1 Introduction

This book has two parts: the first summarizes the facts of coding and information theory which are needed to understand the essence of estimation and statistics, and the second describes a new theory of estimation, which also covers a good part of statistics as well. After all, both estimation and statistics are about extracting information from the often chaotic looking data in order to learn what it is that makes the data behave the way they do. The first part together with an outline of the algorithmic information in Appendix A is meant for the statistician who wants to understand his or her discipline rather than just learn a bag of tricks with programs to apply them to various data, tricks that are not based on any theory and do not stand a critical examination although some of them can be quite useful, providing solutions for important statistical problems.

The word *information* has many meanings, two of which have been formalized by Shannon. The first is fundamental in communication as just the number of messages, strings of symbols, either to be stored or to be sent over some communication channel, the practical question being the size of the storage device needed or the time it takes to send them. The second meaning is the measure of the strength of the statistical property a string has, which is fundamental in statistics, and very different from that in communication. Shannon formalized the measure of information of both kinds in terms of the famous *entropy* of a probability distribution $P(x)$ on the strings:

$$H(P) = \sum_x P(x) \log 1/P(x). \tag{1.1}$$

This is an abstract and intricate measure, for it does not require interpretation of the nature of the strings and symbols x, as done in the thermodynamic

entropy in physics. It plays a central role in coding, for, as we shall discuss in Section 2.2, it amounts to the greatest lower bound of the mean code length of strings emitted by the *source* P, as the jargon goes. It generalizes Hartley's measure as the logarithm of the number of elements in a finite set, which represents the amount of the first kind of information.

Actually both meanings are central to the problem of sending messages through a noisy communication channel, which leads to the related and even more abstract measure of *mutual information* between two or more random variables. Shannon's remarkable theorem states that there is a maximum number of messages that can be sent through the channel in a suitably encoded form such that they can be recovered with arbitrarily small error when the messages are long enough. This is true even though the channel introduces errors with a fixed probability of each symbol sent. For good efficiency the messages sent in communication are often very long. For instance, instead of sending short individual messages they are often bundled into long blocks.

In statistics and estimation there are no messages unless by a message we mean a set of observed data, and we certainly are not interested in their number for storing them nor sending them anywhere. What we are interested in are the statistical properties of data, which is the second meaning of information, and its gathering. This process is a virtual synonym of estimation, and it lies at the heart of the theory of estimation in this book. It is possible that the confusion of the two meanings of information is the reason that Shannon's work on communication has never really influenced statistics and estimation, and in fact few statisticians have been studying information theory, the result of which, I think, is the disarray in the present discipline of statistics.

The notion of limits and the asymptotical behavior are not part of the foundation in the theory of estimation in this book. Rather, they are regarded as approximations of the fundamental notions, which can be useful when the real behavior cannot be easily calculated. For instance the Central Limit Theorem (CLT) is useful because the estimates of the most important maximum likelihood (ML) estimator have a distribution which converges to a normal distribution, frequently rapidly, while the exact distribution can be difficult to calculate.

In the rest of this introduction we outline the theory developed here. Although the work is still introductory, we do derive theorems that state fundamental properties of estimators, without which no theory of estimation can be regarded as satisfactory.

In Chapter 2 we discuss the basics of coding, perhaps in greater detail than necessary. The reason is that a case can be made that coding is equivalent to the very idea of probability, which to be sure is abstract, and coding will provide an understanding which is useful both in the fundamental sense and in direct applications. Moreover, the elementary facts are elegant and can be fun to study for anyone. After all, we all do and apply coding in everyday life. Coding is also the basis for the very abstract notion of entropy, the genesis of Shannon's information theory, of which the basics are discussed in Chapter 3. There the second main notion in information theory is the *mutual information*, of which a different new version, the *maximum estimation information*, is introduced together with the *maximum capacity*. These are fundamental in estimation and provide the basis for defining optimality. They also give a wider perspective to the channel coding problem, in which Shannon's channel coding theorem appears as an asymptotic but important special case.

