Math 151A – Applied Numerical Methods I

University of California, Los Angeles

Duc Vu

Fall 2021

This is math 151A – Applied Numerical Methods taught by Professor Jiang. We meet weekly on MWF from 1:00 pm to 1:50 pm for lecture. The recommended textbook for the class is *Numerical Analysis* 10^{th} by *Burden*, *Faires* and *Burden*. Other course notes can be found at my blog site. Please let me know through my email if you spot any typos in the note.

Contents

List of Theorems

List of Definitions

§1 Lec 1: Sep 24, 2021

§1.1 Calculus Review

- Intermediate Value Theorem (IVT): For continuous function $C([a,b])$, let $f \in C([a,b])$. Let $k \in \mathbb{R}$ s.t. k is strictly between $f(a)$ and $f(b)$. Then, $\exists$ some $c \in (a,b)$ s.t. $f(c) = k$.

  Question 1.1. Why is IVT useful?

  It guarantees the existence of solution to some nonlinear equations.

  Example 1.1

  Let $f(x) = 4x^2 - e^x$. IVT tells us $\exists x^*$ s.t. $f(x^*) = 0$.

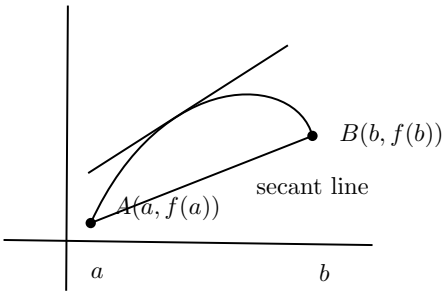
  $$f(0) = 0 - e^0 = -1 < 0$$
  $$f(1) = 4 - e > 0$$

  With $k = 0$, by IVT, $\exists c \in (0,1)$ s.t. $f(c) = 0$.

- Mean Value Theorem (MVT): If $f \in C([a,b])$ and f is differentiable in (a,b), then $\exists c \in (a,b)$ s.t.

  $$f'(c) = \frac{f(b) - f(a)}{b - a}$$

  in which $f'(c)$ is essentially the slope of the tangent line at $(c, f(c))$.



- Taylor's Theorem: Apply for a differentiable function, $f \in C^m([a,b])$ – f is m times continuously differentiable.

  Theorem 1.2 (Taylor)

  Let $f \in C^n([a,b])$. Let $x_0 \in [a,b]$. Assume $f^{(n+1)}$ exists on $[a,b]$. Then $\forall x \in [a,b]$, $\exists \xi(x) \in \mathbb{R}$ s.t. $x_0 < \xi < x$ or $x < \xi < x_0$. Then, we can express f as

  $$f(x) = P_n(x) + R_n(x)$$

  where

  $$P_n(x) = f(x_0) + f'(x_0)(x - x_0) + f''(x_0)\frac{(x - x_0)^2}{2!} + \ldots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n$$

  and

  $$R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)^{n+1}$$

Example 1.3

$f(x) = \cos(x)$, $x_0 = 0$

$$\begin{aligned} f(x) = \cos(x) &= f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \frac{f'''(\xi(x))}{3!}x^3 \\ &= 1 + 0 - \frac{1}{2}x^2 + \frac{1}{6}x^3 \sin(\xi(x)) \end{aligned}$$

Note: Saying $f \in C^1$ is different from saying $f'(x)$ exists.

Example 1.4

Consider

$$f(x) = \begin{cases} x^2 \sin\left(\frac{1}{x}\right), & x \neq 0 \\ 0, & x = 0 \end{cases}$$

Have

$$\begin{aligned} f'(0) &= \lim_{h \to 0} \frac{f(0+h) - f(0)}{h} \\ &= \lim_{h \to 0} h \sin\left(\frac{1}{h}\right) \\ &= 0 \end{aligned}$$

But $f'(x)$ is not continuous. Specifically,

$$f'(x) = \begin{cases} 2x \sin\left(\frac{1}{x}\right) - \cos\left(\frac{1}{x}\right), & x \neq 0 \\ 0, & x = 0 \end{cases}$$

Take sequence $\frac{1}{2k\pi}$, $f' \to -1$ and $\frac{1}{(2k+1)\pi}$, $f' \to 1$. Thus, the function is not continuous as it converges to two different values.

§2 Lec 2: Sep 27, 2021

§2.1 Errors and Convergence Rate

Fact 2.1. 1. Computers have finite memory

2. Only a subset of rational numbers $\mathbb{Q}$ can be exactly represented/stored.
3. Working with floating numbers instead of reals produces round-off error

Definition 2.1 (Error) — Let $p \in \mathbb{R}$, $\tilde{p}$ approximate to p. We define absolute error as

$$e_{\text{abs}} \coloneqq |p - \tilde{p}|$$

We define relative error as

$$e_{\text{rel}} \coloneqq \left| \frac{p - \tilde{p}}{p} \right|$$

Example 2.2 • $p = 1$, $\tilde{p} = 0.9$. In this case,

$$e_{\text{abs}} = 0.1$$
$$e_{\text{rel}} = 0.1$$

• $p = 1000$, $\tilde{p} = 900$

$$e_{\text{abs}} = 100$$
$$e_{\text{rel}} = 0.1$$

Finite Digit Arithmetic

Example 2.3 • π is rounded/chopped by computers

• $x = \frac{5}{7} = 0.\overline{714285}$, $y = \frac{1}{3} = 0.\overline{3}$
Let $fl(x)$ is the floating point approx. to x. For example, we assume 5 digit rounding.

$$fl(x) = 0.71428, \qquad fl(y) = 0.33333$$

Say if we want to add $x + y$ on computer

$$fl\left(fl(x) + fl(y)\right) = fl\left(1.04761\right) = 1.0476$$

Example 2.4

$f(x) = x^3 - 6.1x^2 + 3.2x + 1.5$ where $x = 4.71$. The exact value of $f(x)$ at $x = 4.71$ is -14.263899. Let's assume 3 digit rounding

$$\begin{aligned} fl(x^2) &= fl(4.71 \cdot 4.71) = fl(22.1841) = 22.2 \\ fl(x^3) &= 105 \\ fl(3.2x) &= 15.1 \\ fl\left(f(4.71)\right) &= -13.4 \end{aligned}$$

The relative error here is approximately 6% which is huge. Our example has 7 floating point operations (FLOPs). In order to reduce the floating point error, we want to nest the function

$$f(x) = ((x - 6.1)x + 3.2)x + 1.5 \quad \text{– 5 FLOPs}$$

So $fl\left(f(4.71)\right) = -14.3$ and the $e_{\text{rel}} = 0.25\%$.

Remark 2.5. Every operation introduces error.

Order of convergence for Sequences:

Definition 2.6 (Order of Convergence for Sequences) — For a convergent sequence $(p_n) = (p_1, p_2, p_3, \ldots)$. Let $p_n \to p$ as $n \to \infty$. Assume $p_n \neq p$. Then, if $\exists \lambda,\ \alpha$ with $0 < \lambda < \infty$ and $\alpha > 0$ s.t.

$$\lim_{n \to \infty} \frac{|p_{n+1} - p|}{|p_n - p|^{\alpha}} = \lambda$$

Then we say p_n converges to p with order α.

Example 2.7

$p_1 = 1, p_2 = \frac{1}{5}, \ldots, p_n = \frac{1}{5}p_{n-1}$ or $p_n = \frac{1}{5^{n-1}}$ where $p_n \to 0$ as $n \to \infty$.

$$\frac{|p_{n+1} - 0|}{|p_n - 0|^1} = \frac{\left(\frac{1}{5}\right)^n}{\left(\frac{1}{5}\right)^{n-1}} = \frac{1}{5}$$

So p_n converges with $\alpha = 1$

Problem 2.1. Test with $\alpha = 2$.

Definition 2.8 (Big O Notation) — We have $a(t) = \mathcal{O}\left(b(t)\right)$ where a is on the order of $b \iff$

$$\exists C > 0 \quad \ni \quad |a(t)| \leq Cb(t) \quad \text{for } t \to 0 \text{ or } t \to \infty$$

In practice, the definition is equivalent to

$$\lim_{t \to 0} \frac{|a(t)|}{b(t)} \text{ is bounded by a positive number}$$

§3 Lec 3: Sep 29, 2021

§3.1 Lec 2 (Cont'd)

Example 3.1

The Taylor's theorem for $\cos(h)$ about 0 is

$$\cos(h) = 1 - \frac{1}{2}h^2 + \frac{1}{24}h^4 \cos(\xi(h)) \text{ with some } 0 < \xi(h) < h$$

Denote $f(h) = \cos(h) + \frac{1}{2}h^2 - 1 = \frac{1}{24}h^4 \cos(\xi(h))$

$$\lim_{h\to 0} \frac{|f(h)|}{h^4} = \lim_{h\to 0} \frac{1}{24} |\cos(\xi(h))| \leq \frac{1}{24}$$

Thus y definition of big $\mathcal{O}$ notation,

$$f(h) = \mathcal{O}(h^4)$$

§3.2 Root Finding with Bisection

The goal is to find a root, or a zero, of a function f, i.e., find p s.t. $f(p) = 0$. First, let's assume

1. $f \in C([a, b])$
2. $f(a)f(b) < 0$

Then, $\exists p$ s.t. $f(p) = 0$ (by IVT).

Example 3.2

Consider:

$$f(x) = \sqrt{x} - \cos x, \qquad [a, b] = [0, 1]$$

Then,

$$f(0) = -1, \qquad f(1) = 1 - \cos 1 > 0$$

Therefore, by IVT, $\exists p \in (0, 1)$ s.t. $\sqrt{p} - \cos p = 0$.

Bisection Method (B.M): is an algorithm to approximate p s.t. $f(p) = 0$ on an interval $[a, b]$.
Algorithm 1: Bisection method (given $f(x) \in C([a, b])$, with $f(a)f(b) < 0$)

1. Set $a_1 = a$, $b_1 = b$
2. Set $p_1 = \frac{a_1+b_1}{2}$
3. *if* $f(p_1) == 0$ *then* we are done!
4. *else if* $f(p_1)$ has same sign as $f(a_1)$ *then* $p \in (p_1, b_1)$
   - Set $a_2 = p_1$, $b_2 = b_1$
5. *else if* $f(p_1)$ has same sign as $f(b_1)$ *then* $p \in (a_1, p_1)$
   - Set $a_2 = a_1$, $b_2 = p_1$
6. *end*

7. Set $p_2 = \frac{a_2+b_2}{2}$
8. Reset the entire if/else process.

Remark 3.3. B.M. is similar to binary search in computer algorithms. If there exists multiple roots, e.g., $\{p, q, r\} \in [a, b]$, then the B.M. is guaranteed to find exactly one root, not all of them (but no guarantee exists for which one the method will find).

Stopping Criteria: We need a sequence $(p_1, p_2, \ldots)$ and need specified tolerance ε. Choices for when to stop an algorithm:

- $|p_n - p_{n-1}| < \varepsilon$ – absolute difference between successive elements of the sequence
- $\frac{|p_n - p_{n-1}|}{|p_n|} < \varepsilon$ (assume $p_n \neq 0$) – relative difference
- $|f(p_n)| < \varepsilon$ – sometimes called a residual (how close are we to the answer).

