7.3 Linear Regression Analysis

Most mathematical models in engineering and science are nonlinear in the parameters. However, for a complete understanding of nonlinear regression methods, it is necessary to develop first the linear regression case and show how this extends to nonlinear models.

The exact representation of a linear relationship may be shown as

$$y = \alpha + \beta x \tag{7.109}$$

where y represents the true value of the dependent variable, x is the true value of the independent variable, β is the slope of the line, and α is the y-intercept of the line. This deterministic relationship is not useful in this form because it requires knowledge of the true values of y and x. Instead, the linear model is rewritten in terms of the observations of the values of the variables

$$\boldsymbol{Y}^* = \alpha + \beta \boldsymbol{X} + \boldsymbol{u} \tag{7.110}$$

where $\boldsymbol{Y}^*$ is the vector of observations of the dependent variable, $\boldsymbol{X}$ is the vector of observations of the independent variable, and $\boldsymbol{u}$ is the vector of *disturbance terms*. The purpose of the $\boldsymbol{u}$ term is to characterize the discrepancies that emerge between the true values and the observed values of the variables. These discrepancies can be attributed mainly to experimental error. Later in this section, $\boldsymbol{u}$ will be assumed to be a stochastic variable with some specified probability distribution.

Eq. (7.110) can be extended to include more than one independent variable:

$$\boldsymbol{Y}^* = \beta_1 \boldsymbol{X}_1 + \beta_2 \boldsymbol{X}_2 + \ldots + \beta_k \boldsymbol{X}_k + \boldsymbol{u} \tag{7.111}$$

where $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_k$ are the vectors of observations of k independent variables. To allow for a y-intercept, the vector $\boldsymbol{X}_1$ can be taken as a vector whose components are all unity; thus, β_1 becomes the parameter specifying the value of the y-intercept.

The correlation between parameters causes the axes of the confidence ellipsoids of the linear model to be at an angle to the coordinates of the parameter space. Therefore, the individual parameter confidence limits will not represent the true interval within which a parameter b_i may lie and still remain within the confidence ellipsoid.

In nonlinear models, the confidence hyperspace is no longer a hyperellipsoid. The amount of distortion depends on the extent of the nonlinearity of the model. Therefore, the calculation of the confidence intervals is not as rigorous an exercise as in the linear model. Still, a lot of valuable information can be extracted from the correlation coefficient matrix that approximates the maximum-likelihood hyperspace in the vicinity of the solution where the model is nearly linear. If the absolute values of the off-diagonal elements of $\boldsymbol{R}$ are close to 1.0, the parameters associated with those elements are highly correlated with each other. Davies [6] tests the values of r_{ij} against a normal distribution with zero mean, that is, no correlation. He classifies the correlation as "significant" and "highly significant" if the value of r_{ij} is higher than the 0.05 and 0.01 significance points of the normal distribution, respectively. High correlation between parameters implies that it is very difficult to obtain separate estimates of these parameters with the available data.

The eigenvectors w of the matrix $\boldsymbol{R}$ give the direction of the major and minor axes of the hyperellipsoidal confidence region of the parameter space. The length of the axes are proportional to the square root of the eigenvalues λ of the matrix. Box [7] calculated the values of the parameters at the ends of the axes by

$$\bar{b}_i = b_i \pm w_{ri}\{\lambda_r(s^2 a_{ii}) k F_{(1-\alpha)}(k, n - k)\}^{1/2} \tag{7.156}$$

where $r = 1, 2, \ldots, k$

k = number of parameters

n = number of points used in estimating b_i

$F_{1\text{-}\alpha}(k, n - k)$ = value of the F distribution with k and $(n - k)$ degrees of freedom.

