{g}_b = E[\mathbf{y}e_b] = E[\mathbf{y}(\mathbf{y}^T\mathbf{b})] = R\mathbf{b}$$

Similarly, we have

$$\bar{\mathbf{g}}_a = R\begin{bmatrix} \bar{\mathbf{a}} \\ 0 \end{bmatrix}, \quad \tilde{\mathbf{g}}_b = R\begin{bmatrix} 0 \\ \tilde{\mathbf{b}} \end{bmatrix} \tag{1.8.64}$$

These are easily shown. For example,

$$R\begin{bmatrix} \bar{\mathbf{a}} \\ 0 \end{bmatrix} = E[\mathbf{y}[\bar{\mathbf{y}}^T, \gamma_b]]\begin{bmatrix} \bar{\mathbf{a}} \\ 0 \end{bmatrix} = E[\mathbf{y}\bar{\mathbf{y}}^T]\bar{\mathbf{a}} = E[\mathbf{y}\bar{e}_a] = \bar{\mathbf{g}}_a$$

Therefore, the first part of Eq. (1.8.63) is equivalent to

$$R\mathbf{a} = R\begin{bmatrix} \bar{\mathbf{a}} \\ 0 \end{bmatrix} - \gamma_b R\begin{bmatrix} 0 \\ \tilde{\mathbf{b}} \end{bmatrix}$$

Equation (1.8.40) follows now by canceling out the matrix factor R. One of the essential features of the Schur algorithm is that the reflection coefficients can also be

computed from the knowledge of the lower-order gapped functions $\bar{\mathbf{g}}_a$ and $\tilde{\mathbf{g}}_b$, as follows. Using Eq. (1.8.64) and dotting Eq. (1.8.44) with the unit vectors $\mathbf{u}$ and $\mathbf{v}$, we find

$$\bar{E}_a = \mathbf{u}^T\bar{\mathbf{g}}_a, \quad \tilde{E}_b = \mathbf{v}^T\tilde{\mathbf{g}}_b, \quad \Delta = \mathbf{u}^T\tilde{\mathbf{g}}_b = \mathbf{v}^T\bar{\mathbf{g}}_a \tag{1.8.65}$$

Thus, Eq. (1.8.42) may be written as

$$\gamma_b = \frac{\mathbf{v}^T\bar{\mathbf{g}}_a}{\mathbf{v}^T\tilde{\mathbf{g}}_b}, \quad \gamma_b = \frac{\mathbf{u}^T\tilde{\mathbf{g}}_b}{\mathbf{u}^T\bar{\mathbf{g}}_a} \tag{1.8.66}$$

Summary

We have argued that the solution of the general linear estimation problem can be made more efficient by working with the decorrelated bases $\mathbf{e}_a$ or $\mathbf{e}_b$, which contain no redundancies. Linear prediction ideas come into play in this context because the linear transformations U and L that decorrelate the data vector $\mathbf{y}$ are constructible from the forward and backward linear prediction coefficients $\mathbf{a}$ and $\mathbf{b}$. Moreover, linear prediction was seen to be equivalent to the Gram-Schmidt construction and to the Cholesky factorization of the covariance matrix R. The order-recursive solutions of the linear prediction problem and the linear estimation problem, Eqs. (1.8.24) through (1.8.26), give rise to efficient lattice implementations with many desirable properties, such as robustness under coefficient quantization and modularity of structure admitting parallel VLSI implementations.

In this section, we intentionally did not make any additional assumptions about any structural properties of the covariance matrix R. To close the loop and obtain the efficient computational algorithms mentioned previously, we need to make additional assumptions on R. The simplest case is to assume that R has a Toeplitz structure. This case arises when $\mathbf{y}$ is a block of successive signal samples from a stationary time series. The Toeplitz property means that the matrix elements along each diagonal of R are the same. Equivalently, the matrix element R_{ij} depends only on the difference of the indices, that is, $R_{ij} = R(i-j)$. With respect to the subblock decomposition (1.8.5), it is easily verified that a necessary and sufficient condition for R to be Toeplitz is that

$$\tilde{R} = \bar{R}$$

This condition implies that the linear prediction solutions for $\tilde{R}$ and $\bar{R}$ must be the same, that is,

$$\tilde{\mathbf{b}} = \bar{\mathbf{b}}, \quad \tilde{\mathbf{a}} = \bar{\mathbf{a}}$$

Thus, from the forward and backward linear prediction solutions $\bar{\mathbf{a}}$ and $\bar{\mathbf{b}}$ of the lower-order Toeplitz submatrix $\bar{R}$, we first obtain $\tilde{\mathbf{b}} = \bar{\mathbf{b}}$ and then use Eq. (1.8.40) to get the linear prediction solution of the higher order matrix R. This is the essence behind Levinson's algorithm. It will be discussed further in Chap. 12.

In the nonstationary time series case, the matrix R is not Toeplitz. Even then one can obtain some useful results by means of the so-called *shift-invariance* property. In this case, the data vector $\mathbf{y}$ consists of successive signal samples starting at some arbitrary

sampling instant n

$$\mathbf{y}(n)=\begin{bmatrix} y_n \\ y_{n-1} \\ \vdots \\ y_{n-M+1} \\ y_{n-M} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{y}}(n) \\ y_{n-M} \end{bmatrix} = \begin{bmatrix} y_n \\ \tilde{\mathbf{y}}(n) \end{bmatrix}$$

It follows that

$$\bar{\mathbf{y}}(n)=\begin{bmatrix} y_n \\ \vdots \\ y_{n-M+1} \end{bmatrix}, \quad \tilde{\mathbf{y}}(n)=\begin{bmatrix} y_{n-1} \\ \vdots \\ y_{n-M} \end{bmatrix}, \quad \text{or,} \quad \tilde{\mathbf{y}}(n)=\bar{\mathbf{y}}(n-1)$$

This implies that $\tilde{R}(n)=\bar{R}(n-1)$, and therefore

$$\tilde{\mathbf{a}}(n)=\bar{\mathbf{a}}(n-1)\,, \quad \tilde{\mathbf{b}}(n)=\bar{\mathbf{b}}(n-1)$$

Thus, order updating is coupled with time updating. These results are used in the development of the fast recursive least-squares adaptive filters, discussed in Chap. 16.

1.9 Random Signals

A random signal (random process, or stochastic process) is defined as a sequence of random variables $\{x_0, x_1, x_2, \ldots, x_n, \ldots\}$ where the index n is taken to be the time. The statistical description of so many random variables is very complicated since it requires knowledge of all the joint densities

$$p(x_0, x_1, x_2, \ldots, x_n)\,, \quad \text{for } n = 0, 1, 2, \ldots$$

If the mean $E[x_n]$ of the random signal is not zero, it can be removed by redefining a new signal $x_n - E[x_n]$. From now on, we will assume that this has been done, and shall work with zero-mean random signals. The *autocorrelation function* is defined as

$$R_{xx}(n,m)=E[x_n x_m]\,, \quad n,m = 0, 1, 2, \ldots$$

Sometimes it will be convenient to think of the random signal as a (possibly infinite) random vector $\mathbf{x} = [x_0, x_1, x_2, \ldots, x_n, \ldots]^T$, and of the autocorrelation function as a (possibly infinite) matrix $R_{xx} = E[\mathbf{x}\mathbf{x}^T]$. R_{xx} is positive semi-definite and symmetric. The autocorrelation function may also be written as

$$R_{xx}(n+k,n)=E[x_{n+k}x_n] \tag{1.9.1}$$

It provides a *measure* of the influence of the sample x_n on the sample x_{n+k}, which lies in the future (if $k > 0$) by k units of time. The relative time separation k of the two samples is called the *lag*.

If the signal x_n is *stationary* (or wide-sense stationary), then the above average is independent of the absolute time n, and is a function only of the relative lag k; abusing somewhat the above notation, we may write in the case:

$$R_{xx}(k)=E[x_{n+k}x_n]=E[x_{n'+k}x_{n'}] \qquad \text{(autocorrelation)} \tag{1.9.2}$$

In other words, the self-correlation properties of a stationary signal x_n are same on the average, regardless of when this average is computed. In a way, the stationary random signal x_n looks the same for all times. In this sense, if we take two different blocks of data of length N, as shown in Fig. 1.9.1, we should expect the *average properties*, such as means and autocorrelations, extracted from these blocks of data to be roughly the same. The relative time separation of the two blocks as a whole should not matter.

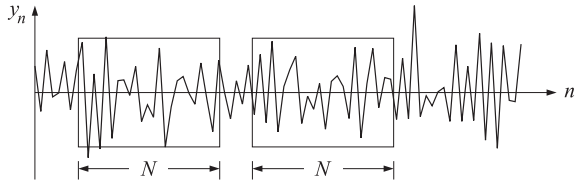


Fig. 1.9.1 Blocks of data from a stationary signal.

A direct consequence of stationarity is the *reflection-invariance* of the autocorrelation function $R_{xx}(k)$ of Eq. (1.9.2):

$$R_{xx}(k)=E[x_{n+k}x_n]=R_{xx}(-k) \tag{1.9.3}$$

One way to introduce a systematization of the various types of random signals is the Markov classification into zeroth-order Markov, first-order Markov, and so on. The simplest possible random signal is the zeroth-order Markov, or *purely random* signal, defined by the requirement that all the (zero-mean) random variables x_n be independent of each other and arise from a common density $p(x)$; this implies

$$p(x_0, x_1, x_2, \ldots, x_n) = p(x_0)\,p(x_1)\,p(x_2)\cdots p(x_n)\cdots$$

$$R_{xx}(n,m)=E[x_n x_m]=0\,, \quad \text{for} \quad n \neq m$$

Such a random signal is stationary. The quantity $R_{xx}(n,n)$ is independent of n, and represents the variance of each sample:

$$R_{xx}(0)=E[x_n^2]=\sigma_x^2$$

In this case, the autocorrelation function $R_{xx}(k)$ may be expressed compactly as

$$R_{xx}(k)=E[x_{n+k}x_n]=\sigma_x^2\delta(k) \tag{1.9.4}$$

A purely random signal has no memory, as can be seen from the property

$$p(x_n, x_{n-1}) = p(x_n)\,p(x_{n-1}) \quad \text{or,} \quad p(x_n|x_{n-1}) = p(x_n)$$

that is, the occurrence of x_{n-1} at time instant $n-1$ does not in any way affect, or restrict, the values of x_n at the next time instant. Successive signal values are entirely independent of each other. Past values do not influence future values. No memory is retained from sample to sample; the next sample will take a value regardless of the value that the previous sample has already taken. Since successive samples are random, such a signal will exhibit very rapid time variations. But it will also exhibit slow time variations. Such time variations are best discussed in the frequency domain. This will lead directly to frequency concepts, power spectra, periodograms, and the like. It is expected that a purely random signal will contain all frequencies, from the very low to the very high, in equal proportions (white noise).

The next least complicated signal is the first-order Markov signal, which has memory only of one sampling instant. Such a signal remembers only the previous sample. It is defined by the requirement that

$$p(x_n|x_{n-1},x_{n-1},\ldots,x_0) = p(x_n|x_{n-1})$$

which states that x_n may be influenced directly only by the previous sample value x_{n-1}, and not by the samples $x_{n-2},\ldots,x_0$ that are further in the past. The complete statistical description of such random signal is considerably simplified. It is sufficient to know only the marginal densities $p(x_n)$ and the conditional densities $p(x_n|x_{n-1})$. Any other joint density may be constructed in terms of these. For instance,

$$\begin{aligned} p(x_3,x_2,x_1,x_0) &= p(x_3|x_2,x_1,x_0)\,p(x_2,x_1,x_0) && \text{(by Bayes' rule)}\\ &= p(x_3|x_2)\,p(x_2,x_1,x_0) && \text{(by the Markov property)}\\ &= p(x_3|x_2)\,p(x_2|x_1,x_0)\,p(x_1,x_0)\\ &= p(x_3|x_2)\,p(x_2|x_1)\,p(x_1,x_0)\\ &= p(x_3|x_2)\,p(x_2|x_1)\,p(x_1|x_0)\,p(x_0) \end{aligned}$$

1.10 Power Spectrum and Its Interpretation

The *power spectral density* of a stationary random signal x_n is defined as the double-sided z-transform of its autocorrelation function

$$S_{xx}(z) = \sum_{k=-\infty}^{\infty} R_{xx}(k)\,z^{-k} \tag{1.10.1}$$

where $R_{xx}(k)$ is given by Eq. (1.9.2). If $R_{xx}(k)$ is strictly stable, the region of convergence of $S_{xx}(z)$ will include the unit circle in the complex z-plane. This allows us to define the *power spectrum* $S_{xx}(\omega)$ of the random signal x_n by setting $z = e^{j\omega}$ in Eq. (1.10.1). Abusing the notation somewhat, we have in this case

$$S_{xx}(\omega) = \sum_{k=-\infty}^{\infty} R_{xx}(k)\,e^{-j\omega k} \tag{1.10.2}$$

This quantity conveys very useful information. It is a measure of the frequency content of the signal x_n and of the distribution of the power of x_n over frequency. To

see this, consider the inverse z-transform

$$R_{xx}(k) = \oint_{\text{u.c.}} S_{xx}(z)\,z^k\,\frac{dz}{2\pi jz} \tag{1.10.3}$$

where, since $R_{xx}(k)$ is stable, the integration contour may be taken to be the unit circle. Using $z = e^{j\omega}$, we find for the integration measure

$$\frac{dz}{2\pi jz} = \frac{d\omega}{2\pi}$$

Thus, Eq. (1.10.3) may also be written as an inverse Fourier transform

$$R_{xx}(k) = \int_{-\pi}^{\pi} S_{xx}(\omega)\,e^{j\omega k}\,\frac{d\omega}{2\pi} \tag{1.10.4}$$

In particular, the variance of x_n can be written as

$$R_{xx}(0) = \sigma_x^2 = E[x_n^2] = \int_{-\pi}^{\pi} S_{xx}(\omega)\,\frac{d\omega}{2\pi} \tag{1.10.5}$$

Since the quantity $E[x_n^2]$ represents the average *total power* contained in x_n, it follows that $S_{xx}(\omega)$ will represent the *power per unit frequency interval*. A typical power spectrum is depicted in Fig. 1.10.1. As suggested by this figure, it is possible for the power to be mostly concentrated about some frequencies and not about others. The area under the curve represents the total power of the signal x_n.

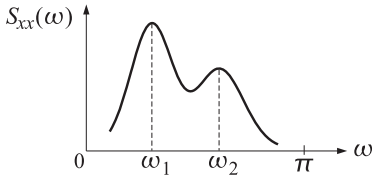


Fig. 1.10.1 Typical power spectrum.

If x_n is an uncorrelated (white-noise) random signal with a delta-function autocorrelation, given by Eq. (1.9.4), it will have a *flat* power spectrum with power level equal to the variance σ_x^2:

$S_{xx}(\omega) = \sigma_x^2$

Another useful concept is that of the *cross-correlation* and *cross-spectrum* between two stationary random sequences x_n and y_n. These are defined by

$$R_{yx}(k) = E[y_{n+k}x_n]\,, \quad S_{yx}(z) = \sum_{k=-\infty}^{\infty} R_{yx}(k)\,z^{-k} \tag{1.10.6}$$

Using stationarity, it is easy to show the reflection symmetry property

$$R_{yx}(k) = R_{xy}(-k) \tag{1.10.7}$$

that is analogous to Eq. (1.9.3). In the z-domain, the reflection symmetry properties (1.9.3) and (1.10.7) are translated into:

$$S_{xx}(z) = S_{xx}(z^{-1}), \quad S_{yx}(z) = S_{xy}(z^{-1}) \tag{1.10.8}$$

respectively; and also

$$S_{xx}(\omega) = S_{xx}(-\omega), \quad S_{yx}(\omega) = S_{xy}(-\omega) \tag{1.10.9}$$

1.11 Sample Autocorrelation and the Periodogram

From now on we will work mostly with stationary random signals. If a block of N signal samples is available, we will assume that it is a segment from a stationary signal. The length N of the available data segment is an important consideration. For example, in computing frequency spectra, we know that high resolution in frequency requires a long record of data. However, if the record is too long the assumption of stationarity may no longer be justified. This is the case in many applications, as for example in speech and EEG signal processing. The speech waveform does not remain stationary for long time intervals. It may be assumed to be stationary only for short time intervals. Such a signal may be called *piece-wise stationary*. If it is divided into short segments of duration of approximately 20–30 milliseconds, then the portion of speech within each segment may be assumed to be a segment from a stationary signal. A typical piece-wise stationary signal is depicted in Fig. 1.11.1.

Fig. 1.11.1 Piece-wise stationary signal.

The main reason for assuming stationarity, or piece-wise stationarity, is that most of our methods of handling random signals depend heavily on this assumption. For example, the statistical autocorrelations based on the ensemble averages (1.9.2) may be replaced in practice by *time averages*. This can be justified only if the signals are stationary (actually, they must be ergodic). If the underlying signal processes are not stationary (and therefore definitely are not ergodic) we cannot use time averages. If a signal is piece-wise stationary and divided into stationary blocks, then for each such block, ensemble averages may be replaced by time averages. The time average approximation of an autocorrelation function is called the *sample autocorrelation* and is defined

as follows: Given a block of length N of measured signal samples

$$\boxed{y_0, y_1, y_2, \ldots, y_{N-1}}$$

define

$$\hat{R}_{yy}(k) = \frac{1}{N} \sum_{n=0}^{N-1-k} y_{n+k} y_n, \quad \text{for} \quad 0 \le k \le N-1 \tag{1.11.1}$$

and

$$\hat{R}_{yy}(k) = \hat{R}_{yy}(-k), \quad \text{for} \quad -(N-1) \le k \le -1$$

The function **acf** takes as inputs two length-N signal blocks y_n, x_n, $n = 0, 1, \ldots, N-1$, and computes their sample cross-correlation defined as

$$\hat{R}_{yx}(k) = \frac{1}{N} \sum_{k=0}^{N-1-k} y_{n+k} x_n, \quad k = 0, 1, \ldots, N-1$$

This function may be used to compute either auto-correlations or cross-correlations. The *periodogram* is defined as the (double-sided) z-transform of the sample autocorrelation

$$\hat{S}_{yy}(z) = \sum_{k=-(N-1)}^{N-1} \hat{R}_{yy}(k) z^{-k} \tag{1.11.2}$$

It may be thought of as an approximation (estimate) of the true power spectral density $S_{yy}(z)$. It is easily shown that the periodogram may be expressed in terms of the z-transform of the data sequence itself, as

$$\hat{S}_{yy}(z) = \frac{1}{N} Y(z) Y(z^{-1}) \tag{1.11.3}$$

where

$$Y(z) = \sum_{n=0}^{N-1} y_n z^{-n} \tag{1.11.4}$$

As a concrete example, consider a length-3 signal $\mathbf{y} = [y_0, y_1, y_2]^T$. Then,

$$\begin{aligned} Y(z)Y(z^{-1}) &= (y_0 + y_1 z^{-1} + y_2 z^{-2})(y_0 + y_1 z + y_2 z^2) \\ &= (y_0^2 + y_1^2 + y_2^2) + (y_0 y_1 + y_1 y_2)(z^{-1} + z) + (y_0 y_2)(z^{-2} + z^2) \end{aligned}$$

from which we extract the inverse z-transform

$$\begin{aligned} \hat{R}_{xx}(0) &= \frac{1}{3}(y_0^2 + y_1^2 + y_2^2) \\ \hat{R}_{xx}(-1) = \hat{R}_{xx}(1) &= \frac{1}{3}(y_0 y_1 + y_1 y_2) \\ \hat{R}_{xx}(-2) = \hat{R}_{xx}(2) &= \frac{1}{3}(y_0 y_2) \end{aligned}$$

These equations may also be written in a nice matrix form, as follows

$$\underbrace{\begin{bmatrix} \hat{R}_{xx}(0) & \hat{R}_{xx}(1) & \hat{R}_{xx}(2) \\ \hat{R}_{xx}(1) & \hat{R}_{xx}(0) & \hat{R}_{xx}(1) \\ \hat{R}_{xx}(2) & \hat{R}_{xx}(1) & \hat{R}_{xx}(0) \end{bmatrix}}_{\hat{R}_{yy}} = \frac{1}{3} \underbrace{\begin{bmatrix} y_0 & y_1 & y_2 & 0 & 0 \\ 0 & y_0 & y_1 & y_2 & 0 \\ 0 & 0 & y_0 & y_1 & y_2 \end{bmatrix}}_{Y^T} \underbrace{\begin{bmatrix} y_0 & 0 & 0 \\ y_1 & y_0 & 0 \\ y_2 & y_1 & y_0 \\ 0 & y_2 & y_1 \\ 0 & 0 & y_2 \end{bmatrix}}_{Y}$$

or,

$$\hat{R}_{yy} = \frac{1}{3} Y^T Y$$

The matrix $\hat{R}_{yy}$ on the left is called the sample autocorrelation matrix. It is a *Toeplitz matrix*, that is, it has the same entry in each diagonal. The right hand side also shows that the autocorrelation matrix is positive definite. In the general case of a length-N sequence y_n, the matrix Y has N columns, each a down-shifted (delayed) version of the previous one, corresponding to a total of $N-1$ delays. This requires the length of each column to be $N+(N-1)$, that is, there are $2N-1$ rows. We will encounter again this matrix factorization in the least-squares design of waveshaping filters.

The sample autocorrelation may also be thought of as *ordinary convolution*. Note that $Y(z^{-1})$ represents the z-transform the original signal $\mathbf{y} = [y_0, y_1, \ldots, y_{N-1}]^T$ reflected about the time origin. The reflected signal may be made causal by a delay of $N-1$ units of time. The reflected-delayed signal has some significance, and is known as the *reversed signal*. Its z-transform is the *reverse polynomial* of $Y(z)$

$$Y^R(z) = z^{-(N-1)} Y(z^{-1})$$

$$\begin{array}{lccccccccccl} [& 0 & 0 & \cdots & 0 & y_0 & y_1 & \cdots & y_{N-2} & y_{N-1} &] & = & \text{original} \\ [& y_{N-1} & y_{N-2} & \cdots & y_1 & y_0 & 0 & \cdots & 0 & 0 &] & = & \text{reflected} \\ [& 0 & 0 & \cdots & 0 & y_{N-1} & y_{N-2} & \cdots & y_1 & y_0 &] & = & \text{reversed} \end{array}$$

The periodogram is expressed then in the form

$$\hat{S}_{xx}(z) = \frac{1}{N} Y(z) Y(z^{-1}) = \frac{1}{N} Y(z) Y^R(z) z^{N-1}$$

which implies that $\hat{R}_{yy}(k)$ may be obtained by convolving the original data sequence with the reversed sequence and then advancing the result in time by $N-1$ time units. This is seen by the following convolution table.

	y_0	y_1	y_2
y_2	y_2y_0	y_2y_1	y_2y_2
y_1	y_1y_0	y_1y_1	y_1y_2
y_0	y_0y_0	y_0y_1	y_0y_2

The *periodogram spectrum* is obtained by substituting $z = e^{j\omega}$

$$\hat{S}_{yy}(\omega) = \frac{1}{N} |Y(\omega)|^2 = \frac{1}{N} \left| \sum_{n=0}^{N-1} y_n e^{-j\omega n} \right|^2 \qquad (1.11.5)$$

The periodogram spectrum (1.11.5) may be computed efficiently using FFT methods. The digital frequency ω in units of [radians/sample] is related to the physical frequency f in [Hz] by

$$\omega = 2\pi f T = \frac{2\pi f}{f_s}$$

where f_s is the sampling rate, and $T = 1/f_s$, the time interval between samples. The frequency resolution afforded by a length-N sequence is

$$\Delta\omega = \frac{2\pi}{N}, \quad \text{or,} \quad \Delta f = \frac{f_s}{N} = \frac{1}{NT} = \frac{1}{T_R} \quad [\text{Hz}]$$

where $T_R = NT$ is the duration of the data record in seconds. The periodogram spectrum suffers from two major drawbacks. First, the rectangular windowing of the data segment introduces significant *sidelobe leakage*. This can cause misinterpretation of sidelobe spectral peaks as being part of the true spectrum. And second, it is well-known that the periodogram spectrum is *not* a good (consistent) estimator of the true power spectrum $S_{yy}(\omega)$.

