5 Finding roots of equations

Topics:
1. Problem statement
2. Bisection Method
3. Newton’s Method
4. Fixed Point Iterations
5. Systems of equations
6. Notes and further reading

5.1 Problem statement
The basic root finding problem is the problem of finding a solution to an equation of the form
\[ f(x) = 0 \]  \hspace{1cm} (36)
for some function \( f \). A solution \( x \) to (36) is called a root of \( f \).

Most of the focus in this section will be on the case where \( f \) is a function of a single variable \( x \).

Example 5.1. Suppose we wish to find the angle \( \theta \) for which \( \sin \theta = 0.25 \). We can cast this problem as a root finding problem of the form (36) by setting \( f(x) = \sin x - 0.25 \).

This unit starts out by developing a number of root finding algorithms in the special setting where \( f \) is a function of one variable. However, for scientific applications, most cases of interest are where \( f \) is a vector valued function that takes vector arguments, i.e. where we have a system of equations. We’ll address this higher dimensional problem in the last subsection, where we’ll see how some of our one-variable root finding techniques carry over to the setting of \( n \) equations in \( n \) unknowns with only minimal changes.

5.2 Bisection Method
The bisection method is based on the Intermediate Value Theorem (IVT) from first year calculus.

Theorem 5.1 (Intermediate Value Theorem). Let \( f(x) \) be a continuous function on the closed interval \([a, b]\), and suppose that \( f(a) \neq f(b) \). Then for any number \( M \) between \( f(a) \) and \( f(b) \), there is a number \( c \in (a, b) \) such that \( f(c) = M \).

In essence, the IVT states that continuous functions do not “skip values.” An immediate corollary of the IVT is the following:

Corollary 5.1. Suppose \( f(x) \) is continuous on \([a, b]\) and \( f(a) \cdot f(b) < 0 \), i.e. \( f(a) \) and \( f(b) \) have opposite signs. Then there is a \( c \in (a, b) \) such that \( f(c) = 0 \).

The bisection method takes as its point of departure a function \( f(x) \) and an interval \([a, b]\) such that \( f(a) \cdot f(b) < 0 \). (If one cannot find such an interval, one cannot use the bisection method.) To find the root \( c \in (a, b) \), the bisection method samples \( f \) at the point \( p = (a + b)/2 \), i.e. at the mid-point of the interval. If \( f(a) \cdot f(p) < 0 \), then by the IVT there is a root of \( f \) in \([a, p]\), whereas if \( f(a) \cdot f(p) > 0 \), there is a root of \( f \) in \([p, b]\). One iteration of the bisection method thus reduces the length of the interval within the root must lies by a factor of \( 1/2 \). The method is then applied iteratively, until the length of the interval in which the root lies is sufficiently small.

Concretely, the bisection method uses the following algorithm:
Algorithm 2: Bisection Method

INPUTS: $f$, $a$, $b$, tol, maxits

1. Set $i=1$, $fa = f(a)$.

2. while $i < \text{maxits}$:
   \begin{align*}
   p &= (a+b)/2 \\
   fp &= f(p) \\
   \text{if } fp = 0 \text{ or } (b-a)/2 < \text{tol}: \\
   & \quad \text{return } p \\
   i &= i + 1 \\
   \text{if } fa * fp > 0:
   & \quad a = p \\
   & \quad fa = fp \\
   \text{else:}
   & \quad b = p
   \end{align*}

3. return $p$

Classroom Activity: Brainstorm about the pros and cons of this algorithm. Possible answers:

**PROS:**
- guaranteed to converge
- easy to code
- doesn’t require any information about $f$ other than samples

**CONS:**
- slow

5.3 Newton’s Method

Like the bisection method, Newton’s method is an iterative method that converges to a root. The point of departure for Newton’s method is the insight that if $f$ were linear, we could find the root exactly, in a single step, just by knowing the slope of $f$ and the value of $f$ at our initial guess $x_0$. Of course in general $f$ is not linear, but the tangent line equation gives a good linear approximation to $f$, at least locally. Indeed, the equation of the tangent line to $f(x)$ at the point $x_0$ is just

$$y - f(x_0) = f'(x_0)(x - x_0),$$

so by setting $y = 0$ and solving for $x$, we see that the root of the tangent line is a value $x_1$ given by

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}.$$  \hfill (37)

At each iteration, Newton’s method solves for the exact root of the tangent line, and uses this root as an ‘updated’ guess.