Subsequently, he uses these parameter values to calculate the sum of squares at each end of the axes and to compare them with the sum of squares at the center of the hyperellipsoid. This sum-of-squares search, which is based on a linear model, may give vital information for nonlinear models as well. In the case where the solution has only converged on a *local* minimum sum of squares, it is very likely that the search in the direction of one of the axes will produce a lower sum of squares. In such a case, the regression must be repeated, starting from a different initial position, so that the local minimum may be bypassed.

7.4 Nonlinear Regression Analysis

We have stated this earlier in the chapter, and we state it again: The mathematical models encountered in engineering and science are often nonlinear in their parameters. Consider, for example, the analysis of a chemical reaction such as

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C + D$$

$$C + A \xrightarrow{k_3} E + F$$

where the rate of formation of each component may be written as

$$\begin{aligned}
\frac{dC_A}{dt} &= -k_1 C_A - k_3 C_A^n C_C^m \\
\frac{dC_B}{dt} &= k_1 C_A - k_2 C_B \\
\frac{dC_C}{dt} &= k_2 C_B - k_3 C_A^n C_C^m \\
\frac{dC_E}{dt} &= k_3 C_A^n C_C^m
\end{aligned} \tag{7.157}$$

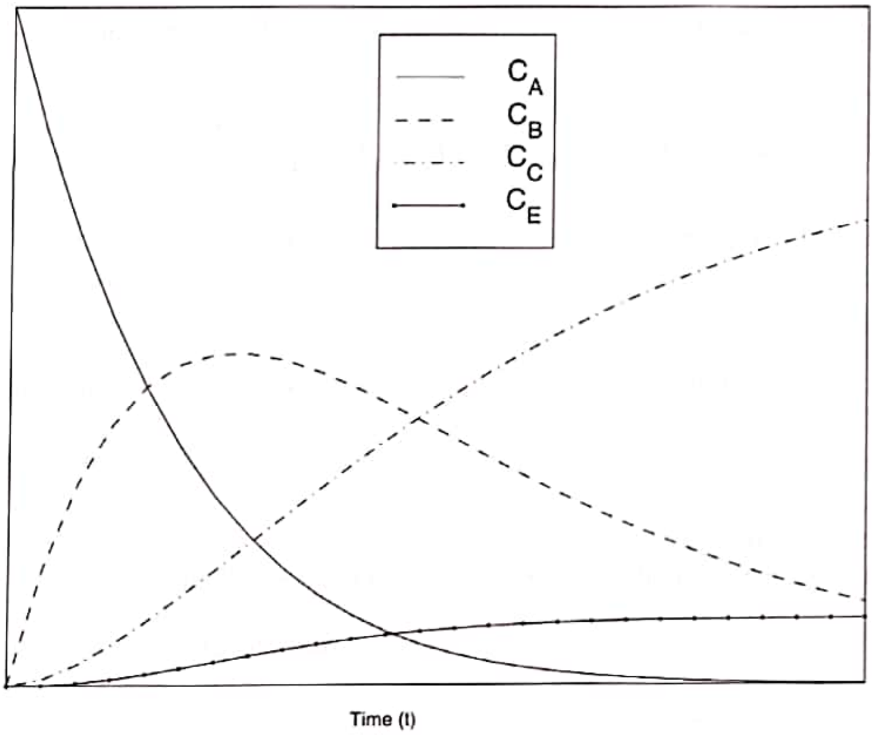

Figure 7.10 Simulated data for batch reactor experiment.

This is only one possible formulation of the reaction mechanism. It contains five unknown parameters, k_1, k_2, k_3, n, and m, which must be calculated by fitting the model to experimental data. Suppose that experiments for this chemical system are carried out in a batch reactor and data of the form shown in Fig. 7.10 are collected. Because experimental data are available for all four dependent variables, C_A, C_B, C_C, and C_E, multiple nonlinear regression can be performed by simultaneously fitting all four equations of (7.157) to the data.

