Chapter I

Error Estimations

Absolute and Relative Error:

Suppose that p^* is an approximation to p. The absolute error is $|p - p^*|$, and the relative error is $\frac{|p-p^*|}{|p|}$, provided that $p \neq 0$.

Consider the absolute and relative errors in representing p by p^* in the following example.

Example:

Determine the absolute and relative errors when approximating p by p^* when
(a) $p = 0.3000 \times 10^1$ and $p^* = 0.3100 \times 10^1$;
(b) $p = 0.3000 \times 10^{-3}$ and $p^* = 0.3100 \times 10^{-3}$;
(c) $p = 0.3000 \times 10^4$ and $p^* = 0.3100 \times 10^4$.

Solution:

(a) For $p = 0.3000 \times 10^1$ and $p^* = 0.3100 \times 10^1$ the absolute error is 0.1, and the relative error is $0.333\overline{3} \times 10^{-1}$.

(b) For $p = 0.3000 \times 10^{-3}$ and $p^* = 0.3100 \times 10^{-3}$ the absolute error is 0.1×10^{-4}, and the relative error is $0.333\overline{3} \times 10^{-1}$.

(c) For $p = 0.3000 \times 10^4$ and $p^* = 0.3100 \times 10^4$, the absolute error is 0.1×10^3, and the relative error is again $0.333\bar{3} \times 10^{-1}$.

This example shows that the same relative error, $0.333\bar{3} \times 10^{-1}$, occurs for widely varying absolute errors. As a measure of accuracy, the absolute error can be misleading and the relative error more meaningful, because the relative error takes into consideration the size of the value.

The following definition uses relative error to give a measure of significant digits of accuracy for an approximation.

Definition:

The number p^* is said to approximate p to t significant digits (or figures) if t is the largest nonnegative integer for which

$$\frac{|p - p^*|}{|p|} \leq 5 \times 10^{-t}$$

Truncation Error:

Taylor Polynomials and Series

Taylor polynomials and series that you studied in calculus are used extensively in numerical analysis.

Taylor's Theorem:

Suppose $f \in C^n[a,b]$, that $f^{(n+1)}$ exists on $[a,b]$, and $x_0 \in [a,b]$. For every $x \in [a,b]$, there exists a number $\xi(x)$ between x_0 and x with

$$f(x) = P_n(x) + R_n(x),$$

where

$$\begin{aligned} P_n(x) &= f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \cdots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n \\ &= \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k \end{aligned}$$

and

$$R_n(x) = \frac{f^{(n+1)}(a)}{(n+1)!}(x - x_0)^{n+1}$$

$$a \in [x_0, x]$$

Here $P_n(x)$ is called the $\boldsymbol{n}$ th Taylor polynomial for f about x_0, and $R_n(x)$ is called the remainder term (or **truncation error**) associated with $P_n(x)$. Since the number $\xi(x)$ in the truncation error $R_n(x)$ depends on the value of x at which the polynomial $P_n(x)$ is being evaluated, it is a function of the variable x.

In the case $x_0 = 0$, the Taylor polynomial is often called a **Maclaurin polynomial**, and the Taylor series is often called a **Maclaurin series**.

Example 1:

Let $f(x) = \cos\ x$ and $x_0 = 0$. Determine

(a) the second Taylor polynomial for f about x_0; and

(b) the third Taylor polynomial for f about x_0.

Solution:

$$f'(x) = -\sin\ x, f''(x) = -\cos\ x, f'''(x) = \sin\ x,\ \text{ and }\ f^{(4)}(x) = \cos\ x,$$

so

$$f(0) = 1, f'(0) = 0, f''(0) = -1,\ \text{ and }\ f'''(0) = 0.$$

(a) For $n = 2$ and $x_0 = 0$, we have

$$\begin{aligned} \cos\ x \quad &= f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \frac{f'''(a)}{3!}x^3 \\ &= 1 - \frac{1}{2}x^2 + \frac{1}{6}x^3 \sin\ a \end{aligned}$$