We begin Part II with a general discussion of the estimation problem together with an introduction to the theory in Chapter 4. Just as other theories are based on axioms or, better, postulates so is this theory. There is just one postulate, which as a background requires a class of distributions called *models* of data. The class replaces the naive assumption that the data are generated by a "true" distribution, and the task is to estimate it. Instead of regarding any model to be "true" or "false" we regard it as good, bad, or something in between, which, importantly, can be assessed. This is done by the postulate as the estimation criterion, which in broad terms can be stated thus: the best estimator as a function taking the observed data to a model in a family is one which is determined by the family, and which maximizes the probability of the observed data. But this looks like an impossible task, because no distribution exists under which all data sets have the maximum probability. In fact, for each data set a distribution exists which assigns the probability unity to it. In the past this simple fact has blocked all similar attempts to found estimation theory on probability maximization.

However, there is a way out of the impasse: first, the requirement that the estimator must be determined by the model class prevents favoring any particular data, and, secondly, the maximization of the probability is interpreted as the necessary condition for the maximum an estimator has to satisfy for optimality. For now, let it be that all this can be made precise in a manner which insures optimality of estimators in virtually every sense we can reasonably ask, and nothing is lost by dropping the assumption of the "truth" but a lot is gained with a richer theory as the result.

After the general discussion and the introduction of the postulate criterion three equivalent characterizations of optimal estimators are described, each illuminating different important properties of them. The first is based on the concept of maximum capacity, achieved by the *maximum capacity* (MC) estimator, of which the ML estimator is a special case, and which permits the estimation of the number of parameters, their structure, and even the intervals. The second redefines the same estimators by the necessary conditions for the maximization of the probability on the data. Finally, the third way to define the optimal estimators is by a new *complete* form of the minimum description length (MDL) principle. It sharpens the older *general* MDL principle for model classes, which are narrower but more accurately specified. We also correct the common misunderstanding that the general MDL principle is a special case of Bayesian methods. This is not true, for no priors are needed in the general MDL principle.

In Chapter 5 we prove that the optimal estimators have further desirable properties, similar to the Cramér–Rao inequality but more comprehensive, because they involve the Kullback–Leibler (KL) distance rather than just the covariance. Moreover, the theorems are non-asymptotic, and they include the estimators for both the real-valued parameters and their number. The rest of the chapter is devoted to consistency either in the KL distance or in the sense of almost surely. The former is appropriate for models of "batch" type, fitted to a single finite string, and the latter for models of random process type, needed for prediction. We show that no estimator defined by any criterion exists which beats the MC estimator defined in this book. The chapter ends with a theorem on consistency of the estimates of the number of parameters, which implies that the optimal estimator has the fastest possible convergence rate. This is actually an old

theorem on universal coding, whose application to estimation has not been realized due to the lack of an estimation theory.

Chapter 6 is devoted to a new theory of interval estimation, based on the general maximum probability criterion postulate. Because the non-interesting interval consisting of the entire parameter space has the maximum probability unity it is a challenge to construct an estimator that could be called optimal for intervals. However, there is a way, and the result is an estimator of which the usual point-wise ML estimator is the special zero-length interval case. By the CLT an optimally estimated one-dimensional interval has the probability about 0.68, while the optimally estimated k-dimensional interval has the probability 0.68^k. These intervals represent optimal precision on the parameters, which varies from point to point in the parameter space.

