Chapter 3

Response Surface Approximations

Most optimization algorithms that are in use for solving analytical engineering optimization problems are sequential in nature. That is, the objective function and constraints are evaluated at one point at a time, and the values at that point, as well as previous design point contribute to a decision on where in design space to move to for the next evaluation.

When the objective functions and (or) the constraints are evaluated by experiments rather than by analytical evaluations, there is usually an incentive to perform the experiments in batches rather than singly. One reason for batching experiments is that most require set-up time, advance planning and reservations of experimental facilities or technicians. Another reason is that experimental errors make it difficult to interpret the results of a single experiment. When a batch of experiments is performed, errors in one or two experiments tend to stand out. Duplicating experiments for identical nominal conditions permits us to estimate the magnitude of experimental scatter due to errors and variability in the properties of the tested designs. Finally, some of the experimental scatter can be averaged out by performing a large number of experiments.

Because of these advantages of running experiments in batches, experimental optimization has followed a different route than analytical optimization. The standard approach is to use an optimization strategy that is based on the results of a batch of experiments. On the basis of the experiments, we construct approximations to objective functions and/or constraints and perform optimization on the basis of these approximations. In most cases, the optimum obtained is then tested, and if satisfactory results are obtained the design procedure is terminated. In some cases, the optimum is used as the central design point for a new batch of experiments, and the process is repeated once or twice. This process is sometimes called sequential approximate optimization. However, because of the cost and time associated with conducting experiments, it is rare that the process is iterated to convergence.

When analytical calculations were mostly based on closed form solutions or numerical models that required minimal modeling and computations, the difference between analysis and experiments was very clear. However, today numerical evaluations of objective functions and constraints often share many of the properties of experimental evaluations. First, numerical models such as finite element structural models require substantial investment of time to set up and debug. Furthermore, the evaluation of such models may require large computational resources, so that the cost of numerical simulation may be comparable to the cost of experiments. Second, with analytical simulations based on complex numerical models, many sources of noise are often found in the results of numerical simulations. These includes round-off errors as well as errors due to incomplete convergence of iterative processes. Additionally,
numerical simulations are usually based on discretization of continua, and the accuracy of this discretization depends on the shape of the domains being discretized. For example, when stress analysis is performed in an elastic body using finite-element discretization, the discretization error does not change smoothly with the shape of the body, because the number of finite elements can vary only in integer increments.

These growing similarities between analytical simulations and experiments create incentives to run analytical simulations in batches and use sequential approximate optimization. Additionally, the growing availability of parallel computers also provides incentives for running analytical simulations in batches. Finally, numerical simulations are often run with software packages that are difficult to connect directly to optimization programs. Approximate sequential optimization provides a mechanism for running these software packages in a stand-alone mode and connecting the optimization programs to the approximation.

3.1 Fitting an Approximation to Given Data

Selecting the points in design space where experiments are to be performed is possibly the most important part of obtaining good approximations to system response quantities that may be used in computing objective functions and constraints. However, this point selection turns to be a difficult optimization problem in itself, and will be discussed in the next chapter. In this section we will address the more modest question of how to obtain the best approximation to a given a set of experiments.

We denote the response function that we want to approximate as $y$, and assume that it can be approximated as a function of the design variable vector $x$ and a vector of $n_\beta$ parameters $\beta$, that is

$$y = \hat{y}(x, \beta) + \epsilon$$  \hspace{1cm} (3.1.1)

where $\hat{y}$ represents the approximation, and $\epsilon$ the error. We have the data from $n_y$ experiments at $n_y$ design points $x_i$

$$y_i = y(x_i, \beta) + \epsilon_i$$  \hspace{1cm} (3.1.2)

and we want to find the parameter vector $\beta$ that will best fit the experimental evidence.

In many non-design applications, $\hat{y}$ represents a numerical model such as a finite-element model, and $\beta$ represents a set of physical parameters such as masses and stiffnesses. The process of finding the values of the physical parameters $\beta$ that best fit the experimental results is called system identification.

In design applications, the vector of parameters $\beta$ often does not have any physical meaning. Rather, we select some functional representation for $\hat{y}$, with $\beta$ representing some coefficient to be determined so as to fit the data well. For example, with a single design variable $x$, a linear approximation is of the form

$$y = \beta_1 + \beta_2 x$$  \hspace{1cm} (3.1.3)

while a rational approximation may take the form

$$y = \frac{\beta_1}{x + \beta_2}$$  \hspace{1cm} (3.1.4)
The process of finding \( \beta \) to best fit the data is called regression, and \( \hat{y} \) is called a response surface. The most commonly used measure of the error in the approximation is the root-mean-square (rms) error
\[
e_{\text{rms}} = \sqrt{\frac{1}{n_y} \sum_{i=1}^{n_y} (y_i - \hat{y}(x_i, \beta))^2}.
\]
(3.1.5)

Other measures that are often used are the average absolute error
\[
e_{av} = \frac{1}{n_y} \sum_{i=1}^{n_y} |y_i - \hat{y}(x_i, \beta)|,
\]
and the maximum error
\[
e_{\text{max}} = \max_i |y_i - \hat{y}(x_i, \beta)|.
\]
(3.1.6)

(3.1.7)

The popularity of \( e_{\text{rms}} \) as a measure of the error is probably due to the fact that it is often easier to estimate the \( \beta \) that minimizes \( e_{\text{rms}} \) than the \( \beta \) that minimizes the other two measures. However, there is also theoretical justification for using it. If the form of the model \( \hat{y} \) is exact (that is the true function is of that form) and the error \( \epsilon \) is normally distributed, with no correlation between data points and all having the same standard deviation, then minimizing \( e_{\text{rms}} \) will provide the best linear unbiased estimate (BLUE) of \( \beta \).