See Figure 1

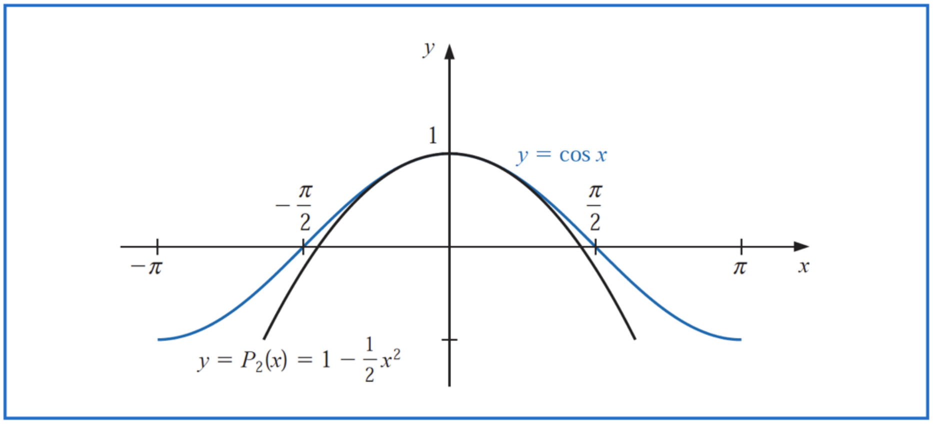

Fig. 1

When $x = 0.01$, this becomes

$$\cos\ 0.01 = 1 - \frac{1}{2}(0.01)^2 + \frac{1}{6}(0.01)^3 \sin\ \xi(0.01) = 0.99995 + \frac{10^{-6}}{6}\sin\ a$$

where $0 < \mathrm{a} < 0.01$.

The approximation to $\cos\ 0.01$ given by the Taylor polynomial is therefore 0.99995. The truncation error, or remainder term, associated with this approximation is

$$\frac{10^{-6}}{6}\sin\ a = 0.1\overline{6} \times 10^{-6}\sin\ a,$$

where the bar over the 6 in $0.1\overline{6}$ is used to indicate that this digit repeats indefinitely. Although we have no way of determining $\sin\ a(0.01)$, we know that all values of the sine lie in the interval $[-1,1]$, so the error occurring if we use the approximation 0.99995 for the value of $\cos\ 0.01$ is bounded by

$$|\cos\,(0.01) - 0.99995| = 0.1\overline{6} \times 10^{-6}|\sin\ a| \leq 0.1\overline{6} \times 10^{-6}$$

(b) Since $f'''(0) = 0$, the third Taylor polynomial with remainder term about $x_0 = 0$ is

$$\cos\ x = 1 - \frac{1}{2}x^2 + \frac{1}{24}x^4 \cos\ \mathrm{a}$$

where $0 < \mathrm{a} < 0.01$.

The approximating polynomial remains the same, and the approximation is still 0.99995, but we now have much better accuracy assurance. Since $|\cos\ \mathrm{a}| \leq 1$ for all x, we have

$$\left|\frac{1}{24}x^4 \cos\ \mathrm{a}\right| \leq \frac{1}{24}(0.01)^4(1) \approx 4.2 \times 10^{-10}$$

So,

$$|\cos\ 0.01 - 0.99995| \leq 4.2 \times 10^{-10}$$

Exercise:

(1) For the following functions, determine the third-order Taylor polynomial for $x_0 = 0.1$. Then estimate the error at the point $x = 0.05$.

a. $f(x) = e^{x^3}$

b. $f(x) = \sin x$

(2) For the function, $f(x) = e^{-3x}$ determine the third-order Taylor polynomial for $x_0 = 0.5$. Then estimate the error at the point $x = 0.1$.

Round-off Errors and Computer Arithmetic

The arithmetic performed by a calculator or computer is different from the arithmetic in algebra and calculus courses. You would likely expect that we always have as true statements things such as $2 + 2 = 4, 4 \cdot 8 = 32$, and $(\sqrt{3})^2 = 3$. However, with computer arithmetic we expect exact results for $2 + 2 = 4$ and $4 \cdot 8 = 32$, but we will not have precisely $(\sqrt{3})^2 = 3$.

To understand why this is true we must explore the world of finite-digit arithmetic.

In our traditional mathematical world, we permit numbers with an infinite number of digits. The arithmetic we use in this world defines $\sqrt{3}$ as that unique positive number that when multiplied by itself produces the integer 3. In the computational world, however, each representable number has only a fixed and finite number of digits. This means, for example, that only rational numbers-and not even all of these-can be represented exactly. Since $\sqrt{3}$ is not rational, it is given an approximate representation, one whose square will not be precisely 3, although it will likely be sufficiently close to 3 to be acceptable in most situations. In most cases, then, this machine arithmetic is satisfactory and passes without notice or concern, but at times problems arise because of this discrepancy.

The error that is produced when a calculator or computer is used to perform real number calculations is called round-off error. It occurs because the arithmetic performed in a machine involves numbers with only a finite number of digits, with the result that calculations are performed with only approximate representations of the actual numbers. In a computer, only a relatively small subset of the real number system is used for the representation of all the real numbers. This subset contains only rational numbers, both positive and negative, and stores the fractional part, together with an exponential part.

