

Numerical methods for finding the roots of a function

The roots of a function $f(x)$ are defined as the values for which the value of the function becomes equal to zero. So, finding the roots of $f(x)$ means solving the equation

$$f(x) = 0.$$

Example 1: If $f(x) = ax^2 + bx + c$ is a quadratic polynomial, the roots are given by the well-known formula

$$x_1, x_2 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Example 2: For a polynomial of degree 3 or higher, it is sometimes (but not very often!) possible to find the roots by factorising the polynomial

$$f(x) = x^3 - 6x^2 + 11x - 6 = (x - 1)(x - 2)(x - 3) \text{ so the roots are 1, 2 and 3}$$

$$f(x) = x^4 - 16 = (x^2 - 4)(x^2 + 4) \quad \text{so the roots are 2 and -2}$$

For a large number of problems it is, however, not possible to find exact values for the roots of the function so we have to find approximations instead.

1. The bisection method

The bisection method is based on the following result from calculus:

The Intermediate Value Theorem:

Assume $f : \mathbb{R} \rightarrow \mathbb{R}$ is a continuous function and there are two real numbers a and b such that $f(a)f(b) < 0$. Then $f(x)$ has at least one zero between a and b .

In other words, if a continuous function has different signs at two points, it has to go through zero somewhere in between!

The bisection method consists of finding two such numbers a and b , then halving the interval $[a, b]$ and keeping the half on which $f(x)$ changes sign and repeating the procedure until this interval shrinks to give the required accuracy for the root.

Bisection Algorithm

An algorithm could be defined as follows. Suppose we need a root for $f(x) = 0$ and we have an error tolerance of ϵ (the absolute error in calculating the root must be less than ϵ).

Bisection Algorithm:

Step 1: Find two numbers a and b at which f has different signs.

Step 2: Define $c = \frac{a+b}{2}$.

Step 3: If $b - c \leq \epsilon$ then accept c as the root and stop.

Step 4: If $f(a)f(c) \leq 0$ then set c as the new b . Otherwise, set c as the new a . Return to step 1.

Example

Exercise 1: Find a root of the equation

$$x^6 - x - 1 = 0$$

accurate to within $\varepsilon = 0.001$.

(Note: This equation has two real roots: -0.7780895987 and 1.134724138 .)

First we have to find an interval where the function $f(x) = x^6 - x - 1$ changes sign. Easy to see that $f(1) \cdot f(2) < 0$ so we take $a = 1$ and $b = 2$. Then $c = (a + b)/2 = 1.5$ and the bisection algorithm is detailed in the following table.

n	a	b	c	b-c	f(c)
1	1.0000	2.0000	1.5000	0.5000	8.8906
2	1.000	1.5000	1.2500	0.2500	1.5647
3	1.0000	1.2500	1.1250	0.1250	-0.0977
4	1.1250	1.2500	1.1875	0.0625	0.6167
5	1.1250	1.1875	1.1562	0.0312	0.2333
6	1.1250	1.1562	1.1406	0.0156	0.0616
7	1.1250	1.1406	1.1328	0.0078	-0.0196
8	1.1328	1.1406	1.1367	0.0039	0.0206
9	1.1328	1.1367	1.1348	0.0020	0.0004
10	1.1328	1.1348	1.1338	0.00098	-0.0096

Note that after 10 steps we have $b - c = 0.00098 < 0.001$ hence the required root approximation is $c = 1.1338$.

Error bounds

Let α be the value of the root, $a \leq \alpha \leq b$. Let a_n , b_n and c_n be the values of a , b and c on the n th iteration of the algorithm.

Then the error bound for c_n is given by

$$|\alpha - c_n| \leq \frac{1}{2^n} (b - a)$$

This inequality can give us the number of iterations needed for a required accuracy ε

$$n \geq \frac{\log\left(\frac{b-a}{\varepsilon}\right)}{\log 2}$$

Advantages and disadvantages of the bisection method

- 1 The method is guaranteed to converge
- 2 The error bound decreases by half with each iteration
- 3 The bisection method converges very slowly
- 4 The bisection method cannot detect multiple roots

Exercise 2: Consider the nonlinear equation $e^x - x - 2 = 0$.

