CHAPTER 1

Propaedeutic

Read.Me: *If you are someone who never reads Chapter 1, please at least read Sections 1.0.2 and 1.0.3 before proceeding*!

1.0 Preamble

1.0.1 The Purpose of Chapter 1

If the reader learns nothing more from this book, it is a safe bet that he or she will learn a new word. A *propaedeutic*[1] is a "preliminary body of knowledge and rules necessary for the study of some art or science" (Barnhart, 1964). This chapter is just that—a propaedeutic for the study of speech processing focusing primarily on two broad areas, digital signal processing (DSP) and stochastic processes, and also on some necessary topics from the fields of statistical pattern recognition and information theory.

The reader of this book is assumed to have a sound background in the first two of these areas, typical of an entry level graduate course in each field. It is not our purpose to comprehensively teach DSP and random processes, and the brief presentation here is not intended to provide an adequate background. There are many fine textbooks to which the reader might refer to review and reinforce prerequisite topics for these subjects. We list a considerable number of widely used books in Appendices 1.A and 1.B.

What, then, is the point of our propaedeutic? The remainder of this chapter is divided into four main sections plus one small section, and the tutorial goals are somewhat different in each. Let us first consider the two main sections on DSP and stochastic processes. In the authors' experience, the speech processing student is somewhat more comfortable with "deterministic" DSP topics than with random processes. What we will do in Section 1.1, which focuses on DSP, therefore, is highlight some of the key concepts which will play central roles in our speech processing work. Where the material seems unfamiliar, the reader is urged to seek help in

[1]Pronounced "prō′-pa-doo′-tic."

one or more of the DSP textbooks cited in Appendix 1.A. Our main objective is to briefly outline the essential DSP topics with a particular interest defining notation that will be used consistently throughout the book. A second objective is to cover a few subtler concepts that will be important in this book, and that might have been missed in the reader's first exposure to DSP.

The goals of Section 1.2 on random processes are somewhat different. We will introduce some fundamental concepts with a bit more formality, uniformity, and detail than the DSP material. This treatment might at first seem unnecessarily detailed for a textbook on speech processing. We do so, however, for several reasons. First, a clear understanding of stochastic process concepts, which are so essential in speech processing, depends strongly on an understanding of the basic probability formalisms. Second, many engineering courses rely heavily on stochastic processes and not so much on the underlying probability concepts, so that the probability concepts become "rusty." Emerging technologies in speech processing depend on the basic probability theory and some review of these ideas could prove useful. Third, it is true that the mastery of any subject requires several "passes" through the material, but engineers often find this especially true of the field of probability and random processes.

The third and fourth major divisions of this chapter, Sections 1.3 and 1.4, treat a few topics which are used in the vast fields of statistical pattern recognition and information theory. In fact, we have included some topics in Section 1.3 which are perhaps more general than "pattern recognition" methods, but the rubric will suffice. These sections are concerned with basic mathematical tools which will be used frequently, and in diverse ways in our study, beginning in Part IV of the book. There is no assumption that the reader has formal coursework in these topics beyond the normal acquaintance with them that would ordinarily be derived from an engineering education. Therefore, the goal of these sections is to give an adequate description of a few important topics which will be critical to our speech work.

Finally, Section 1.5 briefly reviews the essence and notation of phasors and steady-state analysis of systems described by differential equations. A firm grasp of this material will be necessary in our early work on analog acoustic modeling of the speech production system in Chapter 3.

As indicated above, the need for the subjects in Sections 1.3–1.5 is not immediate, so the reader might wish to scan over these sections, then return to them as needed. More guidance on reading strategy follows.

1.0.2 Please Read This Note on Notation

The principal tool of engineering is applied mathematics. The language of mathematics is abstract symbolism. This book is written with a conviction that careful and consistent notation is a sign of clear under-

standing, and clear understanding is derived by forcing oneself to comprehend and use such notation. Painstaking care has been taken in this book to use information-laden and consistent notation in keeping with this philosophy. When we err with notation, we err on the side of excessive notation which is not always conventional, and not always necessary once the topic has been mastered. Therefore, the reader is invited (with your instructor's permission if you are taking a course!) to shorten or simplify the notation as the need for the "tutorial" notation subsides.

Let us give some examples. We will later use an argument m to keep track of the point in time at which certain features are extracted from a speech signal. This argument is key to understanding the "short-term" nature of the processing of speech. The ith "linear prediction" coefficient computed on a "frame" of speech ending at time m will be denoted $\hat{a}(i;m)$. In the development of an algorithm for computing the coefficients, for example, the index m will not be very germane to the development and the reader might wish to omit it once its significance is clear. Another example comes from the random process theory. Numerous examples of sloppy notation abound in probability theory, a likely reason why many engineers find this subject intractable. For example, something like "$f(x)$" is frequently used to denote the probability density function (pdf) for the random variable $\underline{x}$. There are numerous ways in which this notation can cause misunderstandings and even subtle mathematical traps which can lead to incorrect results. We will be careful in this text to delineate random processes, random variables, and values that may be assumed by a random variable. We will denote a random variable, for example, by underscoring the variable name, for example, $\underline{x}$. The pdf for $\underline{x}$ will be denoted $f_{\underline{x}}(x)$, for example. The reader who has a clear understanding of the underlying concepts might choose to resort to some sloppier form of notation, but the reader who does not will benefit greatly by working to understanding the details of the notation.

1.0.3 For People Who Never Read Chapter 1 (and Those Who Do)

To be entitled to use the word "propaedeutic" at your next social engagement, you must read at least some of this chapter.[2] If for no other reason than to become familiar with the notation, we urge you to at least generally review the topics here before proceeding. However, there is a large amount of material in this chapter, and some people will naturally prefer to review these topics on an "as needed" basis. For that reason, we provide the following guide to the use of Chapter 1.

With a few exceptions, most of the topics in Sections 1.1 and 1.2 will be widely used throughout the book and we recommend their review before proceeding. The one exception is the subsection on "State Space Re-

[2]If you have skipped the first part of this chapter, you will be using the word without even knowing what it means.

alizations" in Section 1.1.6, which will be used in a limited way in Chapters 5 and 12. The topics in Sections 1.3 and 1.4, however, are mostly specialized subjects which will be used in particular aspects of our study, beginning in Part IV of the book. Likewise the topic in Section 1.5 is used in one isolated, but important, body of material in Chapter 3.

These latter topics and the "state space" topic in the earlier section will be "flagged" in Reading Notes at the beginning of relevant chapters, and in other appropriate places in the book.

1.1 Review of DSP Concepts and Notation

1.1.1 "Normalized Time and Frequency"

Throughout the book, we will implicitly use what we might call *normalized time and frequency variables.* By this we mean that a discrete time signal (usually speech), say $s(n)$, will be indexed by integers only.[3] Whereas $s(n)$ invariably represents samples of an analog waveform, say $s_a(t)$, at some sample period, T,

$$s(n) = s_a(nT) = s_a(t)\big|_{t=nT} \qquad n = \ldots, -1, 0, 1, 2, \ldots, \tag{1.1}$$

the integer n indexes the *sample number,* but we have lost the absolute time orientation in the argument. To recover the times at which the samples are taken, we simply need to know T.

To understand the "physical" significance of this mathematical convention, it is sometimes convenient to imagine that we have scaled the real-world time axis by a factor of T prior to taking the samples, as illustrated in Fig. 1.1. "Normalized time," say t', is related to real time as

$$t' = \frac{t}{T} \tag{1.2}$$

and the samples of speech are taken at intervals which are exactly[4] "normalized seconds (norm-sec)." In most cases it is perfectly sufficient to refer to the interval between samples as the "sample period," where the conversion to the real-world interval is obvious. However, on a few occasions we will have more than one sampling process occurring in the same problem (i.e., a resampling of the speech sequence), and in these instances the concept of a "normalized second" is useful to refer to the basic sampling interval on the data.

Of course, the normalization of time renders certain frequency quantities invariant. The sample period in normalized time is always unity, and therefore the sample frequency is always unity [dimensionless, but some-

[3]Note that we have referred to $s(n)$ as "discrete time" rather than "digital." Throughout most of this book, we will ignore any quantization of amplitude.

[4]The reader should note that the normalized time axis is actually dimensionless.

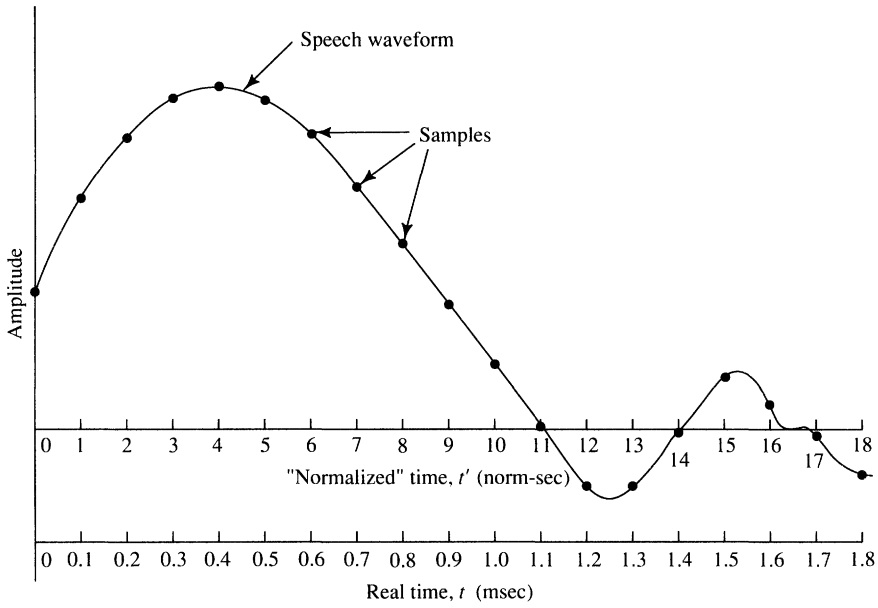


FIGURE 1.1. Segment of a speech waveform used to illustrate the concept of "normalized time." Suppose that samples are to be taken at a rate F_s = 10 kHz so that the sample period is $T = 0.1$ msec. The lower time axis represents real time measured in milliseconds, while the upper represents a normalization of the time axis such that the sample times fall at integers. Normalized time, t', is related to real time, t, as $t' = t/T$. We will on a few occasions refer to the sample period in the scaled case as a "normalized second (norm-sec)."

times "normalized Hertz (norm-Hz)"], and the sample radian frequency is always 2π [dimensionless or "normalized radians per second (norm-rps)"]. Accordingly, the Nyquist frequency is always 0.5 norm-Hz, or π norm-rps. In general, the conversions between "real" frequencies, say F (Hz) and Ω (rps) and their normalized counterparts, say f and ω, are given by

$$f = FT \tag{1.3}$$

$$\omega = \Omega T. \tag{1.4}$$

We can easily verify this by examining a single sinusoid at real frequency Ω,

$$x_a(t) = A \sin(\Omega t + \varphi) = A \sin(\Omega T \frac{t}{T} + \varphi). \tag{1.5}$$

The rightmost term can be regarded as a sinusoid at a different frequency, $\omega = \Omega T$, on a different time axis $t' = t/T$,

$$x'_a(t') = A \sin(\omega t' + \varphi). \tag{1.6}$$

Clearly, we obtain the same samples if we sample $x_a(t)$ at $t = nT$, or $x'_a(t')$ at $t' = n$. A magnitude spectrum is plotted against real and normalized frequencies in Fig. 1.2 to illustrate this concept.

In spite of this rather lengthy explanation of "normalized time and frequency," we do not want to overemphasize this issue. Simply stated, we will find it convenient to index speech waveforms by integers (especially in theoretical developments). This is an accepted convention in DSP. The point of the above is to remind the reader that the resulting "normalized" time and frequency domains are simply related to the "real" time and frequency. When the "normalized" quantities need to be converted to "real" quantities, this is very easily accomplished if the sample frequency or period is known. While using the DSP convention for convenience in many discussions, we will always keep the sampling information close at hand so the "real" quantities are known. To do otherwise would be to deny the physical nature of the process with which we are working. In many instances in which practical systems and applications are being discussed, it will make perfect sense to simply work in terms of "real" quantities.

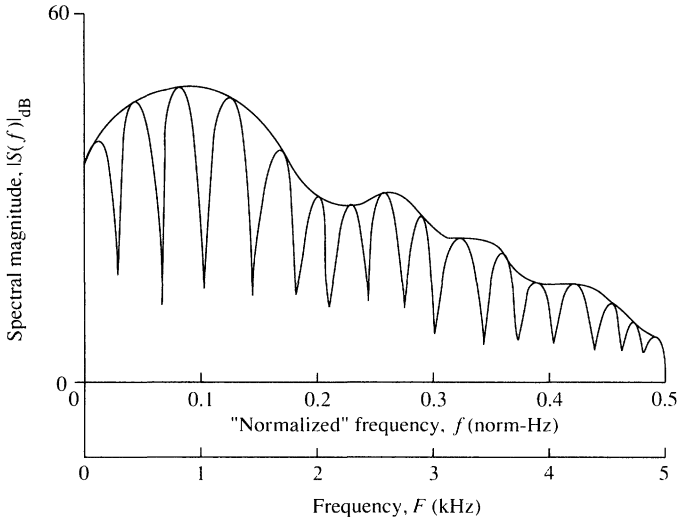


FIGURE 1.2. Magnitude spectrum of a typical speech waveform. This spectrum is based on the DFT of samples of the waveform taken at 10 kHz. The lower frequency axis represents real frequencies measured in kHz, while the upper represents a normalization of the frequency axis concomitant to the time normalization. Normalized frequency, f, is related to real frequency, F, as $f = FT$, where T is the nominal sample period in a given analysis. Accordingly, the "normalized" sampling, say f_s, and Nyquist, f_N, frequencies are invariant with the sample rate, with $f_s = 1$ and $f_N = 1/2$. We will sometimes refer to the units of normalized frequencies as "normalized Hertz (norm-Hz)" or "normalized radians per second (norm-rps)."

1.1.2 Singularity Signals

In the continuous time domain, a *singularity signal* is one for which one or more derivatives do not exist at one or more time points. Although the concept of a derivative is no longer meaningful in discrete time, we often borrow the term *singularity* to describe analogous sequences in sampled time. The two for which we will have the most use are

- The *unit sample sequence* or *discrete-time impulse*, defined by

$$\delta(n) \stackrel{\text{def}}{=} \begin{cases} 1, & \text{if } n = 0 \\ 0, & \text{otherwise} \end{cases}. \tag{1.7}$$

- The *unit step sequence*, defined by[5]

$$u(n) \stackrel{\text{def}}{=} \begin{cases} 1, & \text{if } n \geq 0 \\ 0, & \text{otherwise} \end{cases}. \tag{1.8}$$

The unit step sequence is much more analogous to its analog counterpart than the discrete-time impulse in the sense that $u(n)$ simply represents samples of the analog unit step function (discounting problems that may arise due to different definitions at time zero). On the other hand, recall that the *analog* (Dirac) impulse function, say $\delta_a(t)$, is defined such that it apparently has infinite height, zero width, and unity area. Although the discrete-time impulse plays an analogous role to that played by the analog impulse, it may *not* be interpreted as its samples.

1.1.3 Energy and Power Signals

There are many ways in which a discrete time signal can be classified. One useful grouping is into the categories energy signal, power signal, or neither. Recall that the *energy* of a discrete time signal is defined as[6]

$$E_x \stackrel{\text{def}}{=} \sum_{n=-\infty}^{\infty} |x(n)|^2. \tag{1.9}$$

A signal $x(n)$ is called an *energy signal* if

$$0 < E_x < \infty. \tag{1.10}$$

The *power* in a discrete-time sequence is

[5]The notation $u(n)$ is widely used to indicate the unit step sequence, but $u(n)$ will also refer to a very important waveform, the "glottal volume velocity," throughout the book. Because of the context, there will be no risk of confusion.

[6]The absolute value signs are included because, in general, $x(n)$ is a complex sequence.

$$P_x \stackrel{\text{def}}{=} \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} |x(n)|^2. \tag{1.11}$$

A *power signal* has finite but nonzero power,

$$0 < P_x < \infty. \tag{1.12}$$

A signal cannot be both a power signal and an energy signal simultaneously, since if $E_x < \infty$, then $P_x = 0$. A signal can, however, be *neither* when $P_x = \infty$ or $E_x = 0$.

For our purposes in speech processing, it is sufficient to associate the energy category with two broad classes of signals. These are

- *Transients*, those which decay (usually exponentially) with time. Examples are

$$x_1(n) = \alpha^n u(n), \qquad |\alpha| < 1 \tag{1.13}$$

$$x_2(n) = \alpha^{|n|} \cos(n\omega_0 + \psi), \qquad |\alpha| < 1. \tag{1.14}$$

- *Finite sequences*, those which are zero outside a finite time duration. An example is

$$x_3(n) = e^{\beta n}[u(n+3) - u(n-246)], \qquad |\beta| < \infty. \tag{1.15}$$

Whereas the energy signals either decay out sufficiently fast or "stop" completely, the power signals neither decay nor increase in their envelopes. The power signals can be associated with three broad classes of signals. These are

- *Constant* signals. An example is

$$x_4(n) = \alpha \qquad -\infty < \alpha < \infty. \tag{1.16}$$

- *Periodic* signals, those for which $x(n) = x(n+N)$ for some finite N and for all n. Examples are

$$x_5(n) = \alpha \sin(n\omega_0 + \psi), \qquad -\infty < \alpha < \infty \tag{1.17}$$

$$x_6(n) = [x_3(n)]_{\text{modulo } 512} = \sum_{i=-\infty}^{\infty} x_3(n + i512). \tag{1.18}$$

- *Realizations of stationary, ergodic stochastic processes* (see Section 1.2.3).

The signals which fall into neither category are the trivial zero signal and those which "blow up" with time. Examples of the latter are $x_1(n)$ and $x_2(n)$ above with the magnitude of α taken to be greater than unity.

1.1.4 Transforms and a Few Related Concepts

At the heart of much of engineering analysis are various frequency domain transforms. Three transforms on discrete-time data will be used ex-

tensively throughout this book, and it will be assumed that the reader is familiar with their properties and usage.

The first is the *discrete-time Fourier transform* (DTFT), which, for the sequence $x(n)$, is defined by

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}. \tag{1.19}$$

The *inverse DTFT* (IDTFT) is given by

$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega)e^{j\omega n}\, d\omega. \tag{1.20}$$

The DTFT bears a useful relationship to the continuous-time Fourier transform in the case in which $x(n)$ represents samples of the analog signal[7] $x_a(t')$. In this case $X(\omega)$ will be a periodic (with period 2π), potentially aliased version of $X_a(\omega)$,

$$X(\omega) = \sum_{i=-\infty}^{\infty} X_a(\omega - 2\pi i). \tag{1.21}$$

The existence of the DTFT is not a trivial subject, and we will review only a few important details. A sufficient condition for the DTFT of a sequence $x(n)$ to exist is that the sequence be *absolutely summable*,

$$\sum_{n=-\infty}^{\infty} |x(n)| < \infty. \tag{1.22}$$

This follows immediately from (1.19). Moreover, absolute summability of $x(n)$ is tantamount to *absolute convergence* of the series $\sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}$ implying that this series *converges uniformly* to a continuous function of ω (Churchill, 1960, Secs. 59 and 60). A sequence that is absolutely summable will necessarily be an energy signal, since

$$E_x = \sum_{n=-\infty}^{\infty} |x(n)|^2 \leq \left[\sum_{n=-\infty}^{\infty} |x(n)| \right]^2. \tag{1.23}$$

There are, however, energy signals that are not absolutely summable (see Problem 1.2). These energy signals will still have DTFTs, but ones whose series converge in a weaker (*mean square*) sense. This can be seen by viewing (1.19) as a conventional Fourier series for the periodic function $X(\omega)$ whose coefficients are $x(n)$. One of the properties of Fourier series is that if the energy in a single period of the function is finite, then the

[7]Note the use of "normalized time" here. If "real" time is used, (1.21) becomes

$$X(\Omega) = \frac{1}{T} \sum_{i=-\infty}^{\infty} X_a\left(\Omega - \frac{2\pi}{T} i\right).$$

series will converge in mean square (Churchill, 1963). In the present case (using the Parseval relation),

$$\int_{-\pi}^{\pi} |X(\omega)|^2 d\omega = 2\pi \sum_{n=-\infty}^{\infty} |x(n)|^2 = 2\pi E_x < \infty, \tag{1.24}$$

so the DTFT will converge in the mean square sense. Practically, this means that the DTFT sum will converge to $X(\omega)$ at all points of continuity, and at points of discontinuity it will converge to the "average" value ("halfway" between the values on either side of the discontinuity).

Properties of the DTFT are detailed in the textbooks cited in Appendix 1.A, and some are reviewed in Problem 1.3. The reader should also recall the numerous symmetry properties of the transform relation which can be useful in simplifying various computations and algorithms.

Whereas the DTFT is most useful for theoretical spectral analysis, it is not computable on a digital computer because it is a function of a continuous argument. In principle, it also works with a sequence of doubly infinite length, which also precludes any practical computation. If we restrict ourselves to the practical situation in which a sequence of finite length is being studied, then the *discrete Fourier transform* (DFT) provides a mapping between the sequence, say

$$x(n), \quad n = 0, 1, 2, \ldots, N-1 \tag{1.25}$$

and a discrete set of frequency domain samples, given by

$$X(k) = \begin{cases} \sum_{n=0}^{N-1} x(n) e^{-j(2\pi/N)kn}, & k = 0, 1, \ldots, N-1 \\ 0, & \text{other } k \end{cases}. \tag{1.26}$$

The *Inverse DFT* (IDFT) is given by

$$x(n) = \begin{cases} \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j(2\pi/N)kn}, & n = 0, 1, \ldots, N-1 \\ 0, & \text{other } n \end{cases}. \tag{1.27}$$

The DFT represents exact samples of the DTFT of the finite sequence $x(n)$ at N equally spaced frequencies, $\omega_k = (2\pi/N)k$, for $k \in [0, N-1]$.

The *discrete Fourier series* (DFS) is closely related to the DFT *computationally*, but is quite different philosophically. The DFS is used to represent a *periodic* sequence (hence a *power* signal) with period, say N, using the set of basis functions $e^{j(2\pi/N)kn}$ for $k = 0, \ldots, N-1$. These represent the N harmonic frequencies that may be present in the signal. For a periodic sequence $y(n)$, the expansion is

$$y(n) = \sum_{k=0}^{N-1} C(k) e^{j(2\pi/N)kn}, \tag{1.28}$$

where the coefficients are computed as

$$C(k) = \frac{1}{N} \sum_{n=0}^{N-1} y(n) e^{-j(2\pi/N)kn}. \tag{1.29}$$

[In principle, the $C(k)$'s may be computed over any period of $y(n)$.]

It is occasionally convenient to use an "engineering DTFT" for a periodic signal that technically has no DTFT. The contrived DTFT composed of *analog* impulse functions at the harmonic frequencies weighted by the DFS coefficients is

$$Y(\omega) = 2\pi \sum_{k=-\infty}^{\infty} C(k) \delta_a\left(\omega - k\frac{2\pi}{N}\right). \tag{1.30}$$

Such a construction is not always palatable to a mathematician, but it works for most engineering purposes in the sense that it can be used anywhere that a DTFT is needed for $y(n)$, as long as the rules for *continuous-time* impulse functions are carefully followed. Note that this DTFT correctly asserts, for example, that $y(n)$ has infinite energy at the harmonic frequencies. Consistency with conventional Fourier transform computations is obtained by defining the *magnitude spectrum* of such a DTFT by

$$|Y(\omega)| \stackrel{\text{def}}{=} 2\pi \sum_{k=-\infty}^{\infty} |C(k)| \delta_a\left(\omega - k\frac{2\pi}{N}\right). \tag{1.31}$$

One more contrived quantity is sometimes used. The *power density spectrum* (PDS) for a periodic signal $y(n)$, say $\Gamma_y(\omega)$, is a real-valued function of frequency such that the average power in $y(n)$ on the frequency range ω_1 to ω_2 with $0 \le \omega_1 < \omega_2 < 2\pi$ is given by

$$\begin{aligned} \text{average power in } y(n) \text{ on } \omega \in [\omega_1, \omega_2] &= \frac{1}{\pi} \int_{\omega_1}^{\omega_2} \Gamma_y(\omega)\, d\omega \\ &= 2 \sum_{k=k_1}^{k_2} |C(k)|^2, \end{aligned} \tag{1.32}$$

where k_1 and k_2 represent the integer indices of the lowest and highest harmonic of $y(n)$ in the specified range.[8] It is not difficult to show that a suitable definition is

$$\Gamma_y(\omega) \stackrel{\text{def}}{=} 2\pi \sum_{k=-\infty}^{\infty} |C(k)|^2 \delta_a\left(\omega - k\frac{2\pi}{N}\right). \tag{1.33}$$

[8]If ω_1 and hence k_1 are zero, then the lower expression in (1.32) should read

$$C(0) + 2 \sum_{k=1}^{k_2} |C(k)|^2.$$

[The reader can confirm that this is consistent with (1.32).] By comparing with (1.30), it is clear why some authors choose to write $|Y(\omega)|^2$ as a notation for the PDS. The advantage of doing so is that it gives the contrived DTFT of (1.30) yet another "DTFT-like" property in the following sense: $|X(\omega)|^2$ is properly called the *energy density spectrum* for an *energy* signal $x(n)$ and can be integrated over a specified frequency range to find total energy in that range. $|Y(\omega)|^2$ is thus an analogous notation for an analogous function for a power sequence. The disadvantage is that it introduces more notation which can be easily confused with a more "valid" spectral quantity. We will therefore use only $\Gamma_y(\omega)$ to indicate the PDS of a periodic signal $y(n)$.

The similarity of the DFT to the DFS is apparent, and this similarity is consistent with our understanding that the IDFT, if used outside the range $n \in [0, N-1]$, will produce a periodic replication of the finite sequence $x(n)$. Related to this periodic nature of the DFT are the properties of "circular shift" and "circular convolution" of which the reader must beware in any application of this transform. A few of these notions are reviewed in Problem 1.4.

For interpretive purposes, it will be useful for us to note the following. Although the DTFT does not exist for a periodic signal, we might consider taking the limit[9]

$$\overline{Y}(\omega) = \lim_{N \to \infty} \frac{1}{2N+1} \sum_{n=-N}^{N} y(n) e^{-j\omega n} \tag{1.34}$$

in the hope of making the transform converge. A moment's thought will reveal that this computation is equivalent to the same sum taken over a single period, say

$$\overline{Y}(\omega) = \frac{1}{N} \sum_{n=0}^{N-1} y(n) e^{-j\omega n}. \tag{1.35}$$

We shall refer to $\overline{Y}(\omega)$, $0 \leq \omega < 2\pi$, as the *complex envelope spectrum* for a periodic signal $y(n)$. The theoretical significance of the complex envelope is that it can be sampled at the harmonic frequencies to obtain the DFS coefficients for the sequence. The reader will recall that a similar phenomenon occurs in the analog domain where the FT of one period of a periodic signal can be sampled at the harmonics to obtain the FS coefficients.

Finally, with regard to Fourier techniques, recall that the *fast Fourier transform* (FFT) is a name collectively given to several classes of fast algorithms for computing the DFT. The literature on this subject is vast,

[9]The operator notation $\mathcal{L}\{\cdot\}$ will be used consistently in the text to denote a time average of this form. We will formally define the operator when we discuss averages in Section 1.2.3.

but the textbooks cited above provide a general overview of most of the fundamental treatments of the FFT. Some advanced topics are found in (Burris, 1988).

The final transform that will be used extensively in this book is the (two-sided) *z-transform* (ZT), defined by

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}, \tag{1.36}$$

where z is any complex number for which the sum exists, that is, for which

$$\sum_{n=-\infty}^{\infty} |x(n)z|^{-n} < \infty. \tag{1.37}$$

The values of z for which the series converges comprise the *region of convergence* (ROC) for the ZT. When the series converges, it converges absolutely (Churchill, 1960, Sec. 59), implying that the ZT converges uniformly as a function of z everywhere in the ROC. Depending on the time sequence, the ROC may be the interior of a circle, the exterior of a circle, or an annulus of the form $r_{\text{in}} < |z| < r_{\text{out}}$, where r_{in} may be zero and r_{out} may be infinite. The ROC is often critical in uniquely associating a time sequence with a ZT. For details see the textbooks in Appendix 1.A.

The ZT is formally inverted by contour integration,

$$x(n) = \frac{1}{2\pi j} \oint_{\mathcal{C}} X(z)z^{n-1}\, dz, \tag{1.38}$$

where $\mathcal{C}$ is a counterclockwise contour through the ROC and encircling the origin in the z-plane, but several useful computational methods are well known, notably the *partial fraction expansion* method, and the *residue* method.

The ZT plays a similar role in DSP to that which the Laplace transform does in continuous processing. A good speech processing engineer will learn to "read the z-plane" much the same as the analog systems engineer uses the s-plane. In particular, the reader should be familiar with the correspondence between angles in the z-plane and frequencies, and between z-plane magnitudes and "damping." The interpretation of pole–zero plots in the z-plane is also an essential tool for the speech processing engineer.

Finally, we recall the relationships among the two Fourier transforms and the ZT. From the definitions, it is clear that

$$^{\text{DTFT}}X(\omega) = {}^{\text{ZT}}X(e^{j\omega}) \tag{1.39}$$

for any ω, so that the DTFT at frequency ω is obtained by evaluating the ZT at angle ω on the unit circle in the z-plane. This is only valid, of

course, when the ROC of the ZT includes the unit circle of the z-plane.[10] The periodicity with period 2π of the DTFT is consistent in this regard. Since the DFT represents samples of the DTFT at frequencies ω_k, $k = 0, 1, \ldots, N-1$, it can be obtained by evaluating the ZT at equally spaced angles around the unit circle in the z-plane. Therefore,

$$^{\text{DFT}}X(k) = {}^{\text{DTFT}}X\left(\omega_k = \frac{2\pi}{N}k\right) = {}^{\text{ZT}}X(e^{j(2\pi/N)k}). \tag{1.40}$$

Since we use the same "uppercase," for example, X, notation to indicate all three transforms, it is occasionally necessary in DSP work to explicitly denote the particular transform with, for example, a presuperscript as in (1.40).

1.1.5 Windows and Frames

In all practical signal processing applications, it is necessary to work with *short terms* or *frames* of the signal, unless the signal is of short duration.[11] This is especially true if we are to use conventional analysis techniques on signals (such as speech) with nonstationary dynamics. In this case it is necessary to select a portion of the signal that can reasonably be assumed to be stationary.

