

Statistical Methods as Optimization Problems

Optimization problems — maximization or minimization — arise in many areas of statistics. Statistical estimation and modeling both are usually special types of optimization problems. In a common method of statistical estimation, we *maximize* a likelihood, which is a function proportional to a probability density at the point of the observed data. In another method of estimation and in standard modeling techniques, we *minimize* a norm of the residuals. The best fit of a model is often defined in terms of a minimum of a norm, such as least squares. Other uses of optimization in statistical applications occur prior to collection of data, for example, when we design an experiment or a survey so as to minimize experimental or sampling errors.

When a statistical method is based on the solution of an optimization problem, to formulate that problem unambiguously helps us both to understand the method and to decide whether the method is appropriate to the purposes for which it is applied.

In this chapter we survey several statistical methods that are based on optimization and identify explicitly the optimization problem to be solved in applying the method. The reader may or may not be familiar with these statistical methods; that is not important at this time. After expressing a variety of statistical methods as solutions to optimization problems, we summarize some of the general properties of these optimization problems.

Some of the simpler and more common optimization problems in statistics can be solved easily, often by solving a system of linear equations. Many other problems, however, do not have closed-form solutions, and the solutions must be approximated by iterative methods. We discuss these methods in later chapters.

This chapter also provides a gentle introduction to notation and terminology that is used more-or-less consistently throughout the book.

Statistical Models

A common type of model relates one variable to others in a form such as

$$y \approx f(x), \quad (1.1)$$

in which y and x are observable variables, and f is some rule that gives an approximate relationship. The approximation can be expressed in terms of a probability, an expected value, a likelihood, or a random variable that modifies the value of $f(x)$. (Notice that I do not use special notation to distinguish vectors from scalars; in this model it is likely that y is a scalar and x is a vector.) In this type of model, we sometimes call y the *response variable* and x the *regressors* or the *independent variables* or the *explanatory variables*, although we do not mean “independent” in the sense that the word is used in probability theory.

A general form of a statistical model is

$$y = f(x) + \epsilon, \quad (1.2)$$

in which y and x are observable variables, f is some function, and ϵ is an unobservable residual or error.

The model usually specifies that ϵ is a random variable with a given distribution or family of distributions. We sometimes emphasize that the residual, or error term, is a random variable by use of an upper-case letter.

In a common and useful formulation of this model, we assume f is some given function or a function within a given class of functions, and introduce a parameter that determines the specific function. We then write the model as

$$y = f(x; \theta) + \epsilon, \quad (1.3)$$

in which θ is a vector of *parameters* with unknown and unobservable values, and y , x , and ϵ are as before.

Fitting Models and Statistical Estimation

A basic problem in data analysis is to fit this model using observed data. *Fitting* the model is mechanically equivalent to *estimating* the parameter θ , which we often refer to as the *estimand*. The more formal methods of statistical inference involve estimation of θ as a preliminary step. First, we decide on a method of estimation, and then after estimating θ , we describe properties of the estimator and make inferences about y and its relationship with x .

Linear Statistical Models

The most familiar form of the model in equation (1.3) is the linear model

$$y = x^T \beta + \epsilon, \quad (1.4)$$

where x and β are vectors. We also often assume that ϵ is a random variable with a normal distribution.

In this model β is a fixed, but unknown quantity. The problem of fitting the model is to estimate β . Estimation or fitting of the model requires a set of observations, y_i and x_i . We often write the model for the set of observations as

$$y = X\beta + \epsilon, \quad (1.5)$$

where X is the matrix whose rows are formed by the x_i , and y and ϵ are vectors whose elements are the y_i and ϵ_i . (Recall the statement above that I do not use special notation to distinguish vectors from scalars; the meaning is clear from the context. I usually use upper-case letters to denote matrices.)

Replacement of the Unknown Parameter by a Variable

The first step in fitting the model is to replace the unknown parameter with a variable; that is, we use the expression $f(x_i; t)$ with t in place of θ , or $x_i^T b$ with b in place of β . Often, this substitution is not made explicitly. I think it helps to emphasize that our only information about θ or β is through their role in the data-generating process, and after the data are in hand, we seek a value to be substituted for θ or β in the model.

For each pair of observations, we form a residual that is a function of the variable that has been substituted for the unknown parameter, and of the observed data. We then form some summary function of the residuals, $R(t; x, y)$. This is the function to minimize by proper choice of t . We denote the problem of minimizing this function by

$$\min_t R(t; x, y). \quad (1.6)$$

The expression $\min_t R(t; x, y)$ also denotes the minimum value of $R(t; x, y)$. The value of t , say t_* , for which $R(t; x, y)$ attains the minimum is denoted by

$$t_* = \arg \min_t R(t; x, y). \quad (1.7)$$

The value t_* may or may not be unique. This raises the issue of identifiability or of estimability. If t_* is not unique, we determine invariant quantities (such as the value of $R(t_*; x, y)$ itself) and focus our attention on those quantities. We will not pursue this topic further here.

Smoothed Responses and Fitted Residuals

Once the model has been fit, that is, once a variable has been substituted for the unknown parameter and an optimal t_* has been determined, for any given value of x , we have a predicted or fitted value of y . For $x = x_0$, from equation (1.3), we have a corresponding value of y , say $\hat{y}_0 = f(x_0; t_*)$. (Here, we assume t_* is unique.) In general, the fitted value of y is a function of t as well as of x .

models—)

For the values of x in the data that were used to fit the model, x_1, \dots, x_n , we have corresponding fitted or “smoothed” values of y , $\hat{y}_1, \dots, \hat{y}_n$.

For a given pair (y, x) and a given value of t , the difference $y_i - f(x_i; t)$ is of interest. This residual is a function of t , and to emphasize that, we often denote it as $r_i(t)$. For the linear model (1.4), the residual as a function of b is

$$r_i(b) = y_i - x_i^T b, \quad (1.8)$$

where the variable b is in place of the unknown estimand β .

For the chosen value of t , that is, for t_* , and for each pair of observed values (y_i, x_i) , we call the difference the *fitted residual*:

$$\begin{aligned} r_i(t_*) &= y_i - f(x_i; t_*) \\ &= y_i - \hat{y}_i. \end{aligned} \quad (1.9)$$

We often drop the functional notation, and denote the fitted residual as just r_i .

We should note the distinction between the fitted residuals and the unobservable residuals or errors in the model. In the case of the linear model, we have the fitted residuals, r_i , from equation (1.8) with the fitted value of b and the unobservable residuals or errors, $\epsilon_i = y_i - x_i^T \beta$, from the model equation (1.4). When ϵ is a random variable, ϵ_i is a realization of the random variable, but the fitted residual r_i is *not* a realization of ϵ (as beginning statistics students sometimes think of it).

In fitting the model, we choose a value of the variable which when used in place of the parameter, minimizes the fitted residuals.

We note that the residuals, either fitted or those from the “true” model, are vertical distances, as illustrated in Figure 1.1 for a simple linear model of the form

$$y_i = b_0 + b_1 x_i + r_i \quad (1.10)$$

on the left and for a model of the form

$$y_i = t_1(1 - \exp(t_2 x_i)) + r_i \quad (1.11)$$

on the right.

Choosing a Model

The two fitted models in the example in Figure 1.1 are very different. The pattern of the residuals in the linear model, one negative residual followed by four positive residuals and a final negative residual, suggests that there may actually be some curvature in the relationship between y and x . On the other hand, the shape of the exponential model on the left side of the domain appears highly questionable, given the observed data.

The problem of choosing an appropriate model is a much more complicated one than the problem of fitting a given model. Choice of the form of a model can also be formulated as an optimization problem. We consider this problem again briefly on pages 12 through 14.