§4 | Lec 4: Oct 1, 2021

§4.1 Bisection Method (Cont'd)

Remark 4.1. B.M. is a global method, $f \in C([a,b])$ as long as the assumptions are satisfied, $f(a)f(b) < 0$, the B.M. will converge. In particular, it will converge to some point p s.t. $f(p) = 0$. Here "global" means the algorithm doesn't need a good initial guess p_0 unlike some "local" methods that we will cover later.

Example 4.2

The B.M. won't work for functions like $f(x) = x^2$ even though it has a root at $p = 0$ because we can't find any a, b that satisfies $f(a)f(b) < 0$.

Theorem 4.3 (Convergence Order of B.M.)

The sequence provided by B.M. satisfies

$$|p_n - p| \leq \frac{b-a}{2^n}$$

which approaches to 0 as $n \to \infty$.

This further tells us that the error bound of B.M. converges linearly. Recall from previous lectures that linear convergence for a convergent sequence (p_n) means that

$$\lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_n - p|^1} = \lambda \quad \text{for some finite positive } \lambda$$

and

$$p_n = \frac{b-a}{2^n}, \quad p = 0$$

We can easily show that $\lambda = \frac{1}{2}$.

Remark 4.4. The B.M. converges slowly compared to other methods. We will soon see that Newton's method has quadratic order of convergence.

§4.2 Fixed Points

Definition 4.5 (Fixed Point) — Let function g be $g : [a,b] \to \mathbb{R}$ and $p \in [a,b]$ s.t. $g(p) = p$. Then p is a fixed point of g.

Theorem 4.6

Let p be a fixed point of g, then p is also a root of $G(x) := g(x) - x$.

Proof. Obvious by definition. □

Given a root-finding problem $f(p) = 0$, we can define functions g with a fixed point at p in a number of ways. For example, as $g(x) = x - f(x)$ or as $g(x) = x + 3f(x)$.
A fixed point for g just corresponds to the intersection between $y = g(x)$ and $y = x$.
Fixed Point Iteration (F.P.I): the F.P.I method is quite simple. For $g \in C([a, b])$ and $p_0 \in [a, b]$ we set $p_{n+1} = g(p_n)$. We also need $g(x) \in [a, b]$, otherwise at some point of the algorithm we won't be able to proceed to evaluate g. Also note that the initial guess p_0 is arbitrary.

$$p_1 = g(p_0), \quad p_2 = g(p_1), \ldots, p_{n+1} = g(p_n)$$

We use the same stopping criteria as in B.M.

Example 4.7 (F.P.I Failure Case)
To solve $x^2 - 7 = 0$, it is equivalent to $x = \frac{7}{x}$. We want to use F.P.I to find $p = \sqrt{7} \approx 2.64575\ldots$, so we can set

$$g_1(x) = \frac{7}{x}$$

then the goal is to find p s.t. $p = g_1(p)$. Another option is to use

$$g_2(x) := \frac{x + \frac{7}{x}}{2} = x$$

Let $p_0 = 3$ we can show that

- $g_1(x)$: $p_0 = 3$, $p_1 = \frac{7}{3}$, $p_2 = 3, \ldots$, oscillates between 2 numbers
- $g_2(x)$: $p_0 = 3$, $p_1 = 2.666\ldots$, $p_2 = 2.645833\ldots, \ldots$

Example 4.8
$x^3 + 4x^2 - 10 = 0$ has a unique root in $[1, 2]$, i.e. $p = 1.365230013$.

a) $x = g_1(x) = x - x^3 - 4x^2 + 10$ – does not converge

b) $x = g_2(x) = \left(\frac{10}{x} - 4x\right)^{\frac{1}{2}}$ – does not converge

c) $x = g_3(x) = \frac{1}{2}(10 - x^3)^{\frac{1}{2}}$ – converge

d) $x = g_4(x) = \left(\frac{10}{x+4}\right)^{\frac{1}{2}}$ – converge

e) $x = g_5(x) = x - \frac{x^3 + 4x^2 - 10}{3x^2 + 8x}$ – converge

We can see that the choice of $g(x)$ is critical to determine whether the algorithm converges. Before delving into that problem, let's first establish a theorem about the existence of a fixed point.

Theorem 4.9 (Existence of a Fixed Point)
Let $g \in C([a, b])$ with $a \leq g(x) \leq b$. Then, $\forall x \in [a, b]$, $\exists$ at least one fixed point p s.t. $g(p) = p$.

§5 | Lec 5: Oct 4, 2021

§5.1 Fixed Point Iteration (Cont'd)

Recall

Theorem 5.1
Let $g \in C[a,b]$ with $a \leq g(x) \leq b \forall x \in [a,b]$, then $\exists$ at least one fixed point p s.t. $g(p) = p$.

Let's prove it.

Proof. First, we need to check if an end point is a fixed point, i.e., if

$$g(a) = a \quad \text{or} \quad g(b) = b$$

then we're done. Otherwise, let's define $G(x) \coloneqq g(x) - x$. Our goal is to use IVT to prove that G has a root. Then since $g \in [a,b]$, we know

$$\begin{aligned} G(a) = g(a) - a > 0, \quad G(b) = g(b) - b < 0 \\ \implies G(a)G(b) < 0 \end{aligned}$$

Also, $G \in C([a,b])$. Therefore, by IVT, $\exists p$ s.t. $G(p) = 0$, i.e., $\exists p \in [a,b]$ s.t. $g(p) = p$. □

Remark 5.2. The theorem is just a sufficient condition for existence.

Theorem 5.3 (FPI Convergence with Lipschitz Continuity)
Assume $g \in C([a,b])$, $g \in [a,b]$ (*) and $\exists k \in (0,1)$ s.t.

$$|g(x) - g(y)| \leq k|x - y|, \quad \forall x, y \in [a,b] \tag{**}$$

Then,

1. $\exists$ unique p s.t. $g(p) = p$.
2. The F.P.I ($p_{n+1} = g(p_n)$) will converge to p.
3. Error estimate: $|p_n - p| < k^n \max\{b - p_0, p_0 - a\}$.

Proof. 1. Let's prove by contradiction. Assume $\exists$ two different fixed points p and q, then

$$|g(p) - g(q)| = |p - q|$$

But by (**) we know

$$|g(p) - g(q)| \leq k|p - q|$$

This implies that

$$|p - q| \leq k|p - q|$$

which cannot be true since $p \neq q$ and $k \in (0,1)$ – contradiction!

2. + 3. By (**) we know that differences of g values are bounded. So let's try to convert this to something with g values. We know F.P.I $g(p_n) = p_{n+1}$ and also let p be the solution $g(p) = p$. So

$$|p_n - p| = |g(p_{n-1}) - g(p)|$$

By (**), we know

$$|p_n - p| = |g(p_{n-1}) - g(p)| \leq k\,|p_{n-1} - p|$$

Similarly,

$$k\,|p_{n-1} - p| = k\,|g(p_{n-2}) - g(p)| \leq k^2\,|p_{n-2} - p|$$

Recursively apply this until $n = 0$. Then,

$$|p_n - p| \leq k^n\,|p_0 - p|$$

Notice that

$$|p - p_0| \leq \max\{b - p_0, p_0 - a\}$$

Thus,

$$|p_n - p| \leq k^n \max\{b - p_0, p_0 - a\}$$

Since $k \in (0, 1)$, this goes to 0.

□

Remark 5.4. Speed of convergence depends on k. The closer to 0 k is, the faster it converges.

In practice, to use the theorem, it is sometimes more useful to look at the derivatives instead of Lipschitz condition.

Theorem 5.5 (FPI Convergence with Bounded Derivative)
Assume (*) and $g \in C^1[a, b]$ and that $\forall x \in [a, b]\,,\ \exists k \in (0, 1)$ s.t. $|g'(x)| \leq k$. Then, following the above theorem, F.P.I converges to the unique solution.

Proof. Here we need to prove that bounded derivative gives Lipschitz. Let's use MVT, $\exists c \in (a, b)$ s.t. $\forall x, y \in [a, b]$

$$g'(c) = \frac{g(x) - g(y)}{x - y}$$

Thus,

$$|g(x) - g(y)| = |g'(c)|\,|x - y| \leq k\,|x - y|$$

□

§6 Lec 6: Oct 6, 2021

§6.1 Newton's Method

Newton's Method (N.M.) is a classic technique used in science and engineering, research and industry all the time. There are many ways to derive it, and we will go over 3 today.

Analytic derivation with Taylor's polynomial:

Let $f \in C^2([a,b])$, p is a root ($f(p)=0$). Suppose p_n is "close to" p, i.e., $|p_n - p|$ is "small".

$$0 = f(p) = f(p_n) + f'(p_n)(p-p_n) + f''(\xi)\frac{(p-p_n)^2}{2}$$

If $|p-p_n|$ is "small", then $|p-p_n|^2$ is "really small". Up to an error of size $\approx (p-p_n)^2$,

$$0 = f(p) \approx f(p_n) + f'(p_n)(p-p_n)$$

So

$$p = p_n - \frac{f(p_n)}{f'(p_n)}$$

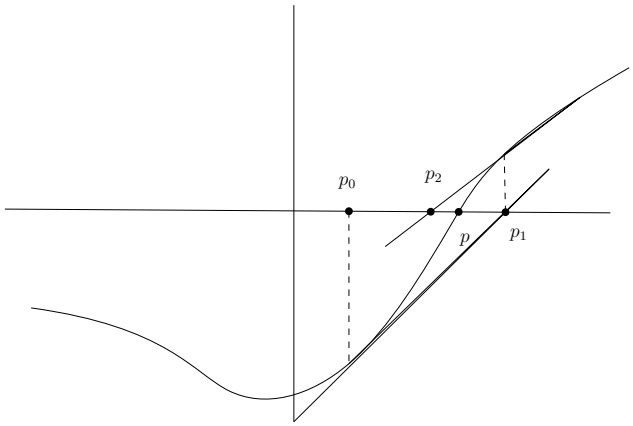
This can be used to "invent" Newton's method.

Definition 6.1 (Newton's Method) — Start with p_0 close to p, then do

$$p_{n+1} = p_n - \frac{f(p_n)}{f'(p_n)}$$

Remark 6.2. The initial guess p_0 must be close to p, otherwise the analytic derivation breaks down.

Graphical Derivation:



Tangent line: $y = ax + b$ and the intersection with the x-axis is p_{n+1}. We know

$$\begin{aligned} f(p_n) &= ap_n + b \\ 0 &= ap_{n+1} + b \\ a &= f'(p_n) \end{aligned}$$

The unknowns are a, b, p_{n+1}. Solving them we obtain

$$p_{n+1} = p_n - \frac{f(p_n)}{f'(p_n)}$$

Fixed Point Derivation Method:

Theorem 6.3
Let $g(x) \coloneqq x - \frac{f(x)}{f'(x)}$ for some $f \in C^1([a,b])$ where $f'(x) \neq 0$ for $x \in [a,b]$. Then $g(p) = p$ if and only if $f(p) = 0$.

Proof. Basic algebra :D □

Define a fixed point iteration from g

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

Remark 6.4. We must have $f'(p_n) \neq 0 \ \forall n$, otherwise, N.M will fail.

Pros of N.M:

- It will converge faster than the B.M. to the root p of function $f(x)$ (when it does converge).