The development of methods to improve on the periodogram is the subject of *classical spectral analysis* [9–19]. We just mention, in passing, one of the most popular of such methods, namely, Welch's method [20]. The given data record of length N is subdivided into K shorter segments which may be overlapping or non-overlapping. If they are non-overlapping then each will have length $M = N/K$; if they are 50% overlapping then $M = 2N/(K+1)$. Each such segment is then windowed by a length-M data window, such as a Hamming window. The window reduces the sidelobe frequency leakage at the expense of resolution. The window $w(n)$ is typically normalized to have unit average energy, that is, $(1/M)\sum_{n=0}^{M-1} w^2(n) = 1$. The periodogram of each windowed segment is then computed by FFT methods and the K periodograms are averaged together to obtain the spectrum estimate

$$S(\omega) = \frac{1}{K} \sum_{i=1}^{K} S_i(\omega)$$

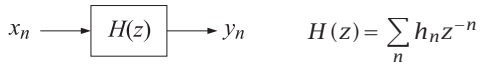
where $S_i(\omega)$ is the periodogram of the ith segment. The above subdivision into segments imitates ensemble averaging, and therefore, it results in a spectrum estimate of improved statistical stability. However, since each periodogram is computed from a length-M sequence, the frequency resolution is reduced from $\Delta\omega = 2\pi/N$ to roughly $\Delta\omega = 2\pi/M$ (for a well-designed window). Therefore, to maintain high frequency resolution (large M), as well as improved statistical stability of the spectrum estimate (large K), a long data record $N = MK$ is required—a condition that can easily come into conflict with stationarity. The so-called "modern methods" of spectrum estimation, which are based on parametric signal models, can provide high resolution spectrum estimates from short data records.

1.12 Filtering of Stationary Random Signals

In this section, we discuss the effect of linear filtering on random signals. The results are very basic and useful in suggesting guidelines for the design of signal processing

systems for many applications, such as noise reduction, signal extraction, parametric spectrum estimation, and so on.

Suppose a stationary random signal x_n is sent into a linear filter defined by a transfer function $H(z)$, resulting in the the output random signal y_n



$$H(z) = \sum_n h_n z^{-n}$$

We would like to derive relationships between the autocorrelation functions of the input and output signals, and also between the corresponding power spectra. We assume, for now, that the signals x_n, y_n, h_n are real-valued. Using the input/output filtering equation in the z-domain,

$$\boxed{Y(z) = H(z)X(z)} \tag{1.12.1}$$

we determine first a relationship between the *periodograms* of the input and output signals. From the factorization of Eq. (1.11.3) and dropping the factor $1/N$ for convenience, we find

$$\begin{aligned} \hat{S}_{yy}(z) &= Y(z)Y(z^{-1}) \\ &= H(z)X(z)H(z^{-1})X(z^{-1}) = H(z)H(z^{-1})X(z)X(z^{-1}) \\ &= H(z)H(z^{-1})\hat{S}_{xx}(z) = S_{hh}(z)\hat{S}_{xx}(z) \end{aligned} \tag{1.12.2}$$

where we used the notation $S_{hh}(z) = H(z)H(z^{-1})$. This quantity is the z-transform of the *autocorrelation function* of the filter, that is,

$$\boxed{S_{hh}(z) = H(z)H(z^{-1}) = \sum_{k=-\infty}^{\infty} R_{hh}(k)z^{-k}} \tag{1.12.3}$$

where $R_{hh}(k)$ is defined as

$$\boxed{R_{hh}(k) = \sum_n h_{n+k}h_n} \quad \text{(filter autocorrelation function)} \tag{1.12.4}$$

Equation (1.12.3) is easily verified by writing,

$$R_{hh}(k) = \sum_{i,j} h_i h_j \delta(k - (i-j))$$

and taking z-transforms, or by writing $R_{hh}(k) = \sum_n h_{k+n}h_n = \sum_n h_{k-n}h_{-n}$, which is recognized as the convolution between the signals h_n and h_{-n} whose z-transforms are $H(z)$ and $H(z^{-1})$, respectively.

Taking inverse z-transforms of Eq. (1.12.2), we obtain the time-domain equivalent relationships between the input and output sample autocorrelations

$$\hat{R}_{yy}(k) = \sum_{m=-\infty}^{\infty} R_{hh}(k)\hat{R}_{xx}(k-m) = \text{convolution of } R_{hh} \text{ with } \hat{R}_{xx} \tag{1.12.5}$$

Similarly, we find for the cross-periodograms

$$\hat{S}_{yx}(z) = Y(z)X(z^{-1}) = H(z)X(z)X(z^{-1}) = H(z)\hat{S}_{xx}(z) \tag{1.12.6}$$

and also, replacing z by z^{-1},

$$\hat{S}_{xy}(z) = \hat{S}_{xx}(z)H(z^{-1}) \tag{1.12.7}$$

The above relationships between input and output periodogram spectra and sample autocorrelations remain the same for the *statistical* autocorrelations and power spectra. In the z-domain the power spectral densities are related by

$$\boxed{\begin{aligned} S_{yy}(z) &= H(z)H(z^{-1})S_{xx}(z) \\ S_{yx}(z) &= H(z)S_{xx}(z) \\ S_{xy}(z) &= S_{xx}(z)H(z^{-1}) \end{aligned}} \tag{1.12.8}$$

Setting $z = e^{j\omega}$, we may also write Eq. (1.12.8) in terms of the corresponding power spectra:

$$\boxed{\begin{aligned} S_{yy}(\omega) &= |H(\omega)|^2 S_{xx}(\omega) \\ S_{yx}(\omega) &= H(\omega)S_{xx}(\omega) \\ S_{xy}(\omega) &= S_{xx}(\omega)H(-\omega) = S_{xx}(\omega)H(\omega)^* \end{aligned}} \tag{1.12.9}$$

In the time domain the correlation functions are related by

$$\boxed{\begin{aligned} R_{yy}(k) &= \sum_{m=-\infty}^{\infty} R_{hh}(m)R_{xx}(k-m) \\ R_{yx}(k) &= \sum_{m=-\infty}^{\infty} h_m R_{xx}(k-m) \end{aligned}} \tag{1.12.10}$$

The proofs of these are straightforward. For example, to show Eq. (1.12.10), we may use stationarity and the I/O convolutional equation,

$$y_n = \sum_m h_m x_{n-m}$$

to find

$$\begin{aligned} R_{yy}(k) &= E[y_{n+k}y_n] = E\left[\sum_i h_i x_{n+k-i} \sum_j h_j x_{n-j}\right] \\ &= \sum_{i,j} h_i h_j E[x_{n+k-i}x_{n-j}] = \sum_{i,j} h_i h_j R_{xx}(k-(i-j)) \\ &= \sum_{i,j,m} h_i h_j \delta(m-(i-j))R_{xx}(k-m) = \sum_m R_{hh}(m)R_{xx}(k-m) \end{aligned}$$

The proof assumes that the transients introduced by the filter have died out and that the output signal is stationary. For a strictly stable filter, the stationarity of the output y_n (i.e., the fact that $E[y_{n+k}y_n]$ is independent of the absolute time n), becomes valid for large times n. To see the effect of such transients, consider a causal filter and assume that the input x_n is applied starting at $n = 0$. Then, the I/O equation reads:

$$y_n = \sum_{m=0}^{n} h_m x_{n-m}$$

and the corresponding output autocorrelation function becomes (for $n, k \geq 0$):

$$E[y_{n+k}y_n] = E\left[\sum_{i=0}^{n+k} h_i x_{n+k-i} \sum_{j=0}^{n} h_j x_{n-j}\right] = \sum_{i=0}^{n+k}\sum_{j=0}^{n} h_i h_j R_{xx}(k+j-i)$$

which does have an explicit n dependence. Assuming that the filter is strictly stable, the above expression will converge to Eq. (1.12.10) in the limit of large n. A further example of this property is discussed in Sec. 1.15.

The above filtering results can be applied to the special case of a zero-mean white-noise signal x_n of variance σ_x^2, which has a delta-function autocorrelation and a flat power spectrum, as shown in Fig. 1.12.1:

$$R_{xx}(k) = E[x_{n+k}x_n] = \sigma_x^2\delta(k)\,, \quad S_{xx}(z) = \sigma_x^2 \tag{1.12.11}$$

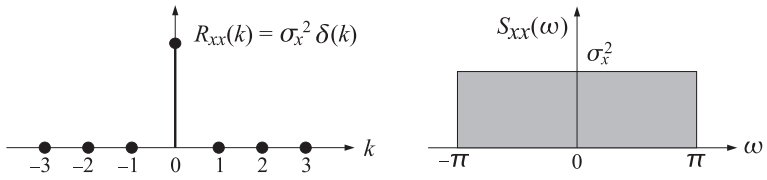


Fig. 1.12.1 Autocorrelation function and power spectrum of white noise.

Then, Eqs. (1.12.8) through (1.12.10) simplify as follows

$$\boxed{\begin{aligned} S_{yy}(z) &= H(z)H(z^{-1})\sigma_x^2 \\ S_{yx}(z) &= H(z)\sigma_x^2 \end{aligned}} \tag{1.12.12}$$

$$\boxed{\begin{aligned} S_{yy}(\omega) &= |H(\omega)|^2\sigma_x^2 \\ S_{yx}(\omega) &= H(\omega)\sigma_x^2 \end{aligned}} \tag{1.12.13}$$

$$\boxed{\begin{aligned} R_{yy}(k) &= \sigma_x^2\sum_n h_{n+k}h_n = \sigma_x^2 R_{hh}(k) \\ R_{yx}(k) &= \sigma_x^2\, h_k \end{aligned}} \tag{1.12.14}$$

The filtering operation reshapes the flat white-noise spectrum of the input signal into a shape defined by the magnitude response $|H(\omega)|^2$ of the filter, and introduces self-correlations in the output signal given by the autocorrelation of the filter. The variance σ_y^2 of the output noise signal y_n is obtained from Eq. (1.10.5), that is,

$$\sigma_y^2 = E[y_n^2] = R_{yy}(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} S_{yy}(\omega)\,d\omega = \frac{1}{2\pi}\int_{-\pi}^{\pi} |H(\omega)|^2\sigma_x^2\,d\omega \tag{1.12.15}$$

The ratio σ_y^2/σ_x^2 is a measure of whether the filter attenuates or amplifies the input noise. We will refer to it as the *noise reduction ratio* (NRR). Using Eq. (1.12.15) and Parseval's identity, we may express it in the equivalent forms:

$$\boxed{\mathcal{R} = \frac{\sigma_y^2}{\sigma_x^2} = \sum_n h_n^2 = \frac{1}{2\pi}\int_{-\pi}^{\pi} |H(\omega)|^2\,d\omega} \quad \text{(noise reduction ratio)} \tag{1.12.16}$$

Example 1.12.1: As an example, consider the first-order Markov signal y_n defined as the output of the filter

$$y_n = ay_{n-1} + \epsilon_n\,, \quad H(z) = \frac{1}{1-az^{-1}}\,, \quad |a| < 1$$

driven by white noise ϵ_n of variance σ_ϵ^2. The impulse response of the filter is

$$h_n = a^n u(n)\,, \quad u(n) = \text{unit step}$$

The output autocorrelation $R_{yy}(k)$ may be computed in two ways. First, in the time domain (assuming first that $k \geq 0$):

$$R_{yy}(k) = \sigma_\epsilon^2\sum_{n=0}^{\infty} h_{n+k}h_n = \sigma_\epsilon^2\sum_{n=0}^{\infty} a^{n+k}a^n = \sigma_\epsilon^2\, a^k\sum_{n=0}^{\infty} a^{2n} = \frac{\sigma_\epsilon^2 a^k}{1-a^2}$$

And second, in the z-domain using power spectral densities and inverse z-transforms (again take $k \geq 0$):

$$S_{yy}(z) = H(z)H(z^{-1})\sigma_\epsilon^2 = \frac{\sigma_\epsilon^2}{(1-az^{-1})(1-az)}$$

$$R_{yy}(k) = \oint_{\text{u.c}} S_{yy}(z)\,z^k\frac{dz}{2\pi jz} = \oint_{\text{u.c.}} \frac{\sigma_\epsilon^2 z^k}{(z-a)(1-az)}\frac{dz}{2\pi j}$$

$$= (\text{Residue at } z = a) = \frac{\sigma_\epsilon^2 a^k}{1-a^2}$$

In particular, we verify the following results to be used later:

$$R_{yy}(0) = \frac{\sigma_\epsilon^2}{1-a^2}\,, \quad R_{yy}(1) = \frac{\sigma_\epsilon^2 a}{1-a^2} = aR_{yy}(0)$$

$$a = \frac{R_{yy}(1)}{R_{yy}(0)}\,, \quad \sigma_\epsilon^2 = (1-a^2)R_{yy}(0)$$

It is interesting to note the exponentially decaying nature of $R_{yy}(k)$ with increasing lag k, as shown in Fig. 1.12.2.

Correlations exist between successive samples due the indirect influence of a given sample y_n on all future samples, as propagated by the difference equation. In going from one sampling instant to the next, the difference equation scales y_n by a factor a; therefore, we expect the correlations to decrease exponentially with increasing lag. □

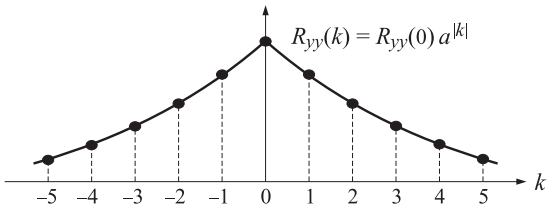


Fig. 1.12.2 Exponentially decaying autocorrelation.

Whenever the autocorrelation drops off very fast with increasing lag, it can be taken as an indication that there exists a stable difference equation model for the random signal. However, not all random signals have exponentially decaying autocorrelations. For example, a pure sinusoid with random phase

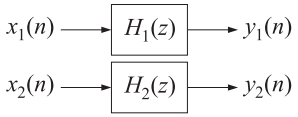
$$y_n = A\cos(\omega_0 n + \phi)$$

where ϕ is a uniformly-distributed random phase, has autocorrelation

$$R_{yy}(k) = \frac{1}{2}A^2\cos(\omega_0 k)$$

which never dies out. A particular realization of the random variable ϕ defines the entire realization of the time series y_n. Thus, as soon as ϕ is fixed, the entire y_n is fixed. Such random signals are called *deterministic*, since a few past values—e.g., three samples—of y_n are sufficient to determine all future values of y_n.

Finally we note that all of the filtering equations in Eqs. (1.12.8)-(1.12.10) can be considered to be special cases of the following more general result involving two filters $H_1(z)$ and $H_2(z)$ and two stationary input random signals $x_1(n)$ and $x_2(n)$, resulting in the output signals $y_1(n)$ and $y_2(n)$ as shown below:



Then, the corresponding cross-power spectral density of the output signals is given by:

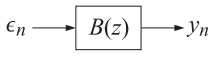
$$\boxed{S_{y_1y_2}(z) = H_1(z)H_2(z^{-1})S_{x_1x_2}(z)} \tag{1.12.17}$$

where $S_{x_1x_2}(z)$ is the z-transform of $R_{x_1x_2}(k) = E[x_1(n+k)\,x_2(n)]$, etc.

1.13 Random Signal Models and Their Uses

Models that provide a characterization of the properties and nature of random signals are of primary importance in the design of optimum signal processing systems. This section offers an overview of such models and outlines their major applications. Many of the ideas presented here will be developed in greater detail in later chapters.

One of the most useful ways to model a random signal [21] is to consider it as being the *output* of a *causal and stable* linear filter $B(z)$ that is driven by a *stationary uncorrelated* (white-noise) sequence ϵ_n,



$$B(z) = \sum_{n=0}^{\infty} b_n z^{-n}$$

where $R_{\epsilon\epsilon}(k) = E[\epsilon_{n+k}\epsilon_n] = \sigma_\epsilon^2\delta(k)$. Assuming a causal input sequence ϵ_n, the output random signal y_n is obtained by convolving ϵ_n with the filter's impulse response b_n:

$$y_n = \sum_{i=0}^{n} b_{n-i}\epsilon_i \tag{1.13.1}$$

The stability of the filter $B(z)$ is essential as it guarantees the stationarity of the sequence y_n. This point will be discussed later on. By readjusting, if necessary, the value of σ_ϵ^2 we may assume that $b_0 = 1$. Then Eq. (1.13.1) corresponds exactly to the Gram-Schmidt form of Eqs. (1.6.15) and (1.6.16), where the matrix elements b_{ni} are given in terms of the impulse response of the filter $B(z)$:

$$b_{ni} = b_{n-i} \tag{1.13.2}$$

In this case, the structure of the matrix B is considerably simplified. Writing the convolutional equation (1.13.1) in matrix form

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ b_1 & 1 & 0 & 0 & 0 \\ b_2 & b_1 & 1 & 0 & 0 \\ b_3 & b_2 & b_1 & 1 & 0 \\ b_4 & b_3 & b_2 & b_1 & 1 \end{bmatrix} \begin{bmatrix} \epsilon_0 \\ \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \end{bmatrix} \tag{1.13.3}$$

we observe that the first column of B is the impulse response b_n of the filter. Each subsequent column is a down-shifted (delayed) version of the previous one, and each diagonal has the same entry (i.e., B is a Toeplitz matrix). The lower-triangular nature of B is equivalent to the assumed *causality* of the filter $B(z)$.

Such signal models are quite general. In fact, there is a general theorem by Wold that essentially guarantees the *existence* of such models for any stationary signal y_n [22,23]. Wold's construction of $B(z)$ is none other than the Gram-Schmidt construction of the orthogonalized basis ϵ_n. However, the practical usage of such models requires further that the transfer function $B(z)$ be *rational*, that is, the ratio of two polynomials in z^{-1}. In this case, the I/O convolutional equation (1.13.1) is most conveniently expressed as a *difference equation*.

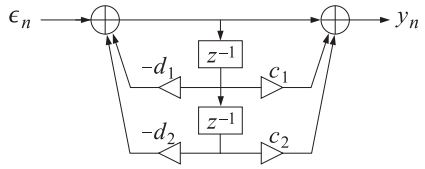
Example 1.13.1: Suppose

$$B(z) = \frac{1 + c_1z^{-1} + c_2z^{-2}}{1 + d_1z^{-1} + d_2z^{-2}} \tag{1.13.4}$$

Then Eq. (1.13.1) is equivalent to the difference equation

$$y_n = -d_1y_{n-1} - d_2y_{n-2} + \epsilon_n + c_1\epsilon_{n-1} + c_2\epsilon_{n-2} \tag{1.13.5}$$

which may be realized as follows



The filter $B(z)$ is called a *synthesis filter* and may be thought of as a random signal generator, or a signal model, for the random signal y_n. The numerator and denominator coefficients of the filter $B(z)$, and the variance σ_ϵ^2 of the input white noise, are referred to as the *model parameters*. For instance, in Example 1.13.1 the model parameters are $\{c_1, c_2, d_1, d_2, \sigma_\epsilon^2\}$.

Such parametric models have received a lot of attention in recent years. They are very common in speech and geophysical signal processing, image processing, EEG signal processing, spectrum estimation, data compression, and other time series analysis applications.

How are such models used? One of the main objectives in such applications has been to develop appropriate *analysis procedures* for extracting the model parameters on the basis of a given set of samples of the signal y_n. This is a *system identification* problem. The analysis procedures are designed to provide effectively the *best fit* of the data samples to a particular model. The procedures typically begin with a measured block of signal samples $\{y_0, y_1, \ldots, y_N\}$—also referred to as an *analysis frame*—and through an appropriate analysis algorithm extract *estimates* of the model parameters. This is depicted in Fig. 1.13.1.

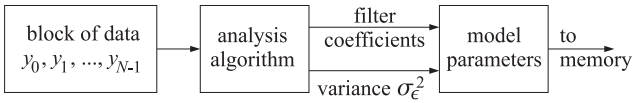


Fig. 1.13.1 Analysis procedure.

The given frame of samples $\{y_0, y_1, \ldots, y_N\}$ is *represented* now by the set of model parameters extracted from it. Following the analysis procedure, the resulting model may be used in a variety of ways. The four major uses of such models are in:

1. Signal synthesis
2. Spectrum estimation
3. Signal classification
4. Data compression

We will discuss each of these briefly. To synthesize a particular realization of the random signal y_n, it is only necessary to recall from memory the appropriate model parameters, generate a random uncorrelated sequence ϵ_n having variance σ_ϵ^2, and send it through the filter $B(z)$. Such uncorrelated sequence may be *computer-generated* using a standard random number generator function. The synthetic signal will appear at the output of the filter. This is shown in Fig. 1.13.2.

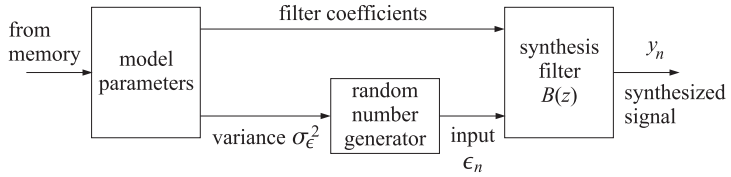


Fig. 1.13.2 Signal synthesis.

This is the basic principle behind most speech synthesis systems. In speech, the synthesis filter $B(z)$ represents a model of the transfer characteristics of the *vocal tract* considered as an acoustic tube. A typical analysis frame of speech has duration of 20 msec. If sampled at a 10-kHz sampling rate, it will consist of $N = 200$ samples. To synthesize a particular frame of 200 samples, the model parameters representing that frame are recalled from memory, and the synthesis filter is run for 200 sampling instances generating 200 output speech samples, which may be sent to a D/A converter. The next frame of 200 samples can be synthesized by recalling from memory *its* model parameters, and so on. Entire words or sentences can be synthesized in such a piecewise, or frame-wise, manner.