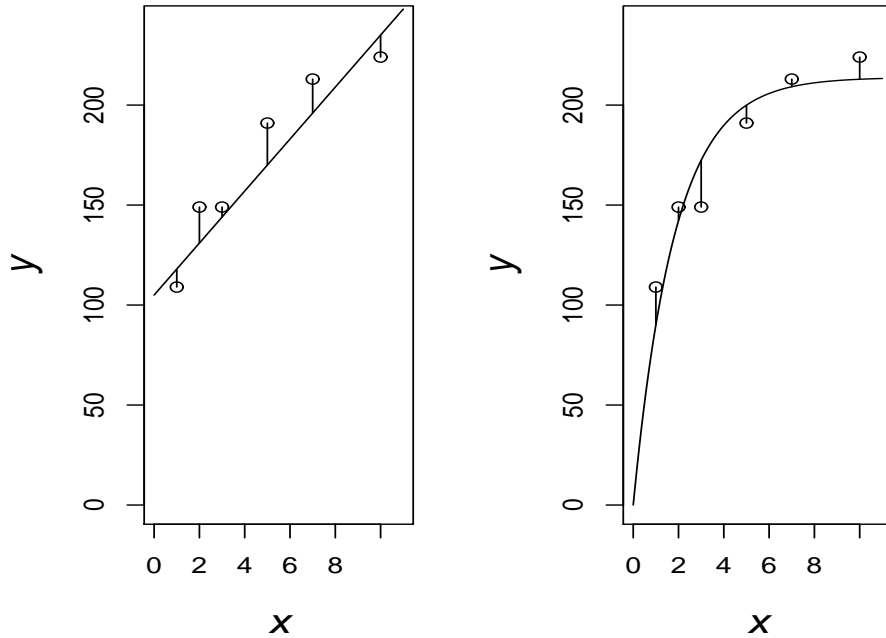


Fig. 1.1. Residuals in a Linear and a Nonlinear Model gro0105

Minimizing Residuals

The idea of choosing a model so as to minimize the residuals from the observed data is intuitively appealing. Because there is a residual at each observation, however, “minimizing the residuals” is not well-defined without additional statements. When there are several things to be minimized, we must decide on some way of combining them into a single measure. It is then the single measure that we seek to minimize, as in the expression (1.6).

Next we note the obvious: some residuals are positive and some are negative, so our objective cannot be to minimize them directly, but rather to minimize some function that increases as the residuals increase positively or decrease negatively. If positive and negative values of the residuals are to be treated the same, as usually makes sense, then we can minimize some function of the absolute values of the residuals.

Because all of the residuals cannot be minimized simultaneously, the next step is to introduce an overall measure of the set of residuals. A useful type of overall measure is a *norm* of the vector of residuals. (We discuss vector norms in more detail in Section 2.1.2, on page 39, but we will consider a couple of

least squares

examples here.) The most obvious measure, perhaps, may just be the sum of the absolute values. For a linear model of the form of equation (1.4) this is

$$R_1(b) = \sum_{i=1}^n |y_i - x_i^T b|. \quad (1.12)$$

This quantity is called the L_1 norm of the vector of residuals $r(b)$, and is denoted as $\|r(b)\|_1$. Another possible measure is the sum of the squares:

$$R_2(b) = \sum_{i=1}^n |y_i - x_i^T b|^2. \quad (1.13)$$

This quantity is the square of what is called the L_2 norm of the vector of residuals $r(b)$. We denote the square root of this nonnegative quantity as $\|r(b)\|_2$.

When there is only a single quantity to minimize, minimizing any increasing function of that quantity, such as its square if the quantity is nonnegative, is equivalent to minimizing the quantity itself. We would arrive at the same point (that is, the same value of the variable over which the minimization is performed) if we minimized some other increasing function of that quantity. If, however, we are to minimize a sum of several quantities as in the problem of fitting a model by minimizing the residuals, applying a given increasing function to each quantity prior to summing may result in a different point of minimization than if we apply some other increasing function.

Least Squares of Residuals

For various reasons, the most common approach to fit the model with the given data is *least squares*; that is, to use $R_2(b)$ as the overall measure of the residuals to minimize. With n observations, the ordinary least squares estimator of β in the model (1.4) is the solution to the *optimization problem*

$$\min_b R_2(b). \quad (1.14)$$

This optimization problem is relatively simple, and its solution can be expressed in a closed form as a system of linear equations.

The solution to the optimization problem is a linear combination of the y_i (we will see this fact later), and under flexible assumptions about the probability distribution of the random error term, some simple statistical properties of the estimator are relatively easy to determine. Furthermore, from statistical theory, it is easy to see that this estimator is optimal in a certain sense among a broad class of estimators. If the distribution of the random error term is normal, even more statistical properties of the estimator can be determined.

When the original model (1.3) is nonlinear, in the same way we form the residuals, $y_i - f(x_i; t)$, and the nonlinear least squares estimate for θ is the solution to the *optimization problem*

$$\min_t \sum_{i=1}^n (y_i - f(x_i; t))^2, \quad (1.15)$$

where we are using the vector t as a variable in place of the fixed but unknown θ . This least squares problem is much more difficult both computationally and conceptually than the linear least squares problem. In general, there is no closed-form solution to this optimization problem.

The Residual Squared Norm as a Function of Parameter Values

The L_2 norm of the residuals is a function of the parameters; that is, of the variables that we have substituted in place of the parameters. The objective, of course, is to find values of these variables that yield the minimum of the residual norm. It is instructive to study the residual norm, or even better, the residual squared norm, as a function of those values. For the simple linear model in equation (1.10) with the data shown in Figure 1.1, Figure 1.2 shows contours of the residual squared norm, $R_2(b_0, b_1)$ over a region of parameter values that seems to include the minimum of the function.

The contours are quite regular, and are typical of a quadratic function. As with any quadratic, the surface of the residual squared norm function appears to be fairly flat at the minimum.

Figure 1.3 shows contours of the residual squared norm of the nonlinear model in equation (1.11) with the data shown in Figure 1.1.

The plot in Figure 1.3 is quite different from that in Figure 1.2. Although the squared norm in Figure 1.3 is also a sum of squares, the surface is not a quadratic in the parameter values. Together these plots illustrate the statements made above that the linear least squares problem is rather simple, but the nonlinear problem can be much more complicated.

Constrained Least Squares

Sometimes we may know that the parameter in the model must satisfy certain restrictions, and we may choose to impose those restrictions in fitting the model; that is, we may require that our estimate of the parameter satisfy the restrictions. (At first glance, it may seem that is obvious that we should do this, but it is not clearly the case. In applying the restrictions to the estimate, we may lose some desirable property the unrestricted estimator possesses, such as unbiasedness.) For example, in the linear model (1.4) we may know $\beta \geq 0$, and so we modify the optimization problem to impose constraints. For the case of $\beta \geq 0$, instead of the optimization problem (1.14), we formulate the nonnegative least squares *optimization problem*

$$\min_{b \geq 0} \sum_{i=1}^n (y_i - x_i^T b)^2. \quad (1.16)$$

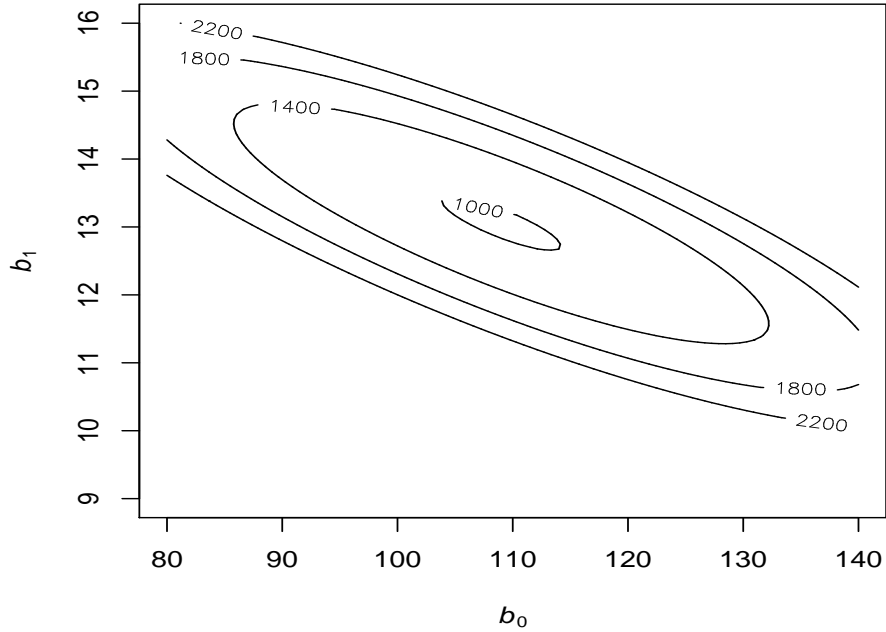


Fig. 1.2. Contours of the Sum of Squares Function for a Linear Model gro0110a

This optimization problem is considerably more complicated than the unconstrained problem. Its solution cannot be expressed in a closed form.

