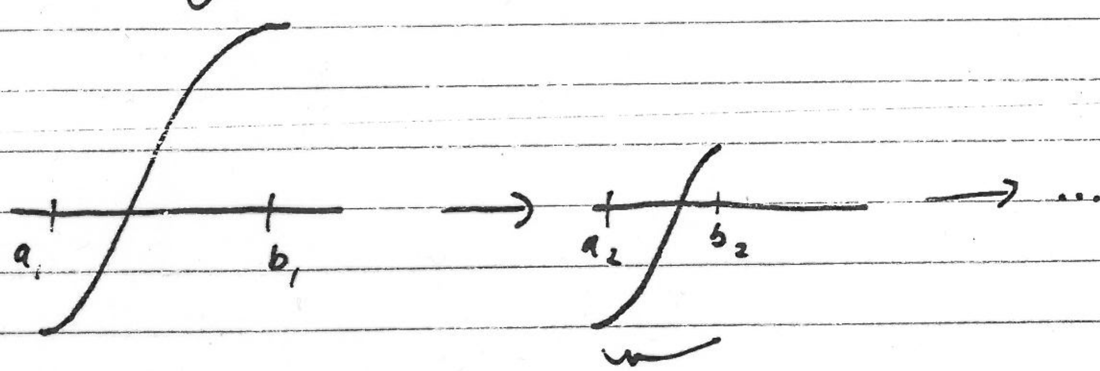
8.1

Both Newton's Method and Secant Method have the limitation that they may diverge when the initial guesses are not sufficiently close to the root.

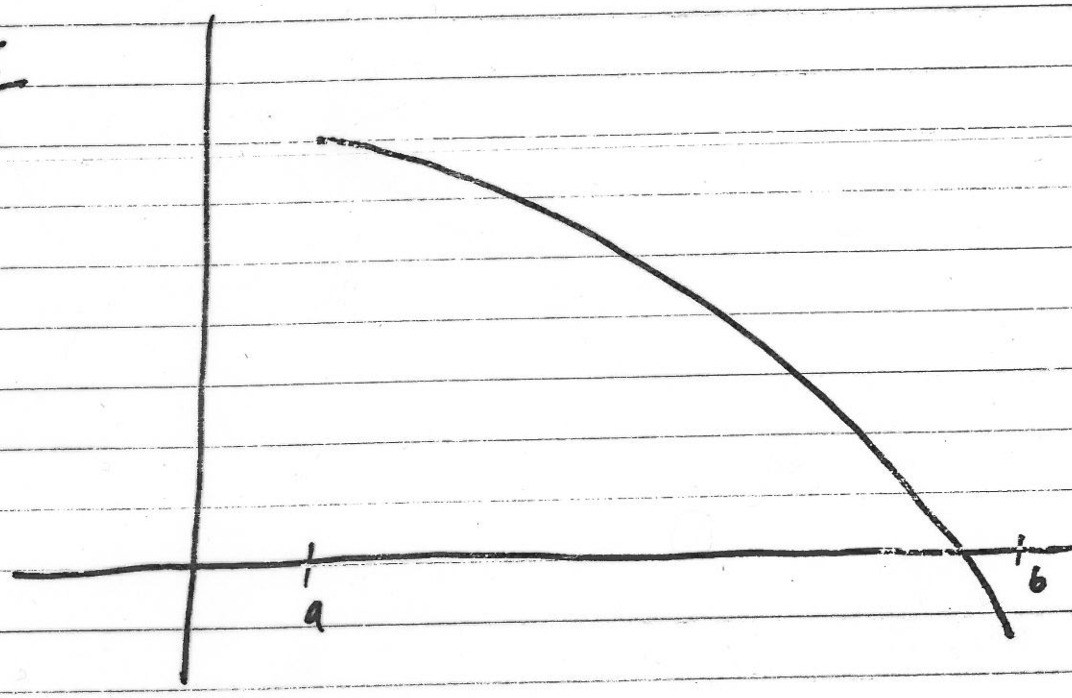
However Bisection used the idea of bracketing the root at each step to ensure convergence...



root guaranteed to be in the interval (a_2, b_2) by the Intermediate Value Thm.

If instead of considering a midpoint approximation for the root, we use a secant approximation for the root (based on the endpoints), then we get the Method of False Position.