Cons of N.M.:

- Unlike the B.M., N.M. is a local method, not global. That means p_0 must be sufficiently close to p for success.
- N.M. requires knowledge of $f'(x)$ and evaluation of $f'(x)$ (especially when f is $\mathbb{R}^n \to \mathbb{R}^m$).

In higher dimension, if f is $\mathbb{R}^n \to \mathbb{R}^n$, then N.M. is

$$\mathbf{x}_{n+1} = \mathbf{x}_n - (\mathbf{J}(\mathbf{x}_n))^{-1}\,\mathbf{f}(\mathbf{x}_n)$$

where $\mathbf{J}(x)$ is the Jacobian matrix and $J_{ii} = \frac{\partial f_i}{\partial x_j}(\mathbf{x})$.

§6.2 Secant Method

N.M. requires the knowledge of $f'(x)$ and evaluation of $f'(x)$, so we can approximate it as follows

$$f'(p_n) \approx \frac{f(p_n) - f(p_{n-1})}{p_n - p_{n-1}}$$

This defines the **Secant Method**.

Definition 6.5 (Secant Method) — Given some p_0 and p_1, define

$$p_{n+1} = p_n - f(p_n)\frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})}$$

Secant method is useful when you don't have access to $f'(x)$, e.g., when we don't have access to f.

§7 | Lec 7: Oct 8, 2021

§7.1 Secant Method (Cont'd)

Recall Newton's Method (N.M.) is defined as follows

$$\text{Given } p_0, \quad p_{n+1} = p_n + \frac{f(p_n)}{f'(p_n)}$$

This requires evaluation of f'. In general, this could be expensive or unknown, e.g., in higher dimension or $f(x)$ comes from experimental data. The definition of derivative is

$$f'(x) \coloneqq \lim_{h \to 0} \frac{f(x) - f(x-h)}{h}$$

So when h is small, the derivative can be approximated by "finite difference",

$$f'(x) \approx \frac{f(x) - f(x-h)}{h}$$

So if we let $x = p_n$, and $x - h = p_{n-1}$, then this becomes

$$f'(p_n) \approx \frac{f(p_n) - f(p_{n-1})}{p_n - p_{n-1}}$$

which holds true when $p_n - p_{n-1}$ is small. This leads us to the definition of secant method.

Definition 7.1 — Given p_0, p_1, secant method is defined as

$$p_{n+1} = p_n - f(p_n)\frac{p_n - p_{n-1}}{f(p_n) - f(p_{n-1})}$$

where the fraction is approximating $(f'(p_n))^{-1}$.

Question 7.1. How to get p_1?

e.g., running one iteration of bisection method.

§7.2 Local Convergence of Newton's Method

Theorem 7.2 (Newton Convergence)
Let $f \in C^2([a,b])$ and $p \in (a,b)$ s.t.

i) $f(p) = 0$

ii) $f'(p) \neq 0$

Then $\exists \delta > 0$ s.t. N.M. will converge for $\forall p_0 \in [p - \delta, p + \delta]$.

There is no guideline to find the exact δ – which means we don't know what close-enough means in practice unfortunately.

Proof. The idea here is to apply the F.P.I. theorem from previous lectures to some to-be-defined function g. What is g?
Key conditions to satisfy:

1. $\left[\hat{a}, \hat{b}\right] \to \left[\hat{a}, \hat{b}\right]$
2. g is C^1
3. g has bounded derivative with bound in $(0, 1)$.

Define $g(x) \coloneqq x - \frac{f(x)}{f'(x)}$. N.M. on $f(x)$ is the same as F.P.I. on $g(x)$

$$p_{n+1} = p_n - \frac{f(p_n)}{f'(p_n)} \iff g(p_n) = p_{n+1}$$

Thus, we just need to show the three postulates about g.

2. $f \in C^2([a, b])$ so $f \in C([a, b])$ and $f' \in C([a, b])$ and $f'' \in C([a, b])$. Let's compute $g'(x)$

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

   There exists a region $[p - \delta_1, p + \delta_1]$ in $[a, b]$ s.t. $f'(x) \neq 0$, so g' is continuous in $[p - \delta_1, p + \delta_1]$. This proves 2.

3. WTS: bounded derivative

$$g'(x) = \frac{f(x)f''(x)}{(f'(x))^2}$$
$$g'(p) = 0$$

   Due to continuity of g' in $[p - \delta_1, p + \delta_1]$, there exists a region (with $0 < \delta < \delta_1$) s.t. $|g'(x)| \leq k$ in $[p - \delta, p + \delta]$ for any $k \in (0, 1)$. This proves 3.

Lastly, let's show 1.

1. Need to prove g maps $[p - \delta, p + \delta]$ to $[p - \delta, p + \delta]$

$$|g(x) - p| = |g(x) - g(p)| = |g'(\xi)| \, |x - p| \leq k \, |x - p| < |x - p|$$

   By M.V.T, $\exists \xi \in (x, p)$. N.M. on $f(x)$ is the same as F.P.I on $g(x)$.

Now, we proved that F.P.I converges to p for any $p_0 \in [p - \delta, p + \delta]$. Equivalently, N.M. converges for f at p. □

Remark 7.3. δ cannot be a priori. In practice, we can

- begin with some $p_0 \in [a, b]$
- run several iterations of B.M. (a global method)
- switch to N.M.

§8 | Lec 8: Oct 11, 2021

§8.1 Convergence Order Theorem

Let's begin with a fact.

Fact 8.1. Let $p = g(p)$ e a fixed point, F.P.I $g(p_n) = p_{n+1}$

- If $g'(p) \neq 0$, we get linear convergence (order of convergence $\alpha = 1$)
- If $g'(p) = 0$, we get quadratic convergence ($\alpha = 2$)

Theorem 8.1

Let $g \in C^1([a,b])$ with $|g'(x)| \leq k$ for some $0 < k < 1$. If $g'(p) \neq 0$, then F.P.I. converges to p linearly.

Proof. From F.P.I convergence theorem, we know that F.P.I converges in this case. So we just need to prove the linear order. Use M.V.T.:

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

where ξ is between p_n and p.

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

in which k is a positive number that is smaller than 1. It's also easy to see that it only has linear convergence, e.g.,

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

Theorem 8.2 (Convergence Order Theorem of FPI)

Let $g \in C^\alpha([a,b])$, $\alpha \geq 2$ is an integer. If

i) $g(p) = p$

ii) $g'(p) = g''(p) = \ldots = g^{(\alpha-1)}(p) = 0$

iii) $g^{(\alpha)}(p) \neq 0$

Then, F.P.I. converges $\forall p_0$ sufficiently close to p with order α.

Proof. First let's prove that $p_n \to p$. We can follow the procedure in the proof in lecture 7.
Sketch: $g'(p) = 0$ and $g' \in C([a,b])$, $\exists \delta$ s.t.

$$|g'(x)| \leq k \in [p - \delta, p + \delta] \text{ for any } k \in (0,1)$$

Also,

$$|g(x) - p| = |g(x) - g(p)| = |g'(\xi)|\,|x - p| \leq k\,|x - p| < |x - p|$$
$$p - \delta \leq g(x) \leq p + \delta$$

These conditions guarantee convergence for $p_n \to p$ by F.P.I Theorem. Next, let's prove that the order is α. Let $n = \alpha - 1$,

$$g(x) = g(x_0) + g'(x_0)(x - x_0) + g''(x_0)\frac{(x-x_0)^2}{2!} + \ldots + g^{(\alpha-1)}(x_0)\frac{(x-x_0)^{\alpha-1}}{(\alpha-1)!} + g^{(\alpha)}(\xi(x))\frac{(x-x_0)^{\alpha}}{\alpha!}$$

where $\xi(x)$ is between x_0 and x is a general unknown. Let $x = p_n$ and $x_0 = p$,

$$g(p_n) = p + g^{(\alpha)}(\xi_n)\frac{(p_n - p)^{\alpha}}{\alpha!}$$

where $\xi_n \coloneqq \xi(p_n)$ is between p_n and p. So

$$p_{n+1} = p + g^{(\alpha)}(\xi_n)\frac{(p_n - p)^{\alpha}}{\alpha!}$$

After some manipulation we get

$$\lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_n - p|^{\alpha}} = \lim_{n\to\infty} \left| \frac{g^{(\alpha)}(\xi_n)}{\alpha!} \right|$$

We know $g \in C^{\alpha}([a, b])$, then

$$\lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_- p|^{\alpha}} = \frac{1}{\alpha!} \left| g^{(\alpha)} \left(\lim_{n\to\infty} \xi_n \right) \right|$$

Recall $\xi_n \in [p_n, p]$ or $[p, p_n]$, so $p_n \to p \implies \xi_n$ converges to p.

$$\lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_n - p|^{\alpha}} = \frac{1}{\alpha!} \left| g^{(\alpha)}(p) \right| \coloneqq \alpha \in (0, \infty)$$

Note that from Extreme Value Theorem we know that continuous function in a bounded interval is bounded. □

$g(p_n) = p_{n+1}$ converges with order 2 (or better, if $g''(p) = 0$).

Remark 8.3. Suppose that derivative vanishes at p, i.e., $f'(p) = 0$, then N.M. may

1. not converge at all
2. or converge very slowly (only linearly) depending on initial guess

If $p_n \to p$ and $f'(p) = 0$, then that timplies $f'(p_n) \approx 0$ for n large.

Example 8.4

Consider: $f(x) = x^2 \implies f'(x) = 2x$

$$f(0) = 0, \quad f'(0) = 0$$

0 is a double root of f. Have

$$f'(p) = f(p) = 0$$

and

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

which converges linearly (bad case).

§9 Lec 9: Oct 13, 2021

§9.1 Multiple Roots

Definition 9.1 (Multiple Root) — A root of $f(x) = 0$, p, is called a root of multiplicity m of $f \iff$ for $x \neq p$, there exists decomposition

$$f(x) = (x-p)^m q(x) \text{ where } \lim_{x\to p} q(x) \neq 0$$

If the multiplicity of a root p is 1, then p is called a simple root/zero.

Theorem 9.2

Let $f \in C^m([a,b])$, $p \in [a,b]$. Then p is a root of multiplicity $m \iff$

$$f(p) = f'(p) = f''(p) = \ldots = f^{m-1}(p) = 0 \text{ but } f^{(m)}(p) \neq 0$$

Example 9.3

Consider $f(x) = x^2$, $f'(x) = 2x$, $f''(x) = 2$. So $p = 0$ and $m = 2$.

$$f(x) = (x-0)^2 \cdot 1, \quad q(x) = 1$$

Example 9.4

Consider $f(x) = e^{x^2} - 1$

$$f(0) = 0$$
$$f'(x) = 2xe^{x^2}$$
$$f'(0) = 0$$
$$f''(x) = 2e^{x^2} + 4x^2 e^{x^2}$$
$$f''(0) = 2$$

Have $p = 0$, $m = 2$

$$f(x) = (x-0)^2 \frac{e^{x^2}-1}{x^2}, \quad q(x) = \frac{e^{x^2}-1}{x^2}$$

So

$$\lim_{x\to 0} q(x) = \lim_{x\to 0} \frac{1 + x^2 + \frac{1}{2}x^4 + \frac{1}{6}x^6 + \mathcal{O}(x^8) - 1}{x^2} = 1$$

Question 9.1. How does this relate to N.M?