Recall that a (time domain) *window*, say $w(n)$, is a real, finite length sequence used to select a desired frame of the original signal, say $x(n)$, by a simple multiplication process. Some of the commonly used window sequences are shown in Fig. 1.3. For consistency, we will assume windows to be *causal* sequences beginning at time $n = 0$. The duration will usually be denoted N. Most commonly used windows are symmetric about the time $(N-1)/2$ where this time may be halfway between two sample points if N is even. Recall that this means that the windows are *linear-phase* sequences [e.g., see (Proakis and Manolakis, 1992)] and therefore have DTFTs that can be written

$$W(\omega) = |W(\omega)|\, e^{-j\omega((N-1)/2)}, \tag{1.41}$$

where the phase term is a simple linear characteristic corresponding to the delay of the window that makes it causal.[12]

It will be our convention in this book to use windows in a certain manner to create a frame of the signal. We first reverse the window in time[13] $[w(-n)]$, then shift it so that its leading edge is at a desired time,

[10]The ROC includes the unit circle if and only if $x(n)$ is absolutely summable. Therefore, in keeping with our discussion above, only a uniformly convergent DTFT can be obtained by evaluating the corresponding ZT on the unit circle.

[11]A similar discussion applies to the design of FIR filters by truncation of a desired IIR (see DSP textbooks cited in Appendix 1.A).

[12]If the window were allowed to be centered on $n = 0$, it would have a purely real DTFT and a zero-phase characteristic.

[13]Since we assume windows to be symmetric about their midpoints, this reversal is just to initially shift the leading edge to time zero.

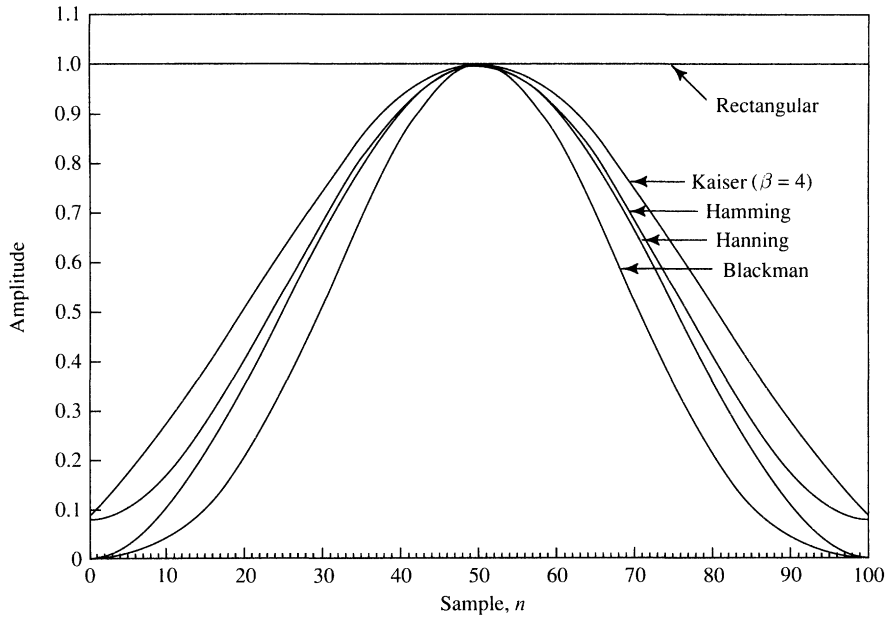


FIGURE 1.3. Definitions and example time plots for the rectangular, Kaiser, Hamming, Hanning, and Blackman windows. All plots are for window lengths $N = 101$, and for the Kaiser window, $\beta = 4$.

m $[w(m-n)]$. A *frame* of the signal $x(n)$ of length N (same as the duration of the window) ending at time m, say $f_x(n;m)$, is obtained as

$$f_x(n;m) = x(n)w(m-n). \tag{1.42}$$

This simple concept will be used extensively in future developments involving frames of speech. In fact, much of the time in this book the frame will be related to a speech sequence denoted $s(n)$ and it will be unnecessary to employ the subscript s because it will be obvious. We will only use a subscript in discussions where frames are being created from more than one signal.

Assume for the moment that $x(n)$ is a stationary signal for all time. Clearly, the temporal properties of $f_x(n;m)$ are distorted with respect to those of $x(n)$ due to the direct modification of the temporal sequence by the window. Correspondingly, the spectral properties also differ as the two transforms are apparently convolved. That is, if $F_x(\omega;m)$ denotes the DTFT of frame $f_x(n;m)$, then

$$F_x(\omega;m) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\omega-\theta)W(-\theta)e^{-j\theta m}\,d\theta. \tag{1.43}$$

Now the relationship between $F_x(\omega;m)$ and $X(\omega)$ will only be clear from (1.43) to those who are able to visualize the process of convolving complex functions! Most of us do not have such an imagination. However, we

can get some insight into the spectral distortion by assuming with some loss of generality that the reversed and shifted window is centered on time $n = 0$ $[m = (N-1)/2]$. Said another way, this simply means that the true signal transform, $X(\omega)$, against which we are going to compare our frame's transform, $F_x(\omega; m)$, is the one whose signal is assumed to have its time origin in the middle of the window. This, of course, is not always the $X(\omega)$ that represents our standard, but we can use it for insight. In this case (1.41) can be used in (1.43) to yield

$$F_x(\omega; m) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega - \theta) \left| W(-\theta) \right| d\theta = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega - \theta) \left| W(\theta) \right| d\theta \tag{1.44}$$

where we have replaced $|W(-\theta)|$ by $|W(\theta)|$ since the magnitude spectrum is an even function of θ. In this light it seems that we want our window to have a magnitude spectrum that approximates an (analog) impulse as closely as possible,

$$\left| W(\theta) \right| \approx 2\pi\delta_a(\theta), \tag{1.45}$$

since this will imply $F_x(\omega; m) \approx X(\omega)$. When we ponder this for a moment we realize that, in the extreme case, we have concluded the obvious because $|W(\theta)| = 2\pi\delta_a(\theta)$ implies that $w(n) = 1$ for all n. The "best" window in terms of preserving the spectrum is *no* window at all! Of course such a "window" will also preserve the temporal properties of the signal perfectly as well.

For any meaningful window, however, it is the extent to which the approximation (1.45) holds which will determine the preservation of the spectral features of $X(\omega)$. Now all commonly used windows tend to have "lowpass" spectra with one main lobe at low frequencies and various attenuated "sidelobes." This is consistent with the fact that, if viewed as the (usually finite) impulse response of a filter, the window has an averaging effect. Shown in Fig. 1.4, for example, are the magnitude spectra of two commonly used windows, the *rectangular* window, defined as

$$w(n) = \begin{cases} 1, & n = 0, 1, \ldots, N-1 \\ 0, & n \text{ otherwise,} \end{cases} \tag{1.46}$$

and the *Hamming* window,

$$w(n) = \begin{cases} 0.54 - 0.46 \cos(2\pi n/N - 1), & n = 0, 1, \ldots, N-1 \\ 0, & n \text{ otherwise.} \end{cases} \tag{1.47}$$

Each is plotted for the case $N = 16$. For any window spectrum to approximate $\delta_a(\omega)$, therefore, there are two desirable features:

- A narrow bandwidth main lobe.
- Large attenuation in the sidelobes.

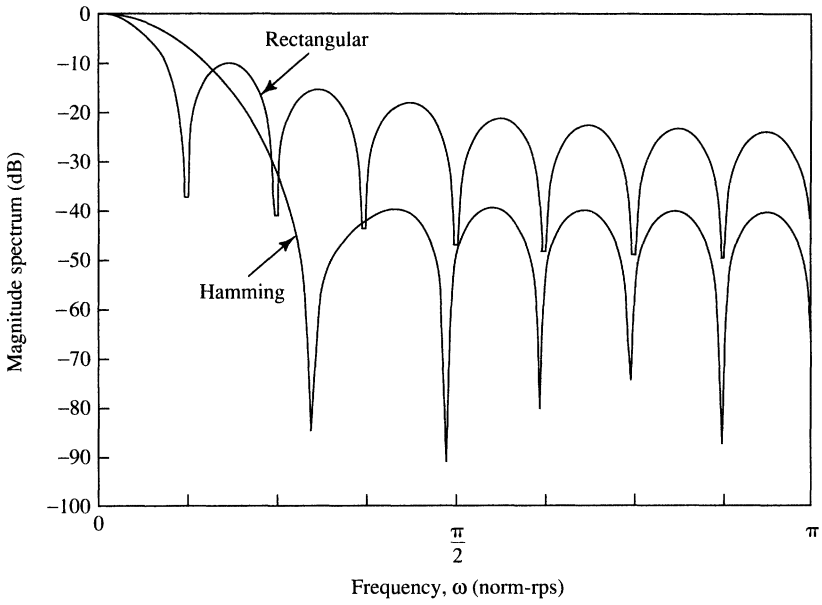


FIGURE 1.4. Magnitude spectra of rectangular and Hamming windows. Window length $N=16$ is used in each case for clarity. Note that the nominal "bandwidth" (width of main lobe) is $2\pi/N=\pi/8$ for the rectangular case and about twice that for the Hamming. The sidelobe attenuation for the Hamming, however, is 20 dB better outside the passband.

Generally speaking, a narrow main lobe will resolve the sharp details of $|X(\omega)|$ as the convolution (1.44) proceeds, while the attenuated sidelobes will prevent "noise" from other parts of the spectrum from corrupting the true spectrum at a given frequency. As one might expect, an engineering trade-off is encountered in this regard in the choice of a window. The rectangular window, which exactly preserves the temporal characteristics of the waveform over a range of N points, but which abruptly truncates the waveform at the boundaries, has the following spectral characteristics:

- A relatively narrow main lobe (see Fig. 1.4) which decreases with N. (In fact, the width of all the lobes decreases with N, but remember that a very large N begins to defeat the purpose of windowing.)
- The height of all lobes grows with N in such a way that the attenuation in the sidelobes is approximately constant as N grows. This sidelobe attenuation is not good for the rectangular case, typically −20 dB with respect to the main lobe, allowing lots of undesirable spectral energy to be dragged into the resulting spectrum by the convolution (1.44) at a given ω.

Windows with smoother truncations, such as the *Kaiser*, *Hamming*, *Hanning*, and *Blackman* are generally used (see Fig. 1.3). These tend to distort the temporal waveform on the range of N points, but with the

benefit of less abrupt truncations at the boundaries. The spectral properties of these windows are generally described as follows:

- For a given N, all will have a wider main lobe than the rectangular. Again, this width decreases with increasing N.
- All have better sidelobe attenuation than the rectangular, typically 10–60 dB better. The popular Hamming window, for example, is −30 dB down in the sidelobes (Fig. 1.4).

Although the choice of windows is somewhat of an art dependent upon experience rather than an exact science, one can use this discussion as a guide to an analytical understanding of the effects of such a choice in the processing of speech or any other signal. Generally, the choice of smoother windows is made because of their preferable sidelobe characteristics.

When analyzing a nonstationary signal like speech, the selection of a window involves another important consideration which is often not treated in introductory textbooks on digital signal processing. From the discussion above, we see that when analyzing a stationary signal, increasing the window length, N, has only beneficial consequences regardless of the type of window used. However, if a window is to be used to sequentially select portions of a nonstationary signal by "sliding" it along in time, a longer window will require a longer period to cross transitional boundaries in the signal and events from different quasi-stationary regions will tend to be blurred together more frequently than if the window were shorter. Therefore another engineering trade-off is encountered in the choice of window *length*. A longer window will tend to produce a better spectral picture of the signal while the window is completely within a stationary region, whereas a shorter window will tend to resolve events in the signal better in time. This trade-off is sometimes called the *spectral-temporal resolution trade-off* and will be discussed further in Chapter 4, where we deal with short-term processing of speech.

1.1.6 Discrete-Time Systems

Elementary Concepts

The following are elementary concepts from discrete time (DT) system theory that will be used intrinsically and extensively throughout the book. It is assumed that the reader has a thorough grounding in these ideas. We list here a number of fundamental topics that will be used without elaboration. If any are unfamiliar, the reader is advised to review them in one of the introductory textbooks indicated in Appendix 1.A.

1. Linearity.
2. Time (shift) invariance.
3. Linear, constant-coefficient difference equation (time domain input–output) description of a linear, time-invariant (LTI) DT system.

4. DT impulse response of an LTI DT system ["$h(n)$"].
5. Convolution sum for an LTI DT system.
6. Bounded input-bounded output (BIBO) stability and relationship to $h(n)$ for an LTI DT system.
7. Causality.
8. System function for an LTI DT system ["$H(z)$"], poles and zeros.
9. Magnitude spectrum and phase spectrum of an LTI DT system and their determination from a pole-zero diagram.
10. Relationship between the linear constant coefficient difference equation and $H(z)$ for an LTI DT system.
11. Relationship between BIBO stability and $H(z)$.
12. Finite impulse response (FIR) and infinite impulse response (IIR) systems and relationships to $H(z)$ and the difference equation.
13. Canonical computational structures for implementing LTI DT systems.

State-Space Realizations of LTI DT Systems

Much of contemporary DT and analog system theory is based upon *state-space* descriptions, rather than input–output descriptions, of systems. In the digital signal processing realm, state-space structures for realizing DT systems have been the subject of intense research and they have been found to have a number of useful numerical properties [e.g., see (Jackson, 1989, Sec. 11.6)]. We have two limited and very specific uses for them in our work, so we review only a few pertinent results here. The reader can refer to a number of other textbooks for further information (see Appendix 1.A).

In the proof of a key result concerning linear prediction analysis in Chapter 5, we will have need of a slight variation of a *Type I* (Proakis and Manolakis, 1992, Sec. 7.5) or *controllable canonical* (Chen, 1984, p. 327) form for a specific LTI DT system with scalar input and output. This form is derived from the input–output description of the system as follows: Consider the system to be governed by the linear constant coefficient difference equation

$$y(n) = \sum_{k=1}^{M} a(k)y(n-k) + \sum_{k=0}^{Q} b(k)x(n-k) \tag{1.48}$$

for which the *direct form II realization* is shown in Fig. 1.5. We assume in that figure and in this discussion that $Q < M$ and we define $b(k) = 0$ for $k > Q$. The (*internal*) *state* of a DT system at time n_0 is defined to be the quantitative information necessary at time n_0 which, together with the input $x(n)$ for $n \geq n_0$, uniquely determines the output $y(n)$ for $n \geq n_0$. The *state variables* of the system are the numerical quantities memorized by the system that comprise the state.

In Fig. 1.5 we have defined the internal variables $v_1(n), \ldots, v_M(n)$.

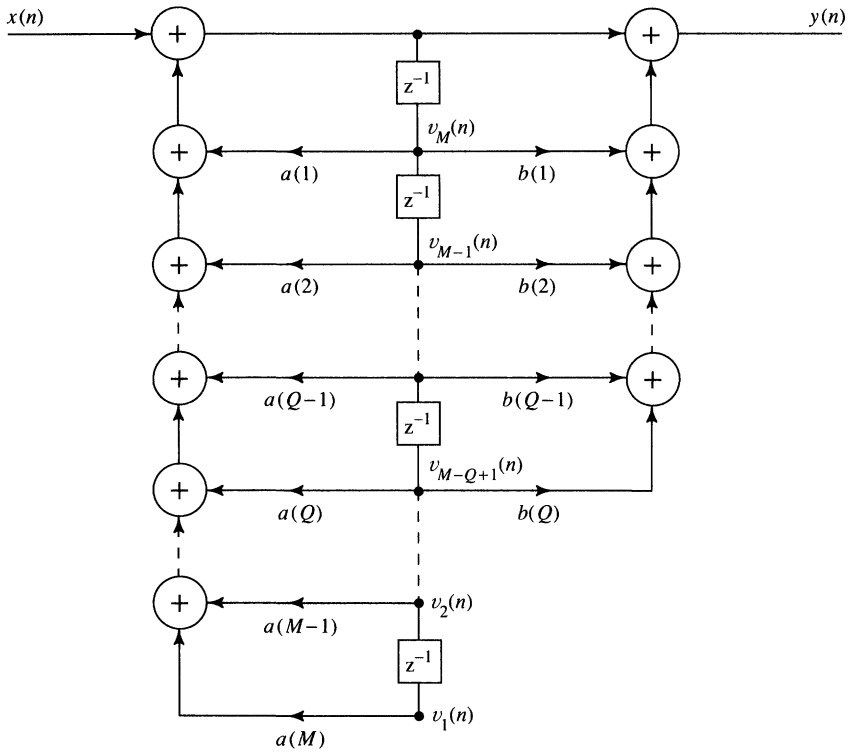


FIGURE 1.5. Direct form II realization of the discrete-time system with input–output description (1.48).

These comprise state variables for this system, as we shall see momentarily. Note that

$$v_i(n+1) = v_{i+1}(n), \qquad i = 1, 2, \ldots, M-1 \tag{1.49}$$

$$v_M(n+1) = x(n) + \sum_{i=1}^{M} a(i) v_{M-i+1}(n). \tag{1.50}$$

These are the *state equations* for the system. Note also that the output can be computed from the state variables at time n using

$$y(n) = b(0)\, v_M(n+1) + \sum_{i=1}^{M} b(i)\, v_{M-i+1}(n) \tag{1.51}$$

$$= b(0)x(n) + \sum_{i=1}^{M} [b(i) + b(0)a(i)]\, v_{M-i+1}(n),$$

which is called simply the *output equation* for the system. It is clear that these state variables do comprise a legitimate state for this system ac-

cording to the definition. For convenience, the state and output equations can be written in vector-matrix form as

$$\mathbf{v}(n+1) = \mathbf{A}\mathbf{v}(n) + \mathbf{c}x(n) \tag{1.52}$$

$$y(n) = \mathbf{b}^T\mathbf{v}(n) + dx(n), \tag{1.53}$$

in which d is the scalar $d = b(0)$, $\mathbf{A}$ is the $M \times M$ *state transition matrix*

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 \\ a(M) & a(M-1) & a(M-2) & a(M-3) & a(M-4) & \cdots & a(1) \end{bmatrix}, \tag{1.54}$$

and $\mathbf{c}$ and $\mathbf{b}$ are M-vectors [recall the assumption $Q < M$ and the definition $b(k) = 0$ for $k > Q$],

$$\mathbf{c} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix}^T \tag{1.55}$$

$$\mathbf{b} = \begin{bmatrix} b(M) + b(0)a(M) \\ b(M-1) + b(0)a(M-1) \\ b(M-2) + b(0)a(M-2) \\ \vdots \\ b(1) + b(0)a(1) \end{bmatrix}. \tag{1.56}$$

Equations (1.52) and (1.53) are very close to the state-space description of an LTI system that will be needed in our work in a limited way. In fact, because of the way we have chosen to define the state variables here, these equations comprise a *lower companion form* state-space model, so named because of the form of the state transition matrix $\mathbf{A}$. A simple redefinition of state variables leads to the *upper companion form* model which we explore in Problem 1.5.

Finally, in our study of hidden Markov models for speech recognition in Chapter 12, we will have need of a state-space description of a system that has a *vector output*. In this case the system will naturally arise in a state-space form and there will be no need for us to undertake a conversion of an input–output description of the system. The system there will

have a similar state equation to (1.52) except that the state transition matrix, **A**, will generally not be of a special form like the one above (indicating more complicated dependencies among the states). The output equation will take the form

$$\mathbf{y}(n) = \mathbf{B}\mathbf{v}(n) + \mathbf{d}\mathbf{x}(n) \tag{1.57}$$

in which $\mathbf{y}(n)$ and $\mathbf{d}$ are P-vectors (P outputs) and $\mathbf{B}$ is a $P \times M$ matrix. We will have more to say about this system when its need arises.

1.1.7 Minimum-, Maximum-, and Mixed-Phase Signals and Systems

We have discussed the grouping of signals into energy or power categories. Here we restrict our attention to the subclass of real signals with legitimate DTFTs (those that are absolutely summable) and consider another useful categorization.

The specification of the magnitude spectrum of a discrete-time signal is generally not sufficient to uniquely specify the signal, or, equivalently, the DTFT of the signal. Consider, for example, the magnitude spectrum, $|X(\omega)|$, shown in Fig. 1.6. This spectrum was actually computed for the signal $x_1(n)$ with z-transform,

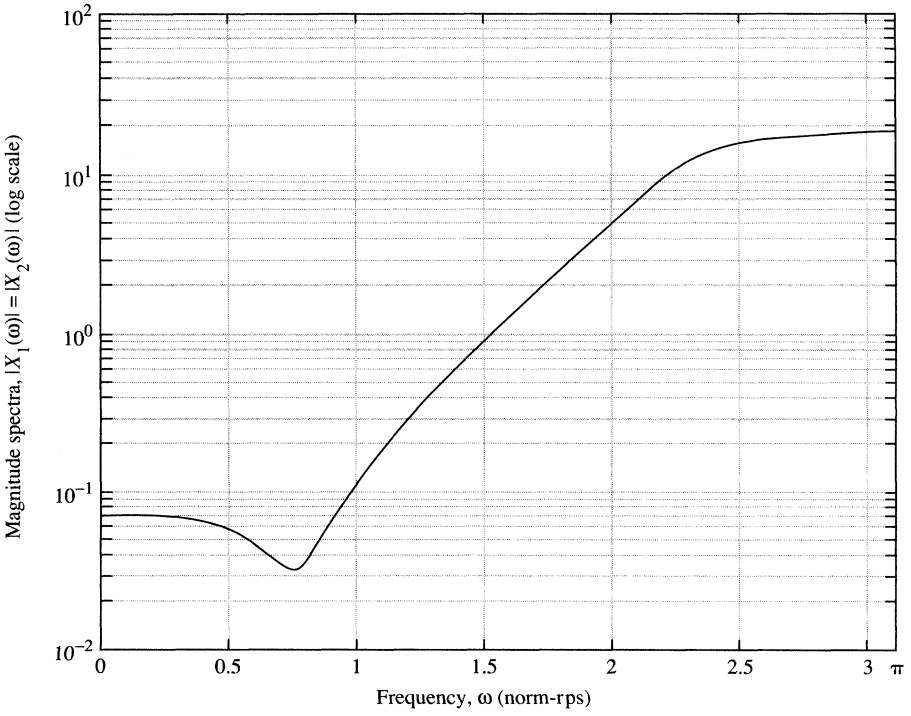


FIGURE 1.6. Common magnitude spectrum for the signals $x_1(n)$ and $x_2(n)$.

$$X_1(z) = \frac{(1-\zeta_1 z^{-1})(1-\zeta_1^* z^{-1})(1-\zeta_2 z^{-1})}{(1-\rho_1 z^{-1})(1-\rho_1^* z^{-1})(1-\rho_2 z^{-1})}, \tag{1.58}$$

with $\zeta_1 = 0.9\angle 45°$, $\zeta_2 = 0.5$, $\rho_1 = 0.7\angle 135°$, and $\rho_2 = -0.5$, and therefore has an analytical description

$$\left|X_1(e^{j\omega})\right| = \left| \frac{(1-\zeta_1 e^{-j\omega})(1-\zeta_1^* e^{-j\omega})(1-\zeta_2 e^{-j\omega})}{(1-\rho_1 e^{-j\omega})(1-\rho_1^* e^{-j\omega})(1-\rho_2 e^{-j\omega})} \right|. \tag{1.59}$$

The true phase characteristic is given by

$$\arg\{X_1(e^{j\omega})\} = \arg\left\{ \frac{(1-\zeta_1 e^{-j\omega})(1-\zeta_1^* e^{-j\omega})(1-\zeta_2 e^{-j\omega})}{(1-\rho_1 e^{-j\omega})(1-\rho_1^* e^{-j\omega})(1-\rho_2 e^{-j\omega})} \right\}. \tag{1.60}$$

The magnitude and phase spectra for the signal $x_1(n)$ are found in Figs. 1.6 and 1.7, respectively, and the pole–zero diagram is shown in Fig. 1.8.

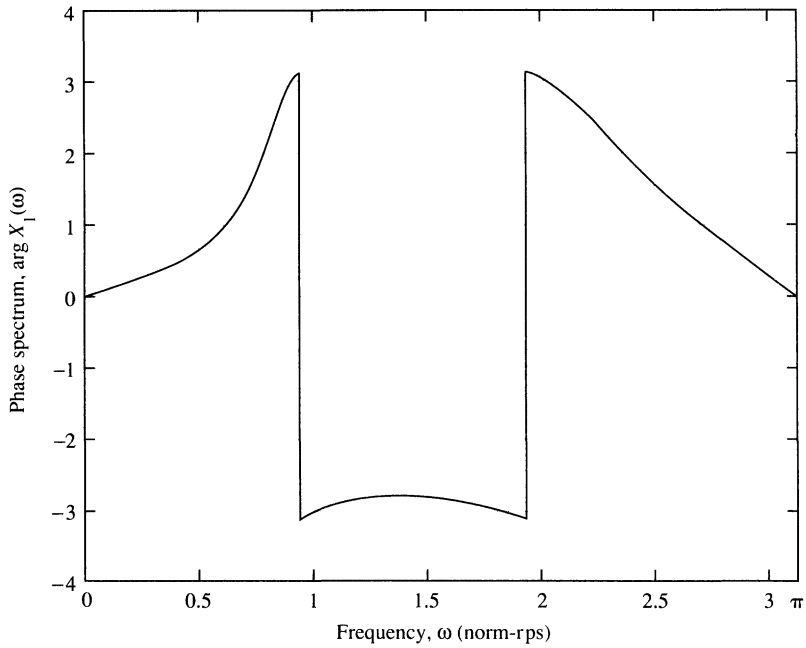
If the magnitude spectrum were all that were known to us, however, it would not be possible to deduce this z-transform and corresponding signal with certainty. Indeed, consider the signal $x_2(n)$ with z-transform

$$X_2(z) = \frac{(z^{-1}-\zeta_1)(z^{-1}-\zeta_1^*)(z^{-1}-\zeta_2)}{(1-\rho_1 z^{-1})(1-\rho_1^* z^{-1})(1-\rho_2 z^{-1})}, \tag{1.61}$$

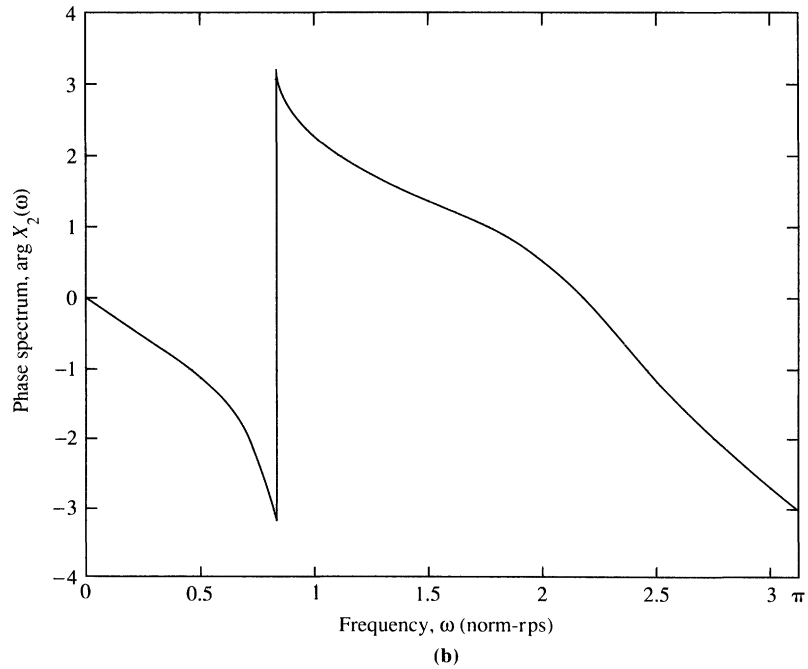
which the reader can confirm has an identical magnitude spectrum to $X_1(z)$ (see Fig. 1.6), but a different phase spectrum that is shown in Fig. 1.7. The pole–zero diagram for the signal $x_2(n)$ is found in Fig. 1.8. Furthermore, there are two other z-transforms which have identical magnitude spectra but different phase spectra. $X_2(z)$ is found from $X_1(z)$ by reflecting both the conjugate zero pair plus the real zero into conjugate reciprocal locations (outside the unit circle) in the z-plane, plus some scaling. The other two magnitude spectrum-equivalent z-transforms are found by reflecting *either* the conjugate pair *or* the real zero.

In general, if a real, causal, absolutely summable signal has a z-transform with C complex pairs of zeros, and R real zeros, then there are $2^{C+R}-1$ other possible signals with identical magnitude spectra but different phase spectra. The signal with all of its zeros inside the unit circle is called a *minimum-phase signal* for reasons explained below. If the signal is the discrete-time impulse response of a system, then the system is said to be a *minimum-phase system or filter.* In the other extreme in which the zeros are completely outside the unit circle, the signal (or system) is called *maximum phase.* All intermediate cases are usually called *mixed phase.*

A little thought about the general relationship between the zero configuration and the phase spectrum (i.e., think about how one deduces a phase plot from the pole–zero diagram) will convince the reader that having all the zeros inside the unit circle will minimize the absolute value of negative phase at a given ω. Conversely, having the zeros outside

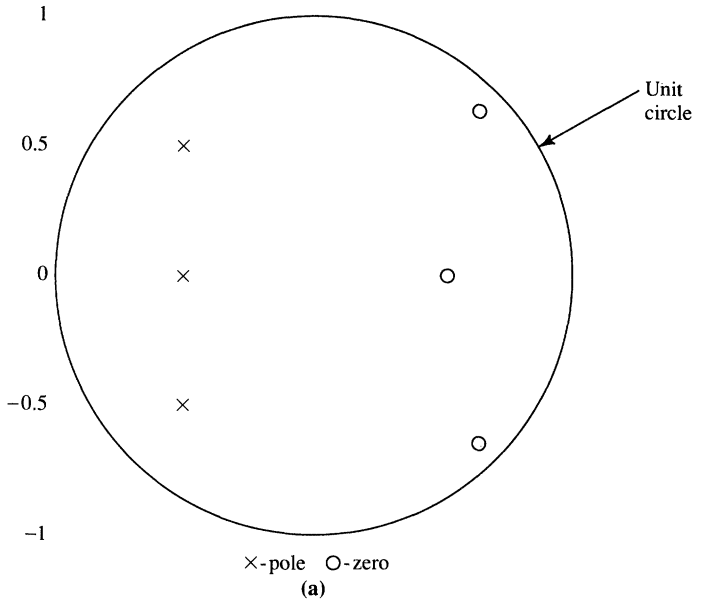


(a)



(b)

FIGURE 1.7. Phase spectra for the signals (a) $x_1(n)$ and (b) $x_2(n)$.