A model consisting of differential equations, such as Eq. (7.157), may be shown in the form:

$$\frac{d\boldsymbol{Y}}{dx} = \boldsymbol{g}(x, \boldsymbol{Y}, \boldsymbol{b}) \tag{7.158}$$

where $d\boldsymbol{Y}/dx$ = vector of derivatives of $\boldsymbol{Y}$
$\boldsymbol{g}$ = vector of functions
x = independent variable
$\boldsymbol{Y}$ = vector of dependent variables
$\boldsymbol{b}$ = vector of parameters.

We assume that if the boundary conditions are given and if the vector $\boldsymbol{b}$ can be estimated, then the differential equations (7.158) can be integrated numerically or analytically to give the integrated results, which are

$$\boldsymbol{Y} = f(x, \boldsymbol{b}) \tag{7.159}$$

For the simple case where the model consists of only one dependent variable, the sum of squared residuals is given by

$$\Phi = \epsilon'\epsilon = (\boldsymbol{Y}^* - \boldsymbol{Y})'(\boldsymbol{Y}^* - \boldsymbol{Y}) \tag{7.160}$$

where $\boldsymbol{Y}^*$ = vector of experimental observations of the dependent variable
$\boldsymbol{Y}$ = vector of calculated values of the dependent variables obtained from Eq. (7.159).

There are several techniques for minimization of the sum of squared residuals described by Eq. (7.160). We review some of these methods in this section. The methods developed in this section will enable us to fit models consisting of multiple dependent variables, such as the one described earlier, to multiresponse experimental data, in order to obtain the values of the parameters of the model that minimize the *overall (weighted) sum of squared residuals*. In addition, a thorough statistical analysis of the regression results will enable us to

1. Decide whether the model gives satisfactory fit within the experimental error of the data.
2. Discriminate between competing models.

3. Measure the accuracy of the estimation of the parameters by constructing the confidence region in the parameter space.
4. Measure the correlation between parameters by examining the correlation coefficient matrix.
5. Perform tests to verify that repeated experimental data come from the same population of experiments.
6. Perform tests to verify whether the residuals between the data and the model are randomly distributed.

MATLAB does the single nonlinear regression calculation by applying the function *curvefit*, which comes with the *Optimization TOOLBOX* of MATLAB. The statement $\boldsymbol{b}$ = *curvefit*('*file_name*', $\boldsymbol{b}_0$, $\boldsymbol{x}$, $\boldsymbol{y}$) starts the regression calculations at the vector of initial guesses of the parameters $\boldsymbol{b}_0$ and uses the least squares technique to find the vector of parameters $\boldsymbol{b}$ that best fit the nonlinear expression, introduced in the MATLAB function *file_name.m*, to the data $\boldsymbol{y}$. Inputs to the function *file_name* should be the vector of parameters $\boldsymbol{b}$ and the vector of independent variable $\boldsymbol{x}$. The function *file_name* should return the vector of dependent variable $\boldsymbol{y}$. The default algorithm is Marquardt (see Sec. 7.4.4). A Gauss-Newton method (see Sec. 7.4.2) may be selected via the *options* input to the function.

7.4.1 The Method of Steepest Descent

A simple method, which has been used to arrive at the minimum sum of squares of a nonlinear model, is that of *steepest descent*. We know that the gradient of a scalar function is a vector that gives the direction of the greatest increase of the function at any point. In the steepest descent method, we take advantage of this property by moving in the opposite direction to reach a lower function value. Therefore, in this method, the initial vector of parameter estimates is corrected in the direction of the negative gradient of Φ:

$$\Delta \boldsymbol{b} = -K\left(\frac{\partial \Phi}{\partial \boldsymbol{b}}\right) \tag{7.161}$$