We know that N.M. fails when $f(p) = 0$ and $f'(p) = 0$. To resolve this, let's introduce $\mu(x) = \frac{f(x)}{f'(x)}$. We have

$$f'(x) = m(x-p)^{m-1} q(x) + (x-p)^m q'(x)$$

So

$$\mu(x) = x - p\frac{q(x)}{mq(x) + (x-p)q'(x)}$$

and $\mu(p) = 0$

$$\frac{q(p)}{mq(p) + (p-p)q'(p)} = \frac{1}{m} \neq 0$$

$\mu(x)$ has root p with multiplicity 1 ($\mu'(p) \neq 0$).
Modified Newton's Method: Given p_0, define

$$\mu(x) \coloneqq \frac{f(x)}{f'(x)}$$

$$p_{n+1} = p_n - \frac{\mu(p_n)}{\mu'(p_n)}$$

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

This allows us to find p without worrying about division by zero. However, the drawback here is we have to compute second derivative...

§9.2 Interpolation

Given n discrete points $(x_0, f(x_0)), (x_1, f(x_1)), \ldots, (x_n, f(x_n))$. We want to find polynomial $P(x)$

$$P(x) = f(x), \quad \text{at } x = x_i, \quad \forall 0 \leq i \leq n$$

Lagrangian polynomials is our solution here. Given $n+1$ data points, these will produce a polynomial of degree n.

Example 9.5

- 1 data point gives a constant function
- 2 data points give a linear function

$$P(x) = f(x_0)\frac{x - x_1}{x_0 - x_1} + f(x_1)\frac{x - x_0}{x_1 - x_0}$$

Clearly, $P(x_0) = f(x_0)$, $P(x_1) = f(x_1)$.

The strategy here is to sum up polynomials so that each piece vanishes at other data points.

$$L_0(x) \coloneqq \frac{x - x_1}{x_0 - x_1}, \quad L_1(x) \coloneqq \frac{x - x_0}{x_1 - x_0}$$

So

$$L_0(x_i) = \delta_{i0}$$
$$L_1(x_i) = \delta_{i1}$$
$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

Then

$$P(x) = f(x_0)L_0(x) + f(x_1)L_1(x)$$

Suppose we have $n+1$ distinct points. Then we define

$$L_i(x) \coloneqq \frac{(x - x_0)(x - x_1)\ldots(x - x_{i-1})(x - x_{i+1})\ldots(x - x_n)}{(x_i - x_0)(x_i - x_1)\ldots(x_i - x_{i-1})(x_i - x_{i+1})\ldots(x_i - x_n)}$$

or more compactly

$$L_i(x) = \prod_{j=0,\, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}, \quad 0 \leq i \leq n, \quad L_i(x_j) = \delta_{ij}$$

Definition 9.6 (Lagrangian Polynomial) — A Lagrangian polynomial of degree n of $f(x)$ is

$$P(x) = \sum_{i=0}^{n} f(x_i) L_i(x)$$

§10 Lec 10: Oct 15, 2021

§10.1 Theoretical Results for Lagrangian Polynomials

Given input data points $\{x_i, f(x_i)\}_{i=0}^n$ we say

$$L_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}, \quad P(x) = \sum_{i=0}^{n} f(x_i) L_i(x)$$

where $P(x)$ is a degree n polynomial.

Example 10.1

Let $f(x) = e^x$, $x_0 = 0$, $x_1 = \frac{1}{2}$, $x_2 = 1$. Then,

$$f(x_0) = 1, \quad f(x_1) = \sqrt{e}, \quad f(x_2) = e$$

So,

$$\begin{aligned} P(x) &= 1 \cdot L_0(x) + \sqrt{e} \cdot L_1(x) + e \cdot L_2(x) \\ &= 1 \cdot \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} + \sqrt{e} \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} + e \cdot \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} \end{aligned}$$

Summing up degree 2 polynomials, the result is a degree 2 polynomial

$$P(1/4) \approx 1.2717$$
$$f(1/4) \approx 1.2840$$

which is roughly 1% error.

In the above example, using more points than $n + 1 = 3$ will result in a better approximation.

Question 10.1. How do we measure error?

First, we need two results from calculus

Theorem 10.2 (Generalized Rolle)

Let $f \in C^n([a, b])$. Suppose $\exists n+1$ distinct roots of f on $[a, b]$. Then $\exists \xi \in (a, b)$ s.t. $f^{(n)}(\xi) = 0$.

This basically says zeros in a function implies a zero of the high-order derivative.

Lemma 10.3

Derivative of Multiplied Monomials

$$\frac{d^{n+1}}{dt^{n+1}}(t - t_0)(t - t_1) \dots (t - t_n) = (n + 1)!$$

Example 10.4

Have

$$\frac{d}{dt}(t-x_0)=1=1!, \quad \frac{d^2}{dt^2}(t-x_0)(t-x_1)=2=2!$$

Induction!

Theorem 10.5 (Error of Lagrangian Polynomial Interpolation)

Let $\{x_0,x_1,\dots,x_n\}\in[a,b]$ be distinct. Let $f\in C^{n+1}([a,b])$, $P(x)=\sum_{i=0}^n f(x_i)L_i(x)$, then $\forall x\in[a,b]$, $\exists\xi(x)\in(a,b)$ s.t.

$$\begin{aligned} f(x) &= P(x)+\frac{f^{(n+1)}(\xi)}{(n+1)!}(x-x_0)(x-x_1)\dots(x-x_n) \\ &= P(x)+\frac{f^{(n+1)}(\xi)}{(n+1)!}\prod_{k=0}^n(x-x_k) \end{aligned}$$

Proof. True if $x=x_i$ since $f(x_i)=P(x_i)$ by construction. So we only deal with $x\neq x_i$. Let x be fixed and define

$$g(t)\coloneqq f(t)-P(t)-(f(x)-P(x))\cdot\prod_{j=0}^n\left(\frac{t-x_j}{x-x_j}\right)$$

We want to apply Generalized Rolle's Theorem on $g(t)$ to claim $g^{(n+1)}(\xi)=0$, and we need to show g is $C^{n+1}([a,b])$ and has $n+2$ distinct roots. Generalized Rolle's Theorem says $g^{(n+1)}(\xi)=0$.

$$\begin{aligned} g^{(n+1)}(t) &= f^{(n+1)}(t)-P^{(n+1)}(t)-(f(x)-P(x))\frac{d^{n+1}}{dt^{n+1}}\prod_{j=0}^n\frac{(t-x_j)}{x-x_j} \\ &= f^{(n+1)}(t)-(f(x)-P(x))\,(n+1)!\prod_{j=0}^n\frac{1}{(x-x_j)} \\ 0=g^{(n+1)}(\xi) &= f^{(n+1)}(\xi)-(f(x)-P(x))\,(n+1)!\prod_{j=0}^n\frac{1}{(x-x_j)} \\ f(x) &= P(x)+\frac{f^{(n+1)}(\xi)}{(n+1)!}\prod_{k=0}^n(x-x_k) \end{aligned}$$

□

Remark 10.6. The pointwise error

$$f(x)-P(x)=\frac{f^{(n+1)}(\xi)}{(n+1)!}\prod_{k=0}^n(x-x_k)$$

In order for it to be useful, we need a bound on $\left|f^{n+1}(\xi)\right|$. And L.P. is unique.

§11 | Lec 11: Oct 18, 2021

§11.1 Neville's Method

Preliminaries: Suppose we have a Lagrangian polynomial from k data points. But now we obtain more information and we want to update $P(x)$'s approximation so some number x.
Neville's Method let's re-use our previous work to update the interpolant. It lets us generate polynomial approximations recursively.

Example 11.1
Given $\{(x_0, f(x_0)), (x_1, f(x_1)), (x_2, f(x_2))\}$,

$$P(x) = f(x_0)\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + f(x_1)\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + f(x_2)\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \quad (*)$$

But we can also build it recursively. Let $P_0 = f(x_0)$, $P_1 = f(x_1)$, $P_2 = f(x_2)$ be the 0th degree polynomials. Then define

$$P_{01}(x) \coloneqq \frac{1}{x-x_0}\left((x-x_0)P_1 - (x-x_1)P_0\right)$$

Similarly, we can form L.P. for $P_{12}(x)$

$$P_{12}(x) \coloneqq \frac{1}{x_2-x_1}\left((x-x_1)P_2 - (x-x_2)P_1\right)$$

Claim 11.1. P_{01} and P_{12} can be combined to form (*) using x_0, x_1, x_2.

$$P(x) = P_{012}(x) \coloneqq \frac{1}{x_2-x_0}\left((x-x_0)P_{12} - (x-x_2)P_{01}\right)$$

Definition 11.2 — Let f defined at points $\{x_i | 0 \le i \le n\}$ and let $m_1, \ldots, m_k \subseteq \{0, 1, 2, \ldots, n\}$ be distinct. Then $P_{m_1 m_2 \ldots m_k}(x)$ is the Lagrangian Polynomial formed by interpolating $f(x)$ at the points $\{x_{m_1}, \ldots, x_{m_k}\}$.

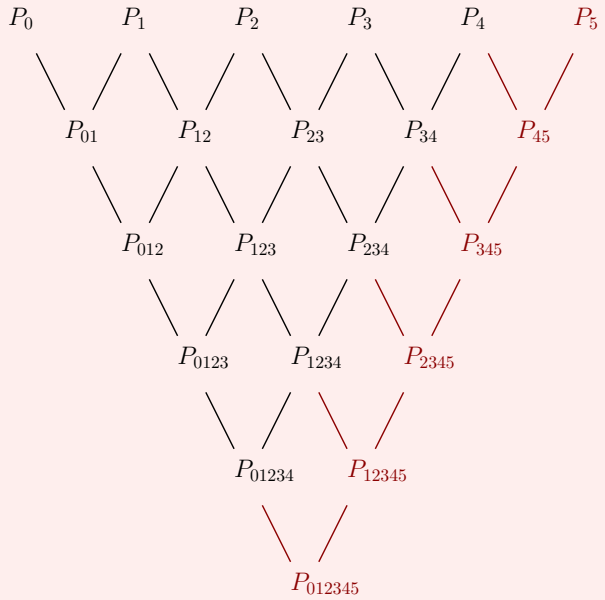
Theorem 11.3
Let f be defined at points $x_0, x_1, x_2, \ldots, x_K$ and let x_i and x_j be distinct points in the set. Then the L.P. that interpolates f at all the $k+1$ points is

$$P(x) = \left[(x-x_j)P_{012\ldots(j-1)(j+1)\ldots k}(x) - (x-x_i)P_{012\ldots(i-1)(i+1)\ldots k}(x)\right]\frac{1}{x_i-x_j}$$

Proof. Verify the interpolation property, the degree, and use the uniqueness of L.P. □

Example 11.4

Given $x_0, x_1, \ldots, x_4$ and $P_0, P_1, \ldots, P_4$ be the constant degree 0 polynomials.



Neville's method is useful when we want to successively generate higher degree polynomial approximation at a specific point.

