2 Methods for solving nonlinear eqns $f(x)=0$ numerically

(1) Picard Iteration

\[ f(x) = \frac{1}{2} x^2 - x + \frac{1}{4} \]

Solve $f(x) = 0$.

\[ x^2 - 2x + \frac{1}{2} = 0 \]

\[ x = \frac{2 \pm \sqrt{4-2}}{2} = 1 \pm \frac{\sqrt{2}}{2} = 1.292893219 \ldots \]

or \[ 1.707106781 \ldots \]

Note $f(x) = g(x) - x$

where \[ g(x) = \frac{1}{2} x^2 + \frac{1}{4} \]

To solve $f(x) = 0$.

Find $x$ s.t. $x = g(x)$

Picard Iteration

\[ x_{n+1} = g(x_n) \]

(initial guess $x_0$)

(Def: $\lim_{n \to \infty} x_n = x^*$ if for all $\varepsilon > 0$ there exists an $N > 0$ s.t. $n > N \Rightarrow |x_n - x^*| < \varepsilon$)

If $g$ is continuous then

\[ x^* = \lim_{n \to \infty} x_{n+1} = \lim_{n \to \infty} g(x_n) = g(x^*) \]

So $x^* = g(x^*)$ and $x^*$ is the solution.

The $x_n$'s are called the iterates and are said to converge to the solution, $x^*$. 
Picard iteration from initial guess $x_0 = 0$

<table>
<thead>
<tr>
<th>$n$</th>
<th>$x_n$</th>
<th>$g(x_n) = \frac{1}{2} x_n^2 + \frac{1}{4}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.25</td>
<td>$0.28125$</td>
</tr>
<tr>
<td>2</td>
<td>0.28125</td>
<td>$0.289550781$</td>
</tr>
<tr>
<td>3</td>
<td>0.289550781</td>
<td>$0.291919827$</td>
</tr>
<tr>
<td>4</td>
<td>0.291919827</td>
<td>$0.292608593$</td>
</tr>
<tr>
<td>5</td>
<td>0.292608593</td>
<td>$0.292809894$</td>
</tr>
<tr>
<td>6</td>
<td>0.292809894</td>
<td>$0.292868817$</td>
</tr>
</tbody>
</table>

$x^* = 0.292893219$

picture of convergence

\[ y = x \]
\[ y = g(x) \]
Convergence analysis for Picard iteration

Suppose \( f(x) = g(x) - x \) where \( g'(x) \) exists and \( f(x^*) = 0 \), i.e., \( x^* = g(x^*) \).

The error at the \( n \)th iterate is

\[
e_n = x^* - x_n
\]

\[
= g(x^*) - g(x_{n-1}) \]

\[
= g'(c_n)(x^* - x_{n-1}) \quad \text{for some } c_n \text{ between } x^* \text{ and } x_{n-1}
\]

\[
= g'(c_n) e_{n-1}
\]

\[
\approx [g'(x^*)]^n e_0 \quad \text{(linear convergence, not so fast)}
\]

since \( g'(c_n) \approx g'(x^*) \) if \( x_n \) is near \( x^* \).

That is, if we have a close enough initial guess the error will improve by a factor of about \( g'(x^*) \) at each step. So if \( g'(x^*) \approx \frac{1}{10} \) we would gain a decimal place each step.

Note for \( g \) in our example

\[
g'(x^*) = x^* = 0.292\ldots
\]
(2) Newton's method for \( f(x) = \frac{1}{2} x^2 - x + \frac{1}{4} \)

\[
f'(x) = x - 1
\]

\[
\chi_{m+1} = \chi_m - \frac{f(\chi_m)}{f'(\chi_m)} = \chi_m - \frac{\frac{1}{2} \chi_m^2 - \chi_m + \frac{1}{4}}{\chi_m - 1}
\]

\[
= \chi_m - \frac{\frac{1}{2} \chi_m^2 - \chi_m + \frac{1}{4}}{\chi_m - 1}
\]

\[
= \frac{\frac{1}{2} \chi_m^2 - \frac{1}{4}}{\chi_m - 1} = \frac{\chi_m^2 - \frac{1}{2}}{2(\chi_m - 1)}
\]

