Numerical Analysis Course Notes

Isaac Holt

December 2, 2022

1 Chebyshev Polynomials

Theorem 1.0.1. let $w_n(x) = (x - x_0)\dots(x - x_n)$ with distinct nodes $x_0, \dots, x_n$, $x_j \subset [-1, 1]$. Then the maximum of $|w_n(x)|$ on $[-1, 1]$ attains its smallest value (2^{-n}) iff x_j are the zeros of $T_{n+1}(x)$.

Proof. "$\Longleftarrow$": By construction $2^{-n}T_{n+1}(x)$ is a monic polynomial (highest power of x is 1) with $n + 1$ roots in $[-1, 1]$. Suppose $S_{n+1}(x) = (x - z_0)\dots(x - z_n)$ is another monic polynomial such that $\max|S_{n+1}(x)| < 2^{-n} = \max|2^{-n}T_{n+1}(x)|$. Let $q_n(x) := 2^{-n}T_{n+1}(x) - S_{n+1}(x)$. Then $q_n(x) \in P_n$ since the coefficient of x^{n+1} in $T_{n+1}(x)$ and $S_{n+1}(x)$ are both 1 and so cancel out.

Then $q_n(y_j) = 2^{-n}T_{n+1}(y_j) - S_{n+1}(y_j)$ (y_j are the extrema of $T_n(x)$). $|S_{n+1}(y_j)| < 1$ by hypothesis. Therefore $q_n(y_j) > 0$ if j is odd and < 0 otherwise.

Since we have $n + 2$ of y_j, q_n has at least $n + 1$ zeros. But since $q_n \in P_n$, we must have $q_n(x) = 0$. Therefore $S_{n+1}(x) = 2^{-n}T_{n+1}(x)$. □

Remark. To use this in $[a, b]$ instead of $[-1, 1]$, one simply maps $x_j \to a + (x_j + 1)\frac{b-a}{2}$.

Remark. Putting the above into Cauchy's error formula, we have

$$\sup|f(x) - p(x)| \le 2^{-n}\left(\frac{b-a}{2}\right)^{n+1}\frac{1}{(n+1)!}$$

Remark. We have by the above theorem, $\max|w_n(x)| = \max|(x - x_0)\dots(x - x_n) \ge 2^{-n}$ for any choice of $x_1, \dots, x_n$, $x_j \subset [-1, 1]$. So 2^{-n} is a lower bound for $|w_n(x)|$.

The upper bound is given by $\max|w_n(x)| \le \epsilon|b - a|^n$.

2 Root Finding

2.1 Bracketing: Bisection

Given $f \in C^0([a, b])$ with $f(a)f(b) < 0$, repeat:

- let $(a_0, b_0) = (a, b)$
- let $m_n = \frac{1}{2}(a_n + b_n)$
- if $f(m_n)f(a_n) \ge 0$, set $(a_{n+1}, b_{n+1}) = (m_n, b_n)$
- otherwise, set $(a_{n+1}, b_{n+1}) = (a_n, m_n)$

$b_{n+1} - a_{n+1} = \frac{1}{2}(b_n - a_n)$. By the Intermediate Value Theorem, if $f(m_n) \neq 0$, for some $p \in (a_n, b_n)$, $f(p) = 0$.

$|p - m_n| \leq 2^{-(n+1)}(b - a)$

Remark. Each time, the width of the interval halves. In principle, we could get an approximation to any desired accuracy, but there are some caveats (e.g. with floating points).

2.2 Bracketing: False Position

Suppose we have $|f(b)| \ll |f(a)|$, then we would expect p to be closer to b than to a. Instead of $m_n = \frac{1}{2}(a_n + b_n)$, set

$$m_n = b_n - f(b_n)\frac{b_n - a_n}{f(b_n) - f(a_n)}$$

i.e. m_n is the x-intercept of the line from $(a_n, f(a_n))$ to $(b_n, f(b_n))$. This should sometimes give much faster approximation than bisection, but not always.

2.3 Aside: Continuity and Convergence

Definition 2.3.1. $f : I \to \mathbb{R}$ is continuous at $x \in I$ if for every $\epsilon > 0$, for some $\delta(x, \epsilon)$, $|y - x| < \delta \Rightarrow |f(x) - f(y)| < \epsilon$ for every $y \in B(x)$ ($B(x)$ is an open interval containing x).

