Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control. James C. Spall

ISBN: 0-471-33052-3

CHAPTER 17

OPTIMAL DESIGN FOR EXPERIMENTAL INPUTS

This chapter wades into the vast subject of experimental design for establishing user-determined input values. We are concerned here with using experimental resources in some intelligent way to obtain the most information possible about a process. While the general subject of experimental design is large, encompassing both qualitative and quantitative aspects, our focus is much narrower (although still quite broad). We discuss *optimal design*, an area within experimental design dealing with the formulation and solution to the problem of picking input values according to a formal criterion.

Section 17.1 provides some general background about optimal experimental design, including a description of the main (D-optimal) criterion of interest. Section 17.2 considers optimal design in the context of linear models, while Section 17.3 discusses some specialized methods for response surface methods. Section 17.4 considers the more difficult nonlinear problem. After some concluding remarks in Section 17.5, an appendix (Section 17.6) summarizes some results for optimal inputs in dynamic systems. Stochastic methods of search and optimization play a role in recursively determining the optimal design.

17.1 INTRODUCTION

17.1.1 Motivation

Experimental design is an important subject in areas such as statistics, engineering design and quality control, social science, agriculture, process control, and medical clinical trials and pharmaceuticals. The subject involves both qualitative and quantitative aspects. In particular, experimental design deals with issues such as:

- **Comparison of treatments.** A key aspect of experimental design is to determine the best of several candidate inputs. For example, in the operations of a bank, one might be interested in comparing the profit for various combinations of administrative structure, number of tellers, interest rates

paid on bank deposits, and so on. The testing might involve computer simulation and/or tests in the actual bank.

- **Variable screening.** When there are a large number of possible variables, screening experiments can be used for a sensitivity analysis that helps identify the input and output variables most relevant to the application.
- **Response surface exploration.** After the key inputs have been identified, it is often of interest to get detailed insight into the effect of these inputs on the output. This is usually carried out via a low-order (first- or second-order) polynomial in a local region of the input space.
- **Enhanced estimation.** Building mathematical models usually requires the collection of data from the real system. Among other applications, these data are used to estimate unknown parameters for the model. It is desirable to pick the experimental inputs so that the resulting output data can be used to estimate model parameters as accurately as possible.
- **Model validation.** Suppose that an analyst wishes to evaluate whether an existing model (simulation or otherwise) is an accurate representation of the process under study. If the response surface analysis above (or the closely related metamodel discussed later in this section) indicates that one or more factors have an effect very different from that based on predictions from an existing model, the model may be flawed.[1]
- **Choice of inputs in dynamic systems.** An important issue in building control systems and related models of dynamic systems is the choice of the inputs while the dynamic system is operating. The interest here is to excite the physical system in a way that is useful for generating information that will subsequently be used in building a model and controlling the system (for linear time series models, the term *persistency of excitation* is used to refer to a choice of inputs guaranteeing that all model parameters can be estimated).

In the confines of one chapter, we are not able to treat all of the above subjects in any detail. Textbooks such as Box and Draper (1987), Khuri and Cornell (1987), Atkinson and Donev (1992), Wu and Hamada (2000), and Montgomery (2001), and review papers such as Nair (1992), Rosenberger (1996), Miller and Wu (1996), and Cox and Reid (2000), provide a much fuller discussion of some or all of the aspects above. Reviews from the control systems perspective are given in Mehra (1974), Goodwin and Payne (1977), Walter and Pronzato (1990), and Ljung (1999, Chap. 13).

Consider the familiar input–output process relating a scalar output z to an input vector $\boldsymbol{x}$. Suppose, as in previous chapters, that we model this input–output process with a regression function $h(\boldsymbol{\theta}, \boldsymbol{x})$, where $\boldsymbol{\theta}$ is the usual p-dimensional parameter vector that needs to be determined. In particular, the model for the output z is

[1] Kleijnen (1998, p. 210) mentions a validation study where the design of experiments together with a regression metamodel exposed a serious flaw in a simulation model of the environment related to the greenhouse gas problem.

$$z = h(\boldsymbol{\theta}, \boldsymbol{x}) + v, \tag{17.1}$$

where v is a noise term possibly dependent on $\boldsymbol{x}$ or $\boldsymbol{\theta}$. Suppose that the dimension of $\boldsymbol{x}$ is r.

In the language of experimental design, $\boldsymbol{x}$ is composed of factors, each factor representing one of the quantities that is going to be studied in the experiment. Each component of $\boldsymbol{x}$ represents one factor. The range of possible physical meanings for $\boldsymbol{x}$ is endless, but might include terms related to factory or machine settings, drug dosages, the amount of studying needed for an exam, and so on. It is assumed that the analyst has full control over the levels of the factors in producing a set of data. The fundamental problem here is to choose n (say) input vectors, $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$, such that when experiments are run at these n values, the corresponding output values, $z_1, z_2, \ldots, z_n$, are as informative as possible with respect to building the model in (17.1). Let

$$\boldsymbol{X}_n \equiv [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n]^T$$

be the $n \times r$ matrix of n input vectors that will be used to estimate $\boldsymbol{\theta}$. Of course, finding inputs $\boldsymbol{X}_n$ that provide data that are "informative as possible" is a nebulous and perhaps hopeless goal. The optimal experimental design framework provides enough structure to make this goal attainable in a restricted sense.

The focus here will be on choosing the input levels in some optimal way to enhance the process of estimating $\boldsymbol{\theta}$. This represents only one aspect of the more general goals of experimental design that were listed above. Nevertheless, it is an important aspect. It is assumed that the input and output variables have been defined and that there exists a known structure for the mathematical (regression) model $h(\cdot)$ describing the process. The interest is in determining the best values of the various factors in the vector $\boldsymbol{x}$ to use when running experiments and collecting data. In most input–output processes, the choice of input levels will have a significant effect on the observed outputs.

Optimal design is the rubric used for the aspect of experimental design dealing with optimization of the n design vectors contained in $\boldsymbol{X}_n$ based on a formal optimization criterion. Unlike the prior discussion in this book, the search and optimization focus here is not on $\boldsymbol{\theta}$. Rather, $\boldsymbol{\theta}$ will *ultimately* be estimated based on some optimization criterion after the design is implemented or will sequentially be estimated while the design is being established. Because of the inherent uncertainty in the information collected via experimentation and the possibly difficult form for the optimization criterion for $\boldsymbol{X}_n$, both of the defining characteristics of stochastic search—noisy loss measurements and injected algorithmic randomness, as discussed in Subsection 1.1.3—may be relevant in finding the optimal design factors.

The majority of traditional experimental design methods—including most of the methods treated in the textbooks and review articles mentioned above—

are motivated in a relatively informal way and not derived as formal optimal designs. These include popular approaches such as factorial and fractional factorial designs, composite designs, block designs (e.g., Latin squares or randomized blocks), and so on. Such "classical" methods have a long history of successful application in problems related to estimating $\boldsymbol{\theta}$ in linear models (linear in $\boldsymbol{\theta}$) as well as to other problems such as those listed at the beginning of this section.

In light of the long history of success in classical experimental design, one might wonder about the need for optimal design. Two important contributions of optimal design stand out. One is to add rigor and clarify the properties for some of the classical designs. As we will see, some classical designs are also optimal designs under certain conditions. The second important contribution is to provide an approach for constructing designs in nonstandard situations, including the very important cases where the model is nonlinear in $\boldsymbol{\theta}$ and/or the domain for the input factors is not a "nice" region such as a hypercube or hypersphere (i.e., r-dimensional cube or sphere). Atkinson (1996) discusses some other applications for optimal design. The following summary captures some of the benefits of optimal design.

Example 17.1—Contrast of classical and optimal designs. As an illustration of the benefits of optimal design, Montgomery (2001, p. 469) discusses a problem where an investigator is studying the properties of an adhesive material based on two input factors in $\boldsymbol{x}$ (i.e., $r = 2$). Because of constraints in the experiment, this problem yields a domain for $\boldsymbol{x}$ that is not of the square or circular form typically assumed in classical design when $r = 2$ (although the underlying regression model is linear). The domain is square with two opposite corners diagonally sheared off. On the other hand, the optimal design *is* readily able to cope with the irregular domain. Montgomery compares the accuracy of the estimate for $\boldsymbol{\theta}$ resulting from a classical design (the central composite design, which is a combination of the cube and star designs discussed in Subsection 17.2.3) and from an optimal design (based on D-optimality, as introduced in Subsection 17.1.3). Based on a sum of the variances of the estimates for the elements of $\boldsymbol{\theta}$ as the measure of accuracy, it is found that approximately twice as many experiments are needed with the classical design as with the optimal design to achieve the same accuracy in estimating $\boldsymbol{\theta}$. ❑

While the benefits from optimal design may be significant, one limitation of the approach is its close connection to assumptions in the model.[2] If the underlying model structure is imperfectly known, then the resulting "optimal"

[2]Note that the strong connection to the model here differs from some of the other approaches that we have seen in this text in other (nonexperimental design) contexts. In particular, methods such as random search, FDSA, SPSA, simulated annealing, and evolutionary computation in Chapters 2 and 6–10 require minimal information about underlying models.

design may, in fact, be suboptimal. For example, if the assumed model is linear in x and the true system is nonlinear in x, the choice of inputs under an optimal design may be suboptimal. Further, optimal designs are inherently tied to the choice of an optimality criterion. In practice, the choice of criterion may not be obvious, although an equivalence theorem introduced in Section 17.2 provides a powerful rationale for one popular criterion (the D-optimal criterion). The use of optimal design is sometimes advocated for follow-up studies. Here, information from initial standard designs can be used to produce preliminary information about the model, which then forms the basis for the subsequent optimal design. Among the texts and monographs that emphasize the optimal design aspect of experimental design are Fedorov (1972), Silvey (1980), Atkinson and Donev (1992), and Pukelsheim (1993).

To close this subsection, let us illustrate the different characteristics of optimal design for linear or nonlinear models in the context of Monte Carlo simulation. Recall from Chapters 14 and 15 that the aim is to use a simulation to optimize some aspect of a real system. It is assumed that the simulation itself is fully trained as an accurate representation of the actual system within the domain of interest. Now suppose that we step back and concern ourselves with building either an efficient approximation to an existing simulation or with building the original simulation itself.

When building an approximation of a large-scale simulation, one is concerned with building a *metamodel* of the simulation. Metamodels are typically curve fits of the actual simulation that can be used in cases where the simulation would otherwise have to be run a prohibitive number of times. Simulation runs provide the data for building the metamodel. The advantage of the metamodel is that, once constructed, it can be executed in a small fraction of the time required for the full simulation. Among other uses, metamodels can be applied for sensitivity analysis and model validation. Typical metamodels are low-order polynomials of the input variables x weighted by elements of a parameter vector θ. This corresponds to a curvilinear model (as discussed in Section 3.1) because the model is linear in θ. Hence, design methods for linear models can be used to determine the inputs for the simulation runs required to build the metamodel. Optimal design provides the means for squeezing the most information out of a set of n simulation runs. Response surface methods (Section 17.3) are often used to build metamodels.

The second, more challenging, case for optimal design pertains to the building of the simulation itself. Here, there are parameters θ inside the simulation that need to be estimated to produce reliable simulation output. This is quite different from the metamodel approximation for an *existing* simulation. For example, in a simulation of traffic flow in a network, θ might include terms related to the mean arrival rates into the network, the capacity of various streets inside the network, the mean times for daily traffic surges into various parts of the network, and so on. Hence, the design involves choosing aspects of the real traffic system, corresponding to the inputs x, that enhance the ability to estimate θ. This might include the choice of the timing strategies for the traffic signals in the

network, the placement of variable message signs notifying drivers of congestion, and so on. For the very reason that a simulation is being used to represent the system (versus a simple analytical model), it is expected that $\boldsymbol{\theta}$ enters in a complex nonlinear way. Hence, nonlinear design methods (Section 17.4) are appropriate. In contrast to the metamodel problem above, optimal design provides guidance on how best to conduct the n experiments on the *physical system*, as needed to build the simulation.

In summary, for the simulation context, *linear* design is appropriate in the building of simulation metamodel if the metamodel is a polynomial form in the inputs (a nonlinear metamodel such as a neural network requires nonlinear design). On the other hand, *nonlinear* design is usually required for parameter estimation inside the simulation.

17.1.2 Finite-Sample and Asymptotic (Continuous) Designs

The above discussion motivates some of the interest in experimental design and the need to pick input values intelligently. Let us now introduce more formally the concept of an optimal design for finding the "best" n input vectors $\boldsymbol{X}_n$. In particular, as a step toward developing practical methods for determining an optimal design, this subsection discusses the distinction between finite-sample and limiting (asymptotic) designs.

In finite-sample designs, the n inputs are allocated to achieve an optimal value of some criterion, such as the D-optimal criterion discussed in the next subsection. In an asymptotic (continuous) design, inputs are allocated according to proportions that would apply *as if* there were an infinite amount of data available. The proportions in an optimal asymptotic continuous design may be any real number in the interval [0, 1]. Hence, in finite-sample practice, it may not be possible to find an exact design that corresponds to the optimal continuous design. If n is sufficiently large, however, one can find an approximation to the optimum that is likely to be satisfactory. The example below illustrates the contrast between finite-sample and asymptotic designs.

