Numerical Mathematics
Computer Arithmetic & Algebraic Equations

Pieter Collins

Department of Knowledge Engineering

Maastricht University

pieter.collins@maastrichtuniversity.nl

KEN1540, Block 5, April-May 2021

Organisation

Introduction

Numerical mathematics deals with methods for the solution of problems in continuous mathematics which can be implemented on a digital computer.

Typically, use floating-point arithmetic to perform approximate calculations on real numbers.

Based on ideas and techniques from calculus and linear algebra, but yields numerical values for the solution of specific problems, rather than general formulae.
Important part of data science:

- estimate models from data,
- generate data as predictions from models, and
- compute properties of data directly.

Course

Topics

1. Computer Arithmetic & Algebraic Equations
2. Numerical Solution of Differential Equations
3. Polynomial (and Spline) Interpolation
4. Numerical Integration and Differentiation
5. Least-Squares Approximation
6. Numerical Linear Algebra

Classes Per topic: 2-3h lectures; 3-4h tutorials.
Plus: 2h revision tutorial.

Grading

80% Written exam (with calculator),

20% Homework programming assignments (4$\times$5%).

10% Homework questions (preparation for tutorial).

Regulations

Assignments The graded assignments are *individual* assignments, and follow standard DKE regulations as such.

Guidelines:

- You may not receive help solving a graded assignment from anybody else, including working together or sharing code.
- Any sources (other than the textbook, slides, the Student Portal, and other material presented in-class) must be referenced.
- You may work with other students to understand the material and on non-graded assignments (and are encouraged to do so).
- If you have written previously written code for a related problem together with other students, you should re-write the code yourself for the graded assignment.
- If you are unsure whether any work you have done together is allowed, you should declare this on your homework.

Homeworks

Homeworks *The homeworks are a vital part of the course!!! There is a* very *strong correlation between doing the homeworks and passing the course!!!!!*

Preparation You should attempt a significant proportion of the homeworks before the tutorials. Part of the grade (for DKE students) is based on preparation. This way, we can spend time going over questions which you find difficult.

Learning This course has a lot of formulae, which may seem hard at first, but don't panic! With practise, most of the questions should become routine. But you do *really* need to put the work in!

Computer use

Tutorials Bring your computer to the tutorial classes!

Matlab You are expected to have access to a computer with Matlab.

Alternatively, you may use a Matlab clone, such as GNU Octave or Scilab.

Online Learning

Instead of giving lectures in class time, I will pre-record lecture snippets.

You should read the slides and watch the snippets *before* the first class on a topic.

All class-time will be run as tutorial sessions. This will give you the maximum time to ask questions and receive feedback.

In general, during tutorials, I will answer common questions in a "plenary" session, while the teaching assistants provide individual help.

Online teaching is new to me (and new-ish to you), so this approach may change if it seems not to be working!

Mathematical Preliminaries

Calculus

- Definition of limit, derivative and integral.
- Differentiation including product and chain rules.
- Integrals of polynomials.
  - *No need to be able to perform complex integration :)*
- Intermediate value theorem and mean value theorem.
- We will cover Taylor series later!

Rate of convergence

Positive limits

Write $a_n \searrow 0$ or $a_n \to 0^+$ if all $a_n \geq 0$ and $\lim_{n\to\infty} a_n = 0$.

Big-O Notation

If $a_n, b_n \searrow 0$ as $n \to \infty$, say $a_n = O(b_n)$ if there is a constant $C > 0$ such that $a_n \leq C b_n$ for all n.

If $f, g \searrow 0$ as $h \to 0$, say $f = O(g)$ if there is a constant $C > 0$ such that $f(h) \leq C g(h)$ whenever $|h| < 1$.

Little-o Notation

Say $a_n = o(b_n)$ if $\lim_{n\to\infty} a_n/b_n = 0$.

Say $f = o(g)$ if $\lim_{h\to 0} f(h)/g(h) = 0$.

Example The sequence $a_n = \frac{2n}{n+3}$ satisfies $|a_n - 2| = \frac{6}{n+1} \leq 6 \times \frac{1}{n}$.
Hence $a_n - 2 = O(1/n)$. Say a_n converges to 2 at *rate* $O(1/n)$.

Example If $f'(x) = 0$, then $f(x+h) - f(x) = o(h)$.

Computer Arithmetic

Arithmetic in Matlab

Let’s try doing some simple arithmetic with Matlab:

Arithmetic in Matlab

Let's try doing some simple arithmetic with Matlab:

```
>> 0.1+0.3+0.6
```

Arithmetic in Matlab

Let's try doing some simple arithmetic with Matlab:

```
>> 0.1+0.3+0.6
ans = 1
```

Arithmetic in Matlab

Let's try doing some simple arithmetic with Matlab:

```
>> 0.1+0.3+0.6
ans = 1
```

This is what we expect.

Arithmetic in Matlab

Let's try doing some simple arithmetic with Matlab:

```
>> 0.1+0.3+0.6
ans = 1
```

This is what we expect.

```
>> 0.6+0.3+0.1
```

Arithmetic in Matlab

Let's try doing some simple arithmetic with Matlab:

```
>> 0.1+0.3+0.6
ans = 1
```

This is what we expect.

```
>> 0.6+0.3+0.1
ans = 1.0000
```

Arithmetic in Matlab

Let's try doing some simple arithmetic with Matlab:

```
>> 0.1+0.3+0.6
ans = 1
```

This is what we expect.

```
>> 0.6+0.3+0.1
ans = 1.0000
```

Also as expected. But why this time `1.0000` instead of `1`?

Arithmetic in Matlab

Let's try doing some simple arithmetic with Matlab:

```
>> 0.1+0.3+0.6
ans = 1
```

This is what we expect.

```
>> 0.6+0.3+0.1
ans = 1.0000
```

Also as expected. But why this time `1.0000` instead of `1`?

Substract 1 from the answer:

Arithmetic in Matlab

Let's try doing some simple arithmetic with Matlab:

```
>> 0.1+0.3+0.6
ans = 1
```

This is what we expect.

```
>> 0.6+0.3+0.1
ans = 1.0000
```

Also as expected. But why this time `1.0000` instead of `1`?

Substract 1 from the answer:

```
>> (0.6+0.3+0.1)-1
```

Arithmetic in Matlab

Let's try doing some simple arithmetic with Matlab:

```
>> 0.1+0.3+0.6
ans = 1
```

This is what we expect.

```
>> 0.6+0.3+0.1
ans = 1.0000
```

Also as expected. But why this time `1.0000` instead of `1`?

Substract 1 from the answer:

```
>> (0.6+0.3+0.1)-1
ans = -1.1102e-16
```

Arithmetic in Matlab

Let's try doing some simple arithmetic with Matlab:

```
>> 0.1+0.3+0.6
ans = 1
```

This is what we expect.

```
>> 0.6+0.3+0.1
ans = 1.0000
```

Also as expected. But why this time `1.0000` instead of `1`?

Substract 1 from the answer:

```
>> (0.6+0.3+0.1)-1
ans = -1.1102e-16
```

The answer is not exactly 0! But why does this occur??

Computer arithmetic

Try displaying more digits in Matlab:

Computer arithmetic

Try displaying more digits in Matlab:

```
>> format long
```

Computer arithmetic

Try displaying more digits in Matlab:

```
>> format long

>> 0.6+0.3+0.1
ans = 1.000000000000000

>> (0.6+0.3+0.1)-1
ans = -1.110223024625157e-16
```

Computer arithmetic

Try displaying more digits in Matlab:

```
>> format long
>> 0.6+0.3+0.1
ans = 1.000000000000000
>> (0.6+0.3+0.1)-1
ans = -1.110223024625157e-16
```

Try using Python:

Computer arithmetic

Try displaying more digits in Matlab:

```
>> format long

>> 0.6+0.3+0.1
ans = 1.000000000000000

>> (0.6+0.3+0.1)-1
ans = -1.110223024625157e-16
```

Try using Python:

```
>>> 0.6+0.3+0.1
0.9999999999999999
```

Computer arithmetic

Try displaying more digits in Matlab:

```
>> format long
>> 0.6+0.3+0.1
ans = 1.000000000000000
>> (0.6+0.3+0.1)-1
ans = -1.110223024625156e-16
```

Try using Python:

```
>>> 0.6+0.3+0.1
0.9999999999999999
>>> (0.6+0.3+0.1)-1
-1.1102230246251565e-16
```

Computer arithmetic

Try displaying more digits in Matlab:

```
>> format long
>> 0.6+0.3+0.1
ans = 1.000000000000000
>> (0.6+0.3+0.1)-1
ans = -1.110223024625156e-16
```

Try using Python:

```
>>> 0.6+0.3+0.1
0.9999999999999999
>>> (0.6+0.3+0.1)-1
-1.1102230246251565e-16
```

Now we see that $0.6 + 0.3 + 0.1$ is computed to a value different from 1!

Computer arithmetic

Try displaying more digits in Matlab:

```
>> format long
>> 0.6+0.3+0.1
ans = 1.000000000000000
>> (0.6+0.3+0.1)-1
ans = -1.110223024625156e-16
```

Try using Python:

```
>>> 0.6+0.3+0.1
0.9999999999999999
>>> (0.6+0.3+0.1)-1
-1.1102230246251565e-16
```

Now we see that $0.6 + 0.3 + 0.1$ is computed to a value different from 1!

Matlab does not display sufficient digits to distinguish computed value from 1, whereas Python displays enough digits to read a number back in.

Computer arithmetic

Try displaying more digits in Matlab:

```
>> format long
>> 0.6+0.3+0.1
ans = 1.000000000000000
>> (0.6+0.3+0.1)-1
ans = -1.110223024625157e-16
```

Try using Python:

```
>>> 0.6+0.3+0.1
0.9999999999999999
>>> (0.6+0.3+0.1)-1
-1.1102230246251565e-16
```

Now we see that $0.6 + 0.3 + 0.1$ is computed to a value different from 1!

Matlab does not display sufficient digits to distinguish computed value from 1, whereas Python displays enough digits to read a number back in.

We shall see that the computed value of $0.6 + 0.3 + 0.1$ is *exactly* $1 - 2^{-53}$.

Numbers and representations

Numbers What kinds of numbers are there?

Numbers and representations

Numbers What kinds of numbers are there?

Integers, Rationals, Reals, Complex, ...

Numbers and representations

Numbers What kinds of numbers are there?

Integers, Rationals, Reals, Complex, ...

Integers What are integers, and how can we describe them?

Numbers and representations

Numbers What kinds of numbers are there?

Integers, Rationals, Reals, Complex, ...

Integers What are integers, and how can we describe them?

Positive integers *count* “*how many*” objects there are in a finite set.

Numbers and representations

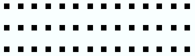
Numbers What kinds of numbers are there?
Integers, Rationals, Reals, Complex, ...

Integers What are integers, and how can we describe them?
Positive integers *count* "*how many*" objects there are in a finite set.
Decimal "42", binary "101010_2", and English "forty-two" are different *representations* of the same number.

Numbers and representations

Numbers What kinds of numbers are there?
Integers, Rationals, Reals, Complex, ...

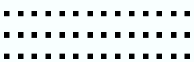
Integers What are integers, and how can we describe them?
Positive integers *count* "*how many*" objects there are in a finite set.
Decimal "42", binary "101010_2", and English "forty-two" are different *representations* of the same number.

The each of these descriptions *means* "as many as ".

Numbers and representations

Numbers What kinds of numbers are there?
Integers, Rationals, Reals, Complex, ...

Integers What are integers, and how can we describe them?
Positive integers *count* "*how many*" objects there are in a finite set.
Decimal "42", binary "101010_2", and English "forty-two" are different *representations* of the same number.

The each of these descriptions *means* "as many as ".
Even though there are *infinitely many* integers, we can specify any integer with a *finite* amount of data.

Numbers and representations

Numbers What kinds of numbers are there?
Integers, Rationals, Reals, Complex, ...

Integers What are integers, and how can we describe them?
Positive integers *count* "*how many*" objects there are in a finite set.
Decimal "42", binary "101010_2", and English "forty-two" are different *representations* of the same number.

The each of these descriptions *means* "as many as ".
Even though there are *infinitely many* integers, we can specify any integer with a *finite* amount of data.

Real numbers What are real numbers, and how can we describe them?

Numbers and representations

Numbers What kinds of numbers are there?
Integers, Rationals, Reals, Complex, ...

Integers What are integers, and how can we describe them?
Positive integers *count* "*how many*" objects there are in a finite set.
Decimal "42", binary "101010_2", and English "forty-two" are different *representations* of the same number.

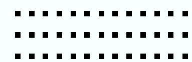
The each of these descriptions *means* "as many as ⋮⋮⋮⋮⋮⋮⋮⋮⋮⋮⋮⋮⋮⋮".
Even though there are *infinitely many* integers, we can specify any integer with a *finite* amount of data.

Real numbers What are real numbers, and how can we describe them? Positive real numbers *measure* "*how much*", "*where*", or "*when*".

Numbers and representations

Numbers What kinds of numbers are there?
Integers, Rationals, Reals, Complex, ...

Integers What are integers, and how can we describe them?
Positive integers *count* "*how many*" objects there are in a finite set.
Decimal "42", binary "101010_2", and English "forty-two" are different *representations* of the same number.

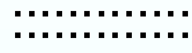
The each of these descriptions *means* "as many as ".
Even though there are *infinitely many* integers, we can specify any integer with a *finite* amount of data.

Real numbers What are real numbers, and how can we describe them? Positive real numbers *measure* "*how much*", "*where*", or "*when*".
Representations include symbolic "$\sqrt{2}$" and decimal "$1.414213562373\cdots$".

Numbers and representations

Numbers What kinds of numbers are there?
Integers, Rationals, Reals, Complex, ...

Integers What are integers, and how can we describe them?
Positive integers *count* "*how many*" objects there are in a finite set.
Decimal "42", binary "101010_2", and English "forty-two" are different *representations* of the same number.

The each of these descriptions *means* "as many as ".
Even though there are *infinitely many* integers, we can specify any integer with a *finite* amount of data.

Real numbers What are real numbers, and how can we describe them? Positive real numbers *measure* "*how much*", "*where*", or "*when*".
Representations include symbolic "$\sqrt{2}$" and decimal "$1.414213562373\cdots$".

Real numbers are *uncountable*, would need an *infinite* amount of data for a representation capable of describing *all* of them!

Decimal expansions of real numbers

Rational Rational numbers have *terminating* or *recurring* decimal expansions.

Decimal expansions of real numbers

Rational Rational numbers have *terminating* or *recurring* decimal expansions.

e.g. $\frac{1}{4} = 0.25, \quad \frac{1}{6} = 0.1\dot{6}, \quad \frac{1}{7} = 0.\dot{1}4285\dot{7} = 0.142857142857\cdots.$

Decimal expansions of real numbers

Rational Rational numbers have *terminating* or *recurring* decimal expansions.

e.g. $\frac{1}{4} = 0.25, \quad \frac{1}{6} = 0.1\dot{6}, \quad \frac{1}{7} = 0.\dot{1}4285\dot{7} = 0.142857142857\cdots.$

— Note that some numbers have two *different*, but *equal* representations!

Decimal expansions of real numbers

Rational Rational numbers have *terminating* or *recurring* decimal expansions.

e.g. $\frac{1}{4} = 0.25, \quad \frac{1}{6} = 0.1\dot{6}, \quad \frac{1}{7} = 0.\dot{1}4285\dot{7} = 0.142857142857\cdots.$

— Note that some numbers have two *different*, but *equal* representations!

e.g. $0.25 = 0.25\dot{0} = 0.24\dot{9} = 0.249999\cdots.$

Decimal expansions of real numbers

Rational Rational numbers have *terminating* or *recurring* decimal expansions.

e.g. $\frac{1}{4} = 0.25, \quad \frac{1}{6} = 0.1\dot{6}, \quad \frac{1}{7} = 0.\dot{1}4285\dot{7} = 0.142857142857\cdots.$

— Note that some numbers have two *different*, but *equal* representations!
e.g. $0.25 = 0.25\dot{0} = 0.24\dot{9} = 0.249999\cdots.$

Irrational Most real numbers are irrational and don't have a repeating decimal.

Decimal expansions of real numbers

Rational Rational numbers have *terminating* or *recurring* decimal expansions.

e.g. $\frac{1}{4} = 0.25, \quad \frac{1}{6} = 0.1\dot{6}, \quad \frac{1}{7} = 0.\dot{1}4285\dot{7} = 0.142857142857\cdots.$

— Note that some numbers have two *different*, but *equal* representations!

e.g. $0.25 = 0.25\dot{0} = 0.24\dot{9} = 0.249999\cdots.$

Irrational Most real numbers are irrational and don't have a repeating decimal.

e.g. $\sqrt{2} = 1.41421356\cdots$, $e = 2.718281828\cdots$, $\pi = 3.1415926535\cdots.$

Decimal expansions of real numbers

Rational Rational numbers have *terminating* or *recurring* decimal expansions.

e.g. $\frac{1}{4} = 0.25$, $\frac{1}{6} = 0.1\dot{6}$, $\frac{1}{7} = 0.\dot{1}4285\dot{7} = 0.142857142857\cdots$.

— Note that some numbers have two *different*, but *equal* representations!
e.g. $0.25 = 0.25\dot{0} = 0.24\dot{9} = 0.249999\cdots$.

Irrational Most real numbers are irrational and don't have a repeating decimal.
e.g. $\sqrt{2} = 1.41421356\cdots$, $e = 2.718281828\cdots$, $\pi = 3.1415926535\cdots$.

— But each of the numbers above *can* be represented by a *finite* formula

Decimal expansions of real numbers

Rational Rational numbers have *terminating* or *recurring* decimal expansions.

e.g. $\frac{1}{4} = 0.25, \quad \frac{1}{6} = 0.1\dot{6}, \quad \frac{1}{7} = 0.\dot{1}4285\dot{7} = 0.142857142857\cdots.$

— Note that some numbers have two *different*, but *equal* representations!

e.g. $0.25 = 0.25\dot{0} = 0.24\dot{9} = 0.249999\cdots.$

Irrational Most real numbers are irrational and don't have a repeating decimal.

e.g. $\sqrt{2} = 1.41421356\cdots$, $e = 2.718281828\cdots$, $\pi = 3.1415926535\cdots$.

— But each of the numbers above *can* be represented by a *finite* formula

e.g. $e = \sum_{n=0}^{\infty} \frac{1}{n!} = \lim_{n\to\infty}(1 + \frac{1}{n})^n$.

Decimal expansions of real numbers

Rational Rational numbers have *terminating* or *recurring* decimal expansions.

e.g. $\frac{1}{4} = 0.25$, $\frac{1}{6} = 0.1\dot{6}$, $\frac{1}{7} = 0.\dot{1}4285\dot{7} = 0.142857142857\cdots$.

— Note that some numbers have two *different*, but *equal* representations!
e.g. $0.25 = 0.25\dot{0} = 0.24\dot{9} = 0.249999\cdots$.

Irrational Most real numbers are irrational and don't have a repeating decimal.
e.g. $\sqrt{2} = 1.41421356\cdots$, $e = 2.718281828\cdots$, $\pi = 3.1415926535\cdots$.

— But each of the numbers above *can* be represented by a *finite* formula
e.g. $e = \sum_{n=0}^{\infty} \frac{1}{n!} = \lim_{n\to\infty}(1+\frac{1}{n})^n$.

— And we can write a program *compute* arbitrarily many digits of the decimal expansion!

Decimal expansions of real numbers

Rational Rational numbers have *terminating* or *recurring* decimal expansions.

e.g. $\frac{1}{4} = 0.25, \quad \frac{1}{6} = 0.1\dot{6}, \quad \frac{1}{7} = 0.\dot{1}4285\dot{7} = 0.142857142857\cdots$.

— Note that some numbers have two *different*, but *equal* representations!
e.g. $0.25 = 0.25\dot{0} = 0.24\dot{9} = 0.249999\cdots$.

Irrational Most real numbers are irrational and don't have a repeating decimal.
e.g. $\sqrt{2} = 1.41421356\cdots$, $e = 2.718281828\cdots$, $\pi = 3.1415926535\cdots$.

— But each of the numbers above *can* be represented by a *finite* formula
e.g. $e = \sum_{n=0}^{\infty} \frac{1}{n!} = \lim_{n\to\infty}(1 + \frac{1}{n})^n$.

— And we can write a program *compute* arbitrarily many digits of the decimal expansion!

Uncomputable For "almost all" real numbers, there is *no* finite description of the decimal expansion!

Decimal expansions of real numbers

Rational Rational numbers have *terminating* or *recurring* decimal expansions.

e.g. $\frac{1}{4} = 0.25, \quad \frac{1}{6} = 0.1\dot{6}, \quad \frac{1}{7} = 0.\dot{1}4285\dot{7} = 0.142857142857\cdots.$

— Note that some numbers have two *different*, but *equal* representations!
e.g. $0.25 = 0.25\dot{0} = 0.24\dot{9} = 0.249999\cdots.$

Irrational Most real numbers are irrational and don't have a repeating decimal.
e.g. $\sqrt{2} = 1.41421356\cdots$, $e = 2.718281828\cdots$, $\pi = 3.1415926535\cdots.$

— But each of the numbers above *can* be represented by a *finite* formula
e.g. $e = \sum_{n=0}^{\infty} \frac{1}{n!} = \lim_{n\to\infty}(1 + \frac{1}{n})^n$.

— And we can write a program *compute* arbitrarily many digits of the decimal expansion!

Uncomputable For "almost all" real numbers, there is *no* finite description of the decimal expansion!

— Requires "Computing with Infinite Data". [Now, *that's* BIG Data!!]

Decimal approximations to real numbers

Approximation Usually we only require a reasonably good *approximation* to a real number!

Decimal approximations to real numbers

Approximation Usually we only require a reasonably good *approximation* to a real number!

Digits far after the point have a small impact on the value.

Decimal approximations to real numbers

Approximation Usually we only require a reasonably good *approximation* to a real number!

Digits far after the point have a small impact on the value.

Decimal places Approximate real numbers to a finite number of *decimal places*.

Decimal approximations to real numbers

Approximation Usually we only require a reasonably good *approximation* to a real number!

Digits far after the point have a small impact on the value.

Decimal places Approximate real numbers to a finite number of *decimal places*.

— Round to the *nearest* representable number.

Decimal approximations to real numbers

Approximation Usually we only require a reasonably good *approximation* to a real number!

Digits far after the point have a small impact on the value.

Decimal places Approximate real numbers to a finite number of *decimal places*.

— Round to the *nearest* representable number.

e.g. $\pi = 3.14159\,(5\,\text{dp}) = 3.1416\,(4\,\text{dp}) = 3.142\,(3\,\text{dp}) = 3.14\,(2\,\text{dp})$.

Decimal approximations to real numbers

Approximation Usually we only require a reasonably good *approximation* to a real number!

Digits far after the point have a small impact on the value.

Decimal places Approximate real numbers to a finite number of *decimal places*.

— Round to the *nearest* representable number.
e.g. $\pi = 3.14159\,(5\,\text{dp}) = 3.1416\,(4\,\text{dp}) = 3.142\,(3\,\text{dp}) = 3.14\,(2\,\text{dp})$.

— Traditionally, round ties (i.e. halves) *away* from zero.

Decimal approximations to real numbers

Approximation Usually we only require a reasonably good *approximation* to a real number!

Digits far after the point have a small impact on the value.

Decimal places Approximate real numbers to a finite number of *decimal places*.

— Round to the *nearest* representable number.
e.g. $\pi = 3.14159\,(5\,\text{dp}) = 3.1416\,(4\,\text{dp}) = 3.142\,(3\,\text{dp}) = 3.14\,(2\,\text{dp})$.

— Traditionally, round ties (i.e. halves) *away* from zero.
e.g. $5.45 = 5.5\,(1\,\text{dp})$; $-5.45 = -5.5\,(1\,\text{dp})$.

Decimal approximations to real numbers

Approximation Usually we only require a reasonably good *approximation* to a real number!

Digits far after the point have a small impact on the value.

Decimal places Approximate real numbers to a finite number of *decimal places*.

— Round to the *nearest* representable number.
  e.g. $\pi = 3.14159\,(5\,\text{dp}) = 3.1416\,(4\,\text{dp}) = 3.142\,(3\,\text{dp}) = 3.14\,(2\,\text{dp})$.

— Traditionally, round ties (i.e. halves) *away* from zero.
  e.g. $5.45 = 5.5\,(1\,\text{dp})$; $-5.45 = -5.5\,(1\,\text{dp})$.

— Don't round an already-rounded number!

Decimal approximations to real numbers

Approximation Usually we only require a reasonably good *approximation* to a real number!

Digits far after the point have a small impact on the value.

Decimal places Approximate real numbers to a finite number of *decimal places*.

— Round to the *nearest* representable number.
e.g. $\pi = 3.14159\,(5\,\text{dp}) = 3.1416\,(4\,\text{dp}) = 3.142\,(3\,\text{dp}) = 3.14\,(2\,\text{dp})$.

— Traditionally, round ties (i.e. halves) *away* from zero.
e.g. $5.45 = 5.5\,(1\,\text{dp})$; $-5.45 = -5.5\,(1\,\text{dp})$.

— Don't round an already-rounded number!
e.g. $5.45 = 5.5\,(1\,\text{dp}) = 5\,(0\,\text{dp})$ even though $5.5 = 6\,(0\,\text{dp})$!

Decimal approximations to real numbers

Approximation Usually we only require a reasonably good *approximation* to a real number!

Digits far after the point have a small impact on the value.

Decimal places Approximate real numbers to a finite number of *decimal places*.

— Round to the *nearest* representable number.
e.g. $\pi = 3.14159\,(5\,\text{dp}) = 3.1416\,(4\,\text{dp}) = 3.142\,(3\,\text{dp}) = 3.14\,(2\,\text{dp})$.

— Traditionally, round ties (i.e. halves) *away* from zero.
e.g. $5.45 = 5.5\,(1\,\text{dp})$; $-5.45 = -5.5\,(1\,\text{dp})$.

— Don't round an already-rounded number!
e.g. $5.45 = 5.5\,(1\,\text{dp}) = 5\,(0\,\text{dp})$ even though $5.5 = 6\,(0\,\text{dp})$! [Sorry]

Significant figures

Significant figures The number of *significant figures* of an approximation is the number of digits *excluding* leading zeros.

Significant figures

Significant figures The number of *significant figures* of an approximation is the number of digits *excluding* leading zeros.

$\pi = 3.14159$ (to 5 decimal places, 6 significant figures)

Significant figures

Significant figures The number of *significant figures* of an approximation is the number of digits *excluding* leading zeros.

$\pi = 3.14159$ (to 5 decimal places, 6 significant figures)

$\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf})$

Significant figures

Significant figures The number of *significant figures* of an approximation is the number of digits *excluding* leading zeros.

$\pi = 3.14159$ (to 5 decimal places, 6 significant figures)

$\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf})$

Zero Note that 0 has *no* significant figures!

Significant figures

Significant figures The number of *significant figures* of an approximation is the number of digits *excluding* leading zeros.

$\pi = 3.14159$ (to 5 decimal places, 6 significant figures)

$\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf})$

Zero Note that 0 has *no* significant figures!

Units The number of significant figures is *independent* of the unit used.

Significant figures

Significant figures The number of *significant figures* of an approximation is the number of digits *excluding* leading zeros.

$\pi = 3.14159$ (to 5 decimal places, 6 significant figures)

$\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf})$

Zero Note that 0 has *no* significant figures!

Units The number of significant figures is *independent* of the unit used.
e.g. The density of gold is $19.32\,\mathrm{g\,cm^{-3}}\,(4\,\text{sf}, 2\,\text{dp}) = 19320\,\mathrm{kg\,m^{-3}}\,(4\,\text{sf})$.

Significant figures

Significant figures The number of *significant figures* of an approximation is the number of digits *excluding* leading zeros.

