Last time: Methods for solving $f(x) = 0$

1- Secant Method

$$x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f_k - f_{k-1}} f_k$$

approximation of $\sqrt[3]{f(x_k)}$

2- Newton's Method

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Convergence is quadratic:

$$\lfloor x_k \rfloor \sim \frac{1}{3} x_k^2$$
Now: Revisit the secant method:

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

an approximation
to \( \frac{1}{f'(x_k)} \)

In general, the secant method converges slower than Newton's method, with a rate of:

\[ \lim_{k \to \infty} \frac{|x_k - x_{k+1}|}{|x_k - x_{k-1}|^q} = \text{const.} \]

\[ q = \frac{1}{2} \left( 1 + \sqrt{5} \right) \approx 1.6 \]
(proof omitted)

How about in 2 dimensions? What does Newton's method look like?

\[ \begin{align*}
\text{Ex: Solve} & \quad f_1(x_1, x_2) = 0 \quad \Rightarrow \quad f(x) = 0 \\
& \quad f_2(x_1, x_2) = 0 \\
\text{vector form.} &
\end{align*} \]
Let's recall the multi-variable Taylor series expansion:

\[
f(x_1, x_2) = f(y_1, y_2) + \left[ \frac{\partial f}{\partial x_1}(y_1, y_2) \right] (x_1 - y_1) + \left[ \frac{\partial f}{\partial x_2}(y_1, y_2) \right] (x_2 - y_2)
\]

\[
+ \left[ \frac{\partial^2 f}{\partial x_1 \partial x_2}(y_1, y_2) \right] (x_1 - y_1)(x_2 - y_2)
\]

\[
+ \frac{1}{2} \left[ \frac{\partial^2 f}{\partial x_1^2}(y_1, y_2) \right] (x_1 - y_1)^2
\]

\[
+ \frac{1}{2} \left[ \frac{\partial^2 f}{\partial x_2^2}(y_1, y_2) \right] (x_2 - y_2)^2 + \ldots
\]

2nd-order terms

But \( f \) was scalar-valued. What is the vector version of this Taylor expansion?

\[\hat{f}(\hat{x}) = \hat{f}(\hat{y}) + \nabla \hat{f}(\hat{y}) (\hat{x} - \hat{y}) + (\hat{x} - \hat{y})^T \nabla^2 \hat{f}(\hat{y}) (\hat{x} - \hat{y}) + \ldots\]
when \( D \) is the Jacobian (also sometimes called the derivā of \( \vec{f} \)) and \( H \) is the Hessian of \( \vec{f} \), involving 2nd derivative; (it is a 3rd order tensor).

\[
\begin{pmatrix}
\frac{df_1}{dx_1} & \frac{df_1}{dx_2} \\
\frac{df_2}{dx_1} & \frac{df_2}{dx_2} \\
\end{pmatrix}
\]

\( Df \) is a matrix

\( Hf \) is not a matrix, but a tensor.

Drop the \( H \) term, then we have:

\[
\vec{f}(\vec{x}) \approx \vec{f}(\vec{y}) + D\vec{f}(\vec{y})(\vec{x}-\vec{y})
\]

At \( \vec{f}(\vec{x}) = \vec{0} \), we have that (i.e. solve for \( \vec{x} \))

\[
\vec{x} \approx \vec{y} - (D\vec{f}(\vec{y}))^{-1}\vec{f}(\vec{y})
\]

the inverse of the Jacobian, evaluated at \( \vec{y} \).
Therefore Newton's method in higher dimensions is given by:

\[ \hat{x}_{k+1} = \hat{x}_k - (D\hat{f}(\hat{x}))^{-1} \hat{f}(\hat{x}) \]

and convergence is quadratic, assuming that 

\( D\hat{f} \) is invertible, given by

\[ \| \hat{x} - \hat{x}_{k+1} \| \approx \| \hat{x} - \hat{x}_k \|^2 \]

\( \| \cdot \| \) denotes the usual Euclidean norm.

The above formula is the same for n-dimensional root finding:

\[ f(\hat{x}) = 0 \]

\[
\begin{pmatrix}
    f_1(x_1, x_2, \ldots, x_n) \\
    f_2(x_1, x_2, \ldots, x_n) \\
    \vdots \\
    f_n(x_1, x_2, \ldots, x_n)
\end{pmatrix} = \begin{pmatrix}
    0 \\
    0 \\
    \vdots \\
    0
\end{pmatrix}
\]
All of these methods for solving \( f(x) = 0 \) have the following in common: the true root \( \delta \) is a fixed-point of the iteration.

Ex: Newton's Method:

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

If \( x_k = \delta \), then

\[
x_{k+1} = \delta - \frac{f(\delta)}{f'(\delta)} = \delta - \frac{0}{f'(\delta)} = \delta
\]

So \( x_k = \delta \) \( \Rightarrow \) \( x_{k+1} = \delta \).

This way of thinking can be generalized to the question: Are there any values of \( x \) such that for some function \( g \), \( g(x) = x \)?

\( \Rightarrow \) Examine the iteration \( x_{k+1} = g(x_k) \).
Graphically:

\[ y = x \]

Fixed points when \( g(x) = x \).

Note that solving \( f(x) = 0 \) is equivalent to finding a fixed point of \( g(x) = f(x) + x \).

Since \( g(x) = x \)

\[ \Rightarrow f(x) + x = x \]

Examples: Newton's Method looks for a fixed point of the function

\[ g(x) = x - \frac{f(x)}{f'(x)} \]
Definition: A simple iteration is given by
\[ x_{n+1} = g(x_n) \]
when we assume \( g \) is continuous on an interval \([a,b]\), and that \( g(x) \in [a,b] \) for all \( x \in [a,b] \) (this condition guarantees the existence of a fixed point by the following theorem):

Brouwer's Fixed Point Theorem: If \( g \) is as above, then there exists \( z \in [a,b] \) such that \( z = g(z) \).

Proof: If \( g(x) \in [a,b] \) for all \( x \in [a,b] \), then it is easy to show that \( g(x) - x = h(x) \) has at least one root in \([a,b]\); show that \( h(a)h(b) < 0 \).
Next, without looking at the range of \(g\) on \([a,b]\), examine how it maps nearby points:

**Definition (Contraction)** Let \(g\) be continuous on \([a,b]\). The function \(g\) is a contraction on \([a,b]\) if there exists a number \(L\) with \(0 < L < 1\) such that

\[
|g(x) - g(y)| < L|x-y| \quad \text{for all } x, y \in [a,b].
\]

This means that \(g\) maps points to values which are closer together.

If \(L\) is allowed to be any positive number, this is known as a Lipschitz condition.