Where K is a suitable constant factor and $\Delta \boldsymbol{b}$ is the correction vector to be applied to the estimated value of $\boldsymbol{b}$ to obtain a new estimate of the parameter vector:

$$\boldsymbol{b}^{(m+1)} = \boldsymbol{b}^{(m)} + \Delta \boldsymbol{b} \tag{7.162}$$

where m is the iteration counter. Combining Eqs. (7.160) and (7.161) results in

$$\Delta \boldsymbol{b} = 2K\boldsymbol{J}'(\boldsymbol{Y}^* - \boldsymbol{Y}) \tag{7.163}$$

where $\boldsymbol{J}$ is the Jacobian matrix of partial derivatives of $\boldsymbol{Y}$ with respect to $\boldsymbol{b}$ evaluated at all n points where experimental observations are available:

$$J = \begin{bmatrix} \frac{\partial Y_1}{\partial b_1} & \cdots & \frac{\partial Y_1}{\partial b_k} \\ & \cdots\cdots\cdot & \\ \frac{\partial Y_n}{\partial b_1} & \cdots & \frac{\partial Y_n}{\partial b_k} \end{bmatrix} \tag{7.164}$$

The steepest descent method has the advantage that guarantees moving toward the minimum sum of squares without diverging, provided that the value of K, which determines the step size, is small enough. The value of K may be a constant throughout the calculations, changed arbitrarily at each calculation step, or obtained from optimization of the step size [8]. However, the rate of convergence to the minimum decreases as the search approaches this minimum, and the method loses its attractiveness because of this shortcoming.

7.4.2 The Gauss-Newton Method

Once again, we restate that in the least squares method, our objective is to find the vector of parameters $\boldsymbol{b}$ such that it minimizes the sum of squared residuals Φ. Thus, the vector $\boldsymbol{b}$ may be found by taking the partial derivative of Φ with respect to $\boldsymbol{b}$ and setting it to zero:

$$\frac{\partial \Phi}{\partial \boldsymbol{b}} = 0 \tag{7.165}$$

Because $\boldsymbol{Y}$ is nonlinear with respect to the parameters, Eq. (7.165) will yield a nonlinear equation that would be difficult to solve for $\boldsymbol{b}$. This problem was alleviated by Gauss, who determined that fitting nonlinear functions by least squares can be achieved by an iterative method involving a series of linear approximations. At each stage of the iteration, linear squares theory can be used to obtain the next approximation.

This method, known as the *Gauss-Newton method*, converts the nonlinear problem into a linear one by approximating the function $\boldsymbol{Y}$ by a Taylor series expansion around an estimated value of the parameter vector $\boldsymbol{b}$:

$$\boldsymbol{Y}(x,\boldsymbol{b}) = \boldsymbol{Y}(x,\boldsymbol{b}^{(m)} + \Delta\boldsymbol{b}) = \boldsymbol{Y}(x,\boldsymbol{b}^{(m)}) + \frac{\partial \boldsymbol{Y}}{\partial \boldsymbol{b}}\Big|_{\boldsymbol{b}^{(m)}}\Delta\boldsymbol{b} = \boldsymbol{Y} + \boldsymbol{J}\Delta\boldsymbol{b} \tag{7.166}$$

where the Taylor series has been truncated after the second term. Eq. (7.166) is linear in $\Delta\boldsymbol{b}$. Therefore, the problem has been transformed from finding $\boldsymbol{b}$ to that of finding the correction to $\boldsymbol{b}$, that is, $\Delta\boldsymbol{b}$, which must be added to an estimate of $\boldsymbol{b}$ to minimize the sum of squared residuals. To do this we replace $\boldsymbol{Y}$ in Eq. (7.160) with the right-hand side of Eq. (7.166) to get

$$\Phi = (\boldsymbol{Y}^* - \boldsymbol{Y} - \boldsymbol{J}\Delta\boldsymbol{b})'(\boldsymbol{Y}^* - \boldsymbol{Y} - \boldsymbol{J}\Delta\boldsymbol{b}) \tag{7.167}$$