Often the best approach is to solve the unconstrained problem first. If it happens that the unconstrained solution satisfies the constraints, we have the solution. This would just happen to be the case for the problem shown in Figure 1.2 with nonnegativity constraints added. Whether or not the unconstrained problem yields the solution to the constrained problem, it may be instructive to evaluate the effects that the constraints would have on the solution. Sometimes in applications, the constraints are somewhat subjective, and understanding their effects may aid in formulating the problem better.

Minimizing Other Functions of the Residuals

For the general objective of minimizing the residuals we have alternatives. We may measure the overall size of the residuals by

$$R_\rho(t) = \sum_{i=1}^n \rho(y_i - f(x_i; t)), \quad (1.17)$$

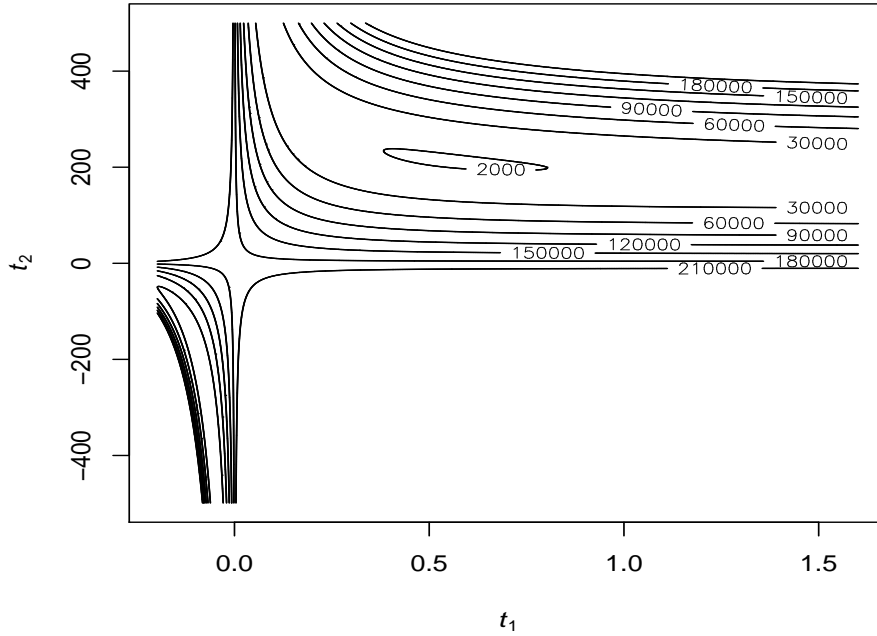


Fig. 1.3. Contours of the Sum of Squares Function for a Nonlinear Model gro0110b

where $\rho(\cdot)$ is some function of $r_i = y_i - f(x_i; t)$. Instead of minimizing the sum of the squares of the residuals, we fit the model by minimizing this measure; that is, by solving an *optimization problem* such as

$$\min_t \sum_{i=1}^n \rho(y_i - f(x_i; t)). \tag{1.18}$$

Depending on $\rho(\cdot)$, this problem is much more difficult both computationally and conceptually than the least squares problem, in which $\rho(r) = r^2$. One common choice of ρ is just the absolute value itself, and the problem of fitting the model is the *optimization problem*

$$\min_t \sum_{i=1}^n |y_i - f(x_i; t)|. \tag{1.19}$$

There is no closed-form solution to this simple least-absolute-values problem, even in the linear case.

In addition to choosing a function of the individual r_i , we might also reconsider how we choose to combine several individual residual values into a

regularization

single measure. Simply summing them, as we have done above, is an obvious way. In practice, however, there may be other considerations. We may want to treat some residuals differently from others. This may be because we may consider the observation on which a particular residual is based to be more precise than some other observation; therefore we may choose to give that residual more weight. Alternatively, we may realize that some observations do not seem to fit our model the same way most of the other observations fit the model; therefore, we may adaptively choose to give those residuals less weight. These considerations lead to a slightly more general formulation of the problem of fitting the statistical model by minimizing residuals, resulting in the *optimization problem*

$$\min_t \sum_{i=1}^n w(y_i, x_i, t) \rho(y_i - f(x_i; t)), \quad (1.20)$$

where $w(y_i, x_i, t)$ is a nonnegative function. Because in practice, for this minimization problem, it is usually not explicitly a function of y_i , x_i , and t , we often write $w(y_i, x_i, t)$ as a simple fixed weight, w_i .

A common instance of problem (1.20) is the weighted linear least squares problem with fixed weights, in which the function to be minimized is

$$R_{w2}(b) = \sum_{i=1}^n w_i (y_i - x_i^T b)^2.$$

The weights do not materially change the complexity of this problem. It has a closed-form solution, just as the unweighted (or equally-weighted) problem (1.14).

Regularization of the Solution

We may also regularize the minimum residuals problem with additional criteria. We form a weighted linear combination of two functions of t ,

$$\sum_{i=1}^n w(y_i, x_i, t) \rho(y_i - f(x_i; t)) + \lambda g(t), \quad (1.21)$$

where g is some nonnegative function and λ is some nonnegative scalar used to tune the optimization problem. The simplest instance of this kind of regularization is in ridge regression with a linear model, in which w is constant, $\rho(z) = z^2$, $f(x_i; b) = x_i^T b$, and $g(b) = b^T b$. The *optimization problem* is

$$\min_b \left(\sum_{i=1}^n (y_i - x_i^T b)^2 + \lambda b^T b \right).$$

In ridge regression, we minimize a weighted combination of L_2 norms of the vector of residuals, $r(b)$, and of the coefficients, b . In lasso regression with

a linear model, an L_2 norm is applied to the residuals and an L_1 norm is applied to the coefficients, and the *optimization problem* is function estimation

$$\min_b (\|r(b)\|_2 + \lambda \|b\|_1).$$

Minimizing Residuals Nonparametrically

There are other ways of approaching the problem of fitting the model (1.2). Instead of fixing the form of the function f and determining a suitable value of θ , we may assume the form of f is unknown (or uninteresting) and approximate it in a way that the approximation $\tilde{f}(x)$ fits the data closely. This kind of approach is *nonparametric*.

The optimization problem that involves determination of $\tilde{f}(x)$ is a quite different problem from our previous examples. In any case, however, the first step is to be clear about our objective. Just to minimize some function of the residuals is not sufficient. Unless we add some conditions on $\tilde{f}(x)$, there are infinitely many solutions that yield 0 residuals (assuming no two observations have the same value of x but different values of y).

In a nonparametric approach, in addition to a requirement that the residuals be small, we may regularize the problem with other criteria for fitting the model. For example, we may require that our approximation $\tilde{f}(x)$ be twice-differentiable and be “smooth”. If we measure the roughness or non-smoothness of a twice-differentiable function f over a domain D by the integral of the square of the second derivative,

$$\mathcal{R}_{22}(f) = \int_D (f''(x))^2 dx, \quad (1.22)$$

we can include this expression in our optimization problem. Our overall optimization would be a weighted combination of this expression and some measure of the residuals. In one common approach, we measure the residuals by the sum of their squares, and formulate the *optimization problem*

$$\min_{\tilde{f}} \sum_{i=1}^n (y_i - \tilde{f}(x_i))^2 + \lambda \mathcal{R}_{22}(\tilde{f}), \quad (1.23)$$

with the restriction that \tilde{f} be twice-differentiable so that $\mathcal{R}_{22}(\tilde{f})$ is defined. In this formulation, λ is a nonnegative smoothing parameter.