Remark. In general, δ depends on ϵ and x. When δ is independent of x, f is uniformly continuous.

Definition 2.3.2. $f : I \to \mathbb{R}$ is Lipschitz continuous in I if for some $L > 0$, $|f(y) - f(x)| \leq L|y - x|$ for every $x \in I, y \in I$. In this case, $\delta = \epsilon/L$.

Remark. L (like δ above) is not unique. The smallest such L is called the Lipschitz constant of f in I.

Lemma 2.3.3.

1. If f is differentiable and I is compact, f is Lipschitz in I.

2. If f is Lipschitz, f is continuous.

Proof.

$$f(y) - f(x) = \int_x^y f'(x) ds$$

$$|f(y) - f(x)| = |\int_x^y f'(x) ds| \leq \int_x^y |f'(x)| ds$$

$$\leq \max_{s \in I} |f'(s)| \int_x^y ds = \max_{s \in I} |f'(s)||y - x|$$

We can take $L = \max_{s \in I} |f'(s)|$ □

Remark. The converses of 1. and 2. are false.

Remark.
- When f is continuous in I, we write $f \in C^0(I)$.
- When f is differentiable in I, we write $f \in C^1(I)$.

- When f is Lipschitz in I, we write $f \in C^{0,1}(I)$.
- We can then write $C^1(I) \subsetneq C^{0,1}(I) \subsetneq C^0(I)$.

Definition 2.3.4. A sequence (x_n) in $\mathbb{R}^d$ converges to x if for every $\epsilon > 0$, for some $N(\epsilon)$, for every $n \geq N(\epsilon)$, $|x_n - x| < \epsilon$.

This relies on us knowing x in the first place.

Definition 2.3.5. A sequence (x_n) is a Cauchy sequence if for every $\epsilon > 0$, for some $N(\epsilon)$, for every $m \geq N, n \geq N$, $|x_n - x_m| < \epsilon$.

Theorem 2.3.6. Let (x_n) be a Cauchy sequence in $\mathbb{R}^d$. Then (x_n) converges.

This is useful as it allows us to prove convergence without knowing x.

2.4 Fixed Point Iterations

We seek x such that $f(x) = 0$ for a function f. We rewrite this as

$$x = g(x)$$

We then seek to solve this equation by iterations:

1. pick some x_0
2. set $x_{n+1} = g(x_n)$

Theorem 2.4.1. (1d local convergence theorem): Let $g \in C'([a,b])$ have a fixed point $x_\star \in [a,b]$ $(g(x_\star) = x_\star)$ with $|g'(x_\star) < 1$. Then for x_0 sufficiently close to $x_\star$, the iteeration $x_{n+1} = g(x_n)$ converges to $x_\star$.

Proof. Let $g'(x_\star) = L \in (0,1)$ ($g'(x_\star) < 0$ is analogous). Since g' is continuous at $x_\star$, for every $L' \in (L,1)$, for some $\delta(L') > 0$, $g'(x) \leq L' < 1$ for every $x \in (x_\star - \delta, x_\star + \delta) = B_\delta$, therefore for every $x \in B_\delta$, $y \in B_\delta$, $|g(x) - g(y)| \leq \sup_{s \in B_\delta} |g'(s)||x - y| = L'|x - y|$ with $L' < 1$.

Let $x_\star \in B_\delta$, then $|g(x) - x_\star| = |g(x) - g(x_\star)| \leq L'|x - x_\star|$ since $x_\star = g(x_\star)$. So $x - x_\star \delta$ as $x \in B_\delta$, so $|g(x) - x_\star| \leq L'\delta \leq \delta$, therefore $g(B_\delta) \subseteq B_\delta$. □

Remark. We do not need to know $x_\star$ to apply the 1d local convergence theorem, we just need to know that $|g'(x)| < 1$ for every $x \in I$ for some interval I.

2.5 Order of convergence

Order of convergence is a rough measure of how quickly $x_n \to x$. We mainly look at sequences arising from iterations with a nice RHS (so not bisection).