Taking the partial derivative of Φ with respect to $\Delta \boldsymbol{b}$, setting it equal to zero, and solving for $\Delta \boldsymbol{b}$, we obtain

$$\Delta \boldsymbol{b} = (\boldsymbol{J}'\boldsymbol{J})^{-1}\boldsymbol{J}'(\boldsymbol{Y}^{*} - \boldsymbol{Y}) \tag{7.168}$$

The Gauss-Newton method applies to both the one-variable model and the multiple regression case (see Sec. 7.4.5). The algorithm of the Gauss-Newton method involves the following steps:

2. Assume initial guesses for the parameter vector $\boldsymbol{b}$.
3. If the model is in the form of differential equation(s), then use the vector $\boldsymbol{b}$ and the boundary condition(s) to integrate the equation(s) to obtain the profile(s) of $\boldsymbol{Y}$. If the model is in the form of algebraic equation(s), then simply use the vector $\boldsymbol{b}$ to evaluate $\boldsymbol{Y}$ from the equation(s).
4. Evaluate the Jacobian matrix $\boldsymbol{J}$ from the equation(s) of the model.
5. Use Eq. (7.168) to obtain the correction vector $\Delta \boldsymbol{b}$.
6. Evaluate the new estimate of the parameter vector from Eq. (7.162):

$$\boldsymbol{b}^{(m+1)} = \boldsymbol{b}^{(m)} + \Delta \boldsymbol{b} \tag{7.162}$$

7. It is also possible to apply the relaxation factor in order to prevent the calculation from diverging (see Sec. 1.8).
8. Repeat steps 2-5 until either (or both) of the following conditions are satisfied:
   a. Φ does not change appreciably.
   b. $\Delta \boldsymbol{b}$ becomes very small.

The Gauss-Newton method is based on the linearization of a nonlinear model; therefore, this method is expected to work well if the model is not highly nonlinear, or if the initial estimate of the parameter vector is near the minimum sum squares. The contours of constant Φ in the parameter space of a linear model are ellipsoids (Fig. 7.11*a*). For a nonlinear model, these contours are distorted (Fig. 7.11*b*), but in the vicinity of the minimum Φ, the contours are very nearly elliptical. Therefore, the Gauss-Newton method is quite effective if the initial starting point for the search is in the nearly elliptical region. On the other hand, this method may diverge if the starting point is in the highly distorted region of the parameter hyperspace.

7.4.3 Newton's Method

Eq. (7.165) represents a set of nonlinear equations; therefore, Newton's method may be applied to solve this set of nonlinear equations. First, let us expand Φ by Taylor series up to the third term:

$$\Phi(x, \boldsymbol{b}) = \Phi(x, \boldsymbol{b}^{(m)}) + \left(\frac{\partial \Phi}{\partial \boldsymbol{b}}\right)^{(m)} \Delta \boldsymbol{b} + \frac{1}{2} \Delta \boldsymbol{b}' \left(\frac{\partial^2 \Phi}{\partial \boldsymbol{b}^2}\right)^{(m)} \Delta \boldsymbol{b} \tag{7.169}$$

Taking the partial derivative of both sides of Eq. (7.169) with respect to ***b*** gives

$$\frac{\partial \Phi}{\partial \boldsymbol{b}} = \left(\frac{\partial \Phi}{\partial \boldsymbol{b}}\right)^{(m)} + \left(\frac{\partial^2 \Phi}{\partial \boldsymbol{b}^2}\right)^{(m)} \Delta \boldsymbol{b} \tag{7.170}$$