A realistic representation of each speech frame requires the specification of two additional parameters besides the filter coefficients and σ_ϵ^2, namely, the *pitch* period and a *voiced/unvoiced* (V/UV) decision. Unvoiced sounds, such as the "sh" in the word "should", have a white-noise sounding nature, and are generated by the turbulent flow of air through constrictions of the vocal tract. Such sounds may be represented adequately by the above random signal models. On the other hand, voiced sounds, such as vowels, are pitched sounds, and have a pitch period associated with them. They may be assumed to be generated by the periodic excitation of the vocal tract by a train of impulses separated by the pitch period. The vocal tract responds to each of these impulses by producing its impulse response, resulting therefore in a quasi-periodic output which is characteristic of such sounds. Thus, depending on the type of sound, the nature of the generator of the excitation input to the synthesis filter will be different, namely, it will be a *random noise generator* for unvoiced sounds, and a *pulse train generator* for voiced sounds. A typical speech synthesis system that incorporates the above features is shown in Fig. 1.13.3.

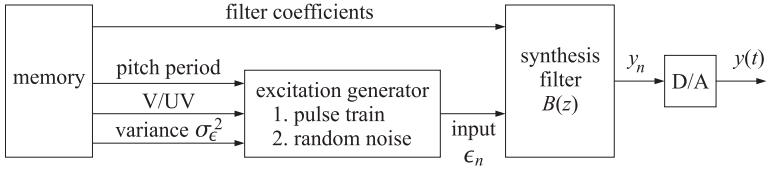


Fig. 1.13.3 Typical speech synthesis system.

Another major application of parametric models is to *spectrum estimation*. This is

based on the property that

$$S_{yy}(\omega) = \sigma_\epsilon^2 \, |B(\omega)|^2 \tag{1.13.6}$$

which will be proved later. It states that the spectral shape of the power spectrum $S_{yy}(\omega)$ of the signal y_n arises only from the spectral shape of the model filter $B(\omega)$. For example, the signal y_n generated by the model of Example 1.13.1 will have

$$S_{yy}(\omega) = \sigma_\epsilon^2 \left| \frac{1 + c_1 e^{-j\omega} + c_2 e^{-2j\omega}}{1 + d_1 e^{-j\omega} + d_2 e^{-2j\omega}} \right|^2$$

This approach to spectrum estimation is depicted in Fig. 1.13.4. The parametric approach to spectrum estimation must be contrasted with the classical approach which is based on direct computation of the Fourier transform of the available data record, that is, the periodogram spectrum, or its improvements. The classical periodogram method is shown in Fig. 1.13.5. As we mentioned in the previous section, spectrum estimates based on such parametric models tend to have much better frequency resolution properties than the classical methods, especially when the length N of the available data record is short.

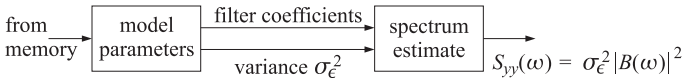


Fig. 1.13.4 Spectrum estimation with parametric models.

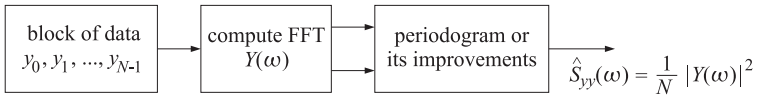


Fig. 1.13.5 Classical spectrum estimation.

In *signal classification* applications, such as speech recognition, speaker verification, or EEG pattern classification, the basic problem is to compare two available blocks of data samples and decide whether they belong to the same class or not. One of the two blocks might be a prestored and preanalyzed *reference template* against which the other block is to be compared. Instead of comparing the data records sample by sample, what are compared are the corresponding model parameters extracted from these blocks. In pattern recognition nomenclature, the vector of model parameters is the "feature vector." The closeness of the two sets of model parameters to each other is decided on the basis of an appropriate *distance measure*. We will discuss examples of distance measures for speech and EEG signals in Chap. 12. This approach to signal classification is depicted in Fig. 1.13.6.

Next, we discuss the application of such models to *data compression*. The signal synthesis method described above is a form of data compression because instead of saving the N data samples y_n as such, what are saved are the corresponding model parameters which are typically much fewer in number than N. For example, in speech

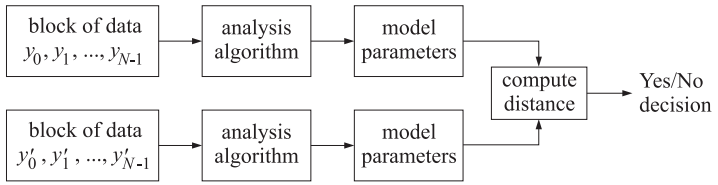


Fig. 1.13.6 Signal classification with parametric models.

synthesis systems a savings of about a factor of 20 in memory may be achieved with this approach. Indeed, as we discussed above, a typical frame of speech consists of 200 samples, whereas the number of model parameters typically needed to represent this frame is about 10 to 15. The main limitation of this approach is that the reproduction of the original signal segment is not exact but depends on the particular realization of the computer-generated input sequence ϵ_n that drives the model. Speech synthesized in such manner is still intelligible, but it has lost some of its naturalness. Such signal synthesis methods are not necessarily as successful or appropriate in all applications. For example, in image processing, if one makes a parametric model of an image and attempts to "synthesize" it by driving the model with a computer-generated uncorrelated sequence, the reproduced image will bear no resemblance to the original image.

For *exact* reproduction, both the model parameters and the entire sequence ϵ_n must be stored. This would still provide some form of data compression, as will be explained below. Such an approach to data compression is widely used in digital data transmission or digital data storage applications for all types of data, including speech and image data. The method may be described as follows: the given data record $\{y_0, y_1, \ldots, y_{N-1}\}$ is subjected to an appropriate analysis algorithm to extract the model parameters, and then the segment is filtered through the *inverse filter*,

$$A(z) = \frac{1}{B(z)} \qquad y_n \longrightarrow \boxed{A(z)} \longrightarrow \epsilon_n \tag{1.13.7}$$

to provide the sequence ϵ_n. The inverse filter $A(z)$ is also known as the *whitening filter*, the *prediction-error* filter, or the *analysis filter*. The resulting sequence ϵ_n has a compressed dynamic range relative to y_n and therefore it requires fewer number of bits for the representation of each sample ϵ_n. A quantitative measure for the data compression gain is given by the ratio $G = \sigma_y^2 / \sigma_\epsilon^2$, which is always greater than one. This can be seen easily using Eqs. (1.13.6) and (1.10.5)

$$\sigma_y^2 = \int_{-\pi}^{\pi} S_{yy}(\omega) \, \frac{d\omega}{2\pi} = \sigma_\epsilon^2 \int_{-\pi}^{\pi} |B(\omega)|^2 \, \frac{d\omega}{2\pi} = \sigma_\epsilon^2 \sum_{n=0}^{\infty} b_n^2$$

Since $b_0 = 1$, we find

$$G = \frac{\sigma_y^2}{\sigma_\epsilon^2} = \sum_{n=0}^{\infty} b_n^2 = 1 + b_1^2 + b_2^2 + \cdots \tag{1.13.8}$$

The entire sequence ϵ_n and the model parameters are then transmitted over the data link, or stored in memory. At the receiving end, the original sequence y_n may be

reconstructed exactly using the synthesis filter $B(z)$ driven by ϵ_n. This approach to data compression is depicted in Fig. 1.13.7. Not shown in Fig. 1.13.7 are the quantization and encoding operations that must be performed on ϵ_n in order to transmit it over the digital channel.

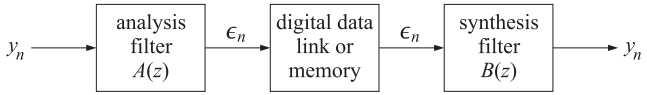


Fig. 1.13.7 Data compression.

Filtering the sequence y_n through the inverse filter requires that $A(z)$ be stable and causal. If we write $B(z)$ as the ratio of two polynomials

$$B(z) = \frac{N(z)}{D(z)} \tag{1.13.9}$$

then the stability and causality of $B(z)$ requires that the zeros of the polynomial $D(z)$ lie inside the unit circle in the complex z-plane; whereas the stability and causality of the inverse $A(z) = D(z)/N(z)$ requires the zeros of $N(z)$ to be inside the unit circle. Thus, both the poles and the zeros of $B(z)$ must be inside the unit circle. Such filters are called *minimal phase filters*. When $A(z)$ is stable and causal it may be expanded in the form

$$A(z) = \sum_{n=0}^{\infty} a_n z^{-n} = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots \tag{1.13.10}$$

and the I/O equation of Eq. (1.13.7) becomes

$$\epsilon_n = \sum_{i=0}^{n} a_i y_{n-i} = y_n + a_1 y_{n-1} + a_2 y_{n-2} + \cdots \tag{1.13.11}$$

for $n = 0, 1, 2, \ldots$. It may be written in matrix form $\boldsymbol{\epsilon} = A\mathbf{y}$ as

$$\begin{bmatrix} \epsilon_0 \\ \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ a_1 & 1 & 0 & 0 & 0 \\ a_2 & a_1 & 1 & 0 & 0 \\ a_3 & a_2 & a_1 & 1 & 0 \\ a_4 & a_3 & a_2 & a_1 & 1 \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$$

Both this matrix form and Eq. (1.13.11) are recognized as special cases of Eqs. (1.7.1) and (1.7.10). According to Eq. (1.7.11), the quantity

$$\hat{y}_{n/n-1} = -\left[a_1 y_{n-1} + a_2 y_{n-2} + \cdots + a_n y_0\right] \tag{1.13.12}$$

is the projection of y_n on the subspace spanned by $Y_{n-1} = \{y_{n-1}, y_{n-2}, \ldots, y_0\}$. Therefore, it represents the best linear estimate of y_n on the basis of (all) its past values Y_{n-1}, that is, $\hat{y}_{n/n-1}$ is the *best prediction* of y_n from its (entire) past. Equation (1.13.11) gives the corresponding prediction error $\epsilon_n = y_n - \hat{y}_{n/n-1}$. We note here an interesting connection between linear prediction concepts and signal modeling concepts [21–25], that

is, that the best linear predictor (1.13.12) determines the whitening filter $A(z)$ which, in turn, determines the generator model $B(z) = 1/A(z)$ of y_n. In other words, solving the prediction problem also solves the modeling problem.

The above modeling approach to the representation of stationary time series, and its relationship to the Gram-Schmidt construction and linear prediction was initiate by Wold and developed further by Kolmogorov [22,24].

The most general model filter $B(z)$ given in Eq. (1.13.9) is called an *autoregressive moving average* (ARMA), or a pole-zero model. Two special cases of interest are the *moving average* (MA), or all-zero models, and the *autoregressive* (AR), or all-pole models. The MA model has a nontrivial numerator only, $B(z) = N(z)$, so that $B(z)$ is a finite polynomial:

$$B(z) = 1 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_M z^{-M} \qquad \text{(MA model)}$$

The AR model has a nontrivial denominator only, $B(z) = 1/D(z)$, so that its inverse $A(z) = D(z)$ is a polynomial:

$$B(z) = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_M z^{-M}} \qquad \text{(AR model)}$$

$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_M z^{-M}$$

Autoregressive models are the most widely used models, because the analysis algorithms for extracting the model parameters $\{a_1, a_2, \ldots, a_M; \sigma_\epsilon^2\}$ are fairly simple. In the sequel, we will concentrate mainly on such models.

1.14 Filter Model of First Order Autoregressive Process

To gain some understanding of filter models of the above type, we consider a very simple example of a first-order recursive filter $B(z)$ driven by a purely random sequence of variance σ_ϵ^2:

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

This serves also as a simple model for generating a first order Markov signal. The signal y_n is generated by the difference equation of the filter:

$$y_n = a y_{n-1} + \epsilon_n \tag{1.14.1}$$

Let the probability of the nth sample ϵ_n be $f(\epsilon_n)$. We would like to show that

$$p(y_n | y_{n-1}, y_{n-2}, \ldots, y_1, y_0) = p(y_n | y_{n-1}) = f(\epsilon_n) = f(y_n - a y_{n-1})$$

which not only shows the Markov property of y_n, but also how to compute the related conditional density. Perhaps the best way to see this is to start at $n = 0$:

$$y_0 = \epsilon_0 \qquad \text{(assuming zero initial conditions)}$$

$$y_1 = a y_0 + \epsilon_1$$

$$y_2 = a y_1 + \epsilon_2\,, \quad \text{etc.}$$

To compute $p(y_2|y_1,y_0)$, suppose that y_1 and y_0 are both given. Since y_1 is given, the third equation above shows that the randomness left in y_2 arises from ϵ_2 only. Thus, $p(y_2|y_1)=f(\epsilon_2)$. From the first two equations it follows that specifying y_0 and y_1 is equivalent to specifying ϵ_0 and ϵ_1. Therefore, $p(y_2|y_1,y_0)=f(\epsilon_2|\epsilon_1,\epsilon_0)=f(\epsilon_2)$, the last equation following from the purely random nature of the sequence ϵ_n. We have shown that

$$p(y_2|y_1,y_0)=p(y_2|y_1)=f(\epsilon_2)=f(y_2-ay_1)$$

Using the results of Sec. 1.9, we also note

$$\begin{aligned} p(y_2,y_1,y_0) &= p(y_2|y_1)p(y_1|y_0)p(y_0) \\ &= f(\epsilon_2)f(\epsilon_1)f(\epsilon_0) \\ &= f(y_2-ay_1)f(y_1-ay_0)f(y_0) \end{aligned}$$

The solution of the difference equation (1.14.1) is obtained by convolving the impulse response of the filter $B(z)$

$$b_n = a^n u(n)\,, \qquad u(n)=\text{unit step}$$

with the input sequence ϵ_n as follows:

$$y_n = \sum_{i=0}^{n} b_i \epsilon_{n-i} = \sum_{i=0}^{n} a^i \epsilon_{n-i} \tag{1.14.2}$$

for $n=0,1,2,\ldots$. This is the innovations representation of y_n given by Eqs. (1.6.15), (1.6.16), and (1.13.1). In matrix form it reads:

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ a & 1 & 0 & 0 \\ a^2 & a & 1 & 0 \\ a^3 & a^2 & a & 1 \end{bmatrix} \begin{bmatrix} \epsilon_0 \\ \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{bmatrix} \tag{1.14.3}$$

The inverse equation, $\boldsymbol{\epsilon} = B^{-1}\mathbf{y} = A\mathbf{y}$, is obtained by writing Eq. (1.14.1) as $\epsilon_n = y_n - ay_{n-1}$. In matrix form, this reads

$$\begin{bmatrix} \epsilon_0 \\ \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -a & 1 & 0 & 0 \\ 0 & -a & 1 & 0 \\ 0 & 0 & -a & 1 \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix} \tag{1.14.4}$$

According to the discussion of Example 1.7.1, the partial correlation coefficients can be read off from the *first column* of this matrix. We conclude, therefore, that all partial correlation coefficients of order greater than two are zero. This property is in accordance with our intuition about first order Markov processes; due to the recursive nature of Eq. (1.14.1) a given sample, say y_n, will have an indirect influence on all future samples. However, the only direct influence is to the next sample.

Higher order autoregressive random signals can be generated by sending white noise through higher order filters. For example, the second-order difference equation

$$y_n = a_1 y_{n-1} + a_2 y_{n-2} + \epsilon_n \tag{1.14.5}$$

will generate a second-order Markov signal. In this case, the difference equation directly couples two successive samples, but not more than two. Therefore, all the partial correlations of order greater than three will be zero. This may be seen also by writing Eq. (1.14.5) in matrix form and inspecting the first column of the matrix A:

$$\begin{bmatrix} \epsilon_0 \\ \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -a_1 & 1 & 0 & 0 & 0 \\ -a_2 & -a_1 & 1 & 0 & 0 \\ 0 & -a_2 & -a_1 & 1 & 0 \\ 0 & 0 & -a_2 & -a_1 & 1 \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$$

1.15 Stability and Stationarity

In this section we discuss the importance of stability of the signal generator filter $B(z)$. We demonstrate that the generated signal y_n will be stationary only if the generating filter is stable. And in this case, the sequence y_n will become stationary only after the transient effects introduced by the filter have died out.

To demonstrate these ideas, consider the lag-0 autocorrelation of our first order Markov signal

$$\begin{aligned} R_{yy}(n,n) &= E[y_n^2] = E\big[(ay_{n-1}+\epsilon_n)^2\big] \\ &= a^2E[y_{n-1}^2] + 2aE[y_{n-1}\epsilon_n] + E[\epsilon_n^2] = a^2R_{yy}(n-1,n-1)+\sigma_\epsilon^2 \end{aligned} \tag{1.15.1}$$

where we set $\sigma_\epsilon^2 = E[\epsilon_n^2]$ and $E[y_{n-1}\epsilon_n]=0$, which follows by using Eq. (1.14.2) to get

$$y_{n-1} = \epsilon_{n-1} + a\epsilon_{n-2} + \cdots + a^{n-1}\epsilon_0$$

and noting that ϵ_n is uncorrelated with all these terms, due to its white-noise nature. The above difference equation for $R_{yy}(n,n)$ can now be solved to get

$$R_{yy}(n,n) = E[y_n^2] = \frac{\sigma_\epsilon^2}{1-a^2} + \sigma_\epsilon^2\left(1-\frac{1}{1-a^2}\right)a^{2n} \tag{1.15.2}$$

where the initial condition was taken to be $E[y_0^2]=E[\epsilon_0^2]=\sigma_\epsilon^2$. If the filter is stable and causal, that is, $|a|<1$, then the second term in (1.15.2) tends to zero exponentially, and $R_{yy}(n,n)$ eventually loses its dependence on the absolute time n. For large n, it tends to the steady-state value

$$R_{yy}(0) = E[y_n^2] = \sigma_y^2 = \frac{\sigma_\epsilon^2}{1-a^2} \tag{1.15.3}$$

The same result is obtained, of course, by assuming stationarity from the start. The difference equation (1.15.1) can be written as

$$E[y_n^2] = a^2E[y_{n-1}^2] + \sigma_\epsilon^2$$

If y_n is assumed to be already stationary, then $E[y_n^2]=E[y_{n-1}^2]$. This implies the same steady-state solution as Eq. (1.15.3).

If the filter is unstable, that is, $|a| > 1$, then the second term of Eq. (1.15.2) diverges exponentially. The marginal case $a = 1$ is also unacceptable, but is of historical interest being the famous Wiener process, or random walk. In this case, the signal model is

$$y_n = y_{n-1} + \epsilon_n$$

and the difference equation for the variance becomes

$$R_{yy}(n,n) = R_{yy}(n-1,n-1) + \sigma_\epsilon^2$$

with solution

$$R_{yy}(n,n) = E[y_n^2] = (n+1)\,\sigma_\epsilon^2$$

In summary, for true stationarity to set in, the signal generator filter $B(z)$ must be *strictly* stable (all its poles must be strictly inside the unit circle).

1.16 Parameter Estimation

One of the most important practical questions is how to extract the model parameters, such as the above filter parameter a, from the actual data values. As an introduction to the analysis methods used to answer this question, let us suppose that the white noise input sequence ϵ_n is gaussian

$$f(\epsilon_n) = \frac{1}{\sqrt{2\pi}\sigma_\epsilon} \exp\left(-\frac{\epsilon_n^2}{2\sigma_\epsilon^2}\right)$$

and assume that a block of N measured values of the signal y_n is available

$$y_0, y_1, y_2, \ldots, y_{N-1}$$

Can we extract the filter parameter a from this block of data? Can we also extract the variance σ_ϵ^2 of the driving white noise ϵ_n? If so, then instead of saving the N measured values $\{y_0, y_1, y_2, \ldots, y_{N-1}\}$, we can save the extracted filter parameter a and the variance σ_ϵ^2. Whenever we want to synthesize our original sequence y_n, we will simply generate a white-noise input sequence ϵ_n of variance σ_ϵ^2, using a pseudorandom number generator routing, and then drive with it the signal model whose parameter a was previously extracted from the original data. Somehow, all the significant information contained in the original samples, has now been packed or compressed into the two numbers a and σ_ϵ^2.

One possible criterion for extracting the filter parameter a is the maximum likelihood (ML) criterion: The parameter a is selected so as to *maximize* the joint density

$$\begin{aligned} p(y_0, y_1, \ldots, y_{N-1}) &= f(\epsilon_0) f(\epsilon_1) \cdots f(\epsilon_{N-1}) \\ &= \frac{1}{(\sqrt{2\pi}\sigma_\epsilon)^N} \exp\left[-\frac{1}{2\sigma_\epsilon^2} \sum_{n=1}^{N-1} (y_n - a y_{n-1})^2\right] \exp\left[-y_0^2/2\sigma_\epsilon^2\right] \end{aligned}$$

that is, the parameter a is selected so as to render the actual measured values $\{y_0, y_1, y_2, \ldots, y_{N-1}\}$ most likely. The criterion is equivalent to minimizing the exponent with respect to a:

$$\mathcal{E}(a) = \sum_{n=1}^{N-1} (y_n - a y_{n-1})^2 + y_0^2 = \sum_{n=0}^{N-1} e_n^2 = \min \tag{1.16.1}$$

where we set $e_n = y_n - a y_{n-1}$, and $e_0 = y_0$. The minimization of Eq. (1.16.1) gives

$$\frac{\partial \mathcal{E}(a)}{\partial a} = -2 \sum_{n=1}^{N-1} (y_n - a y_{n-1}) y_{n-1} = 0\,, \quad \text{or,}$$

$$a = \frac{\sum_{n=1}^{N-1} y_n y_{n-1}}{\sum_{n=1}^{N-1} y_{n-1}^2} = \frac{y_0 y_1 + y_1 y_2 + \cdots + y_{N-2} y_{N-1}}{y_0^2 + y_1^2 + \cdots + y_{N-2}^2} \tag{1.16.2}$$

There is a potential problem with the above ML criterion for extracting the filter parameter a, namely, the parameter may turn out to have magnitude greater than one, which will correspond to an unstable filter generating the sequence y_n. This is easily seen from Eq. (1.16.2); whereas the numerator has dependence on the last sample y_{N-1}, the denominator does not. Therefore it is possible, for sufficiently large values of y_{N-1}, for the parameter a to be greater than one. There are other criteria for extracting the Markov model parameters that guarantee the stability of the resulting synthesis filters, such as the so-called autocorrelation method, or Burg's method. These will be discussed later on.