Decimal Machine Numbers

The use of binary digits tends to conceal the computational difficulties that occur when a finite collection of machine numbers is used to represent all the real numbers. To examine these problems, we will use more familiar decimal numbers instead of binary representation. Specifically, we assume that machine numbers are represented in the normalized decimal floating-point form

$$\pm 0.d_1d_2\ldots d_k \times 10^n,\ 1 \le d_1 \le 9,\ \text{ and }\ 0 \le d_i \le 9,$$

for each $i = 2, \ldots, k$. Numbers of this form are called k-digit decimal machine numbers.

Any positive real number within the numerical range of the machine can be normalized to the form

$$y = 0.d_1d_2\ldots d_kd_{k+1}d_{k+2}\ldots \times 10^n$$

The floating-point form of y, denoted $fl(y)$, is obtained by terminating the mantissa of y at k decimal digits. There are two common ways of performing this termination. One method, called chopping, is to simply chop off the digits $d_{k+1}d_{k+2}\ldots$ This produces the floating-point form

$$fl(y) = 0.d_1d_2\ldots d_k \times 10^n$$

The other method, called rounding,

$$fl(y) = 0.\delta_1\delta_2\ldots\delta_k \times 10^n$$

For rounding, when $d_{k+1} \ge 5$, we add 1 to d_k to obtain $fl(y)$; that is, we round up. When $d_{k+1} < 5$, we simply chop off all but the first k digits; so we round down.

Example:

Determine the five-digit (a) chopping and (b) rounding values of the irrational number π. Solution The number π has an infinite decimal expansion of the form $\pi = 3.14159265\ldots$ Written in normalized decimal form, we have

$$\pi = 0.314159265 \ldots \times 10^{1}.$$

(a) The floating-point form of π using five-digit chopping is

$$fl(\pi) = 0.31415 \times 10^{1} = 3.1415.$$

(b) The sixth digit of the decimal expansion of π is a 9 , so the floating-point form of π using five-digit rounding is

$$fl(\pi) = (0.31415 + 0.00001) \times 10^{1} = 3.1416$$

Finite-Digit Arithmetic

Assume that the floating-point representations $fl(x)$ and $fl(y)$ are given for the real numbers x and y and that the symbols $\oplus$,$\ominus$,$\otimes$,$\odot$ represent machine addition, subtraction, multiplication, and division operations, respectively. We will assume a finite-digit arithmetic given by

$$\begin{aligned} x \oplus y &= fl(fl(x) + fl(y)), \quad x \otimes y = fl(fl(x) \times fl(y)) \\ x \ominus y &= fl(fl(x) - fl(y)), \quad x \oplus y = fl(fl(x) \div fl(y)) \end{aligned}$$

This arithmetic corresponds to performing exact arithmetic on the floating-point representations of x and y and then converting the exact result to its finite-digit floating-point representation.

Example 3 Suppose that $x = \frac{5}{7}$ and $y = \frac{1}{3}$. Use five-digit chopping for calculating $x + y, x - y, x \times y$, and $x \div y$.
Solution Note that

$$x = \frac{5}{7} = 0.\overline{714285} \text{ and } y = \frac{1}{3} = 0.\overline{3}$$

implies that the five-digit chopping values of x and y are

$$fl(x) = 0.71428 \times 10^0 \text{ and } fl(y) = 0.33333 \times 10^0$$

Thus

$$\begin{aligned} x \oplus y \quad &= fl(fl(x) + fl(y)) = fl(0.71428 \times 10^0 + 0.33333 \times 10^0) \\ &= fl(1.04761 \times 10^0) = 0.10476 \times 10^1 \end{aligned}$$

The true value is $x + y = \frac{5}{7} + \frac{1}{3} = \frac{22}{21}$, so we have

$$\text{Absolute Error } = \left|\frac{22}{21} - 0.10476 \times 10^1\right| = 0.190 \times 10^{-4}$$

and

$$\text{Relative Error } = \left|\frac{0.190 \times 10^{-4}}{22/21}\right| = 0.182 \times 10^{-4}.$$

Table 1.2 lists the values of this and the other calculations.
Table 1.2

Operation	Result	Actual value	Absolute error	Relative error
$x \oplus y$	0.10476×10^1	$22/21$	0.190×10^{-4}	0.182×10^{-4}
$x \ominus y$	0.38095×10^0	$8/21$	0.238×10^{-5}	0.625×10^{-5}
$x \otimes y$	0.23809×10^0	$5/21$	0.524×10^{-5}	0.220×10^{-4}
$x \oplus y$	0.21428×10^1	$15/7$	0.571×10^{-4}	0.267×10^{-4}

The maximum relative error for the operations in Example 3 is 0.267×10^{-4}, so the arithmetic produces satisfactory five-digit results. This is not the case in the following example.
Example 4 Suppose that in addition to $x = \frac{5}{7}$ and $y = \frac{1}{3}$ we have

$$u = 0.714251,\ v = 98765.9, \text{ and } w = 0.111111 \times 10^{-4}$$

so that

$$fl(u) = 0.71425 \times 10^0,\ fl(v) = 0.98765 \times 10^5,\ \text{ and } fl(w) = 0.11111 \times 10^{-4}$$

Determine the five-digit chopping values of $x \ominus u$, $(x \ominus u) \oplus w$, $(x \ominus u) \otimes v$, and $u \oplus v$.