The value of \( \beta \) that would be found if we had an infinite number of experiments is sometimes thought of as the `true' value of the parameter vector. With a finite number of experiments we obtain only an estimate of this 'true' value, which we denote as \( \mathbf{b} \). Estimating the value of \( \mathbf{b} \) that minimizes the rms error is easy for cases where \( \hat{y} \) is a linear function of \( \beta \), that is
\[
y = \sum_{i=1}^{n_{\beta}} \beta_i \xi_i(x),
\]
(3.1.8)

Where \( \xi_i(x) \) are given shape functions, usually monomials. For example, for the linear approximation, Eq. 3.1.3
\[
\xi_1 = 1, \quad \xi_2 = x.
\]
(3.1.9)

Let us denote as \( \mathbf{b} \) an estimate to \( \beta \) that minimizes \( e_{\text{rms}} \) for the given \( n_y \) experiments. The difference between the model and the \( j \)th experiment, \( e_j \), is
\[
e_j = y_j - \sum_{i=1}^{n_{\beta}} b_i \xi_i(x_j),
\]
(3.1.10)

or in vector form,
\[
e = y - X\mathbf{b}.
\]
(3.1.11)

Note that the \((i,j)\) component of the matrix \( X \) is \( \xi_j(x_i) \). The rms error is the rms average of the components of \( e \)
\[
e_{\text{rms}} = \sqrt{\frac{1}{n_y} \sum_{i=1}^{n_y} e_i^2} = \sqrt{\frac{1}{n_y} e^T e},
\]
(3.1.12)

So that minimizing \( e_{\text{rms}} \) is equivalent to minimizing \( e^T e \). Using Eq. 3.1.11 we have
\[
e^T e = (y - X\mathbf{b})^T (y - X\mathbf{b}) = y^T y - y^T X\mathbf{b} - \mathbf{b}^T X^T y + \mathbf{b}^T X^T X\mathbf{b}.
\]
(3.1.13)

To find the vector $\mathbf{b}$ that minimizes $e_{\text{rms}}$, we differentiate $\mathbf{e}^T \mathbf{e}$ with respect to the components of $\mathbf{b}$ and set the derivatives to zero. It is not difficult to check that we get

$$X^T \mathbf{X} \mathbf{b} = X^T \mathbf{y}.$$  

(3.1.14)

Equation 3.1.14 is an $n_\beta \times n_\beta$ system of equations called the normal equation, and we can write its solution of this equation as

$$\mathbf{b} = (X^T X)^{-1} X^T \mathbf{y}. \quad (3.1.15)$$

However, the normal equation is often ill-conditioned, especially when $n_\beta$ is high. To avoid some of the effects of the ill conditioning we can formulate the problem in a slightly different form. Consider the system

$$\mathbf{X} \mathbf{b} = \mathbf{y}. \quad (3.1.16)$$

If we could solve this equation exactly, then from Eq. 3.1.13 we see that the rms error will be zero. However, this system has $n_y$ equations for the $n_\beta$ unknowns, with $n_y$ in general larger than $n_\beta$, so that in general, we cannot find an exact solution. That is, any vector $\mathbf{b}$ will not satisfy Eq. 3.1.16 exactly, but instead there will be a vector of residuals (differences between the left side and the right side of the equation). The solution of the normal equation is the vector $\mathbf{b}$ that minimizes the sum of the squares of the residuals. However, instead of solving the normal equations, there are numerical methods, such as the QR decomposition, that solves for Eq. 3.1.16 directly for the least square solution, and these are usually more numerically stable than numerical solutions of the normal equations. To improve numerical stability, it is also recommended to translate and scale all the variables so that each changes in the range (-1,1).

The vector of errors left after the least-squares fit is denoted as $\mathbf{e}_r$, and the sum of the squares of the errors remaining after the fit is denoted as $SS_e$. With a bit of algebra we can show that

$$SS_e = \mathbf{e}_r^T \mathbf{e}_r = \mathbf{y}^T \mathbf{y} - \mathbf{b}^T X^T \mathbf{y}. \quad (3.1.17)$$
Example 3.1.1

Fit a straight line \( y = b_0 + b_1 x \) to the following three measurements of \( y \): \( y(0) = 0 \), \( y(1) = 1 \), \( y(2) = 0 \). Calculate first a least square fit, and then compare to a fit that will minimize the maximum error, and to one that will minimize the average absolute error.

If we write the three equations that we would like to satisfy for the three points, Eq. 3.1.16 we get

\[
\begin{align*}
y(0) &= b_0 = 0 \\
y(1) &= b_0 + b_1 = 1 \\
y(2) &= b_0 + 2b_1 = 0
\end{align*}
\]  
(3.1.18)

So that the matrix \( X \) and the vector \( y \) are given as

\[
X = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix}, \quad y = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}
\]  
(3.1.19)

So that

\[
X^T X = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \\ 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 & 3 \\ 3 & 5 \end{bmatrix},
\]  
(3.1.20)

And

\[
X^T y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}
\]  
(3.1.21)

We can now write the normal equation, \( X^T X \mathbf{b} = X^T y \),

\[
\begin{align*}
3b_0 + 3b_1 &= 1 \\
3b_0 + 5b_1 &= 1
\end{align*}
\]

and solve it to obtain \( b_0 = 1/3 \) and \( b_1 = 0 \). So that our least-square fit is the line \( y = 1/3 \). With two points at \( y = 0 \) and one point at \( y = 1 \), this seems to be a sensible choice. The errors (residuals) at the three points are \( e_1 = 1/3, e_2 = 2/3, \) and \( e_3 = 1/3 \). So the sum of the squares of the errors is 2/3, and the rms error is

\[
e_{rms} = \sqrt{\frac{1}{3} \left[ \left( \frac{1}{3} \right)^2 + \left( \frac{2}{3} \right)^2 + \left( \frac{1}{3} \right)^2 \right]} = 0.47.
\]  
(3.1.22)