$\pi = 3.14159$ (to 5 decimal places, 6 significant figures)

$\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf})$

Zero Note that 0 has *no* significant figures!

Units The number of significant figures is *independent* of the unit used.
e.g. The density of gold is $19.32\,\mathrm{g\,cm^{-3}}\,(4\,\text{sf}, 2\,\text{dp}) = 19320\,\mathrm{kg\,m^{-3}}\,(4\,\text{sf})$.

Whereas the number of decimal places depends on the units used.

Significant figures

Significant figures The number of *significant figures* of an approximation is the number of digits *excluding* leading zeros.

$\pi = 3.14159$ (to 5 decimal places, 6 significant figures)

$\alpha = 0.007\,297\,352\,57\ (11\,\text{dp}, 9\,\text{sf})$

Zero Note that 0 has *no* significant figures!

Units The number of significant figures is *independent* of the unit used.
e.g. The density of gold is $19.32\,\mathrm{g\,cm^{-3}}\ (4\,\text{sf}, 2\,\text{dp}) = 19320\,\mathrm{kg\,m^{-3}}\ (4\,\text{sf})$.

Whereas the number of decimal places depends on the units used.
e.g. The density of gold is $\rho_{\text{Au}} = 19.32\,\mathrm{g\,cm^{-3}}\ (2\,\text{dp}) = 1932?\,\mathrm{kg\,m^{-3}}$.

Scientific notation

Scientific Notation Write a number as a value $\pm m \times 10^e$ where $1 \leq m < 10$ and e is an integer.

Scientific notation

Scientific Notation Write a number as a value $\pm m \times 10^e$ where $1 \leq m < 10$ and e is an integer.

e.g. $\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf}) = 7.297\,352\,57 \times 10^{-3}\,(9\,\text{sf})$.

Scientific notation

Scientific Notation Write a number as a value $\pm m \times 10^e$ where $1 \leq m < 10$ and e is an integer.

e.g. $\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf}) = 7.297\,352\,57 \times 10^{-3}\,(9\,\text{sf})$.

Mantissa and exponent m is the *mantissa* and e the *exponent*.

Scientific notation

Scientific Notation Write a number as a value $\pm m \times 10^e$ where $1 \leq m < 10$ and e is an integer.

e.g. $\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf}) = 7.297\,352\,57 \times 10^{-3}\,(9\,\text{sf})$.

Mantissa and exponent m is the *mantissa* and e the *exponent*.

Significant figures The length of the mantissa is the number of significant figures.

Scientific notation

Scientific Notation Write a number as a value $\pm m \times 10^e$ where $1 \leq m < 10$ and e is an integer.

e.g. $\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf}) = 7.297\,352\,57 \times 10^{-3}\,(9\,\text{sf})$.

Mantissa and exponent m is the *mantissa* and e the *exponent*.

Significant figures The length of the mantissa is the number of significant figures.

e.g. $10200 = 1.020 \times 10^4\ (4\,\text{sf})$.

Scientific notation

Scientific Notation Write a number as a value $\pm m \times 10^e$ where $1 \leq m < 10$ and e is an integer.

e.g. $\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf}) = 7.297\,352\,57 \times 10^{-3}\,(9\,\text{sf})$.

Mantissa and exponent m is the *mantissa* and e the *exponent*.

Significant figures The length of the mantissa is the number of significant figures.

e.g. $10200 = 1.020 \times 10^4\,(4\,\text{sf})$.

Physical constants Scientific notation is especially useful in physics, where quantities are often very, very large or small:

Scientific notation

Scientific Notation Write a number as a value $\pm m \times 10^e$ where $1 \leq m < 10$ and e is an integer.

e.g. $\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf}) = 7.297\,352\,57\times 10^{-3}\,(9\,\text{sf})$.

Mantissa and exponent m is the *mantissa* and e the *exponent*.

Significant figures The length of the mantissa is the number of significant figures.

e.g. $10200 = 1.020 \times 10^4\,(4\,\text{sf})$.

Physical constants Scientific notation is especially useful in physics, where quantities are often very, very large or small:

e.g. $h = 6.62607015\times 10^{-34}\,\mathrm{J\,s}$

Scientific notation

Scientific Notation Write a number as a value $\pm m \times 10^e$ where $1 \leq m < 10$ and e is an integer.

e.g. $\alpha = 0.007\,297\,352\,57\,(11\,\text{dp}, 9\,\text{sf}) = 7.297\,352\,57 \times 10^{-3}\,(9\,\text{sf})$.

Mantissa and exponent m is the *mantissa* and e the *exponent*.

Significant figures The length of the mantissa is the number of significant figures.

e.g. $10200 = 1.020 \times 10^4\,(4\,\text{sf})$.

Physical constants Scientific notation is especially useful in physics, where quantities are often very, very large or small:

e.g. $h = 6.62607015 \times 10^{-34}\,\mathrm{J\,s}$

$c = 299\,792\,458\,\mathrm{m\,s^{-1}} = 2.99792458 \times 10^8 \mathrm{m\,s^{-1}}$

Scientific notation

Scientific Notation Write a number as a value $\pm m \times 10^e$ where $1 \leq m < 10$ and e is an integer.

e.g. $\alpha = 0.007\,297\,352\,57\,(11\,\textsf{dp}, 9\,\textsf{sf}) = 7.297\,352\,57 \times 10^{-3}\,(9\,\textsf{sf})$.

Mantissa and exponent m is the *mantissa* and e the *exponent*.

Significant figures The length of the mantissa is the number of significant figures.

e.g. $10200 = 1.020 \times 10^4\,(4\,\textsf{sf})$.

Physical constants Scientific notation is especially useful in physics, where quantities are often very, very large or small:

e.g. $h = 6.62607015 \times 10^{-34}\,\mathrm{J\,s}$

$$c = 299\,792\,458\,\mathrm{m\,s^{-1}} = 2.99792458 \times 10^8 \mathrm{m\,s^{-1}}$$
$$= 3.00 \times 10^8 \mathrm{m\,s^{-1}}\,(3\,\textsf{sf})$$

Digital representations

Memory How can we represent numbers on a digital computer?

Digital representations

Memory How can we represent numbers on a digital computer?

Digital computer memory consists of a huge number of electromagnetic switches, capable of storing values $\uparrow$ or $\downarrow$.

Digital representations

Memory How can we represent numbers on a digital computer?
Digital computer memory consists of a huge number of electromagnetic switches, capable of storing values $\uparrow$ or $\downarrow$.

Hardware Current digital computer hardware works most efficiently with *fixed-width* data types.

Digital representations

Memory How can we represent numbers on a digital computer?
Digital computer memory consists of a huge number of electromagnetic switches, capable of storing values $\uparrow$ or $\downarrow$.

Hardware Current digital computer hardware works most efficiently with *fixed-width* data types.

Only *finitely many* values can be represented in a fixed-width type.

Digital representations

Memory How can we represent numbers on a digital computer?
Digital computer memory consists of a huge number of electromagnetic switches, capable of storing values $\uparrow$ or $\downarrow$.

Hardware Current digital computer hardware works most efficiently with *fixed-width* data types.

Only *finitely many* values can be represented in a fixed-width type.

Software Infinite data types must be implemented in software.

Digital representations

Memory How can we represent numbers on a digital computer?
Digital computer memory consists of a huge number of electromagnetic switches, capable of storing values $\uparrow$ or $\downarrow$.

Hardware Current digital computer hardware works most efficiently with *fixed-width* data types.

Only *finitely many* values can be represented in a fixed-width type.

Software Infinite data types must be implemented in software.

Countable types like the integers can be represented by *lists* of fixed-size *words*.

Digital representations

Memory How can we represent numbers on a digital computer?
Digital computer memory consists of a huge number of electromagnetic switches, capable of storing values $\uparrow$ or $\downarrow$.

Hardware Current digital computer hardware works most efficiently with *fixed-width* data types.

Only *finitely many* values can be represented in a fixed-width type.

Software Infinite data types must be implemented in software.

Countable types like the integers can be represented by *lists* of fixed-size *words*.

Uncountable types like the reals can be represented by infinite *streams* of data.
— At any time, we only have a finite *approximation* to the result.

Binary integers

Binary Each memory location can store a single *binary digit* (bit).

Binary integers

Binary Each memory location can store a single *binary digit* (bit).

Represent 0 by $\downarrow$ and 1 by $\uparrow$.

Binary integers

Binary Each memory location can store a single *binary digit* (bit).

Represent 0 by $\downarrow$ and 1 by $\uparrow$.

Fixed-width Use a fixed number of digits for elementary data types.

Binary integers

Binary Each memory location can store a single *binary digit* (bit).

Represent 0 by $\downarrow$ and 1 by $\uparrow$.

Fixed-width Use a fixed number of digits for elementary data types.
e.g. Java's `int` uses 32-bits to hold a value between -2^{31} and $2^{31}-1$.

Binary integers

Binary Each memory location can store a single *binary digit* (bit).

Represent 0 by $\downarrow$ and 1 by $\uparrow$.

Fixed-width Use a fixed number of digits for elementary data types.
e.g. Java's `int` uses 32-bits to hold a value between -2^{31} and $2^{31}-1$.

Example The number 42 in 8-bit binary is 00101010_2, or $\downarrow\downarrow\uparrow\downarrow\uparrow\downarrow\uparrow\downarrow$.

Binary integers

Binary Each memory location can store a single *binary digit* (bit).

Represent 0 by $\downarrow$ and 1 by $\uparrow$.

Fixed-width Use a fixed number of digits for elementary data types.
e.g. Java's `int` uses 32-bits to hold a value between -2^{31} and $2^{31}-1$.

Example The number 42 in 8-bit binary is 00101010_2, or $\downarrow\downarrow\uparrow\downarrow\uparrow\downarrow\uparrow\downarrow$.
Using 16-bits, have 0000000000101010_2, or $\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\uparrow\downarrow\uparrow\downarrow\uparrow\downarrow$.

Binary integers

Binary Each memory location can store a single *binary digit* (bit).

Represent 0 by $\downarrow$ and 1 by $\uparrow$.

Fixed-width Use a fixed number of digits for elementary data types.
e.g. Java's `int` uses 32-bits to hold a value between -2^{31} and $2^{31}-1$.

Example The number 42 in 8-bit binary is 00101010_2, or $\downarrow\downarrow\uparrow\downarrow\uparrow\downarrow\uparrow\downarrow$.
Using 16-bits, have 0000000000101010_2, or $\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\uparrow\downarrow\uparrow\downarrow\uparrow\downarrow$.

Variable-width Arbitrarily-sized integers can be implemented in software using a *list* of (e.g. 32-bit) *words*.

Binary integers

Binary Each memory location can store a single *binary digit* (bit).

Represent 0 by $\downarrow$ and 1 by $\uparrow$.

Fixed-width Use a fixed number of digits for elementary data types.
e.g. Java's `int` uses 32-bits to hold a value between -2^{31} and $2^{31}-1$.

Example The number 42 in 8-bit binary is 00101010_2, or $\downarrow\downarrow\uparrow\downarrow\uparrow\downarrow\uparrow\downarrow$.
Using 16-bits, have 0000000000101010_2, or $\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\uparrow\downarrow\uparrow\downarrow\uparrow\downarrow$.

Variable-width Arbitrarily-sized integers can be implemented in software using a *list* of (e.g. 32-bit) *words*.
e.g. `mpz_t` from the GNU Multiple-Precision Library (GMP).

Fixed- and floating-point numbers

Fixed-point A *fixed-point* representation of the real numbers uses a fixed number of fractional (binary) digits.

Fixed- and floating-point numbers

Fixed-point A *fixed-point* representation of the real numbers uses a fixed number of fractional (binary) digits.

Used in signal and image processing, less often for scientific computing.

Fixed- and floating-point numbers

Fixed-point A *fixed-point* representation of the real numbers uses a fixed number of fractional (binary) digits.

Used in signal and image processing, less often for scientific computing.

Floating-point Use a fixed number of *significant* digits in the *mantissa*, and determine the size by the *exponent*.

Fixed- and floating-point numbers

Fixed-point A *fixed-point* representation of the real numbers uses a fixed number of fractional (binary) digits.

Used in signal and image processing, less often for scientific computing.

Floating-point Use a fixed number of *significant* digits in the *mantissa*, and determine the size by the *exponent*.

Single-precision An IEEE standard, 32-bit floating-point format.

Fixed- and floating-point numbers

Fixed-point A *fixed-point* representation of the real numbers uses a fixed number of fractional (binary) digits.

Used in signal and image processing, less often for scientific computing.

Floating-point Use a fixed number of *significant* digits in the *mantissa*, and determine the size by the *exponent*.

Single-precision An IEEE standard, 32-bit floating-point format.

Double-precision Currently, the most commonly used approach for representing real numbers is the 64-bit *IEEE 754 double-precision binary floating-point format*:

Fixed- and floating-point numbers

Fixed-point A *fixed-point* representation of the real numbers uses a fixed number of fractional (binary) digits.

Used in signal and image processing, less often for scientific computing.

Floating-point Use a fixed number of *significant* digits in the *mantissa*, and determine the size by the *exponent*.

Single-precision An IEEE standard, 32-bit floating-point format.

Double-precision Currently, the most commonly used approach for representing real numbers is the 64-bit *IEEE 754 double-precision binary floating-point format*:

$$\overbrace{\pm}^{\text{1-bit sign}} \underbrace{1{\cdot}XX\cdots X}_{\text{1+52-bit mantissa}} \times 2^{\overbrace{\pm XX\cdots X}^{\text{11-bit exponent}}}$$

Fixed- and floating-point numbers

Fixed-point A *fixed-point* representation of the real numbers uses a fixed number of fractional (binary) digits.

Used in signal and image processing, less often for scientific computing.

Floating-point Use a fixed number of *significant* digits in the *mantissa*, and determine the size by the *exponent*.

Single-precision An IEEE standard, 32-bit floating-point format.

Double-precision Currently, the most commonly used approach for representing real numbers is the 64-bit *IEEE 754 double-precision binary floating-point format*:

$$\overbrace{\pm}^{\text{1-bit sign}} \underbrace{1{\cdot}XX\cdots X}_{\text{1+52-bit mantissa}} \times 2^{\overbrace{\pm XX\cdots X}^{\text{11-bit exponent}}}$$

Example $-6.75 = -1.6875 \times 2^2 = -1{\cdot}1011000\cdots0_2 \times 2^{+0000000010_2}$

Fixed- and floating-point numbers

Fixed-point A *fixed-point* representation of the real numbers uses a fixed number of fractional (binary) digits.

Used in signal and image processing, less often for scientific computing.

Floating-point Use a fixed number of *significant* digits in the *mantissa*, and determine the size by the *exponent*.

Single-precision An IEEE standard, 32-bit floating-point format.

Double-precision Currently, the most commonly used approach for representing real numbers is the 64-bit *IEEE 754 double-precision binary floating-point format*:

$$\overbrace{\pm}^{\text{1-bit sign}} \underbrace{1{\cdot}XX\cdots X}_{\text{1+52-bit mantissa}} \times 2^{\overbrace{\pm XX\cdots X}^{\text{11-bit exponent}}}$$

Example $-6.75 = -1.6875 \times 2^2 = -1{\cdot}1011000\cdots0_2 \times 2^{+0000000010_2}$

Dyadic Any binary fixed- or floating- point number is a *dyadic* of the form $p/2^q$ for integers p, q.

Fixed- and floating-point numbers

Fixed-point A *fixed-point* representation of the real numbers uses a fixed number of fractional (binary) digits.

Used in signal and image processing, less often for scientific computing.

Floating-point Use a fixed number of *significant* digits in the *mantissa*, and determine the size by the *exponent*.

Single-precision An IEEE standard, 32-bit floating-point format.

Double-precision Currently, the most commonly used approach for representing real numbers is the 64-bit *IEEE 754 double-precision binary floating-point format*:

$$\overbrace{\pm}^{\text{1-bit sign}} \underbrace{1{\cdot}XX\cdots X}_{\text{1+52-bit mantissa}} \times 2^{\overbrace{\pm XX\cdots X}^{\text{11-bit exponent}}}$$

Example $-6.75 = -1.6875 \times 2^2 = -1{\cdot}1011000\cdots0_2 \times 2^{+0000000010_2}$

Dyadic Any binary fixed- or floating- point number is a *dyadic* of the form $p/2^q$ for integers p, q.

Floating-point arithmetic

Inexact The *result* of an arithmetical operation on floating-point numbers need not be a floating-point number!

Floating-point arithmetic

Inexact The *result* of an arithmetical operation on floating-point numbers need not be a floating-point number!
e.g. $1/3 = 1.0000000_2/11.000000_2 = 0.0101010101\cdots_2$.

Floating-point arithmetic

Inexact The *result* of an arithmetical operation on floating-point numbers need not be a floating-point number!
e.g. $1/3 = 1.0000000_2/11.000000_2 = 0.0101010101\cdots_2$.

Round-to-Nearest Round the result of any arithmetical operation to the *nearest* representable number.

Floating-point arithmetic

Inexact The *result* of an arithmetical operation on floating-point numbers need not be a floating-point number!
e.g. $1/3 = 1.0000000_2/11.000000_2 = 0.0101010101\cdots_2$.

Round-to-Nearest Round the result of any arithmetical operation to the *nearest* representable number.
e.g. Using a $1+7$-bit mantissa,
$1/3 = 1.0000000_2/11.000000_2 = 0.010101010101\cdots_2 \approx 0.010101011_2$.

Floating-point arithmetic

Inexact The *result* of an arithmetical operation on floating-point numbers need not be a floating-point number!
e.g. $1/3 = 1.0000000_2/11.000000_2 = 0.0101010101\cdots_2$.

Round-to-Nearest Round the result of any arithmetical operation to the *nearest* representable number.
e.g. Using a $1+7$-bit mantissa,
$1/3 = 1.0000000_2/11.000000_2 = 0.0101010101\textcolor{gray}{01\cdots}_2 \approx 0.010101011_2$.

Break-ties-to-even.
e.g. $1.0000001_2 + 0.10000001_2 = 1.10000011_2 \approx 1.1000010_2$.

Floating-point arithmetic

Inexact The *result* of an arithmetical operation on floating-point numbers need not be a floating-point number!
e.g. $1/3 = 1.0000000_2/11.000000_2 = 0.0101010101\cdots_2$.

Round-to-Nearest Round the result of any arithmetical operation to the *nearest* representable number.
e.g. Using a $1{+}7$-bit mantissa,
$1/3 = 1.0000000_2/11.000000_2 = 0.010101010101\cdots_2 \approx 0.010101011_2$.

Break-ties-to-even.
e.g. $1.0000001_2 + 0.10000001_2 = 1.10000011_2 \approx 1.1000010_2$.

Directed Round upward or downward.
$\frac{85}{256} = 0.01010101_2 < \frac{1}{3} = 1.000000_2/11.00000_2 < \frac{86}{256} = 0.01010110_2$

Floating-point arithmetic

Inexact The *result* of an arithmetical operation on floating-point numbers need not be a floating-point number!
e.g. $1/3 = 1.0000000_2/11.000000_2 = 0.0101010101\cdots_2$.

Round-to-Nearest Round the result of any arithmetical operation to the *nearest* representable number.
e.g. Using a $1{+}7$-bit mantissa,
$1/3 = 1.0000000_2/11.000000_2 = 0.010101010101\cdots_2 \approx 0.010101011_2$.

Break-ties-to-even.
e.g. $1.0000001_2 + 0.10000001_2 = 1.10000011_2 \approx 1.1000010_2$.

Directed Round upward or downward.
$\frac{85}{256} = 0.01010101_2 < \frac{1}{3} = 1.000000_2/11.00000_2 < \frac{86}{256} = 0.01010110_2$

Example $(\frac{1}{7} + \frac{4}{7}) + \frac{2}{7} \approx (0.0010010010_2 + 0.10010010_2) + 0.010010010_2$
$= 0.1011011\cancel{010}_2 + 0.010010010_2 = 0.11111111\cancel{0}_2 = 0.11111111_2$

Machine epsilon

Machine epsilon The difference between 1 and the next higher representable number.

Machine epsilon

Machine epsilon The difference between 1 and the next higher representable number.

For -precision floating-point,

$$\epsilon = 1^{+} - 1 = 2^{-23} \approx 1.1921 \times 10^{-7}.$$

Machine epsilon

Machine epsilon The difference between 1 and the next higher representable number.

For -precision floating-point,

$$\epsilon = 1^{+} - 1 = 2^{-23} \approx 1.1921 \times 10^{-7}.$$

For double-precision floating-point,

$$\epsilon = 1^{+} - 1 = 2^{-52} \approx 2.2204 \times 10^{-16}.$$

Machine epsilon

Machine epsilon The difference between 1 and the next higher representable number.

For -precision floating-point,

$$\epsilon = 1^+ - 1 = 2^{-23} \approx 1.1921\times 10^{-7}.$$

For double-precision floating-point,

$$\epsilon = 1^+ - 1 = 2^{-52} \approx 2.2204\times 10^{-16}.$$

Spacing Over the interval $[1, 2]$, numbers have a spacing of ϵ.

Machine epsilon

Machine epsilon The difference between 1 and the next higher representable number.

For -precision floating-point,

$$\epsilon = 1^{+} - 1 = 2^{-23} \approx 1.1921 \times 10^{-7}.$$

For double-precision floating-point,

$$\epsilon = 1^{+} - 1 = 2^{-52} \approx 2.2204 \times 10^{-16}.$$

Spacing Over the interval $[1, 2]$, numbers have a spacing of ϵ.
Over $[\frac{1}{2}, 1]$, the spacing is $\epsilon/2$; on $[2, 4]$ it is 2ϵ.

Machine epsilon

Machine epsilon The difference between 1 and the next higher representable number.

For -precision floating-point,

$$\epsilon = 1^+ - 1 = 2^{-23} \approx 1.1921 \times 10^{-7}.$$

For double-precision floating-point,

$$\epsilon = 1^+ - 1 = 2^{-52} \approx 2.2204 \times 10^{-16}.$$

Spacing Over the interval $[1, 2]$, numbers have a spacing of ϵ.
Over $[\frac{1}{2}, 1]$, the spacing is $\epsilon/2$; on $[2, 4]$ it is 2ϵ.

Small numbers are more closely spaced allowing greater precision; large numbers more widely spaced.

Machine epsilon

Machine epsilon The difference between 1 and the next higher representable number.

For -precision floating-point,
$$\epsilon = 1^+ - 1 = 2^{-23} \approx 1.1921 \times 10^{-7}.$$

For double-precision floating-point,
$$\epsilon = 1^+ - 1 = 2^{-52} \approx 2.2204 \times 10^{-16}.$$

Spacing Over the interval $[1, 2]$, numbers have a spacing of ϵ.
Over $[\frac{1}{2}, 1]$, the spacing is $\epsilon/2$; on $[2, 4]$ it is 2ϵ.

Small numbers are more closely spaced allowing greater precision; large numbers more widely spaced.

Minimum/maximum representable number For double-precision floating-point, the *minimum* strictly-positive representable number is
$$0^+ = 2^{-1074} \approx 4.94 \times 10^{-324}.$$

Machine epsilon

Machine epsilon The difference between 1 and the next higher representable number.

For -precision floating-point,
$\epsilon = 1^{+} - 1 = 2^{-23} \approx 1.1921 \times 10^{-7}$.

For double-precision floating-point,
$\epsilon = 1^{+} - 1 = 2^{-52} \approx 2.2204 \times 10^{-16}$.

Spacing Over the interval $[1, 2]$, numbers have a spacing of ϵ.
Over $[\frac{1}{2}, 1]$, the spacing is $\epsilon/2$; on $[2, 4]$ it is 2ϵ.

Small numbers are more closely spaced allowing greater precision; large numbers more widely spaced.

Minimum/maximum representable number For double-precision floating-point, the *minimum* strictly-positive representable number is
$0^{+} = 2^{-1074} \approx 4.94 \times 10^{-324}$.

The *maximum* representable number is
$\infty^{-} = 2^{1023}(2 - \epsilon) = 2^{1024}(1 - \epsilon/2) \approx 1.798 \times 10^{308}$.

Floating-point numbers in Matlab

Double precision By default, Matlab uses double-precision floating-point numbers.

Floating-point numbers in Matlab

Double precision By default, Matlab uses double-precision floating-point numbers.

Single precision Use `single(x)` to convert x to *single-precision*. Use `double(xs)` to convert to double-precision.

Floating-point numbers in Matlab

Double precision By default, Matlab uses double-precision floating-point numbers.

Single precision Use `single(x)` to convert x to *single-precision*. Use `double(xs)` to convert to double-precision.

Display format By default, Matlab only displays 4-5 significant figures.

Floating-point numbers in Matlab

Double precision By default, Matlab uses double-precision floating-point numbers.

Single precision Use `single(x)` to convert x to *single-precision*. Use `double(xs)` to convert to double-precision.

Display format By default, Matlab only displays 4-5 significant figures. To display 15 significant figures, use

```
>> format long
```

Floating-point numbers in Matlab

Double precision By default, Matlab uses double-precision floating-point numbers.

Single precision Use `single(x)` to convert x to *single-precision*. Use `double(xs)` to convert to double-precision.

Display format By default, Matlab only displays 4-5 significant figures.
To display 15 significant figures, use

```
>> format long
```

To go back to 4-5 significant figues, use

```
>> format short
```

Floating-point numbers in Matlab

Double precision By default, Matlab uses double-precision floating-point numbers.

Single precision Use `single(x)` to convert x to *single-precision*. Use `double(xs)` to convert to double-precision.

Display format By default, Matlab only displays 4-5 significant figures.
To display 15 significant figures, use

```
>> format long
```

To go back to 4-5 significant figues, use

```
>> format short
```

The use of `format long` is vital for displaying intermediates and results of highly accurate calculations!!

Philosophical question

Philosophical question Do Klingons use floating-point?

Errors in Scientific Computing

Sources of error

There are three main sources of error in scientific computing:

Sources of error

There are three main sources of error in scientific computing:

Roundoff errors Errors due to the use of inexact (floating-point) arithmetic for computations.

Sources of error

There are three main sources of error in scientific computing:

Roundoff errors Errors due to the use of inexact (floating-point) arithmetic for computations.

- Usually extremely small for simple double-precision calculations, but may become significant for long calculations or due to *ill-conditioning* of a problem or method.

Sources of error

There are three main sources of error in scientific computing:

Roundoff errors Errors due to the use of inexact (floating-point) arithmetic for computations.

- Usually extremely small for simple double-precision calculations, but may become significant for long calculations or due to *ill-conditioning* of a problem or method.

Truncation errors Errors due to the use of an inexact method.

Sources of error

There are three main sources of error in scientific computing:

Roundoff errors Errors due to the use of inexact (floating-point) arithmetic for computations.

- Usually extremely small for simple double-precision calculations, but may become significant for long calculations or due to *ill-conditioning* of a problem or method.

Truncation errors Errors due to the use of an inexact method.

- For example, the approximation $f'(x) \approx (f(x+h) - f(x-h))/2h$ has a trunctation error $O(h^2)$.

Sources of error

There are three main sources of error in scientific computing:

Roundoff errors Errors due to the use of inexact (floating-point) arithmetic for computations.

- Usually extremely small for simple double-precision calculations, but may become significant for long calculations or due to *ill-conditioning* of a problem or method.

Truncation errors Errors due to the use of an inexact method.

- For example, the approximation $f'(x) \approx (f(x+h) - f(x-h))/2h$ has a trunctation error $O(h^2)$.

Errors in data Data often contains measurement errors.

Sources of error

There are three main sources of error in scientific computing:

Roundoff errors Errors due to the use of inexact (floating-point) arithmetic for computations.

- Usually extremely small for simple double-precision calculations, but may become significant for long calculations or due to *ill-conditioning* of a problem or method.

Truncation errors Errors due to the use of an inexact method.