Example 3

Suppose that $x = \frac{5}{7}$ and $y = \frac{1}{3}$. Use five-digit chopping for calculating $x + y, x - y, x \times y$, and $x \div y$.

Solution Note that

$$x = \frac{5}{7} = 0.\overline{714285} \text{ and } y = \frac{1}{3} = 0.\overline{3}$$

implies that the five-digit chopping values of x and y are

$$fl(x) = 0.71428 \times 10^0 \text{ and } fl(y) = 0.33333 \times 10^0$$

Thus

$$\begin{aligned} x \oplus y \quad &= fl(fl(x) + fl(y)) = fl(0.71428 \times 10^0 + 0.33333 \times 10^0) \\ &= fl(1.04761 \times 10^0) = 0.10476 \times 10^1 \end{aligned}$$

The true value is $x + y = \frac{5}{7} + \frac{1}{3} = \frac{22}{21}$, so we have

$$\text{Absolute Error } = \left|\frac{22}{21} - 0.10476 \times 10^1\right| = 0.190 \times 10^{-4}$$

and

$$\text{Relative Error } = \left|\frac{0.190 \times 10^{-4}}{22/21}\right| = 0.182 \times 10^{-4}.$$

Table 1.2 lists the values of this and the other calculations. Table 1.2

Operation	Result	Actual value	Absolute error	Relative error
$x \oplus y$	0.10476×10^1	$22/21$	0.190×10^{-4}	0.182×10^{-4}
$x \ominus y$	0.38095×10^0	$8/21$	0.238×10^{-5}	0.625×10^{-5}
$x \otimes y$	0.23809×10^0	$5/21$	0.524×10^{-5}	0.220×10^{-4}
$x \oplus y$	0.21428×10^1	$15/7$	0.571×10^{-4}	0.267×10^{-4}

The maximum relative error for the operations in Example 3 is 0.267×10^{-4}, so the arithmetic produces satisfactory five-digit results. This is not the case in the following example.

Example 4

Suppose that in addition to $x = \frac{5}{7}$ and $y = \frac{1}{3}$ we have

$$u = 0.714251,\ v = 98765.9,\ \text{ and } \ w = 0.111111 \times 10^{-4}$$

so that

$$fl(u) = 0.71425 \times 10^0,\ fl(v) = 0.98765 \times 10^5,\ \text{ and } \ fl(w) = 0.11111 \times 10^{-4}$$

Determine the five-digit chopping values of $x \ominus u$, $(x \ominus u) \oplus w$, $(x \ominus u) \otimes v$, and $u \oplus v$.

Solution These numbers were chosen to illustrate some problems that can arise with finite digit arithmetic. Because x and u are nearly the same, their difference is small. The absolute error for $x \ominus u$ is

$$
\begin{aligned}
|(x-u)-(x\ominus u)| &= |(x-u)-(fl(fl(x)-fl(u)))| \\
&= \left|\left(\frac{5}{7}-0.714251\right)-\left(fl(0.71428\times 10^{0}-0.71425\times 10^{0})\right)\right| \\
&= |0.347143\times 10^{-4}-fl(0.00003\times 10^{0})| = 0.47143\times 10^{-5}.
\end{aligned}
$$

This approximation has a small absolute error, but a large relative error

$$\left|\frac{0.47143\times 10^{-5}}{0.347143\times 10^{-4}}\right| \leq 0.136.$$

The subsequent division by the small number w or multiplication by the large number v magnifies the absolute error without modifying the relative error. The addition of the large and small numbers u and v produces large absolute error but not large relative error. These calculations are shown in Table 1.3.

Table 1.3

Operation	Result	Actual value	Absolute error	Relative error
$x \ominus u$	0.30000×10^{-4}	0.34714×10^{-4}	0.471×10^{-5}	0.136
$(x \ominus u) \oplus w$	0.27000×10^{1}	0.31242×10^{1}	0.424	0.136
$(x \ominus u) \otimes v$	0.29629×10^{1}	0.34285×10^{1}	0.465	0.136
$u \oplus v$	0.98765×10^{5}	0.98766×10^{5}	0.161×10^{1}	0.163×10^{-4}