We can obtain the same result also from Eq. 3.1.17 since

\[
y^T y = 1, \quad \mathbf{b}^T = \left[ \frac{1}{3}, 0 \right], \quad \mathbf{b}^T X^T y = 1/3,
\]  
(3.1.23)

then

\[
SS_c = 1 - 1/3 = 2/3, \quad e_{rms} = \sqrt{SS_c / 3} = 0.47.
\]  
(3.1.24)

By comparison, the average and maximum errors are
In general, it is difficult to find the response surface that minimizes the maximum error and the response surface that minimizes the average error. However, for this simple example this can be easily done. We expect that like the line that minimizes the rms error, the lines that minimize the other two errors would be horizontal lines of the form \( y = c \), \( 0 \leq c \leq 1 \). To minimize the maximum error, it is obvious that we must have \( c = 0.5 \), which results in a maximum error \( e_{\text{max}} = 0.5 \). In this case this is also the average error and the rms error because all three points have the same 0.5 error. To minimize the average absolute error, we note that for the range \( c \in [0,1] \) the average absolute error will be

\[
e_{\text{av}} = \frac{1}{3} \left[ c + (1 - c) + c \right] = \frac{1 + c}{3},
\]

so that the minimum average error is achieved for \( c = 0 \). In that case the line is given as \( y = 0 \), the average error is the smallest at 1/3, but \( e_{\text{rms}} = \sqrt{1/3} = 0.577 \), and \( e_{\text{max}} = 1 \). The three cases are shown in Fig. 3.1.

For more complex examples, we should use computer software to perform the process of fitting the response surface to the data. The following example shows results for the soft-drink-can problem.

**Example 3.1.2**

**Soft-Drink Can Design:** A soft-drink company has a new drink that they plan to market, and their preliminary research has determined that the cost to produce and distribute a cylindrical can is approximately given as

\[
C = 0.8V_0 + 0.1S,
\]

where \( C \) is the cost in cents, \( V_0 \) is the volume in fluid ounces, and \( S \) is the surface area of the can in square inches. They also determined that they can sell the drink in cans ranging from 5 to 15 ounces, and then the price \( P \) in cents that can be charged for a can is estimated as

\[
P = 2.5V_0 - 0.02V_0^2.
\]

Based on their past experience they will consider only a can with diameter \( D \) between 1.5 and 3.5 inches, and their market research has shown that soft-drink cans have to have an aspect ratio of at least 2.0 to be easy to drink from. That is, the height \( H \) of the can has to be at least twice the diameter.

We express the volume and the surface area in terms of the diameter and the height of the can

\[
V_0 = 0.25\pi D^3 / 1.8, \quad S = 2(0.25\pi D^2) + \pi DH,
\]

where the 1.8 factor is used to convert from cubic inches to fluid ounces. The profit per can, \( p_c \), may be then written as

\[
p_c = P - C = 0.2361\pi D^2 H - 3.858 \times 10^{-4} \pi^2 D^4 H^2 - 0.1\pi(0.5D^2 + DH),
\]
Field tests are conducted and resulted in the following 9 data points, shown as (diameter, height, profit):

(1.8, 3.6, 5.9), (2.4, 4.8, 13.1), (3.6, 22.), (1.5, 4.6, 4.7), (2.1, 5.8, 12.0)
(2.7, 7., 20.9), (1.4, 5.6, 4.9), (2., 6.8, 12.5), (2.6, 8., 21.4)

We will assume that the equations are more accurate than the field test results (how can that be possible?). From the equations, the corresponding triads are

1.8, 3.6, 5.5899), (2.4, 4.8, 13.074), (3.6, 21.883), (1.5, 4.6, 4.748), (2.1, 5.8, 11.963)
(2.7, 7., 20.854), (1.4, 5.6, 4.912), (2., 6.8, 12.458), (2.6, 8., 21.382)

Fit a quadratic polynomial to the profit per can using the experimental 9 data points.

The quadratic polynomial will be of the form

\[ p_c = b_1 + b_2 D + b_3 H + b_4 D^2 + b_5 DH + b_6 H^2. \]  

So that from the data we will get 9 equations for the six coefficients. The first equation obtained for the experimental data is

\[ b_1 + 1.8b_2 + 3.6b_3 + 3.24b_4 + 6.48b_5 + 12.96b_6 = 5.9, \]

and the last one is

\[ b_1 + 2.6b_2 + 8b_3 + 6.76b_4 + 20.8b_5 + 64b_6 = 21.4, \]

Solving the normal equation and substituting the coefficients into Eq. 3.1.27 we get

\[ p_c = -7.18 + 2.88D + 0.302H + 0.358D^2 + 1.00DH - 0.0707H^2. \]

In comparison, if we use the 'true' values for the profit, we get

\[ p_c = -7.98 + 2.57D + 0.675H + 0.296D^2 + 1.09DH - 0.115H^2. \]
Note that seven of the nine coefficients are fairly close, within about 10 percent of one another. However, the coefficient of $H$ and of $H^2$ are quite different, indicating that these coefficients are less important for fitting the data. This does not mean, however, that these coefficients may not

![Figure 3.2: Profit per can contours from quadratic response surface based on experimental values](image)
affect predictions at other points beside the data points. We are thus warned that these coefficients may need special treatment. The profit-per-can contours based on the experimental data are shown in Fig. 3.2, and the ones based on the 'true' values are shown in Fig. 3.3.
3.2 Estimating the Accuracy of the Response Surface

In most applications, the response surface approximation we construct on the basis of given data is intended for prediction of response at other design points, typically for improving the design. Therefore the ultimate test of the response surface is how well it predicts the response at other points of interest. However, if the response surface does not approximate the response well even at the data points, we cannot expect it to approximate well other points in design space.