The first derivative of Φ with respect to ***b*** can be calculated by differentiating Eq. (7.160):

$$\left(\frac{\partial \Phi}{\partial \boldsymbol{b}}\right)^{(m)} = -2\boldsymbol{J}'(\boldsymbol{Y}^* - \boldsymbol{Y}) \tag{7.171}$$

and the second derivative of Φ with respect to ***b*** is called the *Hessian matrix* of the second-order partial derivatives of Φ with respect to ***b*** evaluated at all n points where experimental observations are available:

$$\boldsymbol{H} = \begin{bmatrix} \frac{\partial^2 \Phi}{\partial b_1^2} & \frac{\partial^2 \Phi}{\partial b_1 \partial b_2} & \cdots & \frac{\partial^2 \Phi}{\partial b_1 \partial b_k} \\ \frac{\partial^2 \Phi}{\partial b_2 \partial b_1} & \frac{\partial^2 \Phi}{\partial b_2^2} & \cdots & \frac{\partial^2 \Phi}{\partial b_2 \partial b_k} \\ & \cdots & \cdots & \\ \frac{\partial^2 \Phi}{\partial b_k \partial b_1} & \frac{\partial^2 \Phi}{\partial b_k \partial b_2} & \cdots & \frac{\partial^2 \Phi}{\partial b_k^2} \end{bmatrix} \tag{7.172}$$

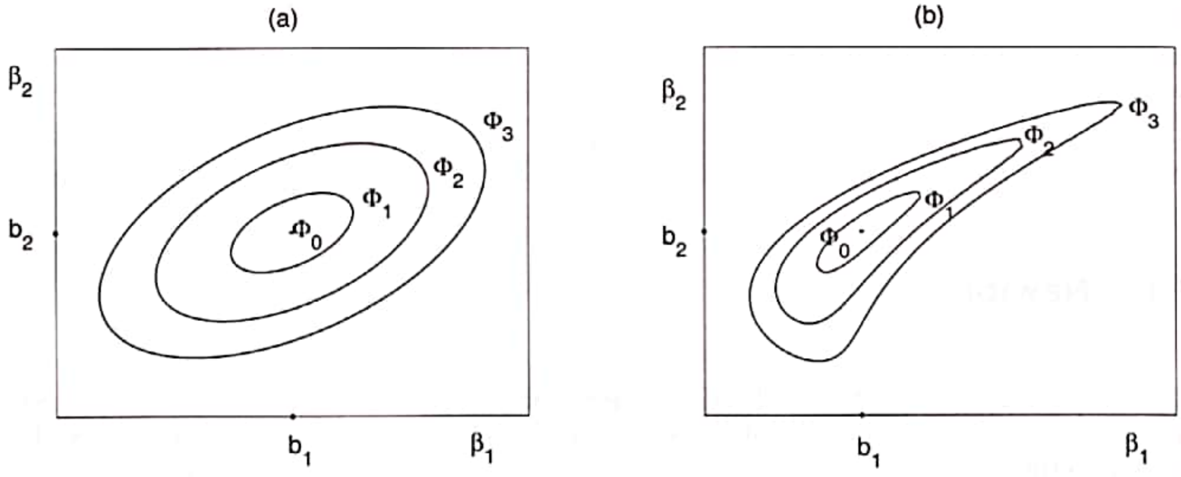

Figure 7.11 Contours for constant sum of squares in parameter space. (*a*) Linear model. (*b*) Nonlinear model.

By applying the necessary condition of having a local minimum of Φ, Eq. (7.165), into Eq. (7.170) and combining with Eqs. (7.171) and (7.172), we can evaluate the correction vector $\Delta \boldsymbol{b}$:

$$\Delta \boldsymbol{b} = 2\boldsymbol{H}^{-1}\boldsymbol{J}'(\boldsymbol{Y}^{*} - \boldsymbol{Y}) \tag{7.173}$$

It is interesting to note that in the case of single parameter regression, Eq. (7.165) becomes

$$\frac{d\Phi}{db} = \Phi' = 0$$

and Eq. (7.173) simplifies to

$$\Delta b = -\frac{\Phi'^{(m)}}{\Phi''^{(m)}}$$

which is the Newton-Raphson solution of the nonlinear equation $\Phi' = 0$.