<table>
<thead>
<tr>
<th>( m )</th>
<th>( \chi_m )</th>
<th>( \chi_{m+1} = \frac{\chi_m^2 - \frac{1}{2}}{2(\chi_m - 1)} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>( \frac{1}{4} = 0.25 )</td>
</tr>
<tr>
<td>1</td>
<td>0.25</td>
<td>( \frac{3}{24} = 0.291666667 )</td>
</tr>
<tr>
<td>2</td>
<td>0.291666667</td>
<td>0.292892157</td>
</tr>
<tr>
<td>3</td>
<td>0.292892157</td>
<td>0.2928933219</td>
</tr>
</tbody>
</table>

Note how much faster Newton's method converges.
Convergence analysis for Newton's method

\[ y_i = f(x_i) \]

\[ y = f'(x_{m-1})(x - x_{m-1}) + f(x_{m-1}) \]

This is like a parabola \( \rightarrow \) (more in Calculus II)

error at \( n \)th iteration =

\[ e_m = \hat{x} - x_m \]

\[ = \hat{x} - x_{m-1} + f'(x_{m-1})^{-1} f(x_{m-1}) \]

\[ = f'(x_{m-1})^{-1} \left[ f'(x_{m-1})(\hat{x} - x_{m-1}) + f(x_{m-1}) \right] \]

\[ = f'(x_{m-1})^{-1} \left[ \frac{d}{dx} \left( \hat{x} - x_{m-1} \right)^2 \right] \] (Calc II)

\[ = f'(x_{m-1})^{-1} C \cdot e_{m-1}^2 \]

Suppose \( K \approx 1 \) then if

\[ e_1 \approx 10^{-1} \]

\[ e_2 \approx 10^{-2} \]

\[ e_3 \approx 10^{-4} \]

\[ e_4 \approx 10^{-8} \]

\[ \frac{\text{quadratic convergence}}{\text{very fast}} \]

i.e. \( e_m = O(e_{m-1}^2) \)
Example Newton's method for finding \( \sqrt{A} \) \((A > 0)\).

Solve \( f(x) = 0 \) where \( f(x) = x^2 - A \)

- tangent line at \((x_k, f(x_k))\)
  \[ y = f(x_k) + f'(x_k)(x - x_k) \]
- intersection of \(x\)-axis at \(x_{k+1}\)
  \[ 0 = f(x_k) + f'(x_k)(x_{k+1} - x_k) \]
  \[ x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \]
  \[ = x_k - \frac{x_k^2 - A}{2x_k} \]
  \[ = \frac{1}{2}(x_k + \frac{A}{x_k}) \]

(Note: \( f' \) must exist.)

Take e.g. \( A = 2 \). Since \( f(0) = -2 \) and \( f(2) = 2 \) and \( f \) is continuous, by the Intermediate Value Theorem \( f \) must have a root in \([0, 2]\)

We just need an initial guess, e.g. \( x_0 = 2 \), to get started.

Simple MATLAB example
To get started, type one of these: helpwin, helpdesk, or demo.
For product information, type tour or visit www.mathworks.com.

format long
x=2;
x=(x + 2/x)/2

x =
  1.50000000000000

x=(x + 2/x)/2

x =
  1.41666666666667

x=(x + 2/x)/2

x =
  1.41421568627451

x=(x + 2/x)/2

x =
  1.41421356237469

x=(x + 2/x)/2

x =
  1.41421356237309

x=(x + 2/x)/2

x =
  1.41421356237309

sqrt(2)

ans =
  1.41421356237310

Simple MATLAB example: Newton's method for finding 
\[ f(x) = x^2 - 2 = 0 \]
\[ x = \sqrt{2} \]
What can go wrong with Newton's method?

1) \( f'(x_*) = 0 \) or \( f'(x_0) \approx 0 \) for \( f(x_*) = 0 \).