The most immediate measures of the accuracy of the fit to the data are the various errors discussed earlier, the rms error, the average absolute error and the maximum error. We denote as \( \hat{y} \) the estimate of the response surface for the response, that is (see Eq. 3.1.8)

\[
y = \sum_{i=1}^{n_\beta} b_i \xi_i(x),
\]

and use \( \mathbf{y} \) to denote the values of the response surface at the data points and \( \mathbf{e} \) the error vector at these points (vector of residuals). It is easy to check that then

\[
\mathbf{y} = \mathbf{X}\mathbf{b}, \quad \mathbf{e} = \hat{\mathbf{y}} - \mathbf{y}.
\]

The average and maximum errors are then

\[
e_{av} = \frac{1}{n_y} \sum_{i=1}^{n_y} |e_i|, \quad e_{max} = \max e_i.
\]

The sum of the squares of the remaining errors, \( SS_e \) is given by Eq. 3.1.17, and we can calculate from it the rms error, \( e_{rms} \) as

\[
e_{rms}^2 = \frac{SS_e}{n_y}.
\]

However, this calculation of rms error is quite misleading if we wanted to use it to assess the accuracy of the response surface. This becomes clear if we note that if the number of data points \( n_y \) is equal to the number of coefficients, \( n_\beta \), then the response surface will pass through the data points, and the error will be zero. We certainly do not expect that the error will be zero at other points, not included in the data. In fact, fitting a response surface to a number of points equal to the number of coefficients (known as saturated design) is known to often lead to poor approximation, especially when there is noise in the data.

An impressive body of theoretical work has been done for the case where the noise in the data is random with normal distribution with zero mean and standard deviation of \( \sigma \), and where the noise at one point is uncorrelated with the noise at other data points (e.g., Myers and Montgomery, 1995).

On the basis of these assumptions we can estimate the standard deviation of the noise in the measurements, which also serves as an estimate of \( e_{rms} \) at all the points in the region of interest. An unbiased estimate \( \hat{\sigma} \) for this standard deviation is given as

\[
\hat{\sigma}^2 = \frac{SS_e}{n_y - n_\beta},
\]
and is known as the standard error. If this estimate of the rms error is larger than we can tolerate for predicting values of the response at candidate design points, we conclude that our response surface is inadequate. In this case we must change the form of the response surface to try to fit the data better.

The simplest approach is to add terms to the polynomial approximation. If we used a linear polynomial to start with, we may want to go to a quadratic. If we used a quadratic, we may want to use a cubic.

Unfortunately, while we can always improve the fit to the data by increasing the number of terms in the response surface, it is not clear that these gains will translate into gains in predicting the response surface at other points. As we add coefficients, we run the danger of ‘overfitting’ the data. This danger is particularly acute when the data contains substantial amount of noise. As we increase the number of coefficients we may be fitting the noise rather than the underlying response. This danger is captured by Eq. 3.2.5. As we add more terms we expect to decrease the numerator, but the denominator will also decrease.

Another measure related to the rms error is obtained by calculating the variation of the data from its average \( \bar{y} \)

\[
\bar{y} = \frac{1}{n_y} \sum_{i=1}^{n_y} y_i. \tag{3.2.6}
\]

The variation from the average is denoted as \( SS_y \) and is given as

\[
SS_y = \sum_{i=1}^{n_y} (y_i - \bar{y})^2. \tag{3.2.7}
\]

Similarly the variation of the response surface \( \hat{y} \) from the same average is denoted as \( SS_r \) so that

\[
SS_r = \sum_{i=1}^{n_y} (\hat{y}_i - \bar{y})^2. \tag{3.2.8}
\]

The ratio of the two, denoted \( R^2 \), measures the fraction of the variation in the data is captured by the response surface

\[
R^2 = \frac{SS_r}{SS_y} = 1 - \frac{SS_e}{SS_y}, \tag{3.2.9}
\]

where the last equality is left to the reader as an exercise.

As we add more coefficients to the polynomial, we reduce \( SS_e \) and therefore increase \( R^2 \). However, as noted before, this does not mean that the prediction capabilities of our model improve. For this reasons, there is an adjusted form of \( R^2 \) which is given as

\[
R_a^2 = 1 - \frac{SS_e}{SS_y/n_y} = 1 - \left(1 - R^2 \right) \frac{n_y - 1}{n_y - n_y}. \tag{3.2.10}
\]

If the adjusted value, \( R_a^2 \), decreases as we increase the number of coefficients, it is a warning that we may be fitting the data better, but losing on predictive capability.

The only true test to the predictive capabilities of the response surface is evaluating it at points not used in its construction. This is done often by practitioners. However, because additional tests or numerical evaluations of \( y \) are often expensive, it
is worthwhile to look for ways of checking predictive capability without performing additional evaluations at new points. If the number of points used for the fit is substantially larger than the number of coefficients, leaving out a few points will not change much the quality of the fit. We can therefore leave out a few points, fit the response surface to the remaining points and check the error at the left-out points. We can then replicate the procedure with other points, so that each point is left out once, in a procedure known as cross validation.

When the cost of repeating the fit is small (as it is for response surface techniques), cross validation is usually done leaving one point at a time. The procedure goes also by the name of PRESS (for prediction error sum of squares), a name that comes from using the sum of the squares of the prediction errors as a measure of the predictive accuracy of the fit. We denote the vector of errors as $e_p$. It is possible to obtain $e_p$ without actually performing all of these fits. When only one point is left out, it can be shown that the $i$th component of $e_p$ is related to the $i$th component of $e_r$ as

$$e_{pi} = \frac{e_{ri}}{1 - E_{ii}},$$

where $E_{ii}$ is the diagonal matrix of the so-called idempotent matrix

$$E = X(X^T X)^{-1} X^T. \quad (3.2.12)$$

It should be noted that the matrix $X^T X$ is often ill conditioned, especially for large problems and then the calculation of the matrix $E$ from Eq. 3.2.12 is not very reliable. However, even the direct calculation of $e_p$ by performing the $n_y$ fits, is usually less expensive than carrying out additional experiments in order to test the accuracy of the response surface.