Ex



Method of False Position

INPUT: initial approximations p_0 and p_1
tolerance TOL
maximum number of iterations N_0

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

Step 2 : While $i \le N_0$ do steps 3-7

Step 3: Set $p = p_1 - q_1 \dfrac{p_1 - p_0}{q_1 - q_0}$

Step 4: If $|p - p_1| < TOL$ then

Output (p)
Stop.

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

Step 6: If $q \cdot q_1 < 0$ then set
$p_0 = p_1$
$q_0 = q_1$

Step 7: Set $p_1 = p$, $q_1 = q$

Step 8: Output ('Method Failed')

- This method nicely illustrates how bracketing a root can be used to develop a more sophisticated root finding method.
- In terms of performance, the Method of False Position is often slightly slower than secant method (it's like paying for some extra insurance in finding the root). It is often (but not always) faster than bisection. (Can you think of an example where bisection is faster?)

Error Analysis

We want to be able to give a more precise description of how a method converges to the solution.

For example, consider finding a root for the polynomial

$$x^3 + 4x^2 - 10 = 0$$

with two different methods (A and B). Suppose that the errors produced by these methods are as given below

1.36552....

1.5

	Method A	Method B
$\lvert p - p_0 \rvert$	0.134769987	0.134769987
$\lvert p - p_1 \rvert$	0.078276245	0.008103332
$\lvert p - p_2 \rvert$	0.037310791	0.000003811
$\lvert p - p_3 \rvert$	0.019771639	0.000000000 to all significant digits
$\lvert p - p_4 \rvert$	0.009940240	

⋮

Notice that the error for Method A decreases by a constant factor (about 2) at each iteration.

For Method B, the error drops off much more quickly — Indeed the error at step n is roughly proportional to the error at step $(n-1)$ squared.

Both of these behaviors can be quantified.

<u>Defn</u> Suppose $\{p_n\}_{n=0}^{\infty}$ is a sequence that converges to p, with $p_n \neq p$ for all n. If positive constants λ and α exist with $\lim_{n\to\infty} \frac{|p_{n+1}-p|}{|p_n-p|^{\alpha}} = \lambda$ then $\{p_n\}_{n=0}^{\infty}$ converges to p of order α, with asymptotic error constant λ.

Notice that

- a sequence with a high order of convergence, converges more rapidly than a sequence with a lower order.

- the constant λ affects the speed of convergence, but is not as important as the order.

There are 2 very common cases:

- If $\alpha = 1$, the sequence is linearly convergent
- If $\alpha = 2$, the sequence is quadratically convergent.

Notice that if $(N - N_0)$ is large & $\alpha = 1$ then

$$|p_N - p| \approx \lambda |p_{N-1} - p| \approx \lambda^2 |p_{N-2} - p| \approx \cdots \approx \lambda^{N_0} |p_{N-N_0} - p|$$

So the error drops off by roughly a constant factor of λ at each step (LIKE METHOD A)

On the other hand if $\alpha = 2$ then

$$
\begin{aligned}
|p_N - p| &\approx \lambda |p_{N-1} - p|^2 \\
&\approx \lambda \left[\lambda |p_{N-2} - p|^2\right]^2 \\
&= \lambda^3 |p_{N-2} - p|^4 \\
&\vdots \\
&\approx \lambda^{2^{N_0}-1} |p_{N-N_0} - p|^{2^{N_0}}
\end{aligned}
$$

So the error drops off much more quickly — at each step n, the error is proportional to the error at step $(n-1)$ squared (This is like method B)

We now consider the convergence of fixed point iterations ...

Thm (2.7 of Text)

Let $g \in C[a,b]$ be such that $g(x) \in [a,b]$ for all $x \in [a,b]$.

Suppose, in addition, that g' is continuous on (a,b) and a positive constant $K<1$ exists with $|g'(x)| \le K$ for all $x \in (a,b)$.

If $g'(p) \neq 0$, then for any number p_0 in $[a,b]$ the sequence

$$p_n = g(p_{n-1}) \quad \text{for } n \ge 1$$

converges only linearly to the unique fixed point p in $[a,b]$.

<u>Proof</u>. We know from the fixed point thm that the sequence converges to p.

Since g' exists on $[a,b]$ we can apply the Mean Value Thm to g:

$$\underbrace{g(p_n) - g(p)}_{p_{n+1} - p} = g'(\xi_n)(p_n - p)$$

where ξ_n is between p_n & p.

