Secant Method

NC 11 • Module 2: Solution of nonlinear equations

Central question	How can two function values replace a derivative in root finding?
Lecture focus	Secant geometry, formula, numerical tables and convergence.
Prerequisite	Root isolation and Newton-Raphson method (NC 06–10).

Learning outcomes

By the end of this lecture, students should be able to...

Derive	Obtain the secant update from a line through two points.
Compute	Apply the method to polynomial and transcendental equations.
Evaluate	Explain convergence, stopping criteria and failure modes.

Basic idea

Find a root r satisfying f(r)=0.

Starting values	Choose two distinct estimates x_0 and x_1 near the desired root.
Approximate slope	Use $[f(x_1)-f(x_0)]/(x_1-x_0)$ instead of a derivative.
Update	Find the x-intercept of the secant line; retain the latest two estimates.

Geometric interpretation

A line through two points on the curve predicts the next estimate.

First point

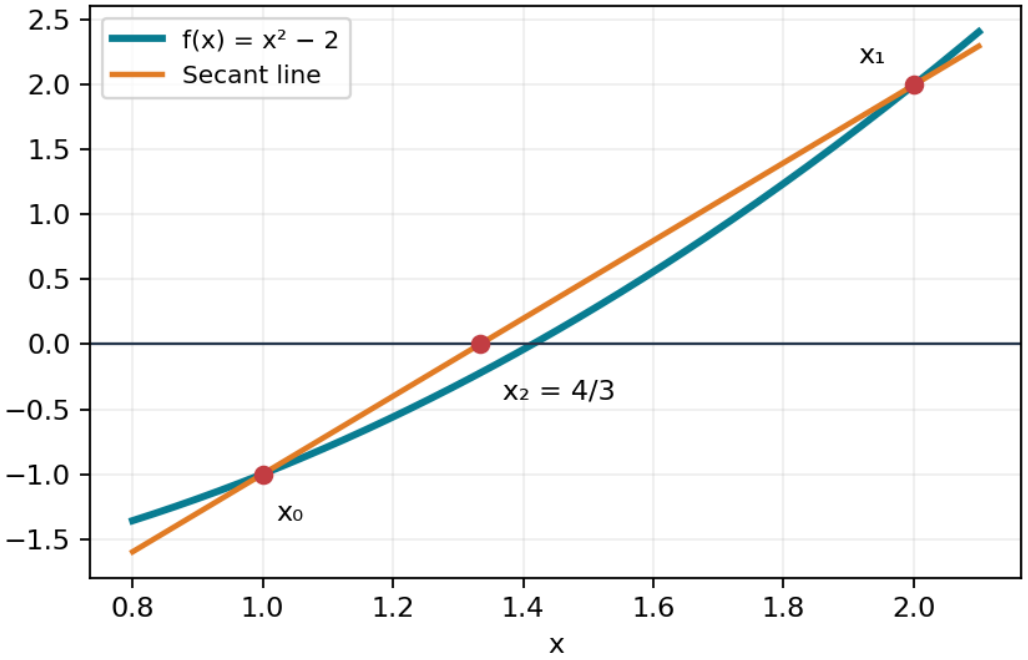
$(x_{n-1}, f(x_{n-1}))$

Second point

$(x_n, f(x_n))$

Intersection

The secant line meets the x-axis at x_{n+1}.



Derivation of the formula

Assume distinct estimates and unequal function values.

Slope

$$m = [f(x_n) - f(x_{n-1})] / (x_n - x_{n-1}).$$

Line equation

$$y - f(x_n) = m(x - x_n).\ \text{Set } y=0.$$

Secant update

$$x_{n+1} = x_n - f(x_n)(x_n - x_{n-1}) / [f(x_n) - f(x_{n-1})].$$

Relation to Newton and false position

Three methods use straight-line approximations.

Newton	Uses the tangent slope $f'(x_n)$; requires a derivative.
Secant	Uses the latest two function values; no derivative or bracket required.
False position	Uses a secant through bracket endpoints; preserves opposite signs.

Algorithm

Inputs: f, x_0, x_1, tolerances and maximum iteration count.

Step 1	Evaluate f at both estimates. Return an exact root if found.
Step 2	Check the denominator; compute the secant update.
Step 3	Test step and residual; otherwise shift the two estimates and repeat.

Example 1: polynomial equation

Solve $f(x)=x^2-2=0$ using $x_0=1$ and $x_1=2$.

