MATHEMATICS AND COMPUTER SCIENCE

I. CALCULUS

The principal topics in calculus are the real and complex number systems, the concept of limits and convergence, and the properties of functions.

Convergence of a sequence of numbers x_i is defined as follows:

The sequence x_i converges to the limit x^* if, given any tolerance $\varepsilon > 0$, there is an index $N = N(\varepsilon)$ so that for all $i \geq N$ we have $|x_i - x^*| \leq \varepsilon$. The notation for this is

$$\lim_{i \to \infty} x_i = x^*.$$

Convergence is also a principal topics of numerical computation, but with a different emphasis. In calculus one studies limits and convergence with analytic tools; one tries to obtain the limit or to show that convergence takes place. In computations, one has the same problem but little or no theoretical knowledge about the sequence. One is frequently reduced to using empirical intuitive tests for convergence; often the principal task is to actually estimate the value of the tolerance for a given x.

The study of functions in calculus revolves about continuity, derivatives, and integrals. A function $f(x)$ is continuous if

$$\lim_{x_i \to x^*} f(x_i) = f(x^*)$$

holds for all x^* and all ways for the x_i to converge to x^*. We list six theorems from calculus which are useful for estimating values that appear in numerical computation.

Theorem 1 (Mean value theorem for continuous functions). *Let $f(x)$ be continuous on the interval $[a, b]$. Consider points XHI and XLOW in $[a, b]$ and a value y so that $f(XLOW) \leq y \leq f(XHI)$.Then there is a point ρ in $[a, b]$ so that*

$$f(\rho) = y$$

.

Theorem 2 (Mean value theorem for sums). *Let $f(x)$ be continuous on the interval $[a, b]$, let $x_1, x_2, \ldots, x_n$ be points in $[a, b]$ and let $w_1, w_2, \ldots, w_n$ be positive numbers. Then there is a point ρ in $[a, b]$ so that*

$$\sum_{i=1}^{n} w_i(x) f(x_i) = f(\rho) \sum_{i=1}^{n} w_i.$$

Theorem 3 (Mean value theorem for integrals). *Let $f(x)$ be continuous on the interval $[a, b]$ and let $w(x)$ be a nonnegative function $[w(x) \geq 0]$ on $[a, b]$. Then there is a point ρ in $[a, b]$ so that*

$$\int_a^b w(x) f(x) dx = f(\rho) \int_a^b w(x) dx.$$

Theorems 2 and 3 show the analogy that exists between sums and integrals. This fact derives from the definition of the integral as

$$\int_a^b f(x) dx = \lim_{\max |x_{i+1} - x_i| \to 0} \sum_i f(x_i)(x_{i+1} - x_i),$$

where the points x_i with $x_i < x_{i+1}$ are a partition of $[a, b]$. This analogy shows up for many numerical methods where one variation applies to sums and another applies to integrals. Theorem 2 is proved from Theorem 1, and then Theorem 3 is proved by a similar method. The assumption that $w(x) \geq 0 (w_i > 0)$ may be replaced by $w(x) \leq 0 (w_i < 0)$ in these theorems; it is essential that $w(x)$ be on one sign shown by the example $w(x) = f(x) = x$ and $[a, b] = [-1, 1]$.

Theorem 4 (Continuous functions assume max/min values). *Let $f(x)$ be continuous on the interval $[a, b]$ with $|a|, |b| \leq \infty$. Then there are points XHI and XLOW in $[a, b]$ so that for all x in $[a, b]$*

$$f(XHI) \leq f(x) \leq f(XLOW).$$

The **derivative** of $f(x)$ is defined by

$$\frac{df}{dx} = f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}.$$

As an illustration of the difference between theory and practice, the quantity $[f(x+h) - f(x)]/h$ can be replaced by $f(x+h) - f(x-h)]/(2h)$ with no change in the theory but with dramatic improvement in the rate of convergence; that is, much more accurate estimates of $f'(x)$ are obtained for a given value of h. The $k-$th derivative is the derivative of the $(k-1)$th derivative; they are denoted by $d^k f/dx^k$ or $f''(x), f'''(x), f^{(4)}(x), f^{(5)}(x), \ldots$

