Solutions of Equations in One Variable

Secant & Regula Falsi Methods

Numerical Analysis (9th Edition)

R L Burden & J D Faires

Beamer Presentation Slides
prepared by
John Carroll
Dublin City University

Outline

1. Secant Method: Derivation & Algorithm

Outline

1. Secant Method: Derivation & Algorithm

2. Comparing the Secant & Newton's Methods

Outline

1. Secant Method: Derivation & Algorithm
2. Comparing the Secant & Newton's Methods
3. The Method of False Position (Regula Falsi)

Outline

1. Secant Method: Derivation & Algorithm
2. Comparing the Secant & Newton's Methods
3. The Method of False Position (Regula Falsi)

Rationale for the Secant Method

Problems with Newton's Method

Rationale for the Secant Method

Problems with Newton's Method

- Newton's method is an extremely powerful technique, but it has a major weakness: the need to know the value of the derivative of f at each approximation.

Rationale for the Secant Method

Problems with Newton's Method

- Newton's method is an extremely powerful technique, but it has a major weakness: the need to know the value of the derivative of f at each approximation.
- Frequently, $f'(x)$ is far more difficult and needs more arithmetic operations to calculate than $f(x)$.

Derivation of the Secant Method

$$f'(p_{n-1}) = \lim_{x \to p_{n-1}} \frac{f(x) - f(p_{n-1})}{x - p_{n-1}}.$$

Circumvent the Derivative Evaluation

Derivation of the Secant Method

$$f'(p_{n-1}) = \lim_{x \to p_{n-1}} \frac{f(x) - f(p_{n-1})}{x - p_{n-1}}.$$

Circumvent the Derivative Evaluation

If p_{n-2} is close to p_{n-1}, then

$$f'(p_{n-1}) \approx \frac{f(p_{n-2}) - f(p_{n-1})}{p_{n-2} - p_{n-1}} = \frac{f(p_{n-1}) - f(p_{n-2})}{p_{n-1} - p_{n-2}}.$$

Derivation of the Secant Method

$$f'(p_{n-1}) = \lim_{x \to p_{n-1}} \frac{f(x) - f(p_{n-1})}{x - p_{n-1}}.$$

Circumvent the Derivative Evaluation

If p_{n-2} is close to p_{n-1}, then

$$f'(p_{n-1}) \approx \frac{f(p_{n-2}) - f(p_{n-1})}{p_{n-2} - p_{n-1}} = \frac{f(p_{n-1}) - f(p_{n-2})}{p_{n-1} - p_{n-2}}.$$

Using this approximation for $f'(p_{n-1})$ in Newton's formula gives

$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}$$

Derivation of the Secant Method

$$f'(p_{n-1}) = \lim_{x \to p_{n-1}} \frac{f(x) - f(p_{n-1})}{x - p_{n-1}}.$$

Circumvent the Derivative Evaluation

If p_{n-2} is close to p_{n-1}, then

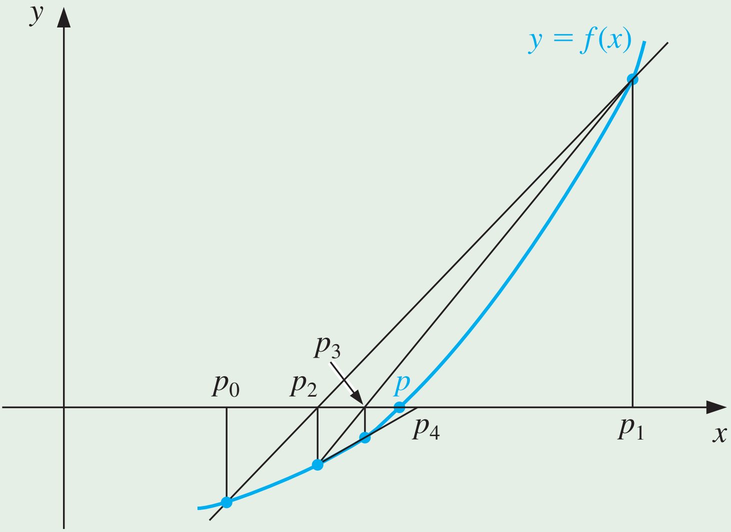
$$f'(p_{n-1}) \approx \frac{f(p_{n-2}) - f(p_{n-1})}{p_{n-2} - p_{n-1}} = \frac{f(p_{n-1}) - f(p_{n-2})}{p_{n-1} - p_{n-2}}.$$

Using this approximation for $f'(p_{n-1})$ in Newton's formula gives

$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}$$

This technique is called the Secant method

Secant Method: Using Successive Secants



The Secant Method

$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}$$

Procedure

The Secant Method

$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}$$

Procedure

- Starting with the two initial approximations p_0 and p_1, the approximation p_2 is the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_1, f(p_1))$.