So long as $\lambda > 0$ and the function is twice-differentiable, the problem is well-posed. As λ grows without bound, the second derivative is driven to zero, and the solution is a linear function; that is, the problem is the same as problem (1.14). For finite positive λ , the solution to the optimization problem (1.23) is a natural cubic spline with knots at the x_i . (Although this is true in higher dimensions also, the formulation is typically used when x is a scalar.)

maximum likelihood
estimation

Maximum Likelihood Estimation

Another way of fitting the model $y = f(x; \theta) + \epsilon$ is by *maximizing the likelihood function* that arises from the probability distribution of ϵ . Given the data, this is the *optimization problem*

$$\max_t \prod_{i=1}^n p(y_i - f(x_i; t)), \quad (1.24)$$

where $p(\cdot)$ is the probability function or the probability density function of the random error. Again we are using the vector t as a variable in place of the fixed but unknown vector θ . Optimization problems of this type can be quite formidable computationally. Although the statistical properties of the maximum likelihood estimators are often quite difficult to work out, they generally have relatively simple asymptotic behavior.

For a given probability density $p(\cdot)$, the maximization problem (1.24) may be equivalent to a minimization problem of the general form (1.18). For example, if the distribution is normal, problem (1.24) becomes the least squares problem, and if the distribution is double exponential (Laplace), the maximization problem is the least absolute values problem (1.19).

In a nonparametric formulation of the maximum likelihood approach, we are faced with the same kind of problem we had in fitting the model (1.2) by minimizing residuals. Unless we regularize the problem with additional criteria, the problem is ill-posed. We can “penalize” the likelihood with a regularization measure that decreases (remember we are maximizing) as we move away from the desirable solution. For example, if we require that the function be smooth (and, hence, twice-differentiable), we may form the *optimization problem*

$$\max_{\tilde{f}} \prod_{i=1}^n p(y_i - \tilde{f}(x_i)) e^{-\lambda \mathcal{R}_{22}(\tilde{f})}, \quad (1.25)$$

where $\mathcal{R}_{22}(\cdot)$ is the functional given in equation (1.22).

Building Models

Use of the methods we have discussed so far relies on some given model or class of models. In the very general parametric model (1.3) we may have different choices for the form of the model, as illustrated in Figure 1.1, or we may assume that f is of a given form and that x and θ have a given structure. These assumptions result in the problem of fitting a given model to the given data; that is, we have the problem that we have discussed to this point. This is the problem of estimating θ .

A very common simple extension of this situation is one in which θ is a vector of *possible* parameters, and our objective is to use the data to determine *which* elements of θ are relevant and to estimate the relevant ones. In

this case, each of the possible parameters is usually associated with a different explanatory variable. The problem of building a model is the problem of choosing the parameters and, consequently, the explanatory variables to include in the model.

The general problem of developing a model is much larger than the problem of estimating the parameters in a given model. There is a well-developed theory for parametric estimation in a given model; optimality properties are well-understood. Likewise, for “nonparametric” estimation, there is well-developed theory for estimation of “nonparametric” functionals, such as quantiles and functions of quantiles. If a model is not given, however, we can always build some model that fits any dataset perfectly. It may be a very complicated model, and our concern is that we not “overfit”. We may build a model that fits the given data well, but really does not fit the underlying phenomenon very well.

The familiar variable-selection problem in linear regression illustrates the issues and approaches for the general problem of building a model from a class of models with a vector of possible parameters. The quantity to be optimized in fitting a model by methods we have discussed above can almost always be further optimized by adding additional components to the model. In least-squares fitting of a linear regression model, the quantity to be minimized (expression (1.14)),

$$\min_b \sum_{i=1}^n (y_i - x_i^T b)^2,$$

if it is not already 0, can be reduced further by augmenting the vector b with an additional element and each vector x_i with an additional element in such a way that the n -vector of additional elements is linearly independent of the n -vectors consisting of elements in each position of the x_i vectors. If b and x_i are m -vectors, we can denote the additional terms as b_{m+1} and $x_{i,m+1}$. In other words, the quantity

$$\min_{b, b_{m+1}} \sum_{i=1}^n (y_i - (x_i^T b + x_{i,m+1} b_{m+1}))^2$$

is smaller than the previous quantity. (This is an obvious and standard property of an optimization problem; if an additional variable can be used to modify the expression, the value of the expression can be made at least as optimal as before.)

The issue is clear: by adding more terms to the model, whether or not they make any sense, we can achieve a better solution to the optimization problem, unless the given solution is a perfect fit. As an aside, we remark that this is a general problem when optimization is used in any method: the optimization ultimately magnifies the effect of any assumption or any model we have chosen.

We might ask whether it makes any difference; that is, why not just go with the more complex model because it will improve our criterion for fitting

experimental design

the model. Among the many reasons we may advance for eschewing complex models, we note first the heuristic desirability of simple models. Secondly, if the model has superfluous terms, the predicted values of y for new values of x may be far from the predicted values of a correct model. This fact is relevant because an important reason for building a model in the first place is to use it for prediction.

There are two general ways to approach this problem in model building. One is to include in the optimization problem a term or a factor to penalize the expression to be optimized with some measure of model complexity. The other is to use internal validation of the model. Internal validation can be accomplished by dividing the sample into subsamples, fitting the model using one subsample, and assessing the fit for another subsample. There are many variations on this general approach.

An example of the use of a penalty factor is in the variable selection problem in linear regression using least squares is selection based on maximum adjusted R^2 . Although there are many deficiencies for this method, it is widely used. The *optimization problem* is

$$\min_{b,P} \sum_{i=1}^n \frac{n}{n - \#P} \left(y_i - \sum_{j \in S} x_{ij} b_j \right)^2, \quad (1.26)$$

where P is a subset of the power set of $\{1, \dots, m\}$, and $\#P$ is the cardinality of P . As the number of variables in the model increases, the squared residuals decrease, but the overall expression may not decrease because of the decrease in the denominator. (The expression in (1.26) to be minimized is a monotone transformation of the adjusted R^2 .)

The various methods of internal validation often result in iterative optimization problems using subsets of the data. If some subset of the given data is not used in fitting the model, that subset may be used to develop some measure of goodness of fit or some measure of residual variation not accounted for by the model.

Optimal Experimental Design

Other optimization problems in statistics arise in optimal design of experiments and in the construction of optimal sampling plans. In design of experiments, we often assume a linear relationship between y and an m -vector x , and we anticipate collecting n observations, (y_i, x_i) , into an n -vector y and an $n \times m$ matrix X . We may express the relationship as

$$y = \beta_0 1 + X\beta + \epsilon.$$

Under the assumption that the residuals are independently distributed with a constant variance, σ^2 , the variance-covariance matrix of estimable linear functions of the least squares solution are formed from

$$(X^T X)^{-1} \sigma^2.$$

sampling design
calibration

Because we may be able to choose the values in X , that is, the settings at which we make observations, we may attempt to choose them in such a way as to minimize variances of certain estimators. The variance of a particular estimator is minimized by maximizing some function of $X^T X$. There are various ways we may attempt to minimize the variances of a collection of estimators. A common method in experimental design results in the *optimization problem*

$$\max_{\text{all factor settings}} \det(X^T X). \quad (1.27)$$

If there are many possible values of X , this may be a difficult computational problem.

Optimal Sampling Design

In designing a sampling plan, we are often presented with the problem of allocating the sample sizes n_h within strata. For given population strata sizes N_h and known (or assumed) within-strata variances v_h , the *optimization problem* has the form

$$\min_{1 \leq n_h \leq N_h} \sum_h N_h \left(\frac{N_h}{n_h} - 1 \right) v_h. \quad (1.28)$$

This is a different kind of optimization problem from any we have discussed up to this point. The values of the variables in this optimization problem, that is, the n_h , are restricted to be positive integers. The methods to solve such a problem are often much more complicated than those for continuous variables.