Concretely, the algorithm for Newton’s method is as follows:
Algorithm 3: Newton’s method

**INPUTS:**

- f, fp, x0, tol, maxits

1. Set i=1, \( y = f(x0) \), \( yp = fp(x0) \).
2. \( x1 = x0 - y/yp \)
3. while \( |x1 - x0| < tol \) and \( i < maxits \):
   - \( i = i+1 \)
   - \( x0 = x1 \)
   - \( y = f(x0), yp = fp(x0) \)
   - \( x1 = x0 - y/yp \)
4. return \( x1 \)

It turns out that Newton’s Method is amazingly effective, at least under certain (modest) assumptions about the function \( f \). Newton’s Method enjoys a property called **quadratic convergence**, which means that the magnitude of the error term (i.e. the difference between the true root and the current guess) is squared at every iteration. (And since we assume that this error term is less than 1 to start with, this means that the error goes to 0 quickly.)

To formalize this, we use the following definition:

**Definition 5.1.** A sequence \( a_n \to a \) is said to converge **quadratically** if

\[
\lim_{n \to \infty} \frac{|a_{n+1} - a|}{|a_n - a|^2} = L < \infty.
\]

In other words, quadratic convergence requires that the square of the error in the previous term eventually be proportional to the error in the next term.

**Theorem 5.2.** Let \( f : \mathbb{R} \to \mathbb{R} \) satisfy \( f(x^*) = 0 \), \( f'(x^*) \neq 0 \), and \( |f''(\zeta)| < M \) for all \( \zeta \). Then Newton’s method converges to \( x^* \) quadratically.

**Proof.** Let \( x_k \) be an element in a Newton sequence of iterations, and use the first order Taylor series of \( f \) around \( x_k \) to evaluate \( f(x^*) \):

\[
f(x^*) = f(x_k) + f'(x_k)(x^* - x_k) + \frac{f''(\zeta)}{2} (x^* - x_k)^2.
\]

Since \( x^* \) is a root of \( f \) and \( f'(x^*) \neq 0 \), this expression simplifies as

\[
x_k - \frac{f(x_k)(x^* - x_k)}{f'(x_k)} = x^* + \frac{f''(\zeta)}{2} (x^* - x_k)^2.
\]

Note that the left hand side is just the Newton update step, i.e. it equals \( x_{k+1} \). Making this substitution, subtracting \( x^* \) from both sides, and taking absolute values yields

\[
|x_{k+1} - x^*| = |x^* - x_k| \cdot \frac{f''(\zeta)}{2} \leq |x^* - x_k|^2 \cdot M,
\]

where the last inequality follows by our assumption on the boundedness of the second derivative. The result follows. \( \square \)
5.4 Secant method

One major drawback to Newton’s method is that it requires one to calculate and code up \( f'(x) \). For simple functions it might be easy enough to find an analytic form for \( f'(x) \), but for complex ones, it can be difficult. Of course numerical differentiation could always be used to find \( f' \), but numerical differentiation induces approximation errors. One easy workaround to the problem is to use the secant method. The secant method is exactly the same as Newton’s method, but the update step use two previous points, not just one:

\[
x_1 = x_0 - f(x_0) \cdot \frac{x_0 - x_{-1}}{f(x_0) - f(x_{-1})},
\]

where \( x_{-1} \) and \( x_0 \) are two previous guesses for the root. Note that since \( \frac{f(x_0) - f(x_{-1})}{x_0 - x_{-1}} \) is just a secant approximation to \( f'(x_0) \), the term

\[
\frac{x_0 - x_{-1}}{f(x_0) - f(x_{-1})} \approx \frac{1}{f'(x_0)}.
\]

In other words, update step (38) is exactly like the Newton update step (37), with the derivative replaced by a secant approximation.

As one might expect, the secant method does not perform quite as well as Newton’s method, but it is easier to implement and saves the trouble of calculating \( f' \).

5.5 Fixed Point Iterations

A fixed point of a function \( f(x) \) is any solution to the equation

\[
x = f(x).
\]

The problem of finding roots and the problem of finding fixed points are equivalent in the following sense: if \( x^* \) is a root of \( f(x) \), then \( x^* \) is a fixed point of the function

\[
h(x) = f(x) + x,
\]

while if \( x^* \) is a fixed point of \( h(x) \), then it is a root of

\[
f(x) = h(x) - x.
\]

In general, we will use \( f \) to denote functions for which we are finding roots, and \( h \) to denote functions for which we are finding fixed points.