Building on these ideas we describe in Chapter 7 the basics of a theory of hypothesis testing, which differs substantially from the existing methods. The hypotheses are either models or sets of models, none of them "true." We consider data to consist of the meaningful information bearing part and two types of "noise," fast fluctuations and slow drift type of variations, which are harder to detect. We model all three of them differently. Hypothesis testing is regarded as the problem of finding out if the observed data are typical for a proposed hypothesis, meaning "acceptance" of the hypothesis, or atypical, "random," relative to the hypothesis, meaning its rejection. The data are represented by test statistics, functions of data, selected so that their distributions are "peaked" in order to have a sharp separation of the typical and the atypical data. In fact, for a number of different types of tests there are test statistics whose distributions are peaked enough to allow acceptance of the hypothesis as sharply as its rejection, which is the only test that can be made in traditional hypothesis testing.

There are two important test statistics and their distributions: the ML estimates with their induced density functions, and the KL distance with a distribution induced by the multinomial. The latter replaces the customary asymptotic χ^2 approximation, which is shown to be grossly inadequate. The same applies to another frequently applied test statistic, the logarithm of the ratio of maximized likelihoods, which asymptotically also admits the χ^2 distribution of

appropriate degrees of freedom, but which also is inadequate. It appears that selecting first the χ^2 distribution as done traditionally and then looking for a fitting test statistic is like putting the cart before the horse.

In Chapter 8 we discuss the linear quadratic denoising problem in the MDL framework. We describe both hard and soft thresholding, the latter of which poses a problem to the MDL theory. The novelty over the past publications on linear quadratic denoising is that the MDL criterion has no arbitrary hyperparameters, for they are optimized.

Finally, in Chapter 9 we discuss a different sequential way to estimate models which is appropriate for time series type of random processes. In them the ML estimators are updated not only from the past data, which predictive "plug-in" estimators do, but the updating also includes the most important latest data point. As a result, the so-maximized likelihood is bigger than the batch-wise maximized likelihood. We discuss such models for the class of discrete Markov chains, where the states have variable length as a function of the past data. Their main advantage is that the problem of an exponentially growing number of parameters in the usual m-grams is avoided. We also study linear least squares models, both of the type where the regressor matrix is fixed and of the autoregressive (AR) and autoregressive moving average (ARMA) type, where the matrix is determined by the observations. We derive the recursive predictors in Kalman's theory in a simpler non-redundant manner without the cumbersome Riccati equations. As the main result we give a short proof of the lower bound for the prediction error when the parameters have been estimated.

Although Appendix A includes just the very basic notions of the algorithmic theory of information or complexity, we introduce a new version of complexity in terms of the shortest program, which need not satisfy Chaitin's and Kolmogorov's prefix condition. It was inspired by the ideas in Chapter 4 together with Solomonoff's original idea. We define a non-asymptotic notion of *relative randomness*, which in turn inspired the new theory of hypothesis testing in Chapter 7.

The background knowledge required from the reader is a solid understanding of basic probability theory, however without the need to know the intricate measure theory since the only measurable sets considered are finite and

countable sets, as well as open sets and their closures. It is not necessary to know much ordinary statistics, because most of its notions are obsolete and replaced by fewer, different, and more fundamental ones. The basic ideas of information and coding theory which are relevant in estimation are explained in the first part of the book.

Part I

Coding and information

The word "information" has several meanings, the simplest of which has been formalized for communication by Hartley [17] as the logarithm of the number of elements in a finite set. Hence the information in the set $A = \{a, b, c, d\}$ is $\log 4 = 2$ (bits) in the base 2 logarithm. Only two types of logarithm are used in this book: the base 2 logarithm which we write simply as log, and the natural logarithm, written as ln. The amount of information in the set $A = \{a, b, c, d, e, f, g, h\}$ is three bits, and so on. Hence such a formalization has nothing to do with any other meaning of the information communicated, such as the utility or quality. If we were asked to describe one element, say c in the second set, we could do it by saying that it is the third element, which could be done with the binary number 11 in both sets, but the element f in the second set would require three bits, namely 011. So we see that if the number of elements in a set is not a power of 2, we need either the maximum $\lceil \log |A| \rceil$ number of bits or one less, as will be explained in the next section. Hence, we start seeing that "information," relative to a set, could be formalized and measured by the shortest code length with which any element in a set could be described. Again, the word "coding" has nothing to do with description for the purpose of hiding or keeping information secret. It simply means a way to specify the elements in a somehow defined set, which to be sure can be done in a number of ways.