Example 17.2—Finite-sample and asymptotic designs. Suppose that $n = 10$, and the input x is a scalar such that $0 \le x \le 100$. Based on optimizing an appropriate criterion for the given sample size n, it might be found that four measurements should be taken at $x = 0$, two at $x = 50$, and four at $x = 100$ (i.e., the optimal $\boldsymbol{X}_n = [0, 0, 0, 0, 50, 50, 100, 100, 100, 100]^T$). Suppose, however, that the best value of the criterion is possible if we place 42 percent of the measurements at the endpoints ($x = 0$ or 100) and the remaining 16 percent at the center point ($x = 50$). This exact proportion is only possible if we have n as some multiple of 50 (e.g., at $n = 50$, 21 measurements are placed at the endpoints and 8 measurements are placed at the center point). The allocation chosen above for $n = 10$ is only an approximation of the asymptotic—or continuous—design. This example is one where it is possible to achieve an exact asymptotic design for at least some values of n (i.e., multiples of 50). In other problems, at least some

allocations may be according to irrational numbers, in which case no finite-sample design can ever achieve an exact asymptotic design. ❑

Mathematical tractability provides the prime rationale for seeking optimal asymptotic designs. In a manner analogous to the use of asymptotic theory in many other aspects of stochastic algorithms, the mathematical problem of finding an optimal design is often simplified by considering asymptotic designs. Hence, in finding an asymptotic design, we ignore the constraint that the number of measurements at any design value (x) must be an integer. In *implementing* the asymptotic solution, we approximate the design by allocating the measurements in proportions as close as possible to the asymptotic design.

The experimental designs of interest here are represented by a set of distinct input support points ($x = \chi_i$) and a corresponding set of weights (w_i) representing the allocations to the support points:

$$\xi \equiv \begin{Bmatrix} \chi_1 & \chi_2 & \cdots & \chi_N \\ w_1 & w_2 & \cdots & w_N \end{Bmatrix}. \tag{17.2}$$

The first row gives the values of the input factors and the second row gives the associated weights. Note that $N \le n$ and $\sum_{i=1}^{N} w_i = 1$ with $0 \le w_i \le 1$. Because the total number of inputs is n, the number of inputs taking value χ_i is exactly or approximately nw_i (*exactly* if the weights are associated with a finite-sample design; *approximately* if the weights are for an asymptotic design that does not allow a strictly proportional allocation over the n measurements).

So, determining a design involves choosing the number of support points N and associated values χ_i, together with the proportions of measurements assigned to those support points (the w_i). Let

$$\begin{aligned} \xi_n^* &\equiv \text{optimal finite-sample } (n) \text{ design,} \\ \xi_\infty^* &\equiv \text{optimal asymptotic (continuous) design.} \end{aligned} \tag{17.3}$$

An optimal design ξ^*, where ξ^* represents either ξ_n^* or ξ_∞^* as appropriate, is the combination of inputs and weights that optimizes the chosen criterion. Hence, for Example 17.2 discussed above, (17.3) implies that the optimal finite-sample ($n = 10$) and asymptotic designs would be written as

$$\xi_{10}^* = \begin{Bmatrix} 0 & 50 & 100 \\ 0.40 & 0.20 & 0.40 \end{Bmatrix} \text{ and } \xi_\infty^* = \begin{Bmatrix} 0 & 50 & 100 \\ 0.42 & 0.16 & 0.42 \end{Bmatrix}.$$

Often, for simple models with θ, as usual, having p elements, the optimal asymptotic design assigns a weight $w_i = 1/p$ to p different input points. So, if $n = p$ (or n is some integer multiple of p), it is possible to implement the optimal

asymptotic design exactly. In more complex models, the allocations are usually less tidy, including cases with weights that are not rational numbers (e.g., $w_i = 1/\sqrt{5}$ for some i). As n gets larger, the allocation of measurements can be made to approach any optimal asymptotic design arbitrarily closely.

The values for x_k are drawn from the $N \leq n$ inputs χ_i in the design ξ from (17.2). The allocation of the N support points to the n inputs applied in estimating θ is done according to the proportions reflected in the weights w_i, as illustrated in the example below.

Example 17.3—Mapping from design inputs (support points) to applied inputs. Suppose that $n = 20$ measurements will be used to estimate θ. Based on an analysis for optimal designs, suppose that the optimal finite-sample design is

$$\xi_{20}^* = \begin{Bmatrix} \chi_1 & \chi_2 & \chi_3 & \chi_4 & \chi_5 & \chi_6 \\ 0.05 & 0.15 & 0.05 & 0.30 & 0.25 & 0.20 \end{Bmatrix},$$

where the χ_i are some specified inputs. Based on allocating the x_k sequentially according to the order of the χ_i, we have the mapping of χ_i to x_k shown in Table 17.1. For example, from the table, $\chi_2 = x_2 = x_3 = x_4$. ❑

17.1.3 Precision Matrix and *D*-Optimality

Recall that the fundamental goal is to pick a design ξ such that θ is estimated with as much precision as possible. An important quantity in optimal design is the $p \times p$ precision matrix, say $M(\theta, \xi)$, which is an expression of the accuracy of

Table 17.1. Design inputs χ_i allocated to the input vectors x_k applied in estimating θ in Example 17.3. For each row, the value in the left column is assigned to the inputs in the right column.

Inputs from ξ_{20}^*	Input vector applied in model (17.1)
χ_1	x_1
χ_2	x_2, x_3, x_4
χ_3	x_5
χ_4	$x_6, x_7, x_8, x_9, x_{10}, x_{11}$
χ_5	$x_{12}, x_{13}, x_{14}, x_{15}, x_{16}$
χ_6	$x_{17}, x_{18}, x_{19}, x_{20}$

the θ estimate based on the n inputs $\boldsymbol{X}_n$. In general, this matrix depends on θ, as indicated. This matrix reflects the variability in the estimate that is induced from the stochastic variability of the outputs $z_1, z_2, \ldots, z_n$. A "larger" value of $\boldsymbol{M}$ reflects more precision—lower variability—in the estimate. We are most interested in two specific representations of $\boldsymbol{M}$: (i) the case where $\boldsymbol{M}$ is proportional to the inverse of the covariance matrix of the θ estimate, and (ii) the case where $\boldsymbol{M}$ is equal to the (Fisher) information matrix. The quantity $\boldsymbol{M}$ is frequently called the information matrix in the optimal design literature, but we avoid that terminology because in case (i) above, $\boldsymbol{M}$ does not strictly equal the information matrix (see the definition in Section 13.3). In fact, the covariance matrix used in case (i) can be defined without the distributional assumptions needed to define the information matrix. Case (i) is appropriate for linear models while case (ii) applies to more general nonlinear models. Case (ii) relies on the close connection between the inverse information matrix and the covariance matrix of the θ estimate (discussed in detail in Section 13.3).

Given the close connection of the inverse of the covariance matrix to the precision matrix $\boldsymbol{M}$, a natural goal in picking the design ξ is to find the design that "maximizes" the matrix $\boldsymbol{M}$. We might first ask if it is possible, in general, to find such a design. That is, can we find a design ξ^* such that $\boldsymbol{M}(\theta, \xi^*) > \boldsymbol{M}(\theta, \xi)$ in the matrix sense[3] for all $\xi \neq \xi^*$? The answer is no.

Because no design exists with a uniformly larger precision matrix, we can work with a weaker metric that is similar in spirit, the *D-optimal criterion*. The *D*-optimal criterion is the determinant of the precision matrix (*D* for determinant). In particular, the optimal design is found according to

$$\xi^* = \arg\max_{\xi} \left\{ \det\left[\boldsymbol{M}(\theta, \xi) \right] \right\}. \tag{17.4}$$

The *D*-optimal criterion $\det\left[\boldsymbol{M}(\theta, \xi)\right]$ is the most popular formal means of determining an optimal design ξ^*. This criterion expresses a goal *similar* to the goal of a larger precision matrix given the interpretation of the determinant as some measure of the size of the matrix. This criterion has several desirable properties, as we discuss below. It is, however, easy to illustrate that a precision matrix based on ξ^* as defined in (17.4) is not uniformly larger than all other precision matrices, as we now show.

Example 17.4—Cautionary illustration related to the *D*-optimal criterion. Let us demonstrate that while the *D*-optimal measure provides a useful basis for comparing the size of matrices, caution is necessary regarding the strength of the conclusion. Suppose that $\boldsymbol{M}^*$ represents the optimal precision matrix based on

[3]The inequality here means that $\boldsymbol{M}(\theta, \xi^*) - \boldsymbol{M}(\theta, \xi)$ is a positive definite matrix—see Appendix A (Section A.2).

ξ^* and that $\boldsymbol{M}$ represents the precision matrix based on some $\xi \neq \xi^*$. Suppose that

$$\boldsymbol{M}^* = \begin{bmatrix} 3 & 2 \\ 2 & 2 \end{bmatrix} \text{ and } \boldsymbol{M} = \begin{bmatrix} 2 & 0.5 \\ 0.5 & 1 \end{bmatrix}.$$

Both of $\boldsymbol{M}^*$ and $\boldsymbol{M}$ are positive definite and $\det(\boldsymbol{M}^*) = 2 \geq \det(\boldsymbol{M}) = 1.75$. However, $\boldsymbol{M}^* - \boldsymbol{M} = \begin{bmatrix} 1 & 1.5 \\ 1.5 & 1 \end{bmatrix}$ is an indefinite (nonpositive definite) matrix, having one positive and one negative eigenvalue. So, $\boldsymbol{M}^* \not> \boldsymbol{M}$. (See Exercise 17.1.) ❑

There has been much further discussion of the interpretation of the *D*-optimal criterion in the experimental design literature together with discussion of alternative criteria (one of which turns out to be equivalent to (17.4) under certain circumstances, as shown in Theorem 17.1). We will not reiterate the debates and discussion here (see, e.g., Silvey, 1980, Chaps. 1 and 2; Atkinson and Donev, 1992, Chap. 10; and Montgomery, 2001, pp. 466–472, for statistical perspectives on these issues; Ljung, 1999, Chap. 13, discusses some criteria popular in control systems engineering, both in open- and closed-loop problems). Rather, we accept this widely used criterion and proceed to make connections to the fundamental themes of this text.

One desirable property of *D*-optimality is *transform invariance*, such as what we saw with the Newton–Raphson and second-order stochastic approximation algorithms (Sections 1.4, 4.5, and 7.8). It is obviously desirable that an optimal design be independent of whether (say) we choose to measure elements of θ in meters or centimeters. In particular, invertible linear transformations of θ do not affect ξ^* (Exercise 17.2). Linear transformations *do* affect the solution in nondeterminant-based criteria (Pukelsheim, 1993, pp. 137, 344–345).

Directly solving for ξ^* in most practical problems is difficult. Rather, for linear models, one often "guesses" at a design based on intuitive considerations and then verifies it by an important theorem on equivalence of certain solutions. This theorem is presented in Section 17.2. In nonlinear models, alternative methods are needed.

17.2 LINEAR MODELS

17.2.1 Background and Connections to *D*-Optimality

The subject of experimental design for linear models is a large subject unto itself. For example, almost all of the well-known text by Montgomery (2001–5th ed.) is devoted to classical experimental design for linear models. Nevertheless, some of the issues for linear models are generic and apply in nonlinear problems

as well. Recall that a model is linear if it is linear in θ; nonlinear functions of $x = x_k$ are allowed within the linear framework.

Suppose that the classical linear regression model (eqn. (3.1)) is used to describe the process,

$$z_k = h_k^T \theta + v_k, \quad k = 1, 2, \ldots, n, \tag{17.5}$$

where $h_k = \mathcal{H}(x_k)$ is the design vector of dimension p dependent on input $x = x_k$, $\mathcal{H}(x)$ is a function mapping the inputs to the design vector (as in Chapter 13), and v_k is a mean-zero noise term uncorrelated with v_j for all $j \neq k$ and having common variance σ^2 across k. The values for x_k are drawn from the $N \leq n$ inputs χ_i in the design ξ from (17.2). The allocation of the n inputs is done according to the proportions reflected in the weights w_i, as illustrated in Example 17.3.

As in Subsections 3.1.2 and 13.1.3, let $Z_n = [z_1, z_2, \ldots, z_n]^T$ and H_n be the $n \times p$ stacked matrix of h_k^T row vectors. The leads to the (batch) ordinary least-squares estimate of θ based on the n input–output pairs

$$\hat{\theta}^{(n)} = (H_n^T H_n)^{-1} H_n^T Z_n \tag{17.6}$$

(assuming that the indicated inverse exists, of course). In the linear formulation of (17.6), H_n is a general (possibly nonlinear) function of the collection of inputs X_n. In the common case where the regression model includes an additive constant and where the elements of h_k correspond directly to the elements of x_k, then

$$\begin{aligned} z_k &= h_k^T \theta + v_k \\ &= [1, x_{k1}, x_{k2}, \ldots, x_{k,p-1}]\theta + v_k \\ &= [1, x_k^T]\theta + v_k, \end{aligned} \tag{17.7}$$

where x_{ki} is the ith element of x_k. Here, the first element of θ is the intercept term (the above-mentioned additive constant), and the remaining $p - 1$ elements are the slope terms. The p elements of θ jointly represent the linear regression parameters to be estimated and $x_k = [x_{k1}, x_{k2}, \ldots, x_{k,p-1}]^T$ represents the $p - 1$ input factors that are to be determined in the experimental design at each k. (In the nonlinear problems of Section 17.4, there is generally no such "easy" relationship between the dimension p and the number of terms to be determined in the design.) Note that $n \geq p$ to ensure that the p elements of θ can be uniquely estimated. That is, we need at least as many data points as there are parameters to be estimated.

In the general linear case of (17.5) and (17.6), there is a ready expression for the dispersion of $\hat{\boldsymbol{\theta}}^{(n)}$ via the covariance matrix for the estimate. This provides an expression directly usable in the *D*-optimal criterion. Specifically, the covariance matrix for the standard least-squares regression estimate $\hat{\boldsymbol{\theta}}^{(n)}$ appearing in (17.6) has a particularly simple form:

$$\operatorname{cov}\left(\hat{\boldsymbol{\theta}}^{(n)}\right) = (\boldsymbol{H}_n^T \boldsymbol{H}_n)^{-1}\sigma^2 \tag{17.8}$$

(Subsection 3.1.2; Box and Draper, 1987, p. 74). Note that this covariance matrix is not dependent on $\boldsymbol{\theta}$ or on the probability distribution for the noise (beyond knowledge of the variance σ^2), a property rarely true of the covariance matrix for the estimate of $\boldsymbol{\theta}$ in nonlinear models.