Definition 2.5.1. Let $x_n \to x_*$ and assume that $x_n \neq x_*$ for every $n \geq 0$. $x_n \to x_*$ with order at least $\alpha > 1$ if

$$\lim_{n \to \infty} \frac{|x_{n+1} - x_*|}{|x_n - x_*|^\alpha} = \lambda < todo$$

and with order $\alpha = 1$ if also $\lambda < 1$.

Example 2.5.2. $x_n = n^{-\beta}$, $\beta > 0$

$$\frac{|x_{n+1} - x_*|}{|x_n - x_*|} = (\frac{n}{n+1})^\beta \to 1$$

Example 2.5.3. $x_n = e^{-n}$

$$\frac{|x_{n+1} - x_*|}{|x_n - x_*|} = 1/e < 1$$

Example 2.5.4. $x_n = \frac{1}{n!}$

$$\frac{|x_{n+1} - x_*|}{|x_n - x_*|} = 1/(n+1) \to 0$$

The order of convergence of $x_n \to x_*$ is

$$\alpha = \sup\{\beta : \lim_{n\to\infty} \frac{|x_{n+1} - x_*|}{|x_n - x_*|^\beta} < todo\}$$

for $\alpha > 1$.
For $\alpha = 1$, we also require that the limit < 1.
The convergence is linear if $\alpha = 1$, superlinear if $\alpha > 1$ and sublinear otherwise.

Remark. Order of convergence need not be an integer.

Remark. We need to know x_* in order to determine order of convergence.

Remark. Our definition is not comprehensive for general sequences.

Applying this to iterations:
$x_{n+1} - x_* = g(x_n) - g(x_*) = (x_n - x_*)g'(c_n)$ for some $c \in \operatorname{conv}\{x_n, x_*\}$ by the Mean Value Theorem.
Therefore

$$|x_{n+1} - x_*||x_n - x_*| = |g'(c_n)| \to g'(x_*)$$

We conclude that for $g \in C^2(I)$, the iteration $x_{n+1} = g(x_n)$ converges linearly if $g'(x_*) \neq 0$ and $|g'(x_*)| < 1$, and superlinearly otherwise.

Proposition 2.5.5. Let $g \in C^{N+1}(D)$ for some $D \subseteq \mathbb{R}$ and let $g(x_*) = x_*$, with x_* in the interior of D.

Then the iteration $x_{n+1} = g(x_n)$ converges to x_* for x_0 sufficiently close to x_* with order $N+1$ iff $g'(x_*) = g''(x_*) = \cdots = g^{(N)}(x_*) = 0$ and $g^{(N+1)}(x_*) \neq 0$.

Proof. $x_{n+1} - x_* = g(x_n) - g(x_*) = g(x_*) + (x_n - x_*)g'(x_*) + \cdots + \frac{(x_n-x_*)^N}{N!}g^{(N)}(x_*) + \frac{(x_n-x_*)^{N+1}}{(N+1)!}g^{(N+1)}(c_n) - g(x_*) = \frac{(x_n-x_*)^{N+1}}{(N+1)!}g^{(N+1)}(c_n)$. Thus

$$|x_{n+1} - x_*||x_n - x_*|^{N+1} = \frac{|g^{(N+1)}(c_n)|}{(N+1)!} \to \frac{|g^{(N+1)}(x_*)|}{(N+1)!} < todo$$

$\square$

2.6 Higher order iterative methods

We want to rearrange $f(x) = 0$ to get faster convergence.

$x_{n+1} = g(x_n) = x_n + \phi(x_n)f(x_n)$ for some ϕ.

Using the above proposition, we need $g'(x_*) = 0$.

$g'(x_*) = 1 + \phi'(x_*)f(x_*) + \phi(x_*)f'(x_*) = 0$

So if $f'(x_*) \neq 0$, we take $\phi(x) = -\frac{1}{f'(x)}$.

This is the Newton-Raphson method:

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

Multiplication of a computer is parallelisable while division is not. So we can use Newton-Raphson to divide numbers with multiplication.