Another interpretation has been suggested for this formalized meaning of information, namely "complexity," for clearly the bigger the set the more complex the description of its elements is in that their coding requires more bits. Big numbers not only require more digits to describe them but also whatever they represent their meaning is hard to comprehend. The classical example is "light year" that physicists use for measuring huge distances that would be hard to comprehend otherwise. By contrast, we understand the size of sets up to five or so without

having to count, but the complexity increases surprisingly rapidly with the size of the set. The meaning of the word complexity is not only restricted to the description length of the elements but also to the difficulty of the task of finding an element in a set, which grows with the size of the set.

As we shall see the complexity of the task of estimation can be measured in the same manner, and, in fact, the very idea of probability amounts to the same thing! Whether to use the word "information" or "complexity" depends on which aspect we want to emphasize. There is also a budding theory of complexity of computations, but the difficulties of getting a general theory appear to be orders of magnitude greater, and we regard the word "complexity" as synonymous with the three notions of information, namely, Shannon information, combinatorial information, and algorithmic information, as described in a brief paper by Kolmogorov [21], which are all essentially a code length.

We can rarely formalize intuitive ideas entirely, because the complexity of the formal description grows rapidly beyond our means. We only need to bear in mind the language used in law books, which tries to cover all the contingencies of the law and which is nearly indecipherable. The formal definition of information covers only the aspect of the word that relates to the code length. Hence it says nothing about whether the information conveyed is useful for anything or is just "noise." However, the code length turns out to be equivalent to probability and hence adds to our understanding of this fundamental but elusive idea. Moreover, the objective in estimation and statistics is to extract "information" from data, and rather than settle for the intuitive ideas it seems to be of interest to see if the code length interpretation can after all suggest a sense of the quality of the "information" by separating the useless noise from the useful part.

2 Basics of coding

We begin with the description of the most basic and primitive codes. Let $A = a_1, \ldots, a_k$ be a finite set: the set is called an *alphabet*, and its elements are called *symbols*. The main interest is in the set of finite sequences $s^n = a_i a_j \ldots$ of the symbols of some length n, called *messages* or for us just *data*. The problem is to send or store them in a manner that costs the sending device little time and storage space. Again for practical reasons these devices use binary symbols 0 and 1, while the original symbols are represented as a sequence of binary symbols, such as the eight bits long "bytes." Let for each symbol a in A, $C : a \mapsto C(a)$ be a one-to-one map, called a *code*, from the alphabet into the set of binary strings. It is extended to the messages by *concatenation* $C : a_i a_j \ldots a_n \mapsto C(a_i)C(a_j)\ldots C(a_n)$. Both binary strings $C(a)$ and $C(a_i)C(a_j)\ldots C(a_n)$ are called *codewords*.

This will give us an upside down binary tree, with the root on top, whose nodes are the codewords, first of the symbols for $n = 1$ and then of the length 2 messages, and so on. The left hand tree in Figure 2.1 illustrates the code for the symbols of the alphabet $A = \{a, b, c\}$. The extension of the codes from symbols to sequences by concatenation creates a problem: We would like to decode the message string symbol for symbol from the binary string representing the codeword of the message. This can be done if we put the restriction on the code trees that only the leaves can be codewords. This means that no prefix of a codeword can be a codeword of another symbol. The right hand tree in Figure 2.1 illustrates such a *prefix* code. This permits decoding of the strings without commas to separate the codewords for the symbols by just climbing down the code tree starting at the root until a leaf is met. As an example, unlike with the left hand tree, the codeword $C(0110001)$ defined by the right hand tree decodes as the string *bcab*. A code is called *complete* if all the leaves are codewords.