**Example 3.2.1**

A response quantity $y$ is a function of a single variable $x$, and has eventually been identified to follow the simple relationship $y = x$. However, the first set of measurements that were taken are given the four $(x, y)$ data pairs (-2,-1.5), (-1, -1.5), (1, 1.25), (2, 1.75). From theoretical considerations you also know that $y(0) = 0$. Fit a linear function and a quadratic to the data, compare the two fits and see how they model the true function.

**Linear fit:** The response surface is of the form $y = b_1 + b_2 x$, and using the point $(0,0)$ as the third points we have

$$X = \begin{bmatrix} 1 & -2 \\ 1 & -1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix}, \quad y = \begin{bmatrix} -1.5 \\ -1.5 \\ 0 \\ 1.25 \\ 1.75 \end{bmatrix}. \quad (3.2.13)$$

So that

$$X^T X = \begin{bmatrix} 5 & 0 \\ 0 & 10 \end{bmatrix}, \quad X^T y = \begin{bmatrix} 0 \\ 9.25 \end{bmatrix}. \quad (3.2.14)$$

So that we can solve for $b$ as
\[
\begin{align*}
(X^T X)^{-1} &= \begin{bmatrix} 0.2 & 0 \\ 0 & 0.1 \end{bmatrix}, & b = (X^T X)^{-1} X^T y = \begin{bmatrix} 0 \\ 0.925 \end{bmatrix}, \\
\end{align*}
\] (3.2.15)

and the linear response surface, and the error vector are
\[
\hat{y} = 0.925 x, & \quad e^T = [0.35, -0.575, 0, 0.325, -0.1].
\] (3.2.16)

We now calculate the sums of the squares, and use Eqs. 3.2.9 and 3.2.10 to obtain
\[
SS_y = 9.125, SS_e = 0.56875, R^2 = 0.9377, R_a^2 = 1 - (1 - 0.9377) \frac{4}{3} = 0.9169.
\] (3.2.17)

This linear fit appears to be quite satisfactory, with the errors being
\[
e_{\text{rms}} = 0.337, e_{\text{av}} = 0.27, e_{\max} = 0.575.
\] (3.2.18)

The estimate of the standard deviation of the noise (standard error) in the data from Eq. 3.2.5 is
\[
\hat{\sigma}^2 = \frac{SS_e}{n_y - n_p} = 0.1896, & \quad \hat{\sigma} = 0.4354,
\] (3.2.19)

which is in line with the actual function being \( y = x \), since the errors we introduced in the data are 0.5 and 0.25.

**Quadratic model**: The response surface is of the form \( y = b_1 + b_2 x + b_3 x^2 \) and we get
\[
X = \begin{bmatrix} 1 & -2 & 4 \\
1 & -1 & 1 \\
1 & 0 & 0 \\
1 & 1 & 1 \\
1 & 2 & 4 \end{bmatrix}, & \quad y = \begin{bmatrix} -1.5 \\
-1.5 \\
0 \\
1.25 \\
1.75 \end{bmatrix},
\] (3.2.20)

and
\[
X^T X = \begin{bmatrix} 5 & 0 & 10 \\
0 & 10 & 0 \\
10 & 0 & 34 \end{bmatrix}, & \quad X^T y = \begin{bmatrix} 0 \\
9.25 \\
0.75 \end{bmatrix},
\] (3.2.21)

We solve for \( b \) to obtain
\[
b = (X^T X)^{-1} X^T y = \begin{bmatrix} -0.1071 \\
0.925 \\
0.05357 \end{bmatrix},
\] (3.2.22)

So that our approximation is
\[
y = -0.1071 + 0.925 x + 0.05357 x^2, \quad e^T = [-0.243, 0.521, -0.107, -0.378, 0.207].
\] (3.2.23)

We again calculate the sums of the squares to obtain
\[
SS_y = 9.125, SS_e = 0.52857, R^2 = 0.9421, R_a^2 = 1 - (1 - 0.9421) \frac{4}{2} = 0.8841.
\] (3.2.24)

We see that there is only a small improvement in \( R^2 \), and that \( R_a^2 \) is actually poorer, which is an indication that we have not gained any predictive capabilities by adding the quadratic terms. The error measures for the quadratic model are
\[
e_{\text{rms}} = 0.325, \quad e_{\text{av}} = 0.291, \quad e_{\max} = 0.521,
\] (3.2.25)
These have not changed much compared to the linear model, with small improvements in the rms and maximum errors, and small increase in the average error. The estimate of the standard deviation of the noise in the data (standard error) from Eq. 3.2.5 is

$$\hat{\sigma}^2 = \frac{SS^c}{n_x - n_B} = 0.2643, \quad \hat{\sigma} = 0.5141,$$

which is a small increase over the linear case. This increase is another indication that the quadratic approximation is not better than the linear approximation. The two approximations are compared in Fig. 3.4.

In Example 3.2.1 we compared a linear model to a quadratic model. However, we do not have to limit ourselves to models that have all the terms up to a particular order. For example, we can consider a quadratic model without the constant term, 

$$y = b_1 x + b_2 x^2.$$ Similarly, with two variables, it is common for people to consider a model of the form 

$$y = b_1 + b_2 x_1 + b_3 x_2 + b_4 x_1 x_2.$$ This model does not include quadratic terms in 

$x_1$ and $x_2$, but it includes the 'interaction' term $x_1 x_2$. We can stipulate such a partial model on the basis of some knowledge of the behavior of the true function. However, most often such partial models are created by discarding terms with coefficients which cannot be accurately estimated on the basis of the data. Such coefficients do not have much effect on the accuracy of the fit of the given data, and leaving them in the model can reduce its predictive quality for design points where these coefficients have large effect on the prediction.