Thus $$\frac{p_{n+1} - p}{p_n - p} = g'(\xi_n)$$

and
$$\lim_{n\to\infty} \frac{p_{n+1} - p}{p_n - p} = \lim_{n\to\infty} g'(\xi_n)$$
$$= g'\left(\lim_{n\to\infty} \xi_n\right) \quad \text{by continuity of } g'$$
$$= g'(p) \quad \text{since } \xi_n \xrightarrow[n\to\infty]{} p$$

Thus $$\lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_n - p|} = |g'(p)|$$

and fixed point iteration gives linear convergence with asymptotic error constant $|g'(p)|$ whenever $g'(p) \neq 0$

Method A was the fixed point iteration defined by the iteration function

$$g(x) = \frac{1}{2}(10 - x^3)^{1/2}$$

Notice that $g'(p = 1.365230013)$

$$= -\frac{3}{4}x^2(10-x^3)^{-1/2}\Big|_p$$

$$\approx -.51 \neq 0$$

so the theorem applies if we consider the interval $[1, 1.5]$ & we see that linear convergence is obtained.

On the other hand, Method B was the fixed point iteration defined by the iteration function

$$g(x) = x - \frac{x^3+4x^2-10}{3x^2+8x}$$

This method gave quadratic convergence, but the theorem cannot be applied because

$$g'(p) = 0$$

(check on your own)

We saw last day that higher order convergence for fixed point methods can occur only when $g'(p) = 0$. It is possible under certain reasonable conditions to obtain **quadratic** convergence...

Thm (2.8 of Text)

Let p be a solution of the equation $x = g(x)$.

Suppose $g'(p) = 0$ and g'' is continuous and strictly bounded by M on an open interval I containing p.

Then there exists a $\delta > 0$ such that for $p_0 \in [p-\delta, p+\delta]$ the sequence defined by $p_n = g(p_{n-1})$ when $n \geq 1$ converges at least quadratically to p.

Moreover for sufficiently large values of n,

$$|p_{n+1} - p| < \frac{M}{2} |p_n - p|^2$$

Idea of Proof (Main Steps)

① We choose K in $(0,1)$ and $\delta > 0$ such that on the interval $[p-\delta, p+\delta]$ contained in I, we have $|g'(x)| \le K$ and g'' continuous.

② We then show g maps $[p-\delta, p+\delta]$ into itself.

③ The function g is expanded in a Taylor series for $x \in [p-\delta, p+\delta]$

$$g(x) = g(p) + \cancelto{0}{g'(p)}(x-p) + \frac{g''(\xi)}{2}(x-p)^2$$

where ξ lies between x & p.

Now set $x = p_n$

$$\therefore \quad g(p_n) = g(p) + \frac{g''(\xi_n)}{2}(p_n - p)^2$$

$$p_{n+1} - p = \frac{1}{2}\, g''(\xi_n)(p_n - p)^2$$

$$\frac{|p_{n+1} - p|}{(p_n - p)^2} = \frac{1}{2}|g''(\xi_n)|$$

$$\lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_n - p|^2} = \lim_{n\to\infty} \frac{1}{2}|g''(\xi_n)|$$

$$= \frac{1}{2}\left|g''\left(\lim_{n\to\infty} \xi_n\right)\right|$$

g'' is continuous

$$= \frac{1}{2}|g''(p)|$$

$p_n \xrightarrow[\infty]{n} p$ by the Fixed pt' thm and ① and ②. Since ξ_n lies between p_n and p we have $\xi_n \xrightarrow[\infty]{n} p$.

Thus the sequence $\{p_n\}_{n=0}^{\infty}$ is quadratically convergent if $g''(p) \neq 0$ and higher order convergent if $g''(p) = 0$.

Also we know $|g''| < M$ so

$$|p_{n+1} - p| < \frac{M}{2}|p_n - p|^2 .$$

So the idea behind finding iteration methods with a high order of convergence is to look for schemes whose derivatives are zero at the fixed point.

E.g. Newton's Method

$$g(x) = x - f(x)/f'(x)$$

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

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

$\therefore\ g'(p) = 0$ provided $f'(p) \neq 0$

$\therefore$ Newton's Method satisfies the derivative condition.

Let's take another look at Newton's Method:

Consider using Newton's Method to find the roots of

$$p^3 - p^2 - p + 1 = 0.$$

Newton's method here is

$$p_{n+1} = p_n - \frac{p_n^3 - p_n^2 - p_n + 1}{3p_n^2 - 2p_n - 1}$$

Starting from $p_0 = 1.1$ we find

p_0	1.1
p_1	1.05116...
p_2	1.02589...
p_3	1.01303...
p_4	1.00653...
p_5	1.00327...
$\vdots$	

Which is very slow (LINEAR!) convergence to the root (which is 1).