The calculation procedure for Newton's method is almost the same as that of Gauss-Newton method with the exception that the vector of corrections to the parameters is calculated from Eq. (7.173). If Φ is quadratic with respect to $\boldsymbol{b}$ (that is, linear regression), Newton's method converges in only one step. Like all other methods applying Newton's technique for the solution of the set of nonlinear equations, a relaxation factor may be used along with Eq. (7.173) when correcting the parameters.

7.4.4 The Marquardt Method[6]

Marquardt [9] has developed an interpolation technique between the Gauss-Newton and the steepest descent methods. This interpolation is achieved by adding the diagonal matrix $(\lambda \boldsymbol{I})$ to the matrix $(\boldsymbol{J}'\boldsymbol{J})$ in Eq. (7.168):

$$\Delta \boldsymbol{b} = (\boldsymbol{J}'\boldsymbol{J} + \lambda \boldsymbol{I})^{-1}\boldsymbol{J}'(\boldsymbol{Y}^{*} - \boldsymbol{Y}) \tag{7.174}$$

The value of λ is chosen, at each iteration, so that the corrected parameter vector will result in a lower sum of squares in the following iteration. It can be easily seen that when the value of λ is small in comparison with the elements of matrix $(\boldsymbol{J}'\boldsymbol{J})$, the Marquardt method approaches the Gauss-Newton method; when λ is very large, this method is identical to steepest descent, with the exception of a scale factor that does not affect the direction of the parameter correction vector but that gives a small step size.

According to Marquardt, it is desired to minimize Φ in the maximum neighborhood over which the linearized function will give adequate representation of the nonlinear function.

[6] Also known as the Levenberg-Marquardt method.

Therefore, the method for choosing λ must give small values of λ when the Gauss-Newton method would converge efficiently and large values of λ when the steepest descent method is necessary.

The Marquardt method may likewise be applied to Newton's method. In this case, the diagonal matrix $\lambda \boldsymbol{I}$ is added to the Hessian matrix in Eq. (7.173):

$$\Delta \boldsymbol{b} = 2(\boldsymbol{H} + \lambda \boldsymbol{I})^{-1}\boldsymbol{J}'(\boldsymbol{Y}^* - \boldsymbol{Y}) \tag{7.175}$$

The Marquardt method consists of the following steps:

1. Assume initial guesses for the parameter vector $\boldsymbol{b}$.
2. Assign a large value, say 1000, to λ. This means that in the first iteration the steepest descent method is predominant and would assure that the method is moving toward the lower sum of squared residuals.
3. Evaluate the Jacobian matrix $\boldsymbol{J}$ from the equation(s) of the model. Also evaluate the Hessian matrix $\boldsymbol{H}$ if using Newton's method.
4. Use either Eq. (7.174) or Eq. (7.175) to obtain the correction vector $\Delta\boldsymbol{b}$.
5. Evaluate the new estimate of the parameter vector from Eq. (7.162).

$$\boldsymbol{b}^{(m+1)} = \boldsymbol{b}^{(m)} + \Delta \boldsymbol{b} \tag{7.162}$$

6. Calculate the new value of Φ. If $\Phi^{(m+1)} < \Phi^{(m)}$, reduce the value of λ, by a factor of 4, for example. If $\Phi^{(m+1)} > \Phi^{(m)}$, keep the old parameters [$\boldsymbol{b}^{(m+1)} = \boldsymbol{b}^{(m)}$] and increase the value of λ, by a factor of 2, for example.
7. Repeat steps 3-6 until either (or both) of the following conditions are satisfied:
   a. Φ does not change appreciably.
   b. $\Delta\boldsymbol{b}$ becomes very small.