- For example, the approximation $f'(x) \approx (f(x+h) - f(x-h))/2h$ has a trunctation error $O(h^2)$.

Errors in data Data often contains measurement errors.

- Although we as knowledge engineers cannot do anything about these errors, we can try and estimate their impact on the final result, and maybe even choose a method which reduces this.

Absolute and relative errors

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

$$p = \pi = 3.1415927\cdots \qquad p^* = \frac{22}{7} = 3.1428571\cdots$$

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

$$p = \pi = 3.1415927\cdots \qquad p^* = \frac{22}{7} = 3.1428571\cdots$$

Absolute error:

$|\frac{22}{7} - \pi|$

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

$$p = \pi = 3.1415927\cdots \qquad p^* = \frac{22}{7} = 3.1428571\cdots$$

Absolute error:

$|\frac{22}{7} - \pi| = |3.1428571\cdots - 3.1415927\cdots|$

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

$$p = \pi = 3.1415927\cdots \qquad p^* = \tfrac{22}{7} = 3.1428571\cdots$$

Absolute error:

$|\tfrac{22}{7} - \pi| = |3.1428571\cdots - 3.1415927\cdots| = 0.0012645\cdots$

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

$$p = \pi = 3.1415927\cdots \qquad p^* = \frac{22}{7} = 3.1428571\cdots$$

Absolute error:

$|\frac{22}{7} - \pi| = |3.1428571\cdots - 3.1415927\cdots| = 0.0012645\cdots = 1.3\times 10^{-3}\ (2\,\text{sf})$

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

$$p = \pi = 3.1415927\cdots \qquad p^* = \tfrac{22}{7} = 3.1428571\cdots$$

Absolute error:

$|\frac{22}{7} - \pi| = |3.1428571\cdots - 3.1415927\cdots| = 0.0012645\cdots = 1.3\times10^{-3}\,(2\,\text{sf})$

Relative error:

$|\frac{22}{7} - \pi|/|\pi|$

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

$$p = \pi = 3.1415927\cdots \qquad p^* = \tfrac{22}{7} = 3.1428571\cdots$$

Absolute error:

$|\frac{22}{7} - \pi| = |3.1428571\cdots - 3.1415927\cdots| = 0.0012645\cdots = 1.3\times10^{-3}\ (2\,\text{sf})$

Relative error:

$|\frac{22}{7} - \pi|/|\pi| = 0.0012645\cdots/3.1415927\cdots$

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

$$p = \pi = 3.1415927\cdots \qquad p^* = \tfrac{22}{7} = 3.1428571\cdots$$

Absolute error:

$|\frac{22}{7} - \pi| = |3.1428571\cdots - 3.1415927\cdots| = 0.0012645\cdots = 1.3\times 10^{-3}\ (2\,\text{sf})$

Relative error:

$|\frac{22}{7} - \pi|/|\pi| = 0.0012645\cdots/3.1415927\cdots = 0.00040250\cdots$

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

$$p = \pi = 3.1415927\cdots \qquad p^* = \tfrac{22}{7} = 3.1428571\cdots$$

Absolute error:

$$|\tfrac{22}{7} - \pi| = |3.1428571\cdots - 3.1415927\cdots| = 0.0012645\cdots = 1.3\times 10^{-3}\ (2\,\text{sf})$$

Relative error:

$$|\tfrac{22}{7} - \pi|/|\pi| = 0.0012645\cdots/3.1415927\cdots = 0.00040250\cdots = 4.0\times 10^{-4}\ (2\,\text{sf})$$

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

$$p = \pi = 3.1415927\cdots \qquad p^* = \frac{22}{7} = 3.1428571\cdots$$

Absolute error:

$|\frac{22}{7} - \pi| = |3.1428571\cdots - 3.1415927\cdots| = 0.0012645\cdots = 1.3\times 10^{-3}\ (2\,\mathsf{sf})$

Relative error:

$|\frac{22}{7} - \pi|/|\pi| = 0.0012645\cdots/3.1415927\cdots = 0.00040250\cdots = 4.0\times 10^{-4}\ (2\,\mathsf{sf})$

$|\frac{22}{7}/\pi - 1|$

Absolute and relative errors

Absolute error The *absolute error* in approximating p by p^* is $|p^* - p|$; equivalently $|p - p^*|$.

Relative error The *relative error* in approximating p by p^* is $|p^* - p|/|p|$.

An alternative form of the relative error is $|p^*/p - 1|$.

The relative error is *dimensionless*; it is the same in any units.

Example Compute the absolute and relative errors of the approximation $\pi \approx \frac{22}{7}$.

$$p = \pi = 3.1415927\cdots \qquad p^* = \tfrac{22}{7} = 3.1428571\cdots$$

Absolute error:

$$|\tfrac{22}{7} - \pi| = |3.1428571\cdots - 3.1415927\cdots| = 0.0012645\cdots = 1.3\times 10^{-3}\ (2\,\text{sf})$$

Relative error:

$$|\tfrac{22}{7} - \pi|/|\pi| = 0.0012645\cdots/3.1415927\cdots = 0.00040250\cdots = 4.0\times 10^{-4}\ (2\,\text{sf})$$

$$\begin{aligned} |\tfrac{22}{7}/\pi - 1| &= |3.1428571\cdots/3.1415927\cdots - 1| = |1.0004024\cdots - 1| \\ &= 4.0\times 10^{-4}\ (2\,\text{sf}) \end{aligned}$$

Error computation

Error computation Use an *unrounded* version of the exact value, or a version rounded to *much higher* precision than your approximation.

Error computation

Error computation Use an *unrounded* version of the exact value, or a version rounded to *much higher* precision than your approximation.

e.g. $\pi \approx \pi^* = 3.14$ with relative error

$|3.14 - \pi|/\pi = |3.14/\pi - 1| = 0.00050697\cdots = 5.1\times 10^{-4} = 0.051\%\ (2\,\mathsf{sf})$

Error computation

Error computation Use an *unrounded* version of the exact value, or a version rounded to *much higher* precision than your approximation.

e.g. $\pi \approx \pi^* = 3.14$ with relative error

$$|3.14 - \pi|/\pi = |3.14/\pi - 1| = 0.00050697\cdots = 5.1\times 10^{-4} = 0.051\% \, (2\,\text{sf})$$
$$\approx |3.14/3.1416 - 1| = 0.00050929\cdots = 0.051\% \, (2\,\text{sf})$$

Error computation

Error computation Use an *unrounded* version of the exact value, or a version rounded to *much higher* precision than your approximation.

e.g. $\pi \approx \pi^* = 3.14$ with relative error

$$
\begin{aligned}
|3.14 - \pi|/\pi = |3.14/\pi - 1| &= 0.00050697\cdots = 5.1\times 10^{-4} = 0.051\%\,(2\,\text{sf})\\
&\approx |3.14/3.1416 - 1| = 0.00050929\cdots = 0.051\%\,(2\,\text{sf})\\
&\sim |3.14/3.142 - 1| = 0.00063654\cdots = 0.064\%\,(2\,\text{sf})
\end{aligned}
$$

Error computation

Error computation Use an *unrounded* version of the exact value, or a version rounded to *much higher* precision than your approximation.

e.g. $\pi \approx \pi^* = 3.14$ with relative error

$$
\begin{aligned}
|3.14-\pi|/\pi = |3.14/\pi - 1| &= 0.00050697\cdots = 5.1\times 10^{-4} = 0.051\%\,(2\,\text{sf}) \\
&\approx |3.14/3.1416 - 1| = 0.00050929\cdots = 0.051\%\,(2\,\text{sf}) \\
&\sim |3.14/3.142 - 1| = 0.00063654\cdots = 0.064\%\,(2\,\text{sf}) \\
&\not\approx |3.14/3.14 - 1| = 0.
\end{aligned}
$$

Error estimates and bounds

Exact errror In practice, we use numerical estimates because we *don't know* the exact value.

Error estimates and bounds

Exact errror In practice, we use numerical estimates because we *don't know* the exact value. In this situation, we can't compute the actual error!

Error estimates and bounds

Exact errror In practice, we use numerical estimates because we *don't know* the exact value. In this situation, we can't compute the actual error!

However, sometimes we use a numerical method for a problem for which we have an exact answer to test the method itself! Here, the exact error indicates the quality of the method.

Error estimates and bounds

Exact errror In practice, we use numerical estimates because we *don't know* the exact value. In this situation, we can't compute the actual error!

However, sometimes we use a numerical method for a problem for which we have an exact answer to test the method itself! Here, the exact error indicates the quality of the method.

Error estimates An *error estimate* is a value $\tilde{e}$ such that $|p^* - p| \approx \tilde{e}$.

Error estimates and bounds

Exact errror In practice, we use numerical estimates because we *don't know* the exact value. In this situation, we can't compute the actual error!

However, sometimes we use a numerical method for a problem for which we have an exact answer to test the method itself! Here, the exact error indicates the quality of the method.

Error estimates An *error estimate* is a value $\tilde{e}$ such that $|p^* - p| \approx \tilde{e}$.

Error bounds An *error bound* is a value $\bar{e}$ such that $|p^* - p| \leq \bar{e}$.

Error specification

Decimal places Requesting an answer to n *decimal places* corresponds to an *absolute* error of 10^{-n}.

Error specification

Decimal places Requesting an answer to n *decimal places* corresponds to an *absolute* error of 10^{-n}.

Significant figures Requesting an answer to n *significant figures* corresponds to a *relative* error of roughly 10^{-n}.

Error specification

Decimal places Requesting an answer to n *decimal places* corresponds to an *absolute* error of 10^{-n}.

Significant figures Requesting an answer to n *significant figures* corresponds to a *relative* error of roughly 10^{-n}.

Error specification In general, it is better to request a given number of *significant figures* when computing a *positive* quantity (e.g. area), since this is independent of the units used.

Error specification

Decimal places Requesting an answer to n *decimal places* corresponds to an *absolute* error of 10^{-n}.

Significant figures Requesting an answer to n *significant figures* corresponds to a *relative* error of roughly 10^{-n}.

Error specification In general, it is better to request a given number of *significant figures* when computing a *positive* quantity (e.g. area), since this is independent of the units used.

For a *dimensionless* quantity which may be *positive or negative*, it is usually better to request a given number of *decimal places*.

Error specification

Decimal places Requesting an answer to n *decimal places* corresponds to an *absolute* error of 10^{-n}.

Significant figures Requesting an answer to n *significant figures* corresponds to a *relative* error of roughly 10^{-n}.

Error specification In general, it is better to request a given number of *significant figures* when computing a *positive* quantity (e.g. area), since this is independent of the units used.

For a *dimensionless* quantity which may be *positive or negative*, it is usually better to request a given number of *decimal places*.

If we request a number of significant figures, and the quantity is near zero, then we often need to compute with very high precision!

Error specification

Decimal places Requesting an answer to n *decimal places* corresponds to an *absolute* error of 10^{-n}.

Significant figures Requesting an answer to n *significant figures* corresponds to a *relative* error of roughly 10^{-n}.

Error specification In general, it is better to request a given number of *significant figures* when computing a *positive* quantity (e.g. area), since this is independent of the units used.

For a *dimensionless* quantity which may be *positive or negative*, it is usually better to request a given number of *decimal places*.

If we request a number of significant figures, and the quantity is near zero, then we often need to compute with very high precision!

For a physical quantity which may be positive or negative, it is best to specify an accuracy *relative* to a characteristic scaling for the problem.

Error specification

Decimal places Requesting an answer to n *decimal places* corresponds to an *absolute* error of 10^{-n}.

Significant figures Requesting an answer to n *significant figures* corresponds to a *relative* error of roughly 10^{-n}.

Error specification In general, it is better to request a given number of *significant figures* when computing a *positive* quantity (e.g. area), since this is independent of the units used.

For a *dimensionless* quantity which may be *positive or negative*, it is usually better to request a given number of *decimal places*.

> If we request a number of significant figures, and the quantity is near zero, then we often need to compute with very high precision!

For a physical quantity which may be positive or negative, it is best to specify an accuracy *relative* to a characteristic scaling for the problem.

> e.g. For the difference in surface area of two balls whose diameter is measured using a ruler with $1\mathrm{mm}$ markings, might aim find the answer to within $10\mathrm{mm}^2$.

Rounded arithmetic

Floating-point When working with floating-point numbers, the result of *every* arithmetical operation is rounded to the nearest representable number.

Rounded arithmetic

Floating-point When working with floating-point numbers, the result of *every* arithmetical operation is rounded to the nearest representable number.

Accumulation of rounding errors can cause significant errors in the final result.

Rounded arithmetic

Floating-point When working with floating-point numbers, the result of *every* arithmetical operation is rounded to the nearest representable number.

Accumulation of rounding errors can cause significant errors in the final result.

Decimal rounded arithmetic We can simulate the effect of round-off errors by performing hand calculations to a fixed number of *decimal* significant figures.

Rounded arithmetic

Floating-point When working with floating-point numbers, the result of *every* arithmetical operation is rounded to the nearest representable number.

Accumulation of rounding errors can cause significant errors in the final result.

Decimal rounded arithmetic We can simulate the effect of round-off errors by performing hand calculations to a fixed number of *decimal* significant figures.

Example Exact computation

$$\pi \times e = 3.14159\cdots \times 2.71828\cdots = 8.53973\cdots$$

Rounded arithmetic

Floating-point When working with floating-point numbers, the result of *every* arithmetical operation is rounded to the nearest representable number.

Accumulation of rounding errors can cause significant errors in the final result.

Decimal rounded arithmetic We can simulate the effect of round-off errors by performing hand calculations to a fixed number of *decimal* significant figures.

Example Exact computation

$$\pi \times e = 3.14159\cdots \times 2.71828\cdots = 8.53973\cdots = 8.54\ (3\,\mathsf{sf})$$

Rounded arithmetic

Floating-point When working with floating-point numbers, the result of *every* arithmetical operation is rounded to the nearest representable number.

Accumulation of rounding errors can cause significant errors in the final result.

Decimal rounded arithmetic We can simulate the effect of round-off errors by performing hand calculations to a fixed number of *decimal* significant figures.

Example Exact computation

$$\pi \times e = 3.14159\cdots \times 2.71828\cdots = 8.53973\cdots = 8.54\ (3\,\textsf{sf}) = 8.5\ (2\,\textsf{sf}).$$

Rounded arithmetic

Floating-point When working with floating-point numbers, the result of *every* arithmetical operation is rounded to the nearest representable number.

Accumulation of rounding errors can cause significant errors in the final result.

Decimal rounded arithmetic We can simulate the effect of round-off errors by performing hand calculations to a fixed number of *decimal* significant figures.

Example Exact computation

$$\pi \times e = 3.14159\cdots \times 2.71828\cdots = 8.53973\cdots = 8.54\ (3\,\mathsf{sf}) = 8.5\ (2\,\mathsf{sf}).$$

Three-digit rounded arithmetic:

$$\pi \times e \overset{3\,\mathsf{sf}}{\approx} 3.14 \times 2.72 = 8.5408 \overset{3\,\mathsf{sf}}{\approx} 8.54.$$

Rounded arithmetic

Floating-point When working with floating-point numbers, the result of *every* arithmetical operation is rounded to the nearest representable number.

Accumulation of rounding errors can cause significant errors in the final result.

Decimal rounded arithmetic We can simulate the effect of round-off errors by performing hand calculations to a fixed number of *decimal* significant figures.

Example Exact computation

$$\pi \times e = 3.14159\cdots \times 2.71828\cdots = 8.53973\cdots = 8.54\ (3\,\mathsf{sf}) = 8.5\ (2\,\mathsf{sf}).$$

Three-digit rounded arithmetic:

$$\pi \times e \overset{3\,\mathsf{sf}}{\approx} 3.14 \times 2.72 = 8.5408 \overset{3\,\mathsf{sf}}{\approx} 8.54.$$

Two-digit rounded arithmetic:

$$\pi \times e \overset{2\,\mathsf{sf}}{\approx} 3.1 \times 2.7 = 8.37 \overset{2\,\mathsf{sf}}{\approx} 8.4.$$

Rounded arithmetic

Floating-point When working with floating-point numbers, the result of *every* arithmetical operation is rounded to the nearest representable number.

Accumulation of rounding errors can cause significant errors in the final result.

Decimal rounded arithmetic We can simulate the effect of round-off errors by performing hand calculations to a fixed number of *decimal* significant figures.

Example Exact computation

$$\pi \times e = 3.14159\cdots \times 2.71828\cdots = 8.53973\cdots = 8.54\ (3\,\mathsf{sf}) = 8.5\ (2\,\mathsf{sf}).$$

Three-digit rounded arithmetic:

$$\pi \times e \overset{3\,\mathsf{sf}}{\approx} 3.14 \times 2.72 = 8.5408 \overset{3\,\mathsf{sf}}{\approx} 8.54.$$

Two-digit rounded arithmetic:

$$\pi \times e \overset{2\,\mathsf{sf}}{\approx} 3.1 \times 2.7 = 8.37 \overset{2\,\mathsf{sf}}{\approx} 8.4.$$

Important Round after *every* operation!

Rounded arithmetic

Floating-point When working with floating-point numbers, the result of *every* arithmetical operation is rounded to the nearest representable number.

Accumulation of rounding errors can cause significant errors in the final result.

Decimal rounded arithmetic We can simulate the effect of round-off errors by performing hand calculations to a fixed number of *decimal* significant figures.

Example Exact computation

$$\pi \times e = 3.14159\cdots \times 2.71828\cdots = 8.53973\cdots = 8.54\ (3\,\mathsf{sf}) = 8.5\ (2\,\mathsf{sf}).$$

Three-digit rounded arithmetic:

$$\pi \times e \stackrel{3\,\mathsf{sf}}{\approx} 3.14 \times 2.72 = 8.5408 \stackrel{3\,\mathsf{sf}}{\approx} 8.54.$$

Two-digit rounded arithmetic:

$$\pi \times e \stackrel{2\,\mathsf{sf}}{\approx} 3.1 \times 2.7 = 8.37 \stackrel{2\,\mathsf{sf}}{\approx} 8.4.$$

Important Round after *every* operation!

$$\pi \times e^2 \stackrel{2\,\mathsf{sf}}{\approx} 3.1 \times 2.7^2 = 3.1 \times 7.29 \stackrel{2\,\mathsf{sf}}{\approx} 3.1 \times 7.3 = 22.63 \stackrel{2\,\mathsf{sf}}{\approx} 23..$$

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using 3-digit rounded arithmetic. Compare your answer to the exact value.

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using 3-digit rounded arithmetic. Compare your answer to the exact value.

$$x^2 = x \times x = 4.89 \times 4.89 = 23.9121 \overset{3\,\text{sf}}{\approx} 23.9$$

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using 3-digit rounded arithmetic. Compare your answer to the exact value.

$$x^2 = x \times x = 4.89 \times 4.89 = 23.9121 \overset{3\,\text{sf}}{\approx} 23.9$$

$$x^3 = x^2 \times x \overset{3\,\text{sf}}{\approx} 23.9 \times 4.89 = 116.871 \overset{3\,\text{sf}}{\approx} 117.$$

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using 3-digit rounded arithmetic. Compare your answer to the exact value.

$$x^2 = x \times x = 4.89 \times 4.89 = 23.9121 \overset{3\,\text{sf}}{\approx} 23.9$$

$$x^3 = x^2 \times x \overset{3\,\text{sf}}{\approx} 23.9 \times 4.89 = 116.871 \overset{3\,\text{sf}}{\approx} 117.$$

$$5.34x^2 = 5.34 \times 4.89^2 \overset{3\,\text{sf}}{\approx} 5.34 \times 23.9 = 127.626 \overset{3\,\text{sf}}{\approx} 128.$$

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using 3-digit rounded arithmetic. Compare your answer to the exact value.

$$x^2 = x \times x = 4.89 \times 4.89 = 23.9121 \overset{3\,\text{sf}}{\approx} 23.9$$

$$x^3 = x^2 \times x \overset{3\,\text{sf}}{\approx} 23.9 \times 4.89 = 116.871 \overset{3\,\text{sf}}{\approx} 117.$$

$$5.34x^2 = 5.34 \times 4.89^2 \overset{3\,\text{sf}}{\approx} 5.34 \times 23.9 = 127.626 \overset{3\,\text{sf}}{\approx} 128.$$

$$1.52x = 1.52 \times 4.89 = 7.4328 \overset{3\,\text{sf}}{\approx} 7.43$$

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using 3-digit rounded arithmetic. Compare your answer to the exact value.

$$
\begin{aligned}
x^2 &= x \times x = 4.89 \times 4.89 = 23.9121 \overset{3\,\text{sf}}{\approx} 23.9 \\
x^3 &= x^2 \times x \overset{3\,\text{sf}}{\approx} 23.9 \times 4.89 = 116.871 \overset{3\,\text{sf}}{\approx} 117. \\
5.34x^2 &= 5.34 \times 4.89^2 \overset{3\,\text{sf}}{\approx} 5.34 \times 23.9 = 127.626 \overset{3\,\text{sf}}{\approx} 128. \\
1.52x &= 1.52 \times 4.89 = 7.4328 \overset{3\,\text{sf}}{\approx} 7.43 \\
f(x) &= ((x^3 - 5.34x^2) + 1.52x) + 4.61
\end{aligned}
$$

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using 3-digit rounded arithmetic. Compare your answer to the exact value.

$$x^2 = x \times x = 4.89 \times 4.89 = 23.9121 \overset{3\,\mathsf{sf}}{\approx} 23.9$$

$$x^3 = x^2 \times x \overset{3\,\mathsf{sf}}{\approx} 23.9 \times 4.89 = 116.871 \overset{3\,\mathsf{sf}}{\approx} 117.$$

$$5.34x^2 = 5.34 \times 4.89^2 \overset{3\,\mathsf{sf}}{\approx} 5.34 \times 23.9 = 127.626 \overset{3\,\mathsf{sf}}{\approx} 128.$$

$$1.52x = 1.52 \times 4.89 = 7.4328 \overset{3\,\mathsf{sf}}{\approx} 7.43$$

$$\begin{aligned} f(x) &= ((x^3 - 5.34x^2) + 1.52x) + 4.61 \\ &\overset{3\,\mathsf{sf}}{\approx} ((117. - 128.) + 7.43) + 4.61 \end{aligned}$$

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using 3-digit rounded arithmetic. Compare your answer to the exact value.

$$
\begin{aligned}
x^2 &= x \times x = 4.89 \times 4.89 = 23.9121 \overset{3\,\mathrm{sf}}{\approx} 23.9 \\
x^3 &= x^2 \times x \overset{3\,\mathrm{sf}}{\approx} 23.9 \times 4.89 = 116.871 \overset{3\,\mathrm{sf}}{\approx} 117. \\
5.34x^2 &= 5.34 \times 4.89^2 \overset{3\,\mathrm{sf}}{\approx} 5.34 \times 23.9 = 127.626 \overset{3\,\mathrm{sf}}{\approx} 128. \\
1.52x &= 1.52 \times 4.89 = 7.4328 \overset{3\,\mathrm{sf}}{\approx} 7.43 \\
f(x) &= ((x^3 - 5.34x^2) + 1.52x) + 4.61 \\
&\overset{3\,\mathrm{sf}}{\approx} ((117. - 128.) + 7.43) + 4.61 = (-11.0 + 7.43) + 4.61 = -3.57 + 4.61
\end{aligned}
$$

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using 3-digit rounded arithmetic. Compare your answer to the exact value.

$$\begin{aligned}
x^2 &= x \times x = 4.89 \times 4.89 = 23.9121 \overset{3\,\mathsf{sf}}{\approx} 23.9 \\
x^3 &= x^2 \times x \overset{3\,\mathsf{sf}}{\approx} 23.9 \times 4.89 = 116.871 \overset{3\,\mathsf{sf}}{\approx} 117. \\
5.34x^2 &= 5.34 \times 4.89^2 \overset{3\,\mathsf{sf}}{\approx} 5.34 \times 23.9 = 127.626 \overset{3\,\mathsf{sf}}{\approx} 128. \\
1.52x &= 1.52 \times 4.89 = 7.4328 \overset{3\,\mathsf{sf}}{\approx} 7.43 \\
f(x) &= ((x^3 - 5.34x^2) + 1.52x) + 4.61 \\
&\overset{3\,\mathsf{sf}}{\approx} ((117. - 128.) + 7.43) + 4.61 = (-11.0 + 7.43) + 4.61 = -3.57 + 4.61 \\
&\qquad = 1.04.
\end{aligned}$$

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using 3-digit rounded arithmetic. Compare your answer to the exact value.

$$x^2 = x \times x = 4.89 \times 4.89 = 23.9121 \overset{3\,\text{sf}}{\approx} 23.9$$

$$x^3 = x^2 \times x \overset{3\,\text{sf}}{\approx} 23.9 \times 4.89 = 116.871 \overset{3\,\text{sf}}{\approx} 117.$$

$$5.34x^2 = 5.34 \times 4.89^2 \overset{3\,\text{sf}}{\approx} 5.34 \times 23.9 = 127.626 \overset{3\,\text{sf}}{\approx} 128.$$

$$1.52x = 1.52 \times 4.89 = 7.4328 \overset{3\,\text{sf}}{\approx} 7.43$$

$$\begin{aligned} f(x) &= ((x^3 - 5.34x^2) + 1.52x) + 4.61 \\ &\overset{3\,\text{sf}}{\approx} ((117. - 128.) + 7.43) + 4.61 = (-11.0 + 7.43) + 4.61 = -3.57 + 4.61 \\ &= 1.04. \end{aligned}$$

Exact answer $f(4.89) = 1.282355 = 1.28$ $(3\,\text{sf})$.

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using 3-digit rounded arithmetic. Compare your answer to the exact value.

$$
\begin{aligned}
x^2 &= x \times x = 4.89 \times 4.89 = 23.9121 \overset{3\,\text{sf}}{\approx} 23.9 \\
x^3 &= x^2 \times x \overset{3\,\text{sf}}{\approx} 23.9 \times 4.89 = 116.871 \overset{3\,\text{sf}}{\approx} 117. \\
5.34x^2 &= 5.34 \times 4.89^2 \overset{3\,\text{sf}}{\approx} 5.34 \times 23.9 = 127.626 \overset{3\,\text{sf}}{\approx} 128. \\
1.52x &= 1.52 \times 4.89 = 7.4328 \overset{3\,\text{sf}}{\approx} 7.43 \\
f(x) &= ((x^3 - 5.34x^2) + 1.52x) + 4.61 \\
&\overset{3\,\text{sf}}{\approx} ((117. - 128.) + 7.43) + 4.61 = (-11.0 + 7.43) + 4.61 = -3.57 + 4.61 \\
&\quad = 1.04.
\end{aligned}
$$

Exact answer $f(4.89) = 1.282355 = 1.28$ (3 sf).

Relative error 19%, even though each step has a relative error of 0.1%!

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using single-precision arithmetic. Compare your answer to the exact value (estimated using double-precision arithmetic).

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using single-precision arithmetic. Compare your answer to the exact value (estimated using double-precision arithmetic).

In Matlab:

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using single-precision arithmetic. Compare your answer to the exact value (estimated using double-precision arithmetic).

In Matlab:

```
c=[1.00,-5.34,1.52,4.61], x=4.89
```

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using single-precision arithmetic. Compare your answer to the exact value (estimated using double-precision arithmetic).

In Matlab:

```
c=[1.00,-5.34,1.52,4.61], x=4.89
fx=polyval(c,x)
```

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using single-precision arithmetic. Compare your answer to the exact value (estimated using double-precision arithmetic).

In Matlab:

```
c=[1.00,-5.34,1.52,4.61], x=4.89
fx=polyval(c,x)
sc=single(c), sx=single(x)
```

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using single-precision arithmetic. Compare your answer to the exact value (estimated using double-precision arithmetic).