Because the inverse of the covariance matrix is a representation of the precision of the estimate, (17.8) provides the basis for the precision matrix used in the *D*-optimal criterion. In particular, the precision matrix is proportional to the inverse of the covariance matrix. So, the *D*-optimal solution of maximizing the determinant of the precision matrix is equivalent to minimizing the determinant of the covariance matrix. Because σ^2 does not affect the minimization of $\det[\operatorname{cov}(\hat{\boldsymbol{\theta}}_n)]$, the *D*-optimal solution is found according to

$$\begin{aligned}\xi^* &= \arg\min_{\xi}\left\{\det\left[(\boldsymbol{H}_n^T \boldsymbol{H}_n)^{-1}\right]\right\} = \arg\min_{\xi}\left\{\left[\det(\boldsymbol{H}_n^T \boldsymbol{H}_n)\right]^{-1}\right\} \\ &= \arg\max_{\xi}\left\{\det(\boldsymbol{H}_n^T \boldsymbol{H}_n)\right\},\end{aligned} \tag{17.9}$$

where $\xi^* = \xi_\infty^*$ if one is determining the optimal asymptotic design and $\xi^* = \xi_n^*$ if the optimization is based directly on only n measurements being available (so $w_i n$ is an integer for all i). Note that for any selected ξ,

$$\boldsymbol{H}_n^T \boldsymbol{H}_n = \sum_{k=1}^{n} \boldsymbol{h}_k \boldsymbol{h}_k^T = n\sum_{i=1}^{N} w_i \boldsymbol{f}(\boldsymbol{\chi}_i)\boldsymbol{f}(\boldsymbol{\chi}_i)^T \tag{17.10}$$

(Exercise 17.3).

For convenience and consistency with most literature in optimal design, the precision matrix for use in calculating a *D*-optimal solution is usually scaled so that it does not grow with n. Then, the maximization of the criterion may be interpreted as maximizing the information per sample. Hence, in the notation of the *D*-optimal criterion (17.4), let the precision matrix $\boldsymbol{M}(\xi) = \boldsymbol{M}(\boldsymbol{\theta}, \xi)$ be

$$\boldsymbol{M}(\xi) = \sum_{i=1}^{N} w_i \boldsymbol{f}(\boldsymbol{\chi}_i)\boldsymbol{f}(\boldsymbol{\chi}_i)^T, \tag{17.11}$$

where $\boldsymbol{M}(\xi)$ is written for $\boldsymbol{M}(\boldsymbol{\theta}, \xi)$ because the right-hand side of (17.11) does not depend on $\boldsymbol{\theta}$ (this contrasts with nonlinear models in Section 17.4, where the precision matrix *does* depend on $\boldsymbol{\theta}$). Note that $\boldsymbol{M}(\xi)$ differs from $\boldsymbol{H}_n^T \boldsymbol{H}_n$ by only the multiple n in (17.10) (i.e., $n\boldsymbol{M}(\xi) = \boldsymbol{H}_n^T \boldsymbol{H}_n$). From (17.8)–(17.10), maximizing the determinant of $\boldsymbol{M}(\xi)$ in (17.11) is equivalent to minimizing $\det[\mathrm{cov}(\hat{\boldsymbol{\theta}}^{(n)})]$. Hence, we let the expression in (17.11) be the precision matrix for use with *D*-optimality in (17.4) for the linear model. Although not shown in (17.11), another form of weighting (to complement the w_i) is to weight the summands by an "efficiency function" (e.g., Fedorov, 1972, p. 88). This weighting can be used, say, to compensate for greater variance in the outputs at particular inputs. We do not pursue that extension here.

There is a useful geometric interpretation of the problem in (17.9) in terms of confidence regions for $\boldsymbol{\theta}$. Suppose that the errors v_k are normally distributed. Then, the set

$$\left\{\boldsymbol{\theta}: (\hat{\boldsymbol{\theta}}^{(n)} - \boldsymbol{\theta})^T \boldsymbol{H}_n^T \boldsymbol{H}_n (\hat{\boldsymbol{\theta}}^{(n)} - \boldsymbol{\theta}) \le \text{constant}\right\} \tag{17.12}$$

is the collection of $\boldsymbol{\theta}$ values that would be in a confidence ellipsoid about $\hat{\boldsymbol{\theta}}^{(n)}$ at a specified probability level. That is, the null hypothesis that $\boldsymbol{\theta}$ is the true parameter vector would not be rejected for any $\boldsymbol{\theta}$ in the above set. The indicated constant depends on σ^2 and the specified probability level (e.g., 0.95). Regardless of the specific value of the constant, the volume of the set above is proportional to $[\det(\boldsymbol{H}_n^T \boldsymbol{H}_n)]^{-1/2}$ (Silvey, 1980, p. 10; Box and Draper, 1987, pp. 490–491). It is natural to want to make this confidence region as "tight" (small) as possible. Because maximizing $\det[\boldsymbol{M}(\xi)]$ according to *D*-optimality is equivalent to minimizing $[\det(\boldsymbol{H}_n^T \boldsymbol{H}_n)]^{-1}$, the *D*-optimal solution minimizes the size of the confidence region.

Let us now illustrate *D*-optimality based on a simple example. This example illustrates the common phenomenon of multiple (equivalent) *D*-optimal solutions. This is also one of the relatively rare problems that can be solved by simple analytical means. In most practical cases, a numerical search procedure is required to find a solution.

Example 17.5—*D*-optimality for simple regression model. Consider the linear model

$$z = [1, x]\boldsymbol{\theta} + v,$$

where $-1 \le x \le 1$ is a scalar input. Hence, $\boldsymbol{\theta}$ is composed of two terms, an intercept and a slope. Suppose that $n = 3$ measurements will be used to estimate $\boldsymbol{\theta}$. The aim is to determine the three input values x_1, x_2, and x_3. The measurement and precision matrices are

$$\boldsymbol{H}_3 = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \end{bmatrix} \text{ and } \boldsymbol{M}(\xi) = \tfrac{1}{3}\boldsymbol{H}_3^T\boldsymbol{H}_3 = \tfrac{1}{3}\begin{bmatrix} 3 & x_1+x_2+x_3 \\ x_1+x_2+x_3 & x_1^2+x_2^2+x_3^2 \end{bmatrix}.$$

Hence,

$$\det[3\boldsymbol{M}(\xi)] = \det(\boldsymbol{H}_3^T\boldsymbol{H}_3) = 3(x_1^2+x_2^2+x_3^2) - (x_1+x_2+x_3)^2 .$$

By inspection, there are six combinations of input values that yield the maximum value $\det(\boldsymbol{H}_3^T\boldsymbol{H}_3) = 8$. Two of the six solutions are

$$x_1 = x_2 = 1, x_3 = -1 \text{ and } x_1 = x_2 = -1, x_3 = 1. \tag{17.13}$$

More generally, all six solutions fit one of the two following general patterns for the optimal finite-sample design ξ_3^*:

$$\xi_3^* = \begin{Bmatrix} \chi_1 = 1 & \chi_2 = -1 \\ \frac{2}{3} & \frac{1}{3} \end{Bmatrix} \text{ or } \xi_3^* = \begin{Bmatrix} \chi_1 = -1 & \chi_2 = 1 \\ \frac{2}{3} & \frac{1}{3} \end{Bmatrix}.$$

So, for the two sample solutions in (17.13) above, $\chi_1 = x_1 = x_2$ and $\chi_2 = x_3$. ❑

17.2.2 Some Properties of Asymptotic Designs

Asymptotic designs hold a special place in the theory and practice of optimal designs. They provide a useful approximation to the often-intractable optimal finite-sample design. Listed below are a few key properties of *D*-optimal asymptotic designs as applied to linear models (e.g., Atkinson and Donev, 1992, pp. 116–117; Pukelsheim, 1993, Chap. 8).

(i) **Nonuniqueness of design.** For a given problem, a *D*-optimal design ξ_∞^* may not be unique. If $\xi_{\infty;1}^*$ and $\xi_{\infty;2}^*$ are two asymptotically optimal designs, $\lambda\xi_{\infty;1}^* + (1-\lambda)\xi_{\infty;2}^*$ is also an asymptotically optimal design for any $0 \le \lambda \le 1$.

(ii) **Bounds on number of support points.** There exists a *D*-optimal design ξ_∞^* where the number of support points N satisfies $p \le N \le p(p+1)/2$. (Property (i) above also indicates that there may be optimal designs with $N > p(p+1)/2$. The number of points N cannot be less than p because that would cause $\boldsymbol{H}_n^T\boldsymbol{H}_n$, which is inverted to form the least-squares estimate in (17.6), to be singular.)

(iii) **Bounds on optimality criteria.** The *D*-optimal criteria for the asymptotic and finite-sample designs are related by

$$1 \le \frac{\det \boldsymbol{M}(\xi_\infty^*)}{\det \boldsymbol{M}(\xi_n^*)} \le \frac{n^p}{n(n-1)\cdots(n+1-p)}.$$

A famous result in experimental design, the equivalence theorem of Kiefer and Wolfowitz (1960), provides the foundation for several methods of solving for D-optimal designs.[4] This result shows that the asymptotic D-optimal design ξ_∞^* is the same as a design from a minimax strategy producing the minimum variance for model predictions. Aside from this useful alternative interpretation of the D-optimal solution, the necessary and sufficient conditions of this theorem provide a practical structure for developing numerical algorithms to obtain solutions in nontrivial problems.

Theorem 17.1 (Kiefer–Wolfowitz equivalence theorem). Consider the linear model in (17.5) and let $\mathcal{X}$ be a compact (closed and bounded) set representing the domain of allowable values for inputs x. The following statements are equivalent:

(a) The asymptotic design ξ_∞^* is D-optimal.

(b) $\boldsymbol{f}(x)^T \boldsymbol{M}(\xi_\infty^*)^{-1} \boldsymbol{f}(x) \le p$ for all $x \in \mathcal{X}$.

(c) ξ_∞^* minimizes $\max_{x \in \mathcal{X}} \boldsymbol{f}(x)^T \boldsymbol{M}(\xi)^{-1} \boldsymbol{f}(x)$ with this minimax solution implying that $\boldsymbol{h}_i^T \boldsymbol{M}(\xi_\infty^*)^{-1} \boldsymbol{h}_i = p$ at all points $\chi_i \in \mathcal{X}$ of the design ξ_∞^*, where $\boldsymbol{h}_i = \boldsymbol{f}(\chi_i)$.

Aside from Kiefer and Wolfowitz (1960), Theorem 17.1 is proved in Silvey (1980, pp. 19–23) and Pukelsheim (1993, p. 212). Because $\hat{z}(x) = \boldsymbol{f}(x)^T \hat{\theta}^{(n)}$ and $\text{var}[\hat{z}(x)]$ is proportional to $\boldsymbol{f}(x)^T \boldsymbol{M}(\xi_\infty^*)^{-1} \boldsymbol{f}(x)$ (from (17.8), (17.10) and (17.11)), (b) and (c) are statements regarding the variance of the predictions $\hat{z}(x)$. The maximum variance (over $\mathcal{X}$) of the predictions $\hat{z}(x)$ is minimized when an optimal design ξ_∞^* is used to form $\hat{\theta}^{(n)}$. Further, this maximum variance is realized at the design points $\boldsymbol{h}_i = \boldsymbol{f}(\chi_i)$. The example below shows how the equivalence theorem can be used to obtain a practical solution to a design problem. In the example, the theorem is used to confirm a hunch regarding a likely optimal design.

Example 17.6—Application of equivalence theorem to simple curvilinear model. Consider a $p = 3$ model involving a second-order polynomial in a scalar x: $z = [1, x, x^2]\theta + v$, where it is known that $-1 \le x \le 1$. Based on intuition, it seems that an $N = 3$ design is a candidate for the optimal, where we take $n/N = n/3$ observations at each of the design points $\chi_1 = -1$, $\chi_2 = 0$, and $\chi_3 = 1$. So, the same three input values are repeated until n measurements are reached. For small n, this design may not be achievable (since n may not be divisible by 3), but for

[4] This is the Kiefer–Wolfowitz of finite-difference stochastic approximation (FDSA) fame (Chapter 6).

large n, this design is achievable to within a negligible error even if n is not divisible by 3. By (17.2), this candidate design may be written as

$$\xi = \begin{Bmatrix} \chi_1 & \chi_2 & \chi_3 \\ w_1 & w_2 & w_3 \end{Bmatrix} = \begin{Bmatrix} -1 & 0 & 1 \\ 1/3 & 1/3 & 1/3 \end{Bmatrix}. \tag{17.14}$$

Because $\mathbf{f}(x) = [1, x, x^2]^T$, this design leads to the three input vectors:

$$\mathbf{f}(\chi_1) = [1 \quad -1 \quad 1]^T,$$

$$\mathbf{f}(\chi_2) = [1 \quad 0 \quad 0]^T,$$

$$\mathbf{f}(\chi_3) = [1 \quad 1 \quad 1]^T.$$

From (17.11), a simple calculation shows that

$$\boldsymbol{M}(\xi) = \sum_{i=1}^{3} w_i \mathbf{f}(\chi_i)\mathbf{f}(\chi_i)^T = \frac{1}{3}\begin{bmatrix} 3 & 0 & 2 \\ 0 & 2 & 0 \\ 2 & 0 & 2 \end{bmatrix}.$$

