# Newton's Method: details and variants

Thursday, September 10, 2020 9:35 PM

Newton's method has been around awhile, so there are many variants/extensions etc.

Outline:
- History/example
- Convergence revisited (multiple roots)
  - Modified Newton
  - Deflation
- Practical Newton's method
- Secant Method
- Pros/Cons of Newton

Ex. Babylonian Algo

2000+ years ago, Archimedes claimed $\frac{265}{153} < \sqrt{3} < \frac{1351}{780}$. Not bad!

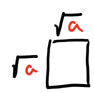
1.73202     1.73205

How did he find it? Not sure, but maybe via the Babylonian Algorithm, aka Heron's method.

Algo: Find $x = \sqrt{a}$

Iterate $x_{n+1} = \frac{1}{2}\left(x_n + \frac{a}{x_n}\right)$

motivation:




Square w/ area $a$


rectangle with same area $a$. If one side is $x$, other side must be $a/x$.

If $x > a$ then $a/x < a$ and vice-versa (one side too short $\Rightarrow$ other side too long)

So we have under and over approximations... so average these

It turns out this is Newton's method applied to $f(x) = x^2 - a$ (i.e. $f'(x) = 2x$)

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$
$$= x_n - \frac{(x_n^2 - a)}{2x_n}$$
$$= x_n - \tfrac{1}{2}x_n + a/2x_n = \tfrac{1}{2}(x_n + a/x_n)$$

## Convergence, revisited

Recall our result from last time:

Thm: (combining Thm 2.6 and Thm 2.9)

Let $f \in C^2([a,b])$ have a root $p \in (a,b)$ with multiplicity 1 (i.e. $f'(p) \neq 0$) then if initialized sufficiently close to $p$, Newton's method will converge to $p$.

Furthermore, if additionally $|g''(x)|$ is bounded on some open interval around $p$, then the convergence rate is quadratic. $\longrightarrow$ $g(x) := x - \frac{f(x)}{f'(x)}$

Ok, but what if $p$ isn't a simple root? — aka multiplicity 1

**Def** Recall, a root $p$ of $f$ is $\boxed{\text{multiplicity}}$ $m$ if the 1st $m-1$ derivatives of $f$ are $0$ at $p$

i.e., $f(p) = f'(p) = \ldots = f^{(m-1)}(p) = 0$ and $f^{(m)}(p) \neq 0$

Equivalently, if $f(x) = (x-p)^m g(x)$ and $g(p) \neq 0$

Ex: $f(x) = x(x-1)(x-2)$ has a simple root at $x=0$ (and at $x=1, x=2$) $m=1$
$f(x) = x^2(x-1)$ has a double root at $x=0$ $m=2$
$f(x) = x^3$ has a triple root at $x=0$ $m=3$

Our theorem doesn't apply. Often we still get convergence but just not at a quadratic rate

Ex $f(x)=x^2$, $x=0$ is a double root. $f'(x)=2x$

Newton's method is $x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)} = x_n - \dfrac{x_n^2}{2x_n} = \frac{1}{2}x_n$

ie., $x_{n+1} = \frac{1}{2}x_n$,
so, e.g., $x_0 = 1$, then we iterate $1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \ldots$

error at step $n$ is $\frac{1}{2^n}$, $e_n = \frac{1}{2^n}$. Is this quadratic?

Check: $\displaystyle\lim_{n\to\infty} \frac{e_{n+1}}{e_n^2} = \lim \left( \frac{\frac{1}{2^{n+1}}}{\left(\frac{1}{2^n}\right)^2} = \frac{\frac{1}{2^{n+1}}}{\frac{1}{2^{2n}}} = \frac{2^{2n}}{2^{n+1}} = 2^{n-1} \right) = \infty$

No, not quadratic conv.

In fact, this is linear convergence since it fits the form $\rho^n$ $(\rho = \frac{1}{2})$ which we're already discussed.

One fix to this multiplicity issue is...