Example 11.5

Values of various interpolating polynomials at $x = 1.5$

$$x_0 = 1.0, \quad x_1 = 1.3, \quad x_2 = 1.6$$

$$\begin{array}{lll} 1.0 & 0.7651977 & P0 \\ 1.3 & 0.6200680 & P1 \\ 1.6 & 0.4554022 & P2 \\ \boxed{1.9} & \boxed{0.2818186} & P3 \end{array}$$

and

$$\begin{array}{ll} 0.5233449 & P01(1.5) \\ 0.5102968 & P12(1.5) \\ \boxed{0.5132634} & P23(1.5) \end{array}$$

and

$$\begin{array}{ll} 0.5124715 & P012(1.5) \\ \boxed{0.5112857} & P123(1.5) \end{array}$$

and

$$\boxed{0.51181}$$

So $x_3 = 1.9$

§11.2 Divided Differences

Divided difference method is useful for successively generating higher degere polynomial expression (as a function of x).
Let $\{x_0, x_1, \ldots, x_n\}$ be distinct and $P(x)$ is L.P. of $f(x)$.

$$L_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}, \quad P(x) = \sum_{i=0}^{n} f(x_i) L_i(x)$$

We know $P(x)$ is unique, but it can be written in many different ways. One of these ways is called "Newton's Divided Differences", it defines a function looking like

$$P_n(x) \coloneqq a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \ldots + a_n(x - x_0)(x - x_1)\ldots(x - x_{n-1})$$

Observation:

$$\begin{aligned} P_n(x_0) &= a_0 \\ P_n(x_1) &= a_0 + a_1(x_1 - x_0) \\ &\ldots \end{aligned}$$

$P_n(x_k)$ contains the first $k+1$ terms of $P_n(x)$.

Theorem 11.6
$P_n(x) = P(x)$ if a_j's are chosen correctly.

For example, if we want $P_n(x_0) = P(x_0) = f(x_0)$, then $a_0 = f(x_0)$. If we want $P_n(x_1) = P(x_1) = f(x_1)$, then

$$f(x_1) = P_n(x_1) = a_0 + a_1(x_1 - x_0) \implies a_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

Definition 11.7 (Divided Differences) — We introduce notation:

$$f[x_i] = f(x_i) \quad \text{(0th divided differences)}$$

$$f[x_i, x_{i+1}] = \frac{f[x_{i+1}] - f[x_i]}{x_{i+1} - x_i} \quad \text{(first divided differences)}$$

$$f[x_i, x_{i+1}, x_{i+2}] = \frac{f[x_{i+1}, x_{i+2}] - f[x_{i+1}, x_i]}{x_{i+2} - x_i} \quad \text{(second divided differences)}$$

The *kth* divided differences is

$$f[x_i, x_{i+1}, \ldots, x_{i+k}] = \frac{f[x_{i+1}, \ldots, x_{i+k}] - f[x_i, \ldots, x_{i+k-1}]}{x_{i+k} - x_i}$$

It turns out that $P_n(x) = P(x)$ can be achieved by choosing

$$a_k = f[x_0, x_1, x_2, \ldots, x_k]$$

thus, the Newton's Divided Difference way of writing the L.P. is

$$\begin{aligned} P_n(x) = P(x) = f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \ldots \\ + f[x_0, x_1, \ldots, x_n](x - x_0)(x - x_1)\ldots(x - x_{n-1}) \end{aligned}$$

§12 | Lec 12: Oct 20, 2021

§12.1 Divided Differences (Cont'd)

1st order divided difference is highly related to the first derivative. How about the high order ones? How does the k-th order divided difference relate to the k-th derivative?

Theorem 12.1

Suppose $f \in C^n([a,b])$ with $\{x_i\}_{i=0}^n \in [a,b]$ distinct, then $\exists \xi \in (a,b)$ s.t. $f[x_0, x_1, \ldots, x_n] = \frac{f^{(n)}(\xi)}{n!}$

Proof. The proof uses generalized Rolle's theorem and derivative of multiplied monomials (lecture 10).
Let $g(x) \coloneqq f(x) - P_n(x)$

$$P_n(x) = P(x) = f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \ldots$$
$$+ f[x_0, x_1, \ldots, x_n](x - x_0)(x - x_1)\ldots(x - x_{n-1})$$

Then $g(x_i) = f(x_i) - P_n(x_i) = 0$ for $0 \le i \le n$. By Generalized Rolle's Theorem, $\exists \xi \in [a,b]$ s.t. $g^{(n)}(\xi) = 0$.

$$g^{(n)}(x) = f^{(n)}(x) - P_n^{(n)}(x) = f^{(n)}(x) - f[x_0, x_1, \ldots, x_n]\, n!$$

where all terms vanish except the final one, and use the following lemma

Lemma 12.2

$\frac{d^{n+1}}{dt^{n+1}}(t - t_0)(t - t_1)\ldots(t - t_n) = (n+1)!$.

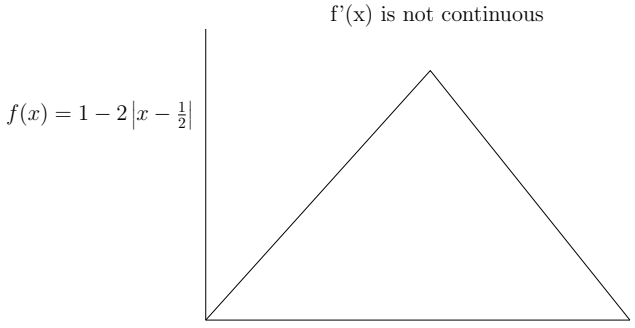
to obtain

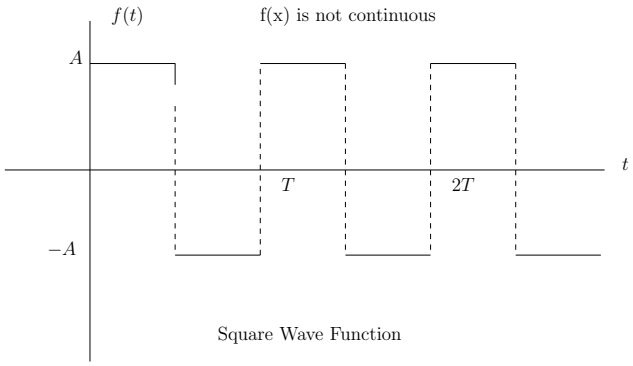
$$g^{(n)}(\xi) = f^{(n)}(\xi) - f[x_0, x_1, \ldots, x_n]\, n! = 0$$

$$\implies \frac{f^{(n)}(\xi)}{n!} = f[x_0, x_1, \ldots, x_n].$$ □

§12.2 Runge's Phenomenon

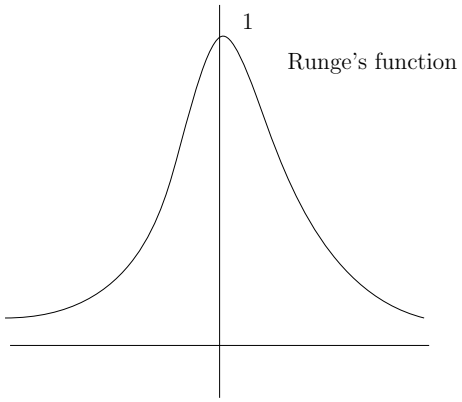
Now let's look into potential challenges with Lagrangian polynomial.





Let's look at a super smooth function instead

$$f(x) = \frac{1}{1 + 25x^2} \in C^{\infty}[-1, 1]$$



If L.P. with equispaced nodes is used: $x_i = x_0 + ih$, $0 \leq i \leq n$, $h = \frac{b-a}{n}$. If we increase n further, oscillation will have higher magnitude. This is called Runge's phenomenon. It was discovered by David Runge (1901) when exploring the behavior of errors when using polynomial interpolation to approximate certain functions. **The discovery was important because it shows that going to higher degrees does not always improve accuracy**.

§13 | Lec 13: Oct 22, 2021

§13.1 High Order Interpolation Issues

Recall the error of Lagrangian polynomial interpolation

$$f(x) = P(x) + \frac{f^{(n+1)}(\xi)}{(n+1)!}\prod_{k=0}^{n}(x - x_k)$$

Remark 13.1. If we can bound the $n+1$ derivatives, $\exists M > 0$ s.t.

$$\max_{a\le x\le b}\left|f^{(n+1)}(x)\right| \le M$$

then

$$|f(x) - P(x)| \le \frac{M}{(n+1)!}\left|\prod_{k=0}^{n}(x - x_k)\right|$$

Suppose now x_i's are equispaced, e.g., $x_i = x_0 + ih$, $h = \frac{b-a}{n}$. Then we can bound

$$\max_{a\le x\le b}\left|\prod_{k=0}^{n}(x - x_k)\right| \le \frac{1}{4}h^{n+1}n!$$

Proof. Skip (Optional reading). □

$$\max_{a\le x\le b}|f(x) - P(x)| \le \frac{M}{(n+1)!}\frac{1}{4}h^{n+1}\cdot n! = \frac{1}{4}\frac{M}{n+1}h^{n+1}$$

For some nice functions, decreasing h (i.e., increasing n) will decrease the error.

Example 13.2

$f(x) = e^{-x}$, $x \in [0,1]$. If we want

$$\max_{0\le x\le 1}|f(x) - P(x)| < 10^{-6}$$

One can show that using the bound, we need at least $n+1 = 7$ data points. Notice that $e^{-x} \in C^{\infty}([0,1])$ and its derivative are bounded.

Let $[a,b] = [-1,1]$

$$\max_{-1\le x\le 1}|f(x) - P(x)| \le \frac{1}{4}\frac{M}{n+1}h^{n+1}, \quad M = \max_{-1\le x\le 1}\left|f^{(n+1)}(x)\right|$$

The trouble is that for $f(x) = \frac{1}{1+25x^2}$, $M \to \infty$ as $n \to \infty$.
Dealing with Runge's Phenomenon: In general, one can

- avoid using equispaced points – we can cleverly choose $\{x_i\}$ to minimize error. chebyshev polynomials of the first kind

$$T_0(x) = 1$$
$$T_1(x) = x$$
$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

For a given positive integer n the Chebyshev nodes in the interval $(-1, 1)$ are

$$x_k = \cos\left(\frac{2k-1}{2n}\pi\right), \quad k = 1, \ldots, n$$

These are the roots of the Chebyshev polynomial of the first kind of degree n. The roots of the polynomials are the projection onto x axis of equal pieces on the circle. The resulting interpolation polynomial minimizes the effect of Runge's phenomenon.

If we cannot easily pick points (e.g., doing an experiment), another option is to approximate $f(x)$ with something other than polynomials, e.g., Fourier (approximating a function with sines and cosines at different frequencies). Another option: use piecewise polynomial approximation or we can use higher order piecewise polynomials. Cubic polynomials are popular. They are often called splines and are very powerful.

Definition 13.3 (Cubic Spline Interpolant) — Given f defined on $[a, b]$, $\{x_j\}_{j=0}^n \in [a, b]$

$$a = x_0 < x_1 < x_2 < \ldots < x_{n-1} < x_n = b$$

The spline is a function $S(x)$ that satisfies

1. On each sub-interval $[x_j, x_{j+1}]$, $j = 0, \ldots, n-1$, $S(x)$ is a cubic polynomials
$$S(x) = S_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3$$
2. $S(x)$ interpolates f at each x_j, i.e., $S(x_j) = f(x_j)$
3. Continuity: $S \in C([a, b])$
4. Differentiability: $S \in C^2([a, b])$.