Theorem 5 (Mean value theorem for derivatives). *Let $f(x)$ be continuous and differentiable in $[a, b]$, with $|a|, |b| < \infty$. Then there is a point ρ in $[a, b]$ so that*

$$\frac{f(b) - f(a)}{b - a} = f'(\rho)$$

$$f(x) = f(c) + f'(\rho)(x - c)$$

The special case of Theorem 5. with $f(a) = f(b) = 0$ is known as Rolle's theorem. It states that if $f(a) = f(b) = 0$, then there is a point ρ between a and b so that $f'(\rho) = 0$. This is derived from Theorem 5 by multiplying through by $b - a$, renaming a, b as x, c, and then applying the first form to the smaller interval $[x, c]$ or $[c, x]$, depending on the relation between x and c.

A very important tool in numerical analysis is the extension of the second part of Theorem 5 to use higher derivatives.

Theorem 6 (Tailor series with remainder). *Let $f(x)$ have $n+1$ continuous derivatives in $[a,b]$.*

Given points x and c in $[a,b]$ we have

$$f(x) = f(c)+f'(c)(x-c)+f''(c)\frac{(x-c)^2}{2!}+f'''\frac{(x-c)^3}{3!}+\cdots+f^{(n)}(c)\frac{(x-c)^n}{n!}$$

$$+R_(n+1)(x),$$

where R_{n+1} has either one of the following forms (ρ is a point between x and c):

$$R_{n+1}(x) = f^{(n+1)}(\rho)\frac{(x-c)^{n+1}}{(n+1)!}$$

$$R_{n+1}(x) = \frac{1}{n!}\int_c^x (x-t)^n f^{(n+1)}(t)dt$$

If a function f depends on several variables, one can differentiate it with respect to one variable, say x, while keeping all the rest fixed. This is a **partial derivative** of f and it is denoted by $\delta f/\delta x$ or f_x. Higher order and mixed derivatives are defined by successive differentiation. Taylor's series for functions of several variables is a direct extension of the formula in Theorem 6, although the number of terms in it grows rapidly. For two variables it is

$$f(x,y) = f(c,d)+f_x(x-c)+f_y(y-d)+\frac{1}{2}[f_{xx}(x-c)^2+2f_{xy}(x-c)(y-d)$$

$$+f_{yy}(y-d)^2]+\cdots,$$

where all the partial derivatives are evaluated at the point (c,d).

Theorem 7 (Chain rule for derivatives). *Let $f(x,y.\ldots,z)$ have continuous first partial derivatives with respect to all its variables. Let*

$x = x(t), y = y(t), \ldots, z = z(t)$ be continuous differentiable functions of t. Then

$$g(t) = f(x(t), y(t), \ldots, z(t))$$

is continuously differentiable and

$$g'(t) = f_x x'(t) + f_y y'(t) + \cdots + f_z z'(t).$$

Finally, we state

Theorem 8 (Fundamental theorem of algebra). *Let $p(x)$ be a polynomial of degree $n \geq 1$, that is,*

$$p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n,$$

where the a_i are real or complex numbers and $a_n \neq 0$. Then, there is a complex number ρ so that $p(\rho) = 0$.

II. VECTORS, MATRICES, AND LINEAR EQUATIONS

Vectors are *directed line segments* (they have length, direction, and position) in N-dimensional space. They are considered to be **column vectors** unless otherwise stated, and thus

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}$$

The transpose is indicated by the superscript T, which changes columns into rows and vice versa. Vectors are usually expressed in terms of **a basis**; a standard set $b_1, b_2, \ldots, b_N$ of vectors is chosen, and all other vectors are expressed in terms of the basis $B_i, i = 1, 2, \ldots, N$:

$$\mathbf{y} = y_1\mathbf{b_1} + y_2\mathbf{b_2} + \cdots + y_N\mathbf{b_N}.$$

The coefficients y_i of this representation are the components of y and the representation is commonly written in the compact form