The Secant Method

$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}$$

Procedure

- Starting with the two initial approximations p_0 and p_1, the approximation p_2 is the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_1, f(p_1))$.
- The approximation p_3 is the x-intercept of the line joining $(p_1, f(p_1))$ and $(p_2, f(p_2))$, and so on.

The Secant Method

$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}$$

Procedure

- Starting with the two initial approximations p_0 and p_1, the approximation p_2 is the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_1, f(p_1))$.
- The approximation p_3 is the x-intercept of the line joining $(p_1, f(p_1))$ and $(p_2, f(p_2))$, and so on.
- Note that only one function evaluation is needed per step for the Secant method after p_2 has been determined.

The Secant Method

$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}$$

Procedure

- Starting with the two initial approximations p_0 and p_1, the approximation p_2 is the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_1, f(p_1))$.
- The approximation p_3 is the x-intercept of the line joining $(p_1, f(p_1))$ and $(p_2, f(p_2))$, and so on.
- Note that only one function evaluation is needed per step for the Secant method after p_2 has been determined.
- In contrast, each step of Newton's method requires an evaluation of both the function and its derivative.

The Secant Method: Algorithm

To find a solution to $f(x) = 0$ given initial approximations p_0 and p_1; tolerance TOL; maximum number of iterations N_0.

The Secant Method: Algorithm

To find a solution to $f(x) = 0$ given initial approximations p_0 and p_1; tolerance TOL; maximum number of iterations N_0.

1 Set $i = 2$, $q_0 = f(p_0)$, $q_1 = f(p_1)$

The Secant Method: Algorithm

To find a solution to $f(x) = 0$ given initial approximations p_0 and p_1; tolerance TOL; maximum number of iterations N_0.

1 Set $i = 2$, $q_0 = f(p_0)$, $q_1 = f(p_1)$

2 While $i \leq N_0$ do Steps 3–6:

The Secant Method: Algorithm

To find a solution to $f(x) = 0$ given initial approximations p_0 and p_1; tolerance *TOL*; maximum number of iterations N_0.

1 Set $i = 2$, $q_0 = f(p_0)$, $q_1 = f(p_1)$

2 While $i \leq N_0$ do Steps 3–6:

  3 Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute* p_i)

The Secant Method: Algorithm

To find a solution to $f(x) = 0$ given initial approximations p_0 and p_1; tolerance TOL; maximum number of iterations N_0.

1 Set $i = 2$, $q_0 = f(p_0)$, $q_1 = f(p_1)$

2 While $i \leq N_0$ do Steps 3–6:

   3 Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute* p_i)

   4 If $|p - p_1| < TOL$ then
   OUTPUT (p); (*The procedure was successful.*) STOP

The Secant Method: Algorithm

To find a solution to $f(x) = 0$ given initial approximations p_0 and p_1; tolerance TOL; maximum number of iterations N_0.

1 Set $i = 2$, $q_0 = f(p_0)$, $q_1 = f(p_1)$

2 While $i \leq N_0$ do Steps 3–6:

   3 Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute p_i*)

   4 If $|p - p_1| < TOL$ then
   OUTPUT (p); (*The procedure was successful.*) STOP

   5 Set $i = i + 1$

The Secant Method: Algorithm

To find a solution to $f(x) = 0$ given initial approximations p_0 and p_1; tolerance TOL; maximum number of iterations N_0.

1 Set $i = 2$, $q_0 = f(p_0)$, $q_1 = f(p_1)$

2 While $i \leq N_0$ do Steps 3–6:

 3 Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute* p_i)

 4 If $|p - p_1| < TOL$ then
 OUTPUT (p); (*The procedure was successful.*) STOP

 5 Set $i = i + 1$

 6 Set $p_0 = p_1$; (*Update* p_0, q_0, p_1, q_1)
 $q_0 = q_1$; $p_1 = p$; $q_1 = f(p)$

The Secant Method: Algorithm

To find a solution to $f(x) = 0$ given initial approximations p_0 and p_1; tolerance TOL; maximum number of iterations N_0.