In trying to fit a response surface to the profit per can, in Example 3.1.2, we observed that some of the coefficients were sensitive to small differences in the data. We can identify these coefficients without changing the data by estimating the standard deviation of the coefficients.

Let us first introduce the covariance matrix $\Sigma_b$ of the vector of coefficients $b$. This covariance matrix is defined as
That is, due to noise we get various vectors $\mathbf{b}$ by fitting the data from several sets of experiments and $E(\mathbf{b})$ is the expected value (or average over very large number of experiments) of $\mathbf{b}$ and $\mathbf{b} - E(\mathbf{b})$ is the deviation of $\mathbf{b}$ from its expected value. Then the covariance matrix is the expected value of products of various components of the difference. In particular, the diagonal terms of the matrix are by definition the squares of the standard deviations of the components of $\mathbf{b}$, while the off-diagonal terms are a measure of the correlation between components. It is possible to show that

$$
\sum_{\mathbf{b}} = \sigma^2 \left( X^T X \right)^{-1},
$$

so that with our estimate $\hat{\sigma}^2$ of $\sigma^2$ (see Eq. 3.2.5), we can estimate the standard deviations of the individual terms. The estimate of the standard deviation of the components of $\mathbf{b}$ is called the standard error, $se$, so that

$$
se(b_i) = \hat{\sigma} \sqrt{\left( X^T X \right)^{-1} i},
$$

A useful measure of the accuracy of a component of $\mathbf{b}$ is the estimate of the coefficient of variation, $c$, which is the standard deviation of the component divided by the absolute value of that component. The coefficient of variation of the $i$th component $c_i$ is therefore the standard error divided by the component

$$
c_i = \frac{se(b_i)}{|b_i|},
$$

In many response surface procedures the quantity which is used to assess the need for a coefficient is the inverse of $c_i$, which is called the test statistic, or the $t$-statistic. The coefficient of variation is used in a strategy called backward elimination (Myers and Montgomery, p. 650). In this strategy, we eliminate the coefficient with the largest coefficient of variation, perform another regression with the remaining terms, eliminate the coefficient with the highest coefficient of variation, and so on. We can stop once the coefficient of variation of the remaining terms is small enough, or we can use a measure such as $R^2_a$ to indicate to us when eliminating additional terms is hurting us. This is illustrated by applying this strategy to Example 3.2.1.

**Example 3.2.2**

Use backward elimination, starting with the quadratic model of Example 3.2.1, to find the model with the highest value of $R^2_a$. Confirm your conclusion by a recalculating coefficients for a slight perturbation of the data at $x = 1$ to $y(1) = 1.35$. For the quadratic model in Example 3.2.1, we found that

$$
X^T X = \begin{bmatrix}
5 & 0 & 10 \\
0 & 10 & 0 \\
10 & 0 & 34
\end{bmatrix}, \quad \hat{\sigma}^2 = 0.2643,
$$

From this we calculate
The vector coefficients for the quadratic model was \( \mathbf{b} = [-0.1071, 0.925, 0.05357]^T \), so from Eq. 3.2.31 we get

\[
\begin{pmatrix}
1 & 0 & -1 \\
0 & 1 & 0 \\
-1 & 0 & 1
\end{pmatrix}
\]

\( \mathbf{X} \) and

\[
\sum_b = \begin{bmatrix}
0.1284 & 0 & -0.03776 \\
0 & 0.02643 & 0 \\
-0.03776 & 0 & 0.01888
\end{bmatrix}
\]

(3.2.33)

The vector coefficients for the quadratic model was \( \mathbf{b} = [-0.1071, 0.925, 0.05357]^T \), so from Eq. 3.2.31 we get

\[
c_1 = \frac{\sqrt{0.1284}}{0.1071} = 3.35, \quad c_2 = \frac{\sqrt{0.02643}}{0.925} = 0.176, \quad c_3 = \frac{\sqrt{0.01888}}{0.05357} = 2.56, \quad (3.2.34)
\]

Note that two of the coefficients of variation are very large. A value larger than one indicates that the standard deviation of the coefficient is larger than the value itself, so that we have very little confidence in the value obtained from the regression. The coefficient of variation associated with the constant term \( b_1 \) is the largest one, so we eliminate it. This leads to the partial quadratic model \( y = b_1 x + b_2 x^2 \). For this model we get

\[
X = \begin{pmatrix}
-2 & 4 \\
-1 & 1 \\
0 & 0 \\
1 & 1 \\
2 & 4
\end{pmatrix}, \quad y = \begin{pmatrix}
-1.5 \\
-1.5 \\
0 \\
1.25 \\
1.75
\end{pmatrix}
\]

And

\[
X^T X = \begin{pmatrix}
10 & 0 \\
0 & 34
\end{pmatrix}, \quad X^T y = \begin{pmatrix}
9.25 \\
0.75
\end{pmatrix}
\]

(3.2.35)

We solve for \( \mathbf{b} \) to obtain

\[
\mathbf{b} = (X^T X)^{-1} X^T y = \begin{pmatrix}
0.925 \\
0.02205
\end{pmatrix}
\]

(3.2.37)

so that the approximation is

\[
y = 0.925x + 0.03676x^2, \quad \mathbf{e}^T = [-0.262, 0.597, 0, -0.3038, 0.188].
\]

(3.2.38)

We calculate the sums of the squares

\[
SS_y = 9.125, SS_e = 0.5522, R^2 = 0.9394, R_a^2 = 1(1 - 0.9394) \frac{4}{3} = 0.9193 \quad (3.2.39)
\]