$$\mathbf{y} = (y_1, y_2, \ldots, y_N)^T.$$

The basis vectors define **a coordinate system**, and the components y_i are the coordinates of the point at the end of the vector. The usual basis vectors are of the form $(0, 0, \ldots, 0, 1, 0, \ldots, 0, 0)$. The standard **arithmetic** operations are (for vectors $\mathbf{x}, \mathbf{y}, \mathbf{z}$, and scalar a) as follows:

$$Addition: \begin{array}{l} \mathbf{x}+\mathbf{y} = \mathbf{y}+\mathbf{x} = (\mathbf{x_1}+\mathbf{y_1}, \mathbf{x_2}+\mathbf{y_2}, \ldots, \mathbf{x_N}+\mathbf{y_N})^\mathbf{T} \\ \mathbf{x}-\mathbf{y} = -(\mathbf{y}-\mathbf{x}); \qquad (\mathbf{x}+\mathbf{y})+\mathbf{z} = \mathbf{x}+(\mathbf{y}+\mathbf{z}) \end{array}$$

$$Multiplication(byscalar): \begin{array}{r} a\mathbf{x} = (a\mathbf{x_1}, a\mathbf{x_2}, \ldots, a\mathbf{x_N}) \\ a(\mathbf{x}+\mathbf{y}) = a\mathbf{x}+a\mathbf{y} \end{array}$$

A set $\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_n}$ of vectors is **linearly independent** if no linear combination $\sum_{i=1}^{N} \alpha_i x_i$ of them is zero except for the zero combination; that is

$$\sum_{i=1}^{N} \alpha_i x_i = 0 \text{ implies } \alpha_i = 0 \text{ for all } i.$$

A set of vectors $\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_N}$ spans a space if every vector in that space can be written as a linear combination of the set $\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_N}$. A set of basis vectors must be linearly independent. The dimension of a vector space is the minimal number of vectors required to span the space; each basis of an N dimensional space must have N vectors in it.

The **dot product** or inner product of two vectors $\mathbf{x}$ and $\mathbf{y}$ is

$$\mathbf{x}^T\mathbf{y} = (\mathbf{xy}) = \sum_{i=1}^{N} x_i y_i$$

Two vectors are **orthogonal** vectors (perpendicular) if $\mathbf{x^T y = 0}$. The size of a vector may be measured by the Euclidean norm $||x||_2$ where

$$||\mathbf{x}||_2^2 = \mathbf{x^T x} = \sum_i x_i^2.$$

This is the usual Euclidean length in the case of two or three dimensions. The double bar denotes a **norm**, and another norm frequently convenient is

$$||\mathbf{x}||_\infty = \max_i |x_i|.$$

The angle θ between two vectors is defined from

$$\cos\theta = \frac{\mathbf{x^T y}}{||\mathbf{x}||_\mathbf{2} ||\mathbf{y}||_\mathbf{2}}.$$

The format for vector must mach that of matrices, and so vectors are normally considered to be **column vectors**. To write a vector as

$$y = \begin{pmatrix} 2 \\ 1 \\ 4 \\ -2 \end{pmatrix}$$

complicates the format of the text, and so we write **x** as $(2, 1, 4, -2)^T$ and, in general, write vectors horizontally with the transpose **T** unless the column format is necessary for clarity. At times we also use **row vectors**, which are vectors whose matrix format is actually horizontal, e.g. a row from a matrix.

Once coordinates are introduced for the vectors, then linear functions of vectors can be concretely represented by a two-dimensional array of numbers, **a matrix**:

$$y = \begin{pmatrix} 1 & 6 & -2 \\ 4 & 17 & -12 \\ 0 & 42 & 6.1 \end{pmatrix} = (a_{ij})$$

If **y** is a linear function of $\mathbf{x_i}$ then each component y_k of **y** is a linear function of the components x_i of **x**, and we have for each k that

$$y_i = a_{k1}x_i + a_{k2}x_2 + \ldots + a_{kN}x_N.$$

The coefficients are collected into the matrix **A**, and the linear function is denoted by **Ax**.