Modified* Newton's Method

* there are many ways to modify it, this is just our book's notation

Let $p$ be a root of $f$ w/ multiplicity $m$ ($m=1$ is ok, but mostly interested in $m>1$)

Define $h(x) = \dfrac{f(x)}{f'(x)}$. Then claim $p$ is a root of $h$ also

proof: $m=1$ then $f'(p) \neq 0$ so immediately $h(p) = 0$

$m>1$ then $f'(p) = 0$, $\dfrac{f(p)}{f'(p)} = \dfrac{0}{0}$ ... use L'Hôpital

So $\displaystyle\lim_{x\to p} \frac{f(x)}{f'(x)} = \lim_{x\to p} \frac{f'(x)}{f''(x)} = \ldots = \lim_{x\to p} \frac{f^{(m-1)}(x)}{\underbrace{f^{(m)}(x)}_{\neq 0 \text{ at } p}} = 0$

Furthermore, $p$ is a simple root of $h$

proof:
Can write $f(x) = (x-p)^m g(x)$
w/ $g(p) \neq 0$

then
$$h(x) = \frac{(x-p)^m g(x)}{m(x-p)^{m-1} g(x) + (x-p)^m g'(x)} = (x-p) \frac{g(x)}{m \cdot g(x) + (x-p) g'(x)} \qquad \tilde{g}(p) = \frac{g(p)}{\phantom{x}},$$

$$\underbrace{\qquad\qquad}_{\tilde{g}(x)},$$

$m\,g(p) + (p-p)g'(p)$

$= 1/m \neq 0$

So $h(x) = (x-p)\tilde{g}(x)$ with $\tilde{g}(p) \neq 0$ $\Rightarrow$ is a simple root

... back to the point.

$p$ is a simple root of $h(x) = \dfrac{f(x)}{f'(x)}$

So run Newton on $h$ instead of $f$.

Simplifying $h'$, we get **MODIFIED NEWTON** ugly quotient rule stuff

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

Not a perfect fix:

DRAWBACKS: ① must compute $f''$

② subtractive cancellations

Note: if $m$ is known, $x_{n+1} = x_n - m\dfrac{f(x_n)}{f'(x_n)}$ will work and w/ quadratic convergence

## Deflation

A related idea to dealing with roots of multiplicity $> 1$

Suppose $f$ has simple roots at $p_1$ and $p_2$ (if $p_1 = p_2$ it's a double root)

If $|p_1 - p_2|$ is very small, starts to act like a double root, especially w/ roundoff errors. The condition number of the root-finding problem is large

One practical consequence:

Suppose we find $p_1$. How to find $p_2$? We have to start sufficiently close to it, which is hard since we don't know where it is! We might get "sucked into" the $p_1$ root.

... and a fix: deflation

Define $h(x) = \dfrac{f(x)}{x - p_1}$ so $h$ doesn't have a root at $p_1$, but still has a root at $p_2$

(this is also the name for a broader class of techniques, eg. in eigenvalue problems)

## Practical Newton's method, ie. globalization strategies

We won't go into details

(1) combine w/ bracketing or another root-finding method

(our book mentions a special version called False Position / Regula Falsi)

(2) safeguarding / linesearch:

don't let $x_{n+1}$ go too far

(ex: if we must keep $x_n > 0$)

or $x_{n+1} = x_n - \alpha\dfrac{f(x_n)}{f'(x_n)}$, $\alpha \le 1$

Need $\alpha = 1$ for quadratic convergence but often take $\alpha < 1$ for the first few iterations

Don't worry about these, just use Matlab / Scipy libraries

# Secant Method

Idea: Newton's method is $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$. What if we want to avoid calculating $f'(\,)$? *

*See future lecture on "Automatic Differentiation" ($\neq$ symbolic diff.) ex. backprop.

Well, rule-of-thumb: when a step produces an approximate result, you are free to carry it out approximately.

Since Newton's method was derived via Taylor Series, ignoring higher-order terms, lets try approximating the derivative:

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{(x+h) - (x)}, \text{ ie, } f'(x_n) \approx \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}} \text{ if } |x_n - x_{n-1}| \text{ small}$$

(re-use our old computation)

So SECANT METHOD

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

Only needs 1 function-evaluation per step, no derivative needed. Nice!

You can extend in 2 ways:

1. use more previous points, "inverse interpolation". Not very common
2. In multi-dimensional problems, there is "more freedom", and we actually have a whole class of QUASI-NEWTON METHODS. Very useful! often "state-of-the-art"
(names like "Broyden class", "SR1", "BFGS")

Also, some extra computational savings that are irrelevant for scalar problems.