Determination of optimal sampling plans can be a very difficult optimization problem if more complicated designs are used, or if other considerations, such as cost of conducting the sample, are brought to bear. In a two-stage design, for example, we have the problem of allocating the sample sizes n_h and m_h within various strata and across multiple stages. For given population sizes N_h and known within-strata variances for the first and second stages v_h and v_{2h} , we have the *optimization problem*

$$\min_{n_h, m_h} \left(\sum_h N_h \left(\frac{N_h}{n_h} - 1 \right) v_h + \sum_h \frac{N_h^2}{n_h m_h} v_{2h} \right).$$

Calibration

Often data collected as responses to questionnaires contain obvious inaccuracies, and the sampling agency wants to correct those inaccuracies while changing the data as little as possible. Sometimes a list of “edits” is available that contains rules that data items must obey. (These may be simple rules,

orthogonal residuals

such as the requirement that an item be nonnegative. They may be reasonable but somewhat subjective, such as the rule that the age of a parent must be at least ten years greater than the age of a child.)

The techniques of data editing are also employed in record matching. In this application, the amount of change required to make an identifying set of fields in one record to correspond to those fields in another record is assessed. If no change or only a small change is required, the two records are deemed to match.

A common instance of data adjustment, usually called “calibration” rather than “editing”, is the adjustment of tabular data to fixed marginal totals.

In sampling applications, an observational record may consist of many individual responses, and certain information may already be available for some items. For a given item, X , a total, τ_{X_D} , over a domain, D , may be known. (The domain may be a stratum in the sampling design, or it may just be some administrative subdistrict.) If the observations on X in the domain are x_1, \dots, x_n , and the corresponding sampling weights are d_1, \dots, d_n , the estimated domain total is $\hat{\tau}_{X_D} = \sum d_k x_k$. The calibration problem is to choose new weights w_1, \dots, w_n , “close to” the original weight, but such that $\sum w_k x_k = \tau_{X_D}$. The differences in w_k and d_k can be measured in terms of $|w_k - d_k|$ or w_k/d_k . (All weights are nonzero.) If $g(\cdot)$ is a measure of the distance from w_k to d_k measured in terms of w_k/d_k we form the calibration *optimization problem* as

$$\min_{w_k} \sum_{k=1}^n d_k g(w_k/d_k), \quad (1.29)$$

subject to the requirement $\sum w_k x_k = \tau_{X_D}$. The terms in the function to be minimized are weighted by the original weights, which generally correspond to the importance attached to the individual observations.

Calibration performed as described above is called “regression calibration” in the sampling literature. A more general calibration problem arises in a regression model $y = f(x; \theta) + \epsilon$ (most often in its linear form of equation (1.4)), in which the objective is to calibrate x so that it agrees with the observed y as closely as possible. This is equivalent to defining the fitted residuals in the horizontal direction, rather than in the vertical direction as in Figure 1.1.

Orthogonal Residuals

An interesting problem arises in regression when the model accommodates observational or sampling errors in x . The errors-in-variables problem may lead to an orthogonal regression approach in which the residuals are not measured in a vertical direction, as in the usual formulation, or in a horizontal direction, as in a calibration problem, but rather in an orthogonal (or normal) direction from the model surface. As in the case of the ordinary vertical residuals, various norms could be applied to the vector of orthogonal residuals. The most

common norm, just as for ordinary residuals, is the L_2 norm. For ordinary residuals, this can be expressed very simply as in equation (1.13), leading to an ordinary least-squares minimization problem of the form (1.14). For orthogonal residuals, however, because the direction of the residuals depends on the fitted model, we cannot express the optimization problem in terms of a vector norm. For a linear model of the form

$$y \approx X\beta,$$

the problem of finding b so as to minimize the L_2 norm of the orthogonal residuals is the problem of finding b that satisfies

$$\tilde{y} = \tilde{X}b$$

where \tilde{y} and \tilde{X} are solutions to the *optimization problem*

$$\min_{\tilde{y} \in \text{span}(\tilde{X})} \left\| [X \ y] - [\tilde{X} \ \tilde{y}] \right\|_F, \quad (1.30)$$

where $\text{span}(\tilde{X})$ is the column space of \tilde{X} , $[X \ y]$ and $[\tilde{X} \ \tilde{y}]$ are the matrices formed by adjoining as the last column the vector in the second position to the matrix in the first position, and $\|\cdot\|_F$ is the Frobenius matrix norm (see page 43).

To minimize other norms of the orthogonal residuals requires solution of even more complicated optimization problems. Even minimizing the L_2 norm of the orthogonal residuals, as above, cannot be performed by evaluating a closed-form expression. The orthogonal regression solution is in the subspace of the eigenvectors of the X matrix augmented with a column formed by the y vector; that is, it is a combination of principal components, which we discuss below.

Response Surface Methodology

In an important class of applications, a model such as (1.3) is used to represent the effects of some controllable factors measured by x on the response of some quantity of interest measured by y . In a manufacturing process, for example, y may be a measure of tensile strength of the product and the vector x may be composed of measurements of the temperature and the pressure in the mold during some phase of production process. Temperature and pressure settings that are either too low or too high yield product with low tensile strength. The objective in *response surface methods* is to determine the optimal combination of settings of x for the response y ; that is, to solve the *optimization problem*

$$\max_x f(x; \theta). \quad (1.31)$$

Note that the problem as posed here is not the problem of fitting a model that we have been discussing. This optimization problem cannot be solved because

function estimation

θ is unknown; hence, we must first fit the model, that is, estimate $f(x; \theta)$. Response surface methodology therefore involves two optimization problems: fitting of the model (which is usually just a trivial application of least squares), that is, determination of the “best” value of t in place of θ , and then finding the optimum of the function $f(x; t)$ over x .

Some of the more interesting questions in response methodology involve the choice of settings of x at which to make observations on y , so as to ensure that $f(x; t)$ approximates $f(x; \theta)$ very well in the neighborhood of the optimum of $f(x; \theta)$, and so that the optimum of the function $f(x; t)$ can be found easily.

Estimation of Functions

An interesting statistical problem is the estimation of a function that relates one variable to the expected value of another variable, $E(Y) = f(x)$. In some cases we can write this model as in equation (1.2):

$$Y = f(x) + \epsilon.$$

In this version, the difference $Y - f(x)$ is modeled as the random variable ϵ . There are other possible formulations, but we will consider only this additive model. The problem is to estimate f given a set of observations y_i and x_i .

We mentioned a general nonparametric approach to this problem above, and gave the regularized optimization problem (1.23) for a solution. We also referred to a maximum likelihood approach to this problem above, mentioned the difficulty in posing the problem, and then gave a penalized likelihood in the optimization problem (1.25). Maximum likelihood estimation, of course, requires a probability distribution.

A general approach to this problem is the same as the one we took in estimating θ in the model $y = f(x; \theta) + \epsilon$: first we replace the unknown (but fixed) estimand with a variable, and then minimize the residuals. In this case, we consider some class of functions that have the same domain D as the domain of f . We represent a possible function as $h(x)$ and then seek the $h(x)$ that is as close to $f(x)$ as possible, under some measure of closeness. The measure of distance between the function is usually defined in terms of a function norm of the difference, $\|f - h\|$. (We discuss norms of functions in more detail beginning on page 50.)

Because the only information we have about $f(x)$ is contained in the observations y_i and x_i , the problem is not well-posed. Any $h(x)$ such that $h(x_i) = y_i$ provides a good fit to the data. To make the problem well-posed we must impose additional conditions.

In the maximum likelihood approach on page 12, we imposed a probability distribution and a smoothness criterion. One common method in function estimation is to define a set of *basis functions* on D that span some class of functions that either contains f or contains functions “sufficiently close” to f by our measure of closeness. After a set of basis functions, $\{q_k\}$, is chosen, our estimating function h is chosen as

$$h(x) = \sum_k c_k q_k(x),$$

where the c_k are taken as a solution to the *optimization problem*

$$\min_{c_k} \|f - c_k q_k\|. \quad (1.32)$$

The basis functions are often chosen as orthogonal polynomials.

Nonparametric Probability Density Estimation

One of the most interesting problems of function estimation in statistics is nonparametric probability density estimation. In this case, the function to be estimated does not relate one variable to another, as in the case above where the observable y was related to the observable x , but rather the function specifies a *density*, which is not directly observable. Estimation of the function in this case must be based on relative frequencies of the values of x that are observed.

For this problem, the probability density function can be approximated in terms of basis functions as indicated above; more commonly, however, other techniques based on bins of observations or kernels over the observation space are used.