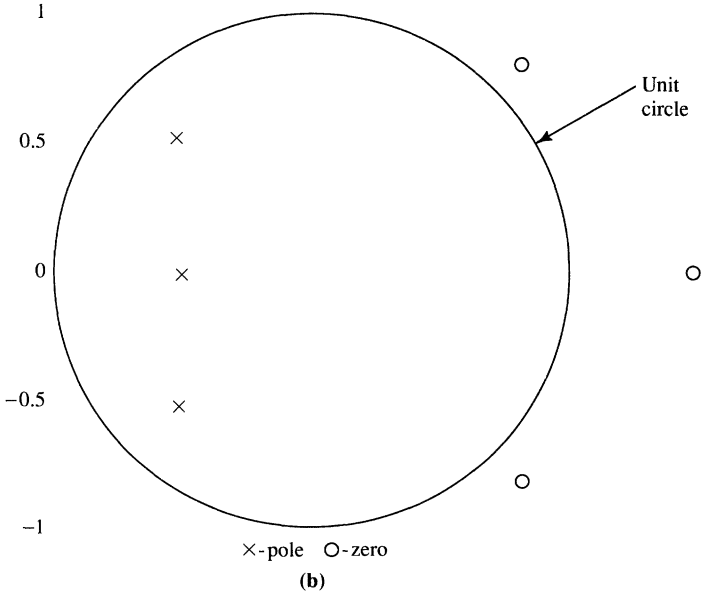


FIGURE 1.8. Pole-zero diagrams for the signals (a) $x_1(n)$ and (b) $x_2(n)$.

the unit circle will maximize the negative phase (see Figs. 1.7 and 1.8). Hence the names minimum and maximum phase are reasonable from this point of view. A more intuitive notion is to found in the time domain, however.

Since the (negative) phase at ω is directly related to the amount of temporal delay of a narrowband component at that frequency, we can infer that the minimum-phase signal is the one which, for a given magnitude spectrum, has a minimum delay of each frequency component in the spectrum. The minimum-phase signal will therefore have the highest concentration of energy near time $n = 0$ of any signal with the same magnitude spectrum. Specifically, if $x_{\min}(n)$ is the minimum-phase signal, and $E_x(m)$ represents the energy in any sequence $x(n)$ in the interval $n \in [0, m]$,

$$E_x(m) \stackrel{\text{def}}{=} \sum_{n=0}^{m} x^2(n), \tag{1.62}$$

then it will be true that[14]

$$E_{x_{\min}}(m) \geq E_x(m) \tag{1.63}$$

for any absolutely summable signal $x(n)$ with the same magnitude spectrum, and for any m. Precisely the opposite holds for the maximum-phase signal, say $x_{\max}(n)$,

$$E_{x_{\max}}(m) \leq E_x(m) \tag{1.64}$$

for any absolutely summable signal $x(n)$ with the same magnitude spectrum, and for any m. The significance of these expressions can be appreciated in Fig. 1.9, where we show the time domain waveforms for $x_1(n)$ above, which we now know is minimum phase, and for $x_2(n)$, which is maximum phase.

Yet another way to view a minimum-phase signal, particularly when it represents the impulse response of a system, is as follows: If $h(n)$ represents a minimum-phase impulse response of a causal stable system, then the z-domain system function, $H(z)$, will have all of its poles and zeros inside the unit circle. Hence there exists a causal, stable *inverse system*, $H^{-1}(z)$, such that

$$H(z)H^{-1}(z) = 1 \tag{1.65}$$

everywhere in the z-plane. If there were even one zero outside the unit circle in $H(z)$, a stable inverse would not exist, since at least one pole in the inverse would be obliged to be outside the unit circle. The existence of a causal stable inverse z-transform for $H(z)$ is therefore a sufficient condition to assure that the signal $h(n)$ (or its corresponding system) is minimum phase.

[14]A proof of this fact is outlined in Problem 5.36 of (Oppenheim and Schafer, 1989).

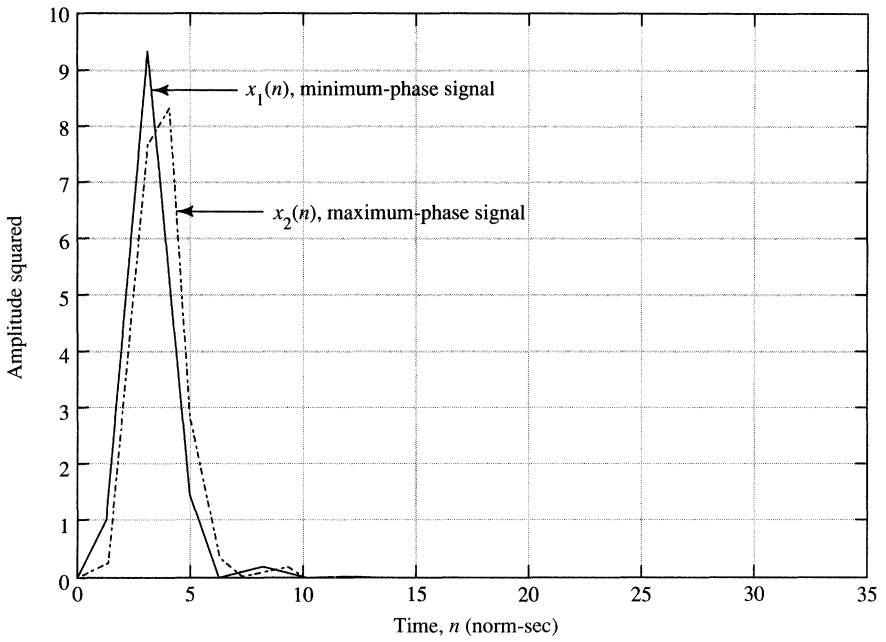


FIGURE 1.9. Time domain plots of minimum-phase signal $x_1(n)$ and maximum-phase signal $x_2(n)$. The signals are squared for convenience.

Finally, we note that we have assumed that signals in this discussion are generally infinite in duration by allowing them to have poles in their z-transforms. (By restricting our discussion to absolutely summable signals, however, we have constrained the poles to be inside the unit circle.) In the case of a real, finite duration ("all zero"), minimum-phase sequence of length N (perhaps the impulse response of an FIR filter), it can be shown that its maximum-phase counterpart is given by

$$x_{\max}(n) = x_{\min}(N-1-n) \tag{1.66}$$

or

$$X_{\max}(z) = z^{-(N-1)} X_{\min}(z^{-1}). \tag{1.67}$$

The concepts of minimum-phase signals and systems will play a key role in the theory of linear prediction and surrounding modeling concepts.

1.2 Review of Probability and Stochastic Processes

We will discover in the next chapter that there are two basic classes of speech sounds, "voiced" and "unvoiced." Generally speaking, the former is characterized by deterministic acoustic waveforms, while the latter cor-

responds to stochastic waveforms. The difference can be heard in the two sounds present in the word "it," for example. Although random process theory will be necessary to analyze unvoiced signals, we will find that even in the case of voiced sounds it will be very useful to employ analytical techniques which are fundamentally motivated by stochastic process theory, notably the autocorrelation function. In different ways from those used to analyze speech waveforms, we will employ concepts from probability in our study of stochastic models for the coding and recognition of speech. In these and other aspects of our study of speech processing, basic concepts from random process theory will be prerequisite to our pursuits.

As is the case with digital signal processing concepts, it will be necessary for the reader to have a working knowledge of the concepts of probability and stochastic processes, at least at the level of a typical senior or entry-level graduate course. Some of the widely used books in the field are listed in Appendix 1.B, and the reader is encouraged to refer to these textbooks to review concepts as needed.

As noted, one of the central tools of speech processing is the autocorrelation sequence. Several of the more fundamental concepts, in particular stationarity and ergodicity, also play key roles in our work. In the recognition domain, an understanding of basic concepts concerning joint experiments and statistical independence will be essential. It is our purpose here to briefly review these fundamental notions with the autocorrelation sequence and surrounding ideas as a target of this discussion. We will focus on discrete time random processes because of the nature of our application. As was the case in our DSP review, a second objective is to set forth notation for the remainder of the book. This short section is not intended to substitute for a solid course in random processes and will not provide an adequate background for a deep understanding of the stochastic aspects of speech or general signal processing.

1.2.1 Probability Spaces

The science of probability is customarily introduced to engineering students using an axiomatic approach for the sake of mathematical generality and formality. In this context, a formal definition of probability involves the specification of a *sample space*, a *field* or *algebra* of events, and a *probability measure*, which is assumed to conform to some basic axioms. The sample space, say $\mathcal{S}$, is the set of all outcomes of an experiment, plus the null outcome. Each element of $\mathcal{S}$ is called a *sample point*. Collections of sample points (connected by an OR condition) are called *events*. An event may consist of a single sample point.

Although the second component of the probability space is critical to theoretical developments, it is usually of least concern in typical engineering applications. Generally, it is necessary to give some careful thought to which events are to be assigned probabilities. In certain cases, we cannot assign probabilities all possible events, nor can we have too

few events, and still have a consistent and meaningful theory of probability. A proper "event space" will turn out to be a *sigma-field* or *sigma-algebra* over $\mathcal{S}$, which is a set of subsets of $\mathcal{S}$ that is closed under complementation, union, and (if $\mathcal{S}$ has an infinite number of elements) countable union. Let us call the algebra $\mathcal{A}$. In typical engineering problems, the algebra of events is often all intervals in some continuum of possible outcomes, or the "power set" of discrete outcomes if $\mathcal{S}$ is finite and discrete. These and other algebras in different situations are naturally used in problems without much forethought.

The third component, probability, is a normalized measure assigned to these "well thought out" sets of events that adheres to four basic axioms. If $P(A)$ denotes the probability of event A, these are

1. $P(\mathcal{S}) = 1$.
2. $P(A) \geq 0$, for all $A \in \mathcal{A}$.
3. For two *mutually exclusive* events A, $B \in \mathcal{A}$,
$$P(A \cup B) = P(A) + P(B). \tag{1.68}$$
Mutually exclusive means $A \cap B = \emptyset$, where $\emptyset$ is the null event.
4. For a *countably infinite* set of mutually exclusive events $A_i \in \mathcal{A}$, $i = 1, 2, \ldots,$
$$P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i). \tag{1.69}$$

The first three axioms are very intuitive and reasonable, and indeed are all that are necessary when $\mathcal{A}$ contains a finite set of events. The fourth axiom is necessary for proving certain important convergence results when the sample space is infinite (see the textbooks in Appendix 1.B not labeled "elementary"). The probability measure assigned to an event is usually consistent with the intuitive notion of the relative frequency of occurrence of that event.

The three components of a *probability space* are sufficient to derive and define virtually all important concepts and results in probability theory. Notably, the concepts of statistical independence, and joint and conditional probability, follow from this basic formalism. The two events $A, B \in \mathcal{A}$ are said to be *statistically independent* if[15]

$$P(A \cap B) = P(A)P(B). \tag{1.70}$$

The *joint probability* of events $A, B \in \mathcal{A}$ is defined simply as $P(A \cap B)$, and the *conditional probability* of B given A is

$$P(B|A) \stackrel{\text{def}}{=} \frac{P(B \cap A)}{P(A)}. \tag{1.71}$$

[15]To conserve space in complicated expressions, later in the book we will begin to write $P(A \cap B)$ as $P(A, B)$. That is, the "AND" condition between events will be denoted by a comma. In these introductory sections, however, we use the explicit "$\cap$."

Combined Experiments

This is a good place to review the notion of a *combined experiment.* We will have need of this theory only if such experiments have independent events, so we will restrict our discussion accordingly. Formally, an "experiment" is equivalent to the probability space used to treat that experiment. For example, let experiment 1, $\mathcal{E}_1$, be associated with a probability space as follows:

$$\mathcal{E}_1 = (\mathcal{S}_1, \mathcal{A}_1, P). \tag{1.72}$$

If we wish to combine a second experiment, $\mathcal{E}_2$, we need to have a way of assigning probabilities to combined events. An example will be useful to illustrate the points.

Let $\mathcal{E}_1$ be concerned with a measurement on a speech waveform at a specified time that may take a continuum of values between 0 and 10 volts. Therefore,

$$\mathcal{S}_1 = \{x : 0 \leq x \leq 10\}. \tag{1.73}$$

The events to which we will assign probabilities consist of all open and closed intervals on this range. Therefore,

$$\mathcal{A}_1 = \{x : x \in (a, b) \text{ or } (a, b] \text{ or } [a, b) \text{ or } [a, b], \text{ where } 0 \leq a \leq b \leq 10\}. \tag{1.74}$$

A second experiment, $\mathcal{E}_2$, consists of a second measurement at a later time, which is assumed to be independent of the first. The voltage in this case ranges from -30 to $+30$ volts, so

$$\mathcal{S}_2 = \{y : -30 \leq y \leq 30\} \tag{1.75}$$

and $\mathcal{A}_2$ will again consist of open and closed intervals in $\mathcal{S}_2$,

$$\begin{aligned} \mathcal{A}_2 = \{&y : y \in (a, b) \text{ or } (a, b] \text{ or } [a, b) \text{ or } [a, b], \\ &\text{where } -30 \leq a \leq b \leq 30\}. \end{aligned} \tag{1.76}$$

Now suppose that we want to assign probabilities to joint events such as

$$\begin{aligned} C &= (\text{Event } A \text{ from } \mathcal{E}_1 \cap \text{Event } B \text{ from } \mathcal{E}_2) \\ &= (1 \leq x < 5 \cap 15 < y < 25). \end{aligned} \tag{1.77}$$

In this case we simply form a combined experiment or combined probability space that involves a *product sample space* and *product algebra of events,*

$$\mathcal{E} = \mathcal{E}_1 \times \mathcal{E}_2 = (\mathcal{S}, \mathcal{A}, P) = (\mathcal{S}_1 \times \mathcal{S}_2, \mathcal{A}_1 \times \mathcal{A}_2, P). \tag{1.78}$$

This is illustrated in Fig. 1.10. Formally, the event $C \in \mathcal{A}$ is formed by intersecting events $A \times \mathcal{S}_2$ (also in $\mathcal{A}$) with $B \times \mathcal{S}_1$ (also in $\mathcal{A}$) to get

$$C = (A \times \mathcal{S}_2) \cap (B \times \mathcal{S}_1). \tag{1.79}$$

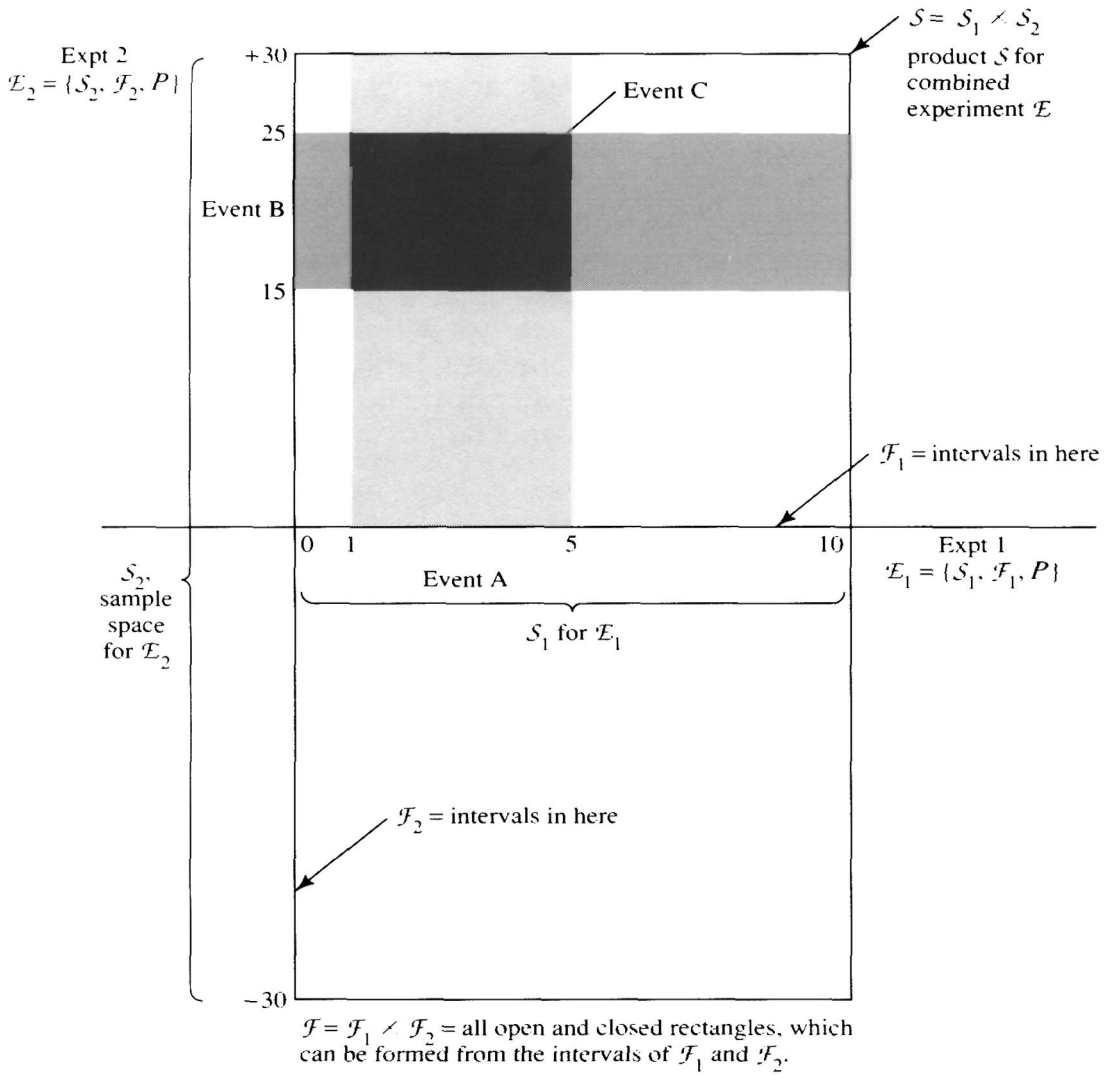


FIGURE 1.10. Combined probability space.

The probability assigned to C will be

$$P(C) = P(A)P(B), \tag{1.80}$$

since we are assuming A and B to be independent. These ideas are easily extended to more than two experiments (see textbooks in Appendix 1.B).

1.2.2 Random Variables

Single Random Variables

Note: We henceforth use a simple comma to indicate the AND *condition between two events in the argument of a probability. For example,* $P(A \cap B)$ will be written $P(A, B)$.

Definition of a Random Variable. A (real) *random variable* is the mapping of the sample points in a sample space of an experiment to the real number line. For example, the sample space, $\mathcal{S}$, might be the set of all cities on Earth, and the random variable, say $\underline{x}$, a mapping of the city to its metropolitan population in millions in 1990:

$$\underline{x}(\text{Chicago}) = 6.1. \tag{1.81}$$

Throughout the book, we will follow the convention established here of underscoring a quantity to distinguish it as a random variable (or random vector as discussed below).[16] Later we will use a similar notation to indicate a random *process*. In this context, a lowercase letter which is not underscored is used to indicate, in the abstract, values of the mapping. For example, if the variable σ represents points in $\mathcal{S}$, then we might write something like

$$\underline{x}(\sigma) = x \tag{1.82}$$

to indicate that the random variable $\underline{x}$ maps the outcome σ to the real value x. Note that it is the mapping itself which is the random variable, not the value of the mapping. Nevertheless, we often say "the random variable is 4," when we mean "the random variable has produced a value 4."

It is customary to employ some notation like $\mathcal{S}_{\underline{x}}$ to indicate the *range space* of $\underline{x}$, the intervals and/or points on the real line which constitute the range of the mapping $\underline{x}$. It is also formally convenient to consider an algebra of events in $\mathcal{S}_{\underline{x}}$, say $\mathcal{A}_{\underline{x}}$, to which we might want to assign probabilities.

There are certain conditions that must be met for $\underline{x}$ to be a random variable. Each is ultimately concerned with the concept of *measurability*, a property which allows us to assign probabilities to events in $\mathcal{A}_{\underline{x}}$. Generally speaking, every event in $\mathcal{A}_{\underline{x}}$ must be traceable back to a well-defined event in the original $\mathcal{A}$ probability space so that we know what probability to assign to it. One important criterion is that $\underline{x}$ not be a one-to-many mapping. (A deeper discussion of this issue is found in the textbooks of Appendix 1.B not labeled "elementary.")

The random variable in some engineering problems is only an abstract formality in the sense that the original outcomes of experiments are already real numbers and no mapping is actually necessary. Such will often be the case in this book where the "outcomes of experiments" will be values of a speech sequence at a given point in time. Accordingly, we will encounter no problems with measurability.

A random variable can be either *continuous*, *discrete*, or *mixed*, referring to whether the mapping produces continua of outcomes, discrete points, or a mixture of the two types. In our speech analysis work, the

[16]Some textbooks use uppercase letters to indicate random variables; still others use boldface. Uppercase letters will have many other significances in this book and boldface quantities are used to indicate vectors or matrices.

random variables will usually represent speech amplitudes; we will primarily work with the continuous case. (Remember that we usually ignore amplitude quantization in this book! The important exception will be in Chapter 7, where we actually consider the issue of quantizing speech for compression purposes.) Later in our recognition work, the random variables will be found to be primarily discrete.

Denoting Probabilities. The outcomes of random variable $\underline{x}$ (elements of $\mathcal{A}_{\underline{x}}$) are formally assigned probabilities using traceback to the original event space $\mathcal{A}$. The "events" in $\mathcal{A}_{\underline{x}}$ with which we will be concerned in this book are points $\underline{x} = x$, and intervals that may be open or closed at either end, for example, $a \leq \underline{x} < b$ or $\underline{x} > 6$. Probabilities of these events will be denoted in the expected way, such as $P(a \leq \underline{x} < b)$ or $P(\underline{x} > 6)$. The generalization to multiple random variables is obvious, for example, $P(\underline{x} \leq \alpha, \beta < \underline{y} \leq \gamma)$. A nuance occurs when describing the probabilities of *point* outcomes. When it is obvious which random variable(s) is (are) involved, we may write $P(x)$ instead of $P(\underline{x} = x)$, or $P(x, y)$ instead of $P(\underline{x} = x, \underline{y} = y)$. Frequently, for absolute clarity, we will retain the random variable in the argument. If the reader finds this unnecessary in certain cases, then he or she should simply use the abbreviated form in any notes or solutions.

As noted above, when a random variable can take only a countable number of values, it is called a *discrete* random variable.[17] In this case the statistical description of, say $\underline{x}$, is modeled entirely by its *probability distribution* $P(\underline{x} = x_i)$, $i = 1, 2, \ldots$. Occasionally, we will want to refer to the probability distribution of $\underline{x}$ in general, and we will write simply[18] $P(\underline{x})$.

cdf and pdf. Associated with a random variable, $\underline{x}$, is a *cumulative distribution function* (cdf), say $F_{\underline{x}}(x)$, defined as

$$F_{\underline{x}}(x) \stackrel{\text{def}}{=} P(\underline{x} \leq x), \tag{1.83}$$

where $P(\underline{x} \leq x)$ means the probability that the random variable $\underline{x}$ produces a value less than or equal to x. Of more use to us is the *probability density function* (pdf),

$$f_{\underline{x}}(x) \stackrel{\text{def}}{=} \frac{d}{dx} F_{\underline{x}}(x). \tag{1.84}$$

We use the derivative in the "engineering" sense in which discontinuities in $F_{\underline{x}}(x)$ (caused by discrete points with nonzero probability) produce impulses in $f_{\underline{x}}(x)$. The continuous part of $F_{\underline{x}}(x)$ might not be differentiable in certain cases which will not concern us [e.g., see (Wong and Hajek,

[17]Carefully note that this term has nothing whatsoever to do with discrete *time*.

[18]A notation which is more consistent with $f_{\underline{x}}(x)$ would be $P_{\underline{x}}(x)$, but this has other obvious disadvantages. For example, how would we denote the probability of the event $\underline{x} \geq x$?

1984, Ch. 1) for details]. Note that a discrete random variable will have a pdf that will consist entirely of impulses at the point outcomes of the random variable. The weighting on the impulse at x_i is $P(\underline{x} = x_i)$.

Returning to (1.84), from the Fundamental Theorem of Calculus, we have

$$P(a < \underline{x} \leq b) = F_{\underline{x}}(b) - F_{\underline{x}}(a) = \int_a^b f_{\underline{x}}(\xi)\, d\xi, \tag{1.85}$$

implying the well-known result that the area under the pdf on the range $(a, b]$ yields the probability that $\underline{x}$ produces a value in that interval.[19]

Some of the commonly used pdf's in speech processing are

1. *Gaussian*:

$$f_{\underline{x}}(x) = \frac{1}{\sqrt{2\pi\sigma_{\underline{x}}^2}} \exp\left\{-\frac{(x-\mu_{\underline{x}})^2}{2\sigma_{\underline{x}}^2}\right\}, \tag{1.86}$$

where $\mu_{\underline{x}}$ is the *average* or *mean* of $\underline{x}$, and $\sigma_{\underline{x}}^2$ is the *variance*, or $\sigma_{\underline{x}}$ is the *standard deviation* (discussed below).

2. *Uniform*:

$$f_{\underline{x}}(x) = \begin{cases} \dfrac{1}{b-a}, & a \leq x < b \\ 0, & \text{otherwise} \end{cases} \tag{1.87}$$

for some $b > a$.

3. *Laplacian*:

$$f_{\underline{x}}(x) = \frac{1}{\sqrt{2\sigma_{\underline{x}}^2}} \exp\left\{-\frac{\sqrt{2}\,|x|}{\sigma_{\underline{x}}}\right\}, \tag{1.88}$$

where $\sigma_{\underline{x}}$ is the standard deviation of $\underline{x}$.

Finally, let us recall the meaning of the *conditional cdf* and *conditional pdf*, which are just natural extensions of the theory,

$$F_{\underline{x}}(x|D) \stackrel{\text{def}}{=} P(\underline{x} \leq x|D) = \frac{P(\underline{x} \leq x, D)}{P(D)} \tag{1.89}$$

and

$$f_{\underline{x}}(x|D) \stackrel{\text{def}}{=} \frac{d}{dx} F_{\underline{x}}(x|D), \tag{1.90}$$

where D is any outcome (event or point) of $\underline{x}$ of nonzero probability.

[19]Care must be taken with impulse functions at the limits of integration if they exist.

Multiple Random Variables

Preliminaries. We are gradually building toward a review of random processes. The next step is to consider relationships among several random variables. We begin by considering relationships between two random variables, noting that many of the concepts we review here have natural generalizations to more than two random variables. At the end of the section, we focus on random *vectors* in which some of these generalizations will arise.

In combining experiments above, we encountered the task of combining two sample spaces at the fundamental level. We assumed that the events in the individual sample spaces were independent. Here we implicitly assume that two random variables, say $\underline{x}$ and $\underline{y}$, map the same $\mathcal{S}$ into two different range spaces, $\mathcal{S}_{\underline{x}}$ and $\mathcal{S}_{\underline{y}}$. The joint range space is simply a product space,

$$\mathcal{S}_{\underline{xy}} = \mathcal{S}_{\underline{x}} \times \mathcal{S}_{\underline{y}}, \tag{1.91}$$

formed in a similar manner to product sample spaces for combined experiments. The joint event algebra, say $\mathcal{A}_{\underline{xy}}$, are events chosen from $\mathcal{S}_{\underline{xy}}$. For most purposes, these will be open and closed rectangles and points in $\mathcal{S}_{\underline{xy}}$. A significant difference between this theory and that of combined experiments is that we do not assume that events in the individual range spaces, $\mathcal{S}_{\underline{x}}$ and $\mathcal{S}_{\underline{y}}$, are independent. We formally assign probabilities to events in $\mathcal{A}_{\underline{xy}}$ by tracing them back to $\mathcal{A}$ to see what event they represent there.

These ideas are readily extended to more than two random variables.

Joint cdf and pdf; Conditional Probability. The *joint cdf* and *joint pdf* are defined formally as

$$F_{\underline{xy}}(x, y) \stackrel{\text{def}}{=} P(\underline{x} \leq x, \underline{y} \leq y) \tag{1.92}$$

and

$$f_{\underline{xy}}(x, y) = \frac{\partial^2}{\partial x \partial y} F_{\underline{xy}}(x, y), \tag{1.93}$$

respectively. Some properties of the these functions are studied in the problems at the end of the chapter. A prevalent joint pdf in engineering is the *joint Gaussian*,

$$f_{\underline{xy}}(x, y) = \frac{1}{2\pi\sigma_{\underline{x}}\sigma_{\underline{y}}\sqrt{1-\rho_{\underline{xy}}^2}} \exp\left\{-\frac{1}{2}Q(x, y)\right\}, \tag{1.94}$$

where

$$Q(x, y) = \frac{1}{1-\rho_{\underline{x}\underline{y}}^2} \left\{ \left(\frac{x-\mu_{\underline{x}}}{\sigma_{\underline{y}}} \right)^2 - 2\rho_{\underline{x}\underline{y}} \left(\frac{x-\mu_{\underline{x}}}{\sigma_{\underline{x}}} \right) \left(\frac{y-\mu_{\underline{y}}}{\sigma_{\underline{y}}} \right) + \left(\frac{y-\mu_{\underline{y}}}{\sigma_{\underline{y}}} \right)^2 \right\}, \tag{1.95}$$

in which $\mu_{\underline{x}}$ and $\mu_{\underline{y}}$ are the means of $\underline{x}$ and $\underline{y}$, $\sigma_{\underline{x}}$ and $\sigma_{\underline{y}}$ are the standard deviations, and $\rho_{\underline{x}\underline{y}}$ is the correlation coefficient. These quantities are special moments, which are reviewed below.

The *conditional probability* of event $A \in \mathcal{A}_{\underline{x}}$ given the occurrence of event $B \in \mathcal{A}_{\underline{y}}$ is defined in the usual way,

$$P(A \in \mathcal{A}_{\underline{x}} | B \in \mathcal{A}_{\underline{y}}) \stackrel{\text{def}}{=} \frac{P(A, B)}{P(B)}, \tag{1.96}$$

where the numerator and denominator are determined by traceback to $\mathcal{A}$. If A is the interval $\underline{x} \leq x$, then we have the *conditional cdf*,

$$F_{\underline{x}|\underline{y}}(x|B) \stackrel{\text{def}}{=} P([\underline{x} \leq x] \in \mathcal{A}_{\underline{x}} | B \in \mathcal{S}_{\underline{y}}) = \frac{P(x \leq x, B)}{P(B)} \tag{1.97}$$

and the *conditional pdf*,

$$f_{\underline{x}|\underline{y}}(x|B) \stackrel{\text{def}}{=} \frac{d}{dx} F_{\underline{x}}(x|B). \tag{1.98}$$

All the usual relationships between the cdf and pdf hold with the conditioning information added. For example,

$$F_{\underline{x}|\underline{y}}(x_2|B) - F_{\underline{x}|\underline{y}}(x_1|B) = \int_{x_1}^{x_2} f_{\underline{x}|\underline{y}}(\xi|B)\, d\xi. \tag{1.99}$$

Independence. Two random variables, $\underline{x}$ and $\underline{y}$, are *statistically independent* if and only if for any two events, $A \in \mathcal{A}_{\underline{x}}$ and $B \in \mathcal{A}_{\underline{y}}$,

$$P(A, B) = P(A)P(B). \tag{1.100}$$

It follows immediately for two statistically independent random variables that

$$F_{\underline{x}\underline{y}}(x, y) = F_{\underline{x}}(x) F_{\underline{y}}(y) \tag{1.101}$$

and

$$f_{\underline{x}\underline{y}}(x, y) = f_{\underline{x}}(x) f_{\underline{y}}(y). \tag{1.102}$$

Statistical independence is a very strong condition. It says that outcomes of $\underline{x}$ and $\underline{y}$ tend not to be related in *any* functional way, linear or nonlinear. When two random variables are related linearly, we say that they are *correlated*. To say that $\underline{x}$ and $\underline{y}$ are uncorrelated is to say that

there is no *linear* dependence between them. This does not say that they are necessarily statistically independent, for there can still be nonlinear dependence between them. We will see this issue in the topic of vector quantization (Section 7.2.2), where there will be an effort made to extract not only linear dependency (correlation) out of the speech data, but also nonlinear dependency, to produce efficient coding procedures.