An alternative parameter estimation method is the *autocorrelation* or *Yule-Walker* method of extracting the model parameters from a block of data. We begin by expressing the model parameters in terms of output statistical quantities and then replace ensemble averages by time averages. Assuming stationarity has set in, we find

$$R_{yy}(1) = E[y_n y_{n-1}] = E[(a y_{n-1} + \epsilon_n) y_{n-1}] = a E[y_{n-1}^2] + E[\epsilon_n y_{n-1}] = a R_{yy}(0)$$

from which

$$a = \frac{R_{yy}(1)}{R_{yy}(0)}$$

The input parameter σ_ϵ^2 can be expressed as

$$\sigma_\epsilon^2 = (1 - a^2)\,\sigma_y^2 = (1 - a^2) R_{yy}(0)$$

These two equations may be written in matrix form as

$$\begin{bmatrix} R_{yy}(0) & R_{yy}(1) \\ R_{yy}(1) & R_{yy}(0) \end{bmatrix} \begin{bmatrix} 1 \\ -a \end{bmatrix} = \begin{bmatrix} \sigma_\epsilon^2 \\ 0 \end{bmatrix}$$

These are called the *normal equations* of linear prediction. Their generalization will be considered later on. These results are important because they allow the extraction of the signal model parameters directly in terms of *output* quantities, that is, from experimentally accessible quantities.

We may obtain estimates of the model parameters by replacing the theoretical autocorrelations by the corresponding *sample autocorrelations*, defined by Eq. (1.11.1):

$$\hat{a} = \frac{\hat{R}_{yy}(1)}{\hat{R}_{yy}(0)} = \frac{\frac{1}{N}\sum_{n=0}^{N-1-1} y_{n+1}y_n}{\frac{1}{N}\sum_{n=0}^{N-1} y_n y_n} = \frac{y_0y_1 + y_1y_2 + \cdots + y_{N-2}y_{N-1}}{y_0^2 + y_1^2 + \cdots + y_{N-2}^2 + y_{N-1}^2}$$

$$\hat{\sigma}_\epsilon^2 = (1 - \hat{a}^2)\hat{R}_{yy}(0)$$

It is easily checked that the parameter $\hat{a}$, defined as above, is always of magnitude less than one; thus, the stability of the synthesis filter is guaranteed. Note the difference with the ML expression. The numerators are the same, but the denominators differ by an extra term. It is also interesting to note that the above expressions may be obtained by a *minimization criterion*; known as the autocorrelation method, or the Yule-Walker method:

$$\mathcal{E}(a) = \sum_{n=0}^{N} e_n^2 = \sum_{n=0}^{N} (y_n - ay_{n-1})^2 = \min \tag{1.16.3}$$

This differs from the ML criterion (1.16.1) only in the range of summation for n. Whereas in the ML criterion the summation index n does not run off the ends of the data block, it does so in the Yule-Walker case. We may think of the block of data as having been extended to both directions by padding it with zeros

$$0, \ldots, 0, y_0, y_1, \ldots, y_{N-1}, 0, 0, \ldots, 0$$

The difference between this and the ML criterion arises from the last term in the sum

$$\mathcal{E}(a) = \sum_{n=0}^{N} e_n^2 = \sum_{n=1}^{N-1} e_n^2 + e_N^2 = \sum_{n=1}^{N-1} (y_n - ay_{n-1})^2 + (0 - ay_{N-1})^2$$

The Yule-Walker analysis algorithm for this first order example is summarized in Fig. 1.16.1.

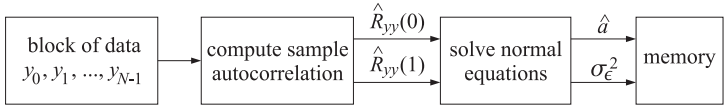


Fig. 1.16.1 Yule-Walker analysis method.

How good are $\hat{a}$ and $\hat{\sigma}_\epsilon^2$ as estimates of the model parameters a and σ_ϵ^2? It can be shown that they, and the maximum likelihood estimates of the previous section, are asymptotically unbiased and consistent. The corresponding variances are given for large N by [4-6]

$$E[(\Delta a)^2] = \frac{1 - a^2}{N}, \quad E[(\Delta\sigma_\epsilon^2)^2] = \frac{2\sigma_\epsilon^4}{N} \tag{1.16.4}$$

where $\Delta a = \hat{a} - a$ and $\Delta\sigma_\epsilon^2 = \hat{\sigma}_\epsilon^2 - \sigma_\epsilon^2$. Such asymptotic properties are discussed in greater detail in Chap. 14. Here, we present some simulation examples showing that (1.16.4) are adequate even for fairly small N.

Example 1.16.1: The following $N = 30$ signal samples of y_n have been generated by passing zero-mean white noise through the difference equation $y_n = ay_{n-1} + \epsilon_n$, with $a = 0.8$ and $\sigma_\epsilon^2 = 1$:

$$y_n = \{2.583,\ 2.617,\ 2.289,\ 2.783,\ 2.862,\ 3.345,\ 2.704,\ 1.527,\ 2.096,\ 2.050,\ 2.314,$$
$$0.438,\ 1.276,\ 0.524,\ -0.449,\ -1.736,\ -2.599,\ -1.633,\ 1.096,\ 0.348,\ 0.745,$$
$$0.797,\ 1.123,\ 1.031,\ -0.219,\ 0.593,\ 2.855,\ 0.890,\ 0.970,\ 0.924\}$$

Using the Yule-Walker method, we obtain the following estimates of the model parameters

$$\hat{a} = 0.806, \quad \sigma_\epsilon^2 = 1.17$$

Both estimates are consistent with the theoretically expected fluctuations about their means given by Eq. (1.16.4), falling within the one-standard deviation intervals $a \pm \delta a$ and $\sigma_\epsilon^2 \pm \delta\sigma_\epsilon^2$, where δa and $\delta\sigma_\epsilon^2$ are the square roots of Eq. (1.16.4). For $N = 30$, the numerical values of these intervals are: $0.690 \le \hat{a} \le 0.910$ and $0.742 \le \sigma_\epsilon^2 \le 1.258$. Given the theoretical and estimated model parameters, we can obtain the theoretical and estimated power spectral densities of y_n by

$$S_{\text{TH}}(\omega) = \frac{\sigma_\epsilon^2}{|1 - ae^{-j\omega}|^2}, \quad S_{\text{YW}}(\omega) = \frac{\hat{\sigma}_\epsilon^2}{|1 - \hat{a}e^{-j\omega}|^2}$$

The periodogram spectrum based on the given length-N data block is

$$S_{\text{PER}}(\omega) = \frac{1}{N}\left|\sum_{n=0}^{N-1} y_n e^{-jn\omega}\right|^2$$

The three spectra are plotted in Fig. 1.16.2, in units of decibels; that is, $10\log_{10} S$, over the right half of the Nyquist interval $0 \le \omega \le \pi$. Note the excellent agreement of the Yule-Walker spectrum with the theoretical spectrum and the several sidelobes of the periodogram spectrum caused by the windowing of y_n.

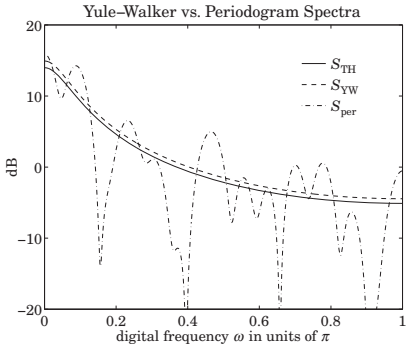


Fig. 1.16.2 Comparison of Yule-Walker and periodogram spectrum estimates.

Example 1.16.2: The purpose of this example is to demonstrate the reasonableness of the asymptotic variances, Eq. (1.16.4). For the first-order model defined in the previous example, we generated 100 different realizations of the length-30 signal block y_n. From each realization, we extracted the Yule-Walker estimates of the model parameters $\hat{a}$ and $\hat{\sigma}_\epsilon^2$. They are shown in Figs. 1.16.3 versus realization index, together with the corresponding asymptotic one-standard deviation intervals that were computed in the previous example.

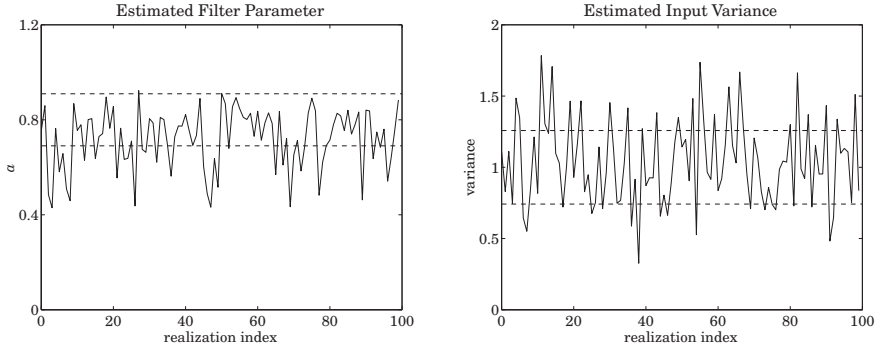


Fig. 1.16.3 Parameters a, σ_ϵ^2 estimated from 100 realizations of the length-30 data block y_n.

1.17 Linear Prediction and Signal Modeling

Linear prediction ideas are introduced in the context of our simple example by noting that the least-squares minimization criteria (1.16.1) and (1.16.3)

$$\mathcal{E}(a) = \sum_n e_n^2 = \text{minimum} \tag{1.17.1}$$

essentially force each e_n to be small. Thus, if we reinterpret

$$\hat{y}_n = a y_{n-1}$$

as the linear prediction of the sample y_n made on the basis of just the previous sample y_{n-1}, then $e_n = y_n - a y_{n-1} = y_n - \hat{y}_n$ may be thought of as the prediction error. The minimization criterion (1.17.1) essentially minimizes the prediction error in an average least-squares sense, thus attempting to make the best prediction possible.

As we mentioned in Sec. 1.13, the solution of the linear prediction problem provides the corresponding random signal generator model for y_n, which can be used, in turn, in a number of ways as outlined in Sec. 1.13. This is the main reason for our interest in linear prediction.

A more intuitive way to understand the connection between linear prediction and signal models is as follows: Suppose we have a predictor $\hat{y}_n$ of y_n which is not necessarily the best predictor. The predictor $\hat{y}_n$ is given as a linear combination of the past values $\{y_{n-1}, y_{n-2}, \dots\}$:

$$\hat{y}_n = -[a_1 y_{n-1} + a_2 y_{n-2} + \cdots] \tag{1.17.2}$$

The corresponding prediction error will be

$$e_n = y_n - \hat{y}_n = y_n + a_1 y_{n-1} + a_2 y_{n-2} + \cdots \tag{1.17.3}$$

and it may be considered as the output of the prediction-error filter $A(z)$ (which is assumed to be stable and causal):

$$y_n \longrightarrow \boxed{A(z)} \longrightarrow e_n \qquad A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots$$

Suppose further that $A(z)$ has a stable and causal inverse filter

$$e_n \longrightarrow \boxed{B(z)} \longrightarrow y_n \qquad B(z) = \frac{1}{A(z)} = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots}$$

so that y_n may be expressed *causally* in terms of e_n, that is,

$$y_n = e_n + b_1 e_{n-1} + b_2 e_{n-2} + \cdots \tag{1.17.4}$$

Then, Eqs. (1.17.3) and (1.17.4) imply that the linear spaces generated by the random variables

$$\{y_{n-1}, y_{n-2}, \dots\} \quad \text{and} \quad \{e_{n-1}, e_{n-2}, \dots\}$$

are the same space. One can pass from one set to the other by a causal and causally invertible linear filtering operation.

Now, if the prediction $\hat{y}_n$ of y_n is the best possible prediction, then what remains after the prediction is made—namely, the error signal e_n—should be entirely *unpredictable* on the basis of the past values $\{y_{n-1}, y_{n-2}, \dots\}$. That is, e_n must be uncorrelated with all of these. But this implies that e_n must be uncorrelated with all $\{e_{n-1}, e_{n-2}, \dots\}$, and therefore e_n must be a white-noise sequence. It follows that $A(z)$ and $B(z)$ are the analysis and synthesis filters for the sequence y_n.

The least-squares minimization criteria of the type (1.17.1) that are based on time averages, provide a practical way to solve the linear prediction problem and hence also the modeling problem. Their generalization to higher order predictors will be discussed in Chap. 12.

1.18 Cramér–Rao Bound and Maximum Likelihood

The Cramér-Rao inequality [2–5,27] provides a lower bound for the variance of unbiased estimators of parameters. Thus, the best any parameter estimator can do is to meet its Cramér-Rao bound. Such estimators are called *efficient*. Parameter estimators based on the principle of *maximum likelihood*, such as the one presented in Sec. 1.16, have several nice properties, namely, as the number of observations becomes large, they are asymptotically unbiased, consistent, efficient, and are asymptotically normally distributed about the theoretical value of the parameter with covariance given by the Cramér-Rao bound.

In this section, we present a derivation of the Cramér-Rao inequality using correlation canceling methods and discuss its connection to maximum likelihood. Consider

N observations $Y = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N\}$, where each observation is assumed to be an M-dimensional random vector. Based on these observations, we would like to estimate a number of (deterministic) parameters, assembled into a parameter vector $\boldsymbol{\lambda}$. We will write $p(Y, \boldsymbol{\lambda})$ to indicate the dependence of the joint probability density on $\boldsymbol{\lambda}$. As a concrete example, consider the case of N independent scalar observations drawn from a normal distribution with mean m and variance σ^2. The joint density is

$$p(Y, \boldsymbol{\lambda}) = (2\pi\sigma^2)^{-N/2} \exp\left[-\frac{1}{2\sigma^2} \sum_{n=1}^{N} (y_n - m)^2 \right] \tag{1.18.1}$$

For the parameter vector we may choose $\boldsymbol{\lambda} = [m, \sigma^2]^T$, if we want to estimate both the mean and variance.

The dependence of $p(Y, \boldsymbol{\lambda})$ on $\boldsymbol{\lambda}$ may be expressed in terms of the gradient with respect to $\boldsymbol{\lambda}$ of the log-likelihood function

$$\boldsymbol{\psi}(Y, \boldsymbol{\lambda}) \equiv \frac{\partial}{\partial \boldsymbol{\lambda}} \ln p(Y, \boldsymbol{\lambda}) = \frac{1}{p} \frac{\partial p}{\partial \boldsymbol{\lambda}} \tag{1.18.2}$$

Expectation values with respect to the joint density will, in general, depend on the parameter $\boldsymbol{\lambda}$. We have the following result for the expectation value of an arbitrary function $F(Y, \boldsymbol{\lambda})$:

$$\frac{\partial}{\partial \boldsymbol{\lambda}} E[F] = E\left[\frac{\partial F}{\partial \boldsymbol{\lambda}} \right] + E[F\boldsymbol{\psi}] \tag{1.18.3}$$

Writing $dY = d^M\mathbf{y}_1 d^M\mathbf{y}_2 \cdots d^M\mathbf{y}_N$ for the volume element over the space of observations, the proof of Eq. (1.18.3) follows from

$$\frac{\partial}{\partial \boldsymbol{\lambda}} \int pF\, dY = \int \frac{\partial}{\partial \boldsymbol{\lambda}} (pF)\, dY = \int p \frac{\partial F}{\partial \boldsymbol{\lambda}}\, dY + \int pF \frac{\partial \ln p}{\partial \boldsymbol{\lambda}}\, dY$$

Applying this property to $F = 1$, we find $E[\boldsymbol{\psi}] = 0$. Applying it to $\boldsymbol{\psi}$ itself, that is, $F = \boldsymbol{\psi}$, we find

$$J \equiv E[\boldsymbol{\psi}\boldsymbol{\psi}^T] = E[\Psi] \tag{1.18.4}$$

where

$$\Psi \equiv -\frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{\lambda}}$$

Eq. (1.18.4) is known as the *Fisher information matrix* based on Y. Component-wise, we have

$$J_{ij} = E[\psi_i \psi_j] = E[\Psi_{ij}]$$

where

$$\psi_i = \frac{\partial \ln p}{\partial \lambda_i}, \quad \Psi_{ij} = -\frac{\partial \psi_i}{\partial \lambda_j} = -\frac{\partial^2 \ln p}{\partial \lambda_i \partial \lambda_j}$$

Next, we derive the Cramér-Rao bound. Let $\hat{\boldsymbol{\lambda}}(Y)$ be any estimator of $\boldsymbol{\lambda}$ based on Y. Because $\hat{\boldsymbol{\lambda}}(Y)$ and $\boldsymbol{\psi}(Y, \boldsymbol{\lambda})$ both depend on Y, they will be correlated with each other. Using the correlation canceling methods of Sec. 1.4, we can remove these correlations by writing

$$\mathbf{e} = \hat{\boldsymbol{\lambda}} - E[\hat{\boldsymbol{\lambda}}\boldsymbol{\psi}^T] E[\boldsymbol{\psi}\boldsymbol{\psi}^T]^{-1} \boldsymbol{\psi}$$

Then, $\mathbf{e}$ will not be correlated with $\boldsymbol{\psi}$. Because $\boldsymbol{\psi}$ has zero mean, it follows that $E[\hat{\boldsymbol{\lambda}}] = E[\mathbf{e}]$. Working with the deviations about the corresponding means, namely, $\Delta\boldsymbol{\lambda} = \hat{\boldsymbol{\lambda}} - E[\hat{\boldsymbol{\lambda}}]$ and $\Delta\mathbf{e} = \mathbf{e} - E[\mathbf{e}]$, we have

$$\Delta\mathbf{e} = \Delta\boldsymbol{\lambda} - MJ^{-1}\boldsymbol{\psi} \tag{1.18.5}$$

where we denoted $M = E[\hat{\boldsymbol{\lambda}}\boldsymbol{\psi}^T]$. Following Eq. (1.4.4), we obtain for the covariance of $\Delta\mathbf{e}$

$$E[\Delta\mathbf{e}\Delta\mathbf{e}^T] = E[\Delta\boldsymbol{\lambda}\Delta\boldsymbol{\lambda}^T] - MJ^{-1}M^T \tag{1.18.6}$$

Thus, the difference of terms in the right-hand side is a positive semi-definite matrix. This may be expressed symbolically as $E[\Delta\mathbf{e}\Delta\mathbf{e}^T] \geq 0$, or, $E[\Delta\boldsymbol{\lambda}\Delta\boldsymbol{\lambda}^T] \geq MJ^{-1}M^T$. The quantity M depends on the *bias* of the estimator. For an *unbiased* estimator, M is the identity matrix, $M = I$, and we obtain the Cramér-Rao inequality

$$\operatorname{cov}(\hat{\boldsymbol{\lambda}}) = E[\Delta\boldsymbol{\lambda}\Delta\boldsymbol{\lambda}^T] \geq J^{-1} \qquad \text{(Cramér-Rao)} \tag{1.18.7}$$

The dependence of M on the bias can be seen as follows. Because $\hat{\boldsymbol{\lambda}}(Y)$ has no explicit dependence on $\boldsymbol{\lambda}$, it follows from property (1.18.3) that

$$M = E[\hat{\boldsymbol{\lambda}}\boldsymbol{\psi}^T] = \frac{\partial}{\partial \boldsymbol{\lambda}} E[\hat{\boldsymbol{\lambda}}]$$

Define the bias of the estimator as the deviation of the mean from the true value of the parameter, that is, $E[\hat{\boldsymbol{\lambda}}] = \boldsymbol{\lambda} + \mathbf{b}(\boldsymbol{\lambda})$, where $\mathbf{b}(\boldsymbol{\lambda})$ is the bias

$$M = I + \frac{\partial \mathbf{b}}{\partial \boldsymbol{\lambda}} \equiv I + B$$

For an unbiased estimator, $B = 0$ and $M = I$. It follows from Eq. (1.18.6) that for the Cramér-Rao inequality to be satisfied as an equality, it is necessary that $\Delta\mathbf{e} = 0$ in Eq. (1.18.5), i.e., $\Delta\boldsymbol{\lambda} = MJ^{-1}\boldsymbol{\psi}$ and in the unbiased case, we obtain the condition $\boldsymbol{\psi} = J\Delta\boldsymbol{\lambda}$:

$$\frac{\partial}{\partial \boldsymbol{\lambda}} \ln p(Y, \boldsymbol{\lambda}) = J\Delta\boldsymbol{\lambda} = J[\hat{\boldsymbol{\lambda}}(Y) - \boldsymbol{\lambda}] \tag{1.18.8}$$

Estimators that satisfy this condition and thus, meet their Cramér-Rao bound, are called efficient.