1 Set $i = 2$, $q_0 = f(p_0)$, $q_1 = f(p_1)$

2 While $i \leq N_0$ do Steps 3–6:

 3 Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute* p_i)
 4 If $|p - p_1| < TOL$ then
 OUTPUT (p); (*The procedure was successful.*) STOP
 5 Set $i = i + 1$
 6 Set $p_0 = p_1$; (*Update* p_0, q_0, p_1, q_1)
 $q_0 = q_1$; $p_1 = p$; $q_1 = f(p)$

7 OUTPUT ('The method failed after N_0 iterations, $N_0 =$', N_0);
(*The procedure was unsuccessful*) STOP

Outline

1 Secant Method: Derivation & Algorithm

2 Comparing the Secant & Newton's Methods

3 The Method of False Position (Regula Falsi)

Comparing the Secant & Newton's Methods

Example: $f(x) = \cos x - x$

Use the Secant method to find a solution to $x = \cos x$, and compare the approximations with those given by Newton's method with $p_0 = \pi/4$.

Formula for the Secant Method

Comparing the Secant & Newton's Methods

Example: $f(x) = \cos x - x$

Use the Secant method to find a solution to $x = \cos x$, and compare the approximations with those given by Newton's method with $p_0 = \pi/4$.

Formula for the Secant Method

We need two initial approximations. Suppose we use $p_0 = 0.5$ and $p_1 = \pi/4$.

Comparing the Secant & Newton's Methods

Example: $f(x) = \cos x - x$

Use the Secant method to find a solution to $x = \cos x$, and compare the approximations with those given by Newton's method with $p_0 = \pi/4$.

Formula for the Secant Method

We need two initial approximations. Suppose we use $p_0 = 0.5$ and $p_1 = \pi/4$. Succeeding approximations are generated by the formula

$$p_n = p_{n-1} - \frac{(p_{n-1} - p_{n-2})(\cos p_{n-1} - p_{n-1})}{(\cos p_{n-1} - p_{n-1}) - (\cos p_{n-2} - p_{n-2})}, \quad \text{for } n \geq 2.$$

Comparing the Secant & Newton's Methods

Newton's Method for $f(x) = \cos(x) - x$, $p_0 = \frac{\pi}{4}$

n	p_{n-1}	$f(p_{n-1})$	$f'(p_{n-1})$	p_n	$\lvert p_n - p_{n-1}\rvert$
1	0.78539816	-0.078291	-1.707107	0.73953613	0.04586203
2	0.73953613	-0.000755	-1.673945	0.73908518	0.00045096
3	0.73908518	-0.000000	-1.673612	0.73908513	0.00000004
4	0.73908513	-0.000000	-1.673612	0.73908513	0.00000000

Comparing the Secant & Newton's Methods

Newton's Method for $f(x) = \cos(x) - x$, $p_0 = \frac{\pi}{4}$

n	p_{n-1}	$f(p_{n-1})$	$f'(p_{n-1})$	p_n	$\lvert p_n - p_{n-1} \rvert$
1	0.78539816	-0.078291	-1.707107	0.73953613	0.04586203
2	0.73953613	-0.000755	-1.673945	0.73908518	0.00045096
3	0.73908518	-0.000000	-1.673612	0.73908513	0.00000004
4	0.73908513	-0.000000	-1.673612	0.73908513	0.00000000

- An excellent approximation is obtained with $n = 3$.

Comparing the Secant & Newton's Methods

Newton's Method for $f(x) = \cos(x) - x$, $p_0 = \frac{\pi}{4}$

n	p_{n-1}	$f(p_{n-1})$	$f'(p_{n-1})$	p_n	$\lvert p_n - p_{n-1}\rvert$
1	0.78539816	-0.078291	-1.707107	0.73953613	0.04586203
2	0.73953613	-0.000755	-1.673945	0.73908518	0.00045096
3	0.73908518	-0.000000	-1.673612	0.73908513	0.00000004
4	0.73908513	-0.000000	-1.673612	0.73908513	0.00000000

- An excellent approximation is obtained with $n = 3$.
- Because of the agreement of p_3 and p_4 we could reasonably expect this result to be accurate to the places listed.