In Matlab:

```
c=[1.00,-5.34,1.52,4.61], x=4.89
fx=polyval(c,x)
sc=single(c), sx=single(x)
sfx=polyval(sc,sx)
```

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using single-precision arithmetic. Compare your answer to the exact value (estimated using double-precision arithmetic).

In Matlab:

```
c=[1.00,-5.34,1.52,4.61], x=4.89
fx=polyval(c,x)
sc=single(c), sx=single(x)
sfx=polyval(sc,sx)
es = abs(double(sfx)-fx)/abs(fx)
```

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using single-precision arithmetic. Compare your answer to the exact value (estimated using double-precision arithmetic).

In Matlab:

```
c=[1.00,-5.34,1.52,4.61], x=4.89
fx=polyval(c,x)
sc=single(c), sx=single(x)
sfx=polyval(sc,sx)
es = abs(double(sfx)-fx)/abs(fx)
```

Exact value $f(x) = 1.282355$ (given by `fx`).

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using single-precision arithmetic. Compare your answer to the exact value (estimated using double-precision arithmetic).

In Matlab:

```
c=[1.00,-5.34,1.52,4.61], x=4.89
fx=polyval(c,x)
sc=single(c), sx=single(x)
sfx=polyval(sc,sx)
es = abs(double(sfx)-fx)/abs(fx)
```

Exact value $f(x) = 1.282355$ (given by `fx`).

Single-precison result $f(x) \approx 1.2823482$ (given by `sfx`).

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using single-precision arithmetic. Compare your answer to the exact value (estimated using double-precision arithmetic).

In Matlab:

```
c=[1.00,-5.34,1.52,4.61], x=4.89
fx=polyval(c,x)
sc=single(c), sx=single(x)
sfx=polyval(sc,sx)
es = abs(double(sfx)-fx)/abs(fx)
```

Exact value $f(x) = 1.282355$ (given by `fx`).

Single-precison result $f(x) \approx 1.2823482$ (given by `sfx`).

Absolute error of 6.84×10^{-6}, relative error 5.3×10^{-6}.

Example of rounded arithmetic

Example Let $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$. Evaluate f at 4.89 using single-precision arithmetic. Compare your answer to the exact value (estimated using double-precision arithmetic).

In Matlab:

```
c=[1.00,-5.34,1.52,4.61], x=4.89
fx=polyval(c,x)
sc=single(c), sx=single(x)
sfx=polyval(sc,sx)
es = abs(double(sfx)-fx)/abs(fx)
```

Exact value $f(x) = 1.282355$ (given by `fx`).

Single-precison result $f(x) \approx 1.2823482$ (given by `sfx`).

Absolute error of 6.84×10^{-6}, relative error 5.3×10^{-6}.

Again, the accumulated error 5.3×10^{-6} is much higher than the machine epsilon for single-precision $\epsilon = 2^{-23} \approx 1.2 \times 10^{-7}$.

Fixed-point versus floating-point

Fixed-point versus floating-point

Fixed-point Addition and subtraction are *exact* when working to a fixed number of decimal places!

Fixed-point versus floating-point

Fixed-point Addition and subtraction are *exact* when working to a fixed number of decimal places!

e.g. $148.41316 + 0.00067 \overset{5\,\text{dp}}{=} 148.41383; \quad 149.905 - 146.936 \overset{3\,\text{dp}}{=} 2.969.$

Fixed-point versus floating-point

Fixed-point Addition and subtraction are *exact* when working to a fixed number of decimal places!

e.g. $148.41316 + 0.00067 \overset{5\,\text{dp}}{=} 148.41383; \quad 149.905 - 146.936 \overset{3\,\text{dp}}{=} 2.969.$

Multiplication of small numbers behaves poorly in fixed-point arithmetic!

Fixed-point versus floating-point

Fixed-point Addition and subtraction are *exact* when working to a fixed number of decimal places!

e.g. $148.41316 + 0.00067 \overset{5\,\text{dp}}{=} 148.41383; \quad 149.905 - 146.936 \overset{3\,\text{dp}}{=} 2.969.$

Multiplication of small numbers behaves poorly in fixed-point arithmetic!

e.g. $0.00674 \times 0.00034 \overset{5\,\text{dp}}{\approx} 0.00000.$

Fixed-point versus floating-point

Fixed-point Addition and subtraction are *exact* when working to a fixed number of decimal places!

e.g. $148.41316 + 0.00067 \overset{5\,\text{dp}}{=} 148.41383; \quad 149.905 - 146.936 \overset{3\,\text{dp}}{=} 2.969.$

Multiplication of small numbers behaves poorly in fixed-point arithmetic!

e.g. $0.00674 \times 0.00034 \overset{5\,\text{dp}}{\approx} 0.00000.$

Floating-point

Fixed-point versus floating-point

Fixed-point Addition and subtraction are *exact* when working to a fixed number of decimal places!

e.g. $148.41316 + 0.00067 \overset{5\,\text{dp}}{=} 148.41383; \quad 149.905 - 146.936 \overset{3\,\text{dp}}{=} 2.969.$

Multiplication of small numbers behaves poorly in fixed-point arithmetic!

e.g. $0.00674 \times 0.00034 \overset{5\,\text{dp}}{\approx} 0.00000.$

Floating-point Under multiplication and division in floating-point arithmetic, small relative errors remain small!

Fixed-point versus floating-point

Fixed-point Addition and subtraction are *exact* when working to a fixed number of decimal places!

e.g. $148.41316 + 0.00067 \overset{5\,\text{dp}}{=} 148.41383; \quad 149.905 - 146.936 \overset{3\,\text{dp}}{=} 2.969.$

Multiplication of small numbers behaves poorly in fixed-point arithmetic!

e.g. $0.00674 \times 0.00034 \overset{5\,\text{dp}}{\approx} 0.00000.$

Floating-point Under multiplication and division in floating-point arithmetic, small relative errors remain small!

e.g. $403.4 \times 0.006738 = 2.7181092 \overset{4\,\text{sf}}{\approx} 2.718$

Fixed-point versus floating-point

Fixed-point Addition and subtraction are *exact* when working to a fixed number of decimal places!

e.g. $148.41316 + 0.00067 \overset{5\,\text{dp}}{=} 148.41383; \quad 149.905 - 146.936 \overset{3\,\text{dp}}{=} 2.969.$

Multiplication of small numbers behaves poorly in fixed-point arithmetic!

e.g. $0.00674 \times 0.00034 \overset{5\,\text{dp}}{\approx} 0.00000.$

Floating-point Under multiplication and division in floating-point arithmetic, small relative errors remain small!

e.g. $403.4 \times 0.006738 = 2.7181092 \overset{4\,\text{sf}}{\approx} 2.718$

e.g. $0.00674 \times 0.000335 \overset{3\,\text{sf}}{\approx} 0.0000022579 = 2.26 \times 10^{-6}$

Fixed-point versus floating-point

Fixed-point Addition and subtraction are *exact* when working to a fixed number of decimal places!

e.g. $148.41316 + 0.00067 \overset{5\,\text{dp}}{=} 148.41383; \quad 149.905 - 146.936 \overset{3\,\text{dp}}{=} 2.969.$

Multiplication of small numbers behaves poorly in fixed-point arithmetic!

e.g. $0.00674 \times 0.00034 \overset{5\,\text{dp}}{\approx} 0.00000.$

Floating-point Under multiplication and division in floating-point arithmetic, small relative errors remain small!

e.g. $403.4 \times 0.006738 = 2.7181092 \overset{4\,\text{sf}}{\approx} 2.718$

e.g. $0.00674 \times 0.000335 \overset{3\,\text{sf}}{\approx} 0.0000022579 = 2.26 \times 10^{-6}$

Subtraction of almost-equal numbers causes loss of precision!

Fixed-point versus floating-point

Fixed-point Addition and subtraction are *exact* when working to a fixed number of decimal places!

e.g. $148.41316 + 0.00067 \overset{5\,\text{dp}}{=} 148.41383; \quad 149.905 - 146.936 \overset{3\,\text{dp}}{=} 2.969.$

Multiplication of small numbers behaves poorly in fixed-point arithmetic!

e.g. $0.00674 \times 0.00034 \overset{5\,\text{dp}}{\approx} 0.00000.$

Floating-point Under multiplication and division in floating-point arithmetic, small relative errors remain small!

e.g. $403.4 \times 0.006738 = 2.7181092 \overset{4\,\text{sf}}{\approx} 2.718$

e.g. $0.00674 \times 0.000335 \overset{3\,\text{sf}}{\approx} 0.0000022579 = 2.26 \times 10^{-6}$

Subtraction of almost-equal numbers causes loss of precision!

e.g. $149.905 - 146.936 = 2.969$

Fixed-point versus floating-point

Fixed-point Addition and subtraction are *exact* when working to a fixed number of decimal places!

e.g. $148.41316 + 0.00067 \overset{5\,\text{dp}}{=} 148.41383; \quad 149.905 - 146.936 \overset{3\,\text{dp}}{=} 2.969.$

Multiplication of small numbers behaves poorly in fixed-point arithmetic!

e.g. $0.00674 \times 0.00034 \overset{5\,\text{dp}}{\approx} 0.00000.$

Floating-point Under multiplication and division in floating-point arithmetic, small relative errors remain small!

e.g. $403.4 \times 0.006738 = 2.7181092 \overset{4\,\text{sf}}{\approx} 2.718$

e.g. $0.00674 \times 0.000335 \overset{3\,\text{sf}}{\approx} 0.0000022579 = 2.26 \times 10^{-6}$

Subtraction of almost-equal numbers causes loss of precision!

e.g. $149.905 - 146.936 = 2.969 \overset{6\,\text{sf}}{=} 2.969\color{red}{00}.$

Accuracy and precision

Accuracy and precision

Accuracy Accurate to n digits mean n digits are correct (± 1 in last digit).

Accuracy and precision

Accuracy Accurate to n digits mean n digits are correct (± 1 in last digit).

Precision Precision is number of digits used.

Accuracy and precision

Accuracy Accurate to n digits mean n digits are correct (± 1 in last digit).

Precision Precision is number of digits used.

e.g. 3.1428571 is an approximation to $\pi = 3.1415926\cdots$ specified with a precision of 8 digits, but only accurate to 3 digits.

Accuracy and precision

Accuracy Accurate to n digits mean n digits are correct (± 1 in last digit).

Precision Precision is number of digits used.

e.g. 3.1428571 is an approximation to $\pi = 3.1415926\cdots$ specified with a precision of 8 digits, but only accurate to 3 digits.

Giving an answer *to higher precision than the accuracy is* useless*, and gives a false impression of the accuracy!!*

Accuracy and precision

Accuracy Accurate to n digits mean n digits are correct (± 1 in last digit).

Precision Precision is number of digits used.

e.g. 3.1428571 is an approximation to $\pi = 3.1415926\cdots$ specified with a precision of 8 digits, but only accurate to 3 digits.

Giving an answer *to higher precision than the accuracy is* useless*,*
and gives a false impression of the accuracy!!

A certain amount of extra precision is useful in *intermediate* values to prevent unnecessary loss of accuracy when rounding.

Working guidelines

Working guidelines

Final answer If not specifically asked for, use the precision appropriate for the accuracy.

Working guidelines

Final answer If not specifically asked for, use the precision appropriate for the accuracy.

e.g. If the accuracy is ± 0.02, give 2 decimal places of precision.

Working guidelines

Final answer If not specifically asked for, use the precision appropriate for the accuracy.

e.g. If the accuracy is ± 0.02, give 2 decimal places of precision.

Intermediates Use *more* precision for intermediate results than needed in final answer.

Working guidelines

Final answer If not specifically asked for, use the precision appropriate for the accuracy.

e.g. If the accuracy is ± 0.02, give 2 decimal places of precision.

Intermediates Use *more* precision for intermediate results than needed in final answer.

— For hand calculations, try to use at least two (decimal) significant figures more.

Working guidelines

Final answer If not specifically asked for, use the precision appropriate for the accuracy.

e.g. If the accuracy is ± 0.02, give 2 decimal places of precision.

Intermediates Use *more* precision for intermediate results than needed in final answer.

— For hand calculations, try to use at least two (decimal) significant figures more.

— For computer calculations, use machine precision in intermediate results (and if necessary, write out as for hand calculations).

Working guidelines

Final answer If not specifically asked for, use the precision appropriate for the accuracy.

e.g. If the accuracy is ± 0.02, give 2 decimal places of precision.

Intermediates Use *more* precision for intermediate results than needed in final answer.

— For hand calculations, try to use at least two (decimal) significant figures more.

— For computer calculations, use machine precision in intermediate results (and if necessary, write out as for hand calculations).

Errors Use *at most* two significant figues when giving an error (estimate).

Working guidelines

Final answer If not specifically asked for, use the precision appropriate for the accuracy.

e.g. If the accuracy is ± 0.02, give 2 decimal places of precision.

Intermediates Use *more* precision for intermediate results than needed in final answer.

— For hand calculations, try to use at least two (decimal) significant figures more.

— For computer calculations, use machine precision in intermediate results (and if necessary, write out as for hand calculations).

Errors Use *at most* two significant figues when giving an error (estimate).

— e.g. Absolute error 0.0013, relative error 0.04%.

Working guidelines

Final answer If not specifically asked for, use the precision appropriate for the accuracy.

e.g. If the accuracy is ± 0.02, give 2 decimal places of precision.

Intermediates Use *more* precision for intermediate results than needed in final answer.

- For hand calculations, try to use at least two (decimal) significant figures more.
- For computer calculations, use machine precision in intermediate results (and if necessary, write out as for hand calculations).

Errors Use *at most* two significant figues when giving an error (estimate).

- e.g. Absolute error 0.0013, relative error 0.04%.

If asked to compare an approximate value with the *exact* value, use *more* precision for the exact value!

Reducing Errors in Scientific Computing

Subtraction

Loss of significance When subtracting two almost-equal quantities in rounded or floating-point arithmetic, many significant figures of accuracy can be lost!

Example Compute $x^3 - y^3$ using three-digit arithmetic for $x = 427$, $y = 426$.

Subtraction

Loss of significance When subtracting two almost-equal quantities in rounded or floating-point arithmetic, many significant figures of accuracy can be lost!

Example Compute $x^3 - y^3$ using three-digit arithmetic for $x = 427$, $y = 426$.

$$x^3 - y^3 = 427^3 - 426^3 = 77854483 - 77308776$$

Subtraction

Loss of significance When subtracting two almost-equal quantities in rounded or floating-point arithmetic, many significant figures of accuracy can be lost!

Example Compute $x^3 - y^3$ using three-digit arithmetic for $x = 427$, $y = 426$.

$$\begin{aligned} x^3 - y^3 = 427^3 - 426^3 &= 77854483 - 77308776 \\ &\overset{3\,\text{sf}}{\approx} 77900000 - 77300000 \end{aligned}$$

Subtraction

Loss of significance When subtracting two almost-equal quantities in rounded or floating-point arithmetic, many significant figures of accuracy can be lost!

Example Compute $x^3 - y^3$ using three-digit arithmetic for $x = 427$, $y = 426$.

$$\begin{aligned} x^3 - y^3 = 427^3 - 426^3 &= 77854483 - 77308776 \\ &\overset{3\,\mathsf{sf}}{\approx} 77900000 - 77300000 = 600000 \overset{3\,\mathsf{sf}}{=} 6.00 \times 10^5. \end{aligned}$$

Subtraction

Loss of significance When subtracting two almost-equal quantities in rounded or floating-point arithmetic, many significant figures of accuracy can be lost!

Example Compute $x^3 - y^3$ using three-digit arithmetic for $x = 427$, $y = 426$.

$$x^3 - y^3 = 427^3 - 426^3 = 77854483 - 77308776$$
$$\overset{3\,\text{sf}}{\approx} 77900000 - 77300000 = 600000 \overset{3\,\text{sf}}{=} 6.00\times 10^5.$$

Exact answer 545707. High relative error of 9.9%.

Subtraction

Loss of significance When subtracting two almost-equal quantities in rounded or floating-point arithmetic, many significant figures of accuracy can be lost!

Example Compute $x^3 - y^3$ using three-digit arithmetic for $x = 427$, $y = 426$.

$$x^3 - y^3 = 427^3 - 426^3 = 77854483 - 77308776$$
$$\overset{3\,\text{sf}}{\approx} 77900000 - 77300000 = 600000 \overset{3\,\text{sf}}{=} 6.00 \times 10^5.$$

Exact answer 545707. High relative error of 9.9%.

Re-write $x^3 - y^3 = (x - y) \times (x^2 + xy + y^2)$.

Subtraction

Loss of significance When subtracting two almost-equal quantities in rounded or floating-point arithmetic, many significant figures of accuracy can be lost!

Example Compute $x^3 - y^3$ using three-digit arithmetic for $x = 427$, $y = 426$.

$$x^3 - y^3 = 427^3 - 426^3 = 77854483 - 77308776$$
$$\overset{3\,\mathrm{sf}}{\approx} 77900000 - 77300000 = 600000 \overset{3\,\mathrm{sf}}{=} 6.00 \times 10^5.$$

Exact answer 545707. High relative error of 9.9%.

Re-write $x^3 - y^3 = (x - y) \times (x^2 + xy + y^2)$. Then

$$x^3 - y^3 = (427 - 426) \times (427^2 + 427 \times 426 + 426^2)$$

Subtraction

Loss of significance When subtracting two almost-equal quantities in rounded or floating-point arithmetic, many significant figures of accuracy can be lost!

Example Compute $x^3 - y^3$ using three-digit arithmetic for $x = 427$, $y = 426$.

$$x^3 - y^3 = 427^3 - 426^3 = 77854483 - 77308776$$
$$\stackrel{3\,\text{sf}}{\approx} 77900000 - 77300000 = 600000 \stackrel{3\,\text{sf}}{=} 6.00 \times 10^5.$$

Exact answer 545707. High relative error of 9.9%.

Re-write $x^3 - y^3 = (x - y) \times (x^2 + xy + y^2)$. Then

$$x^3 - y^3 = (427 - 426) \times (427^2 + 427 \times 426 + 426^2)$$
$$= 1 \times (182329 + 181902 + 181476)$$

Subtraction

Loss of significance When subtracting two almost-equal quantities in rounded or floating-point arithmetic, many significant figures of accuracy can be lost!

Example Compute $x^3 - y^3$ using three-digit arithmetic for $x = 427$, $y = 426$.

$$x^3 - y^3 = 427^3 - 426^3 = 77854483 - 77308776$$

$$\overset{3\,\text{sf}}{\approx} 77900000 - 77300000 = 600000 \overset{3\,\text{sf}}{=} 6.00\times10^5.$$

Exact answer 545707. High relative error of 9.9%.

Re-write $x^3 - y^3 = (x-y) \times (x^2 + xy + y^2)$. Then

$$x^3 - y^3 = (427 - 426) \times (427^2 + 427\times426 + 426^2)$$

$$= 1 \times (182329 + 181902 + 181476)$$

$$\overset{3\,\text{sf}}{\approx} 1 \times (182000 + 182000 + 181000) = 545000.$$

Subtraction

Loss of significance When subtracting two almost-equal quantities in rounded or floating-point arithmetic, many significant figures of accuracy can be lost!

Example Compute $x^3 - y^3$ using three-digit arithmetic for $x = 427$, $y = 426$.

$$\begin{aligned} x^3 - y^3 &= 427^3 - 426^3 = 77854483 - 77308776 \\ &\overset{3\,\text{sf}}{\approx} 77900000 - 77300000 = 600000 \overset{3\,\text{sf}}{=} 6.00\times 10^5. \end{aligned}$$

Exact answer 545707. High relative error of 9.9%.

Re-write $x^3 - y^3 = (x - y) \times (x^2 + xy + y^2)$. Then

$$\begin{aligned} x^3 - y^3 &= (427 - 426) \times (427^2 + 427\times 426 + 426^2) \\ &= 1 \times (182329 + 181902 + 181476) \\ &\overset{3\,\text{sf}}{\approx} 1 \times (182000 + 182000 + 181000) = 545000. \end{aligned}$$

Exact answer 545707. Relative error $1.3\times 10^{-3} = 0.13\%$.

Subtraction

Loss of significance When subtracting two almost-equal quantities in rounded or floating-point arithmetic, many significant figures of accuracy can be lost!

Example Compute $x^3 - y^3$ using three-digit arithmetic for $x = 427$, $y = 426$.

$$\begin{aligned} x^3 - y^3 &= 427^3 - 426^3 = 77854483 - 77308776 \\ &\overset{3\,\text{sf}}{\approx} 77900000 - 77300000 = 600000 \overset{3\,\text{sf}}{=} 6.00\times 10^5. \end{aligned}$$

Exact answer 545707. High relative error of 9.9%.

Re-write $x^3 - y^3 = (x - y) \times (x^2 + xy + y^2)$. Then

$$\begin{aligned} x^3 - y^3 &= (427 - 426) \times (427^2 + 427\times 426 + 426^2) \\ &= 1 \times (182329 + 181902 + 181476) \\ &\overset{3\,\text{sf}}{\approx} 1 \times (182000 + 182000 + 181000) = 545000. \end{aligned}$$

Exact answer 545707. Relative error $1.3\times 10^{-3} = 0.13\%$.

Safe subtraction Subtraction of *exact* values *at the first step* is safe! This is because errors have not had a chance to accumulate.

Subtraction

Example Now compute $x^3 - y^3$ using single-precision arithmetic for the values $x = 427$, $y = 426$.

$$\begin{aligned} x^3 - y^3 = 427^3 - 426^3 &= 77854483 - 77308776 \\ &\overset{\text{sp}}{\approx} 77854480 - 77308776 = 545704 \overset{\text{sp}}{=} 545704. \end{aligned}$$

Exact answer 545707. Relative error 5.5×10^{-6}.

Re-write $x^3 - y^3 = (x - y) \times (x^2 + xy + y^2)$. Then

$$\begin{aligned} x^3 - y^3 &= (427 - 426) \times (427^2 + 427! \times 426 + 426^2) \\ &= 1 \times (182329 + 181902 + 181476) \\ &\overset{\text{sp}}{\approx} 1 \times (182329 + 181902 + 181476) = 545707 \overset{\text{sp}}{\approx} 545707. \end{aligned}$$

Answer is exact!

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Take $a = 0.5$, $b = 2$, $c = -0.05$.

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Take $a = 0.5$, $b = 2$, $c = -0.05$.

$$b^2 - 4ac = 2^2 - 4 \times 0.5 \times (-0.05) = 4 \times (-0.1) = 4.1,$$

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Take $a = 0.5$, $b = 2$, $c = -0.05$.

$$\begin{aligned} b^2 - 4ac &= 2^2 - 4 \times 0.5 \times (-0.05) = 4 \times (-0.1) = 4.1, \\ 2a &= 2 \times 0.5 = 1, \end{aligned}$$

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Take $a = 0.5$, $b = 2$, $c = -0.05$.

$$\begin{aligned} b^2 - 4ac &= 2^2 - 4 \times 0.5 \times (-0.05) = 4 \times (-0.1) = 4.1, \\ 2a &= 2 \times 0.5 = 1, \\ x &= \frac{\sqrt{b^2 - 4ac} - b}{2a} \end{aligned}$$

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Take $a = 0.5$, $b = 2$, $c = -0.05$.

$$\begin{aligned} b^2 - 4ac &= 2^2 - 4 \times 0.5 \times (-0.05) = 4 \times (-0.1) = 4.1, \\ 2a &= 2 \times 0.5 = 1, \\ x &= \frac{\sqrt{b^2 - 4ac} - b}{2a} = \frac{\sqrt{4.1} - 2}{1} \end{aligned}$$

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Take $a = 0.5$, $b = 2$, $c = -0.05$.

$$\begin{aligned} b^2 - 4ac &= 2^2 - 4 \times 0.5 \times (-0.05) = 4 \times (-0.1) = 4.1, \\ 2a &= 2 \times 0.5 = 1, \\ x &= \frac{\sqrt{b^2 - 4ac} - b}{2a} = \frac{\sqrt{4.1} - 2}{1} = \frac{2.02498 \cdots - 2}{1} \end{aligned}$$

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Take $a = 0.5$, $b = 2$, $c = -0.05$.

$$\begin{aligned} b^2 - 4ac &= 2^2 - 4 \times 0.5 \times (-0.05) = 4 \times (-0.1) = 4.1, \\ 2a &= 2 \times 0.5 = 1, \\ x &= \frac{\sqrt{b^2 - 4ac} - b}{2a} = \frac{\sqrt{4.1} - 2}{1} = \frac{2.02498 \cdots - 2}{1} \\ &\overset{3\,\text{sf}}{\approx} 2.02 - 2 = 0.02 \overset{3\,\text{sf}}{=} 0.0200. \end{aligned}$$

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Take $a = 0.5$, $b = 2$, $c = -0.05$.

$$\begin{aligned} b^2 - 4ac &= 2^2 - 4 \times 0.5 \times (-0.05) = 4 \times (-0.1) = 4.1, \\ 2a &= 2 \times 0.5 = 1, \\ x &= \frac{\sqrt{b^2 - 4ac} - b}{2a} = \frac{\sqrt{4.1} - 2}{1} = \frac{2.02498 \cdots - 2}{1} \\ &\overset{3\,\text{sf}}{\approx} 2.02 - 2 = 0.02 \overset{3\,\text{sf}}{=} 0.0200. \end{aligned}$$

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Take $a = 0.5$, $b = 2$, $c = -0.05$.

$$\begin{aligned} b^2 - 4ac &= 2^2 - 4 \times 0.5 \times (-0.05) = 4 \times (-0.1) = 4.1, \\ 2a &= 2 \times 0.5 = 1, \\ x &= \frac{\sqrt{b^2 - 4ac} - b}{2a} = \frac{\sqrt{4.1} - 2}{1} = \frac{2.02498\cdots - 2}{1} \\ &\overset{3\,\text{sf}}{\approx} 2.02 - 2 = 0.02 \overset{3\,\text{sf}}{=} 0.0200. \end{aligned}$$

Exact answer $0.02484567\cdots = 0.0248$ $(3\,\text{sf})$.

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Take $a = 0.5$, $b = 2$, $c = -0.05$.

$$\begin{aligned} b^2 - 4ac &= 2^2 - 4 \times 0.5 \times (-0.05) = 4 \times (-0.1) = 4.1, \\ 2a &= 2 \times 0.5 = 1, \\ x &= \frac{\sqrt{b^2 - 4ac} - b}{2a} = \frac{\sqrt{4.1} - 2}{1} = \frac{2.02498 \cdots - 2}{1} \\ &\overset{3\,\text{sf}}{\approx} 2.02 - 2 = 0.02 \overset{3\,\text{sf}}{=} 0.0200. \end{aligned}$$

Exact answer $0.02484567 \cdots = 0.0248\ (3\,\text{sf})$.

Absolute error $|0.02 - 0.02484567| = 0.00484567 \cdots = 0.0048\ (2\,\text{sf})$.

Quadratic formula

Problem Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

Use the quadratic formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Take $a = 0.5$, $b = 2$, $c = -0.05$.

$$\begin{aligned} b^2 - 4ac &= 2^2 - 4 \times 0.5 \times (-0.05) = 4 \times (-0.1) = 4.1, \\ 2a &= 2 \times 0.5 = 1, \\ x &= \frac{\sqrt{b^2 - 4ac} - b}{2a} = \frac{\sqrt{4.1} - 2}{1} = \frac{2.02498\cdots - 2}{1} \\ &\overset{3\,\text{sf}}{\approx} 2.02 - 2 = 0.02 \overset{3\,\text{sf}}{=} 0.0200. \end{aligned}$$

Exact answer $0.02484567\cdots = 0.0248\;(3\,\text{sf})$.

Absolute error $|0.02 - 0.02484567| = 0.00484567\cdots = 0.0048\,(2\,\text{sf})$.
Relative error $|0.00484567|/|0.02484567| = 0.195031\cdots = 0.20\,(2\,\text{sf}) \approx 20\%$!!