We can use the properties 2,3,4 and some extra conditions to determine the coefficients in 1.

Example 13.4

$(1, 2)$, $(2, 3)$, and $(3, 5)$.

$$\begin{aligned} [1,2] \quad & S_0(x) = a_0 + b_0(x-1) + c_0(x-1)^2 + d_0(x-1)^3 \\ [2,3] \quad & S_1(x) = a_1 + b_1(x-2) + c_1(x-2)^2 + d_1(x-2)^3 \end{aligned}$$

There are 8 unknowns.

$$\begin{gathered} 2 = f(1) = a_0, \quad 3 = f(2) = a_0 + b_0 + c_0 + d_0, \quad 3 = f(2) = a_1, \\ 5 = f(3) = a_1 + b_1 + c_1 + d_1 \\ S_0'(2) = S_1'(2): \quad b_0 + 2c_0 + 3d_0 = b_1 \\ S_0''(2) = S_1''(2): \quad 2c_0 + 6d_0 = 2c_1 \\ S_0''(1) = 0: \quad 2c_0 = 0 \\ S_1''(3) = 0: \quad 2c_1 + 6d_1 = 0 \end{gathered}$$

So

$$S(x) = \begin{cases} 2 + \frac{3}{4}(x-1) + \frac{1}{4}(x-1)^3, & x \in [1, 2] \\ 3 + \frac{3}{2}(x-2) + \frac{3}{4}(x-2)^2 - \frac{1}{4}(x-2)^3, & x \in [2, 3] \end{cases}$$

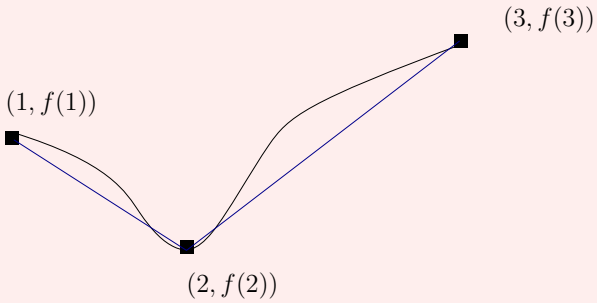
§14 | Lec 14: Oct 27, 2021

§14.1 Cubic Splines

Definition 14.1 (Spline) — A spline is a piecewise defined polynomial.

Example 14.2

Consider a piecewise linear function $L(x)$



where

$$L(x) = \begin{cases} \frac{f(2)-f(1)}{x_2-x_1}(x-x_1)+f(x_1), & x \in [1,2] \\ \dots & x \in [2,3] \end{cases}$$

In general, piecewise linear splines $L(x)$ are simple and powerful. But they are not smooth, i.e., $L(x) \in C([a,b])$ but $L(x) \notin C^1([a,b])$. Note that each piece is a line ($y = mx + b$) which means it has two degrees of freedom: slope m and b. Cubic splines have more degrees of freedom and thus have more regularity (better differentiability properties).

Theorem 14.3

$a = x_0 < x_1 < x_2 < \ldots < x_{n-1} < x_n = b$, $f(x)$ has a unique "natural" spline interpolant on $[a,b]$ for the points $\{x_j\}_{j=0}^n$

$$\text{"natural": } S''(a) = S''(b) = 0$$

Proof. First, by property 1 and 2 in the definition of cubic spline,

$$S_0(x_0) = a_0 = f(x_0)$$
$$S_1(x_1) = a_1 = f(x_1)$$
$$\dots$$
$$S_{n-1}(x_{n-1}) = a_{n-1} = f(x_{n-1})$$

where $S_j(x_h) = a_j = f(x_j)$, $j = 0, 1, \ldots, n-1$ and $a_j = f(x_j)$, $j = 0, 1, 2, \ldots, n$.

$$S_0(x_1) = S_1(x_1)$$
$$S_1(x_2) = S_2(x_2) \dots$$
$$S_{n-2}(x_{n-1}) = S_{n-1}(x_{n-1})$$
$$S_{n-1}(x_n) = f(x_n) =: a_n$$

and

$$\begin{gathered}
h_j = x_{j+1} - x_j \\
a_0 + b_0 h_0 + c_0 h_0^2 + d_0 h_0^3 = S_0(x_1) = a_1 \\
a_1 + b_1 h_1 + c_1 h_1^2 + d_1 h_1^3 = S_1(x_2) = a_2 \\
\cdots \\
a_{n-1} + b_{n-1} h_{n-1} + c_{n-1} h_{n-1}^2 + d_{n-1} h_{n-1}^3 = f(x_n) = a_n
\end{gathered}$$

Thus,

$$a_{j-1} + b_{j-1} h_{j-1} + c_{j-1} h_{j-1}^2 + d_{j-1} h_{j-1}^3 = a_j, \quad j = 1, 2, \ldots, n$$

Similarly, we can deduce from the property 4 on first derivative that

$$b_{j-1} + 2c_{j-1} h_{j-1} + 3d_{j-1} h_{j-1}^2 = b_j, \quad j = 1, 2, \ldots, n$$

and from the property 4 on the second derivative

$$c_{j-1} + 3d_{j-1} h_{j-1} = c_j, \quad j = 1, 2, \ldots, n$$

The natural boundary condition

$$\begin{gathered}
S_0''(x_0) = 0 \\
S_{n-1}''(x_n) = 0
\end{gathered}$$

Summarizing the above

$$\begin{gathered}
a_j = f(x_j), \quad j = 0, 1, 2, \ldots, n \\
b_n = S'(x_n) \\
c_n = S''(x_n)/2 \\
a_j + b_j h_j + c_j h_j^2 + d_j h_j^3 = a_{j+1}, \quad j = 0, 1, 2, \ldots, n-1 \\
b_j + 2c_j h_j + 3d_j h_j^2 = b_{j+1}, \quad j = 0, 1, 2, \ldots, n-1 \\
c_j + 3d_j h_j = c_{j+1}, \quad j = 0, 1, 2, \ldots, n-1 \\
c_0 = c_n = 0
\end{gathered}$$

where the variables are $a_j, b_j, c_j, d_j, a_n, b_n, c_n$, $j = 0, 1, 2, \ldots, n-1$. Note that all a_j values for $j = 0, 1, \ldots, n$ can be determined. □

§15 | Lec 15: Oct 29, 2021

§15.1 Cubic Splines (Cont'd)

Lemma 15.1

If a square $n \times n$ matrix M satisfies $|M_{ii}| > \sum_{j \neq i} |M_{ij}|$ then M is invertible.

Proof. Suppose M is non-invertible which means it's not full rank. Consequently, the null space is at least dimension 1. Thus, $\exists \vec{v} \neq 0$ s.t. $M\vec{v} = 0$. Let's assume $v_i > 0$ has the largest magnitude in $\vec{v}$ (this can always be chosen, because otherwise we could just use $-\vec{v}$ instead as our $\vec{v}$. The ith row of $M\vec{v} = 0$ is then

$$\sum_j M_{ij} v_j = 0 \iff M_{ii} v_i = -\sum_{j \neq i} M_{ij} v_j \iff M_{ii} = -\sum_{j \neq i} M_{ij} \frac{v_j}{v_i}$$

So

$$\implies |M_{ii}| \leq \sum_{j \neq i} \left| M_{ij} \frac{v_j}{v_i} \right| \leq \sum_{j \neq i} |M_{ij}|$$

which is a contradiction! □

Use this lemma and we can prove the theorem in the last lecture.

§15.2 Numerical Differentiation

Recall the definition of derivative

$$f'(x_0) = \lim_{h \to 0} \frac{f(x_0 + h) - f(x_0)}{h}$$

If h is small, then

$$f'(x_0) \approx \frac{f(x_0 + h) - f(x_0)}{h}$$

By Taylor's Theorem, if $f \in C^2([a, b])$, and $x_0, x_1 \in [a, b]$ then

$$f(x_1) = f(x_0) + f'(x_0)(x_1 - x_0) + f''(\xi)\frac{(x_1 - x_0)^2}{2}$$

Let $x_1 = x_0 + h$, then this becomes

$$f(x_0 + h) = f(x_0) + f'(x_0)h + \frac{f''(\xi)h^2}{2}$$

So

$$\frac{f(x_0 + h) - f(x_0)}{h} = f'(x_0) + h\frac{f''(\xi)}{2} \text{ – forward difference formula}$$

If we used x_0 and $x_0 - h$ instead

$$\frac{f(x_0) - f(x_0 - h)}{h} = f'(x_0) + \frac{h}{2} f''(\xi) \quad \text{– backward difference formula}$$

The error is $\frac{h}{2} |f''(\xi)| \leq \frac{h}{2} M$ where $M = \max_{a \leq x \leq b} |f''(x)|$.

Remark 15.2. The error is $\mathcal{O}(h)$ which is not very desirable.

Question 15.1. How to get $\mathcal{O}(h^2)$?

Suppose $f \in C^3([a,b])$, $x_0, x_1 \in [a,b]$ then

$$f(x_1) = f(x_0) + f'(x_0)(x_1 - x_0) + f''(x_0)\frac{(x_1 - x_0)^2}{2} + f'''(\xi)\frac{(x - x_0)^3}{3!}$$

Let $x_1 = x_0 + h$, then

$$f(x_0 + h) = f(x_0) + f'(x_0)h + f''(x_0)\frac{h^2}{2} + f'''(\xi_1)\frac{h^3}{3!}$$

Let $x_1 = x_0 - h$ then

$$f(x_0 - h) = f(x_0) - f'(x_0)h + f''(x_0)\frac{h^2}{2} - f'''(\xi_2)\frac{h^3}{3!}$$

Then, we obtain the centered difference formula

$$\frac{f(x_0 + h) - f(x_0 - h)}{2h} = f'(x_0) + (f'''(\xi_1) + f'''(\xi_2))\frac{h^2}{12}$$

§16 | Lec 16: Nov 1, 2021

§16.1 Richardson Extrapolation

Basic Idea: Generate high accuracy results using low order formulas. Recall for $f \in C^2([a,b])$

$$\frac{f(x_0+h)-f(x_0)}{h} = f'(x_0) + \frac{h}{2}f''(\xi)$$

If $f \in C^3([a,b])$, then

$$\frac{f(x_0+h)-f(x_0)}{h} = f'(x_0) + \frac{h}{2}f''(x_0) + \frac{h^2}{3!}f'''(\xi)$$

where the error is still $\mathcal{O}(h)$ as above but with one more term in Taylor's expansion. TBA

First define a notation for forward difference

$$D_h^+ f(x_0) \coloneqq \frac{f(x_0+h)-f(x_0)}{h}$$

Then,

$$D_{\frac{h}{2}}^+ f(x_0) = \frac{f\left(x_0+\frac{h}{2}\right)-f(x_0)}{\frac{h}{2}}$$

and

$$2D_{\frac{h}{2}}^+ f(x_0) - D_h^+ f(x_0) = \left(2f'(x_0) + \frac{h_1}{2}f''(x_0) + 2\frac{h^2}{4}\frac{1}{3!}f'''(\xi_1)\right) - \left(f'(x_0) + h\frac{1}{2}f''(x_0) + h^2\frac{1}{3!}f'''(\xi_2)\right)$$
$$f'(x_0) + \mathcal{O}(h^2)$$

In summary, we combine two first order formula to get a second order method.
Let M be the true quantity that we want to compute $N(h)$ be the approximation, e.g.,

$$M = f'(x_0), \quad N = D_h^+ f(x_0) = \frac{f(x_0+h)-f(x_0)}{h}$$

Further, assume M can be written as

$$M = N(h) + k_1 h + k_2 h^2 + k_3 h^3 + \dots \tag{*}$$

where k_1, k_2, k_3 are constant independent of h. Then,

$$M = N\left(\frac{h}{2}\right) + k_1\frac{h}{2} + k_2\left(\frac{h}{2}\right)^2 + k_3\left(\frac{h}{2}\right)^3 + \dots \tag{**}$$

Similarly, $2(**) - (*)$

$$M = 2N\left(\frac{h}{2}\right) - N(h) - \underbrace{\frac{1}{2}k_2h^2 - \frac{3}{4}k_3h^3 + \dots}_{\mathcal{O}(h^2)}$$

Question 16.1. What if we have higher order?