Why is this?

In Newton's Method we need $f'(p) \neq 0$ to obtain quadratic convergence.

Notice that

$$f'(p) = 3p^2 - 2p - 1\Big|_{p=1} = 0$$

so the theorem doesn't hold.

Moreover, factoring f

$$f(x) = (x-1)^2(x+1)$$

we see that $x = 1$ is a zero of multiplicity 2 ...

<u>Defn</u>: A solution p of $f(x) = 0$ is a zero of multiplicity m of f if for $x \neq p$ we can write $f(x) = (x-p)^m q(x)$ where $\lim_{x \to p} q(x) \neq 0$.

Simple zeros are those that have multiplicity one.

Thus Newton's Method can only be applied to simple zeros of a function.

Identification of the multiplicity of a zero is often made easier by the two following Thms:

Thm (2.10 of Text)

$f \in C^1[a, b]$ has a simple zero at p in (a, b) if and only if $f(p) = 0$ but $f'(p) \neq 0$.

Thm (2.11 of Text)

The function $f \in C^m[a, b]$ has a zero of multiplicity m at p if and only if

$$0 = f(p) = f'(p) = f''(p) = \cdots = f^{(m-1)}(p)$$

but $f^{(m)}(p) \neq 0$.

We want to obtain quadratic convergence with Newton's Method for multiple roots.

One approach is to define a function

$$\mu(x) = f(x)/f'(x)$$

We assume p is a zero of multiplicity m and $f(x) = (x-p)^m q(x)$ where $q(p) \neq 0$.

Then $\mu(x) = \dfrac{(x-p)^m q(x)}{m(x-p)^{m-1} q(x) + q'(x)(x-p)^m}$

$$= \frac{(x-p)\, q(x)}{m q(x) + q'(x)(x-p)}$$

So $\mu(p) = 0$, but $\dfrac{q(p)}{m q(p) + q'(p)(p-p)} = \dfrac{1}{m} \neq 0$

and p is a zero of multiplicity 1 of $\mu(x)$.

Substituting $\mu(x)$ into Newton's Method gives the iteration function

$$g(x) = x - \frac{\mu(x)}{\mu'(x)} = x - \frac{f(x)\, f'(x)}{[f'(x)]^2 - f(x)\, f''(x)}$$

Advantages: Provided g satisfies the necessary continuity conditions, we will get quadratic convergence regardless of the multiplicity of the zero of f.

Disadvantages:
- Need f''
- More calculations to evaluate g
- possibility of serious cancellation in the denominator.

Back to finding the roots of

$$p^3 - p^2 - p + 1 = 0$$

Apply

	Newton's Method	Modified Newton's
p_0	1.1	1.1
p_1	1.05116…	0.997735…
p_2	1.02589…	0.999999…
p_3	1.01303…	
p_4	1.00653…	
p_5	1.00327…	

Linear convergence.

Once again, we find the desired quadratic convergence

Convergence Speed and Acceleration

Suppose that we are given a linearly convergent sequence, and that we want to speed up convergence.

We want to analyze the behaviour of the error and use this knowledge to greatly reduce the error.

For example, we saw last day that the iteration method $p_{n+1} = g(p_n)$ with $g(x) = \frac{1}{2}\sqrt{10 - x^3}$

gives only linear convergence to the limit $p = 1.3652\ldots$

$p_0 = 1.5$

$p - p_0$	$-0.13476\ldots$
$p - p_1$	$+0.07827\ldots$
$p - p_2$	$-0.03710\ldots$
$p - p_3$	$+0.01977\ldots$
$p - p_4$	$-0.00994\ldots$
$\vdots$	

Notice that the ratio of the errors is fairly constant — we can use this idea to accelerate the convergence of the method.

Suppose $\dfrac{p_{n+1}-p}{p_n-p} \approx \dfrac{p_{n+2}-p}{p_{n+1}-p}$

$$(p_{n+1}-p)^2 \approx (p_n-p)(p_{n+2}-p)$$

$$p_{n+1}^2-2p_{n+1}p+p^2 \approx p_n p_{n+2}-p_n p-p p_{n+2}+p^2$$

$$(p_{n+2}-2p_{n+1}+p_n)p \approx p_n p_{n+2}-p_{n+1}^2$$

$$\therefore \quad p \approx \frac{p_n p_{n+1}-p_{n+1}^2}{p_{n+2}-2p_{n+1}+p_n}$$

<u>or</u> $$p \approx p_n - \left(\frac{p_{n+1}^2-2p_n p_{n+1}+p_n^2}{p_{n+2}-2p_{n+1}+p_n}\right)$$

(derived by adding & subtracting p_n to rhs)

The corresponding sequence

$$\hat{p}_{n+1} = p_n - \left[\frac{p_{n+1}^2-2p_n p_{n+1}+p_n^2}{p_{n+2}-2p_{n+1}+p_n}\right]$$

is known as Aitken's Method.