To compute $x_* = 1/b$ so $f(x_*) = \frac{1}{x_*} - b = 0$, so using Newton-Raphson, $x_{n+1} = x_n - \frac{{x_n}^{-1} - b}{-{x_n}^{-2}} = x_n(2 - bx_n)$ which involes only multiplication and subtraction. When taking out the exponent, $x_0 \in \left[\frac{1}{2}, 1\right)$, and this converges quickly.

Remark. This only works with floating points, not integers. Floating point division is 5 times faster than integer division.

Remark. It can be difficult in practice to determine the interval/domain of convergence for Newton-Raphson. We should always perform a "sanity check" when using it.

Example 2.6.1. One advantage of iterative methods is that they also work (in principle) in higher dimensions.

Suppose $f : \mathbb{R}^2 \to \mathbb{R}^2$ has a root at $\underline{p} = (p_1, p_2)$. Using Taylor expansion at the current point $\underline{x_n}$, derive Newton-Raphson (2D):

$$\underline{x_{n+1}} = \underline{x_n} - (Df)^{-1}\underline{f}(\underline{x_n})$$

We week $\underline{x_*} = \underline{p}$ such that $f_1(p_1, p_2) = f_2(p_1, p_2) = 0$. Taylor-expanding at $\underline{x}$:

$$0 = f_1(p_1, p_2) = f_1(x_1, x_2) + (p_1 - x_1)\frac{\partial f_1}{\partial x_1}(x_1, x_2) + (p_2 - x_2)\frac{\partial f_1}{\partial x_2}(x_1, x_2) + O(|\underline{p} - \underline{x}|^2)$$

$$0 = f_2(p_1, p_2) = f_2(x_1, x_2) + (p_1 - x_1)\frac{\partial f_2}{\partial x_1}(x_1, x_2) + (p_2 - x_2)\frac{\partial f_2}{\partial x_2}(x_1, x_2) + O(|\underline{p} - \underline{x}|^2)$$

In matrix form:

$$(0, 0) = (f_1(\underline{x}), f_2(\underline{x})) = (Df)(x_1, x_2) \cdot (x_1 - p_1, x_2 - p_2) + O(|\underline{p} - \underline{x}|^2)$$

Assuming that Df is invertible (equivalently, $f'(\underline{x}) \neq 0$), we can multiply the equation by $(Df)^{-1}$ to get

$$(p_1, p_2) = (x_1, x_2) - (((Df)^{-1})(x_1, x_2)) \cdot (f_1(\underline{x}), f_2(\underline{x})) + O(|\underline{p} - \underline{x}|^2)$$

So

$$\underline{p} = \underline{x} - (((Df)^{-1})(x_1, x_2))\underline{f}(\underline{x}) + O(|\underline{p} - \underline{x}|^2)$$

We can use this to construct our iteration by replacing $\underline{x}$ with $\underline{x_n}$ and $\underline{p}$ with $\underline{x_{n+1}}$, and removing the $O(|\underline{p} - \underline{x}|^2)$.

2.7 Secant method

One disadvantage of Newton-Raphson is that we need the derivative, f'. If f is complicated or is itself computed numerically, we need to approximate f'. An alternative method is the secant method.

The secant method approximates f' with:

$$f'(x_n) \approx \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$

so the iteration becomes

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

Remark. This is a (scalar) two-step method: $x_{n+1} = g(x_n, x_{n-1})$, where x_n and x_{n-1} are needed to calculate x_{n+1}.

Theorem 2.7.1. Let $f \in C^2$ with $f(x_*) = 0$ and $f'(x_*) \neq 0$. Then the secant method is convergent with order $\alpha = \frac{1+\sqrt{5}}{2}$ for every $x_0 \neq x_1$ sufficiently close to x_*.

Proof. TODO: See video on Panopto $\square$

Remark.

1. When implementing the secant method, one must be careful with floating point effects:

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

   becomes very inaccurate as $x_n - x_{n-1} \to 0$.

2. $e_n := x_n - x_*$ alternates in sign if $g'(x_k) < 0$ and has the same sign if $g'(x_*) > 0$. From the proof of the method,

   $$e_{n+1} = e_n e_{n-1} \frac{f''(\theta_n)}{f'(\phi_n)}$$

   where $\theta_n, \phi_n \in \text{conv}\{x_{n-1}, x_n, x_{n+1}\}$.