Example 1.18.1: The log-likelihood function of Eq. (1.18.1) is

$$\ln p = -\frac{N}{2} \ln(2\pi) - \frac{N}{2} \ln\sigma^2 - \frac{1}{2\sigma^2} \sum_{n=1}^{N} (y_n - m)^2$$

The gradients with respect to the parameters m and σ^2 are

$$\begin{aligned} \frac{\partial \ln p}{\partial m} &= \frac{1}{\sigma^2} \sum_{n=1}^{N} (y_n - m) \\ \frac{\partial \ln p}{\partial \sigma^2} &= -\frac{N}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{n=1}^{N} (y_n - m)^2 \end{aligned} \tag{1.18.9}$$

The second derivatives are the matrix elements of the matrix Ψ:

$$\Psi_{mm} = -\frac{\partial^2 \ln p}{\partial m \partial m} = \frac{N}{\sigma^2}$$

$$\Psi_{m\sigma^2} = -\frac{\partial^2 \ln p}{\partial m \partial \sigma^2} = \frac{1}{\sigma^4}\sum_{n=1}^{N}(y_n - m)$$

$$\Psi_{\sigma^2\sigma^2} = -\frac{\partial^2 \ln p}{\partial \sigma^2 \partial \sigma^2} = -\frac{N}{2\sigma^4} + \frac{1}{\sigma^6}\sum_{n=1}^{N}(y_n - m)^2$$

Taking expectation values, we find the matrix elements of J

$$J_{mm} = \frac{N}{\sigma^2}\,, \quad J_{m\sigma^2} = 0\,, \quad J_{\sigma^2\sigma^2} = \frac{N}{2\sigma^4}$$

Therefore, the Cramér-Rao bound of any unbiased estimator of m and σ^2 will be

$$\begin{bmatrix} E[\Delta m \Delta m] & E[\Delta m \Delta \sigma^2] \\ E[\Delta\sigma^2 \Delta m] & E[\Delta\sigma^2 \Delta\sigma^2] \end{bmatrix} \geq \begin{bmatrix} \sigma^2/N & 0 \\ 0 & 2\sigma^4/N \end{bmatrix}$$

Example 1.18.2: We note that the sample mean $\hat{m}$ defined by Eq. (1.2.1) has variance equal to its Cramér-Rao bound, and therefore, it is an efficient estimator. It also satisfies condition (1.18.8). Writing $\sum_{n=1}^{N} y_n = N\hat{m}$, we obtain from Eq. (1.18.9)

$$\frac{\partial \ln p}{\partial m} = \frac{1}{\sigma^2}\sum_{n=1}^{N}(y_n - m) = \frac{1}{\sigma^2}\left[\sum_{n=1}^{N} y_n - Nm\right] = \frac{1}{\sigma^2}(N\hat{m} - Nm) = J_{mm}(\hat{m} - m)$$

We also note that the sample variance s^2 having variance $2\sigma^4/(N-1)$ meets its Cramér-Rao bound only asymptotically. The biased definition of the sample variance, Eq. (1.2.3), has variance given by Eq. (1.2.4). It is easily verified that it is *smaller* than its Cramér-Rao bound (1.18.7). But this is no contradiction because Eq. (1.18.7) is valid only for unbiased estimators. For a biased estimator, the lower bound $MJ^{-1}M^T$ must be used. Equation (1.2.4) does satisfy this bound. □

Next, we discuss the principle of maximum likelihood. The *maximum likelihood estimator* of a parameter $\boldsymbol{\lambda}$ is the value $\hat{\boldsymbol{\lambda}}$ that maximizes the joint density $p(Y, \boldsymbol{\lambda})$; i.e.,

$$p(Y, \boldsymbol{\lambda})\,|_{\boldsymbol{\lambda}=\hat{\boldsymbol{\lambda}}} = \text{maximum} \tag{1.18.10}$$

Equivalently,

$$\boldsymbol{\psi}(\hat{\boldsymbol{\lambda}}) = \frac{\partial}{\partial \boldsymbol{\lambda}} \ln p(Y, \boldsymbol{\lambda})\,\Big|_{\boldsymbol{\lambda}=\hat{\boldsymbol{\lambda}}} = 0 \tag{1.18.11}$$

In general, this equation is difficult to solve. However, the asymptotic properties of the solution for large N are simple enough to obtain. Assuming that $\hat{\boldsymbol{\lambda}}$ is near the true value of the parameter $\boldsymbol{\lambda}$ we may expand the gradient $\boldsymbol{\psi}$ about the true value:

$$\boldsymbol{\psi}(\hat{\boldsymbol{\lambda}}) \simeq \boldsymbol{\psi} + \frac{\partial \boldsymbol{\psi}(\boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}}(\hat{\boldsymbol{\lambda}} - \boldsymbol{\lambda}) = \boldsymbol{\psi} - \Psi(\hat{\boldsymbol{\lambda}} - \boldsymbol{\lambda})$$

where we used the matrix Ψ defined in Eq. (1.18.4). For the maximum likelihood solution, the left-hand side is zero. Thus, solving for $\Delta\boldsymbol{\lambda} = \hat{\boldsymbol{\lambda}} - \boldsymbol{\lambda}$, we obtain

$$\Delta\boldsymbol{\lambda} = \Psi^{-1}\boldsymbol{\psi} \tag{1.18.12}$$

Assuming that the N observations are independent of each other, the joint density $p(Y, \boldsymbol{\lambda})$ factors into the marginal densities $\prod_{n=1}^{N} p(\mathbf{y}_n, \boldsymbol{\lambda})$. Therefore, the gradient $\boldsymbol{\psi}$ will be a sum of gradients

$$\boldsymbol{\psi} = \frac{\partial}{\partial \boldsymbol{\lambda}} \ln p = \sum_{n=1}^{N} \frac{\partial}{\partial \boldsymbol{\lambda}} \ln p(\mathbf{y}_n, \boldsymbol{\lambda}) = \sum_{n=1}^{N} \boldsymbol{\psi}_n$$

Similarly,

$$\Psi = -\frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{\lambda}} - \sum_{n=1}^{N} \frac{\partial \boldsymbol{\psi}_n}{\partial \boldsymbol{\lambda}} = \sum_{N=1}^{N} \Psi_n$$

Individual terms in these sums are mutually independent. Thus, from the law of large numbers, we can replace Ψ by its mean $\Psi \simeq E[\Psi] = J$, and Eq. (1.18.12) becomes

$$\Delta\boldsymbol{\lambda} = J^{-1}\boldsymbol{\psi} \tag{1.18.13}$$

This asymptotic equation contains essentially all the nice properties of the maximum likelihood estimator. First, from $E[\Psi] = 0$, it follows that $E[\Delta\boldsymbol{\lambda}] = 0$, or that $\hat{\boldsymbol{\lambda}}$ is *asymptotically unbiased*. Second, its asymptotic covariance agrees with the Cramér-Rao bound

$$E[\Delta\boldsymbol{\lambda}\Delta\boldsymbol{\lambda}^T] = J^{-1}E[\boldsymbol{\psi}\boldsymbol{\psi}^T]J^{-1} = J^{-1}JJ^{-1} = J^{-1}$$

Thus, $\hat{\boldsymbol{\lambda}}$ is *asymptotically efficient*. The same conclusion can be reached by noting that Eq. (1.18.13) is the same as condition (1.18.8). Third, $\hat{\boldsymbol{\lambda}}$ is *asymptotically consistent*, in the sense that its covariance tends to zero for large N. This follows from the fact that the information matrix for N independent observations is equal to N times the information matrix for one observation:

$$J = E[\Psi] = \sum_{n=1}^{N} E[\Psi_n] = NE[\Psi_1] = NJ_1$$

Therefore, $J^{-1} = J_1^{-1}/N$ tends to zero for large N. Fourth, because $\boldsymbol{\psi}$ is the sum of N independent terms, it follows from the vector version of the central limit theorem that $\boldsymbol{\psi}$ will be *asymptotically normally distributed*. Thus, so will be $\hat{\boldsymbol{\lambda}}$, with mean $\boldsymbol{\lambda}$ and covariance J^{-1}.

Example 1.18.3: Setting the gradients (1.18.9) to zero, we obtain the maximum likelihood estimates of the parameters m and σ^2. It is easily verified that they coincide with the sample mean and sample variance defined by Eqs. (1.2.1) and (1.2.3). □

Example 1.18.4: In many applications, the mean is known to be zero and only the variance needs to be estimated. For example, setting $m = 0$ in Eq. (1.18.1) we obtain the log-likelihood

$$\ln p = -\frac{N}{2}\ln(2\pi) - \frac{N}{2}\ln\sigma^2 - \frac{1}{2\sigma^2}\sum_{n=1}^{N} y_n^2$$

The maximum likelihood estimate of σ^2 is obtained from

$$\frac{\partial \ln p}{\partial \sigma^2} = -\frac{N}{2\sigma^2} + \frac{1}{2\sigma^4}\sum_{n=1}^{N} y_n^2 = 0$$

with solution

$$\hat{\sigma}^2 = \frac{1}{N}\sum_{n=1}^{N} y_n^2$$

It is easily verified that this is an *unbiased* estimate. It is the scalar version of Eq. (1.6.21). Using $E[y_n^2 y_m^2] = \sigma^4 + 2\delta_{nm}\sigma^4$, which is valid for independent zero-mean gaussian y_ns, we find for the variance of $\hat{\sigma}^2$

$$E[\Delta\sigma^2 \Delta\sigma^2] = \frac{2\sigma^4}{N}, \quad \text{where} \quad \Delta\sigma^2 = \hat{\sigma}^2 - \sigma^2 \tag{1.18.14}$$

This agrees with the corresponding Cramér-Rao bound. Thus, $\hat{\sigma}^2$ is efficient. Equation (1.18.14) is the scalar version of Eq. (1.6.23). □

Example 1.18.5: Show that the multivariate sample covariance matrix, $\hat{R}$, given by Eq. (1.6.21) is the maximum likelihood estimate of R, assuming the mean is zero.

Solution: The log-likelihood function is, up to a constant

$$\ln p(\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N) = -\frac{N}{2}\ln(\det R) - \frac{1}{2}\sum_{n=1}^{N} \mathbf{y}_n^T R^{-1} \mathbf{y}_n$$

The second term may be written as the trace:

$$\sum_{n=1}^{N} \mathbf{y}_n^T R^{-1} \mathbf{y}_n = \operatorname{tr}[R^{-1} \sum_{n=1}^{N} \mathbf{y}_n \mathbf{y}_n^T] = N \operatorname{tr}[R^{-1}\hat{R}]$$

where we used $\sum_{n=1}^{N} \mathbf{y}_n \mathbf{y}_n^T = N\hat{R}$. Using the matrix property $\ln(\det R) = \operatorname{tr}(\ln R)$, we may write the log-likelihood in the form

$$\ln p = -\frac{N}{2}\operatorname{tr}[\ln R + R^{-1}\hat{R}]$$

The maximum likelihood solution for R satisfies $\partial \ln p / \partial R = 0$. To solve it, we find it more convenient to work with differentials. Using the two matrix properties

$$d\operatorname{tr}(\ln R) = \operatorname{tr}(R^{-1} dR), \quad dR^{-1} = -R^{-1}(dR)R^{-1} \tag{1.18.15}$$

we obtain,

$$d\ln p = -\frac{N}{2}\operatorname{tr}[R^{-1}dR - R^{-1}(dR)R^{-1}\hat{R}] = -\frac{N}{2}\operatorname{tr}[R^{-1}(dR)R^{-1}(R - \hat{R})] \tag{1.18.16}$$

Because dR is arbitrary, the vanishing of $d\ln p$ implies $R = \hat{R}$. An alternative proof is to show that $f(R) \geq f(\hat{R})$, where $f(R) \equiv \operatorname{tr}(\ln R + R^{-1}\hat{R})$. This is shown easily using the inequality $x - 1 - \ln x \geq 0$, for $x \geq 0$, with equality reached at $x = 1$. □

In many applications, the desired parameter $\boldsymbol{\lambda}$ to be estimated appears only through the covariance matrix R of the observations $\mathbf{y}$, that is, $R = R(\boldsymbol{\lambda})$. For example, we will see in Chap. 14 that the covariance matrix of a plane wave incident on an array of two sensors in the presence of noise is given by

$$R = \begin{bmatrix} P + \sigma^2 & Pe^{jk} \\ Pe^{-jk} & P + \sigma^2 \end{bmatrix}$$

where possible parameters to be estimated are the power P and wavenumber k of the wave, and the variance σ^2 of the background noise. Thus, $\boldsymbol{\lambda} = [P, k, \sigma^2]^T$.

In such cases, we have the following general expression for the Fisher information matrix J, valid for independent zero-mean gaussian observations:

$$J_{ij} = \frac{N}{2}\operatorname{tr}\left[R^{-1}\frac{\partial R}{\partial \lambda_i}R^{-1}\frac{\partial R}{\partial \lambda_j}\right] \tag{1.18.17}$$

Writing $\partial_i = \partial/\partial\lambda_i$ for brevity, we have from Eq. (1.18.16)

$$\partial_i \ln p = -\frac{N}{2}\operatorname{tr}[R^{-1}\partial_i R\, R^{-1}(R - \hat{R})]$$

Differentiating once more, we find

$$\Psi_{ij} = -\partial_i\partial_j \ln p = \frac{N}{2}\operatorname{tr}[\partial_j(R^{-1}\partial_i R\, R^{-1})(R - \hat{R}) + R^{-1}\partial_i R\, R^{-1}\partial_j R]$$

Equation (1.18.17) follows now by taking expectation values $J_{ij} = E[\Psi_{ij}]$ and noting that the expectation value of the first term vanishes. This follows from the fact that $\hat{R}$ is an unbiased estimator of R and therefore, $E[\operatorname{tr}(F(R - \hat{R}))] = 0$, for any matrix F.

1.19 Minimum-Phase Signals and Filters

A *minimum-phase sequence* $\mathbf{a} = [a_0, a_1, \ldots, a_M]$ has a z-transform with all its zeros inside the unit circle in the complex z-plane

$$A(z) = a_0 + a_1 z^{-1} + \cdots + a_M z^{-M} = a_0(1 - z_1 z^{-1})(1 - z_2 z^{-1}) \cdots (1 - z_M z^{-1}) \tag{1.19.1}$$

with $|z_i| < 1$, $i = 1, 2, \ldots, M$. Such a polynomial is also called a *minimum-delay* polynomial. Define the following related polynomials:

$$A^*(z) = a_0^* + a_1^* z^{-1} + \cdots + a_M^* z^{-M} = \text{complex-conjugated coefficients}$$

$$\bar{A}(z) = a_0^* + a_1^* z + \cdots + a_M^* z^M = \text{conjugated and reflected}$$

$$A^R(z) = a_M^* + a_{M-1}^* z^{-1} + \cdots + a_0^* z^{-M} = \text{reversed and conjugated}$$

We note the relationships:

$$\bar{A}(z) = A^*(z^{-1}) \quad \text{and} \quad A^R(z) = z^{-M}\bar{A}(z) = z^{-M}A^*(z^{-1}) \tag{1.19.2}$$

We also note that when we set $z = e^{j\omega}$ to obtain the corresponding frequency responses, $\bar{A}(\omega)$ becomes the complex conjugate of $A(\omega)$

$$\bar{A}(\omega) = A(\omega)^* \tag{1.19.3}$$

It is easily verified that all these polynomials have the *same* magnitude spectrum:

$$|A(\omega)|^2 = |\bar{A}(\omega)|^2 = |A^*(\omega)|^2 = |A^R(\omega)|^2 \tag{1.19.4}$$

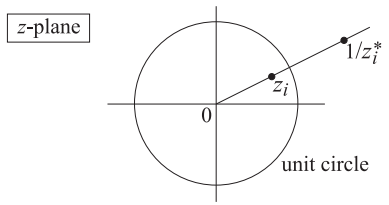
For example, in the case of a doublet $\mathbf{a} = (a_0, a_1)$ and its reverse $\mathbf{a}^R = (a_1^*, a_0^*)$, we verify explicitly

$$\begin{aligned} |A(\omega)|^2 = A(\omega)A(\omega)^* &= (a_0 + a_1 e^{-j\omega})(a_0^* + a_1^* e^{j\omega}) \\ &= (a_1^* + a_0^* e^{-j\omega})(a_1 + a_0 e^{j\omega}) \\ &= A^R(\omega)A^R(\omega)^* = |A^R(\omega)|^2 \end{aligned}$$

Thus, on the basis the magnitude spectrum, one cannot distinguish the doublet $\mathbf{a} = (a_0, a_1)$ from its reverse $\mathbf{a}^R = (a_1^*, a_0^*)$. In the more general case of a polynomial of degree M, factored into doublets as in Eq. (1.19.1), we note that each doublet can be replaced by its reverse

$$(1, -z_i) \to (-z_i^*, 1) \quad \text{or} \quad (1 - z_i z^{-1}) \to (-z_i^* + z^{-1})$$

without affecting the overall magnitude spectrum $|A(\omega)|^2$. Since there are M such factors, there will be a total of 2^M different Mth degree polynomials, or equivalently, 2^M different length-$(M+1)$ sequences, all having the *same* magnitude spectrum. Every time a factor $(1 - z_i z^{-1})$ is reversed to become $(-z_i^* + z^{-1})$, the corresponding zero changes from $z = z_i$ to $z = 1/z_i^*$. If z_i is inside the unit circle, the $1/z_i^*$ is outside, as shown



To enumerate all these sequences, start by taking all zeros z_i to be inside the unit circle and successively keep reversing each factor until all 2^M possibilities have been exhausted. At the last step, all the factors will have been flipped, corresponding to all the zeros being outside the unit circle. The resulting polynomial and sequence are referred to as having *maximal phase*, or *maximal delay*. As an example consider the two doublets

$$\mathbf{a} = (2, 1) \quad \text{and} \quad \mathbf{b} = (3, 2)$$

and form the four different sequences, where $*$ denotes convolution:

$$\begin{aligned} \mathbf{c}_0 = \mathbf{a} * \mathbf{b} &= (2,1) * (3,2) = (6,7,2), & C_0(z) &= A(z)B(z) \\ \mathbf{c}_1 = \mathbf{a}^R * \mathbf{b} &= (1,2) * (3,2) = (3,8,4), & C_1(z) &= A^R(z)B(z) \\ \mathbf{c}_2 = \mathbf{a} * \mathbf{b}^R &= (2,1) * (2,3) = (4,8,3), & C_2(z) &= A(z)B^R(z) \\ \mathbf{c}_3 = \mathbf{a}^R * \mathbf{b}^R &= (1,2) * (2,3) = (2,7,6), & C_3(z) &= A(z)B(z) \end{aligned}$$

All four sequences have the same magnitude spectrum.

Partial Energy and Minimal Delay

Since the total energy of a sequence $\mathbf{a} = (a_0, a_1, \ldots, a_M)$ is given by Parseval's equality

$$\sum_{m=0}^{M} |a_m|^2 = \int_{-\pi}^{\pi} |A(\omega)|^2 \frac{d\omega}{2\pi}$$

it follows that all of the above 2^M sequences, having the same magnitude spectrum, will also have the same *total energy*. However, the *distribution* of the total energy over time may be different. And this will allow an alternative characterization of the minimum phase sequences, first given by Robinson. Define the partial energy by

$$P_a(n) = \sum_{m=0}^{n} |a_m|^2 = |a_0|^2 + |a_1|^2 + \cdots + |a_n|^2, \quad n = 0, 1, \ldots, M$$

then, for the above example, the partial energies for the four different sequences are given in the table

	$\mathbf{c}_0$	$\mathbf{c}_1$	$\mathbf{c}_2$	$\mathbf{c}_3$
$P(0)$	36	9	16	4
$P(1)$	85	73	80	53
$P(2)$	89	89	89	89

We note that $\mathbf{c}_0$ which has both its zeros inside the unit circle (i.e., minimal phase) is also the sequence that has most of its energy concentrated at the *earlier* times, that is, it makes its impact as early as possible, with *minimal delay*. In contrast, the maximal-phase sequence $\mathbf{c}_3$ has most of its energy concentrated at its tail thus, making most of its impact at the end, with maximal delay.

Invariance of the Autocorrelation Function

This section presents yet another characterization of the above class of sequences. It will be important in proving the minimum-phase property of the linear prediction filters.

The *sample autocorrelation* of a (possibly complex-valued) sequence $\mathbf{a} = (a_0, a_1, \ldots, a_M)$ is defined by

$$\begin{aligned} R_{aa}(k) &= \sum_{n=0}^{M-k} a_{n+k} a_n^*, \quad \text{for} \quad 0 \le k \le M \\ R_{aa}(k) &= R_{aa}(-k)^*, \quad \text{for} \quad -M \le k \le -1 \end{aligned} \tag{1.19.5}$$

It is easily verified that the corresponding power spectral density is factored as

$$S_{aa}(z) = \sum_{k=-M}^{M} R_{aa}(k) z^{-k} = A(z)\bar{A}(z) \tag{1.19.6}$$

The magnitude response is obtained by setting $z = e^{j\omega}$

$$S_{aa}(\omega) = |A(\omega)|^2 \tag{1.19.7}$$

with an inversion formula

$$R_{aa}(k) = \int_{-\pi}^{\pi} |A(\omega)|^2 e^{j\omega k} \frac{d\omega}{2\pi} \tag{1.19.8}$$

It follows from Eq. (1.19.8) that the above 2^M different sequences having the same magnitude spectrum, also have the same sample *autocorrelation*. They cannot be distinguished on the basis of their autocorrelation. Therefore, there are 2^M different spectral factorizations of $S_{aa}(z)$ of the form

$$S_{aa}(z) = A(z)\bar{A}(z) \tag{1.19.9}$$

but there is only one with minimum-phase factors. The procedure for obtaining it is straightforward: Find the zeros of $S_{aa}(z)$, which come in pairs z_i and $1/z_i^*$, thus, there are $2M$ such zeros. And, group those that lie inside the unit circle into a common factor. This defines $A(z)$ as a minimum phase polynomial.

Minimum-Delay Property

Here, we discuss the effect of flipping a zero from the inside to the outside of the unit circle, on the minimum-delay and minimum-phase properties of the signal. Suppose $A(z)$ is of degree M and has a zero z_1 inside the unit circle. Let $B(z)$ be the polynomial that results by flipping this zero to the outside; that is, $z_1 \to 1/z_1^*$

$$\begin{aligned} A(z) &= (1 - z_1 z^{-1})F(z) \\ B(z) &= (-z_1^* + z^{-1})F(z) \end{aligned} \tag{1.19.10}$$

where $F(z)$ is a polynomial of degree $M-1$. Both $A(z)$ and $B(z)$ have the same magnitude spectrum. We may think of this operation as sending $A(z)$ through an *allpass* filter

$$B(z) = \frac{-z_1^* + z^{-1}}{1 - z_1 z^{-1}} A(z)$$

In terms of the polynomial coefficients, Eq. (1.19.10) becomes

$$\begin{aligned} a_n &= f_n - z_1 f_{n-1} \\ b_n &= -z_1^* f_n + f_{n-1} \end{aligned} \tag{1.19.11}$$

for $n = 0, 1, \ldots, M$, from which we obtain

$$|a_n|^2 - |b_n|^2 = (1 - |z_1|^2)(|f_n|^2 - |f_{n-1}|^2) \tag{1.19.12}$$

Summing to get the partial energies, $P_a(n) = \sum_{m=0}^{n} |a_m|^2$, we find

$$P_a(n) - P_b(n) = (1 - |z_1|^2)|f_n|^2, \quad n = 0, 1, \ldots, M \tag{1.19.13}$$

Thus, the partial energy of the sequence **a** remains greater than that of **b** for all times n; that is, $A(z)$ is of earlier delay than $B(z)$. The total energy is, of course, the same

as follows from the fact that $F(z)$ is of degree $M-1$, thus, missing the Mth term or $f_M = 0$. We have then

$$P_a(n) \geq P_b(n), \quad n = 0, 1, \ldots, M$$

and in particular

$$P_a(M) = P_b(M) \quad \text{and} \quad P_a(0) \geq P_b(0)$$

The last inequality can also be stated as $|a_0| \geq |b_0|$, and will be important in our proof of the minimum-phase property of the prediction-error filter of linear prediction.

Minimum-Phase Property

The effect of reversing the zero z_1 on the phase responses of $A(z)$ and $B(z)$ of Eq. (1.19.10) can be seen as follows. For $z = e^{j\omega}$, define the *phase lag* as the negative of the phase response

$$A(\omega) = |A(\omega)|\, e^{j\mathrm{Arg}(\omega)}$$

$$\theta_A(\omega) = -\mathrm{Arg}(\omega) = \text{phase-lag response}$$

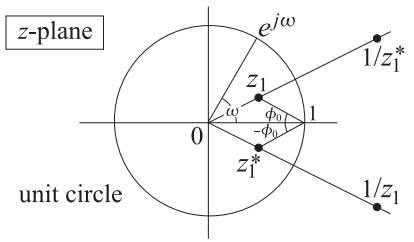
and similarly for $B(z)$. Since $A(\omega)$ and $B(\omega)$ have the same magnitude, they will differ only by a phase

$$\frac{A(\omega)}{B(\omega)} = e^{j(\theta_B - \theta_A)} = \frac{1 - z_1 e^{-j\omega}}{-z_1^* + e^{-j\omega}} = \frac{e^{j\omega} - z_1}{1 - z_1^* e^{j\omega}} = e^{j\phi(\omega)}$$

where $\phi(\omega)$ is the phase-response of the all-pass factor $(e^{j\omega} - z_1)/(1 - z_1^* e^{j\omega})$, so that $\theta_B(\omega) - \theta_A(\omega) = \phi(\omega)$. By taking derivatives with respect to ω in the above definition of $\phi(\omega)$, it can be easily shown that

$$\frac{d\phi(\omega)}{d\omega} = \frac{1 - |z_1|^2}{|e^{j\omega} - z_1|^2} > 0$$

which shows that $\phi(\omega)$ is an increasing function of ω. Thus, over the frequency interval $0 \leq \omega \leq \pi$, we have $\phi(\omega) \geq \phi(0)$. It can be verified easily that $\phi(0) = -2\phi_0$, where ϕ_0 is the angle with the x-axis of the line between the points z_1 and 1, as shown in the figure below.