Applying Aitken's Method to our previous sequence

Fixed pt itn with $g(x) = \frac{1}{2}\sqrt{10-x^2}$		Aitken's Method	
$p-p_0$	$-0.13476\cdots$	$p-\hat{p}_0$	0.00090088
$p-p_1$	$+0.07827\cdots$	$p-\hat{p}_1$	0.00023088
$p-p_2$	$-0.03731\cdots$	$p-\hat{p}_2$	0.00006107
$p-p_3$	$+0.01977\cdots$	$p-\hat{p}_3$	0.00001592
$p-p_4$	$-0.00994\cdots$	$p-\hat{p}_4$	0.00000418

This is much faster...

Indeed it can be shown that the following theorem holds:

Thm (2.13 of txt)

Suppose the $\{p_n\}$ is a sequence that converges linearly to the limit p and that for all sufficiently large values of n we have $(p_n - p)(p_{n+1} - p) > 0$.

Then the sequence $\{\hat{p}_n\}_{n=0}^{\infty}$ converges faster than $\{p_n\}_{n=0}^{\infty}$ to p in the sense that $\lim_{n\to\infty} \frac{\hat{p}_n - p}{p_n - p} = 0$.

The theorem does not apply to alternating sequences, but as we saw from our example it is often very useful even there for accelerating convergence.

Finally we remark that this method is often written using difference operators to simplify the notation

$$\Delta x_n = x_{n+1} - x_n$$

$$\Delta^2 x_n = \Delta(\Delta x_n) = x_{n+2} - 2x_{n+1} + x_n$$

Then Aitken's Method reads as

$$\hat{p}_{n+1} = p_n - \frac{(\Delta p_n)^2}{\Delta^2 p_n}$$

We saw Aitken's Method for accelerating the convergence of a linearly convergent sequence:

$$\hat{p}_{n+1} = p_n - \frac{(p_{n+1} - p_n)^2}{p_{n+2} - 2p_{n+1} + p_n}$$

The acceleration becomes even more effective when we restart the iteration with the improved value as soon as one becomes available:

Set
$$q_{n+1} = g(p_n)$$
$$q_{n+2} = g(q_{n+1})$$
$$p_{n+1} = p_n - \frac{(q_{n+1} - p_n)^2}{q_{n+2} - 2q_{n+1} + p_n}$$

This is the iteration used in

Steffansen's Method.

Steffensen's Method gives quadratic convergence without evaluating a derivative:

<u>Thm</u>: Suppose that $x = g(x)$ has the solution p with $g'(p) \neq 1$. If there exists a $\delta > 0$ such that $g \in C^3[p-\delta, p+\delta]$ then Steffensen's method gives quadratic convergence for any $p_0 \in [p-\delta, p+\delta]$.

Zeros of Polynomials

We want to compute the zeros of polynomials.

A polynomial of degree n has the form

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

where the a_i's are constants and $a_n \neq 0$.

Some key properties of polynomials:

<u>Fundamental Theorem of Algebra</u>

If P is a polynomial of degree $n \geq 1$ then $P(x)$ has at least 1 (possibly complex) root.

<u>Corollary</u>: If $P(x)$ is a polynomial of degree $n \geq 1$ then there exists unique constants $x_1, x_2, \ldots, x_k$ possibly complex, and positive integers $m_1, m_2, \ldots, m_k$ such that

$$\sum_{i=1}^{k} m_i = n \quad \text{and} \quad P(x) = a_n (x-x_1)^{m_1} (x-x_2)^{m_2} \cdots (x-x_k)^{m_k}$$

Corollary: Let P and Q be polynomials of degree at most n. If $x_1, \ldots, x_k$ with $k > n$ are distinct numbers with $P(x_i) = Q(x_i)$ for $i = 1, 2, \ldots, k$ then $P(x) = Q(x)$ for all x.