One numerical technique used to find fixed points is that of generating a fixed point iteration: if \( x_0 \) is an initial guess for a fixed point, then subsequent guesses are determined by the formula

\[
x_{k+1} = h(x_k).
\]

Under certain conditions on the function \( h(x) \), the sequence of iterates \( x_k \) actually converges to a fixed point \( x^* \).

**Definition 5.2.** Let \( I \subset \mathbb{R}^n \) be a closed and bounded set, \( n \geq 1 \). A function \( h(x) : \mathbb{R}^n \to \mathbb{R}^n \) is said to be contractive on \( I \) if the following two conditions hold:

1. \( h(I) \subset I \).
2. \( |h(u) - h(v)| < \delta|u - v| \) for some \( \delta \in (0, 1) \) and any \( u \) and \( v \) in \( I \).

It turns out that if \( f \) is contractive on \( I \), then \( h \) has a unique fixed point on \( I \), and fixed point iterations will converge to that fixed point. Before stating and proving this theorem, we look at some examples.

**Example 5.2.** For a simple example of contractive maps, consider a linear map of the form \( h(x) = ax \). This map has a fixed point at \( x^* = 0 \). Consider a neighborhood around \( x^* \) of the form \( I = [-r, r] \) for some \( r > 0 \). Note that if \( |a| < 1 \), \( h(I) \subset I \) and

\[
|h(u) - h(v)| = |au - av| = a|u - v|,
\]

i.e. \( h(x) \) is contractive on \( I \).
Note that in this example $|h'(x^*)| < 1$. The following example shows that this same property characterizes arbitrary linear maps.

**Example 5.3.** Consider a linear map of the form $h(x) = ax + b$, $a > 0$. The fixed point of such maps is

$$x^* = \frac{b}{1-a}.$$ 

Consider a neighbor around $x^*$ of the form

$$I = [x^* - r, x^* + r].$$

Note that

$$h(I) = \left[a \left(\frac{b}{1-a} - r\right) + b, a \left(\frac{b}{1-a} + r\right) + b\right] = [x^* - ar, x^* + ar],$$

whereupon we conclude that $h(I) \subset I$ if and only if $a < 1$. A similar calculation shows that $h$ shrinks distances between points in $I$, and thus that it is contractive on $I$.

Both of the above examples illustrate that at least for linear maps in one dimension, being contractive is equivalent to having $|h'(x^*)| < 1$. It turns out that the same is true even for non-linear maps, assuming the maps are “nice.” This is the content of the following theorem.

**Theorem 5.3.** Let $h : \mathbb{R} \to \mathbb{R}$, and suppose $h$, $h'$, and $h''$ are continuous in a neighborhood of a fixed point $x^*$, with $|h'(x^*)| < 1$. Then for some $r > 0$, the mapping $h(x)$ is contractive on the neighborhood $I = [x^* - r, x^* + r]$.

**Proof.** Since $h'$ and $h''$ are continuous in a neighborhood of $x^*$, we can find a number $\tilde{r}$ such that for all $\zeta$ in the neighborhood $\tilde{I} = [x^* - \tilde{r}, x^* + \tilde{r}]$,

$$|h''(\zeta)| < M$$

for some finite number $M$. Note that since $x^*$ is a fixed point, the first order Taylor series of $h(x)$ around base point $x^*$ has the form

$$h(x) = x^* + h'(x^*)(x - x^*) + \frac{h''(\zeta)}{2}(x - x^*)^2.$$ 

Let $u = x^* + r_1$ and $v = x^* + r_2$ be elements of $\tilde{I}$. Using the Taylor series to estimate $h(u)$ and $h(v)$ and then taking a difference yields:

$$h(u) - h(v) = h'(x^*)(r_1 - r_2) + \frac{f''(\zeta_1)}{2}r_1^2 + \frac{f''(\zeta_2)}{2}r_2^2.$$ 