   For $e_0 e_1 < 0$ and n sufficiently large, the error e_n follows the pattern $+, +, -$ or $-, -, +$.

3 Numerical differentiation

3.1 Forward/backward difference

$f'(x) = \lim_{h\to 0} \frac{f(x+h)-f(x)}{h} \approx \frac{f(x+h)-f(x)}{h}$ for some small $h \neq 0$.

More rigorously, we can use Taylor series:

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(\xi)$$

where $\xi \in \operatorname{conv}\{x, x+h\}$. So

$$f'(x) = \frac{f(x+h)-f(x)}{h} - \frac{h}{2}f''(\xi)$$

The error $\frac{h}{2}f''(\xi)$ is of order $O(h)$.

For $h > 0$ this is called forward difference.

For $h < 0$, this is called backward difference.

3.2 Centred difference

$$f(x \pm h) = f(x) \pm hf'(x) + \frac{h^2}{2}f''(x) \pm \frac{h^3}{6}f'''(\xi_\pm)$$

Then

$$f'(x) = \frac{f(x+h)-f(x-h)}{2h} + \frac{h^2}{12}\left(f'''(\xi_+) + f'''(\xi_-)\right)$$

The error $\frac{h^2}{12}\left(f'''(\xi_+) + f'''(\xi_-)\right)$ is of order $O(h^2)$.

Remark. It is important to **not take h too small** when computing numerically. There is always a tradeoff between formal analytical accuracy and floating point errors. Formal analytical accuracy is better for smaller $|h|$, floating point errors are worse for smaller $|h|$.

Remark. Sometimes, we can only take $h < 0$ or $h > 0$ (not both), e.g. when solving differential equations numerically. If we have an ODE

$$\frac{du}{dt} = F(u)$$

Given $u(0)$ we want to solve for $u(t), t \geq 0$. We approximate $u(t)$ by a $u(t_n)$ with $t_n = n\delta t, n \in \mathbb{N}$, with $\delta t > 0$ small. Then

$$\frac{du}{dt} \approx \frac{u(t_n + \delta t) - u(t_n)}{\delta t} \approx F(u(t_n))$$

3.3 Richardson extrapolation

Let $f'(x) - \frac{f(x+h)-f(x)}{h} = f'(x) - R_h^{(1)}(x) = c_1(x)h + c_2(x)h^2 + c_3(x)h^3 + \cdots$ but suppose we cannot compute the c_k.

$R_{h/2}^{(1)}(x) = f'(x) - c_1\frac{h}{2} - c_2\frac{h^2}{4} - c_3\frac{h^3}{6} - \cdots$. We can use this to eleminate c_1:

$$2R_{h/2}^{(1)}(x) - R_h^{(1)}(x) = f'(x) - c_2'h^2 - c_3'h^3 - \cdots$$

So the error is of order $O(h^2)$. Then

$$f'(x) = R_h^{(2)}(x) + O(h^2)$$

where $R_h^{(2)} = 2R_{h/2}^{(1)}(x) - R_h^{(1)}(x)$.

Now, $R_{h/2}^{(2)}(x) = f'(x) - c_2' \frac{h^2}{4} - c_3' \frac{h^3}{8}$ and we use this to eliminate c_2':

$$4R_{h/2}^{(2)}(x) - R_h^{(2)}(x) = 3f'(x) + O(h^3)$$

So we set

$$R_h^{(3)} := \frac{2^2 R_{h/2}^{(2)}(x) - R_h^{(2)}(x)}{2^2 - 1} = f'(x) + O(h^3)$$

Remark.

1. We only need to specify the powers of h, not the coefficients c_k as long as they are **non-zero**. So when using centred difference to calculate $R_h^{(1)}$ then $R_h^{(2)}$, $R_h^{(3)}$ will be different.

2. Richardson extrapolation works with many other approximation methods involving a small parameter (not just differentiation).

3. There is no standard notation for this method.

4. Some series expansions have irregular/non-integer powers, e.g. the Airy function $\mathrm{Ai}(x)$ which is a solution of $\frac{d^2 f}{dx^2} = x f(x)$.