Thus, we have $\theta_B - \theta_A \geq \phi \geq -2\phi_0$. The angle ϕ_0 is positive, if z_1 lies within the upper half semi-circle, and negative, if it lies in the lower semi-circle; and, ϕ_0 is zero if z_1 lies on the real axis. If z_1 is real-valued, then $\theta_B \geq \theta_A$ for $0 \leq \omega \leq \pi$. If z_1

is complex valued and we consider the combined effect of flipping the zero z_1 and its conjugate z_1^*, that is, $A(z)$ and $B(z)$ are given by

$$A(z) = (1 - z_1 z^{-1})(1 - z_1^* z^{-1})F(z)$$

$$B(z) = (-z_1^* + z^{-1})(-z_1 + z^{-1})F(z)$$

then, for the phase of the combined factor

$$e^{j\phi(\omega)} = \frac{e^{j\omega} - z_1}{1 - z_1^* e^{j\omega}} \cdot \frac{e^{j\omega} - z_1^*}{1 - z_1 e^{j\omega}}$$

we will have $\phi(\omega) \geq (-2\phi_0) + (2\phi_0) = 0$, so that $\theta_B(\omega) - \theta_A(\omega) = \phi(\omega) \geq 0$.

Thus, the phase lag of $A(z)$ remains smaller than that of $B(z)$. The phase-lag curve for the case when $A(z)$ has all its zeros inside the unit circle will remain below all the other phase-lag curves. The term *minimum-phase* strictly speaking means minimum phase lag (over $0 \leq \omega \leq \pi$).

1.20 Spectral Factorization Theorem

We finish our digression on minimum-phase sequences by quoting the spectral factorization theorem [5].

Any *rational* power spectral density $S_{yy}(z)$ of a (real-valued) stationary signal y_n can be factored in a minimum-phase form

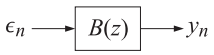
$$S_{yy}(z) = \sigma_\epsilon^2 B(z) B(z^{-1}) \tag{1.20.1}$$

where

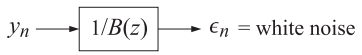
$$B(z) = \frac{N(z)}{D(z)} \tag{1.20.2}$$

with both $D(z)$ and $N(z)$ being minimum-phase polynomials; that is, having all their zeros inside the unit circle. By adjusting the overall constant σ_ϵ^2, both $D(z)$ and $N(z)$ may be taken to be *monic* polynomials. Then, they are *unique*.

This theorem guarantees the existence of a *causal and stable* random signal generator filter $B(z)$ for the signal y_n of the type discussed in Sec. 1.13:



with ϵ_n white noise of variance σ_ϵ^2. The minimum-phase property of $B(z)$ also guarantees the stability and causality of the inverse filter $1/B(z)$, that is, the whitening filter



The proof of the spectral factorization theorem is straightforward. Since $S_{yy}(z)$ is the power spectral density of a (real-valued) stationary process y_n, it will satisfy the symmetry conditions $S_{yy}(z) = S_{yy}(z^{-1})$. Therefore, if z_i is a zero then $1/z_i$ is also a zero, and if z_i is complex then the reality of $R_{yy}(k)$ implies that z_i^* will also be a

zero. Thus, both z_i and $1/z_i^*$ are zeros. Therefore, the numerator polynomial of $S_{yy}(z)$ is of the type of Eq. (1.19.9) and can be factored into its minimum phase polynomials $N(z)N(z^{-1})$. This is also true of the denominator of $S_{yy}(z)$.

All sequential correlations in the original signal y_n arise from the *filtering* action of $B(z)$ on the white-noise input ϵ_n. This follows from Eq. (1.12.14):

$$R_{yy}(k) = \sigma_\epsilon^2 \sum_n b_{n+k} b_n\,, \quad B(z) = \sum_{n=0}^{\infty} b_n z^{-n} \tag{1.20.3}$$

Effectively, we have modeled the statistical autocorrelation $R_{yy}(k)$ by the sample autocorrelation of the impulse response of the synthesis filter $B(z)$. Since $B(z)$ is causal, such factorization corresponds to an LU, or Cholesky, factorization of the autocorrelation matrix.

This matrix representation can be seen as follows: Let B be the lower triangular Toeplitz matrix defined exactly as in Eq. (1.13.2)

$$b_{ni} = b_{n-i}$$

and let the autocorrelation matrix of y_n be

$$R_{yy}(i,j) = R_{yy}(i-j)$$

Then, the transposed matrix B^T will have matrix elements

$$(B^T)_{ni} = b_{i-n}$$

and Eq. (1.20.3) can be written in the form

$$R_{yy}(i,j) = R_{yy}(i-j) = \sigma_\epsilon^2 \sum_n b_{n+i-j} b_n = \sigma_\epsilon^2 \sum_k b_{i-k} b_{j-k}$$

$$= \sigma_\epsilon^2 \sum_k (B)_{ik} (B^T)_{kj} = \sigma_\epsilon^2 (BB^T)_{ij}$$

Thus, in matrix notation

$$R_{yy} = \sigma_\epsilon^2 BB^T \tag{1.20.4}$$

This equation is a special case of the more general LU factorization of the Gram-Schmidt construction given by Eq. (1.6.17). Indeed, the assumption of stationarity implies that the quantity

$$\sigma_\epsilon^2 = E[\epsilon_n^2]$$

is independent of the time n, and therefore, the diagonal matrix $R_{\epsilon\epsilon}$ of Eq. (1.6.17) becomes a multiple of the identity matrix.

1.21 Minimum-Phase Property of the Prediction-Error Filter

The minimum-phase property of the prediction-error filter $A(z)$ of linear prediction is an important property because it guarantees the stability of the causal inverse synthesis filter $1/A(z)$. There are many proofs of this property in the literature [6–10]. Here, we

would like to present a simple proof [11] which is based directly on the fact that the optimal prediction coefficients minimize the mean-square prediction error. Although we have only discussed first and second order linear predictors, for the purposes of this proof we will work with the more general case of an Mth order predictor defined by

$$\hat{y}_n = -[a_1 y_{n-1} + a_2 y_{n-2} + \cdots + a_M y_{n-M}]$$

which is taken to represent the best prediction of y_n based on the past M samples $Y_n = \{y_{n-1}, y_{n-2}, \ldots, y_{n-M}\}$. The corresponding prediction error is

$$e_n = y_n - \hat{y}_n = y_n + a_1 y_{n-1} + a_2 y_{n-2} + \cdots + a_M y_{n-M}$$

The best set of prediction coefficients $\{a_1, a_2, \ldots a_M\}$ is found by minimizing the mean-square prediction error

$$\begin{aligned} \mathcal{E}(a_1, a_2, \ldots a_M) &= E[e_n^* e_n] = \sum_{m,k=0}^{M} a_m^* E[y_{n-m}^* y_{n-k}] a_k \\ &= \sum_{m,k=0}^{M} a_m^* R_{yy}(k-m) a_k = \sum_{m,k=0}^{M} a_m^* R_{yy}(m-k) a_k \end{aligned} \tag{1.21.1}$$

where we set $a_0 = 1$. For the proof of the minimum phase property, we do not need the explicit solution of this minimization problem; we only use the fact that the optimal coefficients minimize Eq. (1.21.1). The key to the proof is based on the observation that (3.7.1) can be written in the alternative form

$$\mathcal{E}(\mathbf{a}) = \sum_{k=-M}^{M} R_{yy}(k) R_{aa}(k) \tag{1.21.2}$$

where $R_{aa}(k)$ is the sample autocorrelation of the prediction-error filter sequence $\mathbf{a} = [1, a_1, a_2, \ldots, a_M]^T$ as defined in Eq. (1.19.5). The equivalence of Eqs. (1.21.1) and (1.21.2) can be seen easily, either by rearranging the summation indices of (1.21.1), or by using the results of Problems 1.37 and 1.39.

Example 1.21.1: We demonstrate this explicitly for the $M = 2$ case. Using the definition (1.19.5) we have

$$\begin{aligned} R_{aa}(0) &= |a_0|^2 + |a_1|^2 + |a_2|^2 = 1 + |a_1|^2 + |a_2|^2 \\ R_{aa}(1) &= R_{aa}(-1)^* = a_1 a_0^* + a_2 a_1^* = a_1 + a_2 a_1^* \\ R_{aa}(2) &= R_{aa}(-2)^* = a_2 a_0^* = a_2 \end{aligned}$$

Since y_n is real-valued stationary, we have $R_{yy}(k) = R_{yy}(-k)$. Then, Eq. (1.21.1) becomes explicitly

$$\mathcal{E}(\mathbf{a}) = \sum_{m,k=0}^{M} a_m^* R_{yy}(m-k) a_k = [1, a_1^*, a_2^*] \begin{bmatrix} R_{yy}(0) & R_{yy}(1) & R_{yy}(2) \\ R_{yy}(1) & R_{yy}(0) & R_{yy}(1) \\ R_{yy}(0) & R_{yy}(1) & R_{yy}(2) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ a_2 \end{bmatrix}$$

$$= R_{yy}(0)[1 + a_1^* a_1 + a_2^* a_2] + R_{yy}(1)[(a_1 + a_2 a_1^*) + (a_1^* + a_2^* a_1)] + R_{yy}(2)[a_2 + a_2^*]$$

$$= R_{yy}(0) R_{aa}(0) + R_{yy}(1)[R_{aa}(1) + R_{aa}(-1)] + R_{yy}(2)[R_{aa}(2) + R_{aa}(-2)] \qquad \square$$

Let $\mathbf{a} = [1, a_1, a_2, \ldots, a_M]^T$ be the optimal set of coefficients that minimizes $\mathcal{E}(\mathbf{a})$ and let z_i, $i = 1, 2 \ldots, M$, be the zeros of the corresponding prediction-error filter:

$$1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_M z^{-M} = (1 - z_1 z^{-1})(1 - z_2 z^{-1}) \cdots (1 - z_M z^{-1}) \tag{1.21.3}$$

Reversing any one of the zero factors in this equation, that is, replacing $(1 - z_i z^{-1})$ by its reverse $(-z_i^* + z^{-1})$, results in a sequence that has the same sample autocorrelation as $\mathbf{a}$. As we have seen, there are 2^M such sequences, all with the same sample autocorrelation. We would like to show that among these, $\mathbf{a}$ is the one having the minimum-phase property.

To this end, let $\mathbf{b} = [b_0, b_1, \ldots b_M]^T$ be any one of these 2^M sequences, and define the normalized sequence

$$\mathbf{c} = \mathbf{b}/b_0 = [1, b_1/b_0, b_2/b_0, \ldots b_M/b_0]^T \tag{1.21.4}$$

Using the fact that $\mathbf{b}$ has the same sample autocorrelation as $\mathbf{a}$, we find for the sample autocorrelation of $\mathbf{c}$:

$$R_{cc}(k) = R_{bb}(k)/|b_0|^2 = R_{aa}(k)/|b_0|^2 \tag{1.21.5}$$

The performance index (1.21.2) evaluated at $\mathbf{c}$ is then

$$\mathcal{E}(\mathbf{c}) = \sum_{k=-M}^{M} R_{yy}(k) R_{cc}(k) = \sum_{k=-M}^{M} R_{yy}(k) R_{aa}(k)/|b_0|^2 \tag{1.21.6}$$

or,

$$\mathcal{E}(\mathbf{c}) = \mathcal{E}(\mathbf{a})/|b_0|^2 \tag{1.21.7}$$

Since $\mathbf{a}$ minimizes $\mathcal{E}$, it follows that $\mathcal{E}(\mathbf{c}) \geq \mathcal{E}(\mathbf{a})$. Therefore, Eq. (1.21.7) implies that

$$|b_0| \leq 1 \tag{1.21.8}$$

This must be true of all $\mathbf{b}$s in the above class. Eq. (1.21.8) then, immediately implies the minimum-phase property of $\mathbf{a}$. Indeed, choosing $\mathbf{b}$ to be that sequence obtained from (1.21.3) by reversing only the ith zero factor $(1 - z_i z^{-1})$ and not the other zero factors, it follows that

$$b_0 = -z_i^*$$

and therefore Eq. (1.21.8) implies that

$$|z_i| \leq 1 \tag{1.21.9}$$

which shows that all the zeros of $A(z)$ are inside the unit circle and thus, $A(z)$ has minimum phase. An alternative proof based on the Levinson recursion and Rouche's theorem of complex analysis will be presented in Chap. 12.

1.22 Computer Project – Adaptive AR(1) and AR(2) Models

This computer project, divided into separate parts, deals with adaptive AR models that are capable of tracking time-varying systems. It is also applied to the benchmark sunspot data, comparing the results with Yule's original application of an AR(2) model.

1. *Time-varying AR(1) model.* Consider the following AR(1), first-order, autoregressive signal model with a time-varying parameter:

$$y_n = a(n)y_{n-1} + \epsilon_n \tag{1.22.1}$$

where ϵ_n is zero-mean, unit-variance, white noise. The filter parameter $a(n)$ can be tracked by the following adaptation equations (which are equivalent to the exact recursive least-squares order-1 adaptive predictor):

$$\begin{aligned} R_0(n) &= \lambda R_0(n-1) + \alpha\, y_{n-1}^2 \\ R_1(n) &= \lambda R_1(n-1) + \alpha\, y_n y_{n-1} \\ \hat{a}(n) &= \frac{R_1(n)}{R_0(n)} \end{aligned}$$

where $\alpha = 1-\lambda$. The two filtering equations amount to sending the quantities y_{n-1}^2 and $y_n y_{n-1}$ through an exponential smoother. To avoid possible zero denominators, initialize R_0 to some small positive constant, $R_0(-1) = \delta$, such as $\delta = 10^{-3}$.

(a) Show that $\hat{a}(n)$ satisfies the recursion:

$$\boxed{\hat{a}(n) = \hat{a}(n-1) + \frac{\alpha}{R_0(n)} y_{n-1} e_{n/n-1}} \qquad e_{n/n-1} = y_n - \hat{a}(n-1)y_{n-1} \tag{1.22.2}$$

where $e_{n/n-1}$ is referred to as the a priori estimation (prediction) error.

(b) Using Eq. (1.22.1), generate a data sequence y_n, $n = 0, 1, \ldots, N-1$ using the following time varying coefficient, sinusoidally switching from a positive value to a negative one (the synthesis filter switches from lowpass to highpass):

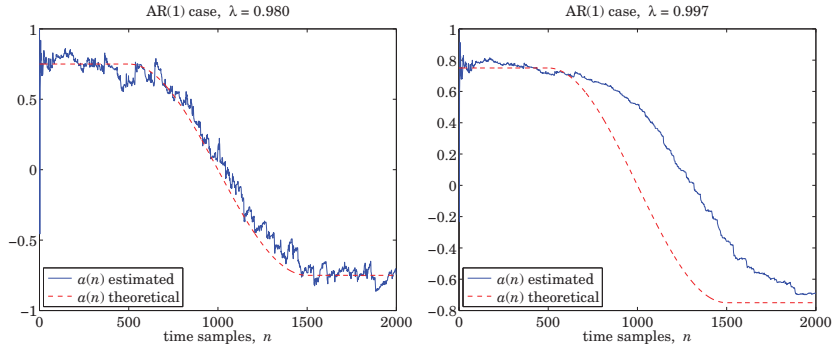
$$a(n) = \begin{cases} 0.75, & 0 \le n \le N_a - 1 \\ 0.75\cos\left(\pi \dfrac{n - N_a}{N_b - N_a}\right), & N_a \le n \le N_b \\ -0.75, & N_b + 1 \le n \le N - 1 \end{cases}$$

Use the following numerical values:

$$N_a = 500, \quad N_b = 1500, \quad N = 2000$$

Calculate the estimated $\hat{a}(n)$ using the recursion (1.22.2) and plot it versus n together with the theoretical $a(n)$ using the parameter value $\lambda = 0.980$. Repeat using the value $\lambda = 0.997$. Comment on the tracking capability of the algorithm versus the accuracy of the estimate.

(c) Study the sensitivity of the algorithm to the initialization parameter δ.



2. *Tine-varying AR(2) model.* Next, consider an AR(2), second-order, model with time-varying coefficients:

$$y_n = -a_1(n)y_{n-1} - a_2(n)y_{n-2} + \epsilon_n \tag{1.22.3}$$

If the coefficients were stationary, then the theoretical Wiener solution for the prediction coefficients a_1 and a_2 would be:

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = -\begin{bmatrix} R_0 & R_1 \\ R_1 & R_0 \end{bmatrix}^{-1} \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} = -\frac{1}{R_0^2 - R_1^2} \begin{bmatrix} R_0 R_1 - R_1 R_2 \\ R_0 R_2 - R_1^2 \end{bmatrix} \tag{1.22.4}$$

where $R_k = E[y_n y_{n-k}]$. For a time-varying model, the coefficients can be tracked by replacing the theoretical autocorrelation lags R_k with their recursive, exponentially smoothed, versions:

$$\begin{aligned} R_0(n) &= \lambda R_0(n-1) + \alpha\, y_n^2 \\ R_1(n) &= \lambda R_1(n-1) + \alpha\, y_n y_{n-1} \\ R_2(n) &= \lambda R_2(n-1) + \alpha\, y_n y_{n-2} \end{aligned}$$

(a) Using Eq. (1.22.3), generate a non-stationary data sequence y_n by driving the second-order model with a unit-variance, zero-mean, white noise signal ϵ_n and using the following theoretical time-varying coefficients:

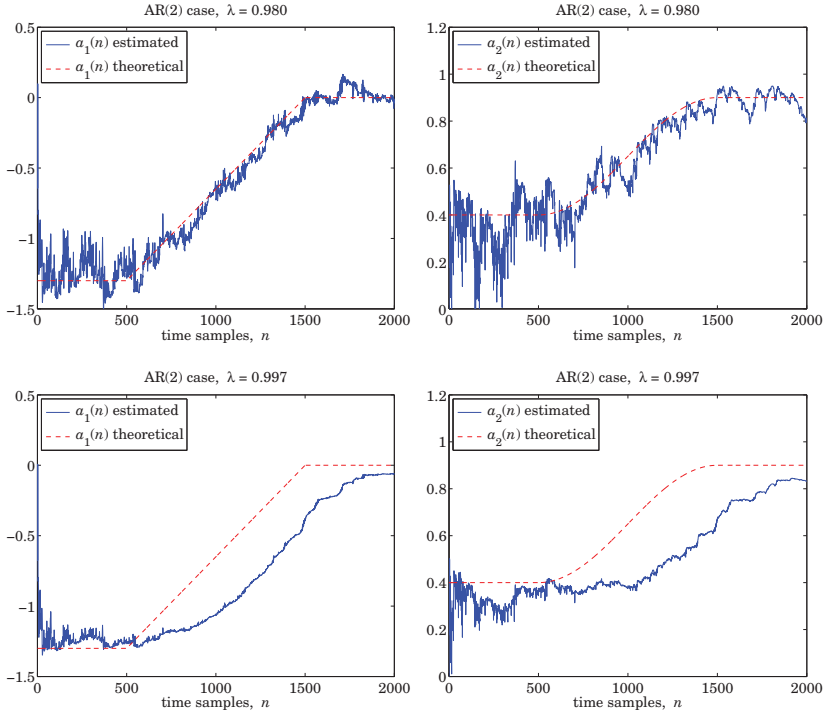
$$a_1(n) = \begin{cases} -1.3, & 0 \le n \le N_a - 1 \\ 1.3\,\dfrac{n - N_b}{N_b - N_a}, & N_a \le n \le N_b \\ 0, & N_b + 1 \le n \le N - 1 \end{cases}$$

$$a_2(n) = \begin{cases} 0.4, & 0 \le n \le N_a - 1 \\ 0.65 - 0.25\cos\left(\pi \dfrac{n - N_a}{N_b - N_a}\right), & N_a \le n \le N_b \\ 0.9, & N_b + 1 \le n \le N - 1 \end{cases}$$

Thus, the signal model for y_n switches continuously between the synthesis filters:

$$B(z) = \frac{1}{1 - 1.3z^{-1} + 0.4z^{-2}} \quad \Rightarrow \quad B(z) = \frac{1}{1 + 0.9z^{-2}}$$

(b) Compute the adaptive coefficients $\hat{a}_1(n)$ and $\hat{a}_2(n)$ using the two forgetting factors $\lambda = 0.980$ and $\lambda = 0.997$. Plot the adaptive coefficients versus n, together with the theoretical time-varying coefficients and discuss the tracking capability of the adaptive processor.



3. *AR(2) modeling of sunspot data.* Next, we will apply the adaptive method of part-2 to some real data. The file `sunspots.dat` contains the yearly mean number of sunspots for the years 1700–2008. To unclutter the resulting graphs, we will use only the data for the last 200 years, over 1809–2008. These can be read into MATLAB as follows:

```
Y = loadfile('sunspots.dat');
i = find(Y(:,1)==1809);
y = Y(i:end,2);                % number of sunspots
N = length(y);                 % here, N=200
m = mean(y); y = y-m;          % zero-mean data
```

where the last line determines the mean of the data block and subtracts it from the data. The mean m will be restored at the end.

Yule was the first to introduce the concept of an autoregressive signal model and applied it to the sunspot time series assuming a second-order model. The so-called Yule-Walker method is a block processing method in which the entire (zero-mean) data block is used to estimate the autocorrelation lags R_0, R_1, R_2 using sample autocorrelations:

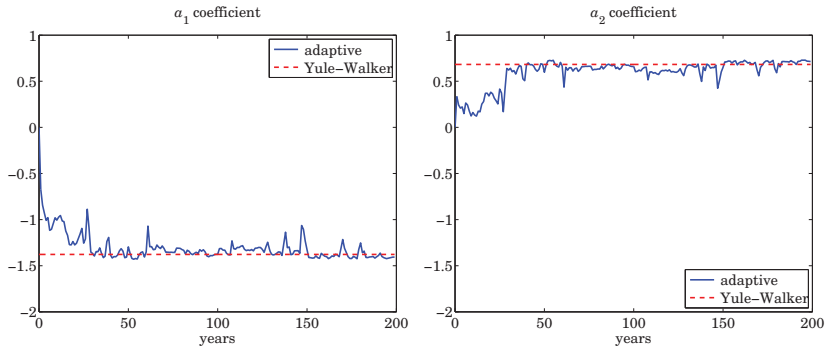
$$\hat{R}_0 = \frac{1}{N}\sum_{n=0}^{N-1} y_n^2\,, \quad \hat{R}_1 = \frac{1}{N}\sum_{n=0}^{N-2} y_{n+1}y_n\,, \quad \hat{R}_2 = \frac{1}{N}\sum_{n=0}^{N-3} y_{n+2}y_n$$

Then, the model parameters a_1, a_2 are estimated using Eq. (1.22.4):

$$\begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \end{bmatrix} = -\begin{bmatrix} \hat{R}_0 & \hat{R}_1 \\ \hat{R}_1 & \hat{R}_0 \end{bmatrix}^{-1} \begin{bmatrix} \hat{R}_1 \\ \hat{R}_2 \end{bmatrix} \qquad \text{(Yule-Walker method)}$$

(a) First, compute the values of $\hat{a}_1, \hat{a}_2$ based on the given length-200 data block.