Comparing the Secant & Newton's Methods

Secant Method for $f(x) = \cos(x) - x$, $p_0 = 0.5$, $p_1 = \frac{\pi}{4}$

n	p_{n-2}	p_{n-1}	p_n	$\|p_n - p_{n-1}\|$
2	0.500000000	0.785398163	0.736384139	0.0490140246
3	0.785398163	0.736384139	0.739058139	0.0026740004
4	0.736384139	0.739058139	0.739085149	0.0000270101
5	0.739058139	0.739085149	0.739085133	0.0000000161

Comparing the Secant & Newton's Methods

Secant Method for $f(x) = \cos(x) - x$, $p_0 = 0.5$, $p_1 = \frac{\pi}{4}$

n	p_{n-2}	p_{n-1}	p_n	$\lvert p_n - p_{n-1} \rvert$
2	0.500000000	0.785398163	0.736384139	0.0490140246
3	0.785398163	0.736384139	0.739058139	0.0026740004
4	0.736384139	0.739058139	0.739085149	0.0000270101
5	0.739058139	0.739085149	0.739085133	0.0000000161

- Comparing results, we see that the Secant Method approximation p_5 is accurate to the tenth decimal place, whereas Newton's method obtained this accuracy by p_3.

Comparing the Secant & Newton's Methods

Secant Method for $f(x) = \cos(x) - x$, $p_0 = 0.5$, $p_1 = \frac{\pi}{4}$

n	p_{n-2}	p_{n-1}	p_n	$\lvert p_n - p_{n-1}\rvert$
2	0.500000000	0.785398163	0.736384139	0.0490140246
3	0.785398163	0.736384139	0.739058139	0.0026740004
4	0.736384139	0.739058139	0.739085149	0.0000270101
5	0.739058139	0.739085149	0.739085133	0.0000000161

- Comparing results, we see that the Secant Method approximation p_5 is accurate to the tenth decimal place, whereas Newton's method obtained this accuracy by p_3.
- Here, the convergence of the Secant method is much faster than functional iteration but slightly slower than Newton's method.

Comparing the Secant & Newton's Methods

Secant Method for $f(x) = \cos(x) - x$, $p_0 = 0.5$, $p_1 = \frac{\pi}{4}$

n	p_{n-2}	p_{n-1}	p_n	$\lvert p_n - p_{n-1} \rvert$
2	0.500000000	0.785398163	0.736384139	0.0490140246
3	0.785398163	0.736384139	0.739058139	0.0026740004
4	0.736384139	0.739058139	0.739085149	0.0000270101
5	0.739058139	0.739085149	0.739085133	0.0000000161

- Comparing results, we see that the Secant Method approximation p_5 is accurate to the tenth decimal place, whereas Newton's method obtained this accuracy by p_3.
- Here, the convergence of the Secant method is much faster than functional iteration but slightly slower than Newton's method.
- This is generally the case. Order of Convergence

The Secant Method

Final Remarks

The Secant Method

Final Remarks

- The Secant method and Newton's method are often used to refine an answer obtained by another technique (such as the Bisection Method).

The Secant Method

Final Remarks

- The Secant method and Newton's method are often used to refine an answer obtained by another technique (such as the Bisection Method).
- Both methods require good first approximations but generally give rapid acceleration.

Outline

1. Secant Method: Derivation & Algorithm
2. Comparing the Secant & Newton's Methods
3. The Method of False Position (Regula Falsi)

The Method of False Position

Bracketing a Root

The Method of False Position

Bracketing a Root

- Unlike the Bisection Method, root bracketing is not guaranteed for either Newton's method or the Secant method.

The Method of False Position

Bracketing a Root

- Unlike the Bisection Method, root bracketing is not guaranteed for either Newton's method or the Secant method.
- The method of False Position (also called *Regula Falsi*) generates approximations in the same manner as the Secant method, but it includes a test to ensure that the root is always bracketed between successive iterations.

The Method of False Position

Bracketing a Root

- Unlike the Bisection Method, root bracketing is not guaranteed for either Newton's method or the Secant method.
- The method of False Position (also called *Regula Falsi*) generates approximations in the same manner as the Secant method, but it includes a test to ensure that the root is always bracketed between successive iterations.
- Although it is not a method we generally recommend, it illustrates how bracketing can be incorporated.