For instance,

$$\frac{f(x_0+h)-f(x_0-h)}{2h} = f'(x_0) + \left(f'''(\xi_1)+f'''(\xi_2)\right)\frac{h^2}{12}$$

We can repeat the process, cancel out h^2 terms and get

$$M = \frac{1}{3}\left(4N\left(\frac{h}{2}\right) - N(h)\right) + \mathcal{O}(h^4)$$

Proof. Hw 6. □

§17 | Lec 17: Nov 3, 2021

§17.1 Numerical Quadrature (Integration)

Goal: Approximate $\int_a^b f(x)\,dx$ when no explicit antiderivative $F(x)$ is known.

Example 17.1
Consider $\int_{-1}^{1} e^{-x^2}\,dx$ which we don't know the explicit antiderivative.

Definition 17.2 (Quadrature) — A quadrature formula to approximate $\int_a^b f(x)\,dx$.

$$\begin{cases}\text{nodes } \{x_i\}_{i=0}^n \\ \text{weights } \{w_i\}_{i=0}^n\end{cases}$$

and is given by

$$\sum_{i=0}^{n} w_i f(x_i)$$

To quantify accuracy let's define

Definition 17.3 (Degree of Exactness (D.O.E)) — D.O.E of a quadrature formula is largest non-zero integer N s.t. the quadrature formula is exact for

$$f(x) = x^k, \quad k = 0, 1, \ldots, N$$

i.e., reproducing up to degree N polynomials.

How to derive quadrature formulas? One way to derive quadrature formulas is to approximate $f(x)$?

$$\int_a^b f(x)\,dx = \int_a^b P(x)\,dx + \int_a^b E(x)\,dx$$

$$P(x) = \sum_{i=0}^{n} f(x_i) L_i(x)$$

$$E(x) = \frac{f^{(n+1)}(\xi(x))}{(n+1)!}(x - x_0)\ldots(x - x_n)$$

$$\int_a^b P(x)\,dx = \sum_{i=0}^{n} f(x_i) \int_a^b L_i(x)\,dx$$

Thus,

$$w_i := \int_a^b L_i(x)\,dx$$

We can also compute the error (integral of $E(x)$).

§17.2 Trapezoidal Rule

Idea: Approximate f with a line.

figure here

$f(x)$ defined in $[a,b]$, let $x_0 = a$ and $x_1 = b$. Then we know $P(x) = f(x_0)\frac{x-x_1}{x_0-x_1} + f(x_1)\frac{x-x_0}{x_1-x_0}$

$$\int_a^b f(x)\,dx \approx \int_a^b P(x)\,dx = \sum_{i=0}^{1} w_i f(x_i)$$

where $w_i = \int_{x_0}^{x_1} L_i(x)\,dx$. We can do the integrals to get $w_0 = w_1 = \frac{x_1-x_0}{2} = \frac{h}{2}$. Finally,

$$\int_{x_0}^{x_1} f(x)\,dx \approx \frac{h}{2}\left(f(x_0) + f(x_1)\right)$$

which is indeed the area of a trapezoid. The error in the trapezoidal rule is

$$\int_{x_0}^{x_1} E(x)\,dx = \int_{x_0}^{x_1} \frac{f''(\xi(x))}{2}(x-x_0)(x-x_1)\,dx$$

How do we understand this error? To calculate this, we need

Theorem 17.4 (Weighted Mean Value)

Suppose $f \in C([a,b])$ and the integral of $g(x)$ exists on $[a,b]$. Suppose further that the sign of $g(x)$ does not change on $[a,b]$. Then $\exists c \in (a,b)$ s.t.

$$\int_a^b f(x)g(x)\,dx = f(c)\int_a^b g(x)\,dx$$

Proof. Not covered in this course. □

So, apply the theorem, we obtain

$$\begin{aligned}\int_{x_0}^{x_1} E(x)\,dx &= \frac{f''(c)}{2}\int_{x_0}^{x_1}(x-x_0)(x-x_1)\,dx \\ &= -\frac{f''(c)}{2}\frac{h^3}{6}\end{aligned}$$

Finally,

$$\int_{x_0}^{x_1} f(x)\,dx = \frac{h}{2}\left(f(x_0)+f(x_1)\right) - \frac{f''(c)}{2}\frac{h^3}{6}$$

Note that D.O.E for trapezoidal rule is 1. In practice, we integrate by summing through subintervals

$$\int_a^b f(x)\,dx \approx T_n = \frac{\Delta x}{2}\left[f(x_0) + 2f(x_1) + 2f(x_2) + \ldots + 2f(x_{n-1}) + f(x_n)\right]$$

Example 17.5

Consider $\int_1^5 \frac{1}{x}$ use 4 subintervals

$$\frac{\Delta x}{2}\left[f(x_0) + 2f(x_1) + 2f(x_2) + 2f(x_3) + f(x_4)\right] = \frac{101}{60} = 1.6833333\ldots$$

where the exact answer $\ln(5) - \ln(1) = 1.60943791243$

§17.3 Simpson's Rule

Now with 3 points ($n = 2$)

$$a = x_0, \quad x_1 = \frac{a+b}{2}, \quad x_2 = b$$

Everything stays relatively the same

$$\int_a^b P(x)\,dx = \sum_{i=0}^{2} w_i f(x_i)$$

$$w_i = \int_a^b L_i(x)\,dx$$

$$\int_a^b E(x)\,dx = \int_a^b \frac{f'''(\xi(x))}{3!}(x-x_0)(x-x_1)(x-x_2)\,dx$$

Note that $(x-x_0)(x-x_1)(x-x_2)$ changes sign so we can't apply the weighted mean value theorem here.

§18 | Lec 18: Nov 5, 2021

§18.1 Simpson's Rule & Newton-Cotes

The trapezoidal rule, Simpsons' rule, Newton-Cotes, etc are quadrature rules t approximate $\int_a^b f(x)\,dx$ where $f(x)$ is replaced by a polynomial approximation, e.g., Lagrange polynomial, Taylor polynomial. We can estimate error and easily see degree of exactness.

Example 18.1

Consider

$$\int_a^b f(x)dx = \frac{h}{2}\left(f(a)+f(b)\right) - \frac{f''(c)}{12}h^3$$

for trapezoidal rule ($h = b-a$), DOE: N = 1.

In practice, replacing $f(x)$ with a single low order polynomial across $x = a$ to $x = b$, e.g., with linear, or quadratic, is not sufficient. We should replace $f(x)$ with piecewise polynomials, which is more accurate. Break up $[a, b]$ into a sequence of intervals and approximate $f(x)$ with a polynomial on each one - use splines. The piecewise polynomial approach is called Composite Quadrature Formulas. To analyze the properties of the composite formulas, we still need to understand the properties of the "original" approach. We derived the error for trapezoidal rule using weighted mean value theorem

$$\int_{x_0}^{x_1} f(x)dx = \frac{h}{2}\left(f(x_0)+f(x_1)\right) - \frac{f''(c)}{2}\frac{h^3}{6}$$

Now, let's get into Simpson's rule. The bad derivation uses Lagrange polynomial

$$f(x) = P(x) + E(x)$$

using 3 points $x_0 = a$, $x_1 = a+h$, $x_2 = b$, $h = \frac{b-a}{2}$.

$$\int_{x_0}^{x_2} f(x)dx = \int_{x_0}^{x_2} P(x)dx + \int_{x_0}^{x_2} E(x)dx$$

$$\int_{x_0}^{x_2} P(x)dx = f(x_0)\int_{x_0}^{x_2} L_0(x)dx + f(x_1)\int_{x_0}^{x_2} L_1(x)dx + f(x_2)\int_{x_0}^{x_2} L_2(x)dx$$

$$\int_{x_0}^{x_2} E(x)dx = \int_{x_0}^{x_2} \frac{f'''\left(\xi(x)\right)}{3!}(x-x_0)(x-x_1)(x-x_2)dx$$

Since when $f(x) = x^2$, $f''(x) = 0$, we know DOE $N = 2$. The error is bounded by

$$|\text{error}| \le \left(\max_{a\le x\le b}|f'''(x)|\right)\frac{1}{6}\int_{x_0}^{x_2} \underbrace{|x-x_0|}_{\le 2h}\underbrace{|x-x_1|}_{\le h}\underbrace{|x-x_2|}_{\le 2h}$$

In summary, using Lagrange polynomial for Simpson's rule gives DOE N = 2 and error $\mathcal{O}(h^4)$ (bad derivation).

Remark 18.2. In Hw6, we will use Taylor polynomial to re-derive Simpson's rule with DOE N = 3 and error $\mathcal{O}(h^5)$.

Both results in quadrature formula

$$\int_{x_0}^{x_2} f(x)dx \approx \frac{h}{3}\left(f(x_0)+4f(x_1)+f(x_2)\right)$$

Newton-Cotes: uses $\{x_0, x_1, \ldots, x_{n-1}, x_n\}$ which are $n+1$ pints. It's defined to be the quadrature formula $\sum_{i=0}^{n} w_i f(x_i)$ where $a = x_0 \le x_1 \le \ldots \le x_n = b$ are equispaced and

$$w_i = \int_a^b L_i(x)dx = \int_a^b \prod_{j=0, j\neq i}^{n} \left(\frac{x - x_j}{x_i - x_j}\right) dx$$

Note $f(x) = P(x) + E(x)$ where $P(x) = \sum_{i=0}^{n} f(x_i)L_i(x)$. It's good to know what Newton-Cotes is. But in practice it's not useful – because of Runge's phenomenon.