Quadratic formula

Rearrange the formula by completing the square:

$$
\begin{aligned}
x = \frac{\sqrt{b^2 - 4ac} - b}{2a} &= \frac{\sqrt{b^2 - 4ac} - b}{2a} \times \frac{\sqrt{b^2 - 4ac} + b}{\sqrt{b^2 - 4ac} + b} \\
&= \frac{(b^2 - 4ac) - b^2}{2a(\sqrt{b^2 - 4ac} + b)} = \frac{-4ac}{2a(\sqrt{b^2 - 4ac} + b)} \\
&= \frac{-2c}{\sqrt{b^2 - 4ac} + b}
\end{aligned}
$$

Quadratic formula

Rearrange the formula by completing the square:

$$\begin{aligned} x &= \frac{\sqrt{b^2-4ac}-b}{2a} = \frac{\sqrt{b^2-4ac}-b}{2a} \times \frac{\sqrt{b^2-4ac}+b}{\sqrt{b^2-4ac}+b} \\ &= \frac{(b^2-4ac)-b^2}{2a(\sqrt{b^2-4ac}+b)} = \frac{-4ac}{2a(\sqrt{b^2-4ac}+b)} \\ &= \frac{-2c}{\sqrt{b^2-4ac}+b} \end{aligned}$$

Example Compute the positive root of $0.5x^2+2x-0.05$ using 3-digit arithmetic.

Quadratic formula

Rearrange the formula by completing the square:

$$\begin{aligned} x &= \frac{\sqrt{b^2-4ac}-b}{2a} = \frac{\sqrt{b^2-4ac}-b}{2a} \times \frac{\sqrt{b^2-4ac}+b}{\sqrt{b^2-4ac}+b} \\ &= \frac{(b^2-4ac)-b^2}{2a(\sqrt{b^2-4ac}+b)} = \frac{-4ac}{2a(\sqrt{b^2-4ac}+b)} \\ &= \frac{-2c}{\sqrt{b^2-4ac}+b} \end{aligned}$$

Example Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

$$x = \frac{-2c}{\sqrt{b^2-4ac}+b}$$

Quadratic formula

Rearrange the formula by completing the square:

$$
\begin{aligned}
x &= \frac{\sqrt{b^2-4ac}-b}{2a} = \frac{\sqrt{b^2-4ac}-b}{2a} \times \frac{\sqrt{b^2-4ac}+b}{\sqrt{b^2-4ac}+b} \\
&= \frac{(b^2-4ac)-b^2}{2a(\sqrt{b^2-4ac}+b)} = \frac{-4ac}{2a(\sqrt{b^2-4ac}+b)} \\
&= \frac{-2c}{\sqrt{b^2-4ac}+b}
\end{aligned}
$$

Example Compute the positive root of $0.5x^2+2x-0.05$ using 3-digit arithmetic.

$$
x = \frac{-2c}{\sqrt{b^2-4ac}+b} = \frac{-2\times(-0.05)}{\sqrt{4.1}+2} = \frac{0.1}{2.02\cancel{498\cdots}+2}
$$

Quadratic formula

Rearrange the formula by completing the square:

$$\begin{aligned} x &= \frac{\sqrt{b^2-4ac}-b}{2a} = \frac{\sqrt{b^2-4ac}-b}{2a} \times \frac{\sqrt{b^2-4ac}+b}{\sqrt{b^2-4ac}+b} \\ &= \frac{(b^2-4ac)-b^2}{2a(\sqrt{b^2-4ac}+b)} = \frac{-4ac}{2a(\sqrt{b^2-4ac}+b)} \\ &= \frac{-2c}{\sqrt{b^2-4ac}+b} \end{aligned}$$

Example Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

$$\begin{aligned} x &= \frac{-2c}{\sqrt{b^2-4ac}+b} = \frac{-2\times(-0.05)}{\sqrt{4.1}+2} = \frac{0.1}{2.02\cancel{498\cdots}+2} \\ &\overset{3\,\text{sf}}{\approx} \frac{0.1}{2.02+2} \end{aligned}$$

Quadratic formula

Rearrange the formula by completing the square:

$$
\begin{aligned}
x &= \frac{\sqrt{b^2-4ac}-b}{2a} = \frac{\sqrt{b^2-4ac}-b}{2a} \times \frac{\sqrt{b^2-4ac}+b}{\sqrt{b^2-4ac}+b} \\
&= \frac{(b^2-4ac)-b^2}{2a(\sqrt{b^2-4ac}+b)} = \frac{-4ac}{2a(\sqrt{b^2-4ac}+b)} \\
&= \frac{-2c}{\sqrt{b^2-4ac}+b}
\end{aligned}
$$

Example Compute the positive root of $0.5x^2+2x-0.05$ using 3-digit arithmetic.

$$
\begin{aligned}
x &= \frac{-2c}{\sqrt{b^2-4ac}+b} = \frac{-2\times(-0.05)}{\sqrt{4.1}+2} = \frac{0.1}{2.02\cancel{498\cdots}+2} \\
&\overset{3\,\text{sf}}{\approx} \frac{0.1}{2.02+2} = 0.1/4.02 = 0.0248756\cdots \overset{3\,\text{sf}}{\approx} 0.0249.
\end{aligned}
$$

Quadratic formula

Rearrange the formula by completing the square:

$$
\begin{aligned}
x &= \frac{\sqrt{b^2-4ac}-b}{2a} = \frac{\sqrt{b^2-4ac}-b}{2a} \times \frac{\sqrt{b^2-4ac}+b}{\sqrt{b^2-4ac}+b} \\
&= \frac{(b^2-4ac)-b^2}{2a(\sqrt{b^2-4ac}+b)} = \frac{-4ac}{2a(\sqrt{b^2-4ac}+b)} \\
&= \frac{-2c}{\sqrt{b^2-4ac}+b}
\end{aligned}
$$

Example Compute the positive root of $0.5x^2+2x-0.05$ using 3-digit arithmetic.

$$
\begin{aligned}
x &= \frac{-2c}{\sqrt{b^2-4ac}+b} = \frac{-2\times(-0.05)}{\sqrt{4.1}+2} = \frac{0.1}{2.02\cancel{498\cdots}+2} \\
&\overset{3\,\text{sf}}{\approx} \frac{0.1}{2.02+2} = 0.1/4.02 = 0.0248756\cdots \overset{3\,\text{sf}}{\approx} 0.0249.
\end{aligned}
$$

Quadratic formula

Rearrange the formula by completing the square:

$$\begin{aligned} x &= \frac{\sqrt{b^2-4ac}-b}{2a} = \frac{\sqrt{b^2-4ac}-b}{2a} \times \frac{\sqrt{b^2-4ac}+b}{\sqrt{b^2-4ac}+b} \\ &= \frac{(b^2-4ac)-b^2}{2a(\sqrt{b^2-4ac}+b)} = \frac{-4ac}{2a(\sqrt{b^2-4ac}+b)} \\ &= \frac{-2c}{\sqrt{b^2-4ac}+b} \end{aligned}$$

Example Compute the positive root of $0.5x^2+2x-0.05$ using 3-digit arithmetic.

$$x = \frac{-2c}{\sqrt{b^2-4ac}+b} = \frac{-2\times(-0.05)}{\sqrt{4.1}+2} = \frac{0.1}{2.02\cancel{498}\cdots+2}$$

$$\overset{3\,\text{sf}}{\approx} \frac{0.1}{2.02+2} = 0.1/4.02 = 0.0248756\cdots \overset{3\,\text{sf}}{\approx} 0.0249.$$

Exact answer $x = 0.02484567\cdots = 0.0248\,(3\,\text{sf})$.

Quadratic formula

Rearrange the formula by completing the square:

$$\begin{aligned} x &= \frac{\sqrt{b^2-4ac}-b}{2a} = \frac{\sqrt{b^2-4ac}-b}{2a} \times \frac{\sqrt{b^2-4ac}+b}{\sqrt{b^2-4ac}+b} \\ &= \frac{(b^2-4ac)-b^2}{2a(\sqrt{b^2-4ac}+b)} = \frac{-4ac}{2a(\sqrt{b^2-4ac}+b)} \\ &= \frac{-2c}{\sqrt{b^2-4ac}+b} \end{aligned}$$

Example Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

$$x = \frac{-2c}{\sqrt{b^2-4ac}+b} = \frac{-2\times(-0.05)}{\sqrt{4.1}+2} = \frac{0.1}{2.02\cancel{498\cdots}+2}$$

$$\stackrel{3\,\text{sf}}{\approx} \frac{0.1}{2.02+2} = 0.1/4.02 = 0.0248756\cdots \stackrel{3\,\text{sf}}{\approx} 0.0249.$$

Exact answer $x = 0.02484567\cdots = 0.0248\,(3\,\text{sf})$.

Absolute error $|0.0249 - 0.02484567| = 0.00054326\cdots = 0.00054\,(2\,\text{sf})$.

Quadratic formula

Rearrange the formula by completing the square:

$$
\begin{aligned}
x = \frac{\sqrt{b^2-4ac}-b}{2a} &= \frac{\sqrt{b^2-4ac}-b}{2a} \times \frac{\sqrt{b^2-4ac}+b}{\sqrt{b^2-4ac}+b} \\
&= \frac{(b^2-4ac)-b^2}{2a(\sqrt{b^2-4ac}+b)} = \frac{-4ac}{2a(\sqrt{b^2-4ac}+b)} \\
&= \frac{-2c}{\sqrt{b^2-4ac}+b}
\end{aligned}
$$

Example Compute the positive root of $0.5x^2 + 2x - 0.05$ using 3-digit arithmetic.

$$
\begin{aligned}
x &= \frac{-2c}{\sqrt{b^2-4ac}+b} = \frac{-2\times(-0.05)}{\sqrt{4.1}+2} = \frac{0.1}{2.02\cancel{498\cdots}+2} \\
&\overset{3\,\text{sf}}{\approx} \frac{0.1}{2.02+2} = 0.1/4.02 = 0.0248756\cdots \overset{3\,\text{sf}}{\approx} 0.0249.
\end{aligned}
$$

Exact answer $x = 0.02484567\cdots = 0.0248\,(3\,\text{sf})$.

Absolute error $|0.0249 - 0.02484567| = 0.00054326\cdots = 0.00054\,(2\,\text{sf})$.

Relative error $|0.00054326|/|0.02484567| = 0.002187\cdots = 0.0021\,(2\,\text{sf}) \approx 0.2\%$

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

We previously found $f(x) \approx 1.04$ by direct evaluation; relative error 19%.

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Re-write in *nested form* (also known as *Horner's rule*):

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Re-write in *nested form* (also known as *Horner's rule*):

$$x^3 - 5.34x^2 + 1.52x + 4.61$$

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Re-write in *nested form* (also known as *Horner's rule*):

$$x^3 - 5.34x^2 + 1.52x + 4.61 = (x^2 - 5.34x + 1.52) \cdot x + 4.61$$

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Re-write in *nested form* (also known as *Horner's rule*):

$$
\begin{aligned}
x^3 - 5.34x^2 + 1.52x + 4.61 &= (x^2 - 5.34x + 1.52) \cdot x + 4.61 \\
&= \big((x - 5.34) \cdot x + 1.52\big) \cdot x + 4.61
\end{aligned}
$$

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Re-write in *nested form* (also known as *Horner's rule*):

$$\begin{aligned} x^3 - 5.34x^2 + 1.52x + 4.61 &= (x^2 - 5.34x + 1.52) \cdot x + 4.61 \\ &= \big((x - 5.34) \cdot x + 1.52\big) \cdot x + 4.61 \end{aligned}$$

$f(4.89) = ((4.89 - 5.34) \times 4.89 + 1.52) \times 4.89 + 4.61$

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Re-write in *nested form* (also known as *Horner's rule*):

$$\begin{aligned} x^3 - 5.34x^2 + 1.52x + 4.61 &= (x^2 - 5.34x + 1.52) \cdot x + 4.61 \\ &= \big((x - 5.34) \cdot x + 1.52\big) \cdot x + 4.61 \end{aligned}$$

$$\begin{aligned} f(4.89) &= ((4.89 - 5.34) \times 4.89 + 1.52) \times 4.89 + 4.61 \\ &= (-0.45 \times 4.89 + 1.52) \times 4.89 + 4.61 \end{aligned}$$

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Re-write in *nested form* (also known as *Horner's rule*):

$$\begin{aligned} x^3 - 5.34x^2 + 1.52x + 4.61 &= (x^2 - 5.34x + 1.52) \cdot x + 4.61 \\ &= \big((x - 5.34) \cdot x + 1.52\big) \cdot x + 4.61 \end{aligned}$$

$$\begin{aligned} f(4.89) &= ((4.89 - 5.34) \times 4.89 + 1.52) \times 4.89 + 4.61 \\ &= (-0.45 \times 4.89 + 1.52) \times 4.89 + 4.61 \\ &= (-2.2005 + 1.52) \times 4.89 + 4.61 \overset{3\,\text{sf}}{\approx} (-2.20 + 1.52) \times 4.89 + 4.61 \end{aligned}$$

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Re-write in *nested form* (also known as *Horner's rule*):

$$\begin{aligned} x^3 - 5.34x^2 + 1.52x + 4.61 &= (x^2 - 5.34x + 1.52) \cdot x + 4.61 \\ &= \big((x - 5.34) \cdot x + 1.52\big) \cdot x + 4.61 \end{aligned}$$

$$\begin{aligned} f(4.89) &= ((4.89 - 5.34) \times 4.89 + 1.52) \times 4.89 + 4.61 \\ &= (-0.45 \times 4.89 + 1.52) \times 4.89 + 4.61 \\ &= (-2.2005 + 1.52) \times 4.89 + 4.61 \overset{3\,\text{sf}}{\approx} (-2.20 + 1.52) \times 4.89 + 4.61 \\ &= -0.68 \times 4.89 + 4.61 = -3.3252 + 4.61 \overset{3\,\text{sf}}{\approx} -3.33 + 4.61 \end{aligned}$$

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Re-write in *nested form* (also known as *Horner's rule*):

$$\begin{aligned} x^3 - 5.34x^2 + 1.52x + 4.61 &= (x^2 - 5.34x + 1.52) \cdot x + 4.61 \\ &= \big((x - 5.34) \cdot x + 1.52\big) \cdot x + 4.61 \end{aligned}$$

$$\begin{aligned} f(4.89) &= ((4.89 - 5.34) \times 4.89 + 1.52) \times 4.89 + 4.61 \\ &= (-0.45 \times 4.89 + 1.52) \times 4.89 + 4.61 \\ &= (-2.2005 + 1.52) \times 4.89 + 4.61 \overset{3\,\text{sf}}{\approx} (-2.20 + 1.52) \times 4.89 + 4.61 \\ &\quad = -0.68 \times 4.89 + 4.61 = -3.3252 + 4.61 \overset{3\,\text{sf}}{\approx} -3.33 + 4.61 \\ &\quad\quad \overset{3\,\text{sf}}{=} 1.28 \end{aligned}$$

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Re-write in *nested form* (also known as *Horner's rule*):

$$\begin{aligned} x^3 - 5.34x^2 + 1.52x + 4.61 &= (x^2 - 5.34x + 1.52) \cdot x + 4.61 \\ &= \big((x - 5.34) \cdot x + 1.52\big) \cdot x + 4.61 \end{aligned}$$

$$\begin{aligned} f(4.89) &= ((4.89 - 5.34) \times 4.89 + 1.52) \times 4.89 + 4.61 \\ &= (-0.45 \times 4.89 + 1.52) \times 4.89 + 4.61 \\ &= (-2.2005 + 1.52) \times 4.89 + 4.61 \overset{3\,\text{sf}}{\approx} (-2.20 + 1.52) \times 4.89 + 4.61 \\ &\quad = -0.68 \times 4.89 + 4.61 = -3.3252 + 4.61 \overset{3\,\text{sf}}{\approx} -3.33 + 4.61 \\ &\qquad \overset{3\,\text{sf}}{=} 1.28 \end{aligned}$$

Exact answer $f(4.89) = 1.282355 = 1.28$ (3 sf).

Polynomials in Horner nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic.

Re-write in *nested form* (also known as *Horner's rule*):

$$\begin{aligned} x^3 - 5.34x^2 + 1.52x + 4.61 &= (x^2 - 5.34x + 1.52) \cdot x + 4.61 \\ &= \big((x - 5.34) \cdot x + 1.52\big) \cdot x + 4.61 \end{aligned}$$

$$\begin{aligned} f(4.89) &= ((4.89 - 5.34) \times 4.89 + 1.52) \times 4.89 + 4.61 \\ &= (-0.45 \times 4.89 + 1.52) \times 4.89 + 4.61 \\ &= (-2.2005 + 1.52) \times 4.89 + 4.61 \overset{3\,\text{sf}}{\approx} (-2.20 + 1.52) \times 4.89 + 4.61 \\ &\quad = -0.68 \times 4.89 + 4.61 = -3.3252 + 4.61 \overset{3\,\text{sf}}{\approx} -3.33 + 4.61 \\ &\quad\quad \overset{3\,\text{sf}}{=} 1.28 \end{aligned}$$

Exact answer $f(4.89) = 1.282355 = 1.28$ $(3\,\text{sf})$.

Correct to given precision!

Nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic in Matlab.

Nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic in Matlab.

Use `round(x,n,'significant')` or the `rnd(x,n)` method on the Student Portal to round `x` to `n` significant figures.

Nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic in Matlab.

Use `round(x,n,'significant')` or the `rnd(x,n)` method on the Student Portal to round `x` to `n` significant figures.

Use the shorthand `r=@(x)round(x,3,'significant')` to reduce implementation.

Nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic in Matlab.

Use `round(x,n,'significant')` or the `rnd(x,n)` method on the Student Portal to round `x` to `n` significant figures.

Use the shorthand `r=@(x)round(x,3,'significant')` to reduce implementation.

```
c=[1.0,-5.34,1.52,4.61]
fdirect = @(x) c(1)*x^3 + c(2)*x^2 + c(3)*x + c(4)
fnested = @(x) ((c(1)*x+c(2))*x+c(3))*x+c(4)
fdirectrounded = @(x) r(r(r(r(r(x*x)*x)-r(5.34*r(x*x)))
                        +r(1.52*x))+4.61)
fnestedrounded = @(x) r(r(r(r(r(x-5.34)*x)+1.52)*x)+4.61)
```

Nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic in Matlab.

Use `round(x,n,'significant')` or the `rnd(x,n)` method on the Student Portal to round `x` to `n` significant figures.

Use the shorthand `r=@(x)round(x,3,'significant')` to reduce implementation.

```
c=[1.0,-5.34,1.52,4.61]
fdirect = @(x) c(1)*x^3 + c(2)*x^2 + c(3)*x + c(4)
fnested = @(x) ((c(1)*x+c(2))*x+c(3))*x+c(4)
fdirectrounded = @(x) r(r(r(r(r(x*x)*x)-r(5.34*r(x*x)))
                          +r(1.52*x))+4.61)
fnestedrounded = @(x) r(r(r(r(r(x-5.34)*x)+1.52)*x)+4.61)
```

Alternatively, use the `Rounded` class from the Student Portal.

Nested form

Problem Evaluate $f(x) = x^3 - 5.34x^2 + 1.52x + 4.61$ at $x = 4.89$ using 3-digit arithmetic in Matlab.

Use `round(x,n,'significant')` or the `rnd(x,n)` method on the Student Portal to round `x` to `n` significant figures.

Use the shorthand `r=@(x)round(x,3,'significant')` to reduce implementation.

```
c=[1.0,-5.34,1.52,4.61]
fdirect = @(x) c(1)*x^3 + c(2)*x^2 + c(3)*x + c(4)
fnested = @(x) ((c(1)*x+c(2))*x+c(3))*x+c(4)
fdirectrounded = @(x) r(r(r(r(r(x*x)*x)-r(5.34*r(x*x)))
                       +r(1.52*x))+4.61)
fnestedrounded = @(x) r(r(r(r(r(x-5.34)*x)+1.52)*x)+4.61)
```

Alternatively, use the `Rounded` class from the Student Portal.

```
xr=Rounded(x,3)
ydr=fdirect(xr); ydr.value
ynr=fnested(xr); ynr.value
```

Nested form

The nested form of

$$\sum_{k=0}^{n} a_k x^k = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0$$

is

$$\Big(\Big((\cdots(a_n\, x + a_{n-1}) \cdot x + \cdots) \cdot x + a_2) \cdot x + a_1) \cdot x + a_0\Big)\Big)$$

Nested form

The nested form of

$$\sum_{k=0}^{n} a_k x^k = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0$$

is

$$\Big(\Big(\big(\cdots\big(a_n\, x + a_{n-1}\big) \cdot x + \cdots\big) \cdot x + a_2\big) \cdot x + a_1\big) \cdot x + a_0\Big)\Big)$$

Here, the formula is simply evaluated from left to right.

Nested form

The nested form of

$$\sum_{k=0}^{n} a_k x^k = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0$$

is

$$\Big(\Big((\cdots(a_n\, x + a_{n-1}) \cdot x + \cdots) \cdot x + a_2) \cdot x + a_1\Big) \cdot x + a_0\Big)\Big)$$

Here, the formula is simply evaluated from left to right.

Alternatively, starting with the lowest power first:

$$\sum_{k=0}^{n} = a_0 + x \cdot \Big(a_1 + x \cdot \Big(a_2 + x \cdot (\cdots + x \cdot (a_{n-1} + x \cdot a_n) \cdots)\Big)\Big)$$

But here, we evaluate from right to left.

Nested form

The nested form of

$$\sum_{k=0}^{n} a_k x^k = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0$$

is

$$\Big(\big((\cdots(a_n\, x + a_{n-1}) \cdot x + \cdots) \cdot x + a_2\big) \cdot x + a_1\big) \cdot x + a_0\Big)$$

Here, the formula is simply evaluated from left to right.

Alternatively, starting with the lowest power first:

$$\sum_{k=0}^{n} = a_0 + x \cdot \Big(a_1 + x \cdot \Big(a_2 + x \cdot (\cdots + x \cdot (a_{n-1} + x \cdot a_n) \cdots)\Big)\Big)$$

But here, we evaluate from right to left.

e.g. For $n = 5$,

$$\begin{aligned} & a_5 x^5 + a_4 x^4 + a_3 x^3 + a_2 x^2 + a_1 x + a_0 \\ & \qquad = ((((a_5 \cdot x + a_4) \cdot x + a_3) \cdot x + a_2) \cdot x + a_1) \cdot x + a_0 \\ & \qquad = a_0 + x \cdot (a_1 + x \cdot (a_2 + x \cdot (a_3 + x \cdot (a_4 + x \cdot a_5)))) \end{aligned}$$

Quality of methods

A *good* method for a problem will *always* give an accurate answer, regardless of the input.

Quality of methods

A *good* method for a problem will *always* give an accurate answer, regardless of the input.

A *bad* method gives an inaccurate answer on *some* (or *most*) inputs, but *may* give a (very) accurate answer in some cases.

Quality of methods

A *good* method for a problem will *always* give an accurate answer, regardless of the input.

A *bad* method gives an inaccurate answer on *some* (or *most*) inputs,
but *may* give a (very) accurate answer in some cases.

e.g. Horner's method does not *always* give a more accurate result than direct evaluation, does not have as bad a worst-case.

Solutions of Equations of One Variable

Algebraic equations

Example problem

Algebraic equations

Example problem Suppose $x^2 = a$.

Algebraic equations

Example problem Suppose $x^2 = a$. Can compute a easily from x by multiplication.

Algebraic equations

Example problem Suppose $x^2 = a$. Can compute a easily from x by multiplication. But how can we determine x given a? i.e. Compute $x = \sqrt{a}$.

Algebraic equations

Example problem Suppose $x^2 = a$. Can compute a easily from x by multiplication. But how can we determine x given a? i.e. Compute $x = \sqrt{a}$.

Approach Solve the equation $f(x) = x^2 - a = 0$ for x in terms of a.

Algebraic equations

Example problem Suppose $x^2 = a$. Can compute a easily from x by multiplication. But how can we determine x given a? i.e. Compute $x = \sqrt{a}$.

Approach Solve the equation $f(x) = x^2 - a = 0$ for x in terms of a.

Example problem Suppose we know variables x and y are related by

$$\cos(x) - x + e^x y + y^3 = 0.$$

How can we determine y for various values of x? Or x for a given value of y?

Algebraic equations

Example problem Suppose $x^2 = a$. Can compute a easily from x by multiplication. But how can we determine x given a? i.e. Compute $x = \sqrt{a}$.

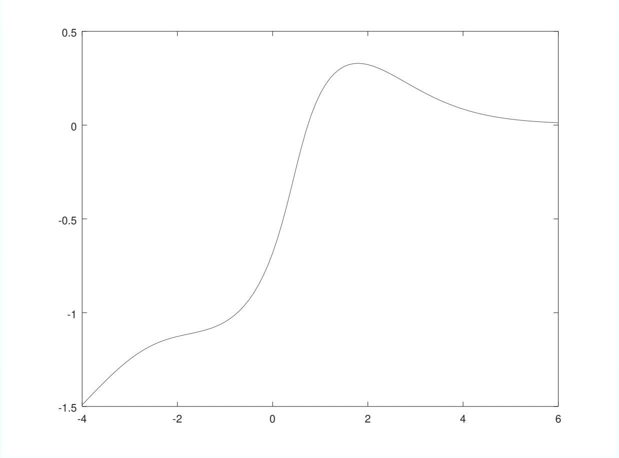
Approach Solve the equation $f(x) = x^2 - a = 0$ for x in terms of a.

Example problem Suppose we know variables x and y are related by

$$\cos(x) - x + e^x y + y^3 = 0.$$

How can we determine y for various values of x? Or x for a given value of y?

Approach Fix x-values $(x_0, x_1, \ldots, x_n)$, and try to find y-values $(y_0, y_1, \ldots, y_n)$. i.e. Solve equation of the form $f(x_i, y) = 0$ to find y_i.



Algebraic equations

General Problem Given a continuous function $f : \mathbb{R} \to \mathbb{R}$ and real numbers $a < b$, solve $f(x) = 0$ for $x \in [a, b]$.

Algebraic equations

General Problem Given a continuous function $f : \mathbb{R} \to \mathbb{R}$ and real numbers $a < b$, solve $f(x) = 0$ for $x \in [a, b]$.

Roots A value p such that $f(p) = 0$. p is called a *root* of f.

Algebraic equations

General Problem Given a continuous function $f : \mathbb{R} \to \mathbb{R}$ and real numbers $a < b$, solve $f(x) = 0$ for $x \in [a, b]$.

Roots A value p such that $f(p) = 0$. p is called a *root* of f.
Note that f may have many roots in $[a, b]$, or none at all!

Algebraic equations

General Problem Given a continuous function $f : \mathbb{R} \to \mathbb{R}$ and real numbers $a < b$, solve $f(x) = 0$ for $x \in [a, b]$.

Roots A value p such that $f(p) = 0$. p is called a *root* of f.
Note that f may have many roots in $[a, b]$, or none at all!

Approximation Given a tolerance ϵ, compute some p^* within ϵ of an actual root p.

Algebraic equations

General Problem Given a continuous function $f : \mathbb{R} \to \mathbb{R}$ and real numbers $a < b$, solve $f(x) = 0$ for $x \in [a, b]$.

Roots A value p such that $f(p) = 0$. p is called a *root* of f.
Note that f may have many roots in $[a, b]$, or none at all!

Approximation Given a tolerance ϵ, compute some p^* within ϵ of an actual root p.

Error The (absolute) *error* is $|p^* - p|$.

Algebraic equations

General Problem Given a continuous function $f : \mathbb{R} \to \mathbb{R}$ and real numbers $a < b$, solve $f(x) = 0$ for $x \in [a, b]$.

Roots A value p such that $f(p) = 0$. p is called a *root* of f.
Note that f may have many roots in $[a, b]$, or none at all!