The Method of False Position

Construction of the Method

- First choose initial approximations p_0 and p_1 with $f(p_0) \cdot f(p_1) < 0$.

The Method of False Position

Construction of the Method

- First choose initial approximations p_0 and p_1 with $f(p_0) \cdot f(p_1) < 0$.
- The approximation p_2 is chosen in the same manner as in the Secant method, as the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_1, f(p_1))$.

The Method of False Position

Construction of the Method

- First choose initial approximations p_0 and p_1 with $f(p_0) \cdot f(p_1) < 0$.
- The approximation p_2 is chosen in the same manner as in the Secant method, as the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_1, f(p_1))$.
- To decide which secant line to use to compute p_3, consider $f(p_2) \cdot f(p_1)$, or more correctly $\operatorname{sgn} f(p_2) \cdot \operatorname{sgn} f(p_1)$:

The Method of False Position

Construction of the Method

- First choose initial approximations p_0 and p_1 with $f(p_0) \cdot f(p_1) < 0$.
- The approximation p_2 is chosen in the same manner as in the Secant method, as the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_1, f(p_1))$.
- To decide which secant line to use to compute p_3, consider $f(p_2) \cdot f(p_1)$, or more correctly $\operatorname{sgn} f(p_2) \cdot \operatorname{sgn} f(p_1)$:
  - If $\operatorname{sgn} f(p_2) \cdot \operatorname{sgn} f(p_1) < 0$, then p_1 and p_2 bracket a root. Choose p_3 as the x-intercept of the line joining $(p_1, f(p_1))$ and $(p_2, f(p_2))$.

The Method of False Position

Construction of the Method

- First choose initial approximations p_0 and p_1 with $f(p_0) \cdot f(p_1) < 0$.
- The approximation p_2 is chosen in the same manner as in the Secant method, as the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_1, f(p_1))$.
- To decide which secant line to use to compute p_3, consider $f(p_2) \cdot f(p_1)$, or more correctly $\operatorname{sgn} f(p_2) \cdot \operatorname{sgn} f(p_1)$:
  - If $\operatorname{sgn} f(p_2) \cdot \operatorname{sgn} f(p_1) < 0$, then p_1 and p_2 bracket a root. Choose p_3 as the x-intercept of the line joining $(p_1, f(p_1))$ and $(p_2, f(p_2))$.
  - If not, choose p_3 as the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_2, f(p_2))$, and then interchange the indices on p_0 and p_1.

The Method of False Position

Construction of the Method (Cont'd)

The Method of False Position

Construction of the Method (Cont'd)

- In a similar manner, once p_3 is found, the sign of $f(p_3) \cdot f(p_2)$ determines whether we use p_2 and p_3 or p_3 and p_1 to compute p_4.

The Method of False Position

Construction of the Method (Cont'd)

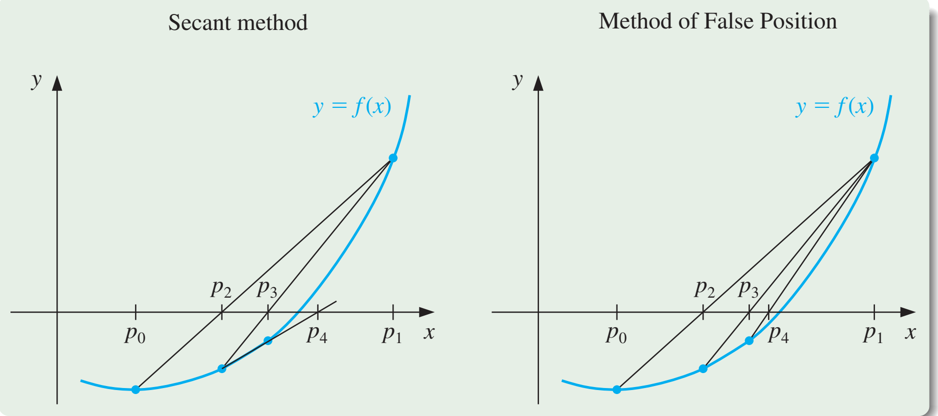
- In a similar manner, once p_3 is found, the sign of $f(p_3) \cdot f(p_2)$ determines whether we use p_2 and p_3 or p_3 and p_1 to compute p_4.
- In the latter case, a relabeling of p_2 and p_1 is performed.