Expectation and Moments. The *statistical expectation* or *statistical average* of a scalar function of a random variable, say $g(\underline{x})$, is defined by

$$\mathcal{E}\{g(\underline{x})\} \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} g(x) f_{\underline{x}}(x)\, dx \tag{1.103}$$

assuming that the pdf exists. When $g(\underline{x}) = \underline{x}$, this produces the *average* or *mean value* of $\underline{x}$, $\mu_{\underline{x}}$. Note that when $\underline{x}$ produces only discrete values, say $x_1, x_2, \ldots,$ then the pdf consists of impulses and the definition produces

$$\mathcal{E}\{g(\underline{x})\} = \sum_{i=1}^{\infty} x_i P(\underline{x} = x_i). \tag{1.104}$$

The definition is readily generalized to functions of two or more random variables. For example,

$$\mathcal{E}\{g(\underline{x}, \underline{y})\} \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) f_{\underline{x}\underline{y}}(x, y)\, dx\, dy. \tag{1.105}$$

Particularly useful averages are the *moments* of a random variable. The ith *moment* of the random variable $\underline{x}$ is the number

$$\mathcal{E}\{\underline{x}^i\} = \int_{-\infty}^{\infty} x^i f_{\underline{x}}(x)\, dx. \tag{1.106}$$

Obviously, the first moment is the mean of $\underline{x}$, $\mu_{\underline{x}}$. The ith *central moment* of the random variable $\underline{x}$ is the number

$$\mathcal{E}\{(\underline{x} - \mu_{\underline{x}})^i\} = \int_{-\infty}^{\infty} (x - \mu_{\underline{x}})^i f_{\underline{x}}(x)\, dx. \tag{1.107}$$

A special central moment is the second one ($i = 2$), which we call the *variance* and denote $\sigma_{\underline{x}}^2$. The square root of the variance, $\sigma_{\underline{x}}$, is called the *standard deviation* of $\underline{x}$.

The i, k *joint moment* between random variables $\underline{x}$ and $\underline{y}$ is the number

$$\mathcal{E}\{\underline{x}^i \underline{y}^k\} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^i y^k f_{\underline{x}\underline{y}}(x, y)\, dx\, dy \tag{1.108}$$

and the i,k *joint central moment* is the number

$$\mathcal{E}\{(\underline{x}-\mu_{\underline{x}})^i(\underline{y}-\mu_{\underline{y}})^k\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x-\mu_{\underline{x}})^i(y-\mu_{\underline{y}})^k f_{\underline{xy}}(x,y)\,dx\,dy. \tag{1.109}$$

When $i=k=1$, the joint moment is called the *correlation* between $\underline{x}$ and $\underline{y}$, and the joint central moment, the *covariance*. Let us call these numbers $r_{\underline{xy}}$ and $c_{\underline{xy}}$, respectively. A parameter frequently used in the statistical analysis of data (and which appears in the joint Gaussian pdf above) is the *correlation coefficient* given by

$$\rho_{\underline{xy}} = \frac{c_{\underline{xy}}}{\sigma_{\underline{x}}\sigma_{\underline{y}}}. \tag{1.110}$$

We see that the correlation coefficient is the covariance between $\underline{x}$ and $\underline{y}$ normalized to the product of the individual standard deviations.

Correlation and covariance will occur repeatedly in our study of speech, and it is advisable to master their meanings if they are not already very familiar. This is especially true because the terms "autocorrelation" and "covariance" are used in ways that are not consistent with their definitions in some aspects of speech processing. A related pair of somewhat unfortunate terms[20] is the following: $\underline{x}$ and $\underline{y}$ are said to be *orthogonal* if their correlation is zero, and *uncorrelated* if their covariance is zero. Finally, we note that the covariance and correlation are related as

$$c_{\underline{xy}} = r_{\underline{xy}} - \mu_{\underline{x}}\mu_{\underline{y}}. \tag{1.111}$$

The *conditional expectation* of $\underline{y}$, given some event related to random variable $\underline{x}$, say $B \in \mathcal{A}_{\underline{x}}$, is defined as

$$\mathcal{E}\{\underline{y}|B\} \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} y f_{\underline{y}|\underline{x}}(y|B)\,dy. \tag{1.112}$$

It is well known that the best predictor of $\underline{y}$, in the sense of least square error, given some event concerning $\underline{x}$ is given by the conditional expectation. If $\underline{x}$ and $\underline{y}$ are also joint Gaussian, then the conditional expectation also provides the *linear* least square error predictor (see textbooks in Appendix 1.B).

Random Vectors. In discussing more than one random variable at a time, say $\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_N$, it is frequently convenient to package them into a *random vector*,

$$\underline{\mathbf{x}} \stackrel{\text{def}}{=} [\underline{x}_1 \underline{x}_2 \cdots \underline{x}_N]^T. \tag{1.113}$$

[20]Speech processing engineers are not responsible for this terminology!

Note that the vector is indicated by a boldface quantity, and the fact that it is a *random* vector is indicated by the line beneath it. The pdf associated with a random vector is very simply the joint pdf among its component random variables,

$$f_{\underline{\mathbf{x}}}(x_1, x_2, \ldots, x_N) \stackrel{\text{def}}{=} f_{\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_N}(x_1, x_2, \ldots, x_N). \tag{1.114}$$

Operations among random vectors follow the usual rules of matrix arithmetic. For example, the operations of inner and outer products of random vectors will be significant in our work. Recall that the *inner product* or l_2 *norm* of a vector (in this case a *random* vector, say $\underline{\mathbf{x}} = [\underline{x}_1 \cdots \underline{x}_N]^T$), is the sum of its squared components. This can be written in a variety of ways,

$$\|\underline{\mathbf{x}}\|^2 = \underline{\mathbf{x}}^T \underline{\mathbf{x}} = \sum_{i=1}^{N} \underline{x}_i^2. \tag{1.115}$$

Note that the inner product of a random vector is itself a *random variable.* The *outer product*, on the other hand, is the product $\underline{\mathbf{x}}\underline{\mathbf{x}}^T$ which creates a *random matrix* whose (i,j) element is the random variable $\underline{x}_i \underline{x}_j$. Of course, the inner and outer products may be computed between two different random vectors.

The *expectation* of a random vector (or matrix) is just the vector (or matrix) of expectations of the individual elements. For example, $\mathcal{E}\{\underline{\mathbf{x}}\}$ is simply the vector of means $[\mu_{\underline{x}_1} \cdots \mu_{\underline{x}_N}]^T$, which we might denote $\boldsymbol{\mu}_{\underline{\mathbf{x}}}$. Another important example is the expectation of the outer product,

$$\mathbf{R}_{\underline{\mathbf{x}}} \stackrel{\text{def}}{=} \mathcal{E}\{\underline{\mathbf{x}}\underline{\mathbf{x}}^T\}, \tag{1.116}$$

which is called the *autocorrelation matrix* for the random vector $\underline{\mathbf{x}}$, since its (i,j) element is the correlation between random variables $\underline{x}_i$ and $\underline{x}_j$. The matrix

$$\mathbf{C}_{\underline{\mathbf{x}}} \stackrel{\text{def}}{=} \mathcal{E}\{(\underline{\mathbf{x}} - \boldsymbol{\mu}_{\underline{\mathbf{x}}})(\underline{\mathbf{x}} - \boldsymbol{\mu}_{\underline{\mathbf{x}}})^T\} \tag{1.117}$$

is called the *covariance matrix* for $\underline{\mathbf{x}}$ for the similar reason.

An example that occurs frequently in engineering problems is the *Gaussian random vector* for which any subset of its random variable components has a joint Gaussian pdf. In particular, the joint pdf among the entire set of N is an N-dimensional Gaussian pdf. That is, if $\underline{\mathbf{x}}$ is a Gaussian random vector, then

$$\begin{aligned} f_{\underline{\mathbf{x}}}(x_1, \ldots, x_N) &= f_{\underline{x}_1, \ldots, \underline{x}_N}(x_1, \ldots, x_N) \\ &= \frac{1}{(2\pi)^{N/2} \sqrt{\det \mathbf{C}_{\underline{\mathbf{x}}}}} \exp\left\{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_{\underline{\mathbf{x}}})^T \mathbf{C}_{\underline{\mathbf{x}}}^{-1}(\mathbf{x} - \boldsymbol{\mu}_{\underline{\mathbf{x}}})\right\}, \end{aligned} \tag{1.118}$$

where $\mathbf{x}$ denotes the vector of arguments $[x_1 \cdots x_N]^T$ and $\mathbf{C}_{\underline{\mathbf{x}}}$ and $\boldsymbol{\mu}_{\underline{\mathbf{x}}}$ are the covariance matrix and mean vector as defined above. It can be shown that this form reduces to (1.94) in the two-dimensional case.

1.2.3 Random Processes

Basic Concepts

Definition of a Random Process. A (real) discrete[21] *random*, or *stochastic*, *process* is defined as a collection of random variables, each indexed by a point in discrete time. For example, the following set comprises a random process:

$$\{\ldots, \underline{x}(-1), \underline{x}(0), \underline{x}(1), \ldots\} = \{\underline{x}(n), \quad n \in (-\infty, \infty)\}, \tag{1.119}$$

where each random variable represents a model for the generation of values at its corresponding time. There will be many occasions when we will want to refer to a random process by a name. For example, it is too clumsy to write something like "the random process $\{\ldots, \underline{x}(-1), \underline{x}(0), \underline{x}(1), \ldots\}$ is used to model the speech signal. . . ." What shall we call the random process? This is one place where we shall bow to convention and use a less-than-ideal choice. It is common to refer to a random process by the same name as that used for the random variables which constitute it. For example, the random process in (1.119) would be called simply $\underline{x}$,

$$\underline{x} = \{\ldots, \underline{x}(-1), \underline{x}(0), \underline{x}(1), \ldots\} = \{\underline{x}(n), \quad n \in (-\infty, \infty)\}, \tag{1.120}$$

so that we can write "the random process $\underline{x}$ is used to model the speech signal. . . ." Of course, the problem which arises is that $\underline{x}$ may refer to a random variable or a random process. We could further distinguish a random process by using yet another notation, for example, $\underline{\underline{x}}$, but this will turn out to be unnecessary in almost every circumstance. From context, it should always be clear whether an underscored quantity is a random variable or a random process. Note carefully that the notation $\underline{x}$ *never* refers to both. Once it is known that $\underline{x}$ is a random process, then all associated random variables should have time indices, for example, $\underline{x}(n)$. Finally, it should be noted that the random variables in a random process will almost always be indexed by integers in parentheses to indicate their association with discrete time. There will be only limited use for continuous-time random processes in this book.

An example will illustrate how a random process is related to a physical problem. Suppose that we define a simple experiment in which an integer representing one of L digitized speech waveforms is selected at random. For illustrative purposes, we plot segments of all of the waveforms (for $L = 3$) in Fig. 1.11. We can imagine that each time is governed by a random variable, say $\underline{x}(n)$ at time n, and the ordered collection of these random variables is the underlying random process, $\underline{x}$. When the experiment is complete, each random variable will go to work mapping the outcome, for example, "waveform 2," to an amplitude level corresponding to that outcome. For example, $\underline{x}(8)$ maps the outcome "waveform 2" to a value 82 in our figure. For this one experiment, therefore, the totality of all the

[21] We will focus on the discrete case because of our primary interest in discrete signals in this book.

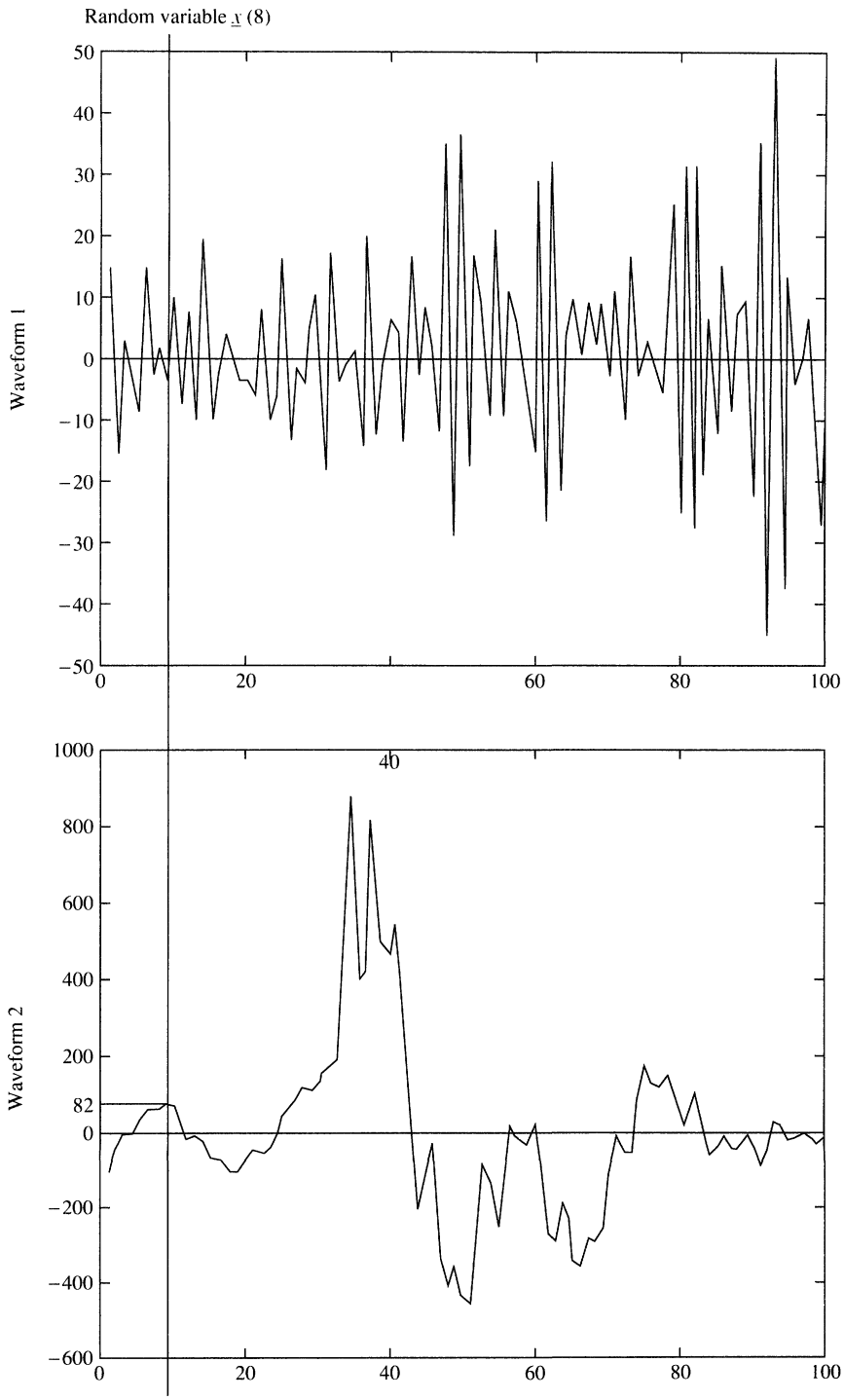


FIGURE 1.11. An ensemble of speech waveforms modeled by random process $\underline{x}$ with random variables $\underline{x}(n)$. (Figure continued on p. 44.)

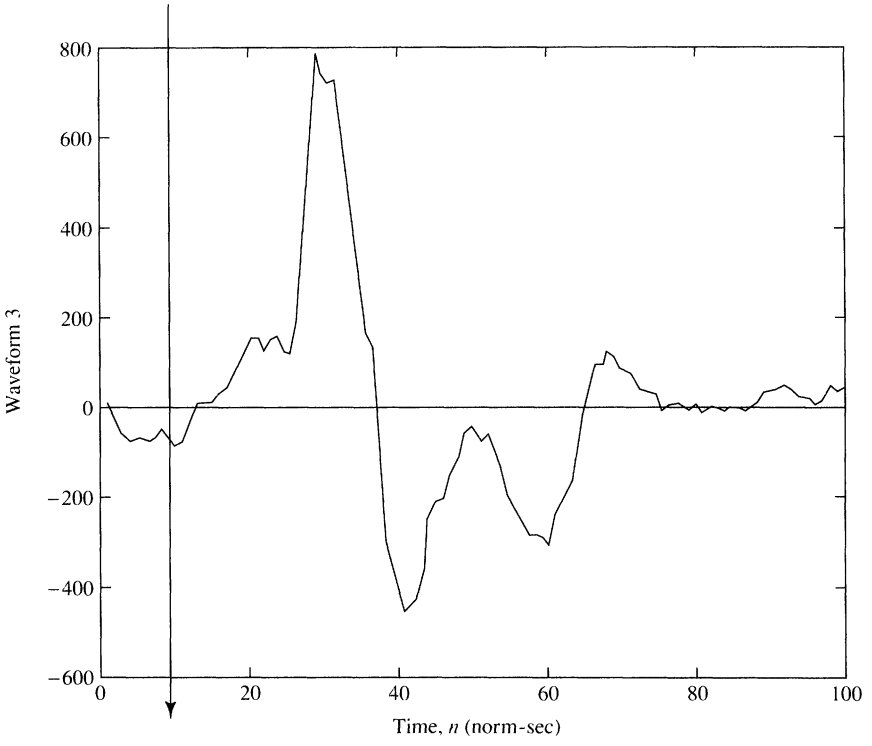


FIGURE 1.11. (continued)

random variables will produce a particular waveform from the experimental outcome, each random variable being responsible for one point. This one waveform is called a *sample function* or *realization* of the random process. The collection of all realizations (resulting from all the experiments) is called an *ensemble.* It should be clear that if we select a time, we will get a random variable. If we select an experimental outcome, we get a realization. If we select both a time and an outcome, we get a *number,* which is the result of the mapping of that outcome to the real line by the random variable at the time we select.

pdf for a Random Process. Associated with any i random variables in a random process is an ith order pdf. For example, for $\underline{x}(n_1)$, $\underline{x}(n_2)$, and $\underline{x}(n_3)$, we have the third-order density

$$f_{\underline{x}(n_1),\underline{x}(n_2),\underline{x}(n_3)}(\xi_1, \xi_2, \xi_3). \tag{1.121}$$

This is consistent with our previous convention of listing all random variables in the joint pdf as subscripts of f.

Independence of Random Processes. We have reviewed the meaning of independence of random variables above. We must also recall the meaning

of independent random processes. Two random processes, $\underline{x}$ and $\underline{y}$, are *statistically independent* if, for any times $n_1, n_2, \ldots, n_i$, and $m_1, m_2, \ldots, m_j$, the random variable group $\underline{x}(n_1), \underline{x}(n_2), \ldots, \underline{x}(n_i)$ is independent of $\underline{y}(m_1), \underline{y}(m_2), \ldots, \underline{y}(m_j)$. This, in turn, requires that the joint pdf be factorable as

$$\begin{aligned} f_{\underline{x}_1(n_1),\ldots,\underline{x}_i(n_i),\underline{y}_1(m_1),\ldots,\underline{y}_j(m_j)}(\xi_1,\ldots,\xi_i,v_1,\ldots,v_j) \\ = f_{\underline{x}_1(n_1),\ldots,\underline{x}_i(n_i)}(\xi_1,\ldots,\xi_i) f_{\underline{y}_1(m_1),\ldots,\underline{y}_j(m_j)}(v_1,\ldots,v_j). \end{aligned} \tag{1.122}$$

Stationarity. A random process $\underline{x}$ is said to be *stationary to order i* or *i*th-*order stationary* if

$$f_{\underline{x}_1(n_1),\ldots,\underline{x}_i(n_i)}(\xi_1,\ldots,\xi_i) = f_{\underline{x}_1(n_1+\Delta),\ldots,\underline{x}_i(n_i+\Delta)}(\xi_1,\ldots,\xi_i) \tag{1.123}$$

for any times $n_1, n_2, \ldots, n_i$ and any Δ. This means that the joint pdf does not change if we consider any set of i random variables from $\underline{x}$ with the same relative spacings as the original set (which is arbitrary). If $\underline{x}$ is stationary to any order, then it is said to be *strict sense*, or *strong sense*, *stationary* (SSS). We will review a weaker form of stationarity below.

Stationarity has important implications for engineering analysis of a stochastic process. It implies that certain statistical properties of the process are invariant with time, making the process more amenable to modeling and analysis. Consider, for example, the case in which $\underline{x}$ is first-order stationary. Then

$$f_{\underline{x}(n)}(\xi) = f_{\underline{x}(n+\Delta)}(\xi) \tag{1.124}$$

for any n and Δ, from which it follows immediately that every random variable in $\underline{x}$ has the same mean. In this case, it is reasonable to talk about *the* average of the random process, but in general there are as many averages as random variables in a random process. This leads us to the important issue of ergodicity.

Ergodicity and Temporal Averages. Consider a random process, $\underline{x}$, known to be first-order stationary. We might find ourselves in the lab with only one realization of the process, say $x_1(n), n \in (-\infty, \infty)$, wondering whether we could somehow estimate *the* average of $\underline{x}$, say $\mu_{\underline{x}}$. In principle, we should acquire a large number of realizations and use them to compute an empirical average (estimate) of any random variable, say $\underline{x}(n)$ at time n. (It wouldn't matter which n, since the averages should all be the same due to stationarity.) This estimate, obtained by averaging down through the ensemble at a point, is referred to as an *ensemble average*. The ensemble average represents an attempt to estimate $\mathcal{E}\{\underline{x}(n)\}$ at time n, hence to estimate *the* average of process. Since we do not have an ensemble, it would be tempting to estimate $\mu_{\underline{x}}$ by computing a *temporal average* of the realization, $x_1(n)$,

$$\mu_{x_1} = \mathcal{L}\{x_1(n)\} \stackrel{\text{def}}{=} \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} x_1(n). \tag{1.125}$$

Note that we have explicitly used a *signal* name, x_1, as a subscript of μ to indicate that it has been computed using the realization rather than an ensemble. Note also the operator $\mathcal{L}$ used to indicate the *long-term time average.* This notation will be used consistently:

$$\mathcal{L}\{\cdot\} \stackrel{\text{def}}{=} \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} \{\cdot\}. \tag{1.126}$$

When will μ_{x_1}, the time average, equal $\mu_{\underline{x}}$, the ensemble or statistical average? Generally speaking, a random process is *ergodic* if ensemble averages can be replaced by time averages.[22] In our example, if $\mu_{x_1} = \mu_{\underline{x}}$, $\underline{x}$ is said to be *mean-ergodic*, since this property holds for the mean. Ergodicity will be an important assumption in our work with speech because we frequently will have only one realization with which to compute averages. In particular, second-order ergodicity will play an important role and we will look more carefully at this concept shortly.

Correlation and Covariance Applied to Random Processes

Consider two random variables, say $\underline{x}(n_1)$ and $\underline{x}(n_2)$, taken from a random process $\underline{x}$. Recall that the correlation of these two random variables is $\mathcal{E}\{\underline{x}(n_1)\underline{x}(n_2)\}$. Since the two random variables in this case are drawn from the same random process, we give this the name *autocorrelation* and feature it with a special notation

$$r_{\underline{x}}(n_1, n_2) \stackrel{\text{def}}{=} \mathcal{E}\{\underline{x}(n_1)\underline{x}(n_2)\}. \tag{1.127}$$

Similarly, the *autocovariance* function is given by

$$c_{\underline{x}}(n_1, n_2) \stackrel{\text{def}}{=} \mathcal{E}\Big\{[\underline{x}(n_1) - \mathcal{E}\{\underline{x}(n_1)\}][\underline{x}(n_2) - \mathcal{E}\{\underline{x}(n_2)\}]\Big\}. \tag{1.128}$$

It is a simple matter to show that

$$c_{\underline{x}}(n_1, n_2) = r_{\underline{x}}(n_1, n_2) - \mathcal{E}\{\underline{x}(n_1)\}\,\mathcal{E}\{\underline{x}(n_2)\}. \tag{1.129}$$

It follows immediately from the definition of stationarity that if the random process $\underline{x}$ is at least second-order stationary, then the value of the autocorrelation does not depend on *which* two random variables are selected from $\underline{x}$, but rather their *separation in time.* In this case, we adopt the somewhat sloppy, but very conventional, notation

$$\begin{aligned} r_{\underline{x}}(\eta) &\stackrel{\text{def}}{=} \text{autocorrelation of any two random variables in } \underline{x}, \\ &\qquad \text{which are separated by } \eta \text{ in time} \\ &= \mathcal{E}\{\underline{x}(n)\underline{x}(n-\eta)\} \text{ for any } n. \end{aligned} \tag{1.130}$$

[22]This definition of ergodicity is entrenched in engineering textbooks, but it is not strictly accurate [see (Gray and Davisson, 1986, Ch. 7)].

If a random process is ith-order stationary, it is also $(i-1)$th-order stationary. Therefore a second-order stationary process is also first order and has a constant mean,

$$\mu_{\underline{x}} = \mathcal{E}\{\underline{x}(n)\} \text{ for any } n. \tag{1.131}$$

This leads us to the definition of a weak form of stationarity, which is often sufficient to allow many useful engineering analyses.

A random process $\underline{x}$ is said to be *wide sense*, or *weak sense*, *stationary* (WSS) if

1. Its autocorrelation is a function of time difference only as in (1.130).
2. Its mean is constant as in (1.131).

We note that

$$\text{SSS} \Rightarrow \text{second-order stationarity} \Rightarrow \text{WSS}, \tag{1.132}$$

but neither of the implications reverses except in the special case of joint Gaussian random variables (see Problem 1.11).

Finally, but very important, note that if $\underline{x}$ is *correlation-ergodic*, then the autocorrelation can be computed using a temporal average

$$r_x(\eta) = \mathcal{L}\{x(n)x(n-\eta)\} = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} x(n)x(n-\eta), \tag{1.133}$$

where $x(n)$ is some realization of $\underline{x}$. This is an extension of the idea of ergodicity discussed above, to a second-order case. Note carefully that the subscript on r is a signal name x, indicating the use of a signal to compute a time average, rather than random variables to compute an ensemble average. We have already introduced this notation, but it is worth reiterating here so that the reader is clear about its significance.

Since speech processing is an applied discipline, we will frequently use temporal, rather than ensemble, averages in our developments. Of course, this is because we have signals, rather than stochastic models, to deal with. On the other hand, there is often much to be gained by modeling speech as a stochastic process. Accordingly, when a speech signal is thought of as a realization of a stochastic process, the underlying process must be *assumed* to have the appropriate stationarity and ergodicity properties to allow the computation of meaningful temporal statistics.[23]

[23]A philosophical point is in order here. A moment's thought will reveal that speech, if thought of as a random process, cannot possibly comprise a *stationary* random process, since speech is a very dynamic phenomenon. This is an indication of the need for "short-term" analytical tools which can be applied to short temporal regions of assumed stationarity. At this point we begin to use formal theory in some rather *ad hoc* and *ad lib* ways. Of course, it is often the case in engineering problems that we use formal theories in rather loose ways in practice. However, the ability to understand the implications of our sloppiness, and the ability to predict and explain success in spite of it, depends entirely on our understanding of the underlying formal principles. In this book, we will stress the dependency of *ad hoc* methods on formal principles.

Multiple Random Processes

We now extend these ideas to the case of two random processes. As a natural extension of the concept of stationarity we have the following: Two random processes, $\underline{x}$ and $\underline{y}$, are said to be *jointly SSS* if

$$\begin{aligned} &f_{\underline{x}_1(n_1),\ldots,\underline{x}_i(n_i),\underline{y}_1(m_1),\ldots,\underline{y}_j(m_j)}(\xi_1,\ldots,\xi_i,v_1,\ldots,v_j) \\ &\quad = f_{\underline{x}_1(n_1+\Delta),\ldots,\underline{x}_i(n_i+\Delta)\underline{y}_i(m_1+\Delta),\ldots,\underline{y}_j(m_j+\Delta)}(\xi_1,\ldots,\xi_i,v_1,\ldots,v_j) \end{aligned} \tag{1.134}$$

for any i random variables from $\underline{x}$, and any j from $\underline{y}$, and for any Δ. It follows that if $\underline{x}$ and $\underline{y}$ are jointly SSS, then each is individually SSS.

From random variables $\underline{x}(n_1)$ and $\underline{y}(n_2)$, chosen from $\underline{x}$ and $\underline{y}$, respectively, we can form the *cross-correlation*,

$$r_{\underline{x}\underline{y}}(n_1,n_2) \stackrel{\text{def}}{=} \mathcal{E}\{\underline{x}(n_1)\underline{y}(n_2)\}, \tag{1.135}$$

and the *cross-covariance*,

$$c_{\underline{x}\underline{y}}(n_1,n_2) \stackrel{\text{def}}{=} \mathcal{E}\Big\{[\underline{x}(n_1)-\mathcal{E}\{\underline{x}(n_1)\}][\underline{y}(n_2)-\mathcal{E}\{\underline{y}(n_2)\}]\Big\}. \tag{1.136}$$

Similarly to (1.129), we obtain

$$c_{\underline{x}\underline{y}}(n_1,n_2) = r_{\underline{x}\underline{y}}(n_1,n_2) - \mathcal{E}\{\underline{x}(n_1)\}\,\mathcal{E}\{\underline{y}(n_2)\}. \tag{1.137}$$

As we did in the individual random process case, it will be useful to have a weaker form of stationarity between two random processes. The following conditions are required for $\underline{x}$ and $\underline{y}$ to be declared *jointly WSS*:

1. $\underline{x}$ and $\underline{y}$ are *individually WSS*;
2. $r_{\underline{x}\underline{y}}(n_1,n_2)$ is a function of $\eta = n_2 - n_1$ only.

It is easy to show that joint SSS $\Rightarrow$ joint WSS (but not the converse). Also, simply by definition, we see that joint WSS $\Rightarrow$ individual SSS, but, again, the converse is not generally true.

As an extension of the concept of ergodicity to the joint random process case, we note that the cross-correlation can be computed using a temporal average over two realizations if the processes are jointly *correlation-ergodic*:

$$r_{\underline{x}\underline{y}}(\eta) = \mathcal{A}\{x(n)y(n-\eta)\} = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} x(n)y(n-\eta). \tag{1.138}$$

Such a computation, of course, makes no sense unless the two random processes are at least jointly WSS.

Power Density Spectrum

Single Random Process. A general discussion of this important topic is unnecessary for our work with speech and would take us too far afield.