Approximation Given a tolerance ϵ, compute some p^* within ϵ of an actual root p.

Error The (absolute) *error* is $|p^* - p|$.

Residual The *residual* is $|f(p^*)|$.

Algebraic equations

General Problem Given a continuous function $f : \mathbb{R} \to \mathbb{R}$ and real numbers $a < b$, solve $f(x) = 0$ for $x \in [a, b]$.

Roots A value p such that $f(p) = 0$. p is called a *root* of f.
Note that f may have many roots in $[a, b]$, or none at all!

Approximation Given a tolerance ϵ, compute some p^* within ϵ of an actual root p.

Error The (absolute) *error* is $|p^* - p|$.

Residual The *residual* is $|f(p^*)|$.

Example $f(x) = x^2 - 2$ has root $p = \sqrt{2} = 1.414\cdots$; approximate by $p^* = 1.4$

Algebraic equations

General Problem Given a continuous function $f : \mathbb{R} \to \mathbb{R}$ and real numbers $a < b$, solve $f(x) = 0$ for $x \in [a, b]$.

Roots A value p such that $f(p) = 0$. p is called a *root* of f.
Note that f may have many roots in $[a, b]$, or none at all!

Approximation Given a tolerance ϵ, compute some p^* within ϵ of an actual root p.

Error The (absolute) *error* is $|p^* - p|$.

Residual The *residual* is $|f(p^*)|$.

Example $f(x) = x^2 - 2$ has root $p = \sqrt{2} = 1.414\cdots$; approximate by $p^* = 1.4$
Error: $|p^* - p| = 0.014\cdots = 1.4\times 10^{-2}$ $(2\,\text{sf})$;

Algebraic equations

General Problem Given a continuous function $f : \mathbb{R} \to \mathbb{R}$ and real numbers $a < b$, solve $f(x) = 0$ for $x \in [a, b]$.

Roots A value p such that $f(p) = 0$. p is called a *root* of f.
Note that f may have many roots in $[a, b]$, or none at all!

Approximation Given a tolerance ϵ, compute some p^* within ϵ of an actual root p.

Error The (absolute) *error* is $|p^* - p|$.

Residual The *residual* is $|f(p^*)|$.

Example $f(x) = x^2 - 2$ has root $p = \sqrt{2} = 1.414\cdots$; approximate by $p^* = 1.4$
Error: $|p^* - p| = 0.014\cdots = 1.4\times 10^{-2}$ $(2\,\text{sf})$;
Residual: $|f(p^*)| = |1.4^2 - 2| = |1.96 - 2| = 0.04 = 4 \times 10^{-2}$.

Existence of solutions

Intermediate value theorem

Existence of solutions

Intermediate value theorem Suppose

(i) $f : [a, b] \to \mathbb{R}$ is continuous, and

Existence of solutions

Intermediate value theorem Suppose

(i) $f : [a, b] \to \mathbb{R}$ is continuous, and

(ii) $f(a)$ and $f(b)$ have different signs
(i.e. $f(a) < 0$ and $f(b) > 0$, or $f(a) > 0$ and $f(b) < 0$).

Existence of solutions

Intermediate value theorem Suppose

(i) $f : [a, b] \to \mathbb{R}$ is continuous, and

(ii) $f(a)$ and $f(b)$ have different signs
(i.e. $f(a) < 0$ and $f(b) > 0$, or $f(a) > 0$ and $f(b) < 0$).

Then f has a root in (a, b).

Existence of solutions

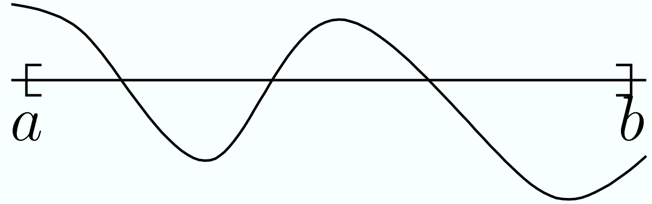
Intermediate value theorem Suppose

(i) $f : [a, b] \to \mathbb{R}$ is continuous, and

(ii) $f(a)$ and $f(b)$ have different signs

(i.e. $f(a) < 0$ and $f(b) > 0$, or $f(a) > 0$ and $f(b) < 0$).

Then f has a root in (a, b).



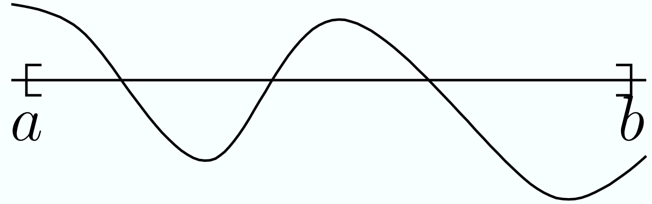
Existence of solutions

Intermediate value theorem Suppose

(i) $f : [a, b] \to \mathbb{R}$ is continuous, and

(ii) $f(a)$ and $f(b)$ have different signs
(i.e. $f(a) < 0$ and $f(b) > 0$, or $f(a) > 0$ and $f(b) < 0$).

Then f has a root in (a, b).



Bracket Call $[a, b]$ a *bracket* for the root(s) of f.

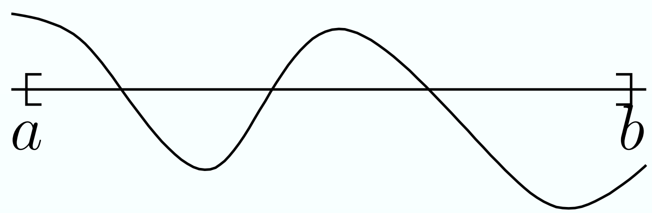
Existence of solutions

Intermediate value theorem Suppose

(i) $f : [a, b] \to \mathbb{R}$ is continuous, and

(ii) $f(a)$ and $f(b)$ have different signs
(i.e. $f(a) < 0$ and $f(b) > 0$, or $f(a) > 0$ and $f(b) < 0$).

Then f has a root in (a, b).



Bracket Call $[a, b]$ a *bracket* for the root(s) of f.

Signs Note that if $f(a), f(b) \neq 0$, then

$$\operatorname{sgn}(f(a)) \neq \operatorname{sgn}(f(b)) \iff f(a)f(b) < 0.$$

The bisection method

Problem Find a root of f given a bracket $[a, b]$.

The bisection method

Problem Find a root of f given a bracket $[a, b]$.

Idea Shrink the bracket $[a, b]$ to a point while preserving the bracket property.

The bisection method

Problem Find a root of f given a bracket $[a, b]$.

Idea Shrink the bracket $[a, b]$ to a point while preserving the bracket property.

Method Let c be the midpoint of $[a, b]$, which is given by $c = \dfrac{a+b}{2}$.

The bisection method

Problem Find a root of f given a bracket $[a, b]$.

Idea Shrink the bracket $[a, b]$ to a point while preserving the bracket property.

Method Let c be the midpoint of $[a, b]$, which is given by $c = \dfrac{a+b}{2}$.

If $f(c) = 0$, then c is a root.

The bisection method

Problem Find a root of f given a bracket $[a, b]$.

Idea Shrink the bracket $[a, b]$ to a point while preserving the bracket property.

Method Let c be the midpoint of $[a, b]$, which is given by $c = \dfrac{a+b}{2}$.

If $f(c) = 0$, then c is a root.

Otherwise $\operatorname{sgn}(f(c))$ differs from either $\operatorname{sgn}(f(a))$ or $\operatorname{sgn}(f(b))$.

The bisection method

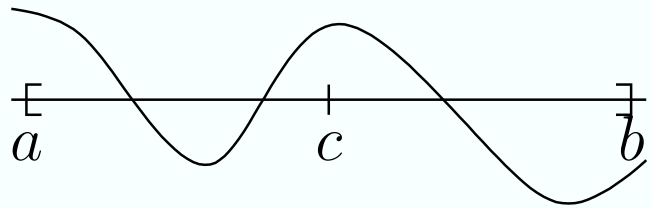
Problem Find a root of f given a bracket $[a, b]$.

Idea Shrink the bracket $[a, b]$ to a point while preserving the bracket property.

Method Let c be the midpoint of $[a, b]$, which is given by $c = \dfrac{a+b}{2}$.

If $f(c) = 0$, then c is a root.

Otherwise $\operatorname{sgn}(f(c))$ differs from either $\operatorname{sgn}(f(a))$ or $\operatorname{sgn}(f(b))$.



The bisection method

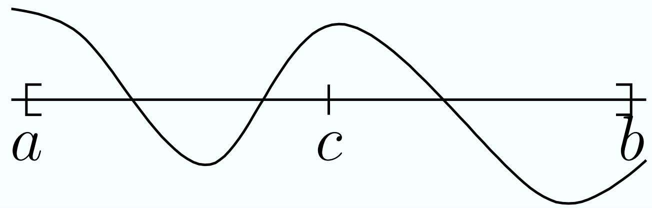
Problem Find a root of f given a bracket $[a, b]$.

Idea Shrink the bracket $[a, b]$ to a point while preserving the bracket property.

Method Let c be the midpoint of $[a, b]$, which is given by $c = \dfrac{a+b}{2}$.

If $f(c) = 0$, then c is a root.

Otherwise $\operatorname{sgn}(f(c))$ differs from either $\operatorname{sgn}(f(a))$ or $\operatorname{sgn}(f(b))$.



Update $a := c$ if $\operatorname{sgn}(f(a)) = \operatorname{sgn}(f(c)) \neq \operatorname{sgn}(f(b))$,
or $b := c$ if $\operatorname{sgn}(f(a)) \neq \operatorname{sgn}(f(c)) = \operatorname{sgn}(f(b))$.

The bisection method

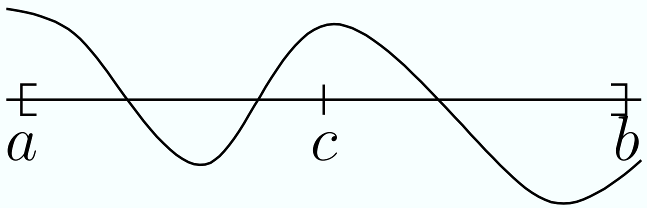
Problem Find a root of f given a bracket $[a, b]$.

Idea Shrink the bracket $[a, b]$ to a point while preserving the bracket property.

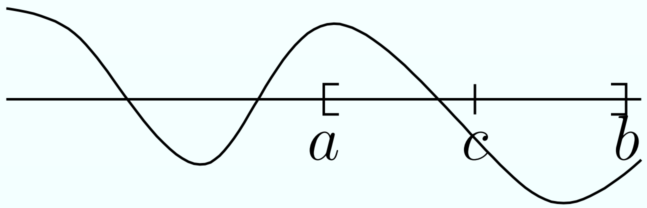
Method Let c be the midpoint of $[a, b]$, which is given by $c = \dfrac{a+b}{2}$.

If $f(c) = 0$, then c is a root.

Otherwise $\operatorname{sgn}(f(c))$ differs from either $\operatorname{sgn}(f(a))$ or $\operatorname{sgn}(f(b))$.



Update $a := c$ if $\operatorname{sgn}(f(a)) = \operatorname{sgn}(f(c)) \neq \operatorname{sgn}(f(b))$,
or $b := c$ if $\operatorname{sgn}(f(a)) \neq \operatorname{sgn}(f(c)) = \operatorname{sgn}(f(b))$.



The bisection method

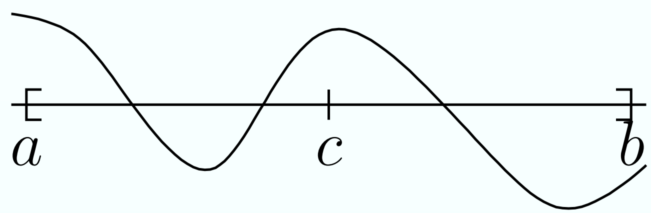
Problem Find a root of f given a bracket $[a, b]$.

Idea Shrink the bracket $[a, b]$ to a point while preserving the bracket property.

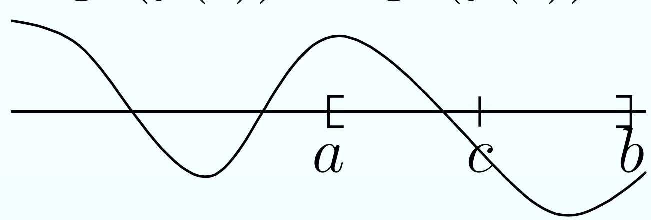
Method Let c be the midpoint of $[a, b]$, which is given by $c = \dfrac{a+b}{2}$.

If $f(c) = 0$, then c is a root.

Otherwise $\operatorname{sgn}(f(c))$ differs from either $\operatorname{sgn}(f(a))$ or $\operatorname{sgn}(f(b))$.



Update $a := c$ if $\operatorname{sgn}(f(a)) = \operatorname{sgn}(f(c)) \neq \operatorname{sgn}(f(b))$,
or $b := c$ if $\operatorname{sgn}(f(a)) \neq \operatorname{sgn}(f(c)) = \operatorname{sgn}(f(b))$.



The width of the interval $[a, b]$ is halved.

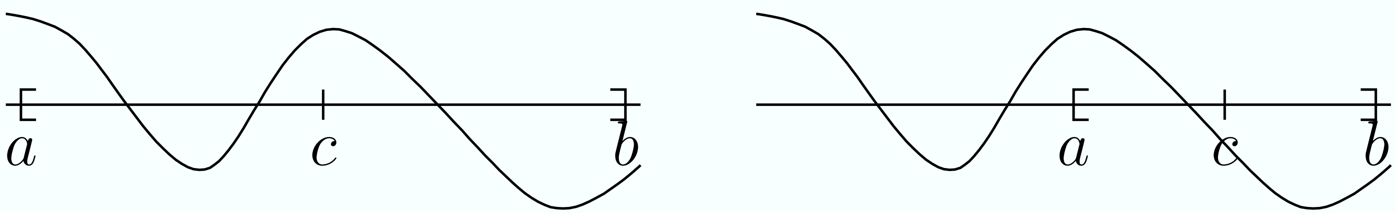
The bisection method

Problem Find a root of f given a bracket $[a, b]$.

Idea Shrink the bracket $[a, b]$ to a point while preserving the bracket property.

Method Let c be the midpoint of $[a, b]$, which is given by $c = \dfrac{a+b}{2}$.

Update $a := c$ if $\operatorname{sgn}(f(a)) = \operatorname{sgn}(f(c)) \neq \operatorname{sgn}(f(b))$,
or $b := c$ if $\operatorname{sgn}(f(a)) \neq \operatorname{sgn}(f(c)) = \operatorname{sgn}(f(b))$.



Termination Stop when we can locate the root to within a tolerance ϵ.

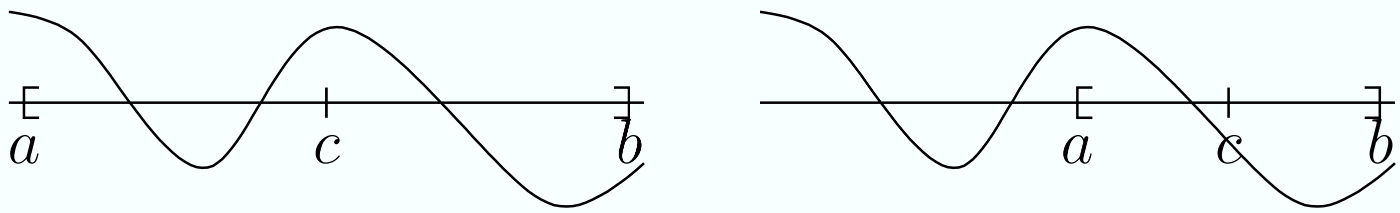
The bisection method

Problem Find a root of f given a bracket $[a, b]$.

Idea Shrink the bracket $[a, b]$ to a point while preserving the bracket property.

Method Let c be the midpoint of $[a, b]$, which is given by $c = \dfrac{a+b}{2}$.

Update $a := c$ if $\operatorname{sgn}(f(a)) = \operatorname{sgn}(f(c)) \neq \operatorname{sgn}(f(b))$,
or $b := c$ if $\operatorname{sgn}(f(a)) \neq \operatorname{sgn}(f(c)) = \operatorname{sgn}(f(b))$.



Termination Stop when we can locate the root to within a tolerance ϵ.

If the radius of $[a, b]$, which is given by $\dfrac{b-a}{2}$, is less than ϵ, then any point in $[a, b]$, *including the root p,* is within ϵ of the midpoint c.

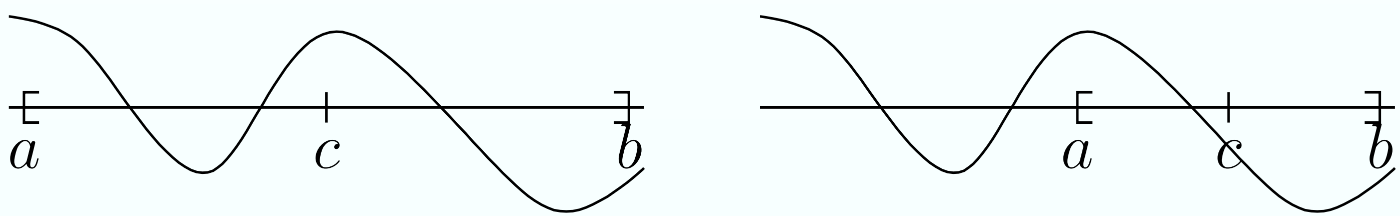
The bisection method

Problem Find a root of f given a bracket $[a, b]$.

Idea Shrink the bracket $[a, b]$ to a point while preserving the bracket property.

Method Let c be the midpoint of $[a, b]$, which is given by $c = \dfrac{a+b}{2}$.

Update $a := c$ if $\operatorname{sgn}(f(a)) = \operatorname{sgn}(f(c)) \neq \operatorname{sgn}(f(b))$,
or $b := c$ if $\operatorname{sgn}(f(a)) \neq \operatorname{sgn}(f(c)) = \operatorname{sgn}(f(b))$.



Termination Stop when we can locate the root to within a tolerance ϵ.

If the radius of $[a, b]$, which is given by $\dfrac{b-a}{2}$, is less than ϵ, then any point in $[a, b]$, *including the root* p*,* is within ϵ of the midpoint c.

Taking $p^* = (a+b)/2$ then yields $|p^* - p| < \epsilon$.

Iterative methods

Iterative methods The bisection method is an *iterative* method: we apply the same steps over and over again.

Iterative methods are typically implemented as a loop:

input f, a, b, ϵ such that $\mathrm{sgn}(f(a)) \neq \mathrm{sgn}(f(b)) < 0$ and $\epsilon > 0$. while $(b-a)/2 > \epsilon$,

$c := (a+b)/2$;

if $\mathrm{sgn}(f(c)) = \mathrm{sgn}(f(a))$

then $a := c,\ b := b$

else $a := a,\ b := c$

end if

end while

$r := (a+b)/2$

Iterative methods

Iterative methods The bisection method is an *iterative* method: we apply the same steps over and over again.

Iterative methods are typically implemented as a loop:

input f, a, b, ϵ such that $\operatorname{sgn}(f(a)) \neq \operatorname{sgn}(f(b)) < 0$ and $\epsilon > 0$. while $(b-a)/2 > \epsilon$,

$c := (a+b)/2;$

if $\operatorname{sgn}(f(c)) = \operatorname{sgn}(f(a))$

then $a := c,\ b := b$

else $a := a,\ b := c$

end if

end while

$r := (a+b)/2$

Here, we *overwrite* variables as they are no longer needed.

Iterative methods

Iterative methods The bisection method is an *iterative* method: we apply the same steps over and over again.

Indexed values In mathematical work, or if a record of previous values is needed, we often *index* the variables by the loop-count:

input f, a_0, b_0, ϵ such that $\operatorname{sgn}(f(a_0)) \neq \operatorname{sgn}(f(b))$ and $\epsilon > 0$. n:=0;

while $(b_n - a_n)/2 > \epsilon$,

$c_n := (a_n + b_n)/2;$

if $\operatorname{sgn}(f(c_n)) = \operatorname{sgn}(f(a_n))$

then $a_{n+1} := c_n,\ b_{n+1} := b_n$

else $a_{n+1} := a_n,\ b_{n+1} := c_n$

end if

$n := n + 1$

end while

$r := (a_n + b_n)/2$

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Compute $\sqrt{2}$ by solving $x^2 = 2$, or equivalently, $x^2 - 2 = 0$.
Since $1 < \sqrt{2} < 2$, solve in interval $[1, 2]$.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Compute $\sqrt{2}$ by solving $x^2 = 2$, or equivalently, $x^2 - 2 = 0$.
Since $1 < \sqrt{2} < 2$, solve in interval $[1, 2]$.

So need to find a root of $f(x) = x^2 - 2$, with initial $a = 1$ and $b = 2$.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Compute $\sqrt{2}$ by solving $x^2 = 2$, or equivalently, $x^2 - 2 = 0$.
Since $1 < \sqrt{2} < 2$, solve in interval $[1, 2]$.

So need to find a root of $f(x) = x^2 - 2$, with initial $a = 1$ and $b = 2$.

Compute $f(a) = f(1) = 1^2 - 2 = -1$ and $f(b) = f(2) = 2^2 - 2 = +2$.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Compute $\sqrt{2}$ by solving $x^2 = 2$, or equivalently, $x^2 - 2 = 0$.
Since $1 < \sqrt{2} < 2$, solve in interval $[1, 2]$.

So need to find a root of $f(x) = x^2 - 2$, with initial $a = 1$ and $b = 2$.

Compute $f(a) = f(1) = 1^2 - 2 = -1$ and $f(b) = f(2) = 2^2 - 2 = +2$.

The midpoint of the interval $[1, 2]$ is $\frac{1+2}{2} = 1.5$, so set $c = 1.5$.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Compute $\sqrt{2}$ by solving $x^2 = 2$, or equivalently, $x^2 - 2 = 0$.
Since $1 < \sqrt{2} < 2$, solve in interval $[1, 2]$.

So need to find a root of $f(x) = x^2 - 2$, with initial $a = 1$ and $b = 2$.

Compute $f(a) = f(1) = 1^2 - 2 = -1$ and $f(b) = f(2) = 2^2 - 2 = +2$.

The midpoint of the interval $[1, 2]$ is $\frac{1+2}{2} = 1.5$, so set $c = 1.5$.

Compute $f(c) = f(1.5) = 1.5^2 - 2 = 2.25 - 2 = 0.25$.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Compute $\sqrt{2}$ by solving $x^2 = 2$, or equivalently, $x^2 - 2 = 0$.
Since $1 < \sqrt{2} < 2$, solve in interval $[1, 2]$.

So need to find a root of $f(x) = x^2 - 2$, with initial $a = 1$ and $b = 2$.

Compute $f(a) = f(1) = 1^2 - 2 = -1$ and $f(b) = f(2) = 2^2 - 2 = +2$.

The midpoint of the interval $[1, 2]$ is $\frac{1+2}{2} = 1.5$, so set $c = 1.5$.

Compute $f(c) = f(1.5) = 1.5^2 - 2 = 2.25 - 2 = 0.25$.

Since $f(c) > 0$ has the opposite sign to $f(a)$, keep $a = 1.0$ and set $b := c = 1.5$.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Continue by finding a root of $f(x) = x^2 - 2$ in the interval $[a, b] = [1.0, 1.5]$.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Continue by finding a root of $f(x) = x^2 - 2$ in the interval $[a, b] = [1.0, 1.5]$.

Set $c = \frac{a+b}{2} = \frac{1.0+1.5}{2} = 1.25$.

Compute $f(c) = 1.25^2 - 2 = 1.5625 - 2 = -0.4375$.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Continue by finding a root of $f(x) = x^2 - 2$ in the interval $[a, b] = [1.0, 1.5]$.

Set $c = \frac{a+b}{2} = \frac{1.0+1.5}{2} = 1.25$.

Compute $f(c) = 1.25^2 - 2 = 1.5625 - 2 = -0.4375$.

Since $f(c) < 0$ has the opposite sign to $f(b)$,
set $a := c = 1.25$ and keep $b = 1.5$.

Set $c = \frac{a+b}{2} = \frac{1.25+1.5}{2} = 1.375$.

Compute $f(c) = 1.375^2 - 2 = -0.109375$.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Continue by finding a root of $f(x) = x^2 - 2$ in the interval $[a, b] = [1.0, 1.5]$.

Set $c = \frac{a+b}{2} = \frac{1.0+1.5}{2} = 1.25$.

Compute $f(c) = 1.25^2 - 2 = 1.5625 - 2 = -0.4375$.

Since $f(c) < 0$ has the opposite sign to $f(b)$,
set $a := c = 1.25$ and keep $b = 1.5$.

Set $c = \frac{a+b}{2} = \frac{1.25+1.5}{2} = 1.375$.

Compute $f(c) = 1.375^2 - 2 = -0.109375$.

Since $f(c) < 0$ has the opposite sign to $f(b)$,
set $a := c = 1.375$ and keep $b = 1.5$.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Continue by finding a root of $f(x) = x^2 - 2$ in the interval $[a, b] = [1.0, 1.5]$.

Set $c = \frac{a+b}{2} = \frac{1.0+1.5}{2} = 1.25$.

Compute $f(c) = 1.25^2 - 2 = 1.5625 - 2 = -0.4375$.

Since $f(c) < 0$ has the opposite sign to $f(b)$,
set $a := c = 1.25$ and keep $b = 1.5$.

Set $c = \frac{a+b}{2} = \frac{1.25+1.5}{2} = 1.375$.

Compute $f(c) = 1.375^2 - 2 = -0.109375$.

Since $f(c) < 0$ has the opposite sign to $f(b)$,
set $a := c = 1.375$ and keep $b = 1.5$.

Since $(b - a)/2 = (1.5 - 1.375)/2 = 0.0625 < 0.1$, taking $p^* = 1.4375$, the midpoint of $[a, b]$, means $|p^* - \sqrt{2}| < 0.0625 < 0.1$.

The bisection method

Example Estimate $\sqrt{2}$ to within 0.1.

Continue by finding a root of $f(x) = x^2 - 2$ in the interval $[a, b] = [1.0, 1.5]$.

Set $c = \frac{a+b}{2} = \frac{1.0+1.5}{2} = 1.25$.

Compute $f(c) = 1.25^2 - 2 = 1.5625 - 2 = -0.4375$.

Since $f(c) < 0$ has the opposite sign to $f(b)$,
set $a := c = 1.25$ and keep $b = 1.5$.

Set $c = \frac{a+b}{2} = \frac{1.25+1.5}{2} = 1.375$.

Compute $f(c) = 1.375^2 - 2 = -0.109375$.

Since $f(c) < 0$ has the opposite sign to $f(b)$,
set $a := c = 1.375$ and keep $b = 1.5$.

Since $(b - a)/2 = (1.5 - 1.375)/2 = 0.0625 < 0.1$, taking $p^* = 1.4375$, the midpoint of $[a, b]$, means $|p^* - \sqrt{2}| < 0.0625 < 0.1$.

In fact, $|p^* - \sqrt{2}| = 0.023$ (2sf)

The bisection method-Example (Complete)

Example Estimate $\sqrt{2}$ to within 0.1.

Start with $a_0 = 1$ and $b_0 = 2$.

Set $c_0 = 1.5$ with $f(c_0) = 0.25 > 0$,
so f has a root in $[a_0, c_0] = [1, 1.5]$.

Update $a_1 = a_0 = 1.0$, $b_1 = c_0 = 1.5$.

Set $c_1 = \frac{a_1+b_1}{2} = \frac{1.0+1.5}{2} = 1.25$

Compute $f(c_1) = -0.4375 < 0$,
so f has a root in $[c_1, b_1] = [1.25, 1.5]$.

Update $a_2 = c_1 = 1.25$, $b_2 = b_1 = 1.5$.