(b) Then, apply the adaptive algorithm of the part-2 with $\lambda = 0.99$ to determine the adaptive versions $a_1(n), a_2(n)$ and plot them versus n, and add on these graphs the straight lines corresponding to the Yule-Walker estimates $\hat{a}_1, \hat{a}_2$.

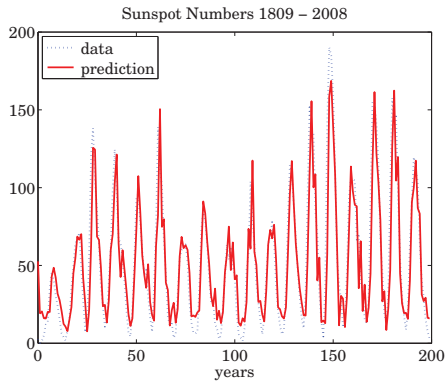


(c) At each time instant n, the value of y_n can be predicted by either of the two formulas:

$$\hat{y}_{n/n-1} = -a_1(n)y_{n-1} - a_2(n)y_{n-2}$$

$$\hat{y}_{n/n-1} = -\hat{a}_1 y_{n-1} - \hat{a}_2 y_{n-2}$$

On the same graph, plot y_n and $\hat{y}_{n/n-1}$ for the above two alternatives. The case of the adaptive predictor is shown below.



(d) Repeat the above questions using $\lambda = 0.95$ and discuss the effect of reducing λ.

(e) Apply a length-200 Hamming window w_n to the (zero-mean) data y_n and calculate the corresponding periodogram spectrum,

$$S_{\text{per}}(\omega) = \frac{1}{N}\left|\sum_{n=0}^{N-1} w_n y_n e^{-j\omega n}\right|^2$$

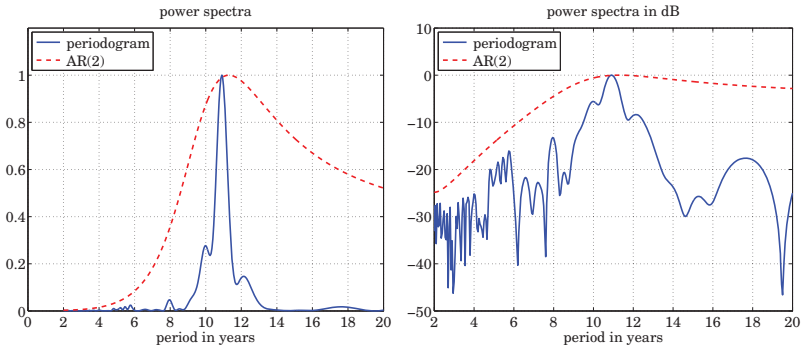
as a function of the yearly period $p = 2\pi/\omega$, over the range $2 \le p \le 20$ years. For the same p's or ω's calculate also the AR(2) spectrum using the Yule-Walker coefficients $\hat{a}_1, \hat{a}_2$:

$$S_{\text{AR}}(\omega) = \frac{\sigma_\epsilon^2}{\left|1 + \hat{a}_1 e^{-j\omega} + \hat{a}_2 e^{-2j\omega}\right|^2}$$

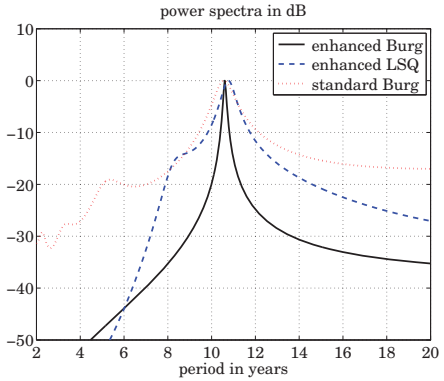
where σ_ϵ^2 can be calculated by

$$\sigma_\epsilon^2 = \hat{R}_0 + \hat{a}_1 \hat{R}_1 + \hat{a}_2 \hat{R}_2$$

Normalize the spectra $S_{\text{per}}(\omega), S_{\text{AR}}(\omega)$ to unity maxima and plot them versus period p on the same graph. Note that both predict the presence of an approximate 11-year cycle, which is also evident from the time data.



We will revisit this example later on by applying SVD methods to get sharper peaks. An example of the improved results is shown below.



1.23 Problems

1.1 Two dice are available for throwing. One is fair, but the other bears only sixes. One die is selected as follows: A coin is tossed. If the outcome is tails then the fair die is selected, but if the outcome is heads, the biased die is selected. The coin itself is not fair, and the probability of bearing heads or tails is $1/3$ or $2/3$, respectively. A die is now selected according to this procedure and tossed twice and the number of sixes is noted.

Let x be a random variable that takes on the value 0 when the fair die is selected or 1 if the biased die is selected. Let y be a random variable denoting the number of sixes obtained in the two tosses; thus, the possible values of y are $0, 1, 2$.

(a) For all possible values of x and y, compute $p(y|x)$, that is, the probability that the number of sixes will be y, given that the x die was selected.

(b) For each y, compute $p(y)$, that is, the probability that the number of sixes will be y, regardless of which die was selected.

(c) Compute the mean number of sixes $E[y]$.

(d) For all values of x and y, compute $p(x|y)$, that is, the probability that we selected die x, given that we already observed a y number of sixes.

1.2 *Inversion Method.* Let $F(x)$ be the cumulative distribution of a probability density $p(x)$. Suppose u is a uniform random number in the interval $[0, 1)$. Show that the solution of the equation $F(x) = u$, or equivalently, $x = F^{-1}(u)$, generates a random number x distributed according to $p(x)$. This is the inversion method of generating random numbers from uniform random numbers.

1.3 *Computer Experiment.* Let x be a random variable with the exponential probability density

$$p(x) = \frac{1}{\mu} e^{-x/\mu}$$

Show that x has mean μ and variance μ^2. Determine the cumulative distribution function $F(x)$ of x. Determine the inverse formula $x = F^{-1}(u)$ for generating x from a uniform

u. Take $\mu = 2$. Using the inversion formula and a uniform random number generator routine, generate a block of 200 random numbers x distributed according to $p(x)$. Compute their sample mean and sample variance, Eqs. (1.2.1) and (1.2.3), and compare them with their theoretical values. Do the estimated values fall within the standard deviation intervals defined by Eqs. (1.2.2) and (1.2.4)?

1.4 The Rayleigh probability density finds application in fading communication channels

$$p(r) = \frac{r}{\sigma^2} e^{-r^2/2\sigma^2}, \quad r \geq 0$$

Using the inversion method, $r = F^{-1}(u)$, show how to generate a Rayleigh-distributed random variable r from a uniform u.

1.5 (a) Following the notation of Sec. 1.4, show the matrix identity, where $H = R_{xy}R_{yy}^{-1}$

$$\left[\begin{array}{c|c} I_N & -H \\ \hline 0 & I_M \end{array}\right] \left[\begin{array}{c|c} R_{xx} & R_{xy} \\ \hline R_{yx} & R_{yy} \end{array}\right] \left[\begin{array}{c|c} I_N & -H \\ \hline 0 & I_M \end{array}\right]^T = \left[\begin{array}{c|c} R_{xx} - R_{xy}R_{yy}^{-1}R_{yx} & 0 \\ \hline 0 & R_{yy} \end{array}\right]$$

(b) Rederive the correlation canceling results of Eqs. (1.4.3) and (1.4.4) using this identity.

1.6 Using the matrix identity of Problem 1.5, derive directly the result of Example 1.4.1, that is, $E[\mathbf{x}|\mathbf{y}] = R_{xy}R_{yy}^{-1}\mathbf{y}$. Work directly with probability densities;

1.7 Show that the orthogonal projection $\hat{\mathbf{x}}$ of a vector $\mathbf{x}$ onto another vector $\mathbf{y}$, defined by Eq. (1.4.5) or Eq. (1.6.18), is a linear function of $\mathbf{x}$, that is, show

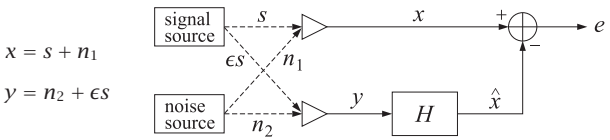
$$\widehat{A_1\mathbf{x}_1 + A_2\mathbf{x}_2} = A_1\hat{\mathbf{x}}_1 + A_2\hat{\mathbf{x}}_2$$

1.8 Suppose x consists of two components $x = s + n_1$, a desired component s, and a noise component n_1. Suppose that y is a related noise component n_2 to which we have access, $y = n_2$. The relationship between n_1 and n_2 is assumed to be linear, $n_1 = Fn_2$. For example, s might represent an electrocardiogram signal which is contaminated by 60 Hz power frequency pick-up noise n_1; then, a reference 60 Hz noise $y = n_2$, can be obtained from the wall outlet.

(a) Show that the correlation canceler is $H = F$, and that complete cancellation of n_1 takes place.

(b) If $n_1 = Fn_2 + v$, where v is uncorrelated with n_2 and s, show that $H = F$ still, and n_1 is canceled completely. The part v remains unaffected.

1.9 *Signal Cancellation Effects.* In the previous problem, we assumed that the reference signal y did not contain any part related to the desired component s. There are applications, however, where both the signal and the noise components contribute to both x and y, as for example in antenna sidelobe cancellation. Since the reference signal y contains part of s, the correlation canceler will act also to cancel part of the useful signal s from the output. To see this effect, consider a simple one-dimensional example



with $n_1 = Fn_2$, where we assume that y contains a small part proportional to the desired signal s. Assume that n_2 and s are uncorrelated. Show that the output e of the correlation canceler will contain a reduced noise component n_1 as well as a partially canceled signal s, as follows:

$$e = as + bn_1, \quad \text{where} \quad a = 1 - \frac{F\epsilon(1 + F\epsilon G)}{1 + F^2\epsilon^2 G}, \quad b = -\epsilon FGa$$

and G is a signal to noise ratio $G = E[s^2]/E[n_1^2]$. Note that when $\epsilon = 0$, then $a = 1$ and $b = 0$, as it should.

1.10 Consider a special case of Example 1.4.3 defined by $c_n = 1$, so that $y_n = x + v_n$, $n = 1, 2, \ldots, M$. This represents the noisy measurement of a constant x. By comparing the corresponding mean-square estimation errors $E[e^2]$, show that the optimal estimate of x given in Eq. (1.4.9) is indeed better than the straight average estimate:

$$\hat{x}_{\text{av}} = \frac{y_1 + y_2 + \cdots + y_M}{M}$$

1.11 *Recursive Estimation.* Consider the subspace $Y_n = \{y_1, y_2, \ldots, y_n\}$ for $n = 1, 2, \ldots, M$, as defined in Sec. 1.6. Eq. (1.6.18) defines the estimate $\hat{\mathbf{x}}$ of a random vector $\mathbf{x}$ based on the largest one of these subspaces, namely, Y_M.

(a) Show that this estimate can also be generated recursively as follows:

$$\hat{\mathbf{x}}_n = \hat{\mathbf{x}}_{n-1} + \mathbf{G}_n(y_n - \hat{y}_{n/n-1})$$

for $n = 1, 2, \ldots, M$, and initialized by $\hat{\mathbf{x}}_0 = 0$ and $\hat{y}_{1/0} = 0$, where $\hat{\mathbf{x}}_n$ denotes the best estimate of $\mathbf{x}$ based on the subspace Y_n and $\mathbf{G}_n$ is a gain coefficient given by $\mathbf{G}_n = E[\mathbf{x}\epsilon_n]E[\epsilon_n\epsilon_n]^{-1}$. (*Hint:* Note $\hat{\mathbf{x}}_n = \sum_{i=1}^{n} E[\mathbf{x}\epsilon_i]E[\epsilon_i\epsilon_i]^{-1}\epsilon_i$.)

(b) Show that the innovations $\epsilon_n = y_n - \hat{y}_{n/n-1}$ is orthogonal to $\hat{\mathbf{x}}_{n-1}$, that is, show that $E[\hat{\mathbf{x}}_{n-1}\epsilon_n] = 0$ for $n = 1, 2, \ldots, M$.

(c) Let $\mathbf{e}_n = \mathbf{x} - \hat{\mathbf{x}}_n$ be the corresponding estimation error of $\mathbf{x}$ with respect to the subspace Y_n. Using Eq. (1.4.4), show that its covariance matrix can be expressed in the ϵ-basis as follows

$$R_{e_ne_n} = R_{xx} - \sum_{i=1}^{n} E[\mathbf{x}\epsilon_i]E[\epsilon_i\epsilon_i]^{-1}E[\epsilon_i\mathbf{x}^T]$$

(d) The above recursive construction represents a successive improvement of the estimate of $\mathbf{x}$, as more and more y_ns are taken into account; that is, as the subspaces Y_n are successively enlarged. Verify that $\hat{\mathbf{x}}_n$ is indeed a better estimate than $\hat{\mathbf{x}}_{n-1}$ by showing that the mean-square estimation error $R_{e_ne_n}$ is smaller than the mean-square error $R_{e_{n-1}e_{n-1}}$. This is a very intuitive result; the more information we use the better the estimate.

Such recursive updating schemes are the essence of Kalman filtering. In that context, $\mathbf{G}_n$ is referred to as the "Kalman gain."

1.12 The recursive updating procedure given in Problem 1.11 is useful only if the gain coefficient $\mathbf{G}_n$ can be computed at each iteration n. For that, a knowledge of the relationship between $\mathbf{x}$ and y_n is required. Consider the case of Example 1.4.3 where $y_n = c_n x + v_n$; define the vectors

$$\mathbf{c}_n = [c_1, c_2, \ldots, c_n]^T, \quad \mathbf{y}_n = [y_1, y_2, \ldots, y_n]^T, \quad \text{for} \quad n = 1, 2, \ldots, M$$

and let $\hat{x}_n$ and $e_n = x - \hat{x}_n$ be the estimate of x on the basis of Y_n and the corresponding estimation error.

(a) Using Eq. (1.4.9), show that

$$\hat{x}_n = \frac{1}{1+\mathbf{c}_n^T\mathbf{c}_n}\mathbf{c}_n^T\mathbf{y}_n \quad \text{and} \quad E[e_n^2] = E[xe_n] = \frac{1}{1+\mathbf{c}_n^T\mathbf{c}_n}$$

(b) Using Eq. (1.6.19), compute $\hat{y}_{n/n-1}$ and show that it may be expressed in the form

$$\hat{y}_{n/n-1} = c_n\hat{x}_{n-1} = \frac{c_n}{1+\mathbf{c}_{n-1}^T\mathbf{c}_{n-1}}\mathbf{c}_{n-1}^T\mathbf{y}_{n-1}$$

(c) Let $e_{n-1} = x - \hat{x}_{n-1}$ be the estimation error based on Y_{n-1}. Writing

$$\epsilon_n = y_n - \hat{y}_{n/n-1} = (c_n x + v_n) - c_n\hat{x}_{n-1} = c_n e_{n-1} + v_n$$

show that

$$E[\epsilon_n\epsilon_n] = (1+\mathbf{c}_n^T\mathbf{c}_n)\,(1+\mathbf{c}_{n-1}^T\mathbf{c}_{n-1})^{-1}$$

$$E[x\epsilon_n] = c_n(1+\mathbf{c}_{n-1}^T\mathbf{c}_{n-1})^{-1}$$

(d) Show that the estimate $\hat{x}_n$ of x can be computed recursively by

$$\hat{x}_n = \hat{x}_{n-1} + G_n(y_n - \hat{y}_{n/n-1}), \quad \text{where} \quad G_n = c_n(1+\mathbf{c}_n^T\mathbf{c}_n)^{-1}$$

1.13 Rederive the recursive updating equation given in Problem 1.12(d), without any reference to innovations or projections, by simply manipulating Eq. (1.4.9) algebraically, and writing it in recursive form.

1.14 *Computer Experiment.* A three-component random vector $\mathbf{y}$ has autocorrelation matrix

$$R = E[\mathbf{y}\mathbf{y}^T] = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 6 & 14 \\ 3 & 14 & 42 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

Carry out the Gram-Schmidt orthogonalization procedure to determine the innovations representation $\mathbf{y} = B\boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, \epsilon_3]^T$ is a vector of uncorrelated components. The vector $\mathbf{y}$ can be simulated by generating a zero-mean gaussian vector of uncorrelated components $\boldsymbol{\epsilon}$ of the appropriate variances and constructing $\mathbf{y} = B\boldsymbol{\epsilon}$. Generate $N = 50$ such vectors $\mathbf{y}_n$, $n = 1, 2, \ldots, N$ and compute the corresponding sample covariance matrix $\hat{R}$ given by Eq. (1.6.21). Compare it with the theoretical R. Is $\hat{R}$ consistent with the standard deviation intervals (1.6.23)? Repeat for $N = 100$.

1.15 The Gram-Schmidt orthogonalization procedure for a subspace $Y = \{y_1, y_2, \ldots, y_M\}$ is initialized at the leftmost random variable y_1 by $\epsilon_1 = y_1$ and progresses to the right by successively orthogonalizing y_2, y_3, and so on. It results in the lower triangular representation $\mathbf{y} = B\boldsymbol{\epsilon}$. The procedure can just as well be started at the rightmost variable y_M and proceed backwards as follows:

$$\eta_M = y_M$$

$$\eta_{M-1} = y_{M-1} - (\text{projection of } y_{M-1} \text{ on } \eta_M)$$

$$\eta_{M-2} = y_{M-2} - (\text{projection of } y_{M-2} \text{ on } \{\eta_M, \eta_{M-1}\})$$

and so on. Show that the resulting uncorrelated vector $\boldsymbol{\eta} = [\eta_1, \eta_2, \ldots, \eta_M]^T$ is related to $\mathbf{y} = [y_1, y_2, \ldots, y_M]^T$ by a linear transformation

$$\mathbf{y} = U\boldsymbol{\eta}$$

where U is a unit *upper-triangular* matrix. Show also that this corresponds to a UL (rather than LU) Cholesky factorization of the covariance matrix R_{yy}.

1.16 Since "orthogonal" means "uncorrelated," the Gram-Schmidt orthogonalization procedure can also be understood as a correlation canceling operation. Explain how Eq. (1.6.20) may be thought of as a special case of the correlation canceler defined by Eqs. (1.4.1) and (1.4.2). What are $\mathbf{x}, \mathbf{y}, \mathbf{e}$, and H, in this case? Draw the correlation canceler diagram of Fig. 1.4.1 as it applies here, showing explicitly the components of all the vectors.

1.17 Using Eq. (1.7.11), show that the vector of coefficients $[a_{n1}, a_{n2}, \ldots, a_{nn}]^T$ can be expressed explicitly in terms of the $\mathbf{y}$-basis as follows:

$$\begin{bmatrix} a_{n1} \\ a_{n2} \\ \vdots \\ a_{nn} \end{bmatrix} = -E[\mathbf{y}_{n-1}\mathbf{y}_{n-1}^T]^{-1}E[y_n\mathbf{y}_{n-1}], \quad \text{where} \quad \mathbf{y}_{n-1} = \begin{bmatrix} y_{n-1} \\ y_{n-2} \\ \vdots \\ y_0 \end{bmatrix}$$

1.18 Show that the mean-square estimation error of y_n on the basis of Y_{n-1}—that is, $E[\epsilon_n^2]$, where $\epsilon_n = y_n - \hat{y}_{n/n-1}$—can be expressed as

$$E[\epsilon_n^2] = E[\epsilon_n y_n] = E[y_n^2] - E[y_n\mathbf{y}_{n-1}^T]E[\mathbf{y}_{n-1}\mathbf{y}_{n-1}^T]^{-1}E[y_n\mathbf{y}_{n-1}]$$

1.19 Let $\mathbf{a}_n = [1, a_{n1}, a_{n2}, \ldots, a_{nn}]^T$ for $n = 1, 2, \ldots, M$. Show that the results of the last two problems can be combined into one enlarged matrix equation

$$E[\mathbf{y}_n\mathbf{y}_n^T]\mathbf{a}_n = E[\epsilon_n^2]\mathbf{u}_n$$

where $\mathbf{u}_n$ is the unit-vector $\mathbf{u}_n = [1, 0, 0, \ldots, 0]^T$ consisting of one followed by n zeros, and $\mathbf{y}_n = [y_n, y_{n-1}, \ldots, y_1, y_0]^T = [y_n, \mathbf{y}_{n-1}^T]^T$.

1.20 The quantity $\hat{y}_{n/n-1}$ of Eq. (1.6.19) is the best estimate of y_n based on all the previous ys, namely, $Y_{n-1} = \{y_0, y_1, \ldots, y_{n-1}\}$. This can be understood in three ways: First, in terms of the orthogonal projection theorem as we demonstrated in the text. Second, in terms of the correlation canceler interpretation as suggested in Problem 1.16. And third, it may be proved directly as follows. Let $\hat{y}_{n/n-1}$ be given as a linear combination of the previous ys as in Eq. (1.7.11); the coefficients $[a_{n1}, a_{n2}, \ldots, a_{nn}]^T$ are to be chosen optimally to minimize the estimation error ϵ_n given by Eq. (1.7.10) in the mean-square sense. In terms of the notation of Problem 1.19, Eq. (1.7.10) and the mean-square error $E[\epsilon_n^2]$ can be written in the compact vectorial form

$$\epsilon_n = \mathbf{a}_n^T\mathbf{y}_n, \quad \mathcal{E}(\mathbf{a}_n) = E[\epsilon_n^2] = \mathbf{a}_n^T E[\mathbf{y}_n\mathbf{y}_n^T]\mathbf{a}_n$$

The quantity $\mathcal{E}(\mathbf{a}_n)$ is to be minimized with respect to $\mathbf{a}_n$. The minimization must be subject to the constraint that the first entry of the vector $\mathbf{a}_n$ be unity. This constraint can be expressed in vector form as

$$\mathbf{a}_n^T\mathbf{u}_n = 1$$

where $\mathbf{u}_n$ is the unit vector defined in Problem 1.19. Incorporate this constraint with a Lagrange multiplier λ and minimize the performance index

$$\mathcal{E}(\mathbf{a}_n) = \mathbf{a}_n^T E[\mathbf{y}_n\mathbf{y}_n^T]\mathbf{a}_n + \lambda(1 - \mathbf{a}_n^T\mathbf{u}_n)$$

with respect to $\mathbf{a}_n$, then fix λ by enforcing the constraint, and finally show that the resulting solution of the minimization problem is identical to that given in Problem 1.19.