We refer the reader to the textbooks in Appendix 1.B for a general background. For our purposes, it is sufficient to define the *power density spectrum* of a WSS random process $\underline{x}$ as the DTFT of its autocorrelation function,

$$\Gamma_{\underline{x}}(\omega) \stackrel{\text{def}}{=} \sum_{\eta=-\infty}^{\infty} r_{\underline{x}}(\eta) e^{-j\omega\eta}. \tag{1.139}$$

Accordingly, the autocorrelation can be computed from the power density spectrum as

$$r_{\underline{x}}(\eta) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \Gamma_{\underline{x}}(\omega) e^{j\omega\eta} \, d\omega. \tag{1.140}$$

If $\underline{x}$ is also correlation-ergodic and the autocorrelation is computed using time averaging, then, according to our convention, the subscripts will denote realizations. For example,

$$\Gamma_x(\omega) = \sum_{\eta=-\infty}^{\infty} r_x(\eta) e^{j\omega\eta}. \tag{1.141}$$

The *total*[24] *power* in a second-order stationary real random process is defined as

$$P_{\underline{x}} \stackrel{\text{def}}{=} \mathcal{E}\{\underline{x}^2(n)\} \qquad \text{for any } n. \tag{1.142}$$

To make sense of this definition, we recall the definition of the power in a *signal*, which according to (1.11) is given by[25]

$$P_x = \mathcal{L}\{|x(n)|^2\}. \tag{1.143}$$

If $x(n)$ happens to be a realization of $\underline{x}$, and $\underline{x}$ is second-order ergodic, then we see that these two computations are equivalent.

As an aside, we recall that realizations of stationary, ergodic, stochastic processes were listed as a class of power signals in Section 1.2.3. Indeed, we now can appreciate that this is the case. If $x(n)$ is such a realization and is not a power signal, then

$$P_{\underline{x}} = P_x = \{0 \text{ or } \infty\} \tag{1.144}$$

and we encounter a contradiction.

Now that the definition of $P_{\underline{x}}$ makes sense, we note that

$$P_{\underline{x}} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \Gamma_{\underline{x}}(\omega) \, d\omega = \frac{1}{\pi} \int_{0}^{\pi} \Gamma_{\underline{x}}(\omega) \, d\omega = r_{\underline{x}}(0). \tag{1.145}$$

[24]The word *total* is used here to connote that the power in all frequencies is considered.

[25]The absolute value signs appear here because $x(n)$ was assumed to be complex-valued in general in definition (1.11). Since we have focused exclusively upon real random processes, they are superfluous in this discussion.

This result follows immediately from definitions and says that the scaled total area under $\Gamma_{\underline{x}}(\omega)$ yields the total power in $\underline{x}$, making it a sort of density of power on frequency, much like the pdf is a probability density on its variable of interest. In fact, to find the power in any frequency range, say ω_1 to ω_2, for $\underline{x}$, we can compute

$$\text{Power in } \underline{x} \text{ in frequencies } \omega_1 \text{ to } \omega_2 = \frac{1}{\pi} \int_{\omega_1}^{\omega_2} \Gamma_{\underline{x}}(\omega)\, d\omega. \tag{1.146}$$

Finally, we remark that some stochastic processes have all of their power concentrated at discrete frequencies. For example, a process $\underline{x}$ whose random variables are $\{\underline{x}(n) = \cos(\omega_0 n + \underline{\theta}),\ n \in (-\infty, \infty)\}$ with $\underline{\theta}$ a random variable, will have all power concentrated at frequency ω_0. In this case, the autocorrelation (ensemble or temporal) will be periodic with the same frequency, and we must resort to the use of impulses in the PDS much like our work with the PDS for a periodic *deterministic* process.

Two Random Processes. Let us focus here on jointly WSS random processes, $\underline{x}$ and $\underline{y}$, with cross-correlation $r_{\underline{xy}}(\eta)$. In this case the *cross-power spectral density* is given by

$$\Gamma_{\underline{xy}}(\omega) = \sum_{\eta=\infty}^{\infty} r_{\underline{xy}}(\eta)\, e^{-j\omega\eta}. \tag{1.147}$$

We can compute the *cross power* between the two processes,

$$P_{\underline{xy}} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \Gamma_{\underline{xy}}(\omega)\, d\omega, \tag{1.148}$$

which is interpretable as the power that the two random processes generate over and above their individual powers due to the fact that they are correlated.

Noise

Realizations of stochastic processes often occur as unwanted disturbances in engineering applications and are referred to as *noise*. Even when the stochastic signal is not a disturbance, we often employ the term noise. Such will be the case in our speech work, for example, when a noise process appears as the driving function for a model for "unvoiced" speech sounds. (Consider, e.g., the sound that the letter "s" implies.)

One of the most important forms of noise in engineering analysis is (*discrete-time*) *white noise*, defined as a stationary process, say $\underline{w}$, with the property that its power density spectrum is constant over the Nyquist range,

$$\Gamma_{\underline{w}}(\omega) = 2\pi, \qquad \text{for } \omega \in [-\pi, \pi). \tag{1.149}$$

Accordingly, the autocorrelation function for white noise is

$$r_{\underline{w}}(\eta) = \delta(\eta). \tag{1.150}$$

The reader is cautioned to distinguish between *continuous-time* white noise and the phenomenon we are discussing here. Continuous white noise has infinite power and a flat power density spectrum over all frequencies. Just as the discrete-time impulse cannot be considered as samples of the continuous time impulse, so discrete-time white noise should not be considered to be samples of continuous time white noise. In fact, discrete-time white noise may be thought to represent samples of a continuous time stochastic process, which is bandlimited to the Nyquist range and which has a flat power density spectrum over that range.

Random Processes and Linear Systems

It will be useful for us to review a few key results concerning the analysis of LTI discrete time systems with stochastic inputs. Let us restrict this discussion to WSS, second-order ergodic, stochastic processes.

Consider first an LTI system with discrete-time impulse response $h(n)$. Suppose that $x(n)$, a realization of random process $\underline{x}$, is input to the system. The output, say $y(n)$, is given by the convolution sum,

$$y(n) = \sum_{i=-\infty}^{\infty} x(n-i)h(i). \tag{1.151}$$

Of course, the same transformation occurs on the input no matter which realization of $\underline{x}$ it happens to be. We could denote this fact by replacing $x(n-i)$ by its corresponding random variable, $\underline{x}(n-i)$, on the right side of (1.151). Without a rigorous argument,[26] it is believable that the mapping of these random variables by the convolution sum will produce another random variable (for a fixed n), $\underline{y}(n)$, so we write

$$\underline{y}(n) = \sum_{i=-\infty}^{\infty} \underline{x}(n-i)h(i). \tag{1.152}$$

As n varies, a second random process is created at the output, $\underline{y}$. We have assumed $\underline{x}$ to be WSS and second-order ergodic. Let us show that the same is true of $\underline{y}$.

By applying the expectation operator to both sides of (1.152) and interchanging the order of summation on the right, we have

$$\mathcal{E}\{\underline{y}(n)\} = \sum_{i=-\infty}^{\infty} \mathcal{E}\{\underline{x}(n-i)\}\, h(i) \tag{1.153}$$

or

[26]This argument centers on concepts of stochastic convergence that are treated in many standard textbooks (see books in Appendix 1.B not labeled "elementary").

$$\mu_{\underline{y}} = \mu_{\underline{x}} \sum_{i=-\infty}^{\infty} h(i). \tag{1.154}$$

Since this result does not depend on n, we see that $\underline{y}$ is stationary in the mean. A similar result obtains with $\mu_{\underline{y}}$ and $\mu_{\underline{y}}$ replaced by μ_x and μ_y if we begin with (1.151) and use temporal averages, so that $\underline{y}$ is also ergodic in the mean.

In a similar way (see Problem 1.14) we can show that the autocorrelation of $\underline{y}$ is dependent only on the time difference in the arguments and is given by

$$r_{\underline{y}}(\eta) = \sum_{i=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} h(i)h(k)r_{\underline{x}}(\eta + k - i), \tag{1.155}$$

or, in terms of temporal autocorrelations,

$$r_y(\eta) = \sum_{i=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} h(i)h(k)r_x(\eta + k - i). \tag{1.156}$$

We conclude, therefore, that a WSS correlation-ergodic input to an LTI system produces a WSS correlation-ergodic output. This is a fundamental result that will be used implicitly in many places in our work.

Finally, we recall the important relationship between the input and output power spectral densities in the case of LTI systems with WSS inputs,

$$\Gamma_{\underline{y}}(\omega) = |H(\omega)|^2 \Gamma_{\underline{x}}(\omega). \tag{1.157}$$

This result is derived by taking the DTFT of both sides of (1.155).

1.2.4 Vector-Valued Random Processes

At several places in this book, we will encounter random processes that are vector-valued. A *vector-valued random process* $\underline{\mathbf{x}}$ is a collection of random vectors indexed by time,[27]

$$\underline{\mathbf{x}} \stackrel{\text{def}}{=} \{\ldots, \underline{\mathbf{x}}(-1), \underline{\mathbf{x}}(0), \underline{\mathbf{x}}(1), \ldots\}. \tag{1.158}$$

Realizations of these random processes comprise vector-valued signals of the form

$$\{\ldots, \mathbf{x}(-1), \mathbf{x}(0), \mathbf{x}(1), \ldots\}, \tag{1.159}$$

which we customarily denote simply $\mathbf{x}(n)$. (*Note*: We are now employing boldface to indicate vector quantities.)

[27]Again, we will restrict our attention to real processes, but the complex case is a simple generalization.

These sequences will arise in two different ways in our work. In the first case, the elements of each random vector will be random variables representing scalar signal samples, which for some reason are conveniently packaged into vectors. For example, suppose we have a *scalar* signal (random process) $\underline{x} = \{\ldots, \underline{x}(-1), \underline{x}(0), \underline{x}(1), \ldots\}$. We might find it necessary to break the signal into 100-point blocks for coding purposes, thereby creating a vector random process

$$\underline{\mathbf{x}} = \left\{ \ldots, \underline{\mathbf{x}}(0) = \begin{bmatrix} \underline{x}(0) \\ \underline{x}(1) \\ \vdots \\ \underline{x}(99) \end{bmatrix}, \underline{\mathbf{x}}(1) = \begin{bmatrix} \underline{x}(100) \\ \underline{x}(101) \\ \vdots \\ \underline{x}(199) \end{bmatrix}, \underline{\mathbf{x}}(2) = \begin{bmatrix} \underline{x}(200) \\ \underline{x}(201) \\ \vdots \\ \underline{x}(299) \end{bmatrix}, \ldots \right\}. \tag{1.160}$$

Note that the "time" indices of the vector random process represent a reindexing of the sample times of the original random process.

A second type of vector random process will result from the extraction of vector-valued features from frames of speech. We might, for example, extract 14 features from 160-point frames of speech. These frames may be overlapping, as shown in Fig. 1.12. In some cases we might choose to index the resulting random vectors by the end-times of the frames; in others we might reindex the vector sequence using consecutive integers. In either case, it is clear that the vector sequence comprises a vector-valued random process.

For a vector random process, the *mean vector* takes the place of the mean in the scalar case, and the *autocorrelation matrix* plays the role of the autocorrelation. These are

$$\boldsymbol{\mu}_{\underline{\mathbf{x}}(n)} \stackrel{\text{def}}{=} \mathcal{E}\{\underline{\mathbf{x}}(n)\} \tag{1.161}$$

and

$$\mathbf{R}_{\underline{\mathbf{x}}}(n_1, n_2) \stackrel{\text{def}}{=} \mathcal{E}\{\underline{\mathbf{x}}(n_1)\underline{\mathbf{x}}^T(n_2)\}, \tag{1.162}$$

respectively. Note that the mean vector contains the mean of each of the component random variables and the correlation matrix contains the cross-correlations between each component pair in the vectors. We can also speak of the *covariance matrix* of the vector random process $\underline{\mathbf{x}}$, defined as

$$\mathbf{C}_{\underline{\mathbf{x}}}(n_1, n_2) \stackrel{\text{def}}{=} \mathcal{E}\left\{[\underline{\mathbf{x}}(n_1) - \boldsymbol{\mu}_{\underline{\mathbf{x}}(n_1)}][\underline{\mathbf{x}}(n_2) - \boldsymbol{\mu}_{\underline{\mathbf{x}}(n_2)}]^T\right\}. \tag{1.163}$$

When the vector random process is WSS, we have a stationary mean vector, and correlation and covariance matrices that depend only on time difference. These are defined, for an arbitrary n, as follows:

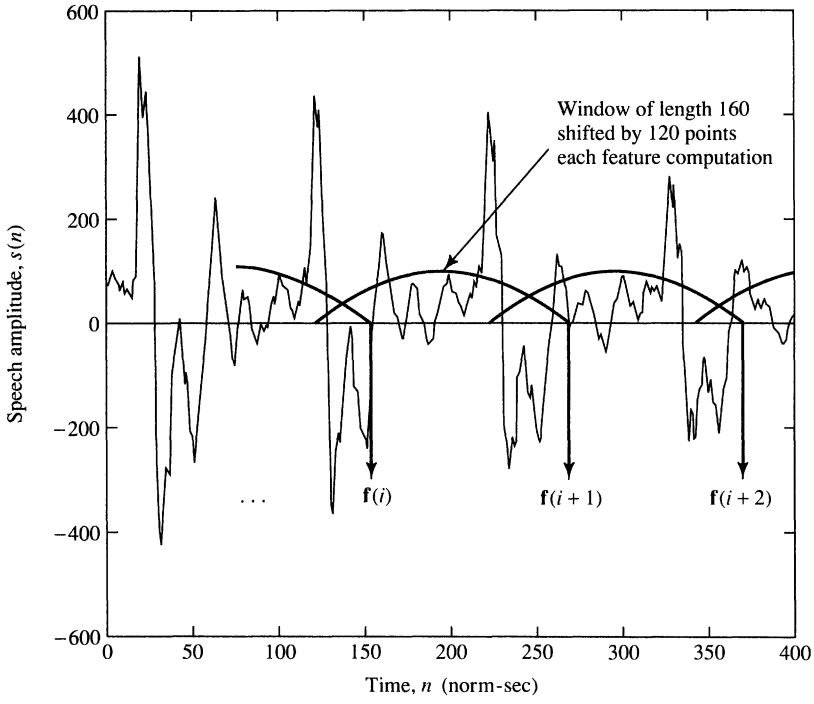


FIGURE 1.12. A vector random process created by extracting vector-valued features from frames of a speech process at periodic intervals. *Note*: Here we index the features by sequential integers. Later we will establish the convention of indexing features by the time of the leading edge of the sliding window.

$$\boldsymbol{\mu}_{\underline{\mathbf{x}}} \stackrel{\text{def}}{=} \boldsymbol{\mu}_{\underline{\mathbf{x}}(n)} = \mathcal{E}\{\underline{\mathbf{x}}(n)\} \tag{1.164}$$

$$\mathbf{R}_{\underline{\mathbf{x}}}(\eta) \stackrel{\text{def}}{=} \mathbf{R}_{\underline{\mathbf{x}}}(n, n-\eta) = \mathcal{E}\{\underline{\mathbf{x}}(n)\underline{\mathbf{x}}^T(n-\eta)\} \tag{1.165}$$

$$\mathbf{C}_{\underline{\mathbf{x}}}(\eta) \stackrel{\text{def}}{=} \mathbf{C}_{\underline{\mathbf{x}}}(n, n-\eta) = \mathcal{E}\left\{[\underline{\mathbf{x}}(n) - \boldsymbol{\mu}_{\underline{\mathbf{x}}}][\underline{\mathbf{x}}^T(n-\eta) - \boldsymbol{\mu}_{\underline{\mathbf{x}}}]^T\right\}. \tag{1.166}$$

Frequently, we are specifically interested in the "zero lag" correlations (or covariance) matrix of a stationary vector random process that plays the role of the variance of the process. For this case, we will write

$$\mathbf{R}_{\underline{\mathbf{x}}} \stackrel{\text{def}}{=} \mathbf{R}_{\underline{\mathbf{x}}}(0) \tag{1.167}$$

and

$$\mathbf{C}_{\underline{\mathbf{x}}} \stackrel{\text{def}}{=} \mathbf{C}_{\underline{\mathbf{x}}}(0) \tag{1.168}$$

for simplicity. The reader should carefully compare these notations with (1.116) and (1.117) and discern the difference in meaning.

Finally, we note that there are temporal versions of these three key sta-

tistical matrices that are meaningful when appropriate ergodicity conditions hold. These are, for an arbitrary n,

$$\boldsymbol{\mu}_{\mathbf{x}} \stackrel{\text{def}}{=} \boldsymbol{\mu}_{\mathbf{x}(n)} = \mathcal{L}\{\mathbf{x}(n)\} \tag{1.169}$$

$$\mathbf{R}_{\mathbf{x}}(\eta) \stackrel{\text{def}}{=} \mathcal{L}\{\mathbf{x}(n)\mathbf{x}^T(n-\eta)\} \tag{1.170}$$

$$\mathbf{C}_{\mathbf{x}}(\eta) \stackrel{\text{def}}{=} \mathcal{L}\left\{[\mathbf{x}(n)-\boldsymbol{\mu}_{\mathbf{x}}][\mathbf{x}^T(n-\eta)-\boldsymbol{\mu}_{\mathbf{x}}]^T\right\}. \tag{1.171}$$

We also define

$$\mathbf{R}_{\mathbf{x}} \stackrel{\text{def}}{=} \mathbf{R}_{\mathbf{x}}(0) \tag{1.172}$$

and

$$\mathbf{C}_{\mathbf{x}} \stackrel{\text{def}}{=} \mathbf{C}_{\mathbf{x}}(0). \tag{1.173}$$

1.3 Topics in Statistical Pattern Recognition

Reading Note: Most of the material in this section will not be used until Parts IV and V. The exception is Section 1.3.1, which will first be encountered in Chapter 5.

As in the previous two subsections of this chapter, the material treated here represents a very small sampling of a vast research discipline, with a focus on a few topics which will be significant to us in our speech processing work. Unlike the other two subsections, however, we make no assumption here or in the main text that the reader has a formal background in pattern recognition beyond a casual acquaintence with certain ideas that are inherent in general engineering study. A few example textbooks from this field are listed in Appendix 1.C.

Much of speech processing is concerned with the analysis and recognition of patterns and draws heavily on results from this field. Although many speech processing developments can be successfully understood with a rather superficial knowledge of pattern recognition theory, advanced research and development are not possible without a rigorous understanding. A few advanced speech processing topics in this book will need to be left to the reader's further pursuit, since it is not intended to assume this advanced pattern recognition background, nor is it possible to provide it within the scope of the book.

There are two main branches of pattern recognition—*statistical* and *syntactic*. Generally speaking, the former deals with statistical relationships among features in a pattern, while the latter approaches patterns as structures that can be composed of primitive patterns according to a set of rules. Although these branches are not exactly distinct, they are quite different in philosophy. In our work, the use of the latter is confined to the special problem of language modeling in automatic speech recogni-

tion.[28] We therefore defer any discussion of syntactic pattern recognition concepts to Chapter 13. Statistical pattern recognition methods, however, are quite prevalent in many aspects of speech processing, and it will be expedient for us to introduce a few key concepts before starting our study of speech.

We reemphasize that we are only discussing a few small topics in a vast and complex subject. Notably missing from our discussion, for example, is an analysis of how one chooses and evaluates in a rigorous sense the features representing a pattern. Frequently, this is accomplished in a rather *ad hoc* manner in speech processing, but the reader should be aware that a rich theory embracing this issue has been developed. A second example pertains to the covergence of clustering algorithms used to group features into classes. This issue will also be left for further study.

1.3.1 Distance Measures

Given two vectors $\mathbf{x}$ and $\mathbf{y}$ in a multidimensional space, we will frequently be interested in knowing "how far apart" they are. These vectors will often correspond to two time points in a realization of a vector-valued random process, or perhaps vectors drawn from two random processes. For the sake of discussion, let us just refer to $\mathbf{x}$ and $\mathbf{y}$.

It is sufficient for us to be concerned with vectors drawn from Cartesian spaces. The *N-dimensional real Cartesian space*, denoted $\mathbb{R}^N$ is the collection of all N-dimensional vectors with real elements. A *metric*, $d(\cdot\,,\cdot)$, on $\mathbb{R}^N$ is a real-valued function with three properties: For all $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{R}^N$,

1. $d(\mathbf{x}, \mathbf{y}) \geq 0$.
2. $d(\mathbf{x}, \mathbf{y}) = 0$ if and only if $\mathbf{x} = \mathbf{y}$.
3. $d(\mathbf{x}, \mathbf{y}) \leq d(\mathbf{x}, \mathbf{z}) + d(\mathbf{z}, \mathbf{y})$.

These properties coincide well with our intuitive notions about a proper measure of distance. Indeed, a metric is often used as a distance measure in mathematics and in engineering.[29]

Any function that meets the properties in the definition above is a legitimate metric on the vector space. Accordingly, there are many metrics, each having its own advantages and disadvantages. Most of the true metrics that we use in speech processing are particular cases of the Minkowski metric, or close relatives. This metric is defined as follows: Let x_k denote the kth component of the N-vector $\mathbf{x}$. Then the *Minkowski metric of order s*, or the l_s *metric*, between vectors $\mathbf{x}$ and $\mathbf{y}$ is

[28] In fact, syntactic pattern recognition has its roots in the theory of formal languages that was motivated by the study of natural languages (see Chapter 13).

[29] We will, however, encounter some distance measures later in the book that are not true metrics.

$$d_s(\mathbf{x}, \mathbf{y}) \stackrel{\text{def}}{=} \sqrt[s]{\sum_{k=1}^{N} |x_k - y_k|^s} \cdot \tag{1.174}$$

Particular cases are

1. The l_1 or *city block* metric,

$$d_1(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{N} |x_k - y_k| \cdot \tag{1.175}$$

2. The l_2 or *Euclidean* metric,

$$d_2(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{k=1}^{N} |x_k - y_k|^2} = \sqrt{[\mathbf{x} - \mathbf{y}]^T [\mathbf{x} - \mathbf{y}]} \, . \tag{1.176}$$

3. The l_∞ or *Chebyshev* metric (corresponds to the Minkowski metric as $s \to \infty$),

$$d_\infty(\mathbf{x}, \mathbf{y}) = \max_k |x_k - y_k| \, . \tag{1.177}$$

We should note that the l_s *norm* of a vector $\mathbf{x}$, denoted $\|\mathbf{x}\|_s$, is defined as

$$\|\mathbf{x}\|_s \stackrel{\text{def}}{=} \sqrt[s]{\sum_{k=1}^{N} |x|^s} \, . \tag{1.178}$$

It follows immediately that the l_s metric between the vectors $\mathbf{x}$ and $\mathbf{y}$ is equivalent to the l_s norm of the difference vector $\mathbf{x} - \mathbf{y}$,

$$d_s(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_s . \tag{1.179}$$

An important generalization of the Euclidean metric is called variously the *weighted Euclidean*, *weighted* l_2, or *quadratic* metric,[30]

$$d_{2w}(\mathbf{x}, \mathbf{y}) \stackrel{\text{def}}{=} \sqrt{[\mathbf{x} - \mathbf{y}]^T \mathbf{W} [\mathbf{x} - \mathbf{y}]} \, . \tag{1.180}$$

where $\mathbf{W}$ is a positive definite matrix that can be used for several purposes discussed below.

Before proceeding, we should be careful to point out that, in theoretical discussions, we might wish to discuss the distance between two stochastic vectors, say $\underline{\mathbf{x}}$ and $\underline{\mathbf{y}}$. In this case we might write, for example, something like

[30]The quadratic metric is often defined without the square root, but we employ the square root to make the distance more parallel to the Euclidean metric.

$$d_{2w}(\underline{\mathbf{x}}, \underline{\mathbf{y}}) = \sqrt{[\underline{\mathbf{x}} - \underline{\mathbf{y}}]^T \mathbf{W} [\underline{\mathbf{x}} - \underline{\mathbf{y}}]} \,. \tag{1.181}$$

The left side must be interpreted as a random variable that only takes a value when outcomes for $\underline{\mathbf{x}}$ and $\underline{\mathbf{y}}$ are known. The existence of this "random distance" depends upon concepts in stochastic calculus that will not concern us here (see textbooks in Appendix 1.B not labeled "elementary"). For our purposes, we just consider this notation to be a formal way of packaging together all possible outcomes of the distance that depend upon the random values of $\underline{\mathbf{x}}$ and $\underline{\mathbf{y}}$.

1.3.2 The Euclidean Metric and "Prewhitening" of Features

In this section we briefly make some points about the use of the Euclidean distance in engineering, which can have important consequences for performance of resulting algorithms and systems. The concepts discussed here have broader implications for abstract Hilbert spaces, but we will confine the remarks to simple vector spaces. The reader interested in a more formal and comprehensive treatment of these ideas should consult textbooks on linear algebra and functional analysis such as (Hoffman and Kunze, 1961, Ch. 2; Nobel, 1969, Ch. 14; Naylor and Sell, 1971; Lusternik and Sobolev, 1974).

Of the formal metrics in $\mathbb{R}^N$, the Euclidean metric is probably the most widely used in engineering problems. The reason for its popularity is that it fits precisely with our physical notion of distance. When the representation of a vector is based upon an orthonormal basis set, then the Euclidean distance between two vectors in the space conforms exactly to the "natural" distance between them. However, when vector representations are based upon a basis set that is not orthonormal (even if the set is orthogonal), then the Euclidean distance will yield "unnatural" results unless a linear operation is applied which transforms the vector representations to ones based on orthonormal vectors.

These ideas are illustrated in 2-space in Fig. 1.13. The representations of the vectors **a** and **b** with respect to the "natural" basis set $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ are

$$\mathbf{x} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \text{and} \quad \mathbf{y} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \tag{1.182}$$

respectively. By this we mean, for example, that

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \tag{1.183}$$

where

$$\mathbf{a} = x_1 \boldsymbol{\beta}_1 + x_2 \boldsymbol{\beta}_2. \tag{1.184}$$

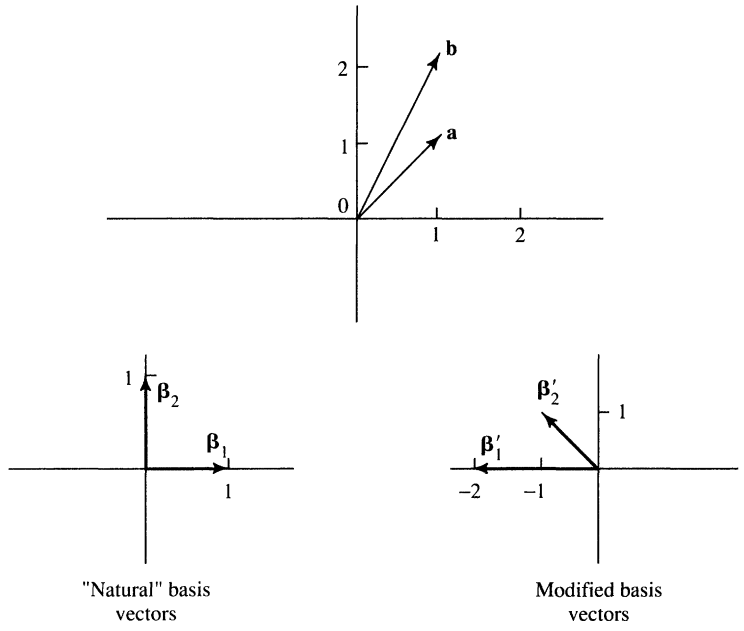


FIGURE 1.13. Vectors used to illustrate concepts of the Euclidean distance metric.

With this construction, everyone would agree that the distance between the vectors **a** and **b** is appropriately given by the Euclidean metric between the representations,

$$d_2(\mathbf{x}, \mathbf{y}) = 1. \tag{1.185}$$

Consistent with our discussion above, it is also true that the distance is given by the l_2, or Euclidean, norm of the difference vector,

$$\|\mathbf{x} - \mathbf{y}\|_2 = 1. \tag{1.186}$$

Suppose, however, that the given basis vectors were $\boldsymbol{\beta}_1'$ and $\boldsymbol{\beta}_2'$. In this case, the representations of **a** and **b** are

$$\mathbf{x}' = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \quad \text{and} \quad \mathbf{y}' = \begin{bmatrix} -\frac{3}{2} \\ 2 \end{bmatrix}, \tag{1.187}$$

respectively. In spite of the fact that **a** and **b** have not moved, the Euclidean distance between these representations is

$$d_2(\mathbf{x}', \mathbf{y}') = \sqrt{\tfrac{5}{4}}. \tag{1.188}$$

We note that the distance would be "incorrect" even if the new basis vectors were orthogonal but not normalized.

What is "wrong" in the second case is the basis upon which the representations are assigned coordinates. The two "coordinates" with the second assignment of basis vectors are not distinct information. Moving in the direction of $\boldsymbol{\beta}_1'$ also includes motion in the direction of $\boldsymbol{\beta}_2'$, and conversely. Only when these bases are made to correspond to distinct (orthonormal) pieces of information does our sense of distance come back into focus and the Euclidean distance become meaningful. Algebraically, if the basis vectors were made to correspond to a proper orthonormal set, then the Euclidean distance would be appropriate. This would require that we transform the vector representations $\mathbf{x}'$ and $\mathbf{y}'$ to their representations on a "proper" set of basis vectors before computing the Euclidean distance. In this case, let us just choose to go back to the orthonormal basis set $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$, in which case we know that $\mathbf{x}'$ and $\mathbf{y}'$ are transformed back to the original $\mathbf{x}$ and $\mathbf{y}$. Let us call the transformation $\mathbf{V}$. We have, then,

$$\mathbf{x} = \mathbf{V}\mathbf{x}' \qquad \text{and} \qquad \mathbf{y} = \mathbf{V}\mathbf{y}'. \tag{1.189}$$

In this contrived example, $\mathbf{V}$ can be found from (1.189) using simple algebra, because we happen to know what the transformed vectors are. In general, however, finding the transformed representation of a vector corresponding to a change of basis is a simple generalization of the following [see, e.g., (Chen, 1984, p. 17)]. We take the two columns of $\mathbf{V}$ to be: the representation of $\boldsymbol{\beta}_1'$ with respect to basis set $\{\boldsymbol{\beta}_1, \boldsymbol{\beta}_2\}$, and the representation of $\boldsymbol{\beta}_2'$ with respect to $\{\boldsymbol{\beta}_1, \boldsymbol{\beta}_2\}$, respectively.

Now consider computing the Euclidean distance of the transformed vectors to obtain a meaningful measure of their distance apart,

$$\begin{aligned} d_2(\mathbf{V}\mathbf{x}', \mathbf{V}\mathbf{y}') &= \sqrt{[\mathbf{V}\mathbf{x}' - \mathbf{V}\mathbf{y}']^T[\mathbf{V}\mathbf{x}' - \mathbf{V}\mathbf{y}']} \\ &= \sqrt{[\mathbf{x}' - \mathbf{y}']^T\mathbf{V}^T\mathbf{V}[\mathbf{x}' - \mathbf{y}']} \\ &= d_{2_W}(\mathbf{x}', \mathbf{y}'). \end{aligned} \tag{1.190}$$

The last line in (1.190) denotes the weighted Euclidean distance with weighting matrix $\mathbf{W} = \mathbf{V}^T\mathbf{V}$. We see that the "meaningful" Euclidean distance for the vectors whose bases are not conducive to proper distance computation can be obtained by using a weighting matrix equivalent to the "square" of the transformation matrix.