The rules for manipulating matrices are those required by the linear mappings.Thus **A**+**B** is to be the representation of the sum of the two linear functions represented by **A** and **B**.One has $\mathbf{A} + \mathbf{B} = \mathbf{C}$ where $c_{ij} = a_{ij} + b_{ij}$. The matrix product **AB** represents the effect of applying the function **B**, then applying the function **A**. The following calculation shows that

$$\mathbf{AB} = \mathbf{C},$$

where $c_{ij} = \sum_k a_{ik} b_{kj}$. We have $\mathbf{y} = \mathbf{B}\mathbf{x}$ and $\mathbf{z} = \mathbf{A}\mathbf{y}$ and want to determine $\mathbf{C}$ so that $\mathbf{z} = \mathbf{C}\mathbf{x}$. We express the relationship in terms of components:

$$y_k = \sum_{j=1}^{N} b_{kj} x_j \qquad\qquad z_i = \sum_{k=1}^{N} a_{ik} y_k.$$

Thus

$$z_i = \sum_{k=1}^{N} a_{ik} \big(\sum_{j=1}^{N} b_{kj} x_j \big) = \sum_{j=1}^{N} \big(\sum_{k=1}^{N} a_{ik} b_{kj} \big) x_j = \sum_{j=1}^{N} c_{ij} x_j,$$

and so c_{ij} is given by the above formula. The (i, j)-th element of $\mathbf{C}$ is the dot product of the row of $\mathbf{A}$ with the j-th column of $\mathbf{B}$. We have the arithmetic rules

$$\begin{aligned} \mathbf{A} + \mathbf{B} &= \mathbf{B} + \mathbf{A} \\ \mathbf{AB} &\neq \mathbf{BA} \quad \textbf{except in special cases}. \end{aligned}$$

The transpose $\mathbf{A^T}$ of $\mathbf{A}$ is obtained by reflecting $\mathbf{A}$ about its diagonal (the a_{ii} elements). That is, $a^T_{ij} = a_{ji}$. The **identity** matrix is all zeros except for 1 on the diagonal:

$$I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

An identity matrix is necessarily square (have the same number of rows and columns). One sees that $\mathbf{IA} = \mathbf{AI} = \mathbf{A}$. The inverse $\mathbf{A^{-1}}$ of $\mathbf{A}$ is a matrix so that $\mathbf{AA^{-1}} = \mathbf{I}$. Not all matrices have an inverse and, indeed, one can have $\mathbf{AB} = \mathbf{0}$ without either $\mathbf{A}$ or $\mathbf{B}$ being the zero matrix:

$$\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ -1 & -1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

If $\mathbf{A}$ has an inverse, then we say $\mathbf{A}$ is nonsingular. We have the following equivalent statements for a square matrix $\mathbf{A}$:

- $\mathbf{A}$ is nonsingular
- $\mathbf{A}^{-1}$exists
- The columns of $\mathbf{A}$ are linearly independent
- The rows of $\mathbf{A}$ are linearly independent
- $\mathbf{Ax} = \mathbf{0}$ implies that $\mathbf{x} = \mathbf{0}$.

The **linear equations** problem is this: Given $\mathbf{A}$ and $\mathbf{b}$, find the vector $\mathbf{x}$ so that $\mathbf{Ax} = \mathbf{b}$. If $\mathbf{A}$ is nonsingular (this makes $\mathbf{A}$ square), then this problem has always a unique solution for each $\mathbf{b}$. If $\mathbf{A}$ has more rows than columns (there are more equations than variables), then the problem is usually unsolvable, and if $\mathbf{A}$ has more columns than rows. there are usually infinitely many solutions. A system of equations is **homogeneous** if the right side is zero, for example $\mathbf{Ax} = \mathbf{0}$. The oldest and standard method for solving this problem is by **Gauss elimination** (forget Cramer's rules - why ?). By Gauss elimination one gets **upper triangular** matrix (with all elements below the diagonal are zero) and **lower triangular** matrix means that all elements above the diagonal are zero. After elimination process, one uses **back substitution** (for upper triangular) or **forward substitution** (for lower triangular matrix, in order to solve quickly the system.