With the design in (17.14), simple calculations show that

$$\mathbf{f}(x)^T \boldsymbol{M}(\xi)^{-1}\mathbf{f}(x) = \frac{3}{4}\left[4 - 6x^2(1-x^2)\right]$$

$$\leq 3 \quad \text{for all } x \in \mathcal{X} = [-1, 1].$$

Because the upper bound above is the same as p, it is known from (b) of the equivalence theorem that the solution in (17.14) is, in fact, the asymptotically optimal solution ξ_∞^* (see also Exercise 17.5). Figure 17.1 depicts the variance function $\mathbf{f}(x)^T \boldsymbol{M}(\xi_\infty^*)^{-1}\mathbf{f}(x)$, showing that it is maximized at the three design points, as indicated in part (c) of Theorem 17.1. ❑

Let us emphasize that the results of Theorem 17.1 pertain to the optimal *asymptotic* design, not to the optimal finite-sample design. This can be illustrated using the problem of Example 17.6. The optimal asymptotic design requires that n be divisible by 3. If, say, $n = 4$, numerical analysis may be used to show that the optimal finite-sample design places the "extra" measurement at any one of the three design points, −1, 0, or 1, above (Atkinson and Donev, 1992, p. 99). The D-optimal criterion is identical for any such allocation of the extra measurement. So, if the center point is the one that is replicated, the optimal finite-sample design is

$$\xi_4^* = \begin{Bmatrix} -1 & 0 & 1 \\ 1/4 & 1/2 & 1/4 \end{Bmatrix}. \tag{17.15}$$

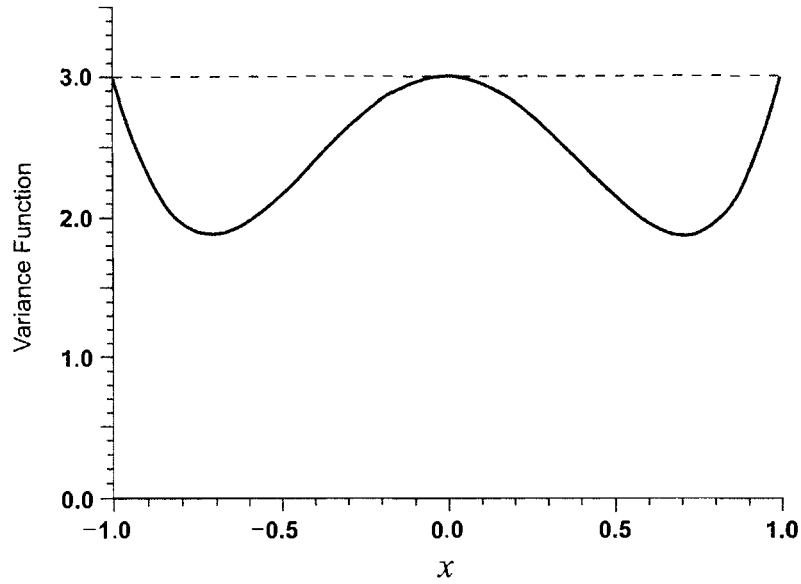

Figure 17.1. Plot of variance function $\boldsymbol{f}(x)^T \boldsymbol{M}(\xi_\infty^*)^{-1}\boldsymbol{f}(x)$ for varying input x in Example 17.6. This plot illustrates a consequence of Theorem 17.1: the variance function is bounded above by p and is maximized at the design points $x = \chi_1, \chi_2$, and χ_3.

Exercise 17.8 asks the reader to produce a plot of the variance function $\boldsymbol{f}(x)^T \boldsymbol{M}(\xi_4^*)^{-1}\boldsymbol{f}(x)$. In contrast to Figure 17.1, this plot illustrates that for this finite-sample design the variance function is *not* maximized at the design points.

The example below illustrates one of the standard "tricks" in developing an optimal design. In particular, it is shown that a change of variables may sometimes transform a messy or intractable problem to one that is quite straightforward. Furthermore, this problem illustrates a case where the emphasis is on maximizing the accuracy of only a subset of the parameters being estimated.

Example 17.7—Change of variables. It is convenient in many problems to redefine the variables. Suppose that $z = [e^{-\lambda x/2}, e^{-\lambda x}]\boldsymbol{\theta} + v$, where $\lambda > 0$ is some known parameter, $0 \le x < \infty$, and v has mean zero. Directly solving for the optimal design in such a model is cumbersome. Consider the change of variable, $u = e^{-\lambda x/2}$. Then, the regression model has the relatively simple curvilinear form $z = [u, u^2]\boldsymbol{\theta} + v$, $0 < u \le 1$.

Let $\boldsymbol{\theta} = [\beta_1, \beta_2]^T$ and $\hat{\boldsymbol{\theta}}^{(n)} = [\hat{\beta}_1, \hat{\beta}_2]^T$. Suppose that there is some question about whether the $e^{-\lambda x}$ contribution (corresponding to u^2) should be included in the model. So, the analyst intends to run a hypothesis test on whether

$\beta_2 = 0$. To strengthen the conclusions of the test, the aim is to choose the inputs (x) to minimize the variance of $\hat{\beta}_2$.

A candidate design for the variable u is

$$\xi_u \equiv \begin{Bmatrix} \mu & 1 \\ w & 1-w \end{Bmatrix}, \quad 0 \le w \le 1,$$

where $0 < \mu \le 1$ is a value to be determined and the other value of the design measure is known to be $u = 1$ because any other design can be scaled up, with a resulting increase in all elements of $\boldsymbol{M}(\xi_u)$ (Atkinson and Donev, 1992, p. 102). Using (17.11),

$$\boldsymbol{M}(\xi_u) = w\begin{bmatrix} \mu \\ \mu^2 \end{bmatrix}\begin{bmatrix} \mu & \mu^2 \end{bmatrix} + (1-w)\begin{bmatrix} 1 \\ 1 \end{bmatrix}\begin{bmatrix} 1 & 1 \end{bmatrix} = \begin{bmatrix} 1-w+w\mu^2 & 1-w+w\mu^3 \\ 1-w+w\mu^3 & 1-w+w\mu^4 \end{bmatrix}.$$

Hence,

$$\begin{aligned} \det[\boldsymbol{M}(\xi_u)] &= (1-w+w\mu^2)(1-w+w\mu^4) - (1-w+w\mu^3)^2 \\ &= w(1-w)\mu^2(1-\mu)^2. \end{aligned}$$

Using the standard formula for the inverse of a 2×2 matrix, the (2, 2) element of $\boldsymbol{M}(\xi_u)^{-1}$ is

$$\frac{1-w+w\mu^2}{w(1-w)\mu^2(1-\mu)^2}.$$

Minimizing the above expression yields $\mu = \sqrt{2} - 1$ and $w = 1/\sqrt{2}$. Converting back to the original variable (x) gives the following asymptotically optimal design for minimizing $\text{var}(\hat{\beta}_2)$:

$$\begin{Bmatrix} \chi_1 & \chi_2 \\ w_1 & w_2 \end{Bmatrix} = \begin{Bmatrix} -2\log(\sqrt{2}-1)/\lambda & 0 \\ 1/\sqrt{2} & 1-1/\sqrt{2} \end{Bmatrix} \approx \begin{Bmatrix} 1.763/\lambda & 0 \\ 0.707 & 0.293 \end{Bmatrix}.$$

(Exercise 17.9 asks you to verify via Theorem 17.1 that this solution is *not* the D-optimal design for the full vector $\boldsymbol{\theta}$.) ❑

17.2.3 Orthogonal Designs

The optimization problem for design with linear models, as presented in (17.9), is quite general. In fact, as we have seen, more than one design ξ may be a D-optimal solution. One of the ways in which the number of potential solutions can

be reduced while introducing other desirable properties is to impose an orthogonality constraint. Intuitively, orthogonality tends to separate the effects of the input factors as much as possible. This is especially useful when effects may otherwise be *aliased* (so one effect masks another effect), as discussed in Wu and Hamada (2000, Sect. 4.4). Note that not all orthogonal designs are *D*-optimal. One advantage of orthogonal designs is that the estimates for the elements within $\boldsymbol{\theta}$ are uncorrelated. Hence, hypothesis tests under the normality assumption (where uncorrelatedness is equivalent to independence) may be carried out for each element separately without having to account for dependence of the estimates.

Recall that the model for the stacked vector of n measurements is $\boldsymbol{Z}_n = \boldsymbol{H}_n\boldsymbol{\theta} + [v_1, v_2, \ldots, v_n]^T$. Suppose that (17.7) applies (i.e., $z_k = [1, \boldsymbol{x}_k^T]\boldsymbol{\theta} + v_k$). Then,

$$\boldsymbol{H}_n = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \boldsymbol{X}_n \Bigg],$$

where $\boldsymbol{X}_n = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n]^T$. The interest here is in designs where the column vectors within $\boldsymbol{H}_n$ are at right angles to one another (each column contains the inputs multiplying a specific element of $\boldsymbol{\theta}$). This achieves orthogonality and makes the estimates for all pairs of elements of $\boldsymbol{\theta}$ uncorrelated, as we now discuss.

Recall that $r = \dim(\boldsymbol{x}_k)$. The linear model (17.7) implies that $r = p - 1$ (for other model forms—e.g., nonlinear—r as a representation of the dimension of the input vector may not equal $p - 1$). Let the ith column of $\boldsymbol{X}_n$ be denoted by $\boldsymbol{x}_n^{(i)}$ (not to be confused with the ith row, $\boldsymbol{x}_i$) and let $\mathbf{1}_n$ denote an n-dimensional vector of 1's. There are $p - 1$ columns in $\boldsymbol{X}_n$. The orthogonality requirement is satisfied if $\left(\boldsymbol{x}_n^{(i)}\right)^T \boldsymbol{x}_n^{(j)} = 0$ for all $i \neq j$ and $\mathbf{1}_n^T \boldsymbol{x}_n^{(i)} = 0$ for all i. That is, the inner product of the n-dimensional vector corresponding to the contribution of one factor times the vector of contributions of a different factor is zero. In matrix notation, $\boldsymbol{X}_n$ should be chosen so that

$$\boldsymbol{H}_n^T \boldsymbol{H}_n = \text{diagonal matrix}. \tag{17.16}$$

Designs satisfying (17.16) are called *fully orthogonal designs*. (Note that each $\boldsymbol{x}_n^{(i)}$ is a special case of the column $\boldsymbol{h}_{\bullet i+1}$, according to the general definition of columns in the $\boldsymbol{H} = \boldsymbol{H}_n$ matrix in Subsection 3.1.2.)

To see how (17.16) implies that the estimates within $\hat{\boldsymbol{\theta}}^{(n)}$ are uncorrelated, recall from (17.8) that $\text{cov}\left(\hat{\boldsymbol{\theta}}^{(n)}\right) = (\boldsymbol{H}_n^T \boldsymbol{H}_n)^{-1}\sigma^2$. In particular, with $\boldsymbol{\theta} = [\beta_0, \beta_1, \ldots, \beta_{p-1}]^T$ and $\hat{\boldsymbol{\theta}}^{(n)} = [\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_{p-1}]^T$, then under the

orthogonality requirement (17.16), $\text{cov}(\hat{\beta}_i, \hat{\beta}_j) = 0$ for all $i, j \geq 0$ and $i \neq j$ (following convention, the "0" subscript denotes the additive constant; i.e., β_0 is the additive constant).

Choices such as 2^r factorial (cubic) designs for experiments with r factors taking on one of two levels (e.g., Draper, 1988; Montgomery, 2001, Chaps. 6 and 7) satisfy the above orthogonality condition. This factorial design requires measurements at $n = 2^r$ levels of the factors during the experimentation process. Factorial designs are those such that factors in $\boldsymbol{x}$ are varied together rather than one at a time. As discussed in Montgomery (2001, Chap. 5), these designs are more efficient at exposing input–output relationships than varying only one input variable at a time. A simplex design (analogous to the simplex patterns of the search algorithm in Section 2.4) also satisfies this orthogonality condition while only requiring $n = r + 1$ levels of the factors. Other examples of acceptable orthogonal designs are given in Draper (1988), among many other references. Illustrations of two orthogonal designs for $r = 2$ and $r = 3$ are given in Figures 17.2 and 17.3, where $\boldsymbol{x}_k = [x_{k1}, x_{k2}]^T$ or $\boldsymbol{x}_k = [x_{k1}, x_{k2}, x_{k3}]^T$, as appropriate.

The above discussion pertains to fully orthogonal designs. In some cases, orthogonality may only apply to a subset of the columns within $\boldsymbol{H}_n$. This is especially the case in the curvilinear setting where elements within $\boldsymbol{X}_n$ may represent related inputs (e.g., two of the columns may represent x and x^2 effects). In such cases, only some of the estimates within $\hat{\boldsymbol{\theta}}^{(n)}$ will generally be uncorrelated. For example, Box and Draper (1987, p. 483) show a curvilinear case, where $\hat{\beta}_0$ is correlated with certain other elements in $\hat{\boldsymbol{\theta}}^{(n)}$, but where $\text{cov}(\hat{\beta}_i, \hat{\beta}_j) = 0$ for all $i, j \geq 1$ and $i \neq j$. Example 17.8 is another case of such a partially orthogonal design. This model has $r = p - 1 = 2$.

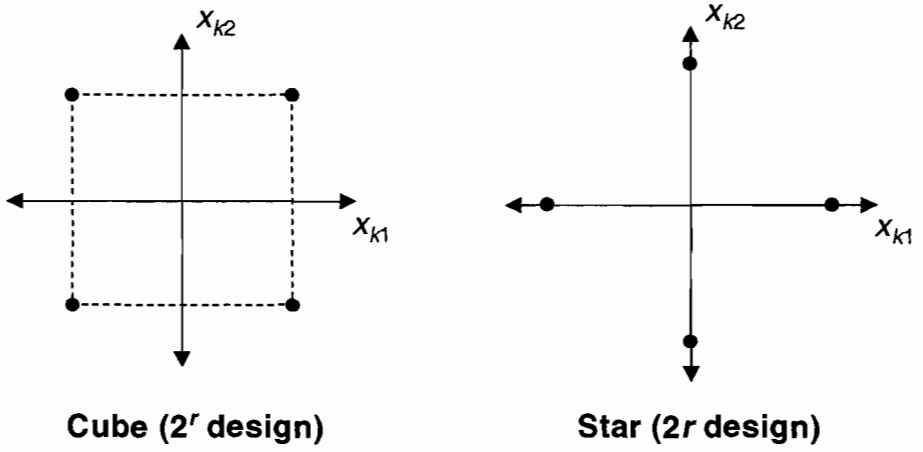

Figure 17.2. Two possible orthogonal designs for $r = 2$; points are symmetric with respect to origin.