## Convergence Analysis of Secant method

Recall for Newton $\lim \frac{|e_{n+1}|}{|e_n^2|} < \infty$ i.e. $\alpha = 2$

where $e_n = p - x_n$ is the error.

For the secant method, assuming $e_n$ is small, we can do a Taylor expansion of our secant method iteration (tedious but straightforward) to get

$$e_{n+1} \approx \underbrace{-\tfrac{1}{2} \frac{f''(p)}{f'(p)}}_{\text{some constant}} e_n e_{n-1} \qquad (*)$$

Let's guess/hope that we have $\alpha$ convergence and can write

$(\forall n)$ $\underbrace{e_{n+1} = c \cdot e_n^{\alpha}}_{\text{like an "ansatz"}}$ and solve for $\alpha$

then $e_{n+1} = c \cdot e_n^{\alpha} = c \cdot (c \cdot e_{n-1}^{\alpha})^{\alpha} = \text{const} \cdot e_{n-1}^{\alpha^2}$

and $e_n e_{n-1} = c \cdot e_{n-1}^{\alpha} e_n = c \cdot e_{n-1}^{\alpha+1}$

Plugging into $(*)$ gives $e_{n-1}^{\alpha^2} = (\text{some constant})\, e_{n-1}^{\alpha+1}$ — involves $c$ and $-\frac{1}{2}\frac{f''(p)}{f'(p)}$

this should be true for all $e_{n-1}$ (when $e_{n-1}$ is near 0)

So need (constant) = 1

and $\boxed{\alpha^2 = \alpha + 1,}$ $\alpha > 0$ → Solution is the Golden Ratio

$$\alpha = \frac{1+\sqrt{5}}{2} \approx 1.618$$

So... Newton's method has rate $\alpha = 2$

Secant method has rate $\alpha = 1.62$

... and in fact if you say that Newton's method takes twice as much work as the secant method (since you must evaluate $f(x_n)$ and $f'(x_n)$) then we can do 2 iterations of the secant method and count the rate as $(1.62)^2 = 2.62$

(or, keep as 1.62 but call Newton's rate $\sqrt{2} \approx 1.41$)

... meaning the secant method is better than Newton, in this sense.

## Summary of Pros / Cons

**Newton's method:**

PROS

+ No need for bracketing interval $[a,b]$. Doesn't need $f(a)\cdot f(b) < 0$ (which excludes $f(x)=x^2$)
+ Very fast convergence eventually (the gold-standard)
+ Simple

CONS

- $x_0$ must be close to root, and hard to know how close. If not close enough, may diverge or converge to wrong root
- Slower convergence for multiple roots $m > 1$. Some fixes but not perfect
- practical implementations need more information, more complicated
- must supply $f'(\ )$

**Secant method**

same pros/cons except no longer need to provide $f'(\ )$

## Addendum

Theorems 2.6 and 2.8 in the book are not stated in the most useful way, especially 2.8. Notation: let $p$ denote a fixed pt.

Book's Thm 2.8: $g \in C[a,b]$, $g(x) \in [a,b]\ \forall x \in [a,b]$, $g'$ continuous on $(a,b)$, and $|g'(x)| \le k\ \forall x \in (a,b)$ for some $k < 1$.

Then unless $p_0$ is a fixed point, the sequence $(p_n)$, defined by $p_{n+1} = g(p_n)$, converges only linearly if $g'(p) \neq 0$.

(They state it this way since Thm 2.9 gives quadratic convergence if $g'(p) = 0$)

⚠ First, recall

1. root-finding, $f(p) = 0$, here $f'(p) = 0$ is BAD
2. fixed-pt. iter, eg. Newton

We can strengthen Thm 2.8 by looking at its proof:

Better Thm 2.8 (same assumptions ...)

... then if $g'(p) \neq 0$, $(p_n)$ converges to $p$ linearly at rate $|g'(p)|$.

$$g(x) = x - \frac{f(x)}{f'(x)}, \quad g(p) = p$$

$$g'(x) = 1 - \frac{f'(x)f'(x) - f(x)f''(x)}{f'(x)^2}$$

$$= \frac{f(x)f''(x)}{(f'(x))^2}$$

$g'(p) = 0$ is GOOD