It is sometimes desirable that a linear transformation of coordinates not change the rank ordering of distances from some reference vector. If in the above, for example, there were some vector z' such that

$$d_2(\mathbf{x}', \mathbf{z}') < d_2(\mathbf{y}', \mathbf{z}'), \tag{1.191}$$

then it might be desirable that

$$d_2(\mathbf{x}, \mathbf{z}) < d_2(\mathbf{y}, \mathbf{z}). \tag{1.192}$$

Whereas we would want the transformation to make the distance more meaningful, we might not wish to have the rank ordering changed in the new feature space. In general, the (weighted) Euclidean distance does *not* preserve this ordering.

In effect, what we have done in the above example is removed the redundant information in the "bad" vector representations that skews our sense of how naturally far apart they are. This is accomplished by linear transformation of the space, or, equivalently, weighting of the distance metric. This example was meant to build intuition about a more realistic and important problem in pattern recognition. We often encounter (random) vectors of features whose elements are highly correlated, or inappropriately scaled. The correlation and scaling effects will occur, for example, when multiple measurements are made on the same process and mixed in the same feature vector. For example, we might measure the average number of zero crossings[31] per norm-sec in a speech frame, and also the average energy. Clearly, there is no reason to believe that these numbers will have similar magnitudes in a given frame, since they represent quite different measurements on the sequence. Suppose, for example, that in one frame we measure 240 "joules," and 0.1 zero crossing, per norm-sec. In the next, we measure 300 and 0.05. Are these vector representations based on an appropriate orthonormal basis set so that Euclidean distances are meaningful? This answer could be argued either way, but the question is really academic. Our satisfaction with the distance measure here will depend upon how faithfully it reflects the difference in the frames in light of the measurements. So let us explore the question: Do these two frames represent the same sound? If so, we would like the distance to be small.

In answering this question, we should notice two things about the measurements. First, there could be less information in these measurements than we might assume. It could be the case that zero crossings tend to decrease when energy increases (correlation) so that the combination of changes does not make the two frames as different as the outcome might suggest. This point is reminiscent of the nonorthonormal basis case above. Second, note that the zero crossing measure is so relatively small in amplitude that its effect on the distance is negligible. In order for this feature to have more "discriminatory power" (which does not potentially get lost in numerical roundoff errors[32]), the relative scale of the features must be adjusted. (This corresponds to basis vectors of grossly different lengths, orthogonal or not.) An approach to solving this scaling problem is to simply normalize the feature magnitudes so that each has unity variance. Presumably, smaller features will have smaller variances (and conversely) and this will tend to bring the measurements into an appropriate relative scale. The "decorrelation" process is also not difficult; in fact, the scaling can be accomplished simultaneously using the following.

[31]The average number of times the sequence changes sign. This gives a rough measure of frequency content.

[32]Also as a practical matter, the measurement on a "low amplitude" feature is potentially much more susceptible to roundoff error problems in numerical computations, and the presence of grossly misscaled features can cause other numerical problems such as an ill-conditioned covariance matrix (Nobel, 1969, Sec. 8.2).

Suppose that the feature vectors between which we are trying to compute a distance are $\mathbf{x}'$ and $\mathbf{y}'$. Each is an outcome of random vector $\underline{\mathbf{x}}'$ with mean $\boldsymbol{\mu}_{\underline{\mathbf{x}}'}$ and covariance matrix $\mathbf{C}_{\underline{\mathbf{x}}'}$. We would like to transform the original random variable $\underline{\mathbf{x}}'$ to a representation, say $\underline{\mathbf{x}}$, in which all components are uncorrelated and are individually of unity variance. This means that the new covariance matrix $\mathbf{C}_{\underline{\mathbf{x}}}$ should equal $\mathbf{I}$, where $\mathbf{I}$ is the identity matrix. According to the heuristic arguments above, Euclidean distance computed on these vectors will then be intuitively appealing. As in the simple vector example above, we will show that the proper Euclidean distance can be computed using an appropriate weighting matrix in the computation.

The requisite transformation on the feature vectors is easily discovered by focusing on the covariance matrix. Since $\mathbf{C}_{\underline{\mathbf{x}}'}$ is a symmetric matrix, it can be written [see, e.g., (Nobel, 1969, Ch. 10)]

$$\mathbf{C}_{\underline{\mathbf{x}}'} = \boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^T, \tag{1.193}$$

where $\boldsymbol{\Phi}$ is an orthogonal matrix whose columns are the normalized eigenvectors of $\mathbf{C}_{\underline{\mathbf{x}}'}$, and $\boldsymbol{\Lambda}$ is a diagonal matrix of eigenvalues of $\mathbf{C}_{\underline{\mathbf{x}}'}$. Therefore,

$$\boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^T = \mathcal{E}\left\{\left[\underline{\mathbf{x}}' - \boldsymbol{\mu}_{\underline{\mathbf{x}}'}\right]\left[\underline{\mathbf{x}}' - \boldsymbol{\mu}_{\underline{\mathbf{x}}'}\right]^T\right\}, \tag{1.194}$$

from which it follows that

$$\mathbf{I} = \mathcal{E}\left\{\boldsymbol{\Lambda}^{-1/2}\boldsymbol{\Phi}^T\left[\underline{\mathbf{x}}' - \boldsymbol{\mu}_{\underline{\mathbf{x}}'}\right]\left[\underline{\mathbf{x}}' - \boldsymbol{\mu}_{\underline{\mathbf{x}}'}\right]^T\boldsymbol{\Phi}\boldsymbol{\Lambda}^{-1/2}\right\}. \tag{1.195}$$

Clearly, therefore, if we transform the feature vectors using the transformation

$$\underline{\mathbf{x}} = \boldsymbol{\Lambda}^{-1/2}\boldsymbol{\Phi}^T\underline{\mathbf{x}}', \tag{1.196}$$

we will be dealing with uncorrelated random vectors for which the Euclidean metric will provide a proper measure of distance. In this case,

$$\begin{aligned} d_2(\boldsymbol{\Lambda}^{-1/2}\boldsymbol{\Phi}^T\underline{\mathbf{x}}', \boldsymbol{\Lambda}^{-1/2}\boldsymbol{\Phi}^T\underline{\mathbf{y}}') &= \sqrt{[\underline{\mathbf{x}}' - \underline{\mathbf{y}}']^T\boldsymbol{\Phi}\boldsymbol{\Lambda}^{-1/2}\boldsymbol{\Lambda}^{-1/2}\boldsymbol{\Phi}^T[\underline{\mathbf{x}}' - \underline{\mathbf{y}}']} \\ &= \sqrt{[\underline{\mathbf{x}}' - \underline{\mathbf{y}}']^T\boldsymbol{\Phi}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi}^T[\underline{\mathbf{x}}' - \underline{\mathbf{y}}']} \\ &= \sqrt{[\underline{\mathbf{x}}' - \underline{\mathbf{y}}']^T\mathbf{C}_{\underline{\mathbf{x}}'}^{-1}[\underline{\mathbf{x}}' - \underline{\mathbf{y}}']} \\ &= d_{2w}(\underline{\mathbf{x}}', \underline{\mathbf{y}}'). \end{aligned} \tag{1.197}$$

We see again that a meaningful Euclidean distance between correlated feature vectors can be computed if an appropriate weight is used. It is worth noting that the weighted Euclidean distance which has arisen here is very similar to the Mahalanobis distance that we discuss below.

The linear operation applied to the feature vectors in this procedure is frequently referred to as a *prewhitening transformation*, since it produces feature vectors whose components are uncorrelated and normalized. This terminology is somewhat abusive because the "white" concept applies to (usually scalar) random *processes* that are not being considered here, and also because "white" features would have zero means. This latter point would require, for a better analogy, that $\mathbf{R}_{\underline{\mathbf{x}}}$, rather than $\mathbf{C}_{\underline{\mathbf{x}}}$ be $\mathbf{I}$. Nevertheless, the terminology is widely used and is well understood by signal processing engineers.

Simplifications of the prewhitening procedure are sometimes employed to avoid the computational expense of using the full covariance matrix in the distance expression. The most common simplification is to assume that the features are mutually uncorrelated, but inappropriately scaled relative to one another. In this case $\mathbf{C}_{\underline{\mathbf{x}}'}$ (it is assumed) has the form

$$\mathbf{C}_{\underline{\mathbf{x}}'} = \mathbf{\Lambda}, \tag{1.198}$$

where $\mathbf{\Lambda}$ is a diagonal matrix whose diagonal elements in general are unequal. The transformation that need be done on each incoming vector is represented by $\mathbf{\Lambda}^{-1/2}\underline{\mathbf{x}}$. This amounts to simply normalizing each feature to its standard deviation so that all features may contribute equally to the distance.

1.3.3 Maximum Likelihood Classification

We frequently encounter problems in engineering in which a pattern representation is to be associated with one of a number of classes of patterns. This paradigm will occur in several significant places in our work. The purpose of this section is to explore a few underlying concepts with a particular interest in studying the distance measures that are often used in this endeavor.

Suppose that we have a set of classes, indexed by integers, say $c = 1, 2, \ldots, K$, which are outcomes of the class random variable, $\underline{c}$. Suppose that we also have a feature vector modeled by the random vector $\underline{\mathbf{x}}$. For example, the classes might represent the words in a vocabulary, and the feature vector a list of acoustic features extracted from the utterance of a word to be recognized. Ideally, given a feature vector outcome $\underline{\mathbf{x}} = \mathbf{x}$, we would select the class for which the conditional probability is highest. That is, c^* is the selected class (word) if

$$c^* = \underset{c}{\operatorname{argmax}}\, P(\underline{c} = c \,|\, \underline{\mathbf{x}} = \mathbf{x}). \tag{1.199}$$

Unfortunately, the training process usually does not permit characterization of the probabilities $P(\underline{c} = c|\underline{\mathbf{x}} = \mathbf{x})$. Instead what we learn is the probability that a given class will generate certain feature vectors, rather than

the converse. The training process yields conditional probabilities of the form $P(\underline{\mathbf{x}} = \mathbf{x} | \underline{c} = c)$. So we ask whether it makes sense to select

$$c^* = \underset{c}{\operatorname{argmax}} P(\underline{\mathbf{x}} = \mathbf{x} | \underline{c} = c). \tag{1.200}$$

By definition

$$P(\underline{c} = c | \underline{\mathbf{x}} = \mathbf{x}) = \frac{P(\underline{c} = c, \underline{\mathbf{x}} = \mathbf{x})}{P(\underline{\mathbf{x}} = \mathbf{x})} \tag{1.201}$$

and

$$P(\underline{\mathbf{x}} = \mathbf{x} | \underline{c} = c) = \frac{P(\underline{c} = c, \underline{\mathbf{x}} = \mathbf{x})}{P(\underline{c} = c)}, \tag{1.202}$$

from which we have

$$P(\underline{c} = c | \underline{\mathbf{x}} = \mathbf{x}) = \frac{P(\underline{\mathbf{x}} = \mathbf{x} | \underline{c} = c) P(\underline{c} = c)}{P(\underline{\mathbf{x}} = \mathbf{x})}. \tag{1.203}$$

Clearly, the choice of c that maximizes the right side will also be the choice of c that maximizes the left side. Therefore,

$$c^* = \underset{c}{\operatorname{argmax}}\ P(\underline{c} = c | \underline{\mathbf{x}} = \mathbf{x}) = \underset{c}{\operatorname{argmax}}\ P(\underline{\mathbf{x}} = \mathbf{x} | \underline{c} = c) P(\underline{c} = c). \tag{1.204}$$

Furthermore, *if the class probabilities are equal,*

$$P(\underline{c} = c) = \frac{1}{K}, \qquad c = 1, 2, \ldots, K, \tag{1.205}$$

then

$$c^* = \underset{c}{\operatorname{argmax}}\ P(\underline{c} = c | \underline{\mathbf{x}} = \mathbf{x}) = \underset{c}{\operatorname{argmax}}\ P(\underline{\mathbf{x}} = \mathbf{x} | \underline{c} = c). \tag{1.206}$$

Therefore, under the condition of equal *a priori* class probabilities, the class decision

$$c^* = \underset{c}{\operatorname{argmax}}\ P(\underline{\mathbf{x}} = \mathbf{x} | \underline{c} = c) \tag{1.207}$$

is equivalent to the more desirable (1.199) for which we do not have probability distributions.

A quantity related to the probability of an event which is used to make a decision about the occurrence of that event is often called a *likelihood measure.* Hence, our decision rule based on given feature vector $\mathbf{x}$ is to choose the class c that maximizes the likelihood $P(\underline{\mathbf{x}} = \mathbf{x} | \underline{c} = c)$. This is called the *maximum likelihood decision.*

There is an implicit assumption in the discussion above that the random feature vector may only assume one of a finite number of outcomes. This is evident in the writing of probability distribution $P(\underline{\mathbf{x}} = \mathbf{x} | \underline{c} = c)$. Where this is not the case, it is frequently assumed that feature vectors associated with a given class are well modeled by a multivariate Gaussian distribution [cf. (1.118)],

$$f_{\underline{\mathbf{x}}|\underline{c}}(x_1, \ldots, x_N|c) = f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c)$$
$$= \frac{1}{\sqrt{(2\pi)^N \det \mathbf{C}_{\underline{\mathbf{x}}|c}}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{\underline{\mathbf{x}}|c})^T \mathbf{C}_{\underline{\mathbf{x}}|c}^{-1}(\mathbf{x}-\boldsymbol{\mu}_{\underline{\mathbf{x}}|c})\right\}, \tag{1.208}$$

where $\mathbf{x}$ denotes the N-vector of arguments (features) $[x_1 \cdots x_N]^T$ and $\mathbf{C}_{\underline{\mathbf{x}}|c}$ and $\boldsymbol{\mu}_{\underline{\mathbf{x}}|c}$ are the class-conditional covariance matrix and mean vector. Without belaboring the issue, it is believable based on our previous discussion that an appropriate likelihood measure for this case is the class-conditional density $f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c)$. The class decision is based on maximizing the likelihood,

$$c^* = \underset{c}{\text{argmax}}\, f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c). \tag{1.209}$$

We can rid ourselves of the need to compute the exponential by electing instead to maximize $\ln f_{\underline{\mathbf{x}}|c}(\mathbf{x}|c)$. This leads to the decision rule

$$c^* = \underset{c}{\text{argmin}}\left\{[\mathbf{x}-\boldsymbol{\mu}_{\underline{\mathbf{x}}|c}]^T \mathbf{C}_{\underline{\mathbf{x}}|c}^{-1}[\mathbf{x}-\boldsymbol{\mu}_{\underline{\mathbf{x}}|c}] + \ln\left\{\det \mathbf{C}_{\underline{\mathbf{x}}|c}\right\}\right. . \tag{1.210}$$

Note that the maximization has become a *minimization* because we have removed a superfluous minus sign from the computation. Notice also that the first term on the right has the form of a weighted Euclidean distance. Let us further develop this point.

The term on the right side of (1.210) is sometimes considered a *distance* between the given feature vector and the cth class mean, $\boldsymbol{\mu}_{\mathbf{x}|c}$. Accordingly, it provides a measure of "how far $\mathbf{x}$ is from class c." For generality, let us replace the specific outcome of the feature vector, $\mathbf{x}$, with its random variable, $\underline{\mathbf{x}}$, and define the *maximum likelihood distance* as

$$d_{ml}(\underline{\mathbf{x}}, \boldsymbol{\mu}_{\underline{\mathbf{x}}|c}) = [\underline{\mathbf{x}}-\boldsymbol{\mu}_{\underline{\mathbf{x}}|c}]^T \mathbf{C}_{\underline{\mathbf{x}}|c}^{-1}[\underline{\mathbf{x}}-\boldsymbol{\mu}_{\underline{\mathbf{x}}|c}] + \ln\left\{\det \mathbf{C}_{\underline{\mathbf{x}}|c}\right\}. \tag{1.211}$$

We see that for a multiclass, multivariate Gaussian feature problem, choosing the class that minimizes this distance is equivalent to choosing the maximum likelihood class.

A simplification occurs when all classes share a common covariance matrix, say

$$\mathbf{C}_{\underline{\mathbf{x}}} \overset{\text{def}}{=} \mathbf{C}_{\underline{\mathbf{x}}|1} = \mathbf{C}_{\underline{\mathbf{x}}|2} = \cdots = \mathbf{C}_{\underline{\mathbf{x}}|K}. \tag{1.212}$$

In this case $\mathbf{C}_{\underline{\mathbf{x}}|c}$ can be replaced by $\mathbf{C}_{\underline{\mathbf{x}}}$ in (1.211) and the final $\ln\{\cdot\}$ can be ignored, since it simply adds a constant to all distances. In this case, we obtain

$$d_M(\underline{\mathbf{x}}, \boldsymbol{\mu}_{\underline{\mathbf{x}}|c}) = [\underline{\mathbf{x}}-\boldsymbol{\mu}_{\underline{\mathbf{x}}|c}]^T \mathbf{C}_{\underline{\mathbf{x}}}^{-1}[\underline{\mathbf{x}}-\boldsymbol{\mu}_{\underline{\mathbf{x}}|c}]. \tag{1.213}$$

This distance is frequently called the *Mahalanobis distance* (Mahalanobis, 1936). We see that for a multiclass, multivariate Gaussian feature

problem in which the classes share a common covariance matrix (the way in which features are correlated is similar across classes), choosing the class to which the given feature vector is closest in the sense of the Mahalanobis distance is tantamount to choosing the maximum likelihood class.

Interestingly, we have come full circle in our discussion, for it is apparent that the Mahalanobis distance is nothing more than a "covariance weighted" (squared) Euclidean distance[33] between the feature vector and a special set of deterministic vectors—the means of the classes. Nevertheless, the name Mahalanobis distance is often applied to this distance in this special maximum likelihood problem. Based on our previous discussion, it should be apparent that the Mahalanobis distance represents an appropriate use of the l_2 metric, since the inverse covariance weighting removes the correlation among the features in the vectors.

1.3.4 Feature Selection and Probabilistic Separability Measures

In the preceding section, we discussed a general problem in which a feature vector was associated with one of a number of classes. A subject that we avoided was the selection of features (this process is often called *feature extraction*) and their evaluation in terms of classification performance. These tasks are inseparable, since performance evaluation is often integrated into the search for appropriate features. In this section we make a few brief comments about these issues. One of the objectives is to let the reader know what material is not being covered with regard to this topic, and why. Another is to touch on the subject of probability separability measures and entropy measures, and to explain their specific relationship to speech processing.

Feature selection and evaluation is a vast subject on which much research has been performed and many papers and books written. To attempt to address this subject in any detail would take us too far afield from the main subject of this book. Several excellent textbooks address this field authoritatively and in detail, and we refer the reader to these books and the research literature for detailed study.[34] Second, the importance of feature evaluation procedures is diminished relative to the early days of speech processing. Although statistical pattern recognition techniques are central to the operation and performance of many speech processing tasks (particularly speech recognition), decades of research and development have led to convergence on a few (spectrally based) fea-

[33]Again, we could introduce a square root into the definition to make this distance exactly a Euclidean metric as defined in (1.180), but that would be breaking with convention. The Mahalanobis distance is almost invariably defined without the square root, and it should be clear that for the maximum likelihood problem, whether the distance is squared or not is of no consequence.

[34]For example, see the textbooks in Appendix 1.C and the *IEEE Transactions on Pattern Analysis and Machine Intelligence.*

tures that perform well, and appear to be enduring. This is not to say that new features have not been tried, and that the field is not evolving. Indeed, we have seen, for example, "cepstral" type of features supplant the "LP" type parameters in certain speech recognition tasks during the 1980s. This shift, however, was between two closely related sets of features and was to some extent motivated by computational expediencies. Further, the most frequently cited study behind this shift relies on experimental evidence of improved recognition performance (Davis and Mermelstein, 1980). Although the course of research is very unpredictable, for the foreseeable future, there appears to be no compelling problems that will demand a deep analysis of features.

As if to contradict the statement above, lurking in one little corner of our study (Section 12.2.7) we will mention some directions in speech recognition research that are based on the notions of probabilistic separability and entropy measures. These measures are customarily encountered in the advanced study of feature extraction and evaluation. We conclude this section by broadly discussing the types of feature evaluation, putting the topic of probabilistic separability measures and entropy measures into perspective.

Probabilistic Distance Measures

Ideally, features would be evaluated on their performance in terms of minimizing the rate of classification error. However, error rate is generally a very difficult quantity to evaluate, and other techniques must be employed. Almost all commonly used techniques for feature evaluation involve some attempt to measure the separation of classes when represented by the features.

The simplest techniques for measuring class separation (or *interclass distance*) are based on distance metrics in multidimensional space, especially the Euclidean distance and its variants, which we discussed extensively above. These measures generally do not utilize much of the probabilistic structure of the classes and therefore do not faithfully represent the degree of overlap of the classes in a statistical sense. The *probabilistic separability measures* represent an attempt to capture that information in the evaluation. There are two related types of probabilistic separability measures, the "probabilistic distances" and the "probabilistic dependencies."

To illustrate what is meant by a "probabilistic distance," consider the two-class problem for which class-conditional pdf's are shown for two different features, $\underline{x}$ and $\underline{y}$, in Fig. 1.14. Let us assume that the *a priori* class probabilities are equal, $P(\underline{c}=1)=P(\underline{c}=2)$. In the first case features (scalars, so we can draw a picture in two dimensions) characterized by random variable $\underline{x}$ are employed, and $f_{\underline{x}|\underline{c}}(x|1)$ and $f_{\underline{x}|\underline{c}}(x|2)$ are well separated with respect to the feature values. The classes appear to be almost fully separable based on these densities. On the other hand, when

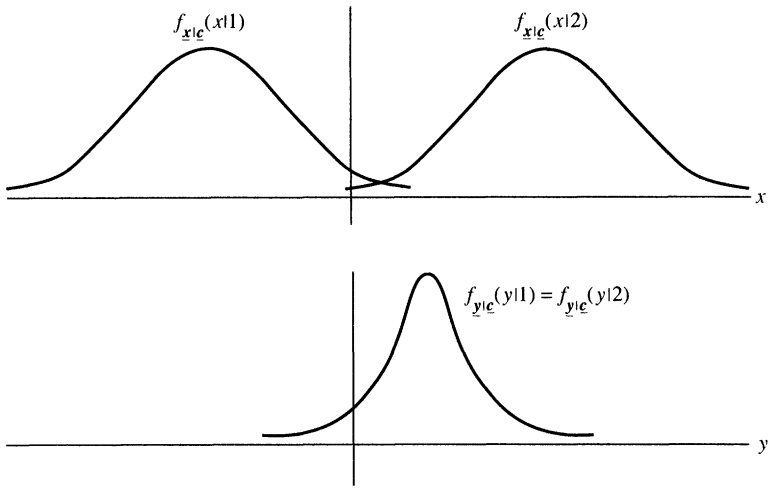


FIGURE 1.14. Class-conditional pdf's for two feature models, $\underline{x}$ and $\underline{y}$, used to introduce the concept of probabilistic distance.

features $\underline{y}$ are used, the separation is extremely poor. In this case $f_{\underline{y}|\underline{c}}(\mathrm{y}|1)$ and $f_{\underline{y}|\underline{c}}(\mathrm{y}|2)$ are identical and the classes would be completely inseparable based on this feature. That is, this feature would provide no better performance than simply guessing, or random assignment, of the class identity.

Probabilistic distance measures attempt to capture the degree of overlap of the class pdf's as a measure of their distance apart. In general, these measures take the form

$$J = \int_{-\infty}^{\infty} g\left\{f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c), P(\underline{c} = c), c = 1, 2, \ldots, K\right\} d\mathbf{x}, \tag{1.214}$$

where $g(\cdot)$ is some function, and $\int_{-\infty}^{\infty}(\cdot)d\mathbf{x}$ indicates the integral over the entire N-dimensional hyperplane with N the dimension of the feature vector $\mathbf{x}$. Probabilistic distance measures have the following properties (Devijver and Kittler, 1982):

1. J is nonnegative, $J \geq 0$.
2. J attains a maximum when all classes in the feature space are disjoint, J is maximum if $f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c) = 0$ when $f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c') \neq 0$ for all $c \neq c'$.
3. $J = 0$ when $f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|1) = f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|2) = \cdots = f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|K)$.

Two examples of probabilistic distance measures for a two-class problem are the *Bhattacharyya distance*,

$$J_B = -\ln \int_{-\infty}^{\infty} \sqrt{f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|1) f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|2)}\, d\mathbf{x}, \tag{1.215}$$

and the *divergence*,

$$J_D = \int_{-\infty}^{\infty} [f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|1) - f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|2)] \ln \frac{f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|1)}{f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|2)} \, d\mathbf{x}, \tag{1.216}$$

both of which reduce to a Mahalanobis-like distance in the case of Gaussian feature vectors and equal class covariances (see Problem 1.19).

Probabilistic Dependence Measures

Another method for indirectly assessing class pdf overlap is provided by the *probabilistic dependence measures*. These measures indicate how strongly the feature outcomes depend upon their class association. In the extreme case in which the features are independent of the class affiliation, the class conditional pdf's are identical to the "mixture" pdf (pdf of the entire universe of feature vectors),

$$f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c) = f_{\underline{\mathbf{x}}}(\mathbf{x}), \qquad \text{for all } c. \tag{1.217}$$

Conversely, when the features depend very strongly on their class association, we expect $f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c)$ to be quite different from the mixture pdf. Therefore, a good indicator of the effectiveness of a set of features at separating the classes is given by the probabilistic dependence measures which quantify the difference between the class conditional pdf's and the mixture pdf. These measures adhere to the same properties noted above for the probabilistic distance measures and are generally of the form

$$J = \int_{-\infty}^{\infty} g\left\{f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c), f_{\underline{\mathbf{x}}}(\mathbf{x}), P(\underline{c} = c), c = 1, 2, \ldots, K\right\} d\mathbf{x}, \tag{1.218}$$

and they adhere to the same properties as those listed above for probabilistic distance measures.

An example of a probabilistic dependence measure that we will encounter in the study of hidden Markov models (Chapter 12) is the *average mutual information*,

$$\overline{M}(\underline{c}, \underline{\mathbf{x}}) = \sum_{c=1}^{K} P(\underline{c} = c) \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c) \log_2 \frac{f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c)}{f_{\underline{\mathbf{x}}}(\mathbf{x})} \, d\mathbf{x} \tag{1.219}$$

where the integral is taken over the multidimensional feature space. We note that if $\mathbf{x}$ takes only a finite number of values, say $\{\mathbf{x}_1, \ldots, \mathbf{x}_L\}$, then (1.219) becomes

$$\overline{M}(\underline{c}, \underline{\mathbf{x}}) = \sum_{c=1}^{K} P(\underline{c} = c) \sum_{l=1}^{L} P(\underline{\mathbf{x}} = \mathbf{x}_l | \underline{c} = c) \log_2 \frac{P(\underline{\mathbf{x}} = \mathbf{x}_l | \underline{c} = c)}{P(\underline{\mathbf{x}} = \mathbf{x}_l)}$$

$$= \sum_{c=1}^{K} \sum_{l=1}^{L} P(\underline{\mathbf{x}} = \mathbf{x}_l, \underline{c} = c) \log_2 \frac{P(\underline{\mathbf{x}} = \mathbf{x}_l, \underline{c} = c)}{P(\underline{\mathbf{x}} = \mathbf{x}_l) P(\underline{c} = c)}. \tag{1.220}$$

This measure, which can be seen to be an indicator of the average deviation of $f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|c)$ from $f_{\underline{\mathbf{x}}}(\mathbf{x})$ [or $P(\underline{\mathbf{x}}|\underline{c})$ from $P(\underline{\mathbf{x}})$] will be given another interpretation when we discuss entropy concepts in Section 1.5.

Entropy Measures

Entropy measures are based on information-theoretic concepts that quantify the amount of uncertainty associated with the outcome of an experiment. In the pattern recognition context, these measures relate how much uncertainty remains about the class membership once a feature measurement is made. This knowledge quantifies the effectiveness of a set of features at conveying information that assists classification. Although we will have no direct use for entropy measures in this book, we will have several occasions to use the concepts of information and entropy. We will therefore address these issues in the next section, and, for completeness, include some comments on entropy measures in pattern recognition at the end of that section.

1.3.5 Clustering Algorithms

The previous discussions were based on the assumption that labeled (according to class) training features were available from which to infer the underlying probability structure of the classes. In some problems, however, information about the class membership of the training vectors is not provided. It is possible that we might not even know the number of classes represented by the training features. The problem of automatically separating training data into groups representing classes is often solved by a *clustering algorithm.*

The process of clustering is part of a more general group of techniques commonly referred to as *unsupervised learning.* As the name would imply, unsupervised learning techniques are concerned with the problem of forming classes from training data without benefit of supervision regarding class membership. Within this group of techniques, clustering algorithms represent a rather *ad hoc* approach to learning classes, which do not attempt to employ deep analysis of the statistical structure of the data. The more formal unsupervised learning methods are called *mode separation techniques* (Devijver and Kittler, 1982, Ch. 10), and we shall not have any use for these methods in our study of speech. Rather, clustering methods are based on the heuristic argument that vectors representing the same class should be "close" to one another in the feature space and "far" from vectors representing other classes. Accordingly, one of the distance metrics discussed above is usually employed in the analysis.

There are two basic classes of clustering algorithms. In *dynamic clustering*, a fixed number of clusters (classes) is used. At each iteration, feature vectors are reassigned according to certain rules until a stable partitioning of the vectors is achieved. We give an important example below. In *hierarchical clustering*, each feature vector is initially a separate cluster, then at each step of the algorithm, the two most similar clusters (according to some similarity criteria) are merged until the desired number of clusters is achieved.

There are a variety of clustering algorithms, but we focus on only one example of an iterative approach which is widely used in speech processing for a number of tasks. This is usually called the *K-means algorithm*, but the "K" simply refers to the number of desired classes and can be replaced by any desired index. The operation of the K-means algorithm is straightforward. Feature vectors are continuously reassigned to clusters, and the cluster centroids updated, until no further reassignment is necessary. The algorithm is given in Fig. 1.15.

The version of K-means given here is sometimes called the *isodata algorithm*. It is different from the original K-means algorithm in that it reassigns the entire set of training vectors before updating the cluster centroids. If means are recomputed after each vector is considered, then the algorithm terminates only after a complete scan of the training set is made without reassignment.

FIGURE 1.15. The K-means algorithm.

Initialization: Choose an arbitrary partition of the training vectors $\{\mathbf{x}\}$ into K clusters, denoted $\Lambda_k, k = 1, 2, \ldots, K$, and compute the mean vector (centroid) of each cluster, $\bar{\mathbf{x}}_k, k = 1, 2, \ldots, K$.