If \( f'(x_*) = 0 \) and \( f \) is sufficiently differentiable then
\[
 f(x) = f(x_*) + f'(x_*) (x - x_*) + \frac{f''(x_*)}{2!} (x - x_*)^2 + O((x - x_*)^3)
\]
\[
= 0 \quad \because \quad 0
\]
i.e.
\[
 f(x) = \frac{f''(x_*)}{2} (x - x_*)^2 + O((x - x_*)^3) \quad \text{for } x \text{ near } x_*.
\]
and
\[
 f'(x) = f''(x_*) (x - x_*) + O((x - x_*)^2)
\]
Assuming \( f''(x_*) \neq 0 \) Newton's method for \( x = x_k \) near \( x_* \) gives
\[
 x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}
\]
\[
= x_k - \frac{\frac{f''(x_*)}{2} (x_k - x_*)^2 + O((x_k - x_*)^3)}{f''(x_*) (x_k - x_*) + O((x_k - x_*)^3)}
\]
\[
= x_k - \frac{1}{2} (x_k - x_*) + O((x_k - x_*)^2)
\]

\text{error at } k+1 \text{st step}
\[
e_{k+1} = x_{k+1} - x_* = \frac{1}{2} (x_k - x_*) + O((x_k - x_*)^2)
\]
\[
\approx \frac{1}{2} e_k = \frac{1}{2} \text{ error at } k\text{th step}
\]

\text{E.g.,} \: f(x) = x^2, \: f'(x_0) = 0 \: \text{for} \: x_* = 0
\]

\[
x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = x_k \frac{x_k^2}{2x_k} = x_k - \frac{1}{2} x_k = \frac{1}{2} x_k
\]

so
\[
x_{k+1} - x_* = \frac{1}{2} (x_k - x_*)
\]
Newton's method for
\[ f(x) = x^2 - 10^{-6} = 0 \]
\[ f'(x) = 0 \quad \text{for} \quad x_0 = 10^{-3} = 0.001 \]
\[ \text{and} \quad f'(x_0) = 2x_0 = 0.002 \approx 0 \]

initial convergence linear
with factor \( \approx \frac{1}{2} \)

See Theorem 8, p. 268 in Van Loan's book

final convergence quadratic
Bisection method (see above)

Suppose \( f(x) \) continuous in \([a, b]\) and \( f(a) f(b) \leq 0 \). Then \( f(x) = 0 \) for some \( x \in [a, b] \).

Let \( m = \frac{a+b}{2} \). Then either \( f(a) f(m) \leq 0 \) or \( f(m) f(b) \leq 0 \), so there is a root in \([a, m]\) or \([m, b]\).

Letting \( a_0 = a \), \( b_0 = b \),

\[ a_{k+1} = a_k, \quad b_{k+1} = \frac{a_k + b_k}{2} \text{ if } f(a_k) f\left(\frac{a_k + b_k}{2}\right) \leq 0 \]

and \( a_{k+1} = \frac{a_k + b_k}{2}, \quad b_{k+1} = b_k \) otherwise.

We setting \( x_k = \frac{a_k + b_k}{2} \) we have

\[ |x_{k+1} - x_k| \leq \frac{1}{2} |x_k - x_{k-1}| \leq \frac{1}{2^{k+1}} |a - b| \]

Linear convergence with factor \( \frac{1}{2} \)
Secant method ≈ Newton's method with \( f'(x_k) \) replaced by \( \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} \)

Example of secant:

\[ y = f(x_k) + \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} (x - x_k) \]

\( y = 0 \) for \( x = x_{k+1} = x_k - \frac{f(x_k) (x_k - x_{k-1})}{f(x_k) - f(x_{k-1})} \)

Requires 2 starting values \( x_0, x_1 \) and 1 evaluation of \( f(x_k) \) per step.

CBS convergence is "superlinear":

\[ |x_k - x_*| \leq C |x_k - x_*|^r \]

where \( r = \frac{1 + \sqrt{5}}{2} \approx 1.6 \). \( 1 \leq r \leq 2 \)

linear conv. \( \quad \) quadratic conv.