Set $c_2 = \frac{a_2+b_2}{2} = \frac{1.25+1.5}{2} = 1.375$.

Since $f(c_2) = f(1.375) = -0.109375 < 0$,
f has a root in $[c_2, b_2] = [1.375, 1.5]$.

Update $a_3 = c_2 = 1.375$, $b_3 = b_2 = 1.5$.

$f(1) = 1^2 - 2 = -1$

$f(2) = 2^2 - 2 = 2$

$f(1.5) = 1.5^2 - 2$
$= 2.25 - 2 = 0.25$

$f(1.25) = 1.25^2 - 2$
$= 1.5625 - 2$
$= -0.4375$

$f(1.375) = 1.375^2 - 2$
$= 1.890625 - 2$
$= -0.109375$

Since $(b_3 - a_3)/2 = (1.5 - 1.375)/2 = 0.0625 < 0.1$, taking $p^* = 1.4375$, the midpoint of $[a_3, b_3] = [1.375, 1.5]$, means $|p^* - \sqrt{2}| < 0.0625 < 0.1$.

The bisection method

Implementation In file `bisection_root.m`

```
function r=bisection_root(f,a,b,e)
% Solve f(x)=0 for x in [a,b] up to a tolerance of e.
    assert a<b; assert e>0;
    assert sign(f(a))==-sign(f(b));
    while (b-a)/2 > e,
        c=(a+b)/2;
        if sign(f(c))==sign(f(a)),
            then a=c;
            else b=c;
        endif
    endwhile
    r=(a+b)/2;
endfunction
```

Usage In a separate script file e.g. `sqrt_two.m`

```
f=@(x)x^2-2; a=1; b=2; tol=0.1;
r=bisection_root(f, a, b, tol)
```

The bisection method

Convergence Since the error halves at each step, the method obtains an approximation to within tolerance ϵ in n steps, where $\frac{1}{2}(b-a)/2^n < \epsilon$, or

$$n > \log_2\bigl((b-a)/2\epsilon\bigr) = O(\log_2(1/\epsilon)).$$

Note $\log_2(x) = \ln(x)/\ln(2)$ where $\ln$ is the natural logarithm.

Example To find a root of f in $[1, 2]$ to tolerance $\epsilon = 0.1$, need

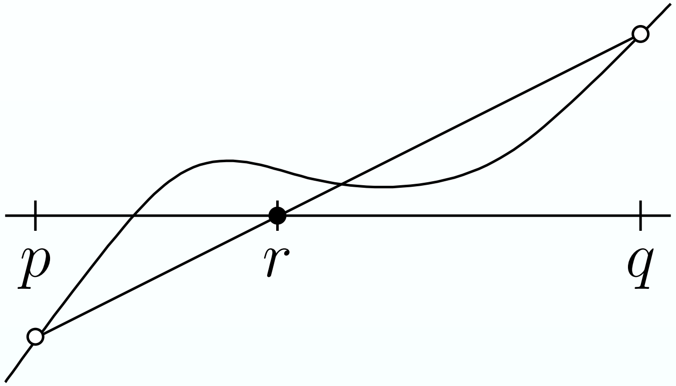
$$n > \log_2\bigl((2-1)/(2\times 0.1)\bigr) = \log_2(5) \approx 2.3,$$

so take $n = 3$ steps.

The secant method

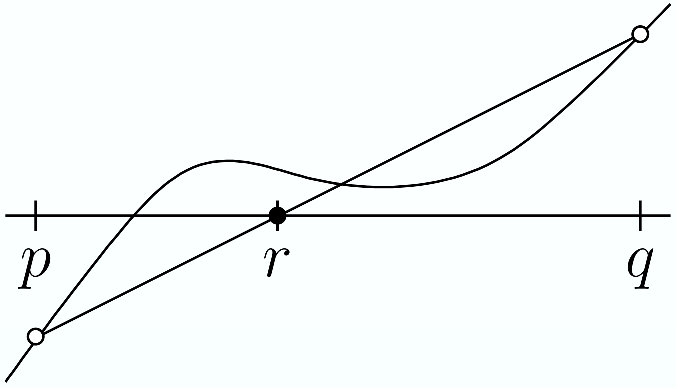
The secant method

Idea Given approximations p, q to a root of f, approximate f by the line joining $(p, f(p))$ and $(q, f(q))$.



The secant method

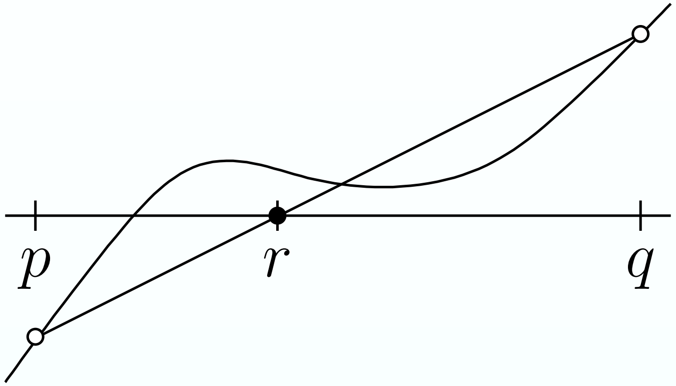
Idea Given approximations p, q to a root of f, approximate f by the line joining $(p, f(p))$ and $(q, f(q))$.



Obtain a better approximation $r = S(f, p, q)$ to the root by finding where this line crosses the x-axis.

The secant method

Idea Given approximations p, q to a root of f, approximate f by the line joining $(p, f(p))$ and $(q, f(q))$.

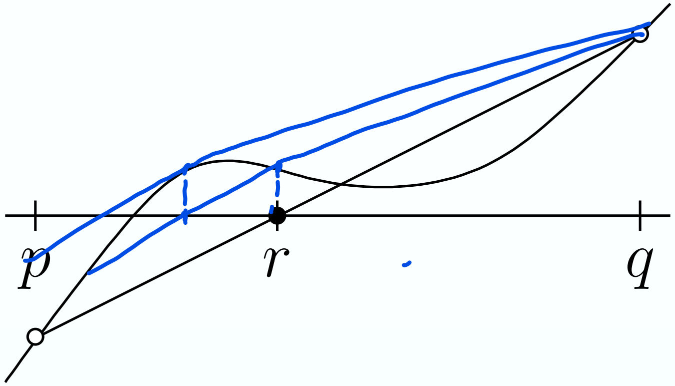


Obtain a better approximation $r = S(f, p, q)$ to the root by finding where this line crosses the x-axis.

Starting from initial points p_0, p_1, iteratively compute $p_2 = S(f, p_0, p_1)$, $p_3 = S(f, p_1, p_2), \ldots$.

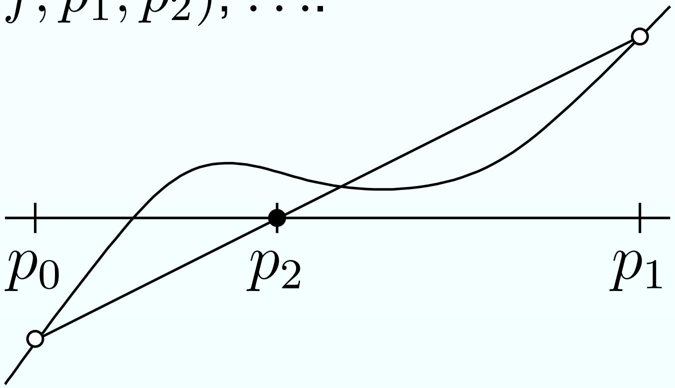
The secant method

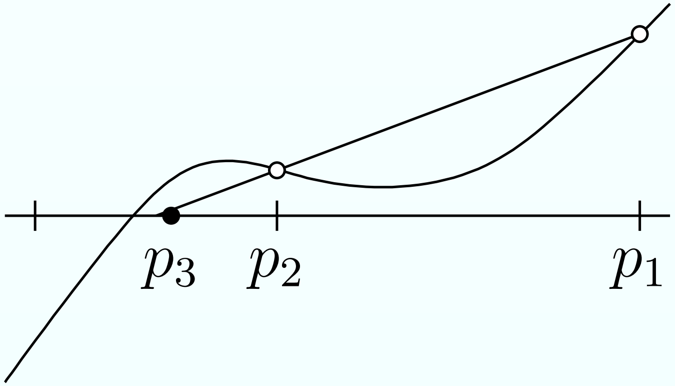
Idea Given approximations p, q to a root of f, approximate f by the line joining $(p, f(p))$ and $(q, f(q))$.



Obtain a better approximation $r = S(f, p, q)$ to the root by finding where this line crosses the x-axis.

Starting from initial points p_0, p_1, iteratively compute $p_2 = S(f, p_0, p_1)$, $p_3 = S(f, p_1, p_2), \ldots$.





The secant method

Derivation

The secant method

Derivation

Line joining $(p, f(p))$ to $(q, f(q))$ has slope $m = \big(f(q) - f(p)\big)/(q - p)$

The secant method

Derivation

Line joining $(p, f(p))$ to $(q, f(q))$ has slope $m = \big(f(q) - f(p)\big)/(q - p)$

Line through $(q, f(q))$ with slope m has equation $y = f(q) + m(x - q)$

The secant method

Derivation

Line joining $(p, f(p))$ to $(q, f(q))$ has slope $m = \bigl(f(q) - f(p)\bigr)/(q - p)$

Line through $(q, f(q))$ with slope m has equation $y = f(q) + m(x - q)$

Setting $y = 0$ and solving for $x = r$ gives

$$f(q) + m(r - q) = 0 \iff r = q - \tfrac{1}{m} f(q).$$

The secant method

Derivation

Line joining $(p, f(p))$ to $(q, f(q))$ has slope $m = \big(f(q) - f(p)\big)/(q - p)$

Line through $(q, f(q))$ with slope m has equation $y = f(q) + m(x - q)$

Setting $y = 0$ and solving for $x = r$ gives

$$f(q) + m(r - q) = 0 \iff r = q - \tfrac{1}{m} f(q).$$

Obtain intercept

$$r = q - \frac{q - p}{f(q) - f(p)} f(q)$$

The secant method

Derivation

Line joining $(p, f(p))$ to $(q, f(q))$ has slope $m = \big(f(q) - f(p)\big)/(q - p)$

Line through $(q, f(q))$ with slope m has equation $y = f(q) + m(x - q)$

Setting $y = 0$ and solving for $x = r$ gives

$$f(q) + m(r - q) = 0 \iff r = q - \tfrac{1}{m} f(q).$$

Obtain intercept

$$r = q - \frac{q - p}{f(q) - f(p)} f(q)$$

Algorithm Apply as an iterative algorithm.

The secant method

Derivation

Line joining $(p, f(p))$ to $(q, f(q))$ has slope $m = \big(f(q) - f(p)\big)/(q - p)$

Line through $(q, f(q))$ with slope m has equation $y = f(q) + m(x - q)$

Setting $y = 0$ and solving for $x = r$ gives

$$f(q) + m(r - q) = 0 \iff r = q - \tfrac{1}{m} f(q).$$

Obtain intercept

$$r = q - \frac{q - p}{f(q) - f(p)} f(q)$$

Algorithm Apply as an iterative algorithm. Start with p_0, p_1,

The secant method

Derivation

Line joining $(p, f(p))$ to $(q, f(q))$ has slope $m = \big(f(q) - f(p)\big)/(q - p)$

Line through $(q, f(q))$ with slope m has equation $y = f(q) + m(x - q)$

Setting $y = 0$ and solving for $x = r$ gives

$$f(q) + m(r - q) = 0 \iff r = q - \tfrac{1}{m} f(q).$$

Obtain intercept

$$r = q - \frac{q - p}{f(q) - f(p)} f(q)$$

Algorithm Apply as an iterative algorithm. Start with p_0, p_1, and set

$$\boxed{p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n).}$$

The secant method

Derivation

Line joining $(p, f(p))$ to $(q, f(q))$ has slope $m = \big(f(q) - f(p)\big)/(q - p)$

Line through $(q, f(q))$ with slope m has equation $y = f(q) + m(x - q)$

Setting $y = 0$ and solving for $x = r$ gives

$$f(q) + m(r - q) = 0 \iff r = q - \tfrac{1}{m} f(q).$$

Obtain intercept

$$r = q - \frac{q - p}{f(q) - f(p)} f(q)$$

Algorithm Apply as an iterative algorithm. Start with p_0, p_1, and set

$$\boxed{p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n).}$$

Bracketing The points p_n, p_{n+1} do *not* need to bracket a root!

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1)$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1)$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) = 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) = 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000$$

$$= 2.000 - \frac{1.000}{3.000} \times 2.000$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) = 2.000 - \frac{2.000 - 1.000}{2.000 - \ (-1.000)} \times 2.000$$

$$= 2.000 - \frac{1.000}{3.000} \times 2.000 = 2.000 - 0.667$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) = 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000$$

$$= 2.000 - \frac{1.000}{3.000} \times 2.000 = 2.000 - 0.667 = 1.333 \ (3\,\text{dp})$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) = 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000$$

$$= 2.000 - \frac{1.000}{3.000} \times 2.000 = 2.000 - 0.667 = 1.333 \ (3\,\text{dp})$$

Second step computes p_3 by taking $n = 2$ in formula.

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) = 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000$$

$$= 2.000 - \frac{1.000}{3.000} \times 2.000 = 2.000 - 0.667 = 1.333 \ (3\,\text{dp})$$

Second step computes p_3 by taking $n = 2$ in formula.

$$p_3 = p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2)$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) = 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000$$

$$= 2.000 - \frac{1.000}{3.000} \times 2.000 = 2.000 - 0.667 = 1.333 \ (3\,\text{dp})$$

Second step computes p_3 by taking $n = 2$ in formula. Need $f(p_2) = -0.222$.

$$p_3 = p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2)$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) = 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000$$

$$= 2.000 - \frac{1.000}{3.000} \times 2.000 = 2.000 - 0.667 = 1.333 \ (3\,\text{dp})$$

Second step computes p_3 by taking $n = 2$ in formula. Need $f(p_2) = -0.222$.

$$p_3 = p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2) \overset{3\,\text{dp}}{=} 1.333 - \frac{1.333 - 2.000}{-0.222 - 2.000} \times (-0.222)$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) = 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000$$

$$= 2.000 - \frac{1.000}{3.000} \times 2.000 = 2.000 - 0.667 = 1.333 \text{ (3 dp)}$$

Second step computes p_3 by taking $n = 2$ in formula. Need $f(p_2) = -0.222$.

$$p_3 = p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2) \overset{3\,\text{dp}}{=} 1.333 - \frac{1.333 - 2.000}{-0.222 - 2.000} \times (-0.222)$$

$$= 1.333 - \frac{-0.667}{-2.222}(-0.222)$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) = 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000$$

$$= 2.000 - \frac{1.000}{3.000} \times 2.000 = 2.000 - 0.667 = 1.333 \ (3\,\text{dp})$$

Second step computes p_3 by taking $n = 2$ in formula. Need $f(p_2) = -0.222$.

$$p_3 = p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2) \overset{3\,\text{dp}}{=} 1.333 - \frac{1.333 - 2.000}{-0.222 - 2.000} \times (-0.222)$$

$$= 1.333 - \frac{-0.667}{-2.222}(-0.222) = 1.333 - (-0.0667)$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

Secant method iterative formula:

$$p_{n+1} = p_n - \frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})} f(p_n)$$

We will work to machine precision, displaying intermediates to 3 decimal places.

Initial step computes p_2 by taking $n = 1$ in general formula.
Need $f(p_0) = f(1) = -1.000$ and $f(p_1) = f(2) = 2.000$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) = 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000$$

$$= 2.000 - \frac{1.000}{3.000} \times 2.000 = 2.000 - 0.667 = 1.333 \ (3\,\text{dp})$$

Second step computes p_3 by taking $n = 2$ in formula. Need $f(p_2) = -0.222$.

$$p_3 = p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2) \overset{3\,\text{dp}}{=} 1.333 - \frac{1.333 - 2.000}{-0.222 - 2.000} \times (-0.222)$$

$$= 1.333 - \frac{-0.667}{-2.222}(-0.222) = 1.333 - (-0.0667) = 1.400.$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

$$f(p_0) = f(1.000) = -1.000$$

$$f(p_1) = f(2.000) = 2.000$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1)$$

$$= 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000 = 1.333$$

$$f(p_0) = f(1.000) = -1.000$$

$$f(p_1) = f(2.000) = 2.000$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

$$
\begin{aligned}
p_2 &= p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) \\
&= 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000 = 1.333
\end{aligned}
$$

$$f(p_0) = f(1.000) = -1.000$$

$$f(p_1) = f(2.000) = 2.000$$

$$
\begin{aligned}
f(p_2) &= f(1.333) = 1.333^2 - 2 \\
&= 1.778 - 2 = -0.222
\end{aligned}
$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1)$$

$$= 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000 = 1.333$$

$$p_3 = p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2)$$

$$= 1.333 - \frac{1.333 - 2.000}{-0.222 - 2.000} \times (-0.222) = 1.400$$

$$f(p_0) = f(1.000) = -1.000$$

$$f(p_1) = f(2.000) = 2.000$$

$$f(p_2) = f(1.333) = 1.333^2 - 2$$

$$= 1.778 - 2 = -0.222$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1)$$
$$= 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000 = 1.333$$

$$p_3 = p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2)$$
$$= 1.333 - \frac{1.333 - 2.000}{-0.222 - 2.000} \times (-0.222) = 1.400$$

$$f(p_0) = f(1.000) = -1.000$$
$$f(p_1) = f(2.000) = 2.000$$
$$f(p_2) = f(1.333) = 1.333^2 - 2$$
$$= 1.778 - 2 = -0.222$$
$$f(p_3) = f(1.400) = 1.400^2 - 2$$
$$= 1.9600 - 2 = -0.0400$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

$$
\begin{aligned}
p_2 &= p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) \\
&= 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000 = 1.333 \\
p_3 &= p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2) \\
&= 1.333 - \frac{1.333 - 2.000}{-0.222 - 2.000} \times (-0.222) = 1.400 \\
p_4 &= 1.4000 - \frac{1.4000 - 1.3333}{-0.0400 - (-0.2222)} \times (-0.0400) \\
&= 1.4000 - (-0.0146) = 1.4146 \ (4\,\text{dp})
\end{aligned}
$$

$$
\begin{aligned}
f(p_0) &= f(1.000) = -1.000 \\
f(p_1) &= f(2.000) = 2.000 \\
f(p_2) &= f(1.333) = 1.333^2 - 2 \\
&= 1.778 - 2 = -0.222 \\
f(p_3) &= f(1.400) = 1.400^2 - 2 \\
&= 1.9600 - 2 = -0.0400
\end{aligned}
$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1)$$
$$= 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000 = 1.333$$

$$p_3 = p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2)$$
$$= 1.333 - \frac{1.333 - 2.000}{-0.222 - 2.000} \times (-0.222) = 1.400$$

$$p_4 = 1.4000 - \frac{1.4000 - 1.3333}{-0.0400 - (-0.2222)} \times (-0.0400)$$
$$= 1.4000 - (-0.0146) = 1.4146 \ (4\,\text{dp})$$

$$f(p_0) = f(1.000) = -1.000$$
$$f(p_1) = f(2.000) = 2.000$$
$$f(p_2) = f(1.333) = 1.333^2 - 2$$
$$= 1.778 - 2 = -0.222$$
$$f(p_3) = f(1.400) = 1.400^2 - 2$$
$$= 1.9600 - 2 = -0.0400$$
$$f(p_4) = f(1.4146) = -0.00119$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

$$p_2 = p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1)$$

$$= 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000 = 1.333$$

$$p_3 = p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2)$$

$$= 1.333 - \frac{1.333 - 2.000}{-0.222 - 2.000} \times (-0.222) = 1.400$$

$$p_4 = 1.4000 - \frac{1.4000 - 1.3333}{-0.0400 - (-0.2222)} \times (-0.0400)$$

$$= 1.4000 - (-0.0146) = 1.4146 \ (4\,\text{dp})$$

$$p_5 = 1.41463 - \frac{1.41463 - 1.40000}{0.00119 - (-0.0400)} \times (0.00119)$$

$$= 1.41463 - 0.00042 = 1.41421 \ (5\,\text{dp})$$

$$f(p_0) = f(1.000) = -1.000$$

$$f(p_1) = f(2.000) = 2.000$$

$$f(p_2) = f(1.333) = 1.333^2 - 2$$
$$= 1.778 - 2 = -0.222$$

$$f(p_3) = f(1.400) = 1.400^2 - 2$$
$$= 1.9600 - 2 = -0.0400$$

$$f(p_4) = f(1.4146) = -0.00119$$

The secant method

Example Solve $f(x) = x^2 - 2 = 0$. Start with $p_0 = 1$, $p_1 = 2$.

$$
\begin{aligned}
p_2 &= p_1 - \frac{p_1 - p_0}{f(p_1) - f(p_0)} f(p_1) \\
&= 2.000 - \frac{2.000 - 1.000}{2.000 - (-1.000)} \times 2.000 = 1.333 \\
p_3 &= p_2 - \frac{p_2 - p_1}{f(p_2) - f(p_1)} f(p_2) \\
&= 1.333 - \frac{1.333 - 2.000}{-0.222 - 2.000} \times (-0.222) = 1.400 \\
p_4 &= 1.4000 - \frac{1.4000 - 1.3333}{-0.0400 - (-0.2222)} \times (-0.0400) \\
&= 1.4000 - (-0.0146) = 1.4146 \ (4\,\text{dp}) \\
p_5 &= 1.41463 - \frac{1.41463 - 1.40000}{0.00119 - (-0.0400)} \times (0.00119) \\
&= 1.41463 - 0.00042 = 1.41421 \ (5\,\text{dp})
\end{aligned}
$$

$$
\begin{aligned}
f(p_0) &= f(1.000) = -1.000 \\
f(p_1) &= f(2.000) = 2.000 \\
f(p_2) &= f(1.333) = 1.333^2 - 2 \\
&= 1.778 - 2 = -0.222 \\
f(p_3) &= f(1.400) = 1.400^2 - 2 \\
&= 1.9600 - 2 = -0.0400 \\
f(p_4) &= f(1.4146) = -0.00119 \\
f(p_5) &= f(1.414211) \\
&= -0.0000060
\end{aligned}
$$

Stopping criteria

Convergence Want to stop when $|p_n - p| < \epsilon$.

Stopping criteria

Convergence Want to stop when $|p_n - p| < \epsilon$.

Problem

Stopping criteria

Convergence Want to stop when $|p_n - p| < \epsilon$.

Problem Don't know exact root p!

Stopping criteria

Convergence Want to stop when $|p_n - p| < \epsilon$.

Problem Don't know exact root p!

If convergence is rapid, expect $|p_n - p| \ll |p_{n-1} - p|$.

Stopping criteria

Convergence Want to stop when $|p_n - p| < \epsilon$.

Problem Don't know exact root p!

If convergence is rapid, expect $|p_n - p| \ll |p_{n-1} - p|$.

If $|p_n - p| = \frac{1}{\gamma}|p_{n-1} - p|$ with $\frac{1}{\gamma} \leq \frac{1}{2}$, find $|p_{n-1} - p_n| \geq |p - p_n|$.

Stopping criteria

Convergence Want to stop when $|p_n - p| < \epsilon$.

Problem Don't know exact root p!

If convergence is rapid, expect $|p_n - p| \ll |p_{n-1} - p|$.

If $|p_n - p| = \frac{1}{\gamma}|p_{n-1} - p|$ with $\frac{1}{\gamma} \leq \frac{1}{2}$, find $|p_{n-1} - p_n| \geq |p - p_n|$.

Practical stopping heuristic $\boxed{\text{Stop when } |p_n - p_{n-1}| < \epsilon.}$

Stopping criteria

Convergence Want to stop when $|p_n - p| < \epsilon$.

Problem Don't know exact root p!

If convergence is rapid, expect $|p_n - p| \ll |p_{n-1} - p|$.

If $|p_n - p| = \frac{1}{\gamma}|p_{n-1} - p|$ with $\frac{1}{\gamma} \leq \frac{1}{2}$, find $|p_{n-1} - p_n| \geq |p - p_n|$.

Practical stopping heuristic $\boxed{\text{Stop when } |p_n - p_{n-1}| < \epsilon.}$

Error estimate For the heuristic $|p_n - p_{n-1}| < \epsilon$, expect $|p_n - p| \lesssim \epsilon$.

Stopping criteria

Convergence Want to stop when $|p_n - p| < \epsilon$.

Problem Don't know exact root p!

If convergence is rapid, expect $|p_n - p| \ll |p_{n-1} - p|$.

If $|p_n - p| = \frac{1}{\gamma}|p_{n-1} - p|$ with $\frac{1}{\gamma} \leq \frac{1}{2}$, find $|p_{n-1} - p_n| \geq |p - p_n|$.

Practical stopping heuristic $\boxed{\text{Stop when } |p_n - p_{n-1}| < \epsilon.}$

Error estimate For the heuristic $|p_n - p_{n-1}| < \epsilon$, expect $|p_n - p| \lesssim \epsilon$.

Error bound If also $f(p_n)$ and $f(p_{n-1})$ have different signs, then $|p_n - p| < \epsilon$.

Stopping criteria

Example Solve $x^2 - 2 = 0$ to an accuracy of 0.01.

Stopping criteria

Example Solve $x^2 - 2 = 0$ to an accuracy of 0.01.

We've already computed (to 4 dp):

$$p_3 = 1.4000,\ p_4 = 1.4146,\ p_5 = 1.4142.$$

Stopping criteria

Example Solve $x^2 - 2 = 0$ to an accuracy of 0.01.

We've already computed (to 4 dp):

$$p_3 = 1.4000,\ p_4 = 1.4146,\ p_5 = 1.4142.$$

Check differences:

Stopping criteria

Example Solve $x^2 - 2 = 0$ to an accuracy of 0.01.

We've already computed (to 4 dp):

$$p_3 = 1.4000,\ p_4 = 1.4146,\ p_5 = 1.4142.$$

Check differences:

$$|p_4 - p_3| = |1.415 - 1.400| = 0.015 > 0.01$$

Stopping criteria

Example Solve $x^2 - 2 = 0$ to an accuracy of 0.01.

We've already computed (to 4 dp):

$$p_3 = 1.4000,\ p_4 = 1.4146,\ p_5 = 1.4142.$$

Check differences:

$$|p_4 - p_3| = |1.415 - 1.400| = 0.015 > 0.01$$

Need another step!

Stopping criteria

Example Solve $x^2 - 2 = 0$ to an accuracy of 0.01.

We've already computed (to 4 dp):

$$p_3 = 1.4000,\ p_4 = 1.4146,\ p_5 = 1.4142.$$

Check differences:

$$|p_4 - p_3| = |1.415 - 1.400| = 0.015 > 0.01$$

Need another step!

$$|p_5 - p_4| = |1.4142 - 1.4146| = 0.0004 < 0.01.$$

Stopping criteria

Example Solve $x^2 - 2 = 0$ to an accuracy of 0.01.

We've already computed (to 4 dp):

$$p_3 = 1.4000,\ p_4 = 1.4146,\ p_5 = 1.4142.$$

Check differences:

$$|p_4 - p_3| = |1.415 - 1.400| = 0.015 > 0.01$$

Need another step!

$$|p_5 - p_4| = |1.4142 - 1.4146| = 0.0004 < 0.01.$$

So we can *expect* $|p_5 - \sqrt{2}| < 0.01$.

Stopping criteria

Example Solve $x^2 - 2 = 0$ to an accuracy of 0.01.

We've already computed (to 4 dp):

$$p_3 = 1.4000,\ p_4 = 1.4146,\ p_5 = 1.4142.$$

Check differences:

$$|p_4 - p_3| = |1.415 - 1.400| = 0.015 > 0.01$$

Need another step!

$$|p_5 - p_4| = |1.4142 - 1.4146| = 0.0004 < 0.01.$$

So we can *expect* $|p_5 - \sqrt{2}| < 0.01$.