Optimization Applications in Multivariate Analysis

In multivariate analysis we seek to understand relationships among variables or among observations on those variables. The standard dataset in multivariate analysis is a matrix, X , in which the columns correspond to variables and the rows correspond to observations.

One of the most commonly used multivariate methods is principal components analysis. The objective is to find a normalized linear combination of the variables that yields the largest sample variance. This is the *optimization problem*:

$$\max_{z \ni \|z\|_2=1} z^T S z, \quad (1.33)$$

where S is the sample variance-covariance matrix based on X .

In multivariate analysis we often are interested in the “distances” or “dissimilarities” within pairs of observations. If x_i and x_j are observations (that is, rows of X), the Euclidean distance between x_i and x_j is $\|x_i - x_j\|_2$. Other kinds of distances or dissimilarities between x_i and x_j could be defined. For a given definition of dissimilarity, we often denote the dissimilarity between x_i and x_j as (δ_{ij}) . Most useful definitions of dissimilarity would result dissimilarities that are nonnegative, that are 0 for $x_i = x_j$, and which could be given a partial ordering that is not inconsistent with the ordering of Euclidean distances.

classification
clustering

An interesting problem in multivariate analysis is, given a matrix of dissimilarities (δ_{ij}) , to find an $n \times k$ matrix Z with rows z_i that are solutions to the *optimization problem*:

$$\min_{i,j} \frac{(\delta_{ij} - \|z_i - z_j\|_2)^2}{\|z_i - z_j\|_2}. \quad (1.34)$$

A number of different optimization problems immediately suggest themselves (such as changing the norm on $(z_i - z_j)$ or changing the measure of the distance from the δ_{ij} to the $\|z_i - z_j\|_2$); and indeed, there are many variations of this basic problem.

Classification and Clustering

Less formal statistical methods also use optimization. A common example is called “classification” or “statistical learning”. In this application, we have a set of multivariate data that consists of subsets from two or more classes. One of the variables, say g , identifies the class to which a given observation belongs. The objective is to determine some function of some of the other variables that can be used to predict the value of g .

Except in very trivial applications in which we assume a discrimination model, classification is always an exercise in building models, as we discussed on page 12. The classification model is of the form $g \approx f(x)$, where g represents some category, x represents the values of some covariates, and f represents some rule that yields a likely category within the values of g . In the general problem, we have a set of data $(g_1, x_1), \dots, (g_n, x_n)$, and we seek to determine a function \tilde{f} that fits the given data well, and could be used to predict the value of g for a newly observed value of x .

One of the simplest approaches to classification is to construct a tree by choosing the first node as being a range of values of x that split the given data into two groups that, among all pairs of groups that could be formed by permissible divisions of the range of x , have the “best” division of categories of g . This classification method is then applied recursively to the subgroups. In this case, the function $\tilde{f}(x)$ is in the form of a decision tree whose terminal nodes represent the classes to which observations are assigned. If a new value of x_0 is observed, the appropriate class of the observation would be $g_0 = \tilde{f}(x_0)$.

In this method of *recursive binary partitioning*, there are obviously many issues. In the general description of recursive binary partitioning, the first questions are what are the permissible divisions of the range of x , and how is “best” defined. Whatever methods are chosen for the partitioning, the next question is how big should the tree be; that is, how many nodes should the tree have. A tree of sufficient size could consist of terminal nodes or leaves that have perfect separation, that is, within a single leaf there would be a single value of g .

To define an optimization problem that is used in a simple method of recursive binary partitioning of k categories (that is k different classes, as

specified by g), in which the range of x is divided into two using one element of x at a time, consider the decision at a given node, that begins with n_t observations. Assume the decision will be made on the basis the value of the j^{th} variable, x_j , with respect to some chosen value x_{j0} ; that is, if for an observation under consideration, $x_j \leq x_{j0}$, the observation is put in the “left” group, and otherwise the observation is put in the “right” group. The split is chosen as the solution to the *optimization problem* is

k-means clustering

$$\max_{x_j, x_{j0}} n_L n_R \left(\sum_{i=1}^k |L_i n_L - R_i n_R| \right)^2, \quad (1.35)$$

where n_L and n_R are the number of observations that are assigned to the left and right groups, respectively, and L_i and R_i are the number of group i assigned to the left and right groups, respectively.

There are many variations of this basic procedure based on different ways of splitting the range of x and different measures of goodness of the split. This optimization problem would be applied recursively until some stopping criterion is reached. For whatever method is chosen for use at a given node, the decision of when to stop is similar to the problem we discussed on page 14, illustrated by variable selection in regression. We can use some penalty factor that increases for larger trees, or we can use internal validation methods, such as cross validation.

A related problem is “unsupervised classification” or clustering. In this case, we have observations x_1, \dots, x_n that belong to different groups, but the groups are unknown. The objective is to cluster the given data into the appropriate classes, based on some measure of similarity of the x s.

In k-means clustering, for example, we seek a partition of a dataset into a preset number of groups k that minimizes the variation within each group. Each variable may have a different variation, of course. The variation of the j^{th} variable in the g^{th} group is measured by the within sum-of-squares:

$$s_{j(g)}^2 = \frac{\sum_{i=1}^{n_g} (x_{ij(g)} - \bar{x}_{j(g)})^2}{n_g - 1}, \quad (1.36)$$

where n_g is the number of observations in the g^{th} group, and $\bar{x}_{j(g)}$ is the mean of the j^{th} variable in the g^{th} group. For data with m variables there are m such quantities. In k-means clustering the *optimization problem* is

$$\min_{\text{all partitions}} \sum_{g=1}^k \sum_{j=1}^m s_{j(g)}^2. \quad (1.37)$$

There are various modifications of the classification and clustering problem. In one kind of modification, we use a continuous variable to define a spectrum of groups. If g is a continuous variable, instead of the problem (1.35) involving $|L_i n_L - R_i n_R|$, we may formulate a minimization problem over x_j

k-models clustering

and x_{j0} that involves a measure of how well y can be fit separately in the two groups, for example, the *optimization problem*

$$\min_{x_j, x_{j0}} \left(\min_a \sum_{i \in L} (y_i - x_i^T a)^2 / (n_k - n_L) + \min_b \sum_{i \in R} (y_i - x_i^T b)^2 / (n_k - n_R) \right). \quad (1.38)$$

This problem is obviously more complicated than problem (1.35).

Another way of forming clusters, related to k-means clustering, can be used when the data in the g^{th} group follow a model such as equation (1.1)

$$y \approx f_g(x).$$

In this method, called k-models clustering, we assume the data within the different clusters follow different models, and we define clusters based on how well the data within a given cluster fit the same model. Given a set of potential models $\{f_g\}$, the best fitting model in each potential group is determined, and the residual sums corresponding to the sums of squares in equation (1.36),

$$R_{(g)}^2 = \frac{\sum_{i=1}^{n_g} (y_{i(g)} - \hat{y}_{i(g)})^2}{n_g - 1}, \quad (1.39)$$

are used as a measure of the homogeneity of the groups. In k-models clustering the *optimization problem* is

$$\min_{\text{all partitions}} \sum_{g=1}^k \sum_{j=1}^m R_{j(g)}^2. \quad (1.40)$$

Constraints on the Variables

In several of the optimization problems we have discussed above, we required that the solution satisfy certain restrictions. For example, in the principal components problem, we required that the solution satisfy $\|z\|_2 = 1$ (or, equivalently, $z^T z = 1$). We often express such *constrained optimization problems* in two parts, one consisting of the function to be optimized and the other in an additional line or lines introduced by “such that” or “s.t.”. The principal components optimization problem (1.33) is thus the constrained problem

$$\begin{aligned} \max_z \quad & z^T S z \\ \text{s.t.} \quad & z^T z = 1. \end{aligned} \quad (1.41)$$

Linear Programming Problems

A wide range of problems can be formulated as a given linear combination of nonnegative variables that is to be maximized or minimized subject to constraints on a set linear combinations of those variables.

$$\begin{aligned} \max_x \quad & c^T x \\ \text{s.t.} \quad & Ax \leq b \\ & x \geq 0. \end{aligned} \tag{1.42}$$

In the first significant work on optimization problems of this type, the problems arose in the context of management of activities and allocation of resources. Because the motivation was for planning, the problems were called “programming problems”, and the particular formulation (1.42) was called a “linear programming problem”. (When this term first came into use, “programming” was not a common term for writing computer code.)