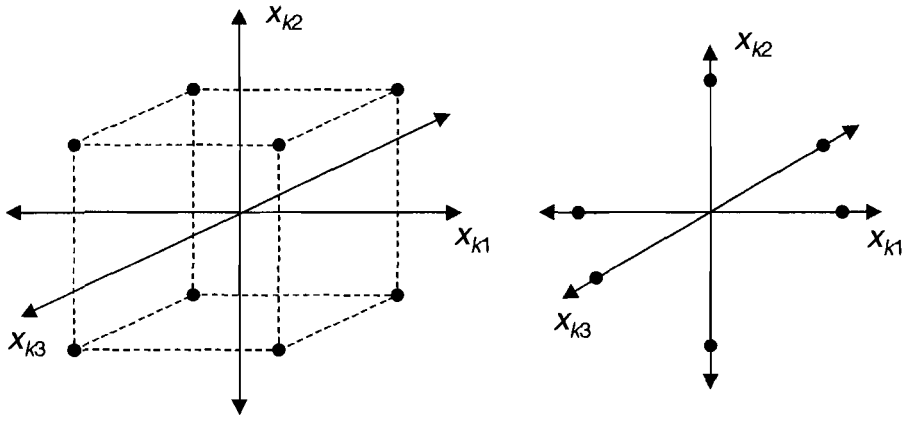

Figure 17.3. Two possible orthogonal designs for $r = 3$; points are symmetric with respect to origin.

Example 17.8—Partially orthogonal design for a simple curvilinear model. Consider again the $r = 2$ curvilinear model of Example 17.6: $z = [1, x, x^2]\boldsymbol{\theta} + v$, where it is known that $-1 \leq x \leq 1$. The aim is to pick a design based on three levels of the input x given that n is divisible by 3. So, the same three input values are repeated until n measurements are reached. An intuitively appealing candidate design—and one shown to be D-optimal in Example 17.6—is the design where $x = -1$, 0, and 1 with equal frequency. The resulting $n \times p$ design matrix $\boldsymbol{H}_n$ is composed of a vertical stack of $n/3$ blocks of the form

$$\boldsymbol{H}_3 = \begin{bmatrix} 1 & \\ 1, & \boldsymbol{X} \\ 1 & \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}.$$

Therefore,

$$\boldsymbol{H}_3^T \boldsymbol{H}_3 = \begin{bmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 0 & 2 \\ 0 & 2 & 0 \\ 2 & 0 & 2 \end{bmatrix}.$$

Hence,

$$(\boldsymbol{H}_n^T \boldsymbol{H}_n)^{-1} = \frac{3}{n} \begin{bmatrix} 1 & 0 & -1 \\ 0 & \frac{1}{2} & 0 \\ -1 & 0 & \frac{3}{2} \end{bmatrix},$$

indicating that $\hat{\beta}_0$ is negatively correlated with $\hat{\beta}_2$, but that the two elements associated with the input x ($\hat{\beta}_1$ and $\hat{\beta}_2$) are uncorrelated with each other. ❑

17.2.4 Sketch of Algorithms for Finding Optimal Designs

One fundamental issue has been omitted in the discussion above: How can a *D*-optimal design be constructed in arbitrary linear models of the form (17.5)? In the examples above—and in many applied problems and examples elsewhere—one "guesses" at a design based on intuitive considerations and then verifies it by the equivalence theorem, Theorem 17.1. In some problems, however, an intuitive solution may be elusive. Fedorov (1972, Sects. 2.5 and 2.6), Silvey (1980, Chap. 4), Rustagi (1994, Sect. 2.5), Müller and Parmigiani (1995), and Angelis et al. (2001) are among the many references that discuss search algorithms for solving this nontrivial problem. Some of these algorithms are variations of the types of stochastic algorithms seen in earlier chapters. These algorithms are directed to finite-sample or asymptotic designs. Three of the recommended methods involve random search, stochastic approximation, and simulated annealing, core stochastic search methods that were considered in Chapters 2 and 4–8.

There are also some popular *exchange algorithms* for determining optimal finite-sample designs. Important variations on this general approach were introduced in Fedorov (1972, Chaps. 3 and 4), Mitchell (1974), and Cook and Nachtsheim (1980). These and other variations are widely available in commercial software (e.g., the OPTEX package in the SAS system of statistical software). Although we do not describe these deterministic algorithms in detail here, let us provide a sketch of the basic idea. Consider a candidate set of design points $\{\boldsymbol{\chi}_i\}$. The list might include the points in a grid of the search space $\boldsymbol{\mathcal{X}}$. Another possibility might be all or some of the points in an orthogonal design (such as the points in a 2^r or 3^r factorial design). The list of points should be rich enough to contain an adequate approximation of the optimal design ξ_n^*.

Based on the candidate set above, there are two broad steps in finding an optimal design by an exchange algorithm. First, one makes an initial guess by choosing the n inputs to occupy a subset of points in the candidate set. Depending on the value of n and the size of the candidate set, there may be points in the candidate set that are assigned multiple inputs or no inputs. This initial design produces an associated value of the *D*-optimal criterion $\det[\boldsymbol{M}(\xi)]$. In the second broad step, an iterative process is implemented to shuffle inputs among the elements in the candidate set with the aim of increasing the criterion. For example, an input that has been assigned to one of the candidates may be shifted to a previously unoccupied point in the candidate set. Various rules exist that govern such exchanges, but the overall theme is that at each iteration one seeks exchanges that maximize the improvement in the criterion. A good review of exchange and other procedures is in Atkinson and Donev (1992, Chap. 15).

An important property for optimization is that $\log\det[\boldsymbol{M}]$ is a strictly concave function on the set of positive definite matrices $\boldsymbol{M}$ (Exercise 17.11).

Such concavity is useful as it guarantees that the optimal $\boldsymbol{M}$ is unique. This follows because maximizing $\det[\boldsymbol{M}]$ is the same as maximizing $\log\det[\boldsymbol{M}]$ by the monotonicity of the log operation (i.e., if $a > b > 0$, then $\log(a) > \log(b)$). Unfortunately, this concavity in $\boldsymbol{M}$ does not tell us that $\log\det[\boldsymbol{M}] = \log\det[\boldsymbol{M}(\xi)]$ is concave in ξ. Hence, there is no guarantee that ξ^* is unique. In particular, multiple global solutions ξ^* may yield the same maximizing value of $\boldsymbol{M}$ (such as in Example 17.5). Further, there may be multiple *local* maxima to $\log\det[\boldsymbol{M}(\xi)]$, complicating the optimization process.

17.3 RESPONSE SURFACE METHODOLOGY

Response surface methodology (RSM) is a sequential experimentation strategy for building an understanding of the relationship between input and output variables in a complex process. This section is a synopsis of RSM, including some discussion of its relationship to other aspects of experimental design.

Suppose that the *true process* (not model) for the output z satisfies

$$z = f(\boldsymbol{x}) + \text{noise}$$

for some unknown (usually nonlinear) function $f(\boldsymbol{x})$ and inputs $\boldsymbol{x}$. For simplicity and consistency with previous analysis, we consider a scalar output z and (generally) vector input $\boldsymbol{x}$. RSM is based on sequentially approximating $f(\boldsymbol{x})$ via low-order regression models of the form $\boldsymbol{\ell}(\boldsymbol{x})^T\boldsymbol{\theta}$, as shown in (17.5). Historically, RSM began as a tool in the chemical process industry (Box and Wilson, 1951), but it has evolved to be a general tool for the sequential analysis of complex processes (e.g., Khuri and Cornell, 1987; Wu and Hamada, 2000, Chap. 9). This section includes some pointers toward more complete discussions elsewhere.

RSM may be used to build an understanding of the input–output relationship, or, more specifically, to determine inputs that optimize the process. RSM is a sequential design strategy in that it is based on a recursive set of localized experiments aimed at converging to global optimal design. RSM applies to nonlinear problems by localizing the function surface to where a linear-type (pure linear or curvilinear) model can be used.

Let us focus on the use of RSM for finding an optimal value of $\boldsymbol{x}$ with the aim of minimizing the mean response z (hence, $E(z)$ is the loss function to be minimized). The basic idea is to first perform several experiments at $\boldsymbol{x}$ values near the currently estimated optimal $\boldsymbol{x}$ value. From these experiments, one builds up a "response surface" describing the local behavior of the process under study. This response surface is typically—but not necessarily—a first- or second-order regression polynomial in the vector of factors $\boldsymbol{x}$. The response surface is then used to move toward a better value of $\boldsymbol{x}$, and the process repeats itself. The success of RSM is based on the validity of repeated localized curve fitting and subsequent optimization. The methodology can be summarized in the following steps.

Outline of RSM for Optimizing $\boldsymbol{x}$

Step 0 **(Initialization)** Make an initial guess at the optimal value of $\boldsymbol{x}$.

Step 1 Collect responses z from several $\boldsymbol{x}$ values in the neighborhood of the current best estimate of the best $\boldsymbol{x}$ value. To enhance the fitting process, use experimental design to determine the values of $\boldsymbol{x}$ with which to collect measurements z.

Step 2 From the $\boldsymbol{x}$, z pairs in step 1, fit a regression (response surface) model over the region around the current best estimate of the optimal $\boldsymbol{x}$ levels.

Step 3 Based on the response surface in step 2, estimate a path of steepest descent in factor space.

Step 4 Perform a series of experiments at $\boldsymbol{x}$ values along the path of steepest descent until no additional improvement in the z response is obtained. This $\boldsymbol{x}$ value represents the new estimate of the best vector of factor levels.

Step 5 Go to step 1 and repeat the process until a final best factor level is obtained. Second-order (Hessian) analysis may be employed to ensure that the final solution is a minimum (given the aim to minimize $E(z)$).

A critical part of the RSM steps above is the choice of the experimental points in step 1. Optimal design for linear models, as described in Section 17.2, may be used to choose these points. The number and value of such points depend on the choice of the regression model in step 2. RSM is almost always based on the pure linear regression model or the second-order curvilinear form:

$$z_k = \beta_0 + \boldsymbol{\beta}^T \boldsymbol{x}_k + \boldsymbol{x}_k^T \boldsymbol{B} \boldsymbol{x}_k + v_k, \tag{17.17}$$

where β_0, $\boldsymbol{\beta}$, and $\boldsymbol{B}$ represent a scalar, vector, and symmetric matrix to be estimated (if the quadratic model is being used). Model (17.17) can be equivalently written in the linear form of (17.5), $z_k = \boldsymbol{h}_k^T \boldsymbol{\theta} + v_k$, where $\boldsymbol{h}_k = \boldsymbol{h}(\boldsymbol{x}_k)$ is the design vector dependent on input $\boldsymbol{x} = \boldsymbol{x}_k$. Hence, $\boldsymbol{\theta}$ represents the unique parameters in β_0, $\boldsymbol{\beta}$, and $\boldsymbol{B}$ that are to be estimated. It is often the case that the first few steps of RSM are based on a simple linear model $z_k = \beta_0 + \boldsymbol{\beta}^T \boldsymbol{x}_k + v_k$. A full factorial ($2^r$) design may be useful; this design is D-optimal for such simple linear models (Montgomery, 2001, p. 469). Later steps, requiring greater accuracy, employ the full quadratic-input model as in (17.17).

In the case where the quadratic (curvilinear) model (17.17) is applied, a large number of parameters may need to be estimated. Hence, some of the classical orthogonal designs discussed in Section 17.2 may no longer be appropriate because they do not provide the information needed to estimate the curvature effects. There are several popular ways to augment classical designs (such as the standard full factorial [cubic] designs illustrated in Figures 17.2 and 17.3). One is to take additional measurements at the center of the cube. Others, such as 3^r factorials, are discussed in Montgomery (2001, Chap. 9).

Given that $\dim(\boldsymbol{x}_k) = r$, the model in (17.17) has one constant, r linear terms, r squared terms, and $r(r-1)/2$ mixed second-order terms. This leads to a total of $p = r^2/2 + 3r/2 + 1$ terms in $\boldsymbol{\theta}$. To estimate the parameters in (17.17), the design must provide at least this number of data points. In cases where r is large, *fractional factorial* designs are sometimes used (e.g., Montgomery, 2001, Chaps. 8 and 9). These designs capture many of the benefits of full factorial designs, but with only a fraction of the number of measurements.

Lucas (1976) reports on the performance of several popular response surface designs using fewer measurements than the 2^r required in a full factorial design, including particular fractional factorial designs. In particular, certain "standard" designs requiring no more than $p + r$ measurements achieve a value of the D-optimal criterion that is at least 90 percent of the best possible value on problems in a given test set.

Given that one has settled on a design strategy, then step 3 of the RSM search proceeds along the path of steepest descent (as in Section 1.4). For the pure linear model, this is simply implemented by changing $\boldsymbol{x}$ by an amount proportional to the estimate of $-\boldsymbol{\beta}$. Responses are observed along this path until there is no significant improvement in the response (step 4). Of course, we do not know $\boldsymbol{\beta}$ (having only an estimate), so the gradient used in the steepest descent is only an estimate of the true gradient. Therefore, this represents a version of *stochastic* steepest descent as in the stochastic gradient form (Chapter 5) of the root-finding (Robbins–Monro) stochastic approximation algorithm. (Section 17.4 also discusses stochastic approximation for sequential design in nonlinear models.)