Initial values	$f(1)=-1$; $f(2)=2$.
First update	$x_2=2-2(2-1)/(2-(-1))=1.33333333$.
Second update	$f(x_2)=-0.22222222$; $x_3=1.4$. The positive root is $\sqrt{2}$.

Example 1: iteration table

Each row shows the new estimate and its residual magnitude.

Updates 2–3	2: x = 1.3333333333; |f(x)| = 0.222 3: x = 1.4000000000; |f(x)| = 0.04
Updates 4–5	4: x = 1.4146341463; |f(x)| = 0.00119 5: x = 1.4142114385; |f(x)| = 6.01e-06
Updates 6–7	6: x = 1.4142135621; |f(x)| = 8.93e-10 7: x = 1.4142135624; |f(x)| = 8.88e-16

Example 2: transcendental equation

Solve $f(x)=\cos(x)-x=0$; angles in radians.

Initial values	$x_0=0$, $x_1=1$; $f(0)=1$, $f(1)\approx-0.45969769$.
First update	$x_2=1-(-0.45969769)/(-1.45969769)\approx0.68507336$.
Goal	The root is approximately 0.7390851332.

Example 2: iteration table

Observe the residual shrinking as the estimates approach the root.

Updates 2–3	2: x = 0.6850733573; |f(x)| = 0.0893 3: x = 0.7362989976; |f(x)| = 0.00466
Updates 4–5	4: x = 0.7391193619; |f(x)| = 5.73e-05 5: x = 0.7390851121; |f(x)| = 3.53e-08
Updates 6–7	6: x = 0.7390851332; |f(x)| = 2.67e-13 7: x = 0.7390851332; |f(x)| = 0

Rate of convergence

Assume f is sufficiently smooth and r is a simple root.

Local error	Let $e_n = x_n - r$. Near a simple root, $e_{n+1} \approx [f''(r)/(2f'(r))]e_n e_{n-1}$.
Order	For the usual nondegenerate case, $p^2 = p+1$, so $p = (1+\sqrt{5})/2 \approx 1.618$.
Interpretation	Superlinear convergence; typically faster than linear, slower than quadratic.

Stopping criteria

Do not rely on a small step alone.

Step test	$\lvert x_{n+1} - x_n \rvert \le \text{atol} + \text{rtol}\lvert x_{n+1} \rvert$.
Residual test	Require $\lvert f(x_{n+1}) \rvert \le \text{ftol}$ as well; use a maximum iteration count.
Error bound	If $\lvert f' \rvert \ge m > 0$ between estimate and root, error $\le \lvert f(x) \rvert / m$.

Failure modes and safeguards

The secant method is an open method, not a bracketing method.

Zero denominator	$f(x)=x^2-2$ with guesses -1 and 1 gives identical function values.
Poor starting values	Updates may leave the domain, diverge or approach an unintended root.
Safeguard	Reject nonfinite values; restart or fall back to a bracketed method.

Comparison of root-finding methods

Compare assumptions and typical local behavior.

Bisection	Bracket required; guaranteed interval reduction for continuous f.
Secant	Two starting estimates; one new function evaluation per update when cached.
Newton	One starting estimate; derivative needed; typically quadratic near a simple root.

Python implementation

Example: secant(lambda x: x*x-2, 1., 2.).

Initialize

```
def secant(f, a, b, tol=1e-8, limit=100):
    fa, fb = f(a), f(b)
    for _ in range(limit):
```

Update

```
        if fb == 0: return b
        if fa == 0: return a
        if fb == fa: raise ValueError("zero slope")
        c = b - fb*(b-a)/(fb-fa)
```

Stop / repeat

```
        fc = f(c)
        if abs(c-b)<=tol and abs(fc)<=tol: return c
        a, b, fa, fb = b, c, fb, fc
    raise RuntimeError("no convergence")
```

Practice questions

Attempt the questions before viewing the solutions.

Q1 Apply two secant updates to $x^2-5=0$ starting at 2 and 3.

Q2 Explain why equal function values cause a failure.

Q3 Compare secant and false position in terms of bracket preservation.

Solutions and recap

Next: NC 12 — Comparison and Convergence Analysis.

A1	$x_2=2.2$; $x_3\approx2.23076923$.
A2	Equal function values give a horizontal secant and zero denominator.
A3 / Takeaway	False position preserves a sign-changing bracket; secant uses the latest two estimates.