- 1 Show there is a root α in the interval $(1, 2)$.
- 2 Estimate how many iterations will be needed in order to approximate this root with an accuracy of $\varepsilon = 0.001$ using the bisection method.
- 3 Then approximate α with an accuracy of $\varepsilon = 0.001$ using the bisection method.

Exercise 3: Find a root of $f(x) = x^3 + 2x^2 - 3x - 1$ ($\varepsilon = 0.001$).

2. The Newton-Raphson method

Background

Recall that the equation of a straight line is given by the equation

$$y = mx + n \quad (1)$$

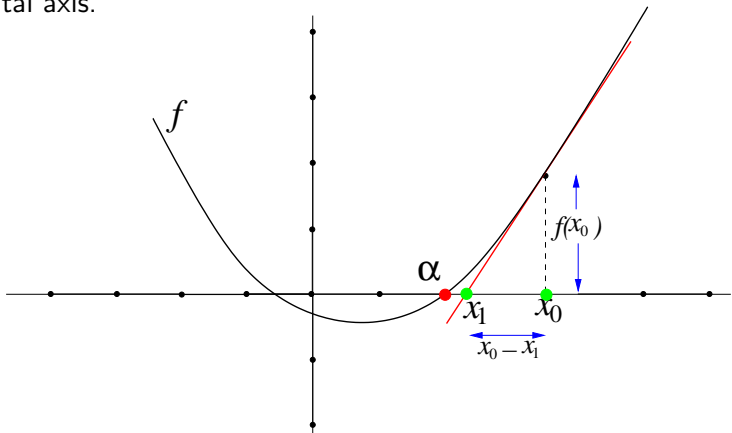
where m is called the *slope* of the line. (This means that all points (x, y) on the line satisfy the equation above.)

If we know the slope m and one point (x_0, y_0) on the line, equation (1) becomes

$$y - y_0 = m(x - x_0) \quad (2)$$

Idea behind Newton's method

Assume we need to find a root of the equation $f(x) = 0$. Consider the graph of the function $f(x)$ and an initial estimate of the root, x_0 . To improve this estimate, take the tangent to the graph of $f(x)$ through the point $(x_0, f(x_0))$ and let x_1 be the point where this line crosses the horizontal axis.



According to eq. (2) above, this point is given by

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

where $f'(x_0)$ is the derivative of f at x_0 . Then take x_1 as the next approximation and continue the procedure. The general iteration will be given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

and so on.

Exercise 1: Find a root of the equation

$$x^6 - x - 1 = 0$$

(Note that the true root is $\alpha = 1.134724138$.)

We already know that this equation has a root between 1 and 2 so we take $x_0 = 1.5$ as our first approximation.

For each iteration we calculate the value $x_n - x_{n-1}$ which, as we shall see later, is a good approximation to the absolute error $\alpha - x_{n-1}$.

The iterations for Newton's algorithm are shown in the table below

n	x_n	$f(x_n)$	$x_n - x_{n-1}$	$\alpha - x_{n-1}$
0	1.5	8.890625		
1	1.30049	2.54	-2.0E-1	-3.65E-1
2	1.18148	5.38E-1	-1.19E-1	-1.66E-1
3	1.13945	4.92E-2	-4.20E-2	-4.68E-2
4	1.13477	5.50E-4	-4.68E-3	-4.73E-3
5	1.13472	7.11E-8	-5.35E-5	-5.35E-5
6	1.13472	1.55E-15	-6.91E-9	-6.91E-9

Error analysis

Let α be the root of $f(x) = 0$ we are trying to approximate. Then, Taylor's formula gives

$$f(\alpha) = f(x_n) + (\alpha - x_n)f'(x_n) + \frac{1}{2}(\alpha - x_n)^2 f''(c_n)$$

where c_n is an unknown point between α and x_n . This eventually leads to

$$\alpha - x_{n+1} = (\alpha - x_n)^2 \left[\frac{-f''(c_n)}{2f'(x_n)} \right]$$

which means that the error in x_{n+1} is proportional to the square of the error in x_n .