Solution $\sqrt{2} \approx p_5 = 1.4142\ (4\,\text{dp}) = 1.41\ (2\,\text{dp})$.

Stopping criteria

Example Solve $x^2 - 2 = 0$ to an accuracy of 0.01.

We've already computed (to 4 dp):

$$p_3 = 1.4000,\ p_4 = 1.4146,\ p_5 = 1.4142.$$

Check differences:

$$|p_4 - p_3| = |1.415 - 1.400| = 0.015 > 0.01$$

Need another step!

$$|p_5 - p_4| = |1.4142 - 1.4146| = 0.0004 < 0.01.$$

So we can *expect* $|p_5 - \sqrt{2}| < 0.01$.

Solution $\sqrt{2} \approx p_5 = 1.4142$ $(4\,\text{dp}) = 1.41$ $(2\,\text{dp})$.

Note: Actual error $|p_5 - \sqrt{2}| = |1.4142 - \sqrt{2}| \approx 2.1 \times 10^{-4} \ll 0.01$.

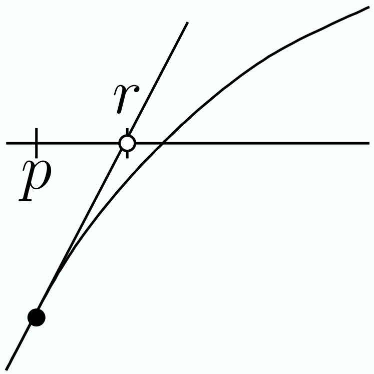
The secant method

Implementation

```
function r=secant(f,p,q,e)
    while abs(q-p) > e,
        r = ... ;
        p = q; q=r;
    endwhile
endfunction
```

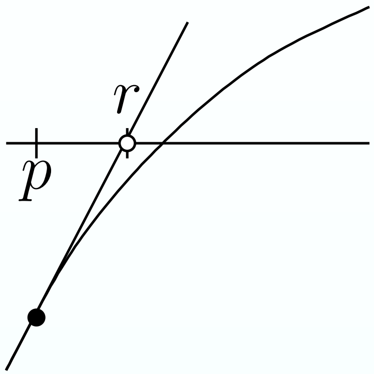
Newton-Raphson method

Idea Instead of using the secant line joining $(p, f(p))$ and $(q, f(q))$, use the tangent line at $(p, f(p))$



Newton-Raphson method

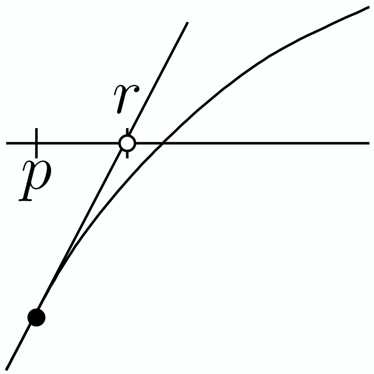
Idea Instead of using the secant line joining $(p, f(p))$ and $(q, f(q))$, use the tangent line at $(p, f(p))$



Tangent line at $(p, f(p))$ has equation $y = f(p) + f'(p)\,(x - p)$.

Newton-Raphson method

Idea Instead of using the secant line joining $(p, f(p))$ and $(q, f(q))$, use the tangent line at $(p, f(p))$

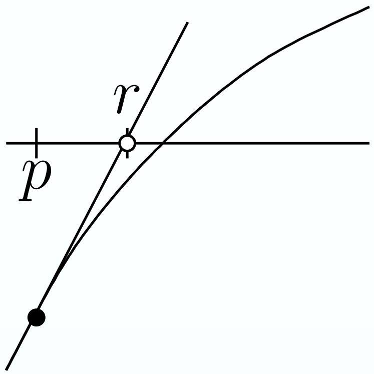


Tangent line at $(p, f(p))$ has equation $y = f(p) + f'(p)\,(x - p)$.

Setting $y = 0$ and solving for $r = x$ gives intercept at $r = p - f(p)/f'(p)$.

Newton-Raphson method

Idea Instead of using the secant line joining $(p, f(p))$ and $(q, f(q))$, use the tangent line at $(p, f(p))$



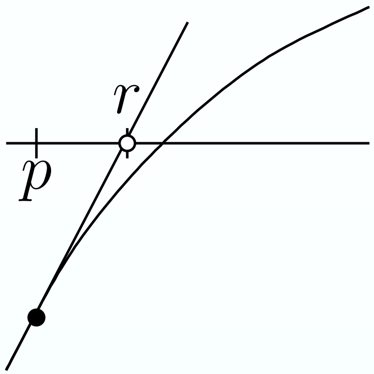
Tangent line at $(p, f(p))$ has equation $y = f(p) + f'(p)(x - p)$.

Setting $y = 0$ and solving for $r = x$ gives intercept at $r = p - f(p)/f'(p)$.

Algorithm

Newton-Raphson method

Idea Instead of using the secant line joining $(p, f(p))$ and $(q, f(q))$, use the tangent line at $(p, f(p))$



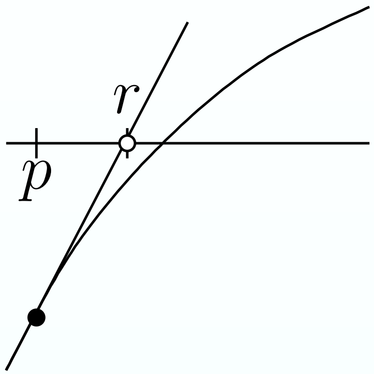
Tangent line at $(p, f(p))$ has equation $y = f(p) + f'(p)(x - p)$.

Setting $y = 0$ and solving for $r = x$ gives intercept at $r = p - f(p)/f'(p)$.

Algorithm Apply iteratively:

Newton-Raphson method

Idea Instead of using the secant line joining $(p, f(p))$ and $(q, f(q))$, use the tangent line at $(p, f(p))$



Tangent line at $(p, f(p))$ has equation $y = f(p) + f'(p)\,(x - p)$.

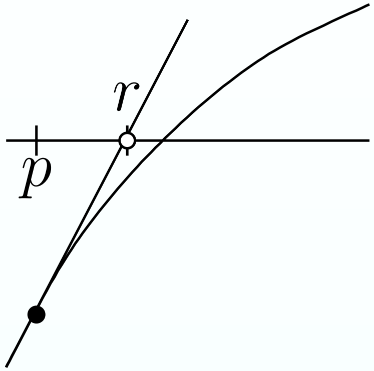
Setting $y = 0$ and solving for $r = x$ gives intercept at $r = p - f(p)/f'(p)$.

Algorithm Apply iteratively:

$$\boxed{p_{n+1} = p_n - \frac{f(p_n)}{f'(p_n)}.}$$

Newton-Raphson method

Idea Instead of using the secant line joining $(p, f(p))$ and $(q, f(q))$, use the tangent line at $(p, f(p))$

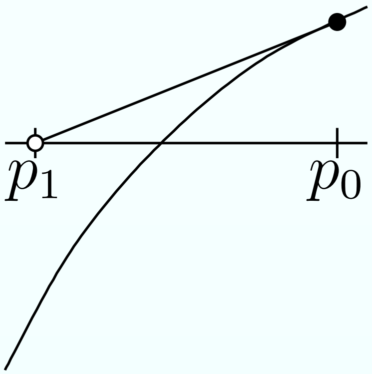


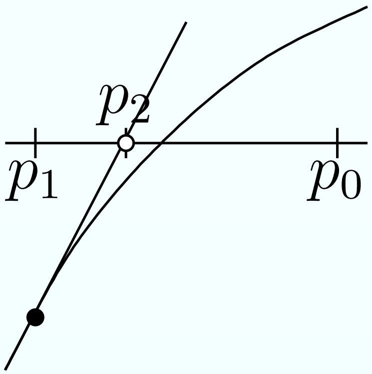
Tangent line at $(p, f(p))$ has equation $y = f(p) + f'(p)\,(x - p)$.

Setting $y = 0$ and solving for $r = x$ gives intercept at $r = p - f(p)/f'(p)$.

Algorithm Apply iteratively:

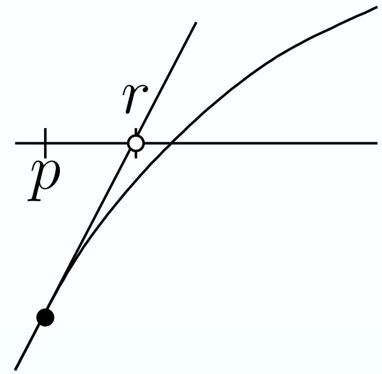
$$\boxed{p_{n+1} = p_n - \frac{f(p_n)}{f'(p_n)}.}$$





Newton-Raphson method

Idea Instead of using the secant line joining $(p, f(p))$ and $(q, f(q))$, use the tangent line at $(p, f(p))$

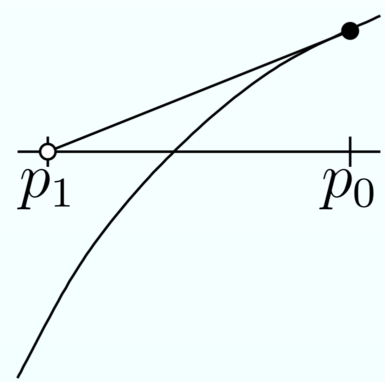


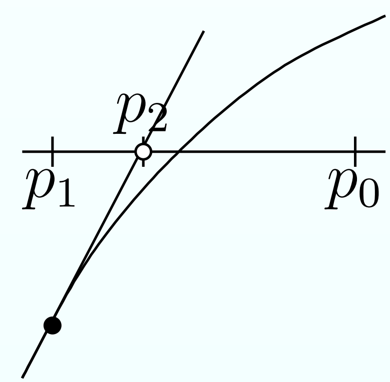
Tangent line at $(p, f(p))$ has equation $y = f(p) + f'(p)(x - p)$.

Setting $y = 0$ and solving for $r = x$ gives intercept at $r = p - f(p)/f'(p)$.

Algorithm Apply iteratively:

$$\boxed{p_{n+1} = p_n - \frac{f(p_n)}{f'(p_n)}.}$$





Stopping heuristic As for the secant method, stop when $|p_n - p_{n-1}| < \epsilon$.

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$p_1 = p_0 - \frac{f(p_0)}{f'(p_0)}$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$p_1 = p_0 - \frac{f(p_0)}{f'(p_0)}$$

$$p_0 = 1.0000$$

$$f(p_0) = 1.0000^2 - 2 = -1.0000;$$

$$f'(p_0) = 2 \times 1.0000 = 2.0000.$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$
\begin{aligned}
p_1 &= p_0 - \frac{f(p_0)}{f'(p_0)} = 1.0000 - \frac{-1.0000}{2.0000} \\
&= 1.5000
\end{aligned}
$$

$$
\begin{aligned}
p_0 &= 1.0000 \\
f(p_0) &= 1.0000^2 - 2 = -1.0000; \\
f'(p_0) &= 2 \times 1.0000 = 2.0000.
\end{aligned}
$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$\begin{aligned} p_1 &= p_0 - \frac{f(p_0)}{f'(p_0)} = 1.0000 - \frac{-1.0000}{2.0000} \\ &= 1.5000 \\ p_2 &= p_1 - \frac{f(p_1)}{f'(p_1)} \end{aligned}$$

$$\begin{aligned} p_0 &= 1.0000 \\ f(p_0) &= 1.0000^2 - 2 = -1.0000; \\ f'(p_0) &= 2 \times 1.0000 = 2.0000. \end{aligned}$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$p_1 = p_0 - \frac{f(p_0)}{f'(p_0)} = 1.0000 - \frac{-1.0000}{2.0000}$$
$$= 1.5000$$
$$p_2 = p_1 - \frac{f(p_1)}{f'(p_1)}$$

$$p_0 = 1.0000$$
$$f(p_0) = 1.0000^2 - 2 = -1.0000;$$
$$f'(p_0) = 2 \times 1.0000 = 2.0000.$$
$$p_1 = 1.5000$$
$$f(p_1) = 1.5000^2 - 2 = 0.2500;$$
$$f'(p_1) = 2 \times 1.5000 = 3.0000.$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$\begin{aligned} p_1 &= p_0 - \frac{f(p_0)}{f'(p_0)} = 1.0000 - \frac{-1.0000}{2.0000} \\ &= 1.5000 \\ p_2 &= p_1 - \frac{f(p_1)}{f'(p_1)} = 1.5000 - \frac{0.2500}{3.0000} \\ &= 1.5000 - 0.0833 = 1.4167. \end{aligned}$$

$$\begin{aligned} p_0 &= 1.0000 \\ f(p_0) &= 1.0000^2 - 2 = -1.0000; \\ f'(p_0) &= 2 \times 1.0000 = 2.0000. \\ p_1 &= 1.5000 \\ f(p_1) &= 1.5000^2 - 2 = 0.2500; \\ f'(p_1) &= 2 \times 1.5000 = 3.0000. \end{aligned}$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$p_1 = p_0 - \frac{f(p_0)}{f'(p_0)} = 1.0000 - \frac{-1.0000}{2.0000}$$
$$= 1.5000$$

$$p_2 = p_1 - \frac{f(p_1)}{f'(p_1)} = 1.5000 - \frac{0.2500}{3.0000}$$
$$= 1.5000 - 0.0833 = 1.4167.$$

Error estimate

$$e_2 := |p_2 - p| \lesssim |p_2 - p_1| = 0.083 > 0.01$$

$$p_0 = 1.0000$$
$$f(p_0) = 1.0000^2 - 2 = -1.0000;$$
$$f'(p_0) = 2 \times 1.0000 = 2.0000.$$

$$p_1 = 1.5000$$
$$f(p_1) = 1.5000^2 - 2 = 0.2500;$$
$$f'(p_1) = 2 \times 1.5000 = 3.0000.$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$p_1 = p_0 - \frac{f(p_0)}{f'(p_0)} = 1.0000 - \frac{-1.0000}{2.0000}$$
$$= 1.5000$$
$$p_2 = p_1 - \frac{f(p_1)}{f'(p_1)} = 1.5000 - \frac{0.2500}{3.0000}$$
$$= 1.5000 - 0.0833 = 1.4167.$$

Error estimate

$$e_2 := |p_2 - p| \lesssim |p_2 - p_1| = 0.083 > 0.01$$

Need another step!

$$\begin{aligned} p_0 &= 1.0000 \\ f(p_0) &= 1.0000^2 - 2 = -1.0000; \\ f'(p_0) &= 2 \times 1.0000 = 2.0000. \end{aligned}$$

$$\begin{aligned} p_1 &= 1.5000 \\ f(p_1) &= 1.5000^2 - 2 = 0.2500; \\ f'(p_1) &= 2 \times 1.5000 = 3.0000. \end{aligned}$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$p_1 = p_0 - \frac{f(p_0)}{f'(p_0)} = 1.0000 - \frac{-1.0000}{2.0000}$$
$$= 1.5000$$

$$p_2 = p_1 - \frac{f(p_1)}{f'(p_1)} = 1.5000 - \frac{0.2500}{3.0000}$$
$$= 1.5000 - 0.0833 = 1.4167.$$

Error estimate

$$e_2 := |p_2 - p| \lesssim |p_2 - p_1| = 0.083 > 0.01$$

Need another step!

$$p_3 = p_2 - \frac{f(p_2)}{f'(p_2)}$$

$$p_0 = 1.0000$$
$$f(p_0) = 1.0000^2 - 2 = -1.0000;$$
$$f'(p_0) = 2 \times 1.0000 = 2.0000.$$

$$p_1 = 1.5000$$
$$f(p_1) = 1.5000^2 - 2 = 0.2500;$$
$$f'(p_1) = 2 \times 1.5000 = 3.0000.$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$\begin{aligned} p_1 &= p_0 - \frac{f(p_0)}{f'(p_0)} = 1.0000 - \frac{-1.0000}{2.0000} \\ &= 1.5000 \\ p_2 &= p_1 - \frac{f(p_1)}{f'(p_1)} = 1.5000 - \frac{0.2500}{3.0000} \\ &= 1.5000 - 0.0833 = 1.4167. \end{aligned}$$

Error estimate

$$e_2 := |p_2 - p| \lesssim |p_2 - p_1| = 0.083 > 0.01$$

Need another step!

$$p_3 = p_2 - \frac{f(p_2)}{f'(p_2)}$$

$$\begin{aligned} p_0 &= 1.0000 \\ f(p_0) &= 1.0000^2 - 2 = -1.0000; \\ f'(p_0) &= 2 \times 1.0000 = 2.0000. \\ p_1 &= 1.5000 \\ f(p_1) &= 1.5000^2 - 2 = 0.2500; \\ f'(p_1) &= 2 \times 1.5000 = 3.0000. \\ p_2 &= 1.4167 \\ f(p_2) &= 1.4167^2 - 2 \\ &= 2.0069 - 2 = 0.0069; \\ f'(p_2) &= 2 \times 1.4167 = 2.8333. \end{aligned}$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$\begin{aligned} p_1 &= p_0 - \frac{f(p_0)}{f'(p_0)} = 1.0000 - \frac{-1.0000}{2.0000} \\ &= 1.5000 \\ p_2 &= p_1 - \frac{f(p_1)}{f'(p_1)} = 1.5000 - \frac{0.2500}{3.0000} \\ &= 1.5000 - 0.0833 = 1.4167. \end{aligned}$$

Error estimate

$$e_2 := |p_2 - p| \lesssim |p_2 - p_1| = 0.083 > 0.01$$

Need another step!

$$\begin{aligned} p_3 &= p_2 - \frac{f(p_2)}{f'(p_2)} = 1.4167 - \frac{0.0069}{2.8333} \\ &= 1.4167 - 0.0025 = 1.4142. \end{aligned}$$

$$\begin{aligned} p_0 &= 1.0000 \\ f(p_0) &= 1.0000^2 - 2 = -1.0000; \\ f'(p_0) &= 2 \times 1.0000 = 2.0000. \\ p_1 &= 1.5000 \\ f(p_1) &= 1.5000^2 - 2 = 0.2500; \\ f'(p_1) &= 2 \times 1.5000 = 3.0000. \\ p_2 &= 1.4167 \\ f(p_2) &= 1.4167^2 - 2 \\ &= 2.0069 - 2 = 0.0069; \\ f'(p_2) &= 2 \times 1.4167 = 2.8333. \end{aligned}$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$
\begin{aligned}
p_1 &= p_0 - \frac{f(p_0)}{f'(p_0)} = 1.0000 - \frac{-1.0000}{2.0000} \\
&= 1.5000 \\
p_2 &= p_1 - \frac{f(p_1)}{f'(p_1)} = 1.5000 - \frac{0.2500}{3.0000} \\
&= 1.5000 - 0.0833 = 1.4167.
\end{aligned}
$$

Error estimate

$$e_2 := |p_2 - p| \lesssim |p_2 - p_1| = 0.083 > 0.01$$

Need another step!

$$
\begin{aligned}
p_3 &= p_2 - \frac{f(p_2)}{f'(p_2)} = 1.4167 - \frac{0.0069}{2.8333} \\
&= 1.4167 - 0.0025 = 1.4142.
\end{aligned}
$$

Error estimate $e_3 \lesssim |p_3 - p_2| = 0.0025 < 0.01$

$$
\begin{aligned}
p_0 &= 1.0000 \\
f(p_0) &= 1.0000^2 - 2 = -1.0000; \\
f'(p_0) &= 2 \times 1.0000 = 2.0000. \\
p_1 &= 1.5000 \\
f(p_1) &= 1.5000^2 - 2 = 0.2500; \\
f'(p_1) &= 2 \times 1.5000 = 3.0000. \\
p_2 &= 1.4167 \\
f(p_2) &= 1.4167^2 - 2 \\
&= 2.0069 - 2 = 0.0069; \\
f'(p_2) &= 2 \times 1.4167 = 2.8333.
\end{aligned}
$$

Newton-Raphson method

Example Solve $f(x) = x^2 - 2 = 0$ to an accuracy of 0.01.

Derivative $f'(x) = 2x$. Start with $p_0 = 1$. Work to 4 decimal places.

$$
\begin{aligned}
p_1 &= p_0 - \frac{f(p_0)}{f'(p_0)} = 1.0000 - \frac{-1.0000}{2.0000} \\
&= 1.5000 \\
p_2 &= p_1 - \frac{f(p_1)}{f'(p_1)} = 1.5000 - \frac{0.2500}{3.0000} \\
&= 1.5000 - 0.0833 = 1.4167.
\end{aligned}
$$

Error estimate

$$e_2 := |p_2 - p| \lesssim |p_2 - p_1| = 0.083 > 0.01$$

Need another step!

$$
\begin{aligned}
p_3 &= p_2 - \frac{f(p_2)}{f'(p_2)} = 1.4167 - \frac{0.0069}{2.8333} \\
&= 1.4167 - 0.0025 = 1.4142.
\end{aligned}
$$

Error estimate $e_3 \lesssim |p_3 - p_2| = 0.0025 < 0.01$

$$
\begin{aligned}
p_0 &= 1.0000 \\
f(p_0) &= 1.0000^2 - 2 = -1.0000; \\
f'(p_0) &= 2 \times 1.0000 = 2.0000. \\
p_1 &= 1.5000 \\
f(p_1) &= 1.5000^2 - 2 = 0.2500; \\
f'(p_1) &= 2 \times 1.5000 = 3.0000. \\
p_2 &= 1.4167 \\
f(p_2) &= 1.4167^2 - 2 \\
&= 2.0069 - 2 = 0.0069; \\
f'(p_2) &= 2 \times 1.4167 = 2.8333.
\end{aligned}
$$

Solution $\sqrt{2} \approx p_3 = 1.4142\,(4\,\text{dp}) = 1.41\,(2\,\text{dp})$.

Rounding effects

Rounding effects There is usually a small difference between rounded and exact computation.

For $p_1 = 1.5$, the *exact* value of $p_2 = \frac{17}{12} = 1\frac{5}{12} = 1.41\dot{6} = 1.4167\,(4\,\text{dp})$

Using exact arithmetic, find $p_3 = \frac{577}{408} = 1\frac{169}{408} = 1.41421568\cdots$.

Taking $p_2 = 1.4167$, using rounded arithmetic to 4 decimal places:

$$f(p_2) = f(1.4167) = 1.4167^2 - 2 \overset{4\,\text{dp}}{=} 2.0070 - 2 = 0.0070.$$
$$f'(p_2) = f'(1.4167) = 2 \times 1.4167 - 2.8334.$$

$$p_3 = p_2 - \frac{f(p_2)}{f'(p_2)} = 1.4167 - \frac{0.0070}{2.8334} \overset{4\,\text{dp}}{=} 1.4167 - 0.0025 = 1.4142.$$

In this case, rounding the exact value of p_3 gives the value computed using rounded arithmetic!

This is fairly common in iterative methods:

In iterative methods, rounding errors in early steps can be compensated for by using higher precision in later steps!

Newton-Raphson method

Convergence analysis Let p_* be the root.

Newton-Raphson method

Convergence analysis Let p_* be the root. Then by Taylor's theorem,

$$0 = f(p_*) = f(p_n) + f'(p_n)(p_* - p_n) + \frac{1}{2}f''(\xi)(p_* - p_n)^2,$$

Newton-Raphson method

Convergence analysis Let p_* be the root. Then by Taylor's theorem,

$$0 = f(p_*) = f(p_n) + f'(p_n)(p_* - p_n) + \frac{1}{2}f''(\xi)(p_* - p_n)^2,$$

so

$$p_{n+1} = p_n - f(p_n)/f'(p_n) = p_* + \big(f''(\xi)/2f'(p_n)\big)(p_* - p_n)^2.$$

Newton-Raphson method

Convergence analysis Let p_* be the root. Then by Taylor's theorem,

$$0 = f(p_*) = f(p_n) + f'(p_n)(p_* - p_n) + \tfrac{1}{2} f''(\xi)(p_* - p_n)^2,$$

so

$$p_{n+1} = p_n - f(p_n)/f'(p_n) = p_* + \big(f''(\xi)/2f'(p_n)\big)(p_* - p_n)^2.$$

Setting error $\epsilon_n = p_n - p_*$ gives

$$\epsilon_{n+1} = \frac{f''(\xi)}{2f'(p_n)}\epsilon_n^2 \approx \frac{f''(p_*)}{2f'(p_*)}\epsilon_n^2 = C\epsilon_n^2.$$

Newton-Raphson method

Convergence analysis Let p_* be the root. Then by Taylor's theorem,

$$0 = f(p_*) = f(p_n) + f'(p_n)(p_* - p_n) + \tfrac{1}{2} f''(\xi)(p_* - p_n)^2,$$

so

$$p_{n+1} = p_n - f(p_n)/f'(p_n) = p_* + \big(f''(\xi)/2f'(p_n)\big)(p_* - p_n)^2.$$

Setting error $\epsilon_n = p_n - p_*$ gives

$$\epsilon_{n+1} = \frac{f''(\xi)}{2f'(p_n)}\epsilon_n^2 \approx \frac{f''(p_*)}{2f'(p_*)}\epsilon_n^2 = C\epsilon_n^2.$$

Error decays quadratically; *very* fast.

Comparison of methods

Reliability

+ The bisection method always works.
– The Newton-Raphson method and the secant method may cycle or diverge.

Requirements

+ The bisection and secant methods only require function values.
– The Newton-Raphson method requires the derivative of the function.

Efficiency

– The bisection method converges only linearly, $\epsilon_{n+1} \sim \frac{1}{2}\epsilon_n$
+ The Newton-Raphson method converges superlinearly at rate $\epsilon_{n+1} \sim C\epsilon_n^2$, the secant method $\epsilon_{n+1} \sim C\epsilon_n^{1.6}$.
– Per evaluation of f or f', the Newton-Raphson method is only $O(\epsilon^{1.4})$, slower than the secant method $O(\epsilon^{1.6})$.

Parametrised equations (Non-examinable)

Problem Solve $f(x, y) = 0$ for y in terms of x at points $(x_0, \ldots, x_n)$.

Equivalently, solve $f_a(x) = 0$ for x in terms of the parameter a.

Solution

1. Solve $f(x_0, y) = 0$ using the Newton-Raphson method (or the secant method) with arbitrary starting y to find y_0.

2. Successively solve $f(x_i, y) = 0$ to find y_i, using the solution y_{i-1} for x_{i-1} to *hot-start* the method.

Parametrised equations (Non-examinable)

Solve $f(x, y) = \cos(x) - x + e^x y + y^3 = 0$ for y in terms of x.

Implementation

```
f=@(x,y)cos(x)-x+exp(x)*y+y*y*y;
dyf=@(x,y)exp(x)+3*y*y;
xmin=-4; xmax=+6;
h=0.1; tol=1e-8;
N=round((xmax-xmin)/h);
xs=linspace(xmin,xmax,N+1); ys=xs*NaN;
y=0;
for i=0:N,
    x=xs(i+1); yp=-inf;
    while abs(y-yp)>tol,
        yn=y-f(x,y)/dyf(x,y);
        yp=y; y=yn;
    end;
    ys(i+1)=y;
end;
plot(xs,ys)
```

Systems of equations (Non-examinable)

Systems of nonlinear equations Find a root of $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$.

Newton-Raphson method Generalises directly:

$$\mathbf{p}_{n+1} = \mathbf{p}_n - \mathrm{D}\mathbf{f}(\mathbf{p}_n)^{-1}\mathbf{f}(\mathbf{p}_n).$$

Secant method Generalises to the *simplex method*.

Brent's method (Non-examinable)

Problem The secant method and the Newton-Raphson method do not always converge!

Description Aim to keep *bracketing* properties of the bisection method with the fast convergence of the secant method.

Idea If a secant step does not sufficiently reduce the size of the bracketing interval, use bisection.

Efficiency Don't allow successive bisections.