Alternatively, if the second-order curvilinear model (17.17) is used, then the stationary (zero-gradient) point of the second-order surface can be found and further analysis can be performed to determine if this stationary point is a minimum. Bisgaard and Ankenman (1996) report on a method for establishing confidence intervals for the eigenvalues of the Hessian matrix for the second-order model. This is useful in step 5 of the RSM process in order to statistically quantify (accounting for noise in the z values) whether the algorithm is converging to a minimum, maximum, or saddlepoint of the criterion $E(z)$. As above, stochastic approximation is relevant here since this response surface model (and associated gradient/Hessian) are only estimated (representing a noisy measurement of $E(z)$ and its derivatives).

While RSM has a long history of success, there is no guarantee of greater efficiency (i.e., fewer experiments) in reaching a solution than other stochastic optimization methods. In particular, stochastic gradient (Chapter 5) or gradient-free stochastic approximation (FDSA and SPSA, Chapters 6 and 7) may be useful in minimizing $E(z)$ without the process of sequential design. As with RSM, these methods are based on noisy measurements of the criterion (z values as noisy measurements of $E(z)$). There appears to be no formal study comparing RSM with such methods.

17.4 NONLINEAR MODELS

17.4.1 Introduction

Although significant challenges in experimental design arise with models $h(\boldsymbol{\theta}, \boldsymbol{x})$ linear in the parameters $\boldsymbol{\theta}$, much can (and has!) been said about the problem. This was discussed in Sections 17.2 and 17.3. For models that are nonlinear in $\boldsymbol{\theta}$ (such as neural networks, where $\boldsymbol{\theta}$ represent the connection weights being estimated), much less is known. A major challenge in design for nonlinear models is that the optimal design depends on $\boldsymbol{\theta}$. In contrast, in linear models of the form (17.5), the design does not depend on knowing $\boldsymbol{\theta}$. This dependence in nonlinear models leads to a conundrum. One is picking the design $\boldsymbol{\xi}$ with the aim of estimating the unknown $\boldsymbol{\theta}$, and yet one has to know $\boldsymbol{\theta}$ to pick the best $\boldsymbol{\xi}$!

This conundrum leads to various means of coping with the dependence on $\boldsymbol{\theta}$. These include simply producing a design based on one's best guess at $\boldsymbol{\theta}$, determining a design in a sequential manner by alternating between forming estimates of $\boldsymbol{\theta}$ and choosing a design, and augmenting the criterion in a Bayesian manner to reflect a prior distribution on $\boldsymbol{\theta}$. We discuss these three general approaches below.

As with the linear case, there are a number of design criteria. The dominant design criterion continues to be D-optimality (expression (17.4)), although several closely related performance measures are also used. As we have seen, D-optimality is directly applicable to both linear and nonlinear models through the Fisher information matrix and is oriented toward maximizing the precision of the parameter estimates. Following the previous sections, we continue to focus on D-optimality here, although the basic principles for search and optimization for an optimal design apply as well with other criteria. Recall that the D-optimal criterion depends on the precision matrix $\boldsymbol{M}(\boldsymbol{\theta}, \boldsymbol{\xi})$. The precision matrix here is the Fisher information matrix. In contrast to the batch estimator emphasized in the linear models of Sections 17.2 and 17.3, this section considers more general batch or recursive estimators. Hence, we use the generic notation $\hat{\boldsymbol{\theta}}_n$ to refer to an arbitrary estimator of $\boldsymbol{\theta}$ based on n data points (rather than the batch-specific notation $\hat{\boldsymbol{\theta}}^{(n)}$ used in Sections 17.2 and 17.3).

From Section 13.3, the information matrix $\boldsymbol{F}_n = \boldsymbol{F}_n(\boldsymbol{\theta}, \boldsymbol{X}_n)$ is

$$\begin{aligned}\boldsymbol{F}_n(\boldsymbol{\theta}, \boldsymbol{X}_n) &= E\left(\frac{\partial \log \ell}{\partial \boldsymbol{\theta}} \cdot \frac{\partial \log \ell}{\partial \boldsymbol{\theta}^T} \middle| \boldsymbol{\theta}, \boldsymbol{X}_n \right) \\ &= -E\left[\boldsymbol{H}_{\log \ell}(\boldsymbol{\theta}, \boldsymbol{Z}_n \,|\, \boldsymbol{X}_n) \middle| \boldsymbol{\theta}, \boldsymbol{X}_n \right],\end{aligned}$$

where ℓ is the likelihood function (the probability density or mass function in most typical cases), $\boldsymbol{H}_{\log \ell}$ is the Hessian matrix of the log-likelihood function, and the expectations are with respect to the randomness in the outputs $\boldsymbol{Z}_n = [z_1, z_2, \ldots, z_n]^T$ conditioned on $\boldsymbol{\theta}$ and the n input vectors represented by $\boldsymbol{X}_n$. We

saw in Subsection 13.3.3 that there is a close connection between $\boldsymbol{F}_n^{-1}$ and the covariance matrix of general parameter estimates $\hat{\boldsymbol{\theta}}_n$. Loosely speaking, $\boldsymbol{F}_n^{-1}$ is an expression of uncertainty in $\hat{\boldsymbol{\theta}}_n$ for a large class of estimators based on a *general* (nonlinear) regression model of the form $z_k = h(\boldsymbol{\theta}, \boldsymbol{x}_k) + v_k$. In particular, $\boldsymbol{F}_n^{-1}$ evaluated at $\boldsymbol{\theta}$ close to the optimum $\boldsymbol{\theta}^*$ is approximately equal to $\text{cov}(\hat{\boldsymbol{\theta}}_n)$ for large n.

The Fisher information matrix depends on $\boldsymbol{\theta}$ in nonlinear models. In contrast, in linear models of the form (17.5), the precision matrix $\boldsymbol{M}(\boldsymbol{\theta}, \xi) = \boldsymbol{M}(\xi) = \boldsymbol{H}_n^T \boldsymbol{H}_n / n$, which is a scaled form of the inverse covariance matrix, does not depend on $\boldsymbol{\theta}$. In Section 13.3 we suppressed the dependence of the information matrix on inputs, $\boldsymbol{X}_n$, simply writing $\boldsymbol{F}_n = \boldsymbol{F}_n(\boldsymbol{\theta})$. Here, of course, that dependence is critical, as we are fundamentally concerned with choosing the inputs, recalling that ξ is a convenient form for summarizing the allocation in $\boldsymbol{X}_n$. So, we will restore the dependence in the expressions for the information matrix here, writing $\boldsymbol{F}_n = \boldsymbol{F}_n(\boldsymbol{\theta}, \boldsymbol{X}_n)$.

Recall the fundamental goal of picking the design ξ that best helps in estimating $\boldsymbol{\theta}$. Given the close connection of $\boldsymbol{F}_n^{-1}$ to $\text{cov}(\hat{\boldsymbol{\theta}}_n)$, it is natural to have the precision matrix of the D-optimal criterion (17.4) satisfy $\boldsymbol{M}(\boldsymbol{\theta}, \xi) = \boldsymbol{F}_n(\boldsymbol{\theta}, \boldsymbol{X}_n)$. Hence, the aim in general nonlinear problems is to find a design satisfying

$$\arg\max_{\xi} \left\{ \det\left[\boldsymbol{M}(\boldsymbol{\theta}, \xi) \right] \right\} = \arg\max_{\boldsymbol{X}_n} \left\{ \det\left[\boldsymbol{F}_n(\boldsymbol{\theta}, \boldsymbol{X}_n) \right] \right\}. \tag{17.18}$$

The two related examples below consider a relatively simple scalar problem to illustrate how, in general, the information matrix and resulting optimal design depend on $\boldsymbol{\theta}$ in a nonlinear problem.

Example 17.9—Nonlinear scalar problem in estimation of mean number of organisms via dilution. Consider a problem involving the estimation of the mean number of organisms per unit volume in some substance. Suppose that the unknown mean to be estimated is θ. If the original substance is diluted with pure water by a multiplicative factor $x \geq 1$, the mean number of organisms per unit volume is θ / x. The interest here is to enhance the estimation of θ by determining the best dilution factor x (the user-specified input here). Suppose the laboratory equipment can only check for the absence or presence of organisms rather than count the number of organisms (e.g., Cochran, 1973). Let $z_k \in \{0, 1\}$ represent a binary random variable denoting the absence or presence of at least one organism in a sample (0 if not present; 1 if present). If it is assumed that the number of organisms per unit volume is random according to a Poisson distribution (as in Cochran, 1973; Cox and Reid, 2000, pp. 178–179), the binary output z_k then has a Bernoulli (binary) probability distribution defined by

$$P(z_k = 0) = \exp(-\theta / x_k) \text{ and } P(z_k = 1) = 1 - \exp(-\theta / x_k).$$

(Of course, more information would be available for estimating θ if it were possible to count the number of organisms in a sample, but it is assumed here that the laboratory equipment provides only the above binary information.)

Assume that the experimenter obtains n unit samples, each diluted according to x_k. The aim in the design problem is to determine the sequence of x_k's to enhance the estimation of θ. From the Bernoulli probability distribution above, the log-likelihood function for θ is

$$\log \ell(\theta|\boldsymbol{Z}_n) = \sum_{k=1}^{n} \left\{ \theta \frac{z_k - 1}{x_k} + z_k \log[1 - \exp(-\theta/x_k)] \right\} \tag{17.19}$$

(see Exercise 17.13). Applying the basic formula for the information "matrix" (number) above yields

$$F_n(\theta, \boldsymbol{X}_n) = \sum_{k=1}^{n} \frac{\exp(-\theta/x_k)}{x_k^2[1 - \exp(-\theta/x_k)]}$$

(see Exercise 17.13). Note that the information number depends on θ; hence the optimal design for the x_k's also depend on θ. Example 17.10 produces the D-optimal solution. ❑

Example 17.10—Optimal design for dilution problem. Consider the problem of Example 17.9. Suppose that the experimenter wants to determine the optimal finite-sample design. Because θ is a scalar, the D-optimal design is the design that maximizes $F_n(\theta, \boldsymbol{X}_n)$ (i.e., no determinant necessary). Note that maximizing one summand in $F_n(\theta, \boldsymbol{X}_n)$ is sufficient because each of the summands in $F_n(\theta, \boldsymbol{X}_n)$ has an identical form. Hence, it is sufficient to find an $x \geq 1$ that maximizes

$$\frac{e^{-\theta/x}}{x^2(1 - e^{-\theta/x})}$$

(so that all n measurements have the same input x). Intuitively, it seems undesirable to have $x >> 1$ because then θ/x is near zero (i.e., in n unit samples, z_k would almost always indicate nonpresence). In the notation of design, it can be shown using the expression above that the optimal design satisfies

$$\xi_n^* = \begin{Bmatrix} 0.63\theta \\ 1.0 \end{Bmatrix}$$

for any n (Exercise 17.14). That is, the inputs are set to $x_k = \chi_1 = 0.63\theta$ for all k (i.e., $w_1 = 1.0$ as shown in ξ_n^*). Given the requirement to have $x \geq 1$, this

solution is only valid if $\theta \geq 1/0.63 \approx 1.59$. This solution depends, of course, on the binary aspect of the data collection process. If, instead, each z_k represented a *count* of the number of organisms, the optimal design would be different. A practical implementation of the solution above would be an example of a "local design," local because it is valid in practice for θ near the value specified to obtain a realizable solution. ❑

17.4.2 Methods for Coping with Dependence on θ

There are three well-established means for coping with the dependence of the optimization criterion on $\boldsymbol{\theta}$ in nonlinear problems:

1. Simply assume a nominal value of $\boldsymbol{\theta}$ and develop an optimal design based on this fixed value during the optimization process. It is usually hoped that this nominal value is close to the unknown true value. This is called a *local design*, as illustrated in Example 17.10.
2. Use a sequential design strategy based on an iterated design and model fitting process. This strategy begins with an initial guess at $\boldsymbol{\theta}$, leading to an optimal design for a small number of inputs. Data are collected from the inputs and the value for $\boldsymbol{\theta}$ is updated, leading to a new design for another group of inputs. This process is repeated until a satisfactory estimate for $\boldsymbol{\theta}$ is obtained. Root-finding and stochastic gradient versions of stochastic approximation (SA) (Chapters 4 and 5) often play a role in sequential design.
3. Use a Bayesian strategy where a prior distribution is assigned to $\boldsymbol{\theta}$, reflecting uncertainty in the knowledge of the true value of $\boldsymbol{\theta}$. In this strategy, each value of the D-optimal (or other) criterion is evaluated at a design $\boldsymbol{\xi}$ after the $\boldsymbol{\theta}$ dependence has been integrated out according to the prior distribution.

While the local design approach 1 above is clearly the simplest, it has serious flaws in many applications. If, as usual, the dependence of the criterion on $\boldsymbol{\theta}$ is significant, the solution will be highly sensitive to the choice of the nominal value. For example, in the dilution problem above (Examples 17.9 and 17.10), the error in design is directly proportional to the error in the choice of the scalar θ. Nonetheless, it is sometimes useful to seek a locally optimal design in cases where one is confident in the knowledge of $\boldsymbol{\theta}$. It may also be useful in hypothesis testing where one is testing a null hypothesis of a specific value of $\boldsymbol{\theta}$; that value may be treated as truth in determining the optimal design. Further, local designs are used to provide the intermediate solutions in the sequential design approach.

We now give a more detailed discussion of approaches 2 and 3, sequential design and Bayesian design. Although there has been a fair amount of work published on these nonlinear design strategies, the number of practical applications has lagged. This is expected to change as commercial software is developed for particular approaches to nonlinear optimal design.