The problem (1.42) is the canonical formulation of the linear programming problem, but there are many variations on this problem. There are two fundamental approaches to solving linear programming problems, which we discuss briefly in Section 6.1.2. These basic methods, however, have specialized versions for different variations of the problem.

The least-absolute-values problem (1.19),

$$\min_b \sum_{i=1}^n |y_i - x_i^T b|,$$

that arises in fitting the linear model $y = x^T \beta + \epsilon$ can be formulated as a linear programming problem. We first define the vector $r^+(b)$ by $r_i^+ = \max(0, y_i - x_i^T b)$ and the vector $r^-(b)$ by $r_i^- = \max(0, -y_i + x_i^T b)$, and then we have

$$\begin{aligned} \min_{r^+(b), r^-(b)} \quad & \sum_{i=1}^n (r_i^+ + r_i^-) \\ \text{s.t.} \quad & Xb + Ir^+(b) - Ir^-(b) = y \\ & r^+(b), r^-(b) \geq 0, \end{aligned} \tag{1.43}$$

where I is the $n \times n$ identity matrix, and X is the matrix of the observed x_i as before. There are specialized linear programming algorithms to solve this particular problem more efficiently than the standard linear programming algorithms.

Traveling Salesperson Problems

Another interesting optimization problem is called the *traveling salesperson problem*. In its simplest form this is the problem of finding the minimum-cost path that connects all cities, given the cost of traveling from any particular city to any other city. There are various modifications of this basic problem that may involve various constraints, or various additional costs that depend on characteristics of the path. Variations of this problem have applications in statistical planning and design.

dense domain

Domain of an Optimization Problem

We can observe important differences among the examples above. One important difference is in the nature of the set of possible values of the variable of the optimization problem. This is called the *domain* of the problem. Two important characteristics of the domain are its *cardinality* and its *order* (or its “dimension”).

In estimation problems similar to (1.14) or (1.24), we have assumed that β or θ could be any real number (or vector), and so the variable b or t in the optimization problem could range over the real numbers. We say the domain for each variable is the reals. The order is the number of elements in b or t . The cardinality is the same as that of the reals. In the optimal experimental design problem (1.27), the domain is the set of $n \times m$ real matrices. The order is nm , and the cardinality is the same as that of the reals.

In the sampling allocation problem (1.28), on the other hand, the values of n_h and m_h must be nonnegative integers bounded from above. The cardinality is finite. The domain of the clustering problem (1.37) is also different; it is the collection of all partitions of a set. In an important way, however, the domain of problem (1.37) is similar to the domain of nonnegative integers with upper bounds; in fact, a reasonable approach to solving this problem depends on an indexing of the partitions using integers. In the traveling salesperson problem, the domain is the set of permutations of the cities.

The domains of these various optimization problems we have mentioned are very different; in some cases the domains are dense in \mathbb{R}^d (for some positive integer d), and in the other cases the domains are countable, in fact, they are finite. (In general terms, a *dense domain* is one in which between any two points there is another point.) In practice, the difficulty of the optimization problem is often less for a dense uncountable domain than it is for a finite domain with a large number of elements, such as the traveling salesperson problem.

The order, or dimension, of the domain is closely related to the difficulty of solving an optimization problem. In the estimation problems (1.14) and (1.24), the order is likely to be relatively small, maybe five to ten parameters. In sampling calibration, such as problem (1.29), the order may be extremely large. The order is the number of records in the dataset.

The countable domains may not be directly equivalent to the positive integers, because they may not have an ordering. Of course, by indexing the elements of the domain, we can impose the natural ordering of the integers, but that ordering may have nothing to do with the problem, and using the ordering may complicate attempts at solving the problem.

Sometimes the domain may consist of a dense subset of \mathbb{R}^d and a separate countable set. In that case the cardinality of the domain is of course the same as the cardinality of the reals, but this kind of domain may present special difficulties in solving an optimization problem. Not all domains are either countable or have the cardinality of the reals. In functional analysis,

say in the nonparametric estimation of functions for example, as discussed on page 18, the domain is a class of functions, and its cardinality is larger than that of the reals. well-posed problem

Formulation of an Optimization Problem

The general examples above represent a variety of optimization problems that arise in statistics. Notice the basic procedure in the estimation problems. We begin with a model or a probability statement with fixed but unknown parameters and *variables that are realizations of random variables*. We define a measure of the overall difference in a fitted model and the observations. We then define a minimization or maximization problem in which the realizations of the random variables are fixed, and *variables are substituted for the unknown parameters*. We call the function being minimized or maximized in an optimization problem the *objective function*.

We have seen examples whose objective functions have domains that are dense and others with domains that are countable. We have seen examples with domains that are unbounded and others with domains that are constrained.

The formulation of a statistical problem or any problem in data analysis as an optimization often helps us to understand the problem and to focus our efforts on the relevant aspects of the problem. In calibration of tables or in data editing, for example, we seek adjustments that represent minimal changes from the original data. If we do not think clearly about the problem, the resulting optimization problem may not be *well-posed*; that is, it may not have an unambiguous solution. The functional optimization problems (1.23) and (1.25), for example, are not well-posed (they are *ill-posed*) without the regularization component (or if $\lambda = 0$).

One of the most worrisome problems arises when the optimization problem has multiple points of optimality. In the least squares problem with contours of the sum of squares of residuals as shown in the graph in Figure 1.3, the presence of multiple local minima should cause us to think about the problem more deeply. As we will see in later chapters, this kind of situation can cause problems for whatever computational method we are using. More serious, of course, are implications for our objectives in data analysis. In this particular example, the two different local minima are quite different in value, so it is clear which is really the minimum, that is, which is the “correct” value. Suppose, however, that because of random sampling, the two minima were closer in value. Suppose, depending on the data, that different sets of contours with shapes similar to those in Figure 1.3 had global minima in different relative positions. Of course, we only have one dataset, so we may not know this unless we use statistical methods such as data splitting or model simulation. The problem of multiple local modes can arise in either of the common statistical methods of minimizing norms of residuals or maximizing likelihoods. It occurs often when the dimensionality of the estimand space is large. In

such cases, even if we could always find the global optimum, it is not clear that it would be the right solution. At the very least, when an optimization problem has multiple local optima it should cause the statistician to think more carefully about the formulation of the problem. The objective function should correspond to the objectives of the analysis.

In formulating a statistical problem as an optimization problem, we must be careful not to change the statistical objectives. The objective function and the constraints should reflect the desired statistical methodology. If a different or incorrect optimization problem is formulated because of computational considerations, we should at least be aware of the effect of the change. An example in the literature of how available software can cause the analyst to reformulate the objective function began with the problem of fitting a linear regression model with linear constraints; that is, a problem like (1.44) below, in which the constraints on b were of the form $Ab \leq c$. It turns out that an optimization problem like (1.19), that is, *least absolute values* regression, with $f(x_i; b) = x_i^T b$, that is, *linear* regression, and with constraints of the form $Ab \leq c$, can be formulated as a linear programming problem (1.43) with some additional constraints (see Charnes, Cooper, and Ferguson, 1955), and solved easily using available software. At the time, there was no readily available software for constrained least squares regression, so the reformulated problem was solved. This reformulation may or may not have undesirable side effects on the overall analysis. To repeat: the objective function should correspond to the objectives of the analysis, and these may specify an inferential method.

The solution to an optimization problem is in some sense “best” for that problem and its objective function. This fact may mean that the solution is considerably less good for some other optimization problem. It is often the case, therefore, that an optimal solution is not robust to assumptions about the phenomenon being studied. Use of optimization methods is likely to magnify the effects of any assumptions.

Optimization Problems with Constraints

As we have seen in the examples above, there may be restrictions placed on the solution of an optimization problem. Only some subset of the domain, called the *feasible region*, is allowed because of constraints on the variables.