Since $r_1 - r_2 = u - v$, it follows that

$$|h(u) - h(v)| \leq |u - v| \cdot \left(|h'(x^*)| + |M(r_1^2 + r_2^2)|\right),$$

where $M$ is an upper bound on $|h''(\zeta)/2|$ over all $\zeta$ in $\tilde{I}$. If we take $r < \tilde{r}$ to be a number such that

$$M \cdot 2r^2 < \frac{1 - |g'(x^*)|}{2}.$$ 

then the right hand side of (39) reduces to

$$|h(u) - h(v)| \leq |u - v| \cdot \frac{1 + |h'(x^*)|}{2} = \delta|u - v|$$

where

$$\delta = \frac{1 + |f'(x^*)|}{2} < 1.$$ 

This establishes property (ii) of the contractive definition. Property (i) is established in the same way, with $u = x^*$ and $v$ an arbitrary point in $[x^* - r, x^* + r]$. 

\[\square\]
5.6 Systems of Equations

So we see that at least in one dimension, have a small derivative at a fixed point is sufficient to guarantee that the map \( h \) is at least locally contractive, i.e. contractive in a small neighborhood around \( x^* \). There are ways to generalize this result to \( n \) dimensions, but we focus instead on the real point of this section, which is that contractive maps have unique fixed points, and these can be solved for via fixed point iterations.

**Theorem 5.4.** Suppose \( h : \mathbb{R}^n \to \mathbb{R}^n \) is contractive on a set \( I \). Then \( h \) has a unique fixed point \( x^* \) in \( I \), and if \( x_0 \in I \), then the fixed point iterates \( x_{k+1} = h(x_k) \) converge to \( x^* \).

Some of the proof lies relatively close to the surface:

**Classroom Activity:** Prove that if \( h \) has a fixed point, the fixed point is unique.

Here is the remainder of the proof for the one-dimensional case:

**Proof.** We need to establish that a fixed point exists, and that the sequence of fixed point iterations converges to that fixed point. We will do this in one step by showing that the sequence of fixed point iterations \( x_k \) converges. Since \( x_{k+1} = h(x_k) \), it follows that \( h(x^*) = x^* \), and thus that \( x^* \) is a fixed point for \( h \).

To show that the sequence \( x_k \) converges, we will show that it is Cauchy, i.e. that for any \( \epsilon > 0 \), there is an \( N \) such that whenever \( n \geq m \geq N \), \( |x_n - x_m| < \epsilon \). In real analysis one learns that sequences that converge are Cauchy sequences, and vise versa. In other words, establishing that \( x_k \) is Cauchy is sufficient to show that it converges.

Let \( \epsilon > 0 \) be given. For any two terms \( x_n \) and \( x_m \), write

\[
|x_n - x_m| = |(x_n - x_{n-1}) + (x_{n-1} - x_{n-2}) + \cdots + (x_{m+1} - x_m)|
\leq \sum_{i=m}^{n-1} |x_{i+1} - x_i|
= \sum_{i=m}^{n-1} |h(x_i) - x_i|
\leq \sum_{i=m}^{n-1} \delta^i \cdot |x_1 - x_0|
\]

where the last line follows by recognizing that each \( x_i = h(h(\cdots h(x_0) \cdots)) \) and invoking the contractive property \( |h(u) - h(v)| < \delta |u - v| \). In other words

\[
|x_n - x_m| \leq |x_1 - x_0| \cdot (\delta^m + \cdots + \delta^{n-1})
\leq |x_1 - x_0| \cdot \delta^m (1 + \delta + \delta^2 + \cdots)
= |x_1 - x_0| \cdot \frac{\delta^m}{1 - \delta}
\]

where the last line follows from the formula for a geometric series. Since \( \delta < 1 \), we can find \( N \) such that

\[
|x_1 - x_0| \cdot \frac{\delta^N}{1 - \delta} < \epsilon.
\]

whereupon \( |x_n - x_m| < \epsilon \) as long as \( n \geq m \geq N \).

**5.6 Systems of Equations**

Newton’s Method and the fixed point iteration method generalize to \( n \) dimensional problems with minimal modification. In \( n \) dimensions, we let \( X \in \mathbb{R}^n \) denote a vector, and \( F : \mathbb{R}^n \to \mathbb{R}^n \) a vector valued function. Solving a system of \( n \) equations in \( n \) unknowns is equivalent to finding the roots of a function \( F(X) \). Similarly, we can cast a root finding problem as a fixed point problem by introducing the function \( H(X) = F(X) + X \).
With this notation, the Newton update step becomes

$$X_{k+1} = X_k - M^{-1} F(X_k)$$

(40)

where $M$ is the matrix of partials of $F$ given by

$$M = \begin{bmatrix}
\frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\
\vdots & \ddots & \vdots \\
\frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n}
\end{bmatrix}$$

where the function $f_i$ are the component functions of $F$ and the variables $x_1, \ldots, x_n$ are the components of $X$. Note that in one dimension, $M$ is simply the scalar $f'(x_k)$, in which case multiplication by $M^{-1}$ corresponds to division: in other words, (40) reduces to the formula we already know in the case of one dimension.