Hence, if the initial estimate was good enough, the error is expected to decrease very fast and the algorithm converges to the root.

To give an actual error estimation, write Taylor's formula again

$$f(\alpha) = f(x_n) + (\alpha - x_n)f'(c_n)$$

where c_n is an unknown point between α and x_n , so

$$\alpha - x_n = \frac{-f(x_n)}{f'(c_n)} \approx \frac{-f(x_n)}{f'(x_n)}$$

so

$$\alpha - x_n \approx x_{n+1} - x_n$$

which is usually quite accurate.

Advantages and disadvantages of Newton's method:

- ➊ The error decreases rapidly with each iteration
- ➋ Newton's method is very fast. (Compare with bisection method!)
- ➌ Unfortunately, for bad choices of x_0 (the initial guess) the method can fail to converge! Therefore the choice of x_0 is VERY IMPORTANT!
- ➍ Each iteration of Newton's method requires two function evaluations, while the bisection method requires only one.

Note: A good strategy for avoiding failure to converge would be to use the bisection method for a few steps (to give an initial estimate and make sure the sequence of guesses is going in the right direction) followed by Newton's method, which should converge very fast at this point.

Exercise 2: Find a root of $f(x) = e^x - 3x$.

Exercise 3: Find a root of $f(x) = x^3 + 2x^2 - 3x - 1$.

Example

Consider the function $f(x) = \tan(\pi x) - x - 6$. Use the Newton method to evaluate a root of this function, with the following initial approximations: 0.48, 0.4 and 0.01.

0	$x_0 = 0.48$	$x_0 = 0.4$	$x_0 = 0.01$
1	0.46817005	0.50415051	2.7976098
2	0.45703550	0.50865280	5.2724799
3	0.4517795	0.51883428	6.8756520
4	0.45105821	0.54490990	11.844078
5	0.45104726	0.63096380	17.899771
6	0.45104726	1.1095932	27.688306
7	0.45104726	3.7616925	31.553226
8	0.45104726	5.9643775	31.940006
9	0.45104726	11.500952	48.842772
10	0.45104726	11.501954	66.985902
11	0.45104726	11.504118	100.98883

3. The secant method

Idea behind the secant method

Assume we need to find a root of the equation $f(x) = 0$, called α . Consider the graph of the function $f(x)$ and two initial estimates of the root, x_0 and x_1 . The two points $(x_0, f(x_0))$ and $(x_1, f(x_1))$ on the graph of $f(x)$ determine a straight line, called a secant line which can be viewed as an approximation to the graph. The point x_2 where this secant line crosses the x axis is then an approximation for the root α .

This is the same idea as in Newton's method, where the tangent line has been approximated by a secant line.

The equation of the secant line will be given by

$$y = f(x_1) + (x - x_1) \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

so

$$x_2 = x_1 - f(x_1) \frac{x_1 - x_0}{f(x_1) - f(x_0)}$$

Then take x_1 and x_2 as the next two estimates and continue the procedure. The general iteration will be given by

$$x_{n+1} = x_n - f(x_n) \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}$$

and so on.

Exercise 1: Find a root of the equation

$$x^6 - x - 1 = 0$$

(Recall that the true root is $\alpha = 1.134724138$.)

Note: We can use a similar argument as in Newton's method to show that

$$\alpha - x_{n-1} \approx x_n - x_{n-1}$$

so that we have a measure of the absolute error at each step.

n	x_n	$f(x_n)$	$x_n - x_{n-1}$	$\alpha - x_{n-1}$
0	2.0	61.0		
1	1.0	-1.0	-1.0	
2	1.016129	-9.15E-1	1.61E-2	1.35E-1
3	1.190578	6.57E-1	1.74E-1	1.19E-1
4	1.117656	-1.68E-1	-7.29E-2	-5.59E-2
5	1.132532	-2.24E-2	1.49E-2	1.71E-2
6	1.134817	9.54E-4	2.29E-3	2.19E-3
7	1.134724	-5.07E-6	-9.32E-5	-9.27E-5
8	1.134724	-1.13E-9	4.92E-7	4.92E-7

Advantages and disadvantages:

- ① The error decreases slowly at first but then rapidly after a few iterations.
- ② The secant method is slower than Newton's method but faster than the bisection method.
- ③ Each iteration of Newton's method requires two function evaluations, while the secant method requires only one
- ④ The secant method does not require differentiation.