Recursion:

1. For each feature vector, $\mathbf{x}$, in the training set, assign $\mathbf{x}$ to Λ_{k^*}, where

$$k^* = \underset{k}{\operatorname{argmin}}\ d(\mathbf{x}, \bar{\mathbf{x}}_k). \tag{1.221}$$

   $d(\cdot, \cdot)$ represents some distance measure in the feature space.
2. Recompute the cluster centroids, and return to Step 1 if any of the centroids change from the last iteration.

A brief history and more details of the K-means approach from an information theory perspective is given in the paper by Makhoul et al. (1985). In an unpublished 1957 paper [more recently published, see (Lloyd, 1982)], Lloyd, independently of the pattern recognition research efforts, had essentially worked out the isodata algorithm for scalar quantization in pulse code modulation. The generalization of the K-means algorithm to "vector quantization," a technique which we will first encounter in Chapter 7, is sometimes called the *generalized Lloyd algo-*

rithm (Gray, 1984). A further generalization involves the fact that the K-means approach can also be applied to representations of the clusters other than centroids, and to measures of similarities other than distance metrics (Devijver and Kittler, 1982). A measure of similarity which does not necessarily adhere to the formal properties of a distance metric is often called a *distortion measure.* Linde et al. (1980) were the first in the communications literature to suggest the use of vector quantization with K-means and nonmetric distortion measures. Consequently, the K-means algorithm (particularly with these generalizations) is frequently called the *Linde–Buzo–Gray* or *LBG algorithm* in the speech processing and other communications literature.

Generally, the objective of the LBG algorithm is to find a set of, say, K feature vectors (*codes*) into which all feature vectors in the training set can be "quantized" with minimum distortion. This is like adjusting the levels of a scalar quantizer to minimize the amount of distortion incurred when a signal is quantized. This set of code vectors comprises a *codebook* for the feature space. The method is generally described in Fig. 1.16. A slight variation on the LBG method is also shown in Fig. 1.16, which differs in the way in which the algorithm is initialized. In the latter case, the number of clusters is iteratively built up to a desired number (power of two) by "splitting" the existing codes at each step and using these split codes to seed the next iteration.

FIGURE 1.16. The generalized Lloyd or Linde–Buzo–Gray (LBG) algorithm.

Initialization: Choose an arbitrary set of K code vectors, say $\bar{\mathbf{x}}_k$, $k = 1, 2, \ldots, K$.

Recursion:

1. For each feature vector, $\mathbf{x}$, in the training set, "quantize" $\mathbf{x}$ into code $\bar{\mathbf{x}}_{k^*}$, where

$$k^* = \underset{k}{\operatorname{argmin}}\ d(\mathbf{x}, \bar{\mathbf{x}}_k). \tag{1.222}$$

   Here $d(\cdot, \cdot)$ represents some distortion measure in the feature space.
2. Compute the total distortion that has occurred as a result of this quantization,

$$D = \sum d[\mathbf{x}, Q(\mathbf{x})], \tag{1.223}$$

   where the sum is taken over all vectors $\mathbf{x}$ in the training set, and $Q(\mathbf{x})$ indicates the code to which $\mathbf{x}$ is assigned in the current iteration. (This is an estimate of $\mathcal{E}\{d[\mathbf{x}, Q(\mathbf{x})]\}$.) If D is sufficiently small, STOP.
3. For each k, compute the centroid of all vectors $\mathbf{x}$ such that $\bar{\mathbf{x}}_k = Q(\mathbf{x})$ during the present iteration. Let this new set of centroids comprise the new codebook, and return to Step 1.

Alternative LBG algorithm with "centroid splitting."

Initialization: Find the centroid of the entire population of vectors. Let this be the (only) initial code vector.

Recursion. There are I total iterations where 2^I code vectors are desired. Let the iterations be $i = 1, 2, \ldots, I$. For iteration i,

1. "Split" any existing code vector, say $\bar{\mathbf{x}}$, into two codes, say $\bar{\mathbf{x}}(1+\varepsilon)$ and $\bar{\mathbf{x}}(1-\varepsilon)$, where ε is a small number, typically 0.01. This results in 2^i new code vectors, say $\bar{\mathbf{x}}_k^i, k = 1, 2, \ldots, 2^i$.
2. For each feature vector, $\mathbf{x}$, in the training set, "quantize" $\mathbf{x}$ into code $\bar{\mathbf{x}}_{k^*}^i$, where $k^* = \underset{k}{\text{argmin}}\, d\left(\mathbf{x}, \bar{\mathbf{x}}_k^i\right)$. Here $d(\cdot, \cdot)$ represents some distortion measure in the feature space.
3. For each k, compute the centroid of all vectors $\mathbf{x}$ such that $\bar{\mathbf{x}}_k^i = Q(\mathbf{x})$ during the present iteration. Let this new set of centroids comprise the new codebook, and, if $i < I$, return to Step 1.

1.4 Information and Entropy

Reading Note: The material in this section will not be needed until Parts IV and V of the text.

The issues discussed here are a few necessary concepts from the field of information theory. The reader interested in this field should consult one of many widely used books on this subject (see Appendix 1.D).

Note that our need for this material in this text will usually occur in cases in which all random vectors (or variables) take discrete values. We will therefore focus on such cases. Similar definitions and developments exist for continuous random vectors [e.g., (Papoulis, 1984)].

1.4.1 Definitions

At a rudimentary level, the field of information theory is concerned with the amount of uncertainty associated with the outcome of an experiment. Once the experiment is performed and the outcome is known, the uncertainty is dispelled. The amount of information we receive when the outcome is known depends upon how much uncertainty there was about its occurrence.

In the pattern recognition problem above, for example, learning which of the K classes (e.g., words) represents the correct answer is informative. How uncertain we were before the answer was revealed (and therefore how much information we receive) depends on the probability distribution of the classes. For example, consider the extreme cases,

$$P(\underline{c} = c) = \frac{1}{K}, \qquad \text{for all } c \tag{1.224}$$

and

$$P(\underline{c} = c) = \begin{cases} 1, & c = c' \\ 0, & c \neq c'. \end{cases} \tag{1.225}$$

In the first case in which the class probabilities are uniformly distributed, we have complete uncertainty about the association of a given feature vector, and gain the maximum information possible (on the average) when its true association is revealed. On the other hand, in the second case we have no doubt that true class is c', and no information is imparted with the revelation of the class identity. In either case, the information we receive is in indirect proportion to the probability of the class.[35]

The same intuitive arguments apply, of course, to the outcomes of any random variable—the quantity $\underline{c}$ need not model class outcomes in a pattern recognition problem. Let us therefore begin to view $\underline{c}$ as a general discrete random variable. In fact, for even broader generality, let us begin to work with a random *vector*, $\underline{\mathbf{c}}$, recognizing, of course, that the scalar random variable is a special case. According to the notion that information is inversely proportional to outcome likelihood, Shannon (1948) proposed the following formalism. We define the *information* associated with a particular outcome, $\mathbf{c}$, of a discrete random vector, $\underline{\mathbf{c}}$, to be

$$I(\underline{\mathbf{c}} = \mathbf{c}) \stackrel{\text{def}}{=} \log_2 \frac{1}{P(\underline{\mathbf{c}} = \mathbf{c})} = -\log_2 P(\underline{\mathbf{c}} = \mathbf{c}). \tag{1.226}$$

The information is a measure of uncertainty associated with outcome $\mathbf{c}$-the less likely is the value $\mathbf{c}$, the more information we receive. Although information may be defined using any logarithmic base, usually base two is used, in which case $I(\cdot)$ is measured in *bits*. The sense of this term is as follows: If there are K *equally likely* outcomes, say $\mathbf{c}_1, \ldots, \mathbf{c}_K$, and each is assigned an integer $1, 2, \ldots, K$, then it requires a binary number with $\log_2 K$ bits to identify the index of a particular outcome. In this case, we receive exactly that number of bits of information when it is revealed that the true outcome is $\mathbf{c}$,

$$I(\underline{\mathbf{c}} = \mathbf{c}) = -\log_2 P(\underline{\mathbf{c}} = \mathbf{c}) = \log_2 K. \tag{1.227}$$

$I(\underline{\mathbf{c}} = \mathbf{c})$ can therefore be interpreted as the number of binary digits required to identify the outcome $\mathbf{c}$ if it is one of $2^{I(\underline{\mathbf{c}}=\mathbf{c})}$ equally likely possibilities.

In general, of course, information is a random quantity that depends on the outcome of the random variable. We denote this by writing simply $I(\underline{\mathbf{c}})$. The *entropy* is a measure of *expected* information across all outcomes of the random vector,

[35]According to Papoulis (1981), Planck was the first to describe the explicit relationship between probability and information in 1906.

$$H(\underline{\mathbf{c}}) \stackrel{\text{def}}{=} \mathcal{E}\{I(\underline{\mathbf{c}})\} = -\sum_{l=1}^{K} P(\underline{\mathbf{c}} = \mathbf{c}_l) \log_2 P(\underline{\mathbf{c}} = \mathbf{c}_l). \tag{1.228}$$

Now consider N random vectors, say $\underline{\mathbf{x}}(1), \ldots, \underline{\mathbf{x}}(N)$, each of which produces outcomes from the same finite set,[36] $\{\mathbf{x}_1, \ldots, \mathbf{x}_L\}$. By a natural generalization of the above, the information associated with the revelation that $\underline{\mathbf{x}}(1) = \mathbf{x}_{k_1}, \ldots, \underline{\mathbf{x}}(N) = \mathbf{x}_{k_N}$ is defined as

$$I[\underline{\mathbf{x}}(1) = \mathbf{x}_{k_1}, \ldots, \underline{\mathbf{x}}(N) = \mathbf{x}_{k_N}] \stackrel{\text{def}}{=} -\log_2 P[\underline{\mathbf{x}}(1) = \mathbf{x}_{k_1}, \ldots, \underline{\mathbf{x}}(N) = \mathbf{x}_{k_N}], \tag{1.229}$$

and the entropy associated with these random variables is

$$\begin{aligned} H[\underline{\mathbf{x}}(1), \ldots, \underline{\mathbf{x}}(N)] &\stackrel{\text{def}}{=} \mathcal{E}\{I[\underline{\mathbf{x}}(1), \ldots, \underline{\mathbf{x}}(N)]\} \\ &= -\sum_{l_1=1}^{L} \cdots \sum_{l_N=1}^{L} P[\underline{\mathbf{x}}(1) = \mathbf{x}_{l_1}, \ldots, \underline{\mathbf{x}}(N) = \mathbf{x}_{l_N}] \\ &\quad \times \log_2 P[\underline{\mathbf{x}}(1) = \mathbf{x}_{l_1}, \ldots, \underline{\mathbf{x}}(N) = \mathbf{x}_{l_N}]. \end{aligned} \tag{1.230}$$

$I[\underline{\mathbf{x}}(1), \ldots, \underline{\mathbf{x}}(N)]$ and $H[\underline{\mathbf{x}}(1), \ldots, \underline{\mathbf{x}}(N)]$ are called the *joint information* and *joint entropy*, respectively. If random vectors $\underline{\mathbf{x}}(1), \ldots, \underline{\mathbf{x}}(N)$ are *independent*, then

$$I[\underline{\mathbf{x}}(1), \ldots, \underline{\mathbf{x}}(N)] = \sum_{n=1}^{N} I[\underline{\mathbf{x}}(n)] \tag{1.231}$$

and

$$H[\underline{\mathbf{x}}(1), \ldots, \underline{\mathbf{x}}(N)] = \sum_{n=1}^{N} H[\underline{\mathbf{x}}(n)]. \tag{1.232}$$

In particular, if $\underline{\mathbf{x}}(1), \ldots, \underline{\mathbf{x}}(N)$ are *independent and identically distributed*, then

$$H[\underline{\mathbf{x}}(1), \ldots, \underline{\mathbf{x}}(N)] = NH[\underline{\mathbf{x}}(n)] \quad \text{for arbitrary } n. \tag{1.233}$$

Intuitively, the information received when we learn the outcome, say $\mathbf{x}_k$, of a random vector, $\underline{\mathbf{x}}$, will be less if we already know the outcome, say $\mathbf{y}_l$, of a correlated random vector, $\underline{\mathbf{y}}$. Accordingly, we define the *conditional information* and *conditional entropy*, respectively, as

$$I(\underline{\mathbf{x}} = \mathbf{x}_k | \underline{\mathbf{y}} = \mathbf{y}) = -\log_2 P(\underline{\mathbf{x}} = \mathbf{x}_k | \underline{\mathbf{y}} = \mathbf{y}) \tag{1.234}$$

[36]This definition is easily generalized to the case in which all random vectors have different sets of outcomes, but we will not have need of this more general case.

and

$$H(\underline{\mathbf{x}}|\underline{\mathbf{y}}) = \mathcal{E}_{\underline{\mathbf{x}},\underline{\mathbf{y}}}\{I(\underline{\mathbf{x}}|\underline{\mathbf{y}})\} \tag{1.235}$$

$$= \sum_{k=1}^{K}\sum_{l=1}^{L} P(\underline{\mathbf{x}} = \mathbf{x}_l, \underline{\mathbf{y}} = \mathbf{y}_k)\log_2 P(\underline{\mathbf{x}} = \mathbf{x}_l|\underline{\mathbf{y}} = \mathbf{y}_k),$$

where $\mathcal{E}_{\underline{\mathbf{x}},\underline{\mathbf{y}}}$ denotes the expectation with respect to both random vectors $\underline{\mathbf{x}}$ and $\underline{\mathbf{y}}$ and where we have assumed that $\underline{\mathbf{y}}$ takes discrete values $\{\mathbf{y}_1, \ldots, \mathbf{y}_K\}$.

Finally, we need to introduce the notion of "mutual information." The pairing of random vector outcomes intuitively produces less information than the sum of the individual outcomes *if the random vectors are not independent*. Upon arriving at the airport, we receive less information when the ticket agent tells us that (1) we missed the plane, and (2) the plane left 15 minutes ago, than either of those pieces of information would provide individually. This is because the two pieces of information are related and contain information about each other. Formally, this means that

$$I(\underline{\mathbf{x}}, \underline{\mathbf{y}}) \leq I(\underline{\mathbf{x}}) + I(\underline{\mathbf{y}}). \tag{1.236}$$

The "shared" information that is inherent in either of the individual outcomes is called the *mutual information* between the random vectors,

$$M(\underline{\mathbf{x}}, \underline{\mathbf{y}}) \stackrel{\text{def}}{=} [I(\underline{\mathbf{x}}) + I(\underline{\mathbf{y}})] - I(\underline{\mathbf{x}}, \underline{\mathbf{y}}). \tag{1.237}$$

It follows from the definitions above that

$$M(\underline{\mathbf{x}}, \underline{\mathbf{y}}) = \log_2 \frac{P(\underline{\mathbf{x}}, \underline{\mathbf{y}})}{P(\underline{\mathbf{x}})P(\underline{\mathbf{y}})}. \tag{1.238}$$

Equation (1.238), in turn, leads to the conclusion that

$$M(\underline{\mathbf{x}}, \underline{\mathbf{y}}) = I(\underline{\mathbf{x}}) - I(\underline{\mathbf{x}}|\underline{\mathbf{y}}) = I(\underline{\mathbf{y}}) - I(\underline{\mathbf{y}}|\underline{\mathbf{x}}). \tag{1.239}$$

This result clearly shows the interpretation of the mutual information as the information that is "shared" by the random vectors.

Like an entropy measure, the *average mutual information*, which we denote $\overline{M}(\underline{\mathbf{x}}, \underline{\mathbf{y}})$, is the *expected* mutual information over all values of the random vectors,

$$\overline{M}(\underline{\mathbf{x}}, \underline{\mathbf{y}}) \stackrel{\text{def}}{=} \mathcal{E}_{\underline{\mathbf{x}},\underline{\mathbf{y}}}\left\{\log_2 \frac{P(\underline{\mathbf{x}}, \underline{\mathbf{y}})}{P(\underline{\mathbf{x}})P(\underline{\mathbf{y}})}\right\}$$

$$= \sum_{k=1}^{K}\sum_{l=1}^{L} P(\underline{\mathbf{x}} = \mathbf{x}_l, \underline{\mathbf{y}} = \mathbf{y}_k)\log_2 \frac{P(\underline{\mathbf{x}} = \mathbf{x}_l, \underline{\mathbf{y}} = \mathbf{y}_k)}{P(\underline{\mathbf{x}} = x_l)P(\underline{\mathbf{y}} = \mathbf{y}_k)}. \tag{1.240}$$

Note from (1.238) and (1.240) that

$$\overline{M}(\underline{\mathbf{x}}, \underline{\mathbf{y}}) = H(\underline{\mathbf{x}}) - H(\underline{\mathbf{x}}|\underline{\mathbf{y}}) = H(\underline{\mathbf{y}}) - H(\underline{\mathbf{y}}|\underline{\mathbf{x}}), \tag{1.241}$$

which immediately leads to the conclusion that if $\underline{\mathbf{x}}$ and $\underline{\mathbf{y}}$ are independent, then there is no average mutual information (no "shared" information on the average).

We will see entropy concepts play a role in several areas of speech coding and recognition in Chapters 7 and 12. The mutual information will be used in an important speech recognition technique in Chapter 12. We illustrate the use of some entropy concepts in pattern recognition in Section 1.4.3.

1.4.2 Random Sources

In several places in our work, we will need to characterize the information conveyed by a (vector) random *process*, say

$$\underline{\mathbf{x}} = \{\ldots, \underline{\mathbf{x}}(-1), \underline{\mathbf{x}}(0), \underline{\mathbf{x}}(1), \ldots\}, \tag{1.242}$$

in which each random vector takes a finite number of discrete outcomes, say $\mathbf{x}_1, \ldots, \mathbf{x}_L$. In communications applications, the random process will often characterize the output of a transmitter where it is given the name *random source.* Nevertheless, from a mathematical point of view, a random source is equivalent to a random process.

How then do we characterize the information from a random source? The usual method is to indicate the entropy per sample (per random vector), which, if the process is[37] *stationary with independent random vectors*, is equivalent to the entropy associated with *any* random vector,

$$H(\underline{\mathbf{x}}) \stackrel{\text{def}}{=} H[\underline{\mathbf{x}}(n)] = -\sum_{l=1}^{L} P[\underline{\mathbf{x}}(n) = \mathbf{x}_l] \log_2 P[\underline{\mathbf{x}}(n) = \mathbf{x}_l]. \tag{1.243}$$

However, if the random vectors are not independent, then we must use[38]

$$H(\underline{\mathbf{x}}) \stackrel{\text{def}}{=} -\lim_{N\to\infty} \sum_{l_1=1}^{L} \cdots \sum_{l_N=1}^{L} P[\underline{\mathbf{x}}(1) = \mathbf{x}_{l_1}, \ldots, \underline{\mathbf{x}}(N) = \mathbf{x}_{l_N}] \times \log_2 P[\underline{\mathbf{x}}(1) = \mathbf{x}_1, \ldots, \underline{\mathbf{x}}(N) = \mathbf{x}_{l_N}]. \tag{1.244}$$

If the random vectors are uncorrelated beyond some finite N, then the expression need not contain the limit. Definition (1.244) is useful for theoretical discussions, but it becomes practically intractable for N's

[37]A stationary source with discrete, independent random variables (or vectors) is called a *discrete memoryless source* in the communications field [see, e.g., (Proakis, 1989, Sec. 2.3.2)].

[38]We assume here that the random process starts at $n = 0$.

much larger than two or three. We will see one interesting application of this expression in our study of language modeling in Chapter 13.

1.4.3 Entropy Concepts in Pattern Recognition

Entropy measures are used in pattern recognition problems. To provide an example of the use of the entropy concepts described above, and also to provide closure to our discussion of probabilistic separability measures, we briefly consider that task here. The material here will turn out to be very similar to a development needed in our study of speech recognition. One disclaimer is in order before we begin the discussion. Because we have only studied entropy concepts for the case of *discrete* conditioning vectors, we will only consider the case of discrete feature vectors here. This is consistent with our need for this material in the text, but is at variance with our discussion of features in Section 1.4. The more general case is found in (Devijver and Kittler, 1982, Sec. 5.3.5).

Generalized entropy measures are used to assess the effectiveness of a set of features at pattern classification. The properties of such measures are quite complex, and are described, for example, in (Devijver and Kittler, 1982, App. C). A special case of a generalized entropy measure is what we would simply call the conditional entropy for the set of classes, characterized by random variable $\underline{c}$, conditioned upon knowledge of a feature vector, modeled by random vector $\underline{\mathbf{x}}$. From (1.235)

$$H(\underline{c}|\underline{\mathbf{x}}) = \sum_{c=1}^{K} \sum_{l=1}^{L} P(\underline{c} = c, \underline{\mathbf{x}} = \mathbf{x}_l) \log_2 P(\underline{c} = c|\underline{\mathbf{x}} = \mathbf{x}_l). \tag{1.245}$$

This quantity provides a measure of the average quality of the chosen features over the entire feature space. $H(\underline{c}|\mathbf{x})$ is sometimes called the *equivocation*. We would want the equivocation to be small, meaning that, on the average, the feature vector $\mathbf{x}$ greatly reduces the uncertainty about the class identity.

A related way to view the feature effectiveness is to examine the average mutual information between the random variable $\underline{c}$ and random vector $\underline{\mathbf{x}}$, $\overline{M}(\underline{c}, \underline{\mathbf{x}})$. Ideally, this measure is large, meaning that a given feature outcome contains a significant amount of information about the class outcome. From (1.240) we can write

$$\overline{M}(\underline{c}, \underline{\mathbf{x}}) = \sum_{c=1}^{K} \sum_{l=1}^{L} P(\underline{c} = c, \underline{\mathbf{x}} = \mathbf{x}_l) \log_2 \frac{P(\underline{c} = c, \underline{\mathbf{x}} = \mathbf{x}_l)}{P(\underline{c} = c) P(\underline{\mathbf{x}} = \mathbf{x}_l)}. \tag{1.246}$$

The reader can confirm that this measure is identical to (1.220) discussed above.

It might also be of interest to characterize the average mutual information between two *jointly stationary random sources*, say $\underline{\mathbf{x}}$ and $\underline{\mathbf{y}}$. By this

we will simply mean the average mutual information between two random variables at any arbitrary time n. We will write $\overline{M}(\underline{\mathbf{x}}, \underline{\mathbf{y}})$ to emphasize that the random variables are taken from the stationary random sources,

$$\overline{M}(\underline{\mathbf{x}}, \underline{\mathbf{y}}) \stackrel{\text{def}}{=} \sum_{l=1}^{L} \sum_{k=1}^{K} P(\underline{\mathbf{x}} = \mathbf{x}_l, \underline{\mathbf{y}} = \mathbf{y}_k) \log_2 \frac{P(\underline{\mathbf{x}} = \mathbf{x}_l, \underline{\mathbf{y}} = \mathbf{y}_k)}{P(\underline{\mathbf{x}} = \mathbf{x}_l) P(\underline{\mathbf{y}} = \mathbf{y}_k)}. \tag{1.247}$$

Here we have assumed that the sources are vector random processes that take discrete vector values. Versions of (1.247) for other cases, for example, scalar random processes that take continuous values, require obvious modifications [cf. (1.219)].

1.5 Phasors and Steady-State Solutions

In our work with analog acoustic modeling of the speech production system, we will be concerned with the solution of a linear, constant coefficient, differential equation (LCCDE). This short section is intended to remind the reader of some commonly used techniques and notation.

Consider a continuous time system described by an LCCDE

$$\sum_{i=1}^{n} a_i \frac{d^i}{dt^i} y(t) = \sum_{i=1}^{m} b_i \frac{d^i}{dt^i} x(t), \tag{1.248}$$

where $x(t)$ and $y(t)$ are the input and output, respectively. It is often desired to know the *steady-state* response of the system (response after all transients have diminished) to a sinusoidal input, say $x(t) = X \cos(\Omega t + \varphi_x)$. It is frequently convenient to replace the cosine (or sine) by a complex exponential,

$$x(t) = Xe^{j(\Omega t + \varphi_x)}, \tag{1.249}$$

recognizing that the solutions to the real (cosine) and imaginary (sine) parts of the exponential will remain separated in the solution because of linearity. Further, it is also frequently useful to rewrite (1.249) as

$$x(t) = \overline{X} e^{j\Omega t}, \tag{1.250}$$

where $\overline{X}$ *is the complex number*

$$\overline{X} = Xe^{j\varphi_x}, \tag{1.251}$$

which is called a *phasor* for the exponential signal $x(t)$. Also due to linearity, we know that the input (1.249) will produce an output of the form $y(t) = Ye^{j(\Omega t + \varphi_y)}$, which may also be written in terms of a phasor,

$$y(t) = \overline{Y}e^{j\Omega t}, \tag{1.252}$$

where $\overline{Y} = Ye^{j\varphi_y}$. Putting the forms (1.250) and (1.252) into (1.248), it is found immediately that the terms $e^{j\Omega t}$ cancel and the differential equation solution reduces to one of solving an algebraic equation for $\overline{Y}$ in terms of $\overline{X}$, powers of Ω, and the coefficients a_i and b_i. Engineers often take advantage of this fact and solve algebraic phasor equations directly for steady-state solutions, sidestepping the differential equations completely. We have developed constructs such as "impedance" to assist in these simplified solutions (see below).

In fact, recall that phasor analysis amounts to steady-state frequency domain analysis. In principle, the phasors $\overline{X}$ and $\overline{Y}$ are frequency dependent, because we may enter a variety of inputs (actually an uncountably infinite number!) of the form $x(t) = X\cos(\Omega t + \varphi_x)$, each with a different frequency, Ω; amplitude, X; and phase, φ_x, to produce corresponding outputs of form $y(t) = Y\cos(\Omega t + \varphi_y)$ with frequency-dependent amplitudes and phases. We may reflect this fact by writing the phasors as $\overline{X}(\Omega)$ and $\overline{Y}(\Omega)$. Plugging forms (1.250) and (1.252) into (1.248) with these explicitly frequency-dependent phasors immediately produces the general expression for the output phasor

$$\overline{Y}(\Omega) = \frac{\sum_{i=0}^{m} b_i\Omega^i}{1 + \sum_{i=1}^{n} a_i\Omega^i}\overline{X}(\Omega). \tag{1.253}$$

The ratio $H(\Omega) \overset{\text{def}}{=} \overline{Y}(\Omega)/\overline{X}(\Omega)$, is of course the (Fourier) transfer function for the system. Other ratios, in particular, impedances and admittances, result from similar analyses. If, for example, $y(t)$ is a voltage across a discrete electrical component in response to current $x(t)$, then the phasor ratio $Z(\Omega) = \overline{Y}(\Omega)/\overline{X}(\Omega)$ resulting from the (usually simple) differential equation governing the component is the *impedance* (frequency dependent) of that component. The algebraic equations resulting from phasor-based solutions of differential equations mimic the simple "Ohm's law" type relations that arise in DC analysis of resistive circuits. As electrical engineers, we sometimes become so familiar with these simple phasor techniques that we forget their fundamental connection to the underlying differential equation.

In connection with the concepts above, we note that the ratio of phasors is always equivalent to the ratio of *complex* signals they represent,

$$\frac{\overline{Y}(\Omega)}{\overline{X}(\Omega)} = \frac{\overline{Y}(\Omega)e^{j\Omega t}}{\overline{X}(\Omega)e^{j\Omega t}} = \frac{y(t)}{x(t)}. \tag{1.254}$$

This fact is sometimes useful in theoretical discussions in which phasor notations have not been defined for certain signals.

We will make use of these ideas in our early work (Chapter 3) concerning analog acoustic modeling of the speech production system. If necessary, the reader should review these topics in any of a number of engineering textbooks [e.g., (Hayt and Kimmerly, 1971)] or textbooks on differential equations [e.g., (Boyce and DiPrima, 1969)].

1.6 Onward to Speech Processing

Thus ends our review and tutorial of selected background material prerequisite to the study of speech processing. The reader will probably want to refer back to this chapter frequently to recall notational conventions and basic analytical tools. Before beginning our formal study, we make a few introductory comments about the speech processing field, and about the organization of the book.

Brief History. The history of speech processing certainly does not begin with the digital signal processing engineer, nor even with the work of electrical engineers. In an interesting article[39] surveying some of the history of speech synthesis, Flanagan (1972) notes humankind's fascination with speech and voice from ancient times, and places the advent of the scientific study of speech in the Rennaisance when clever mechanical models were constructed to imitate speech. The first well-documented efforts at mechanical speech synthesis occurred in St. Petersburg and Vienna in the late eighteenth century. The 1930s, a century and a half later, are often considered to be the beginning of the modern speech technology era, in large part due to two key developments at Bell Laboratories. The first was the development of pulse code modulation (PCM), the first digital representation of speech (and other waveforms) which helped to pioneer the field of digital communications. The second was the demonstration of the Vocoder (*Voice Coder*) by Dudley (1939), a speech synthesizer, the design of which first suggested the possibility of parametric speech representation and coding. The subsequent decades have seen an explosion of activity roughly concentrated into decades. We mention a few key developments: intense research on the basic acoustical aspects of speech production and concomitant interest in electronic synthesizers in the late 1940s through the 1960s (Fant, 1960), which was spurred on by the invention of the spectrograph in 1946 (Potter et al., 1966); advances in analysis and coding algorithms (linear prediction, cepstrum) in the 1960s (see Chapters 5 and 6 in this book) made possible by the new digital computing machines and related work in digital signal processing [e.g., (Cooley and Tukey, 1965)]; development of temporally adaptive speech coding algorithms in the 1970s (see Chapter 7); and vast

[39]Also see (Schroeder, 1966). Each of these papers, as well as others describing early work, are reprinted in (Schafer and Markel, 1979).

interest in speech recognition research in the 1970s and 1980s and continuing into the 1990s, grounded in the development of dynamic programming techniques, hidden Markov modeling, vector quantization, neural networks, and significant advances in processor architectures and fabrication (see the chapters of Part V).

Research Areas and Text Organization. There is no precise way to partition the speech processing research field into its component areas. Nevertheless, we offer the following first approximation to a partition that can roughly be inferred from the discussion above:

Speech Science (Speech Production and Modeling) (Part II of this book)
Analysis (Part III)
Coding, Synthesis, Enhancement, and Quality Assessment (Part IV)
Recognition (Part V)

We have organized the book around these themes.

Part II is concerned with providing necessary topics in speech science and with early efforts to model speech production, which are grounded in the physics of the biological system. By *speech science* we mean the use of engineering techniques—spectral analysis, modeling, and so on—in work that is specifically aimed at a better understanding of the physiological mechanisms, anatomy, acoustic, phonetic, and linguistic aspects of normal and abnormal voice and speech production. Naturally, such work is highly interdisciplinary and is least concerned with immediate application of the research results. Needless to say, however, speech science research has been, and continues to be, central to progress in the more applied fields. In Chapter 2, the first chapter in Part II, we examine speech science concepts necessary to "engineer" speech. Our goal is to learn enough about speech to be able to converse with interdisciplinary researchers in various aspects of speech science and speech processing, and to be able to build useful mathematical models of speech production. Chapter 3 begins the quest for a useful mathematical model by building on the science of speech production discussed in Chapter 2. The journey takes us through a discussion of fundamental attempts to model speech production based on the physics of acoustic tubes. These real acoustic models are revealing and provide a firm foundation for the widely used discrete time model, which will be employed throughout the remainder of the book and whose description is the culmination of the chapter.