Comparing to the quadratic model in Example 3.2.1, we note a substantial improvement in \( R_a \) which for the quadratic model was 0.8841. We also calculate a new \( \hat{\sigma}^2 \) as

\[
\hat{\sigma}^2 = \frac{SS_e}{n_y - n_\beta} = \frac{0.5522}{3} = 0.1841 \quad \hat{\sigma} = 0.429
\]

(3.2.40)

which is also an improvement over the quadratic model. Next we calculate the new covariance matrix.
\[
(X^T X)^{-1} = \begin{bmatrix}
1 & 0 \\
0 & 10 \\
0 & 0
\end{bmatrix}, \quad \Sigma_b = \begin{bmatrix}
0.0184 & 0 \\
0 & 0.00541
\end{bmatrix}, \quad (3.2.41)
\]

The new coefficients of variations are then given as
\[
c_1 = \frac{\sqrt{0.0184}}{0.925} = 0.147, \quad c_2 = \frac{\sqrt{0.00541}}{0.02206} = 3.33, \quad (3.2.42)
\]

We still have a large value of \(c_2\), so to continue the process of backward elimination, we need to eliminate the \(b_2\) coefficient, associated with the \(x^2\) term. However, we know that we will get the linear model back again, because in Example 3.2.1, when we fitted a linear model, this model yielded \(\hat{y} = 0.925x\). Furthermore, both \(\hat{\sigma} = (0.4354)\) and \(R^2 = (0.9169)\) for the linear model obtained in Example 3.2.1 are slightly inferior to our incomplete quadratic model. This indicates that taking one more step in the backward elimination will confirm that the incomplete quadratic model is the best.

Next we check how the coefficients change if we change \(y(1) = 1.35\). If we repeat the quadratic model, we find that we get \(\hat{y} = -0.0729 + 0.935x + 0.0465x^2\). We see that the small change in the data changed the free coefficient by more than 30%, the \(x^2\) coefficient by about 13%, and the coefficient of \(x\) by only about 1%. If we repeat the calculation for the incomplete quadratic model we find \(\hat{y} = 0.935x + 0.025x^2\), which is about 1% of the linear term, and about 13% different in the quadratic term..

### 3.3 Nonlinear Regression

The advantage of having a response surface which is a linear function of the coefficients, is that the regression process requires only the solution of a system of linear equations (the normal equations). However, in many situations such a response surface will not fit the data well. In particular, most of the work with linear regression is done with linear and quadratic polynomials, and these are very limited in terms of the kind of functions that they will fit well.

In nonlinear regression, we assume that the response surface is in the more general form
\[
f(x) = y(x, \beta), \quad (3.3.1)
\]

We again have a vector of function values \(y_i = y(x_i), i = 1, \ldots, n\), and we try to estimate \(\beta\) by a vector \(b\) that minimizes the rms error
\[
e_{rms}^2 = \frac{1}{n_y} \sum_{i=1}^{n_y} r_i^2, \quad (3.3.2)
\]

where \(r_i\) is the residual at the \(i\)th data point
\[
r_i = y(x_i, b) - y_i \quad (3.3.3)
\]

However, unlike the case of linear regression, we cannot perform the fit by solving a set of linear equations, and instead we have to perform unconstrained minimization, typically requiring a numerical solution. The following example demonstrates the advantages of nonlinear regression.
Example 3.3.1

Given the four data points, \( y(1) = 20, y(2) = 7, y(3) = 5, \) and \( y(4) = 4, \) find an appropriate function to fit and compare to polynomial fit.

The large value of \( y(1) \) indicates that the function may become unbounded near that value, so that an approximation using rational functions may be appropriate. So we try

\[
y = b_1 + \frac{b_2}{b_3 - x},
\]

(3.3.4)

The residuals at the four data points are

\[
r_1 = b_1 + \frac{b_2}{b_3 - 1} - 20, \quad r_2 = b_1 + \frac{b_2}{b_3 - 2} - 7,
\]

(3.3.5)

\[
r_3 = b_1 + \frac{b_2}{b_3 - 3} - 5, \quad r_4 = b_1 + \frac{b_2}{b_3 - 4} - 4,
\]

(3.3.6)

Optimizing with Microsoft Excel the sum of the four residuals yielded the following solution \( b_1 = 2, b_2 = -7, b_3 = 0.612 \) that is

\[
y = 2 + \frac{7}{x - 0.612},
\]

(3.3.7)

A comparable polynomial approximation with 3 coefficients will be a quadratic polynomial, and performing regression yields

\[
y = 36.5 - 20x + 3x^2,
\]

(3.3.8)

with an estimated \( \sigma^2 = 5 \), a rather substantial error. The two approximations are compared in the Figure 3.5.

While we cannot use linear regression for the fitting, we can use it after the fitting is done to obtain an estimate of the errors in the coefficients. This can be done by linearization about the solution of the nonlinear regression. We linearize Eq. 3.3.3 about the solution \( \mathbf{b}^* \) obtained from the least square fit

\[
r_i = y(x_i, \mathbf{b}^*) - y_i \approx \sum_{j=1}^m \frac{\partial y}{\partial b_j}(x_i, \mathbf{b}^*) \Delta b_j,
\]

(3.3.9)

That is, if we fit the residuals \( r_i \) using as basis functions the derivatives \( \partial y/\partial b_j \), we can get estimates of the standard deviation of the coefficients.
Example 3.3.2

We linearize the previous example about the value of the coefficients obtained from the nonlinear regression, obtaining

\[ r = \Delta b_1 + \frac{\Delta b_2}{x - 0.612} - \frac{7}{(x - 0.612)^2} \Delta b_3. \]  

(3.3.10)

A linear regression analysis with Microsoft Excel gives \( \Delta b_1 = 0.0090, \Delta b_2 = 0.0080, \Delta b_3 = 0.00026 \). Note that these increments should be subtracted from the values we obtained in the previous example for more accurate fit, because they are a fit to the residual \( \hat{y} - y \). From the linear regression we get an estimate for the standard error \( \hat{\sigma} = 0.102 \), and standard errors for the coefficients as \( se(b_1) = 0.20, \quad se(b_2) = 0.50, \quad se(b_3) = 0.024 \). These indicate high confidence in the values of the coefficients, since the standard errors are less than 10% of the values of the coefficient.