The Method of False Position

Construction of the Method (Cont'd)

- In a similar manner, once p_3 is found, the sign of $f(p_3) \cdot f(p_2)$ determines whether we use p_2 and p_3 or p_3 and p_1 to compute p_4.
- In the latter case, a relabeling of p_2 and p_1 is performed.
- The relabelling ensures that the root is bracketed between successive iterations.

Secant Method & Method of False Position



In this illustration, the first three approximations are the same for both methods, but the fourth approximations differ.

The Method of False Position: Algorithm

To find a solution to $f(x) = 0$, given the continuous function f on the interval $[p_0, p_1]$ (where $f(p_0)$ and $f(p_1)$ have opposite signs) tolerance TOL and maximum number of iterations N_0.

The Method of False Position: Algorithm

To find a solution to $f(x) = 0$, given the continuous function f on the interval $[p_0, p_1]$ (where $f(p_0)$ and $f(p_1)$ have opposite signs) tolerance TOL and maximum number of iterations N_0.

1. Set $i = 2$; $q_0 = f(p_0)$; $q_1 = f(p_1)$.

The Method of False Position: Algorithm

To find a solution to $f(x) = 0$, given the continuous function f on the interval $[p_0, p_1]$ (where $f(p_0)$ and $f(p_1)$ have opposite signs) tolerance *TOL* and maximum number of iterations N_0.

1. Set $i = 2$; $q_0 = f(p_0)$; $q_1 = f(p_1)$.
2. While $i \leq N_0$ do Steps 3–7:

The Method of False Position: Algorithm

To find a solution to $f(x) = 0$, given the continuous function f on the interval $[p_0, p_1]$ (where $f(p_0)$ and $f(p_1)$ have opposite signs) tolerance TOL and maximum number of iterations N_0.

1. Set $i = 2$; $q_0 = f(p_0)$; $q_1 = f(p_1)$.
2. While $i \leq N_0$ do Steps 3–7:
   3. Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. $\quad$ (*Compute* p_i)

The Method of False Position: Algorithm

To find a solution to $f(x) = 0$, given the continuous function f on the interval $[p_0, p_1]$ (where $f(p_0)$ and $f(p_1)$ have opposite signs) tolerance TOL and maximum number of iterations N_0.

1 Set $i = 2$; $q_0 = f(p_0)$; $q_1 = f(p_1)$.

2 While $i \leq N_0$ do Steps 3–7:

   3 Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute* p_i)

   4 If $|p - p_1| < TOL$ then
   OUTPUT (p); (*The procedure was successful*): STOP

The Method of False Position: Algorithm

To find a solution to $f(x) = 0$, given the continuous function f on the interval $[p_0, p_1]$ (where $f(p_0)$ and $f(p_1)$ have opposite signs) tolerance *TOL* and maximum number of iterations N_0.

1 Set $i = 2$; $q_0 = f(p_0)$; $q_1 = f(p_1)$.

2 While $i \leq N_0$ do Steps 3–7:

3 Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute* p_i)

4 If $|p - p_1| < TOL$ then
OUTPUT (p); (*The procedure was successful*): STOP

5 Set $i = i + 1$; $q = f(p)$

The Method of False Position: Algorithm

To find a solution to $f(x) = 0$, given the continuous function f on the interval $[p_0, p_1]$ (where $f(p_0)$ and $f(p_1)$ have opposite signs) tolerance TOL and maximum number of iterations N_0.

1. Set $i = 2$; $q_0 = f(p_0)$; $q_1 = f(p_1)$.
2. While $i \leq N_0$ do Steps 3–7:
   3. Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute p_i*)
   4. If $|p - p_1| < TOL$ then
      OUTPUT (p); (*The procedure was successful*): STOP
   5. Set $i = i + 1$; $q = f(p)$
   6. If $q \cdot q_1 < 0$ then set $p_0 = p_1$; $q_0 = q_1$

The Method of False Position: Algorithm

To find a solution to $f(x) = 0$, given the continuous function f on the interval $[p_0, p_1]$ (where $f(p_0)$ and $f(p_1)$ have opposite signs) tolerance TOL and maximum number of iterations N_0.