Gauss elimination is illustrated by the following example. Given the system

$$\begin{array}{cccc} 2x_1 & +2x_2 & +4x_3 & = 5 \\ 6x_1 & -x_2 & +x_3 & = 7 \\ 4x_1 & -10x_2 & -12x_3 & = -4 \end{array}$$

By subtracting 3 times row 1 from row 2, and by subtracting 2 times row 1 from row 3, one gets

$$\begin{array}{cccc} 2x_1 & +2x_2 & +4x_3 & = 5 \\ & -7x_2 & -11x_3 & = -8 \\ & -14x_2 & -20x_3 & = -14, \end{array}$$

and then, by subtracting 2 times row 2 from row 3,

$$\begin{array}{llll} 2x_1 & +2x_2 & +4x_3 & = 5 \\ & -7x_2 & -11x_3 & = -8 \\ & & 2x_3 & = 2. \end{array}$$

Finally, by back substitution, one gets the solutions:

$$x_3 = \frac{2}{2} = 1$$

$$-7x_2 = -8 + 11x_3 = 3, \text{ so } x_2 = -\frac{3}{7}$$

$$2x_1 = 5 - 2x_2 - 4x_3 = \frac{13}{7} \text{ so } x_3 = \frac{13}{14}.$$

A matrix is **permutation matrix** if each element is a 0 or 1 and there is exactly one 1 per row or column, for example:

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

Multiplication by a permutation matrix, on the left or right, has the effect of permuting or interchanging the rows or columns of the matrix. This property gives them their name, and they are useful in formulas to indicate interchanges of rows and columns; they are rarely used in actual calculations.

In some occasions we refer to an **eigenvalues** of the matrix $\mathbf{A}$. This is a number λ such that $\mathbf{Ax} = \lambda\mathbf{x}$ for some nonzero vector $\mathbf{x}$; the vector $\mathbf{x}$ is called an **eigenvector**. A linear mapping applied to an eigenvector simply multiplies the eigenvector by the constant λ, the eigenvalue. An N by N matrix has N eigenvalues and normally, but not always, has N eigenvectors. The **spectral radius** $\rho(\mathbf{A})$ of A is the largest of the absolute values of the eigenvalues of $\mathbf{A}$. The spectral radius plays a fundamental role in the convergence of iterations involving matrices.

The **norm of matrix** will be mentioned some later, regarding convergence process of iteration procedure for solving a system of linear algebraic equations.

III. PROGRAMMING

There are several areas of knowledge about programming that are needed for scientific computation. These include knowledge about:

- The programming language (FORTRAN, Pascal, C, Java, Mathematica (MatCAD, **Matlab**).
- The computer system in which the language runs
- Program debugging and verifying the correctness of results
- Computation organization and expressing them clearly.

Debugging programs is an art as well as a science, and it must be learned through practice. There are several effective tactics to use, like:

- Intermediate output
- Consultations about program with experienced user
- Use compiler and debugging tools.

Some abilities of compilers:

- Cross-reference tables
- Tracing
- Subscript checking
- Language standards checking.

Some hints:

- Use lots of comments
- Use meaningful names for variables
- Make the types of variables obvious
- Use simple logical control structures
- Use program packages and systems (Mathematica, Matlab) wherever possible
- Use structured programming
- Use (if possible) OOP technics for technical problems.

IV. NUMERICAL SOFTWARE

There are several journals that publish individual computer programs:

- ACM Transactions on Mathematical Software (IMSL,International Mathematical Scientific Library)
- Applied Statistics
- BIT
- The Computer Journal
- Numerische Mathematik

The ACM Algorithms series contains more than thousand items and is available as the **Collected Algorithms of the Association for Computing Machinery**.

Three general libraries of programs for numerical computations are widely available:

IMSL-	IMSL, Inc.
NAG-	Numerical Algorithms Group, Oxford University
SSP-	Scientific Subroutine Package, IBM Corporation

There are also several general statistical program libraries, like:

BMD-	BioMedical Department, UCLA
GPSS-	

SOFTWARE PACKAGES

There are a substantial number of important, specialized software packages. Most of the packages listed below are available from IMSL, Inc.