Sequential Design

Perhaps the most widely used approach to solving the problem posed in (17.18) is the sequential design method. The main idea is that the total resources available for the experiment are divided and the entire experiment is performed in a number of steps. Each step of the experiment uses a portion of the resources. At each step, analysis and parameter estimation are performed based on the step's available resources. Then a decision is made as to how to best use the next step's resources. The central benefit of sequential design is that one can take advantage of the natural learning process associated with experimentation. This allows the analyst to best focus resources in the latter stages of the design as more information becomes available about the nature of the process. For example, in medical studies, this may have ethical implications via allowing an adjustment of a treatment to avoid harm to patients.

The implementation of sequential design typically begins with an initial guess at a design. Then, using a small subset of the budget of input–output measurements, an estimate of $\boldsymbol{\theta}$ is formed from this initial design. This value of $\boldsymbol{\theta}$ is then used in the optimality criterion (17.18) to pick input levels for the next one or several inputs, leading to a new value of $\boldsymbol{\theta}$. This process is repeated until there is adequate convergence of the $\boldsymbol{\theta}$ estimate or until the full budget of measurements has been collected.

The steps below summarize the general sequential design strategy. A specific implementation requires details not specified in the steps below. We comment on some of these details in the discussion and example following the steps below.

Outline of Sequential Approach for Parameter Estimation and Optimal Design

Step 0 **(Initialization)** Make an initial guess at $\boldsymbol{\theta}$, say $\hat{\boldsymbol{\theta}}_0$. Allocate n_0 measurements to the initial design. Set $k = 0$, $n = 0$, and $\boldsymbol{F}_0(\hat{\boldsymbol{\theta}}_0, \boldsymbol{X}_0) = \boldsymbol{0}$ ($\boldsymbol{X}_0$ is only a notational placeholder here; it need not be explicitly defined in the initialization step).

Step 1 Given $\boldsymbol{X}_n$, choose the n_k inputs in $\boldsymbol{X} = \boldsymbol{X}_{n_k}$ to maximize $\det[\boldsymbol{F}_n(\hat{\boldsymbol{\theta}}_n, \boldsymbol{X}_n) + \boldsymbol{F}_{n_k}(\hat{\boldsymbol{\theta}}_n, \boldsymbol{X})]$.

Step 2 Collect n_k output measurements based on the n_k optimally chosen inputs from step 1. Use the n_k measurements to update the value of $\boldsymbol{\theta}$ from $\hat{\boldsymbol{\theta}}_n$ to $\hat{\boldsymbol{\theta}}_{n+n_k}$.

Step 3 Stop if the value of $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}_{n+n_k}$ is satisfactory or if the budget of measurements has been expended. Else return to step 1 with the new k set to the former $k + 1$ and the new n set to the former $n + n_k$ (so the updated $\boldsymbol{X}_n$ now includes the inputs $\boldsymbol{X}_{n_k}$ from step 1).

Let us offer a few comments about the steps above. From step 1, the optimization to find the new input points $\boldsymbol{X}_{n_k}$ takes account of the previously determined inputs ($\boldsymbol{X}_n$). If certain areas of the input space are well represented in the previous data, it is more likely that new inputs will be placed in "underrepresented" areas of the input space. Hence, the solution in step 1 is generally going to differ from a corresponding problem where one simply maximizes $\det[\boldsymbol{F}_{n_k}(\hat{\boldsymbol{\theta}}_n, \boldsymbol{X})]$ with respect to $\boldsymbol{X}$.

One of the distinctions in implementation is between a batch-sequential and a full-sequential design. In the batch-sequential implementation, $n_k > 1$ data points are collected at each iteration. The full-sequential implementation relies on $n_k = 1$ for all k.

There are two potentially difficult optimization problems embedded in the sequential design iteration process. One is in step 1, where the new design points $\boldsymbol{X}_{n_k}$ are to be determined; the other is in step 2, where the updated value of $\boldsymbol{\theta}$ is calculated. There are no universal methods for carrying out either of these optimization tasks.

A number of techniques have been examined for the optimization in step 1. Methods used here can be the same as methods used in local design because one is working with a fixed $\boldsymbol{\theta}$. In some fortunate cases, relatively simple analytical or numerical schemes may be used, such as in Examples 17.8 and 17.9. Atkinson and Donev (1992, Chap. 18) discuss an approach where the model $h(\boldsymbol{\theta}, \boldsymbol{x})$ is linearized about the current $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}_n$. Then, the methods of Section 17.2 for linear models can be used to determine an approximate design at each iteration. A demonstration of simulated annealing for nonlinear local design is given in Rustagi (1994, p. 294).

The optimization problem associated with estimating $\boldsymbol{\theta}$ (step 2) may be carried out in batch or recursive form. In the batch form, the estimate is simply computed "from scratch" at each iteration based on the cumulative $n + n_k$ data points that are available. Little or no use is made of the previous $\boldsymbol{\theta}$ estimate ($\hat{\boldsymbol{\theta}}_n$). This may be acceptable in simple problems. More commonly, the process of estimating $\boldsymbol{\theta}$ in nonlinear problems entails some difficult numerical optimization where one wants to exploit the knowledge of $\boldsymbol{\theta}$ that has been acquired through the previous experiments. In such cases, recursive methods are preferred, where the estimate $\hat{\boldsymbol{\theta}}_{n+n_k}$ is directly updated from the previous estimate $\hat{\boldsymbol{\theta}}_n$. Such recursive estimation is naturally connected to SA with a stochastic gradient input (Chapter 5), as we now discuss.

Consider recursive estimation in the context of the full-sequential version of the design strategy. Here $n_k = 1$ for all k. Hence, the process of choosing $\boldsymbol{X} = \boldsymbol{X}_{n_k}$ in step 1 above reduces to the problem of choosing the single input $\boldsymbol{x} = \boldsymbol{x}_{n+1}$ in the model $h(\hat{\boldsymbol{\theta}}_n, \boldsymbol{x})$. Let $\boldsymbol{Y}_n(\boldsymbol{\theta}|z_{n+1}, \boldsymbol{x}_{n+1})$ represent the instantaneous score vector when updating $\hat{\boldsymbol{\theta}}_n$ to $\hat{\boldsymbol{\theta}}_{n+1}$ in step 2. This corresponds to the gradient with respect to $\boldsymbol{\theta}$ of the part of the estimation criterion that depends on the current

input. In a maximum likelihood setting, this typically involves the negative gradient of the logarithm of the probability density function for the $(n+1)$st measurement ("negative" so that the underlying root-finding problem corresponds to a minimization problem). In a least-squares setting, this will involve the gradient of the current squared error (i.e., $\boldsymbol{Y}_n(\boldsymbol{\theta}|z_{n+1}, \boldsymbol{x}_{n+1}) = \frac{1}{2}\partial[z_{n+1} - h(\boldsymbol{\theta}, \boldsymbol{x}_{n+1})]^2/\partial\boldsymbol{\theta}$).

The standard (unconstrained) SA algorithm in Chapters 4 and 5 in this design context is

$$\hat{\boldsymbol{\theta}}_{n+1} = \hat{\boldsymbol{\theta}}_n - a_n \boldsymbol{Y}_n(\hat{\boldsymbol{\theta}}_n \,|\, z_{n+1}, \boldsymbol{x}_{n+1}), \tag{17.20}$$

where a_n is a decaying, nonnegative gain sequence satisfying the usual conditions (Sections 4.3 and 4.4). Recall that Section 4.3 presents conditions under which the SA algorithm converges to the optimum $\boldsymbol{\theta}^*$ as the amount of data increases. (Because of the emphasis of this chapter on input design, the notation for $\boldsymbol{Y}_n$ above explicitly includes the dependence on z_{n+1} and $\boldsymbol{x}_{n+1}$. This contrasts with the notation in the SA algorithms of Chapters 4 and 5, where any such dependence is suppressed within the notation $\boldsymbol{Y}_n(\hat{\boldsymbol{\theta}}_n)$. The reader should be aware that the fundamental algorithm is the same, despite the notational difference.) The SA form in (17.20) is convenient in that the new estimate for $\boldsymbol{\theta}$ is directly derived from the previous estimate with an adjustment that tends to get smaller as more data are collected (due to $a_n \to 0$). Ford et al. (1989) discuss the use of a second-order SA algorithm (as in Subsection 4.5.2) in sequential design, where a_n is replaced by the scaled inverse of the Hessian matrix of the performance measure.

An important cautionary note on statistical inference is in order. Because each input $\boldsymbol{x}_{n+1}$ depends on prior data $z_1, \boldsymbol{x}_1, z_2, \boldsymbol{x}_2, \ldots, z_n, \boldsymbol{x}_n$ through step 1 above, the data are not independent. This contrasts with the classical assumptions associated with a fixed design, where the $\boldsymbol{x}_i$ are predetermined and the z_i are independent. Most of the theory associated with using the asymptotic distribution and Fisher information matrix for inference (Section 13.3) assumes independent data. Recall that under regularity conditions, a scaled estimate $\hat{\boldsymbol{\theta}}_n$ is asymptotically normally distributed with covariance matrix given by the inverse of the information matrix (Subsection 13.3.3). This result is very useful in statistical inference. However, because the data are not independent in sequential design, one should be cautious about a straightforward application of the cumulative information matrix $\boldsymbol{F}_n$ resulting from the steps above. Nonetheless, theoretical and numerical support exists for treating this matrix in the same way as the traditional information matrix based on independent data (e.g., Ford and Silvey, 1980; Chaudhuri and Mykland, 1995). Further, if the SA recursion (17.20), or its second-order form, is used to update the parameter estimates, theory exists under which the covariance matrix in the asymptotic distribution is

the inverse information matrix (see Subsection 13.3.3). That is, the SA theory directly accommodates the dependence on the inputs and the measurements.

Let us now present an example of sequential design using (17.20). This example builds on the dilution problem of Examples 17.9 and 17.10.

Example 17.11—Sequential design and parameter estimation for dilution problem. Consider the sequential design approach in the context of the scalar problem of Examples 17.9 and 17.10. Suppose that a sequential maximum likelihood method is being used to estimate the mean θ. Hence, each input in the SA recursion (17.20) is the negative derivative of the most recent summand in the log-likelihood function (17.19). Taking the negative of the derivative of the last summand in $\log \ell(\theta | \boldsymbol{Z}_n)$ (see (17.19)), the input $\boldsymbol{Y}_n(\theta | z_{n+1}, x_{n+1})$ for use in (17.20) is

$$\boldsymbol{Y}_n(\theta | z_{n+1}, x_{n+1}) = \frac{1 - z_{n+1}}{x_{n+1}} - \frac{z_{n+1} \exp(-\theta / x_{n+1})}{x_{n+1}[1 - \exp(-\theta / x_{n+1})]}.$$

Consider a setting where $\theta^* = 4$ is used to generate the data (note that θ^* is not generally the same as the batch maximum likelihood estimate that would result from maximizing $\log \ell(\theta | \boldsymbol{Z}_{n+1})$ for a finite n). Let us use the SA recursion (17.20) with gain value a_n in the standard form, $a_n = a/(n+1+A)^{\alpha}$, $n \geq 0$, a and A nonnegative, and $1/2 < \alpha \leq 1$, as given in Section 4.4. Let the initial condition be $\hat{\theta}_0 = 2$. Based on some preliminary numerical experimentation and the theoretical principles of Sections 4.3 and 4.4, we chose $a = 25$, $A = 50$, and $\alpha = 1$. The local design for this model has only one support point $x = \chi_1 = 0.63\theta$ (Example 17.10). Hence, the optimal input in step 1 of the sequential design process is $x_{n+1} = 0.63\hat{\theta}_n$.

Table 17.2 shows the normalized estimation error, $|\hat{\theta}_n - \theta^*| / |\hat{\theta}_0 - \theta^*|$, based on the terminal estimate $\hat{\theta}_n$ for each of three sample sizes (100, 1000, or 10,000). Each table value represents the sample mean of 60 normalized errors resulting from 60 independent replications. The θ estimate throughout the iteration process is constrained to be in $[1.59, \infty)$ as discussed in Example 17.10. The table compares three design strategies. The sequential design method is as described in the steps above with $n_k = 1$ (corresponding to a full sequential design). The local design based on $\theta = \hat{\theta}_0$ (so $x_n = 0.63\hat{\theta}_0$ for all n) is meant to depict the results one would see using a fixed design based on a best guess at θ (corresponding to the initial guess at θ). The other local design based on $\theta = \theta^*$ (i.e., $x_n = 0.63\theta^*$) depicts an idealized case based on the true (unknown in practice) θ. Only the sequential design and the suboptimal local design based on $\theta = \hat{\theta}_0$ would be implementable in practice.

Table 17.2 shows that the sequential design and the idealized local design based on $\theta = \theta^*$ perform comparably. On the other hand, the local design based on $\theta = \hat{\theta}_0$ performs noticeably worse. The table illustrates the value in using a

Table 17.2. Normalized estimation errors for sequential design and for two strategies with local designs. Indicated sample mean is based on 60 realizations.

	Sample mean for $\lvert\hat{\theta}_n - \theta^*\rvert / \lvert\hat{\theta}_0 - \theta^*\rvert$		
Sample size per realization	Sequential design	Local design $\theta = \theta^*$	Local design $\theta = \hat{\theta}_0$
100	0.25	0.23	0.31
1000	0.065	0.063	0.096
10,000	0.021	0.022	0.032

near-optimal design via the sequential strategy. With the sample sizes of 1000 and 10,000, the sequential design produces mean parameter estimation errors that are about 2/3 the size of the errors under the $\hat{\theta}_0$-based local design. This equivalently leads to a significant reduction in the sample size required to obtain a given level of accuracy. In particular, the sequential strategy requires about *half* the measurements that are needed in the suboptimal ($\hat{\theta}_0$-based) local design to achieve the same level of accuracy (Exercises 17.15 and 17.16). ❑

Bayesian Design

In the Bayesian design approach, one copes with the uncertainty in $\boldsymbol{\theta}$ by assuming a prior distribution for $\boldsymbol{\theta}$. This prior distribution is used in averaging the design criterion over reasonable possible values. The prior is a reflection of one's knowledge of likely values for $\boldsymbol{\theta}$. The prior may be specified on a purely subjective basis or on the basis of information from previous estimation of $\boldsymbol{\theta}$ (e.g., using the asymptotic distribution for a maximum likelihood estimate, as discussed in Section 13.3, based on data collected previously). This approach does not require any other aspect of the analysis to be carried out in a Bayesian manner. That is, no posterior distribution is involved in obtaining the design, and the prior distribution need only be used in determining the design, not for the subsequent analysis. (Section 16.6 summarized some generic aspects of the Bayesian approach for parameter estimation.) Hence, one can implement a Bayesian design approach, as described here, without being a "card-carrying" Bayesian in the other aspects of the estimation and inference.