Speech *analysis* research is concerned with processing techniques that are designed to extract information from the speech waveform. In Part III we take up the most important contemporary tools for analyzing speech by computer. Speech is analyzed for many reasons, including analysis for analysis' sake (basic research into phonetics or better models of speech production), but also to reduce it to basic features for coding, synthesis, recognition, or enhancement. Part III of the book, therefore,

comprises the engineering foundation upon which speech processing is built. In the first of these topics (Chapter 4) we examine the general issue of processing short terms of a signal. Most engineering courses ignore the fact that, in the real world, only finite lengths of signals are available for processing. This is particularly true in speech where the signal remains stationary for only milliseconds. The remaining chapters (5 and 6) of Part III introduce the two most important parameterizations of speech in contemporary processing—linear prediction coefficients and cepstral coefficients—their meaning, and the analysis techniques for obtaining them. These parameters are widely used for spectral representations of speech in the areas mentioned above. We shall therefore use them repeatedly as we progress through the material.

Part IV consists of three chapters that cover a rather wide range of topics. This part of the text is concerned with those aspects of speech processing which most directly intersect with the communications technologies. Here we will be concerned with efficient coding for the transmission of speech across channels and its reconstruction at the receiver site. Since the task of synthesis is closely coupled with transmission and reconstruction strategies, we will examine some of the widely used analytical techniques for synthesis in the context of this study. Synthesis for voice response systems, in which a machine is used in place of a human to dispense information, is also an important application domain, and many of the techniques used in communications systems are equally applicable to this problem.

The effectiveness of a coding scheme at preserving the information and the natural quality of the speech can be ascertained by using results from *quality assessment* research. Accordingly, we include this topic in Part IV (Chapter 9). Related to the assessment of quality is the *enhancement* of speech that has been corrupted by any of a number of natural or human-made effects, including coding. This issue will also be addressed in Part IV (Chapter 8).

Speech *recognition* deals with the related problems of designing algorithms that recognize or even understand[40] speech, or which identify the speaker (speech recognition versus speaker recognition).[41] In Part V, we take up the first of these problems, that of recognizing the speech itself. Chapter 10 overviews the problems encountered in trying to recognize speech using a computer. Chapters 11 and 12 introduce the two most widely used techniques for recognizing speech—dynamic time-warping algorithms and the hidden Markov model. The first is a template match-

[40]A speech *recognizer* simply "translates" the message into words, while a speech *understanding* system would be able to ascertain the meaning of the utterance. Speech understanding algorithms can be used as an aid to recognition, by, for example, disallowing nonsensical concatenations of words to be tried, or by "expecting" certain utterances in various conversational contexts.

[41]A slight variation on the latter problem is *speaker verification*, in which the recognizer accepts or rejects the speaker's claim of identity.

ing method following the classical paradigm of statistical pattern recognition with the interesting special problem of time registration of the waveform. The latter is a stochastic method in which statistical characterizations of utterances are automatically learned from training utterances. Chapter 13 introduces the basic principles of language modeling, techniques that reduce entropy by taking advantage of the higher-level structure of spoken utterances to improve recognizer performance. Chapter 14 is a brief introduction to a radically different approach to speech recognition based on massively parallel computing architectures or "artificial neural networks." This field is in its relative infancy compared with techniques based on sequential computing, and it offers interesting challenges and possibilities for future research and development.

Applications. The applications of speech processing are manifold and diverse. In a general way, we have alluded to some of the basic areas above. Among the principal "drivers" of speech processing research in recent years have been the commercial and military support of ambitious endeavors of large scale. These have mainly included speech coding for communications, and speech recognition for an extremely large array of potential applications—robotics, machine data entry by speech, remote control of machines by speech for hazardous or "hands-free" (surgery) environments, communications with pilots in noisy cockpits, and so on. Futuristic machines for human/machine communication and interaction using speech are envisioned (and portrayed in science fiction movies), and in the meantime, more modest systems for recognition of credit card, telephone, and bank account numbers, for example, are in use. In addition, speech processing is employed in "smaller scale" problems such as speaker recognition and verification for military, security, and forensic applications, in biomedicine for the assessment of speech and voice disorders (analysis), and in designing speech and hearing aids for persons with disabilities (analysis and recognition). Inasmuch as speech is the most natural means of communication for almost everyone, the applications of speech processing technology seem nearly limitless, and this field promises to profoundly change our personal and professional lives in coming years.

What Is Not Covered in This Textbook. Speech processing is an inherently interdisciplinary subject. Although the boundaries among academic disciplines are certainly not well defined, this book is written by electrical engineers and tends to focus on topics that have been most actively pursued by digital signal processing engineers.

Significant contributions to this field, especially to speech recognition, have come from research that would usually be classified as computer science. A comprehensive treatment of these "computer science" topics is outside the intended scope of this book. Examples include (detailed discussions of) parsing algorithms for language modeling (see Chapter 13),

and knowledge-based and artificial intelligence approaches to recognition[42] [e.g., (Zue, 1985)]. Although we briefly discuss the former, we do not address the latter. Another example concerns the use of "semantic" and "pragmatic" knowledge in speech recognition (see Chapters 10 and 13). Semantics and pragmatics are subjects that are difficult to formalize in conventional engineering terms, and their complexity has precluded a significant impact on speech recognition technology outside the laboratory. We treat these issues only qualitatively in this book.

The speech (and hearing) science domains—anatomy and physiology of speech production, acoustic phonetics, linguistics, hearing, and psychophysics—are all subjects that are fundamentally important to speech processing. This book provides an essential engineering treatment of most of these subjects, but a thorough treatment of these topics obviously remains beyond the scope of the book. The reader is referred to Appendix 1.E for some resources in the area.

Finally, the explosive growth in this field brought about by digital computing has made it impossible for us to provide a thorough account of the important work in speech processing prior to about 1965. Essential elements of the analog acoustic theory of speech, upon which much of modern speech processing is based, are treated in Chapter 3 and its appendix. A much more extensive treatment of this subject is found in the book *Speech Analysis, Synthesis, and Perception* by J. L. Flanagan (1972). This book is a classic textbook in the field and no serious student of speech processing should be unfamiliar with its contents. Other important papers with useful reference lists can be found in the collection (Schafer and Markel, 1979).

Further Information. The appendixes to this chapter provide the reader with lists of books and other supplementary materials for background and advanced pursuit of the topics in this book. In particular, Section 1.E of this appendix is devoted to materials specifically on speech processing. Among the sections are lists of other textbooks, edited paper collections, journals, and some notes on conference proceedings.

1.7 PROBLEMS

1.1. Whereas the unit step sequence, $u(n)$, can be thought of as samples of the continuous time step, say $u_a(t)$, defined as

$$u_a(t) = \begin{cases} 1, & t \geq 0 \\ 0, & t < 0 \end{cases}, \tag{1.255}$$

[42]This and other papers on knowledge-based approaches are reprinted in (Waibel and Lee, 1990).

a similar relationship does not exist between the discrete-time "impulse," $\delta(n)$ and its continuous-time counterpart $\delta_a(t)$.

(a) Consider sampling the signal $u_a(t)$ with sample period T to obtain the sequence $u(n) \stackrel{\text{def}}{=} u_a(nT)$. If we now subject $u(n)$ to the customary ideal interpolation procedure in an attempt to reconstruct $u_a(t)$ (Proakis and Manolakis, 1992, Sec. 6.3), will the original $u_a(t)$ be recovered? Why or why not?

(b) Roughly sketch the time signal, say $\hat{u}_a(t)$, and the spectrum $|\hat{U}_a(\Omega)|$ of the signal that *will* be recovered in part (a).

(c) That $\delta_a(t)$ cannot be sampled fast enough to preserve the information in the time signal is apparent, since the signal has infinite bandwidth, that is, $\triangle_a(\Omega) = 1$. However, to show that any attempt to sample $\delta_a(t)$ results in an anomalous sequence, consider what happens in the frequency domain with reference to (1.21). What is the anomaly in the time sequence that causes this strange frequency domain result?

(d) Carefully sketch and numerically label the time signal, say $\hat{\delta}_a(t)$, and its spectrum $\hat{\triangle}_a(\Omega)$ that results from an ideal interpolation of the unit sample sequence,

$$\delta(n) = \begin{cases} 1, & n = 0 \\ 0, & \text{otherwise} \end{cases}. \tag{1.256}$$

1.2. Consider the following sequences:

(i) $y(n) = \begin{cases} 1/n, & n > 0 \\ 0, & n \leq 0 \end{cases}$.

(ii) $x(n) = [\sin(\omega_c n)]/\pi n$, restrict ω_c as $0 < \omega_c < \pi$

(a) In each case, classify the sequence according to whether it represents an energy signal, power signal, or neither.

(b) In each case, determine whether the sequence is absolutely summable.

(c) In each case comment on the existence of the DTFT and whether the z-transform ROC includes the unit circle.

1.3. (a) Verify the properties of the DTFT shown in Table 1.1.

(b) Prove Parseval's relation:

$$E_x = \sum_{n=-\infty}^{\infty} |x(n)|^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |X(\omega)|^2 \, d\omega. \tag{1.257}$$

1.4. (a) Verify the properties of the DFT shown in Table 1.2. The notation $W \stackrel{\text{def}}{=} e^{-j2\pi/N}$ is used for convenience and all time sequences are assumed to be of length N.

(b) Prove Parseval's relation:

TABLE 1.1. Properties of the DTFT.

Property	Time Domain	Frequency Domain
Linearity	$ax_1(n) + bx_2(n)$	$aX_1(\omega) + bX_2(\omega)$
Delay	$x(n-d)$	$e^{-j\omega d}X(\omega)$
Modulation	$e^{j\omega_0 n}x(n)$	$X(\omega - \omega_0)$
Time reversal	$x(-n)$	$X(-\omega)$
Multiplication	$x(n)y(n)$	$\frac{1}{2\pi}\int_{-\pi}^{\pi} X(\zeta)Y(\omega-\zeta)\,d\zeta = X(\omega) * Y(\omega)$
Convolution	$x(n) * y(n)$	$X(\omega)Y(\omega)$
Conjugation	$x^*(n)$	$X^*(-\omega)$
Differentiation	$nx(n)$	$j\frac{dX(\omega)}{d\omega}$

$$\sum_{n=0}^{N-1}|x(n)|^2 = \frac{1}{N}\sum_{k=0}^{N-1}|X(k)|^2. \tag{1.258}$$

1.5. Suppose that we redefine state variables in Fig. 1.5. The new set, $\{v_i'(n)\}$, is such that $v_1'(n) = v_N(n)$, $v_2'(n) = v_{N-1}(n), \ldots, v_N'(n) = v_1(n)$, where the $v_i(n)$'s are defined in the figure. Develop an upper companion form state space model of the form

$$\mathbf{v}'(n+1) = \mathbf{A}'\mathbf{v}'(n) + \mathbf{c}'x(n) \tag{1.259}$$

$$y(n) = \mathbf{b}'^T\mathbf{v}'(n) + d'x(n), \tag{1.260}$$

in which $\mathbf{A}'$ is obtained from the lower companion form matrix $\mathbf{A}$ of (1.54) by reflecting all elements around the main diagonal.

1.6. (a) How many sequences are there with P nonzero, finite poles and $Z \leq P$ nonzero, finite zeros that have identical magnitude spectra? The sequences need *not* be real.

(b) How many of these are causal and "stable"? That is, how many are absolutely summable, meaning that their z-transform ROCs include the unit circle? How many are noncausal and "stable"?

(c) If $Z > P$, how do your answers in part (b) change?

TABLE 1.2. Properties of the DFT.*

Property	Time Domain	Frequency Domain
Linearity	$ax_1(n) + bx_2(n)$	$aX_1(k) + bX_2(k)$
Circular shift	$x(n-d)_{\text{mod}N}$	$W^{kd}X(k)$
Modulation	$W^{ln}x(n)$	$X(k+l)_{\text{mod}N}$
Circular convolution	$x(n)_{\text{mod}N} * y(n)$	$X(k)Y(k)$

*The notation $W \stackrel{\text{def}}{=} e^{-j2\pi/N}$ and all sequences are assumed to be of length N.

(d) Of the causal sequences in part (b), how many are minimum phase? Maximum phase?

1.7. (Computer Assignment) Using a signal processing software package, replicate the experiment of Section 1.1.7, that is, reproduce Figures 1.6–1.8.

1.8. Given the joint probability density function $f_{\underline{x}\underline{y}}(x, y)$ for two jointly continuous random variables $\underline{x}$ and $\underline{y}$, verify the following using a pictorial argument:

$$P(x_1 < \underline{x} \leq x_2, y_1 < \underline{y} \leq y_2) = \int_{-\infty}^{x_2} \int_{-\infty}^{y_2} f_{\underline{x}\underline{y}}(x, y)\, dx\, dy - \int_{-\infty}^{x_2} \int_{-\infty}^{y_1} f_{\underline{x}\underline{y}}(x, y)\, dx\, dy - \int_{-\infty}^{x_1} \int_{-\infty}^{y_2} f_{\underline{x}\underline{y}}(x, y)\, dx\, dy + \int_{-\infty}^{x_1} \int_{-\infty}^{y_1} f_{\underline{x}\underline{y}}(x, y)\, dx\, dy. \tag{1.261}$$

1.9. Formally verify that, for two jointly continuous random variables $\underline{x}$ and $\underline{y}$,

$$\mathcal{E}\{\mathcal{E}\{h(\underline{y})|\underline{x}\}\} = \mathcal{E}\{h(\underline{y})\} \tag{1.262}$$

where $h(\cdot)$ is some "well-behaved" function of $\underline{y}$. Assume that all relevant pdf's exist.

1.10. For a random process $\underline{x}$ with random variables $\underline{x}(n)$, show that

$$c_{\underline{x}}(n_1, n_2) = r_{\underline{x}}(n_1, n_2) - \mathcal{E}\{\underline{x}(n_1)\}\, \mathcal{E}\{\underline{x}(n_2)\}. \tag{1.263}$$

1.11. In this problem we will show that the implications in (1.132) reverse in the special case in which random variables within a random process are known to be joint Gaussian. Consider a random process $\underline{x}$, known to be WSS. (*Note*: This means that the mean and correlations are time independent, which we denote by writing $\mu_{\underline{x}}$, $\sigma_{\underline{x}}$, and $\rho_{\underline{x}}$.) If two random variables, $\underline{x}(n_1)$ and $\underline{x}(n_2)$ for any n_1 and n_2, in $\underline{x}$ are joint Gaussian,

$$f_{\underline{x}(n_1)\underline{x}(n_2)}(x_1, x_2) = \frac{1}{2\pi\sigma_{\underline{x}}^2 \sqrt{1-\rho_{\underline{x}}^2}}\, e^{-1/2Q(x_1, x_2)}, \tag{1.264}$$

where

$$Q(x_1, x_2) = \frac{1}{1-\rho_{\underline{x}}^2} \left\{ \left(\frac{x_1 - \mu_{\underline{x}}}{\sigma_{\underline{x}}}\right)^2 - 2\rho_{\underline{x}}^2 \left(\frac{x_1 - \mu_{\underline{x}}}{\sigma_{\underline{x}}}\right)\left(\frac{x_2 - \mu_{\underline{x}}}{\sigma_{\underline{x}}}\right) + \left(\frac{x_2 - \mu_{\underline{x}}}{\sigma_{\underline{x}}}\right)^2 \right\}, \tag{1.265}$$

show that the process is also *second-order stationary.* Show, in fact, that the process is SSS.

1.12. For a WSS random process $\underline{x}$, verify that

$$P_{\underline{x}} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \Gamma_{\underline{x}}(\omega)\, d\omega = \frac{1}{\pi} \int_{0}^{\pi} \Gamma_{\underline{x}}(\omega)\, d\omega = r_{\underline{x}}(0). \tag{1.266}$$

1.13. Show that, if a WSS random process $\underline{x}$, which is ergodic in both mean and autocorrelation, is used as input to a stable, linear, time-invariant discrete time system with impulse response $h(n)$, then the output random process $\underline{y}$ is also ergodic in both senses.

1.14. Verify (1.155) and (1.156).

1.15. Verify (1.157).

1.16. Show that (1.210) follows from (1.209).

1.17. Repeat the analysis leading to (1.190) starting with the basis vectors $\boldsymbol{\beta}_1' = [2\ 0]^T$ and $\boldsymbol{\beta}_2' = [0\ 10]^T$. In this case the basis vectors are orthogonal, but grossly out of scale. This corresponds to the case of two features which are uncorrelated but which have widely different variances. What is the form of the eventual weighting matrix, $\mathbf{W}$, for this case? Is it clear what the weighting matrix is doing "physically?" Explain.

1.18. Consider a nonsingular $N \times N$ matrix $\mathbf{W}$ which operates on three N-vectors, $\mathbf{x}'$, $\mathbf{y}'$, and $\mathbf{z}'$, to produce three new vectors $\mathbf{x} = \mathbf{W}\mathbf{x}'$, $\mathbf{y} = \mathbf{W}\mathbf{y}'$, and $\mathbf{z} = \mathbf{W}\mathbf{z}'$. The original vectors are arbitrary except that $\mathbf{x}'$ is closer to $\mathbf{z}'$ when the Euclidean metric is used to measure distance:

$$d_2(\mathbf{x}', \mathbf{z}') < d_2(\mathbf{y}', \mathbf{z}'). \tag{1.267}$$

Show that the linear transformation need *not* preserve the relative distances by finding a vector triplet satisfying (1.267) and a nonsingular $\mathbf{W}$ such that

$$d_2(\mathbf{x}, \mathbf{z}) \geq d_2(\mathbf{y}, \mathbf{z}). \tag{1.268}$$

Can you find conditions on $\mathbf{W}$ so that the relative distances are preserved?

1.19. An example of a probabilistic separability measure for a two-class problem is the Bhattacharyya distance

$$J_B = -\ln \int_{-\infty}^{\infty} \sqrt{f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|1) f_{\underline{\mathbf{x}}|\underline{c}}(\mathbf{x}|2)}\, d\mathbf{x}. \tag{1.269}$$

Show that this measure reduces to a Mahalanobis-like distance in the case of Gaussian feature vectors and equal class covariances. *Hint*: Use the fact that

$$-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_{\underline{\mathbf{x}}|1})^T \mathbf{C}_{\underline{\mathbf{x}}}^{-1} (\mathbf{x} - \boldsymbol{\mu}_{\underline{\mathbf{x}}|1}) + \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_{\underline{\mathbf{x}}|2})^T \mathbf{C}_{\underline{\mathbf{x}}}^{-1} (\mathbf{x} - \boldsymbol{\mu}_{\underline{\mathbf{x}}|2}) = \tag{1.270}$$

(equation continues next page)

$$\frac{(\mu_{\underline{x}|1} - \mu_{\underline{x}|2})^T}{2} C_{\underline{x}}^{-1} \frac{(\mu_{\underline{x}|1} - \mu_{\underline{x}|2})}{2} + \left[\mathbf{x} - \frac{(\mu_{\underline{x}|1} + \mu_{\underline{x}|2})}{2}\right]^T C_{\underline{x}}^{-1} \left[\mathbf{x} - \frac{(\mu_{\underline{x}|1} + \mu_{\underline{x}|2})}{2}\right].$$

1.20. Two stationary binary sources, $\underline{x}$ and $\underline{y}$, are considered in this problem. In each case the random variables of the source, for example, $\underline{x}(n)$, $n = 1, 2, \ldots,$ are statistically independent.

(a) Given $P[\underline{x}(n) = 1] = 0.3$ for any n, evaluate the entropy of source $\underline{x}$, $H(\underline{x})$.

(b) In the source $\underline{y}$, the entropy is maximal. Use your knowledge of the meaning of entropy to guess the value $P[\underline{y}(n) = 1]$. Explain the reasoning behind your guess. Formally verify that your conjecture is correct.

(c) Given that $P[\underline{x}(n) = x, \underline{y}(n) = y] = 0.25$ for any n and for any possible outcome, $(x, y) = (0,0), (0,1), (1,0), (1,1)$, evaluate the average mutual information, say $\overline{M}(\underline{x}, \underline{y})$, between the *jointly stationary* random sources $\underline{x}$ and $\underline{y}$.

(d) Find the probability distribution $P[\underline{x}(n), \underline{y}(n)]$ such that the two jointly stationary random sources have no average mutual information.

1.21. Verify (1.231)–(1.233).

APPENDICES: Supplemental Bibliography

1.A Example Textbooks on Digital Signal Processing

Cadzow, J. A. *Foundations of Digital Signal Processing and Data Analysis.* New York: Macmillan, 1987.

Jackson, L. B. *Digital Filters and Signal Processing*, 2nd ed. Norwell, Mass.: Kluwer, 1989.

Kuc, R. *Introduction to Digital Signal Processing.* New York: McGraw-Hill, 1988.

Oppenheim, A. V., and R. W. Schafer. *Discrete Time Signal Processing,* Englewood Cliffs, N.J.: Prentice Hall, 1989.

Proakis, J. G., and D. G. Manolakis. *Digital Signal Processing: Principles, Algorithms, and Applications*, 2nd ed. New York: Macmillan, 1992.

1.B Example Textbooks on Stochastic Processes

Davenport, W. B. *Random Processes: An Introduction for Applied Scientists and Engineers.* New York: McGraw-Hill, 1970. [Elementary]

Gardner, W. A. *Introduction to Random Processes with Applications to Signals and Systems*, 2nd ed. New York: McGraw-Hill, 1990.
Gray, R. M., and L. D. Davisson. *Random Processes: A Mathematical Approach for Engineers.* Englewood Cliffs, N.J.: Prentice Hall, 1986.
Grimmett, G. R., and D. R. Stirzaker. *Probability and Random Processes.* Oxford: Clarendon, 1985.
Helstrom, C. W. *Probability and Stochastic Processes for Engineers*, 2nd ed. New York: Macmillan, 1991.
Leon-Garcia, A. *Probability and Random Processes for Electrical Engineering.* Reading, Mass.: Addison-Wesley, 1989. [Elementary]
Papoulis, A. *Probability, Random Variables, and Stochastic Processes*, 2nd ed. New York: McGraw-Hill, 1984.
Peebles, P. Z. *Probability, Random Variables, and Random Signal Principles*, 2nd ed. New York: McGraw-Hill, 1987. [Elementary]
Pfeiffer, P. E. *Concepts of Probability Theory.* New York: Dover, 1965.
Wong, E., and B. Hajek. *Stochastic Processes in Engineering Systems.* New York: Springer-Verlag, 1984. [Advanced]

1.C Example Textbooks on Statistical Pattern Recognition

Devijver, P. A., and J. Kittler. *Pattern Recognition: A Statistical Approach.* London: Prentice Hall International, 1982.
Fukunaga, K. *Introduction to Statistical Pattern Recognition.* New York: Academic Press, 1972.
Jain, A. K., and R. C. Dubes. *Algorithms for Clustering Data.* Englewood Cliffs, N.J.: Prentice Hall, 1988.

1.D Example Textbooks on Information Theory

Blahut, R. E. *Principles and Practice of Information Theory.* Reading, Mass.: Addison-Wesley, 1987.
Csiszár, I., and J. Körner. *Information Theory.* New York: Academic Press, 1981.
Gallagher, R. G. *Information Theory and Reliable Communication.* New York: John Wiley & Sons, 1968.
Guiasu, S. *Information Theory with Applications.* New York: McGraw-Hill, 1976.
Khinchin, A. Y., *Mathematical Foundations of Information Theory.* New York: Dover, 1957.
McEliece, R. J. *The Theory of Information and Coding.* Reading, Mass.: Addison-Wesley, 1977.

1.E Other Resources on Speech Processing

1.E.1 Textbooks

Flanagan, J. L. *Speech Analysis, Synthesis, and Perception*, 2nd ed. New York: Springer-Verlag, 1972.

Furui, S. *Digital Speech Processing*. New York: Marcel Dekker, 1989.

Furui, S., and M. Sondhi. *Recent Progress in Speech Signal Processing*, New York: Marcel Dekker, 1990.

Markel, J. D., and A. H. Gray. *Linear Prediction of Speech*. New York: Springer-Verlag, 1976.

Morgan, D. P., and C. L. Scofield. *Neural Networks and Speech Processing*. Norwell, Mass.: Kluwer, 1991.

O'Shaughnessy, D. *Speech Communication: Human and Machine*. Reading, Mass.: Addison-Wesley, 1987.

Papamichalis, P. E. *Practical Approaches to Speech Coding*. Englewood Cliffs, N.J.: Prentice Hall, 1987.

Parsons, T. W. *Voice and Speech Processing*. New York: McGraw-Hill, 1986.

Rabiner, L. R., and R. W. Schafer. *Digital Processing of Speech Signals*. Englewood Cliffs, N.J.: Prentice Hall, 1978.

1.E.2 Edited Paper Collections

Dixon, N. R., and T. B. Martin, eds., *Automatic Speech and Speaker Recognition*. New York: IEEE Press, 1979.

Fallside, F., and W. A. Woods, eds., *Computer Processing of Speech*. London: Prentice Hall International, 1985.

Lea, W. A., ed., *Trends in Speech Recognition*. Apple Valley, Minn.: Speech Science Publishers, 1980.

Reddy, R., ed., *Speech Recognition*. New York: Academic Press, 1975.

Schafer, R. W., and J. D. Markel, eds., *Speech Analysis*. New York: John Wiley & Sons (for IEEE), 1979.

Waibel, A., and K. F. Lee, eds., *Readings in Speech Recognition*. Palo Alto, Calif.: Morgan-Kauffman, 1990.

1.E.3 Journals

Among the most widely read journals in English covering the field of speech processing are the following:[43]

[43]IEEE is the Institute of Electrical and Electronics Engineers, the world's largest professional organization whose membership is over 300,000. The Signal Processing Society of the IEEE, the society most directly concerned with speech processing, has a membership exceeding 15,000. Other societies of the IEEE also publish transactions which occasionally contain papers on speech processing. Among them are the *Transactions on Information Theory, Computers, Communications, Pattern Analysis and Machine Intelligence, Automatic Control, Systems Man and Cybernetics, Neural Networks*, and *Biomedical Engineering*. IEE is the Institute of Electronics Engineers, the professional electrical engineering society based in the United Kingdom.

AT&T Technical Journal (Prior to 1985, *Bell System Technical Journal*).
Computer Speech and Language.
IEE Proceedings F: Communications, Radar, and Signal Processing.
IEEE Transactions on Signal Processing (Prior to 1991, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, and prior to 1974, *IEEE Transactions on Audio and Electroacoustics*).
IEEE Transactions on Audio and Speech Processing (initiated in 1993).
Journal of the Acoustical Society of America.[44]
Speech Communication: An Interdisciplinary Journal.

In addition, the *Proceedings of the IEEE* and the *IEEE Signal Processing Magazine* occasionally have special issues or individual tutorial papers covering various aspects of speech processing.

1.E.4 Conference Proceedings

The number of engineering conferences and workshops that treat speech processing is vast—we will make no attempt to list them. However, the most widely attended conference in the field, and the forum at which new breakthroughs in speech processing are often reported, is the annual International Conference on Acoustics, Speech, and Signal Processing, sponsored by the Signal Processing Society of the IEEE. The society publishes an annual proceedings of this conference. By scanning the reference lists in these proceedings, as well as those in the journals above, the reader will be led to some of the other important conference proceedings in the area.

Also see Section 1.G.3 of this appendix.

1.F Example Textbooks on Speech and Hearing Sciences

Borden, G., and K. Harris. *Speech Science Primer: Physiology, Acoustics, and Perception.* Baltimore, Md.: Williams & Wilkins, 1980.

Chomsky, N., and M. Halle. *The Sound Pattern of English.* New York: Harper and Row, 1968.

Daniloff, R., G. Shuckers, and L. Feth. *The Physiology of Speech and Hearing.* Englewood Cliffs, N.J.: Prentice Hall, 1980.

Eimas, P., and J. Miller, eds., *Perspectives on the Study of Speech.* Hillsdale, N.J.: Erlbaum, 1981.

Flanagan, J. L. *Speech Analysis, Synthesis, and Perception*, 2nd ed. New York: Springer-Verlag, 1972.

Ladefoged, P. *A Course in Phonetics.* New York: Harcourt Brace Jovanovich, 1975.

[44]Of the journals listed, this one is most oriented toward the presentation of basic science results in speech and hearing.

LeHiste, I., ed., *Readings in Acoustic Phonetics.* Cambridge, Mass.: MIT Press, 1967.
Lieberman, P. *Intonation, Perception, and Language*, Cambridge, Mass.: MIT Press, 1967.
MacNeilage, P. *The Production of Speech.* New York: Springer-Verlag, 1983.
Minifie, F., T. Hixon, and F. Williams, eds., *Normal Aspects of Speech, Hearing, and Language.* Englewood Cliffs, N.J.: Prentice Hall, 1973.
Moore, B. *An Introduction to the Physiology of Hearing.* London: Academic Press, 1982.
O'Shaughnessy, D. *Speech Communication: Human and Machine.* Reading, Mass.: Addison-Wesley, 1987.
Perkell, J., and D. Klatt, eds., *Invariance and Variability in Speech Processes.* Hillside, N.J.: Lawrence Erlbaum Associates, 1986.
Zemlin, W. *Speech and Hearing Science, Anatomy and Physiology.* Englewood Cliffs, N.J.: Prentice Hall, 1968.

1.G Other Resources on Artificial Neural Networks

1.G.1 Textbooks and Monographs

Kohonen, T. *Self-Organization and Associative Memory*, 2nd ed. New York: Springer-Verlag, 1988.
Kosko, B. *Neural Networks and Fuzzy Systems.* Englewood Cliffs, N.J.: Prentice Hall, 1992.
Morgan, D. P., and C. L. Scofield. *Neural Networks and Speech Processing.* Norwell, Mass.: Kluwer, 1991.
Rumelhart, D. E. *Parallel Distributed Processing*, Vol. 1: *Foundations*, Vol 2: *Psychological and Biological Models.* Cambridge, Mass.: MIT Press, 1986.
Simpson, P. K. *Artificial Neural Systems.* Elmsford, N.Y.: Pergamon Press, 1990.
Zurada, J. M. *An Introduction to Artificial Neural Systems.* St. Paul, Minn.: West Publishing, 1992.

1.G.2 Journals

A few of the widely read journals on ANNs in English are the following:

IEEE Transactions on Neural Networks.
International Journal of Neural Systems.
Neural Computation.
Neural Networks Journal.

In addition, many of the journals listed in Section 1.E.3 of this appendix publish articles on neural network applications to speech processing.

1.G.3 Conference Proceedings

The number of conferences devoted to neural network technology is very large. These are two of the important ones:

IEEE International Conference on Neural Networks.
International Joint Conference on Neural Networks.

Many papers on ANNs related to speech processing are also presented at the IEEE International Conference on Acoustics, Speech, and Signal Processing, which is discussed in Section 1.E.4 of this appendix.