Fixed points generalize even more easily: a fixed point in $n$ dimensions is a solution to $X = H(X)$, and our fixed point iterations are just $X_{k+1} = H(X_k)$. Our definition of contractive mappings is already set in an $n$ dimensional context, so it needs to adjustment, and it continues to be the case that if $H$ is contractive in a (closed and bounded) neighborhood, it will have a unique fixed point in that neighborhood, and fixed point iterates will converge to that fixed point.

Solving systems of non-linear equations is a fundamental task when using implicit schemes to solve systems of differential equations.

**Example 5.4.** Consider the system

$$X' = F(t, X).$$

An implicit scheme uses the approximation

$$X' \approx \frac{X_{i+1} - X_i}{\Delta t}$$

to generate the scheme

$$\frac{X_{i+1} - X_i}{\Delta t} = F(t_{i+1}, X_{i+1}),$$

which produces the equation

$$X_{i+1} - X_i - \Delta t F(t_{i+1}, X_{i+1}) = 0.$$

Setting

$$F(X) = X - X_i - \Delta t F(t_{i+1}, X)$$

we see that finding the update $X_{k+1}$ amounts to finding a root of the $F$, i.e. solving a system of $n$ equations in $n$ unknowns. Alternatively, it corresponds to finding a fixed point of the function

$$H(X) = X_i + \Delta t F(t_{i+1}, X_{i+1}).$$

(41)

Note that these systems of equations depend on $X_k$. In other words, at each time step, we need to solve a new system of equations! In theory, this could be a lot of work, unless our methods work really well. The good news is that they do! Using either fixed point iterations or Newton’s method, we can generally solve the systems within just a couple of iterations.

An important feature of these systems is that the map $H(X)$ is generally contractive as long as $\Delta t$ is small enough. Indeed, it is this fact that justifies our use of this method.

**Theorem 5.5.** Suppose $F : \mathbb{R}^n \to \mathbb{R}^n$ is continuous in a neighborhood of $X_i$, with continuous first and second partials. Then for $\Delta t$ sufficiently small, the map $H(X)$ defined in (41) is contractive.
Proof. We’ll prove the theorem for the special case of one dimension, but the general proof is similar. Consider a function $h : \mathbb{R} \to \mathbb{R}$ of the form $h(x) = \Delta t f(x) + c$, where $f$ has continuous first and second derivatives in a neighborhood of $c$. Let $v$ and $w$ be two points in this neighborhood, and write the Taylor series

$$h(v) = h(w) + h'(w)(v - w) + \frac{h''(\zeta)}{2}(v - w)^2.$$ 

Then since $h'(x) = \Delta t \cdot f'(x)$, we can write

$$h(v) - h(w) = \Delta t \cdot f'(w) \cdot (v - w) + \Delta t \cdot \frac{f''(\zeta)}{2}(v - w)^2$$

$$= (v - w) \cdot \Delta t \left(f'(w) + \frac{f''(\zeta)}{2}(v - w)^2 \right)$$

Finally, since $f$, $f'$, and $f''$ are continuous in a neighborhood of $c$, they are also uniformly bounded, whence we can find a $\Delta t$ small enough that

$$\Delta t \left(f'(w) + \frac{f''(\zeta)}{2}(v - w)^2 \right) \leq \delta < 1$$

for any $\zeta$, $v$, and $w$ in the neighborhood. We thus have have

$$|h(v) - h(w)| \leq |v - w| \cdot \delta,$$

which establishes property (ii) of the contractivity condition. Establishing property (i) is similar. \qed

Note that this is really good news. It means that as long as we are willing to take small time steps, our implicit scheme will always generate systems of equations that are associated with contractive maps, and we can thus solve them with fixed point iterations.

### 5.7 Notes and further reading

Most of this material came from our text, though many of the mathematical details were supplied by me. Burden and Faires has a section on root finding algorithms that is worth taking a look at. For a deeper understanding of how these things work, there is no substitute for coding them up and running some trials. You are highly encouraged to do this.