As with the non-Bayesian approach, there are competing lettered criteria (*A*-optimality, *D*-optimality, etc.). We will continue to focus on *D*-optimality. Unlike the non-Bayesian case, however, there is not a unique *D*-optimal criterion. In fact, *five* *D*-optimal criteria are given in Atkinson and Donev (1992, p. 214), and even more such criteria appear in other literature (e.g., Walter and Pronzato, 1990). Among other variations, the differences follow from whether

the averaging for $\boldsymbol{\theta}$ appears inside or outside certain function evaluations. One D-optimal criterion that is advocated in Chaloner and Verdinelli (1995) is

$$E_{\boldsymbol{\theta}}\{\log\det[\boldsymbol{M}(\boldsymbol{\theta}, \xi)]\} = \int_{\Theta} \log\det[\boldsymbol{M}(\boldsymbol{\theta}, \xi)]\, p(\boldsymbol{\theta})\, d\boldsymbol{\theta}, \qquad (17.21)$$

where $E_{\boldsymbol{\theta}}$ denotes expectation with respect to $\boldsymbol{\theta}$, $p(\boldsymbol{\theta})$ is the prior expressed as a density function, and Θ is, as in earlier chapters, the domain for $\boldsymbol{\theta}$. This criterion uses the log transformation as discussed at the end of Section 17.2. To illustrate the variations possible in D-optimality, an alternative criterion to the one in (17.21) is $\log\det\{E_{\boldsymbol{\theta}}[\boldsymbol{M}(\boldsymbol{\theta}, \xi)]\}$. Another alternative is considered in Exercise 17.17. One of the desirable properties of the criterion in (17.21) is that it is an approximation to a generally infeasible criterion that when optimized produces an experiment that maximizes the increase in Shannon information (we will not delve further into the deep subject of Shannon information).

As one might expect, finding the design that maximizes (17.21) (or other criteria) is usually a significant challenge. Among other issues, evaluating the integral in (17.21) may be numerically cumbersome (or worse!) at each candidate design ξ. Deterministic and stochastic methods (such as genetic algorithms) are discussed in Chaloner and Verdinelli (1995), Müller and Parmigiani (1995), and Hamada et al. (2001). A practical method for avoiding the numerical integral in (17.21) is to choose a prior distribution that is discrete with only a small number of support points (the values of $\boldsymbol{\theta}$ having nonzero probability).

One of the major issues in determining a design ξ is the choice of the number of support points N (not to be confused with the support points for a discrete prior distribution as mentioned in the preceding paragraph). A convenient bound on N was presented in Subsection 17.2.2 for *linear* models. In contrast, for nonlinear models, there is no such convenient bound. Chaloner and Verdinelli (1995) discuss this issue with pointers to related literature. They note that the number of support points grows with the dispersion of the prior distribution. They also note that for tight priors (little dispersion in $\boldsymbol{\theta}$), an optimal design often exists with $N = p$, identical to a result for locally optimal solutions based on a fixed $\boldsymbol{\theta}$ (approach 1 at the beginning of this subsection).

The fact that N must be greater than p in many practical problems is sometimes beneficial. The points beyond p provide data for checking the model with other candidate model forms having more than p parameters (as discussed in Section 13.2). This can be done to ensure that the chosen model is, in fact, the "best" representation of the process. Such model checking is useful to address the general criticism of optimal design cited in Section 17.1 that the design is directly dependent on the chosen model form. The extra support points for the design help an analyst check the validity of alternative models.

Let us now present a simple example illustrating the process for finding a D-optimal design based on a prior distribution for a scalar θ.

Example 17.12—Optimal design with discrete prior for θ. Consider a nonlinear model of the form $z = e^{-\theta x} + v$, where $v \sim N(0, 1)$ and $\theta > 0$ and $x \geq 0$ are scalars. Suppose that for $k = 1, 2, \ldots, n$, the noises v_k are independent, identically distributed (i.i.d.), implying that the measurements z_k are independent. The information number for n measurements is

$$F_n(\theta, \boldsymbol{X}_n) = \sum_{k=1}^{n} x_k^2 \exp(-2\theta x_k) .$$

As in Example 17.10 on local design, it is sufficient to work with only one summand in $F_n(\theta, \boldsymbol{X}_n)$ because each of the summands has an identical form. Hence, based on criterion (17.21), the aim is to find a design that maximizes

$$E_\theta[\log(x^2 e^{-2\theta x})] = E_\theta(2\log x - 2\theta x) .$$

Suppose that the prior on θ is a Bernoulli (binary) distribution with probability ρ placed on value θ_1 and probability $1 - \rho$ placed on θ_2. Then, from the expression above, the aim is to find x that maximizes

$$\rho(\log x - \theta_1 x) + (1-\rho)(\log x - \theta_2 x) \tag{17.22}$$

subject to the constraints on θ and x. For convenience, assume that $\theta_2 > \theta_1$. As shown in Haines (1995), standard calculus-based methods for optimization yield a unique (single support point) design when $\theta_2/\theta_1 \leq 2 + \sqrt{3}$. In particular, differentiating (17.22) with respect to x, setting the derivative to 0, and checking the second-order conditions to ensure that the solution is a maximum yields

$$\xi_n^* = \begin{Bmatrix} [\rho\theta_1 + (1-\rho)\theta_2]^{-1} \\ 1.0 \end{Bmatrix} .$$

When $\theta_2/\theta_1 > 2 + \sqrt{3}$, more complicated arguments are needed. The difficulties follow from the need to optimize along the boundary of the constraint set. In such a setting, the solution is either identical to the single-point result above or is a two-point solution of the generic form

$$\xi_n^* = \begin{Bmatrix} \chi_1^* & \chi_2^* \\ w_1^* & w_2^* \end{Bmatrix} ,$$

where χ_i^* and w_i^* are solutions to equations related to the curvature of the constraint set (see Haines, 1995). Whether the design has one or two support

points when $\theta_2/\theta_1 > 2 + \sqrt{3}$ depends on whether a particular inequality is satisfied (see Haines, 1995, expression (3.7)).

Note that these results are consistent with the above discussion relating the number of support points in the design to the spread of the prior distribution. When the candidate θ values are close in the sense that $1 < \theta_2/\theta_1 \leq 2 + \sqrt{3}$, there is only one support point (corresponding to $p = 1$); when the θ values have greater spread in the sense that $\theta_2/\theta_1 > 2 + \sqrt{3}$, the number of support points may increase to two. ❑

17.5 CONCLUDING REMARKS

There are often many goals for an experiment. Some may conflict with each other. As a consequence, there is no "cut and dried" strategy to the problem of determining a good choice for the user-specified inputs. As discussed in this chapter, optimal design is a powerful general strategy. Applications arise in both static regression-type models and in dynamic models that are used, for example, in control systems. Another popular general strategy is classical experimental design (e.g., fractional factorial, Latin squares, etc.), which forms the basis for many popular books in the field. Classical design is based strongly on geometric and intuitive considerations.

The solution to a specific problem can sometimes be shown to be both an optimal design (say, a *D*-optimal design) and a classical design. Unfortunately, in the literature, the dichotomy between general optimal design and classical design often illustrates the adage "never the twain shall meet." For instance, there is only a brief discussion of optimal design in significant books on experimental design such as Wu and Hamada (2000) and Montgomery (2001). In fact, with its letter-based labeling of the design criteria (*A*-optimal, *D*-optimal, etc.), optimal design is sometimes derisively referred to as the "alphabet soup" approach.

Optimal design is often criticized on the basis of its close connection to the assumed model form. That is, a criterion such as the *D*-optimal measure emphasized here can only be computed with an exact specification of the model. (Classical designs also depend on model assumptions, although to a lesser extent than optimal designs. For example, prior knowledge of interactions that are likely to be negligible is essential in creating a practical factorial design by "confounding" [Montgomery, 2001, Chap. 7].) A criticism related to the model dependence is that it is difficult to capture the many goals of an experiment—as discussed in Section 17.1—in one mathematical criterion, as needed in optimal design. In practice, supplemental runs may be carried out to evaluate the model validity or address other goals for the experiment.

On the other hand, the relatively formal structure of optimal design provides a means of coping with the many nonstandard situations that arise in practical applications. The geometric and intuitive basis for classical design frequently breaks down when departing from the standard linear (including

curvilinear) model over "nice" input domains. As noted in Cook and Nachtsheim (1989, p. 345), "...the situations in which classical designs work well are rather confining, occasionally leading to unfortunate results....Experimenters often have an overwhelming urge to tailor the scientific question or trim the experimental material to allow application of a particular classical design."

One important application of optimal design is in determining inputs in models that are nonlinear in the parameters $\boldsymbol{\theta}$ being estimated. This was demonstrated in Section 17.4. Classical design provides little or no guidance in the selection of inputs for nonlinear models. Further, there is an inherent difficulty in nonlinear problems because of the dependence of the criterion on the parameters being estimated. One is picking a design in order to estimate $\boldsymbol{\theta}$, but one has to know $\boldsymbol{\theta}$ to pick a design!

There are several approaches to dealing with the dependence of the criterion on $\boldsymbol{\theta}$ in nonlinear models, foremost among them sequential and Bayesian strategies. Another challenge in nonlinear design is the difficulty of calculating the criterion and carrying out the related optimization process. The information matrix forming the basis for the leading D-optimal criterion is not readily available in most practical nonlinear problems. (The Monte Carlo resampling method of Subsection 13.3.5 provides a means for determining the information matrix in difficult problems, but this is likely to be computationally burdensome in the repeated evaluations required during an optimization process.)

Finally (!), we come to the end of this book on stochastic search and optimization. Those who have read extensive parts of the book have been exposed to many powerful methods for use in estimation, Monte Carlo simulation, and control. It is also clear that there are many exciting areas—in both algorithm development and applications—that have yet to be fully explored. In using the methods here (or anywhere!), one should keep in mind the aphorism mentioned at the end of Chapter 1: "Better a rough answer to the right question than an exact answer to the wrong one." Whether rough or exact, the methods here include many tools useful in addressing many "right questions."

17.6 APPENDIX: OPTIMAL DESIGN IN DYNAMIC MODELS

The main body of this chapter explores optimal design in static input–output systems. There is also a large literature in dynamic systems, including control systems. See, for example, Mehra (1974), Goodwin and Payne (1977), Walter and Pronzato (1990), or Ljung (1999, Chap. 13). Of main interest here is the choice of inputs to enhance the estimation of the parameters of the dynamic model. For control systems, the inputs may be provided in closed-loop mode. One distinction with static models is that even for linear models, the D-optimal criterion (and resulting optimal inputs) may depend on the parameters $\boldsymbol{\theta}$ being estimated (see, e.g., Mehra, 1974, Example VIII.C).

The example sketched below provides a flavor of optimal designs for such systems. This example is extracted from Levadi (1966) and Mehra (1974).

Example 17.13—Optimal input for first-order model. Consider a scalar process $q(\tau)$ that is modeled by the first-order differential equation

$$\frac{dq(\tau)}{d\tau} = -q(\tau) + \theta x(\tau),$$

where τ is the time variable, $x(\tau)$ is a user-specified input (a time-varying analogue of the static $\boldsymbol{x}$ input in the main part of this chapter), and θ is a scale parameter to be estimated. The analyst collects data on a continuous basis over time $\tau \in [0, T]$ according to

$$z(\tau) = q(\tau) + v(\tau),$$

where $v(\tau)$ is a noise term. The aim is to determine $x(\tau)$ so that θ is estimated as accurately as possible. It is assumed that the input is bounded in the sense that $\int_0^T x(\tau)^2 d\tau = 1$. Suppose that the noise has mean zero and covariance function $E[v(\tau_1)v(\tau_2)] = c_0 \exp(-c_1|\tau_1 - \tau_2|)$, where c_0 and c_1 are two constants and τ_1 and τ_2 are two time instants. Hence, the correlation between noises at two different times decreases exponentially as the difference in the times gets larger.

The scalar precision quantity M here for the D-optimal solution is a dynamic analogue of the inverse variance quantity used with linear models, as described in Section 17.2. The inverse quantity $1/M$ reflects the variance in the estimate for θ based on data $z(\tau)$ over the interval $[0, T]$; the estimate for θ is constructed via a dynamic model analogue of the least-squares solution in Section 17.2.

As shown in Levadi (1966) and Mehra (1974), the time-varying D-optimal input $x^*(\tau)$ is

$$x^*(\tau) = c \sin(\omega\tau + \kappa),$$

where c and κ are constants dependent on T and the frequency ω. The value for ω depends on the degree of correlation in the noise as expressed through the magnitude of c_1 (the "bandwidth" of the noise). Unlike some solutions for linear dynamic systems, the optimal input $x^*(\tau)$ above does not depend on θ. ❑

EXERCISES

17.1 As a reminder of the limitations of the D-optimal criterion, produce two positive definite 3×3 matrices $\boldsymbol{M}^*$ and $\boldsymbol{M}$ such that $\det(\boldsymbol{M}^*) > \det(\boldsymbol{M})$, but $\boldsymbol{M}^* \not\succ \boldsymbol{M}$.