MP-	Multiple Precision Arithmetic Package
BLAS-	Basic Linear Algebra Subroutines
DEPACK-	Differential Equation Package
DSS-	Differential System Simulator
EISPACK-	Matrix Eigensystems Routines
FISHPACK-	Routines for the Helmholtz Problem in Two or Three Dimensions
FUNPACK-	Special Function Subroutines
ITPACK-	Iterative Methods
LINPACK-	Linear Algebra Package
PPPACK-	Piecewise Polynomial and Spline Routines
ROSEPACK-	Robust Statistics Package
ELLPACK-	Elliptic Partial Differential Equations
SPSS-	Statistical Package for the Social Sciences.

User interface to the IMSL library:)

PROTRAN-

V. CASE STUDY: ERRORS, ROUND-OFF, AND STABILITY

V.1. Solve quadratic formula

$$ax^2 + bx + c = 0$$

with $5, 10, 15, \ldots 100$ decimal digits using FORTRAN and Mathematica code. Take $a = 1,\ c = 2, b = 5.2123(10)105.2123$. Use the following two codes:

```
DIS=SQRT(B*B-4.*A*C)      DIS=SQRT(B*B-4.*A*C)
X1=(-B+DIS)/(2*A)         IF(B.LT.0) THEN
X2=(-B-DIS)/(2*A)         X1=(-B+DIS)/(2*A)
                          ELSE
                          X1=(-B-DIS)/(2*A)
                          ENDIF
                          X2=C/X1
```

Compare the obtained results.

There are two important lessons to be learned from example **V.1.**:

1. *Round-off error can completely ruin a short, simple computation.*
2. *A simple change in the method might eliminate adverse round-off effects.*

V.2. Stability

Some computations are very sensitive to round-off and others are not. In the example given above, sensitivity to round-off was eliminated by changing the formula or method. This is always possible; there are many problems which are inherently sensitive to round-off and any other uncertainties. Thus we must distinguish between sensitivity of **methods** and sensitivity inherent in **problems**.

The word **stability** appears during numerical computations and refers to continuous dependence of a solution on the **data** of the problem or **method**. If one says that a method is **numerically unstable**, one means that the round-off effects are grossly magnified by the method. Stability also has precise technical meaning (not always the same) in different areas as well as in this general one.

Solving differential equations usually leads to difference equations, like

$$x_{i+2} = -(13/6)x_{i+1} + (5/2)x_i.$$

Here, the sequence $x_1, x_2, \dots$ is defined, and for given initial conditions x_1 and x_2 of differential equation, we get the initial conditions for difference equation. For example, $x_1 = 30,\ x_2 = 25$. Computing in succession for $4, 8, 16, 32, 64$ decimal digits gives the results that can be compared with the exact one, $x_i = 36/(5/6)^i$. (Compute in Mathematica, using $N[x[I+2], k]$, where $k = 4, 8, 16, 32, 64$ number of decimal digits).

i	4	8	16	True value
1	30.00	30.00	30.00	30.00
2	25.00	25.00	25.00	25.00
3	20.83	20.8333	20.8333	20.8333
4	17.36	17.3611	17.3611	17.3611
5	14.46	14.4676	14.4676	14.4676
6	12.07	12.0563	12.0563	12.0563
7	10.00	10.0470	10.0469	10.0469
8	8.518	8.3724	8.3724	8.3724
9	6.541	6.9773	6.9770	6.9770

10	7.121	5.8133	5.8142	5.8142
11	.925	4.8478	4.8452	4.8452
12	15.790	4.0296	4.0376	4.0376
13	-31.920	3.3888	3.3647	3.3647
14	108.700	2.7318	2.8039	2.8039
16	954.600	1.2978	1.9472	1.9472
18	8576.000	-4.4918	1.3522	1.3522
20	77170.000	-51.6565	.9390	.9390
22	6.9×10^5	-472.7080	.6521	.6521
25	-1.8×10^7	12781.1000	.3776	.3774
28	5.0×10^8	-345079.0000	.2134	.2184
30	4.5×10^9	-3.1×10^6	.1071	.1517
35	-1.1×10^{12}	7.5×10^8	10.8822	.0609
40	-1.1×10^{14}	-1.8×10^{11}	-2629.5300	.0245
50	1.5×10^{19}	-1.0×10^{16}	-1.5×10^8	.0039
75	1.3×10^{31}	9.2×10^{27}	1.3×10^{20}	.00

This difference equation is unstable and one can see that the computation quickly ”blows up”. One nice thing about unstable computation is that they usually produce huge, nonsense numbers that one is not tempted to accept as correct. However, imagine that one wanted only 30 terms of the x_i and was using the computer with 16 decimal digits. How would one know that the last term is in error by 50 percent ?