4. The fixed-point iteration method

Definition: Consider a real function $f(x)$. A number a is called a *fixed point* of the function if it satisfies

$$f(a) = a.$$

Suppose we are trying to solve the equation $f(x) = 0$. We can rearrange this equation as $g(x) = x$ so now we are looking for a fixed point for the function g .

We start with an initial guess x_0 and compute a sequence of successive approximations using the formula

$$x_{n+1} = g(x_n)$$

If the sequence x_n converges, then it will converge to the fixed point of $g(x)$ so we have found the root of $f(x) = 0$.

Example: Consider the equation $f(x) = x^2 - 3x + 1 = 0$ (whose true roots are $\alpha_1 = 0.381966$ and $\alpha_2 = 2.618034$). This can be rearranged as a fixed point problem in many different ways. Compare the following two algorithms.

$$\text{I.} \quad x_{n+1} = \frac{1}{3}(x_n^2 + 1) \equiv g_1(x_n)$$

$$\text{II.} \quad x_{n+1} = 3 - \frac{1}{x_n} \equiv g_2(x_n)$$

n	$x_n = g_1(x_n)$	$x_n = g_2(x_n)$
0	1.0	1.0
1	0.666666	2.0
2	0.481481	2.50
3	0.410608	2.60
4	0.389533	2.615385
5	0.383911	2.617647
6	0.382463	2.617977
7	0.382092	2.618026
8	0.381998	2.618033
9	0.381974	2.618034
10	0.381968	2.618034

Note that the two algorithms converge to different roots even though the starting point is the same!

Convergence condition for the fixed point iteration scheme

Consider the equation $f(x) = 0$, which has the root α and can be written as the fixed point problem $g(x) = x$. If the following conditions hold

- 1 $g(x)$ and $g'(x)$ are continuous functions;
- 2 $|g'(\alpha)| < 1$

then the fixed point iteration scheme based on the function g will converge to α . Alternatively, if $|g'(\alpha)| > 1$ then the iteration will not converge to α . (Note that when $|g'(\alpha)| = 1$ no conclusion can be reached.)

For the previous example, we have

$$g_1(x) = \frac{1}{3}(x^2 + 1) \quad \text{so} \quad g_1'(x) = \frac{2x}{3}$$

Evaluating the derivative at the two roots (or fixed points):

$$|g_1'(\alpha_1)| = 0.254... < 1 \quad \text{and} \quad |g_1'(\alpha_2)| = 1.745... > 1$$

so the first algorithm converges to $\alpha_1 = 0.3819...$ but not to $\alpha_2 = 2.618...$

The second algorithm is given by

$$g_2(x) = 3 - \frac{1}{x} \quad \text{so} \quad g_2'(x) = \frac{1}{x^2}$$

which gives

$$|g_2'(\alpha_1)| = 6.92... > 1 \quad \text{and} \quad |g_2'(\alpha_2)| = 0.13... < 1$$

so the second algorithm converges to $\alpha_2 = 2.61...$ but not to $\alpha_1 = 0.38...$

Exercise

Consider the nonlinear equation

$$f(x) = x^3 - 2x^2 - 3 = 0$$

which has a root α between 2 and 3. (The true value is $\alpha = 2.485583998$.)

- 1 Rewrite the equation $f(x) = 0$ as the fixed-point problem $g_1(x) = x$ where

$$g_1(x) = 2 + \frac{3}{x^2}$$

and, using the convergence criterion, show that the iteration algorithm associated with this problem converges to the root α .

- 2 Create two other fixed-point iteration schemes, $g_2(x)$ and $g_3(x)$.
- 3 Perform ten iterations with six exact digits for each of the schemes $g_1(x)$, $g_2(x)$ and $g_3(x)$ and compare the approximations. (Use the same starting point for all algorithms.)