1 Set $i = 2$; $q_0 = f(p_0)$; $q_1 = f(p_1)$.

2 While $i \leq N_0$ do Steps 3–7:

   3 Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute* p_i)
   4 If $|p - p_1| < TOL$ then
   OUTPUT (p); (*The procedure was successful*): STOP
   5 Set $i = i + 1$; $q = f(p)$
   6 If $q \cdot q_1 < 0$ then set $p_0 = p_1$; $q_0 = q_1$
   7 Set $p_1 = p$; $q_1 = q$

The Method of False Position: Algorithm

To find a solution to $f(x) = 0$, given the continuous function f on the interval $[p_0, p_1]$ (where $f(p_0)$ and $f(p_1)$ have opposite signs) tolerance TOL and maximum number of iterations N_0.

1. Set $i = 2$; $q_0 = f(p_0)$; $q_1 = f(p_1)$.
2. While $i \leq N_0$ do Steps 3–7:
   3. Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute* p_i)
   4. If $|p - p_1| < TOL$ then
      OUTPUT (p); (*The procedure was successful*): STOP
   5. Set $i = i + 1$; $q = f(p)$
   6. If $q \cdot q_1 < 0$ then set $p_0 = p_1$; $q_0 = q_1$
   7. Set $p_1 = p$; $q_1 = q$
8. OUTPUT ('Method failed after N_0 iterations, $N_0 =$', N_0);
   (*The procedure was unsuccessful*): STOP

The Method of False Position: Numerical Calculations

Comparison with Newton & Secant Methods

Use the method of False Position to find a solution to $x = \cos x$, and compare the approximations with those given in a previous example which Newton's method and the Secant Method.

The Method of False Position: Numerical Calculations

Comparison with Newton & Secant Methods

Use the method of False Position to find a solution to $x = \cos x$, and compare the approximations with those given in a previous example which Newton's method and the Secant Method.

To make a reasonable comparison we will use the same initial approximations as in the Secant method, that is, $p_0 = 0.5$ and $p_1 = \pi/4$.

The Method of False Position: Numerical Calculations

Comparison with Newton's Method & Secant Method

	False Position	**Secant**	**Newton**
n	p_n	p_n	p_n
0	0.5	0.5	0.7853981635
1	0.7853981635	0.7853981635	0.7395361337
2	0.7363841388	0.7363841388	0.7390851781
3	0.7390581392	0.7390581392	0.7390851332
4	0.7390848638	0.7390851493	0.7390851332
5	0.7390851305	0.7390851332	
6	0.7390851332		

The Method of False Position: Numerical Calculations

Comparison with Newton's Method & Secant Method

	False Position	**Secant**	**Newton**
n	p_n	p_n	p_n
0	0.5	0.5	0.7853981635
1	0.7853981635	0.7853981635	0.7395361337
2	0.7363841388	0.7363841388	0.7390851781
3	0.7390581392	0.7390581392	0.7390851332
4	0.7390848638	0.7390851493	0.7390851332
5	0.7390851305	0.7390851332	
6	0.7390851332		

Note that the False Position and Secant approximations agree through p_3 and that the method of False Position requires an additional iteration to obtain the same accuracy as the Secant method.

The Method of False Position

Final Remarks

The Method of False Position

Final Remarks

- The added insurance of the method of False Position commonly requires more calculation than the Secant method, . . .

The Method of False Position

Final Remarks

- The added insurance of the method of False Position commonly requires more calculation than the Secant method, . . .
- just as the simplification that the Secant method provides over Newton's method usually comes at the expense of additional iterations.

Questions?

Reference Material

Order of Convergence of the Secant Method

Exercise 14, Section 2.4

It can be shown (see, for example, Dahlquist and Å. Björck (1974), pp. 228–229), that if $\{p_n\}_{n=0}^{\infty}$ are convergent Secant method approximations to p, the solution to $f(x) = 0$, then a constant C exists with

$$|p_{n+1} - p| \approx C\,|p_n - p|\,|p_{n-1} - p|$$

for sufficiently large values of n. Assume $\{p_n\}$ converges to p of order α, and show that

$$\alpha = (1 + \sqrt{5})/2$$

(*Note:* This implies that the order of convergence of the Secant method is approximately 1.62).

Return to the Secant Method

Dahlquist, G. and Å. Björck (Translated by N. Anderson), *Numerical methods*, Prentice-Hall, Englewood Cliffs, NJ, 1974.