The word **condition** is used to describe the sensitivity of problems to uncertainty. Imagine the solution of a problem being obtained by evaluation a function $f(x)$. Then, if x is changed a little to $x + \delta x$, the value $f(x)$ also changes. The relative **condition number** of this change is

$$\frac{|f(x+\delta x) - f(x)|}{|f(x)|} / |\frac{\delta x}{x}|,$$

or

$$\frac{f(x+\delta x)-f(x)}{\delta x} \times \frac{x}{f(x)},$$

and, for δx very small, condition number c is

$$c \sim \frac{xf'(x)}{f(x)}.$$

This number estimates how much an uncertainty in the data x of a problem is magnified in its solution $f(x)$. If this number is large, then the problem is said to be **ill-conditioned** or poorly conditioned.

The given formula is for the simplest case of a function of a single variable; it is not easy to obtain such formulas for more complex problems that depend on many variables of different types. We can see three different ways that a problem can have a large condition number:

1. *$f'(x)$ may be large while x and $f(x)$ are not*;

If we evaluate $1+\sqrt{|x-1|}$ for x very close to 1, then x and $f(x)$ are nearly 1, but $f'(x)$ is large and the computed value is highly sensitive to change in x.

2. *$f(x)$ may be small while x and $f'(x)$ are not*;

The Taylor's series for $\sin x$ near π or $\exp^{-x}$ with x large exhibit this form of ill conditioning.

3. *x may be large while $f'(x)$ and $f(x)$ are not*;

The evaluation of $\sin x$ for x near 1000000π is poorly conditioned.

One can also say that computation is ill-conditioned and this is the same as saying it is numerically unstable. The condition number gives more information than just saying something is numerically unstable. It is rarely possible to obtain accurate values for condition numbers but one rarely needs much accuracy; an order of magnitude is often enough to know.

Note that is almost impossible for a method to be numerically stable for an ill-conditioned problem.

Example 5.1 An **ill-conditioned line intersection problem** consists in computing the point of intersection P of two nearly parallel lines. It is clear that a minor change in one line changes the point of intersection to $(P + \delta P)$ which is far from P. A mathematical model of this problem is obtained by introducing a coordinate system and writing equations

$$y = a_1 x + b_1$$
$$y = a_2 x + b_2$$

what leads to solving a system of equations

$$a_1 x - y = -b_1$$
$$a_2 x - y = -b_2$$

with the a_1 and a_2 nearly equal since the lines are nearly parallel. This numerical problem is unstable or ill-conditioned, as it reflects the ill-conditioning of the original problem.

A mathematical model is obtained by introducing a **coordinate system**. Any two vectors will do for a basis, and if we chose to use the unusual basis

$$\mathbf{b_1} = (0.5703958095, 0.8213701274)$$
$$\mathbf{b_2} = (0.5703955766, 0.8213701274)$$

then every vector $\mathbf{x}$ can be expressed as

$$\mathbf{x} = x\mathbf{b_1} + y\mathbf{b_2}$$

so that the equations of the two lines in this coordinate system are

$$y = -0.0000000513 + 0.9999998843x$$

$$y = -0.0000045753 + 1.000001596x$$

with the point of intersection P with coordinates $(-0.8903429339, 0.8903427796)$. Note that mathematical model is very ill-conditioned; a change of 0.0000017117 in the data makes the two lines parallel, with no solution.

The poor choice of a basis in the given example made the problem poorly conditioned. In more complex problems it is not so easy to see that a poor choice has been made. In fact, a poor choice is sometimes the most natural thing to do. For example, in problems involving the polynomials, one naturally takes vectors based on $1, x, x^2, \ldots, x^n$ as a basis, but there are terribly ill-conditioned even for n moderate in size.