We want to use Newton's Method to locate the approximate zeros of P. It will be necessary to evaluate P and its derivative at specified values.

We now direct our attention to efficient methods for this task.

The idea is to use nesting to evaluate an arbitrary n^{th} degree polynomial using only

n multiplications and
n additions.

For illustration consider $n = 4$.

To evaluate $P(x) = a_4 x^4 + a_3 x^3 + a_2 x^2 + a_1 x + a_0$

we write

$$P(x) = \underbrace{(\underbrace{(\underbrace{(\underbrace{(\underbrace{a_4}_{b_4} x + a_3)}_{b_3} x + a_2)}_{b_2} x + a_1)}_{b_1} x + a_0}_{b_0}$$

Algorithmically

Set $b_4 = a_4$

Set $b_3 = b_4 x + a_3$

Set $b_2 = b_3 x + a_2$

Set $b_1 = b_2 x + a_1$

Set $b_0 = b_1 x + a_0$

Now b_0 gives the value $P(x)$.

For general polynomials of degree n:

$$b_n = a_n$$

$$b_k = a_k + b_{k+1} x \qquad 0 \le k \le n-1$$

and $b_0 = P(x)$.

We also want $P'(x)$. Differentiate each of the steps listed above, keeping in mind that b_j is a function of x:

$$b_n' = 0$$

$$b_k' = b_{k+1}' x + b_{k+1}$$

and $b_0' = P'(x)$.

Relabel: $c_{n+1} = b_n'$
Then an efficient method for computing $P'(x)$ is

$$c_n = a_n$$
$$c_k = c_{k+1} x + b_k$$

and then $c_1 = P'(x)$

Writing this as pseudocode

Horner's Method for evaluating a polynomial

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

and its derivative at x_0.

INPUT: n, a_j, x_0 $\quad 0 \leq j \leq n$

OUTPUT: $y = P(x_0)$

$z = P'(x_0)$.

STEP 1: Set $y = a_n$

$z = a_n$

STEP 2: For $j = n-1, n-2, \ldots, 1$

Set $y = x_0 y + a_j$

$z = x_0 z + y$

STEP 3: Set $y = x_0 y + a_0$

STEP 4 Output (y, z).

STOP.

Horner's Method has another useful property

Consider $P(x)$

$$= a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

$$= b_n x^n + (b_{n-1} - b_n x_0) x^{n-1} + \cdots + (b_1 - b_2 x_0) x + (b_0 - b_1 x_0)$$

$$= (b_n x^{n-1} + b_{n-1} x^{n-2} + \cdots + b_2 x + b_1) x$$

$$- (b_n x^{n-1} + b_{n-1} x^{n-2} + \cdots + b_2 x + b_1) x_0$$

$$+ b_0$$

$$= \underbrace{(b_n x^{n-1} + b_{n-1} x^{n-2} + \cdots + b_2 x + b_1)}_{Q(x)} (x - x_0) + b_0$$

This property is useful because it gives us a way to find the next approximate zero after we have found our first zeros.

ie If x_0 is a root $P(x_0) = b_0 = 0$
$\Rightarrow Q(x)(x - x_0) = P(x)$

so we can find next root by examining roots of $Q(x)$!

If x_0 is an approximate root of p, then $b_0 \approx 0$ and

$$P(x) \approx Q(x)(x - x_0)$$

To find the next approximate zero we can apply Newton's Method to

$$P(x) / (x - x_0) = Q(x)$$

(a process called deflation)

Once a zero is found, deflate, and look for the next one.

Of course, an approximate root of $Q(x) = 0$ will generally not approximate a root of $P(x) = 0$ as well as $Q(x) = 0$, and the error increases as more roots are found.

One solution to this problem is to use these zeros as the initial guesses for Newton's Method applied to the original polynomial, $P(x)$.

It is also helpful to determine the smaller magnitude roots first. (why?)

Note also that it is possible to handle polynomials that have complex roots, but all real coefficients. Horner's scheme as well as Newton's Method work equally well for complex numbers, and we just need to begin with a complex initial approximation and do all the computations using complex arithmetic.

See the text for an alternative approach based on Newton's Method and the quadratic formula.

Müller's Method is another root finding technique for polynomials.

It can be viewed as an extension of the Secant Method.

Starting from 3 initial approximations x_0, x_1, x_2, it determines the next approximation by considering the intersection of the x-axis with the parabola through

$$(x_0, f(x_0)),\ (x_1, f(x_1)),\ (x_2, f(x_2))$$