1.21 Show that the normal equations (1.8.12) can also be obtained by minimizing the performance indices (1.8.10) with respect to $\mathbf{a}$ and $\mathbf{b}$, subject to the constraints that the first element of $\mathbf{a}$ and the last element of $\mathbf{b}$ be unity. (*Hint:* These constraints are expressible in the form $\mathbf{u}^T\mathbf{a} = 1$ and $\mathbf{v}^T\mathbf{b} = 1$.)

1.22 Using Eq. (1.8.16), show that E_b can be expressed as the ratio of the two determinants $E_b = \det R / \det \bar{R}$.

1.23 Show Eqs. (1.8.28) and (1.8.35).

1.24 A random signal $x(n)$ is defined as a linear function of time by

$$x(n) = an + b$$

where a and b are independent zero-mean gaussian random variables of variances σ_a^2 and σ_b^2, respectively.

(a) Compute $E[x(n)^2]$.

(b) Is $x(n)$ a stationary process? Is it ergodic? Explain.

(c) For each fixed n, compute the probability density $p(x(n))$.

(d) For each fixed n and m $(n \neq m)$, compute the conditional probability density function $p(x(n)|x(m))$ of $x(n)$ given $x(m)$. (*Hint:* $x(n) - x(m) = (n-m)b$.)

1.25 Compute the sample autocorrelation of the sequences

(a) $y_n = 1$, for $0 \leq n \leq 10$.

(b) $y_n = (-1)^n$, for $0 \leq n \leq 10$.

in two ways: First in the time domain, using Eq. (1.11.1), and then in the z-domain, using Eq. (1.11.3) and computing its inverse z-transform.

1.26 *FFT Computation of Autocorrelations.* In many applications, a fast computation of sample autocorrelations or cross-correlations is required, as in the matched filtering operations in radar data processors. A fast way to compute the sample autocorrelation $\hat{R}_{yy}(k)$ of a length-N data segment $\mathbf{y} = [y_0, y_1, \ldots, y_{N-1}]^T$ is based on Eq. (1.11.5) which can be computed using FFTs. Performing an inverse FFT on Eq. (1.11.5), we find the computationally efficient formula

$$\hat{R}_{yy}(k) = \frac{1}{N}\text{IFFT}\left[\,|\,\text{FFT}(\mathbf{y})\,|^2\right] \tag{P.1}$$

To avoid wrap-around errors introduced by the IFFT, the length N' of the FFT must be selected to be greater than the length of the function $\hat{R}_{yy}(k)$. Since $\hat{R}_{yy}(k)$ is double-sided with an extent $-(N-1) \leq k \leq (N-1)$, it will have length equal to $2N-1$. Thus, we must select $N' \geq 2N-1$. To see the wrap-around effects, consider the length-4 signal $\mathbf{y} = [1, 2, 2, 1]^T$.

(a) Compute $\hat{R}_{yy}(k)$ using the time-domain definition.

(b) Compute $\hat{R}_{yy}(k)$ according to Eq. (P.1) using 4-point FFTs.

(c) Repeat using 8-point FFTs.

1.27 *Computer Experiment.*

(a) Generate 1000 samples $x(n)$, $n = 0, 1, \ldots, 999$, of a zero-mean, unit-variance, white gaussian noise sequence.

(b) Compute and plot the first 100 lags of its sample autocorrelation, that is, $\hat{R}_{yy}(k)$, for $k = 0, 1, \ldots, 99$. Does $\hat{R}_{yy}(k)$ look like a delta function $\delta(k)$?

(c) Generate 10 different realizations of the length-1000 sequence $x(n)$, and compute 100 lags of the corresponding sample autocorrelations. Define an average autocorrelation by

$$\hat{R}(k) = \frac{1}{10}\sum_{i=1}^{10}\hat{R}_i(k)\,, \quad k = 0, 1, \ldots, 99,$$

where $\hat{R}_i(k)$ is the sample autocorrelation of the ith realization of $x(n)$. Plot $\hat{R}(k)$ versus k. Do you notice any improvement?

1.28 A 500-millisecond record of a stationary random signal is sampled at a rate of 2 kHz and the resulting N samples are recorded for further processing. What is N? The record of N samples is then divided into K contiguous segments, each of length M, so that $M = N/K$. The periodograms from each segment are computed and averaged together to obtain an estimate of the power spectrum of the signal. A frequency resolution of $\Delta f = 20$ Hz is required. What is the shortest length M that will guarantee such resolution? (Larger Ms will have better resolution than required but will result in a poorer power spectrum estimate because K will be smaller.) What is K in this case?

1.29 A random signal y_n is generated by sending unit-variance zero-mean white noise ϵ_n through the filters defined by the following difference equations:

1. $y_n = -0.9y_{n-1} + \epsilon_n$
2. $y_n = 0.9y_{n-1} + \epsilon_n + \epsilon_{n-1}$
3. $y_n = \epsilon_n + 2\epsilon_{n-1} + \epsilon_{n-2}$
4. $y_n = -0.81y_{n-2} + \epsilon_n$
5. $y_n = 0.1y_{n-1} + 0.72y_{n-2} + \epsilon_n - 2\epsilon_{n-1} + \epsilon_{n-2}$

(a) For each case, determine the transfer function $B(z)$ of the filter and draw its canonical implementation form, identify the set of model parameters, and decide whether the model is ARMA, MA, or AR.

(b) Write explicitly the power spectrum $S_{yy}(\omega)$ using Eq. (1.13.6).

(c) Based on the pole/zero pattern of the filter $B(z)$, draw a rough sketch of the power spectrum $S_{yy}(\omega)$ for each case.

1.30 *Computer Experiment.*

Two different realizations of a stationary random signal $y(n)$, $n = 0, 1, \ldots, 19$ are given. It is known that this signal has been generated by a model of the form

$$y(n) = ay(n-1) + \epsilon(n)$$

where $\epsilon(n)$ is gaussian zero-mean white noise of variance σ_ϵ^2.

(a) Estimate the model parameters a and σ_ϵ^2 using the maximum likelihood criterion for both realizations. (The exact values were $a = 0.95$ and $\sigma_\epsilon^2 = 1$.)

(b) Repeat using the Yule-Walker method.

This type of problem might, for example, arise in speech processing where $y(n)$ might represent a short segment of sampled unvoiced speech from which the filter parameters (model parameters) are to be extracted and stored for future regeneration of that segment. A realistic speech model would of course require a higher-order filter, typically, of order 10 to 15.

n	$y(n)$	$y(n)$
0	3.848	5.431
1	3.025	5.550
2	5.055	4.873
3	4.976	5.122
4	6.599	5.722
5	6.217	5.860
6	6.572	6.133
7	6.388	5.628
8	6.500	6.479
9	5.564	4.321
10	5.683	5.181
11	5.255	4.279
12	4.523	5.469
13	3.952	5.087
14	3.668	3.819
15	3.668	2.968
16	3.602	2.751
17	1.945	3.306
18	2.420	3.103
19	2.104	3.694

1.31 *Computer Experiment.*

(a) Using the Yule-Walker estimates $\{\hat{a}, \hat{\sigma}_\epsilon^2\}$ of the model parameters extracted from the first realization of $y(n)$ given in Problem 1.30, make a plot of the estimate of the power spectrum following Eq. (1.13.6), that is,

$$\hat{S}_{yy}(\omega) = \frac{\hat{\sigma}_\epsilon^2}{|1 - \hat{a}e^{-j\omega}|^2}$$

versus frequency ω in the interval $0 \le \omega \le \pi$.

(b) Also, plot the true power spectrum

$$S_{yy}(\omega) = \frac{\sigma_\epsilon^2}{|1 - ae^{-j\omega}|^2}$$

defined by the true model parameters $\{a, \sigma_\epsilon^2\} = \{0.95, 1\}$.

(c) Using the given data values $y(n)$ for the first realization, compute and plot the corresponding periodogram spectrum of Eq. (1.11.5). Preferably, plot all three spectra on the same graph. Compute the spectra at 100 or 200 equally spaced frequency points in the interval $[0, \pi]$. Plot all spectra in decibels.

(d) Repeat parts (a) through (c) using the second realization of $y(n)$.

Better agreement between estimated and true spectra can be obtained using Burg's analysis procedure instead of the Yule-Walker method. Burg's method performs remarkably well on the basis of very short data records. The Yule-Walker method also performs well but it requires somewhat longer records. These methods will be compared in Chap. 14.

1.32 In addition to the asymptotic results (1.16.4) for the model parameters, we will show in Chap. 14 that the estimates of filter parameter and the input variance are asymptotically uncorrelated, $E[\Delta a \, \Delta\sigma_\epsilon^2] = 0$. Using this result and Eq. (1.16.4), show that the variance of the spectrum estimate is given asymptotically by

$$E[\Delta S(\omega) \, \Delta S(\omega)] = \frac{2S(\omega)^2}{N}\left[1 + \frac{2(1-a^2)(\cos\omega - a)^2}{(1 - 2a\cos\omega + a^2)^2}\right]$$

where $\Delta S(\omega) = \hat{S}(\omega) - S(\omega)$, with the theoretical and estimated spectra given in terms of the theoretical and estimated model parameters by

$$S(\omega) = \frac{\sigma_\epsilon^2}{|1 - ae^{-j\omega}|^2}, \quad \hat{S}(\omega) = \frac{\hat{\sigma}_\epsilon^2}{|1 - \hat{a}e^{-j\omega}|^2}$$

1.33 For any positive semi-definite matrix B show the inequality $\mathrm{tr}(B - I - \ln B) \ge 0$ with equality achieved for $B = I$. Using this property, show the inequality $f(R) \ge f(\hat{R})$, where $f(R) = \mathrm{tr}(\ln R + R^{-1}\hat{R})$. This implies the maximum likelihood property of $\hat{R}$, discussed in Sec. 1.18.

1.34 Show the following three matrix properties used in Sec. 1.18:

$$\ln(\det R) = \mathrm{tr}(\ln R), \quad d\,\mathrm{tr}(\ln R) = \mathrm{tr}(R^{-1}dR), \quad dR^{-1} = -R^{-1}\,dR\,R^{-1}$$

(*Hints:* for the first two, use the eigenvalue decomposition of R; for the third, start with $R^{-1}R = I$.)

1.35 Let $x(n)$ be a zero-mean white-noise sequence of unit variance. For each of the following filters compute the output autocorrelation $R_{yy}(k)$ for all k, using z-transforms:

1. $y(n) = x(n) - x(n-1)$
2. $y(n) = x(n) - 2x(n-1) + x(n-2)$
3. $y(n) = -0.5y(n-1) + x(n)$
4. $y(n) = 0.25y(n-2) + x(n)$

Also, sketch the output power spectrum $S_{yy}(\omega)$ versus frequency ω.

1.36 Let y_n be the output of a (stable and causal) filter $H(z)$ driven by the signal x_n, and let w_n be another unrelated signal. Assume all signals are stationary random signals. Show the following relationships between power spectral densities:

(a) $S_{yw}(z) = H(z)S_{xw}(z)$

(b) $S_{wy}(z) = S_{wx}(z)H(z^{-1})$

1.37 A stationary random signal y_n is sent through a finite filter $A(z) = a_0 + a_1 z^{-1} + \cdots + a_M z^{-M}$ to obtain the output signal e_n:

$$y_n \longrightarrow \boxed{A(z)} \longrightarrow e_n \qquad e_n = \sum_{m=0}^{M} a_m y_{n-m}$$

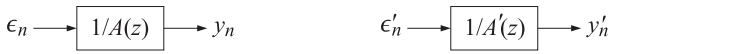
Show that the average power of the output e_n can be expressed in the two alternative forms:

$$E[e_n^2] = \int_{-\pi}^{\pi} S_{yy}(\omega)\,|A(\omega)|^2 \frac{d\omega}{2\pi} = \mathbf{a}^T R_{yy}\mathbf{a}$$

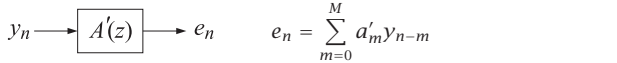
where $\mathbf{a} = [a_0, a_1, \ldots, a_M]^T$ and R_{yy} is the $(M+1)\times(M+1)$ autocorrelation matrix of y_n having matrix elements $R_{yy}(i,j) = E[y_i y_j] = R_{yy}(i-j)$.

1.38 Consider the two autoregressive random signals y_n and y'_n generated by the two signal models:

$$A(z) = 1 + a_1 z^{-1} + \cdots + a_M z^{-M} \quad \text{and} \quad A'(z) = 1 + a'_1 z^{-1} + \cdots + a'_M z^{-M}$$



(a) Suppose y_n is filtered through the analysis filter $A'(z)$ of y'_n producing the output signal e_n; that is,



If y_n were to be filtered through its own analysis filter $A(z)$, it would produce the innovations sequence ϵ_n. Show that the average power of e_n compared to the average power of ϵ_n is given by

$$\frac{\sigma_e^2}{\sigma_\epsilon^2} = \frac{\mathbf{a}'^T R_{yy} \mathbf{a}'}{\mathbf{a}^T R_{yy} \mathbf{a}} = \int_{-\pi}^{\pi} \left| \frac{A'(\omega)}{A(\omega)} \right|^2 \frac{d\omega}{2\pi} = \left\| \frac{A'}{A} \right\|^2$$

where $\mathbf{a}, \mathbf{a}'$ and R_{yy} have the same meaning as in Problem 1.37. This ratio can be taken as a measure of *similarity* between the two signal models. The log of this ratio is Itakura's *LPC distance measure* used in speech recognition.

(b) Alternatively, show that if y'_n were to be filtered through y_n's analysis filter $A(z)$ resulting in $e'_n = \sum_{m=0}^{M} a_m y'_{n-m}$, then

$$\frac{\sigma_{e'}^2}{\sigma_{\epsilon'}^2} = \frac{\mathbf{a}^T R'_{yy} \mathbf{a}}{\mathbf{a}'^T R'_{yy} \mathbf{a}'} = \int_{-\pi}^{\pi} \left| \frac{A(\omega)}{A'(\omega)} \right|^2 \frac{d\omega}{2\pi} = \left\| \frac{A}{A'} \right\|^2$$

1.39 The autocorrelation function of a complex-valued signal is defined by

$$R_{yy}(k) = E[y_{n+k} y_n^*]$$

(a) Show that stationarity implies $R_{yy}(-k) = R_{yy}(k)^*$.

(b) If y_n is filtered through a (possibly complex-valued) filter $A(z) = a_0 + a_1 z^{-1} + \cdots + a_M z^{-M}$, show that the average power of the output signal e_n can be expressed as

$$E[e_n^* e_n] = \mathbf{a}^\dagger R_{yy} \mathbf{a}$$

where $\mathbf{a}^\dagger$ denotes the hermitian conjugate of $\mathbf{a}$ and R_{yy} has matrix elements

$$R_{yy}(i,j) = R_{yy}(i-j)$$

1.40 (a) Let $y_n = A_1 \exp[j(\omega_1 n + \phi_1)]$ be a complex sinusoid of amplitude A_1 and frequency ω_1. The randomness of y_n arises only from the phase ϕ_1 which is assumed to be a random variable uniformly distributed over the interval $0 \le \phi_1 \le 2\pi$. Show that the autocorrelation function of y_n is

$$R_{yy}(k) = |A_1|^2 \exp(j\omega_1 k)$$

(b) Let y_n be the sum of two sinusoids

$$y_n = A_1 \exp[j(\omega_1 n + \phi_1)] + A_2 \exp[j(\omega_2 n + \phi_2)]$$

with uniformly distributed random phases ϕ_1 and ϕ_2 which are also assumed to be independent of each other. Show that the autocorrelation function of y_n is

$$R_{yy}(k) = |A_1|^2 \exp(j\omega_1 k) + |A_2|^2 \exp(j\omega_2 k)$$

1.41 *Sinusoids in Noise.* Suppose y_n is the sum of L complex sinusoids with random phases, in the presence of uncorrelated noise:

$$y_n = v_n + \sum_{i=1}^{L} A_i \exp[j(\omega_i n + \phi_i)]$$

where ϕ_i, $i = 1, 2, \ldots, L$ are uniformly distributed random phases which are assumed to be mutually independent, and v_n is zero-mean white noise of variance σ_v^2. Also, assume that v_n is independent of ϕ_i.

(a) Show that $E[e^{j\phi_i} e^{-j\phi_k}] = \delta_{ik}$, for $i, k = 1, 2, \ldots, L$.

(b) Show that the autocorrelation of y_n is

$$R_{yy}(k) = \sigma_v^2 \delta(k) + \sum_{i=1}^{L} |A_i|^2 \exp(j\omega_i k)$$

(c) Suppose y_n is filtered through a filter $A(z) = a_0 + a_1 z^{-1} + \cdots + a_M z^{-M}$ of order M, producing the output signal e_n. Show that the average output power is expressible as

$$\mathcal{E} = E[e_n^* e_n] = \mathbf{a}^\dagger R_{yy} \mathbf{a} = \sigma_v^2 \mathbf{a}^\dagger \mathbf{a} + \sum_{i=1}^{L} |A_i|^2 \, |A(\omega_i)|^2$$

where $\mathbf{a}, \mathbf{a}^\dagger, R_{yy}$ have the same meaning as in Problem 1.39, and $A(\omega_i)$ is the frequency response of the filter evaluated at the sinusoid frequency ω_i, that is,

$$A(\omega_i) = \sum_{m=0}^{M} a_m e^{-j\omega_i m}, \quad i = 1, 2, \ldots, M$$

(d) If the noise v_n is correlated with autocorrelation $Q(k)$, so that $E[v_{n+k} v_n^*] = Q(k)$, show that in this case

$$\mathcal{E} = E[e_n^* e_n] = \mathbf{a}^\dagger R_{yy} \mathbf{a} = \mathbf{a}^\dagger Q \mathbf{a} + \sum_{i=1}^{L} |A_i|^2 \, |A(\omega_i)|^2$$

where Q is the noise covariance matrix, $Q(i,j) = Q(i-j)$.

1.42 A filter is defined by $y(n) = -0.64 y(n-2) + 0.36 x(n)$.

(a) Suppose the input is zero-mean, unit-variance, white noise. Compute the output spectral density $S_{yy}(z)$ and power spectrum $S_{yy}(\omega)$ and plot it roughly versus frequency.

(b) Compute the output autocorrelation $R_{yy}(k)$ for all lags k.

(c) Compute the noise reduction ratio of this filter.

(d) What signal $s(n)$ can pass through this filter and remain entirely unaffected (at least in the steady-state regime)?

(e) How can the filter coefficients be changed so that (i) the noise reduction capability of the filter is improved, while at the same time (ii) the above signal $s(n)$ still goes through unchanged? Explain any tradeoffs.

1.43 *Computer Experiment.* (a) Generate 1000 samples of a zero-mean, unit-variance, white gaussian noise sequence $x(n)$, $n = 0, 1, \ldots, 999$, and filter them through the filter defined by the difference equation:

$$y(n) = ay(n-1) + (1-a)x(n)$$

with $a = 0.95$. To avoid the transient effects introduced by the filter, discard the first 900 output samples and save the last 100 samples of $y(n)$. Compute the sample autocorrelation of $y(n)$ from this length-100 block of samples.

(b) Determine the theoretical autocorrelation $R_{yy}(k)$, and on the same graph, plot the theoretical and sample autocorrelations versus k. Do they agree?

1.44 Prove Eq. (1.19.6).

1.45 Using Eq. (1.19.10), show Eqs. (1.19.12) and (1.19.13).

1.46 A random signal y_n has autocorrelation function

$$R_{yy}(k) = (0.5)^{|k|}, \quad \text{for all } k$$

Find a random signal generator model for y_n.

1.47 Repeat Problem 1.46 when

$$R_{yy}(k) = (0.5)^{|k|} + (-0.5)^{|k|}, \quad \text{for all } k$$

1.48 The autocorrelation function of a stationary random signal $y(n)$ is

$$R_{yy}(k) = \frac{1-R^2}{1+R^2} R^{|k|} \cos(\pi k/2), \quad \text{for all } k, \quad \text{where } 0 < R < 1$$

(a) Compute the power spectrum $S_{yy}(\omega)$ of $y(n)$ and sketch it versus frequency for various values of R.

(b) Find the signal generator filter for $y(n)$ and determine its difference equation and its poles and zeros.

1.49 A stationary random signal y_n has a rational power spectral density given by

$$S_{yy}(z) = \frac{2.18 - 0.6(z + z^{-1})}{1.25 - 0.5(z + z^{-1})}$$

Determine the signal model filter $B(z)$ and the parameter σ_ϵ^2. Write the difference equation generating y_n.

1.50 Let $y_n = cx_n + v_n$. It is given that

$$S_{xx}(z) = \frac{Q}{(1-az^{-1})(1-az)}, \quad S_{vv}(z) = R, \quad S_{xv}(z) = 0$$

where a, c, Q, R are known constants (assume $|a| < 1$) for the stability of x_n.)

(a) Show that the filter model for y_n is of the form

$$B(z) = \frac{1 - fz^{-1}}{1 - az^{-1}}$$

where f has magnitude less than one and is the solution of the algebraic quadratic equation

$$aR(1 + f^2) = [c^2 Q + R(1 + a^2)]f$$

and show that the other solution has magnitude greater than one.

(b) Show that f can alternatively be expressed as

$$f = \frac{Ra}{R + c^2 P}$$

where P is the *positive* solution of the quadratic equation

$$Q = P - \frac{PRa^2}{R + c^2 P}$$

known as the *algebraic Riccati* equation. Show that the other solution is negative. Show that the positivity of P is essential to guarantee that f has magnitude less than one.

(c) Show that the scale factor σ_ϵ^2 that appears in the spectral factorization (1.20.1) can also be expressed in terms of P as

$$\sigma_\epsilon^2 = R + c^2 P$$

The above method of solution of the spectral factorization problem by reducing it to the solution of an algebraic Riccati equation is quite general and can be extended to the multichannel case.

1.51 Consider a stable (but not necessarily causal) sequence b_n, $-\infty < n < \infty$ with a z-transform $B(z)$

$$B(z) = \sum_{n=-\infty}^{\infty} b_n z^{-n}$$

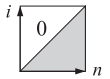
Define an infinite Toeplitz matrix B by

$$B_{ni} = b_{n-i}, \quad \text{for } -\infty < n, i < \infty$$

This establishes a correspondence between stable z-transforms or stable sequences and infinite Toeplitz matrices.

(a) Show that if the sequence b_n is causal, then B is lower triangular, as shown here

In the literature of integral operators and kernels, such matrices are rotated by 90° degrees as shown:



so that the n axis is the horizontal axis. For this reason, in that context they are called "right Volterra kernel," or "causal kernels."

(b) Show that the transposed B^T corresponds to the reflected (about the origin) sequence b_{-n} and to the z-transform $B(z^{-1})$.

(c) Show that the convolution of two sequences a_n and b_n

$$c_n = a_n * b_n \quad \text{or} \quad C(z) = A(z)B(z)$$

corresponds to the commutative matrix product

$$C = AB = BA$$

1.52 Prove Eq. (1.21.2) for any M.