Example 5.2 System of equations (input information)

$$2x + 6y = 8$$

$$2x + 6.0001y = 8.0001$$

have a solutions (output information) $x = 1,\ y = 1$. If the coefficients of second equation slightly change, i.e. if one takes the equation

$$2x + 5.99999y = 8.00002,$$

the solutions are $x = 10,\ y = -2$. This is typical round-off error.

Errors in methods occur usually because in numerical mathematics the problem to be solved is replaced by another one, closed to original, which is easier to solve.

Example 5.3 Integral $\int_a^b f(x)dx$ can be approximately calculated, for example, by replacing the function f by some polynomial P on segment $[a, b]$, which is in some sense close to given function. However, for approximative calculation it is possible to use the sum

$$\sum_{i=1}^{n} f(x_i)\Delta x_i.$$

In both cases the method error occurs.

In some sense, the round-off error are also method errors. Sum of all errors makes the total error.

V.3. Case study: Calculation of π

Using five following algorithms, calculate π in order to illustrate the various effects of round-off on somewhat different computations.

Algorithm 1. Infinite alternate series

$$\pi = 4(1 - 1/3 + 1/5 - 1/7 + 1/9 - \cdots)$$

Algorithm 2. Taylor's series of $\arcsin(1/2) = \pi/6$

$$\pi = 6(0.5 + \frac{(0.5)^2}{2\times 3} + \frac{1\times 3(0.5)^4}{2\times 4\times 5} + \frac{1\times 3\times 5(0.5)^6}{2\times 4\times 6\times 7} + \cdots)$$

Algorithm 3. Archimedes' method. Place $4, 8, 16, \ldots, 2^n$ triangles inside a circle. The area od each triangle is $1/2\sin(\theta)$. The values of $\sin(\theta)$ are computed by the half angle formula

$$\sin(\theta) = \sqrt{[1 - \cos(2\theta)]/2}$$

and

$$\cos(\theta) = \sqrt{1 - \sin^2\theta}.$$

The calculation is initialized by $\sin(\pi/4) = \cos(\pi/4) = 1/\sqrt{2}$. As the number of triangles grows, they fill up the circle and their total area approaches π. (Archimed carried a similar procedure by hand with 96 triangles and obtained

$$3.1409\ldots = 3\frac{1137}{8069} < \pi < 3\frac{1335}{9347} = 3.1428\ldots)$$

Algorithm 4. Instead of inscribing triangles in a circle, we inscribe trapezoids in a quarter circle. As a number of trapezoids increases, the sum of their areas approaches $\pi/4$.

Algorithm 5. Monte Carlo integration.
(Monte Carlo integration for $\int_0^2 \frac{2}{1+x}\ dx$ is proceeded by choosing a pair (x, y) at random with x, y in $[0, 2]$, and compare y with $2/(1+x)$. If $y \leq 2/(1+x)$ then the point (x, y) is under the curve $y = 2/(1+x)$ and variable SUM is increased by `1`. After `M` pairs, the integral is estimated by the fraction `SUM/M` of points that are under the curve).

V.4. How to estimate errors and uncertainty

One almost newer knows the error in a computed result unless one already knows the true solution, and so one must settle for estimates of the error. There are three basic approaches to error estimates. The first is **forward error analysis**, when one uses the theory of the numerical method plus information about the uncertainty in the problem and attempts to predict the error in the computed result. The information one might use includes

- the size of round-off,
- the measurement errors in problem data,
- the truncation errors in obtaining the numerical model from the mathematical model,
- the differences between the mathematical model and the original physical model.

The second approach is **backward error analysis**, where one takes a computed solution and sees how close it comes to solving the original problem. The backward error is is often called the the **residual** in equations. This approach requires that the problems involve satisfying some conditions (such as an equation) which can be tested with a trial solution. This prevents it from being applicable to all numerical computations, e.g. numerically estimating the value of π or the value of an integral.

The third approach is **experimental error analysis**, where one experiments with changing the computations, the method, or the data to see the effect they have on the results. If one truly wants certainty about the accuracy of a computed value, then one should give the problem to two (or even more) different groups and ask to solve it. The groups are not allowed to talk together, preventing a wrong idea from being passing around.

The relationship between these three approaches could be illustrated graphically, as given in the following figure.

(figure missed)