Often a statistical model may include a specification that the unknown parameter is in some given subregion of Euclidean space; that is the *parameter space* may not be the full Euclidean space. In this case, we generally constrain the variable that we substitute for the parameter also to be in the parameter space, as we did in the optimization problem (1.16). More generally, if it is known that θ in the model (1.3) is in some given region D , instead of the unconstrained optimization problem (1.14), we have the *constrained optimization problem*

$$\begin{aligned} \min_t \sum_{i=1}^n (y_i - f(x_i; t))^2 \\ \text{s.t. } t \in D. \end{aligned} \quad (1.44)$$

These constraints make the simple least squares problem much more difficult, both computationally and conceptually. The statistical properties of a least squares estimator subject to these constraints are not easy to determine.

Any of the other optimization problems we formulated above for estimating parameters can also be modified to accommodate constraints on the parameters. The problem of designing a sampling plan, for example, often includes constraints on the total cost of conducting the survey or constraints on the coefficient of variation of the estimators.

Sometimes the constraints in an optimization problem are the primary reason for formulating the problem, as is the case in the calibration problem (1.29).

Constraints can often be accommodated by transformation of variables or by reexpressing the function to be optimized. For example, the principal components problem (1.33) or (1.41) with the constraint $z^T z = 1$ is equivalent to the unconstrained *optimization problem*

$$\max_z \frac{z^T S z}{z^T z}. \quad (1.45)$$

We can also incorporate the constraints into the objective function by adding a term that measures the extent to which the constraints are not being satisfied. In the principal components problem above, equation (1.41), because S is positive definite, the larger the elements of z , the larger the objective function. The difference $(z^T z - 1)$ is the amount by which the larger elements of z would cause the constraint to be exceeded. We can force these constraints to be satisfied by writing the unconstrained *optimization problem*

$$\max_{z, \lambda} (z^T S z - \lambda(z^T z - 1)). \quad (1.46)$$

This objective function is called a Lagrangian, and λ is called a Lagrange multiplier.

Alternatively, we might consider a function that increases as $z^T z$ moves away from 1 in either direction, say $(z^T z - 1)^2$, and subtract a fixed multiple of it from the main part of the objective function:

$$\max_z (z^T S z - \lambda(z^T z - 1)^2). \quad (1.47)$$

Choosing a large value of λ in this problem increases the effect of the constraint. This type of approach makes the constraints “soft”. It allows us to vary the extent to which the constraints control the solution. Setting λ to 0, removes the constraints entirely (which, of course, in this particular case would leave the problem ill-posed). Incorporating the constraints into the objective function in this way is similar to the regularization methods we discussed above.

Optimization of Multiple Objectives

In most practical applications with optimization, there are more than one objective. A simple example of this is the general optimization problem of minimizing the residuals in order to fit a model. Because there are many residuals, we must decide how we want to minimize them all simultaneously. Of course, the obvious solution to this quandary is just to minimize the sum (of some function) of the residuals. Even in this simple example, however, there may be reasons to combine the residuals differentially, as in weighted regression.

In a problem of optimal sampling design the general objective is to minimize the variance of an estimator. Realistically, there are many estimates that result from a single survey; there are several attributes (estimands) crossed with several strata. The problem of how to address the problem of minimizing the variances of all within strata estimators in a single sampling design requires consideration of the relative importance of the estimates, any constraints on variances and/or coefficients of variation, and constraints on the cost of the survey. It may also be possible to relax hard constraints on variances or costs and to include them as additional objectives.

There are various ways of accommodating multiple objectives and constraints. The simplest, of course, is to form a weighted sum. Constraints can be incorporated as a weighted component of the objective function. The weight controls the extent to which the constraint is met.

In most cases, a certain amount of interaction between the decision maker and the optimization procedure is required.

Optimization Methods

The methods of optimization depend on the nature of the functions involved. The first consideration is the domain of the function. Many functions of interest have continuous (dense) domains, and many of the methods of optimization are developed for such functions. Many methods for such functions also assume the functions are differentiable, so for functions with dense domains, the differentiability of the functions are of primary concern. The simplest differentiable functions to work with are polynomials, and among these, linear and quadratic functions are particularly simple. Optimization methods for differentiable functions are often developed using a quadratic function as a prototype function. Many methods for optimization proceed by taking a derivative and setting it equal to 0.

In addition to the wellknown relationship of function derivatives to the optima of functions, there are other properties of functions that may be used in solving optimization problems. For example, the solution to the principal components optimization problem (1.33) is the normalized eigenvector corresponding to the maximum eigenvalue.

Many functions of interest have discrete domains. Points in the domain are combinations of allowable values of a set of variables, and the problem is one of combinatorial optimization.

For either discrete or dense domains, restrictions within the domain can make a relatively simple problem much more difficult.

maximum likelihood estimation
 root of the likelihood equation (RLE)
 RLE (root of the likelihood equation)
 ill-conditioned problem

Solution of an Optimization Problem and Solution of the Problem of Interest

As we mentioned above, we must be careful to formulate an optimization problem that captures the objective of the real problem. By the nature of optimization, the effect of any assumptions about the real problem may be magnified. Therefore, in addition to being concerned about the agreement of the formulation of the optimization problem with the problem of interest, we must also be careful about the analysis of that underlying problem.

Even when the problem is formulated correctly, there may be difficulties in using the optimization problem in a statistical method. A simple example of this occurs in maximum likelihood estimation, that is, solving the optimization problem (1.24) with the objective function $L(t; y, x) = \prod_{i=1}^n p(y_i - f(x_i; t))$. From elementary calculus we know that if $L(t; y, x)$ is differentiable in t , and if it has a maximum at t_* for given y and x , then $\partial L(t; y, x)/\partial t|_{t_*} = 0$. The equation

$$\partial L(t; y, x)/\partial t = 0$$

is called the “likelihood equation”, and a solution to it, t_0 , is called a root of the likelihood equation (RLE). The fact from calculus just quoted, however, does not say that an RLE yields a maximum of L ; the RLE can be just any stationary point, including a minimal point. We can ensure that an RLE is a *local* maximum of L if we find the matrix of second derivatives to be negative definite at the RLE, but another problem in using an RLE as an MLE (maximum likelihood estimate) is that the maximum of the likelihood over the parameter space may occur on the boundary of the space, and the derivatives there may not be zero.

There are many other issues that can arise in using optimization methods. For example, there may be more than one point of optimality. This situation brings into question the formulation of the statistical method as an optimization problem.

Software for Optimization

Complications often occur because of numerical problems in the optimization algorithms used. Optimization problems can be notoriously ill-conditioned, due to, among other things, highly variable scaling in the variables. (An *ill-conditioned problem* can be described simply as one in which small changes

in the input or in computations at intermediate stages may result in large changes in the solution.)

Although good computational algorithms are known for a wide range of optimization problems and stable software packages implementing those algorithms are available, in the current state of the science, we cannot assume that computer software can solve a general optimization problem without some intervention by the user. For many classes of optimization problems, in fact, the required level of user interaction is quite high. We discuss some available software for optimization in Section 8.4.

Literature in Optimization

Because of the differences in optimization problems resulting from differences in the properties of the functions and of their domains, the research literature in optimization is quite diverse. Some journals that emphasize optimization, especially the computational aspects, are *Journal of Optimization Theory and Applications*, *Journal of Global Optimization*, *Mathematical Programming, Optimization Methods and Software*, and *SIAM Journal on Optimization*. In addition, journals on numerical analysis, operations research, statistics or in various fields of application often contain articles on optimization.

There are also many books on various aspects of optimization.

The Next Chapters

We first consider some fundamental properties of functions (Chapter 2) and the solution of systems of equations (Chapter 3). Solution of systems of equations is a subtask in many algorithms for optimization.

In Chapters 4 through 7 we discuss the main topic of the book, that is, optimization. We first describe methods of optimization of continuous (and usually, twice-differentiable) functions; then we discuss optimization of graphs or functions over countable domains; next we consider the problem of optimization when the set of acceptable solutions is subject to constraints; and finally we consider problems in which there are multiple objective functions.

In Chapter 8 we discuss numerical methods in general and then software for optimization.

The final Chapter 9 returns to the general topic of the present introductory chapter, that is, applications in statistics. In that chapter, we consider some applications and the relevant optimization methodology in more detail.