7.4.5 Multiple Nonlinear Regression

In the previous four sections, the sum of squared residuals that was minimized was that given by Eq. (7.160). This was the sum of squared residuals determined from fitting one equation to measurements of one variable. However, most mathematical models may involve simultaneous equations in multiple dependent variables. For such a case, when more than one equation is fitted to multiresponse data, where there are v dependent variables in the model, the *weighted sum of squared residuals* is given by

$$\Phi = \sum_{j=1}^{v} w_j \boldsymbol{\epsilon}_j{}'\boldsymbol{\epsilon}_j = \sum_{j=1}^{v} w_j \phi_j$$

$$= \sum_{j=1}^{v} w_j(\boldsymbol{Y}_j^* - \boldsymbol{Y}_j)'(\boldsymbol{Y}_j^* - \boldsymbol{Y}_j) \tag{7.176}$$

where w_j = weighting factor corresponding to the jth dependent variable
ϕ_j = sum of squared residuals corresponding to the jth dependent variable.

To minimize Φ by the Gauss-Newton method, we first linearize the models using Eq. (7.166) and combine with Eq. (7.176) to obtain

$$\Phi = \sum_{j=1}^{v} w_j(Y_j^* - Y_j - J_j\Delta b)'(Y_j^* - Y_j - J_j\Delta b) \tag{7.177}$$

Taking the partial derivative of Φ with respect to Δb, setting it equal to zero, and solving for Δb we obtain

$$\Delta b = \left[\sum_{j=1}^{v} w_j(J_j'J_j)\right]^{-1}\left[\sum_{j=1}^{v} w_jJ_j'(Y_j^* - Y_j)\right] \tag{7.178}$$

Eq. (7.178) gives the correction of the parameter vector when fitting multiple dependent variables simultaneously. Eq. (7.178) becomes identical to Eq. (7.168) when $v = 1$, that is, when only one dependent variable is fitted. When using the Marquardt method, the correction of the parameter vector is calculated from

$$\Delta b = \left[\lambda I + \sum_{j-1}^{v} w_j(J_j'J_j)\right]^{-1}\left[\sum_{j=1}^{v} w_jJ_j'(Y_j^* - Y_j)\right] \tag{7.179}$$

The weighting factors w_j are determined as follows: The basic assumption in the derivation of the regression algorithm was that the variance σ^2 of the distribution of the error in the measurements was constant throughout the profile of a single dependent variable. However, in the case of multiple regression, it is very unlikely that the variance σ_j^2 of all the curves will be the same. Therefore, in order to form an unbiased weighted sum of squared residuals, the individual sum of squares must be multiplied by a weighting factor that is proportional to $1/\sigma_j^2$. The equation for evaluating the weighting factors is given by

$$w_j = \frac{1/\sigma_j^2}{\dfrac{1}{\sum_{i=1}^{v} n_i}\left[\sum_{i=1}^{v}\sum_{l=1}^{n_i}\dfrac{1}{\sigma_i^2}\right]} \tag{7.180}$$

where σ_j^2 or σ_i^2 = variance for each curve
n_i = number of experimental points available for each curve
v = number of variables being fitted.

The denominator of Eq. (7.180) accounts for the possibility that each curve may have a different number of experimental points n_i and weighs that accordingly. If the assumption that σ_j^2 is constant within one curve does not hold, then Eq. (7.180) can be extended so that weighting factor can be calculated at each point with the appropriate value of σ_j^2.

In most cases, the values of σ_j^2 would not be known; however, the estimates of these variances s_j^2 can be obtained from repeated experiments, and the values of s_j^2 are then used in Eq. (7.180) to calculate the weighting factors. In the worst case, where no repeated experiments are made and no *a priori* knowledge of σ_j^2 is available, then the values of w_j must be guessed. Otherwise, the nonlinear regression algorithm would introduce a bias toward fitting more satisfactorily the curve with the highest ϕ_j and partially ignoring the curves with low ϕ_j.

The nonlinear regression can also be extended to fit multiple experimental values of the dependent variable at each value of the independent variable. This can be done by changing Eq. (7.176) so that the squared residuals are also summed up within each group of points. Finally, if the value of the variance of the error is proportional to the value of the dependent variable, the residual in the sum-of-squares calculation must be divided by the theoretical (calculated) value of the dependent variable at each point in the calculation.