There are many standard forms of nonlinear regression that we will consider in subsequent chapters including Kriging, neural networks, and support vector regression. Finally, the process of finding physical constants from laboratory or field measurements can be viewed as a process of nonlinear regression, since the dependence of the measurements is rarely linear in these constants. Unfortunately, the problem of minimizing the rms errors for fitting physical experiments often has a large number of local optima, so that choosing the best answer is not always simple.
3.4 Ouliers

Both in physical experiments and computer simulations occasionally we have data with large errors. In computer simulations these may reflect failures of the solution algorithm, software implementation, or mistakes by the user of the software. Data points with large errors are called outliers. It is important to detect them and either remove or repair them, because they can have a large detrimental effect on the accuracy of the response surface.

A standard tool for detecting outliers is Iteratively Reweighted Least Squares (IRLS) procedures. In order to understand their basis, let us consider first the weighted least square (WLS) procedure.

Weighted least squares procedures minimize a weighted sum of the squares of the residuals. There are many possible reasons for using WLS rather than standard least squares. We may have more confidence in some data than in others. We may want to weight more heavily points that are close to the region where we will need to predict the response than points far away. The noise at some points may be known to be higher than in other points. In these cases, instead of using Eq. 3.1.12 we use

$$ e_{wrmse} = \frac{1}{\sqrt{\sum_{i=1}^{n_y} w_i e_i^2}} = \frac{1}{\sqrt{\sum_{i=1}^{n_y} e_i^2 w_i}} $$(3.4.1)

where, $w_i$ is the weight associated with the ith point, and $W$ is a diagonal matrix with the weights on the diagonal. Minimizing the weighted rms error, $e_{wrmse}$ yields a modified set of normal equations

$$ X^TWXb = X^Wy, $$ (3.4.2)

Iteratively reweighted least square procedures weight points with large residuals with small weights, with the weight decreasing with increasing magnitude of the residual. Then the WLS procedure is performed. If the point is an outlier, the response surface will move away from it, so its residual will increase. We will assign a smaller weight to the point and repeat the procedure. Eventually, outliers are likely to end up with zero or low weight. There are several weighting schemes, see e.g., Myers and Montgomery, p. 671. One of the simplest is Huber’s

$$ w_i = \begin{cases} 1 & \text{if } |e_i| / \hat{\sigma} \leq 1 \\ \hat{\sigma} / |e_i| & \text{otherwise} \end{cases} $$ (3.4.3)

Example 3.4.1

We need to estimate Young’s modulus $E$ on the basis of four stress-strain measurements. For values of strains of 1, 2, 3, and 4, millistrains, we measures stress values of 9, 22, 36, and 39 ksi. Denoting the stress by $y$ and the strain by $x$ we perform a standard least square fit $y=Ex$.

$$ X = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \quad y = \begin{bmatrix} 9 \\ 22 \\ 36 \\ 39 \end{bmatrix} $$ (3.4.4)
So we have

\[ X^T X = 30, \quad X^T y = 317, \quad E = 10.567 \text{Msi} \]  \hspace{1cm} (3.4.5)

The vector of residuals is \( e^T = [1.567, 0.867, 4.3, 3.267] \) giving

\[ \hat{\sigma} = \sqrt{e^T e / 3} = 3.2846, \quad se(E) = 3.2846 \sqrt{1/30} = 0.6, \]  \hspace{1cm} (3.4.6)

The only residual greater than \( \hat{\sigma} \) is the third one, so its weight \( w_3 = 3.2846/4.3 = 0.764 \)

We now perform a weighted least squares fit, giving us

\[ X^T WX = 27.876, \quad X^T Wy = 295.512, \quad E = 10.46, \]  \hspace{1cm} (3.4.7)

The new vector of residuals is \( e^T = [1.457, 1.085, 4.628, 2.830] \), giving \( \hat{\sigma} = \sqrt{e^T We / 3} = 3.037 \)

Again, the only residual larger than \( \hat{\sigma} \) is the third one, and its weight now reduces to \( w_3 = 3.037/4.628 = 0.656 \). One more iteration gives us

\[ X^T WX = 26.904, \quad X^T Wy = 279.848, \quad E = 10.40, \]  \hspace{1cm} (3.4.8)

We can continue iterating, or we can be satisfied that the process has identified the third point as an outlier and discard it. If we do that we get \( E = 9.95 \text{ Msi} \), with a standard error \( se(E) = 0.376 \)

### 3.5 Exercises

1. For Example 3.2.1 calculate the rms error from the cross-validation PRESS procedure and compare to other estimates of the error.

2. Check the accuracy of the quadratic response surface in Example 3.1.2 in the region \( 1.5 \leq D \leq 3.5, \quad 3 \leq H \leq 7 \). Find the maximum error, average error and rms error (a) for the 9 data points, (b) for the entire region (using the analytical expression). For part (b) you may perform the calculation analytically, or you may cover the domain with a grid of (20X20) points and calculate the error in each one of the 400 points. (c) Calculate the error using the PRESS procedure and compare to the result of part (b)

3. Using the data in Example 3.1.2 fit a quadratic polynomial to the profit per ounce, and perform the same error analysis as requested in Problem 2.

4. Using the data of Problem 2, use backward elimination to find an incomplete quadratic polynomial with the highest \( R_a^2 \). a. Then check for the accuracy of the fit compared to the analytical profit per can over the entire region, and compare to the results obtained in Problem 2.