§18.2 Composite Quadrature Formulas

Similar to splines, in each subinterval we approximate the function with a linear/quadratic function. Then we integrate. Dividing $[a, b]$ into n subintervals of equal width. Let's derive the composite trapezoidal rule. Let $f \in C^2([a, b])$, the error term is

$$\begin{aligned}\int_a^b f(x)\,dx &= \sum_{j=0}^{n-1} \int_{x_j}^{x_{j+1}} f(x)\,dx \\ &= \sum_{j=0}^{n-1} \left(\frac{h}{2} f(x_j) + f(x_{j+1}) - \frac{h^3}{12} f''(\xi_j)\right), \quad \xi_j \in (x_j, x_{j+1})\end{aligned}$$

Thus,

$$\text{C.T.R} = \sum_{j=0}^{n-1} \frac{h}{2}\left(f(x_j) + f(x_{j+1})\right) = \frac{h}{2}\left(f(x_0) + 2\sum_{j=1}^{n-1} f(x_j) + f(x_n)\right)$$

Notice that

$$\text{error} = -\frac{h^3}{12}\sum_{j=0}^{n-1} f''(\xi_j) = \frac{-h^3}{12} n \frac{\sum_{j=0}^{n-1} f''(\xi_j)}{n}$$

$f \in C^2([a, b])$, so $\exists$ a min and max of f'' on $[a, b]$ (by EVT)

$$\min = \min_{a\le x\le b} f''(x) \le f''(\xi_j) \le \max_{a\le x\le b} f''(x) = \max \quad \forall j$$

Summing up we get

$$\min \le \frac{\sum_{j=0}^{n-1} f''(\xi_j)}{n} \le \max$$

By I.V.T, $\exists \mu \in (a, b)$ s.t. $f''(\mu) = \frac{\sum_{j=0}^{n-1} f''(\xi_j)}{n}$

$$\text{error} = -\frac{h^3}{12} n f''(\mu) = -\frac{h^2}{12}\frac{b-a}{n} n f''(\mu) = -\frac{h^2}{12}(b-a) f''(\mu)$$

§19 | Lec 19: Nov 8, 2021

§19.1 Composite Quadrature (Cont'd)

Example 19.1 (CTR)
Consider: $f(x) = e^x$

$$I := \int_0^1 e^x dx = e - 1 \approx 1.7183\ldots$$

$n = 1$ picking $x_0 = 0$, $x_1 = 1$, $h = 1$,

$$I \approx \frac{1}{2}(e^0 + e^1) = 1.8591\ldots$$

$n = 2$,

$$I \approx \frac{1}{4}\left(e^0 + 2e^{\frac{1}{2}} + e^1\right) = 1.7539\ldots$$

$n = 4$,

$$I \approx \frac{1}{8}\left(e^0 + 2e^{\frac{1}{2}} + 2e^{\frac{3}{4}}e^1\right) = 1.7272\ldots$$

$n = 8$,

$$I \approx 1.7205$$

Recall for 3 points x_0, x_1, x_2 (equispaced), in Hw6 Q3 we showed

$$\int_{x_0}^{x_2} f(x)dx = \frac{h}{3}\left(f(x_0) + 4f(x_1) + f(x_2)\right) + \text{error}$$

and for $f \in C^5([x_0, x_2])$ then

$$\text{error} = -\frac{h^5}{12}\left(\frac{1}{3}f^{(4)}(\xi_1) + \frac{1}{5}f^{(4)}(\xi_2)\right)$$

It also can be shown that $\exists \xi \in (x_0, x_2)$

$$\text{error} = -\frac{h^5}{90}f^{(4)}(\xi)$$

The Composite Simpson's Rule (C.S.R) assumes that n is even

Example 19.2
$\{a = x_0, x_2, x_4, x_6, x_8 = b\}$ with $n = 8$

We then use Simpson's rule on each interval $[x_0, x_2]$, $[x_2, x_4]$, $[x_4, x_6]$, $[x_6, x_8]$.

$$\begin{aligned}\int_a^b f(x)dx &= \sum_{j=1}^{\frac{n}{2}} \int_{x_{2j-2}}^{x_{2j}} f(x)dx \\ &= \sum_{j=1}^{\frac{n}{2}} \frac{h}{3}\left(f(x_{2h-2}) + 4f(x_{2j-1}) + f(x_{2j})\right) - \sum_{j=1}^{\frac{n}{2}} \frac{h^5}{90}f^{(4)}(\xi_j)\end{aligned}$$

Error analysis similar to C.T.R

$$\begin{aligned}\text{error} &= \frac{h^5}{90}\frac{n}{2}\sum_{j=1}^{\frac{n}{2}}\frac{f^{(4)}(\xi_j)}{n/2} \\ &= \frac{h^5}{90}\frac{n}{2}f^{(4)}(\zeta) \quad \text{by IVT}\end{aligned}$$

Summary:

- C.T.R formula

$$\int_a^b f(x)dx = \frac{h}{2}\left(f(x_0) + 2\sum_{j=1}^{n-1} f(x_j) + f(x_n)\right) - \frac{h^2}{12}(b-a)f''(\mu)$$

- C.S.R formula

$$\int_a^b f(x)dx = \sum_{j=1}^{\frac{n}{2}} \frac{h}{3}\left(f(x_{2j-2}) + 4f(x_{2j-1}) + f(x_{2j})\right) - \frac{h^4}{180}(b-a)\,f^{(4)}(\zeta)$$

§19.2 Computational Cost Estimate

Suppose we fix an error tolerance τ. For a give numerical quadrature formula and a given $f(x)$, how many points n are required to guarantee that $|\text{error}| < \tau$?

Example 19.3

$f(x) = \frac{1}{x+4}$, $I = \int_0^2 \frac{1}{4+x}dx$. Use Composite Trapezoidal Rule with $\tau = 10^{-5}$

$$|\text{error}| = \frac{h^2}{12}(b-a)\,|f''(\mu)| = \frac{h^2}{12}2\left|\frac{2}{(4+\mu)^3}\right|$$

So,

$$\begin{aligned}|\text{error}| \le \frac{h^2}{6}\frac{2}{64} &< \tau = 10^{-5} \\ h &< 0.04389\end{aligned}$$

$\implies n \ge 46$.
A similar analysis for the C.S.R gives $n \ge 6$.

§19.3 Numerical Stability of Numerical Differentiation and Integration

Theorem 19.4

Composite trapezoidal/Simpson's rule (and other composite quadrature rules) are stable with respect to numerical roundoff errors. Integration is stable.

Theorem 19.5

In contrast, in general numerical differentiation formulas are not stable w.r.t. roundoff errors. Differentiation is unstable.

Let's start with numerical differentiation. Recall

$$x \approx \text{fl}(x)$$

In exact arithmetic, we know

$$f'(x_0) = \frac{f(x_0+h) - f(x_0-h)}{2h} - \frac{h^2}{6} f'''(\xi)$$

Then,

$$\left| f'(x_0) - \frac{f(x_0+h) - f(x_0-h)}{2h} \right| = \frac{h^2}{6} \left| f'''(\xi) \right| \leq \frac{h^2}{6} M$$

where $M = \max_{x \in [a,b]} |f'''(x)|$. Since we are using floating point, let

$$\tilde{f}(x_0+h) \coloneqq \text{fl}\left(f(x_0+h)\right)$$
$$\tilde{f}(x_0-h) \coloneqq \text{fl}\left(f(x_0-h)\right)$$

So

$$f(x_0+h) = \tilde{f}(x_0+h) + \varepsilon_1$$
$$f(x_0-h) = \tilde{f}(x_0-h) + \varepsilon_2$$

Assume h can be represented exactly. Then the floating point approximation to the true $f'(x_0)$ is

$$\frac{\tilde{f}(x_0+h) - \tilde{f}(x_0-h)}{2h} = \frac{f(x_0+h) - f(x_0-h)}{2h} + \frac{\varepsilon_1 - \varepsilon_2}{2h}$$

Error is

$$\begin{aligned} &\left| f'(x_0) - \frac{\tilde{f}(x_0+h) - \tilde{f}(x_0-h)}{2h} \right| \\ \leq &\left| f'(x_0) - \frac{f(x_0+h) - f(x_0-h)}{2h} \right| + \left| \frac{\varepsilon_2 - \varepsilon_1}{2h} \right| \\ \leq &\frac{M}{6} h^2 + \left| \frac{\varepsilon_2 - \varepsilon_1}{2h} \right| \end{aligned}$$

where $\frac{M}{6} h^2$ is the truncation error which decreases as h is small and the second term increases in contrast. To minimize the error, we let

$$\varepsilon \coloneqq \left| \frac{\varepsilon_2 - \varepsilon_1}{2} \right|, \quad g(h) \coloneqq \frac{M}{6} h^2 + \frac{\varepsilon}{h}$$

Then, $g'(h) = \frac{M}{3} h - \frac{\varepsilon}{h^2} = 0$ results in

$$h = \left(\frac{3\varepsilon}{M} \right)^{\frac{1}{3}}$$

This is the optimal choice of h which minimized the error.

§20 | Lec 20: Nov 10, 2021

§20.1 Stability of Numerical Integration

More generally, if an approximation is $\mathcal{O}(h^p)$, e.g. with Richardson extrapolation, the centered difference formula becomes $\mathcal{O}(h^4)$, then the same analysis gives an optimal

$$h \sim \varepsilon^{\frac{1}{p+1}}$$

Question 20.1. How about the stability of integration?

Suppose $f : [a, b] \to \mathbb{R}$, $[a, b]$ divided into n subintervals $[x_j, x_{j+1}]$. Then,

$$\int_a^b f(x)\,dx \approx \frac{h}{3}\left(f(x_0) + 2\sum_{j=1}^{\frac{n}{2}-1} f(x_{2j}) + 4\sum_{j=1}^{\frac{n}{2}} f(x_{2j-1}) + f(x_n)\right)$$

Let $f(x_j) = \tilde{f}(x_j) + \varepsilon_j$. Under floating points, the round off error is

$$E^{R.O} = \frac{h}{3}\left(e_0 + 2\sum_{j=1}^{n/2-1} e_{2j}\right) + 4\sum_{j=1}^{n/2} e_{2j-1} + e_n$$

Then,

$$\left|E^{R.O}(h)\right| \le \frac{h}{3}\left(|e_0| + 2\sum_{j=1}^{\frac{n}{2}-1} |e_{2j}| + 4\sum_{j=1}^{\frac{n}{2}} |e_{2j-1}| + |e_n|\right)$$

Let $\varepsilon = \max |e_j|$, then

$$\varepsilon \le \frac{h}{3}\left(\varepsilon + 2\,(n/2-1)\,\varepsilon + 4(\frac{n}{2})\varepsilon + \varepsilon\right) = nh\varepsilon = \varepsilon(b-a)$$

which is independent of h or n. One can safely decrease h to improve the accuracy numerical integration – stable.

§20.2 Gaussian Quadrature

Recall the general definition of the quadrature formula

$$\sum_{i=1}^{n} w_i f(x_i)$$

where w_i is weight and x_i nodes(?). This gives $2n$ degrees of freedom. The main idea for Gaussian Quadrature (G.Q.) is given n, maximize the degree of exactness (D.O.E). Consider

$$\int_a^b f(x)dx = \sum_{i=1}^{n} w_i f(x_i) \qquad (*)$$

We want this equality to be exact for $f(x)$ being polynomials. We want as high degree polynomials as possible. We get to choose $2n$ numbers on the right, so we can hope that it is good for polynomials that contain $2n$ coefficients, which is at most degree $2n-1$ polynomials.

Assumption: The interval of integration $[a, b]$ is assumed to be $[-1, 1]$. Note that quadrature formulas can be generalized to arbitrary $[a, b]$ using u-substitution by

$$x = \frac{b-a}{2}t + \frac{b+a}{2}$$
$$dx = \frac{b-a}{2}dt$$

Question 20.2. How to find $\{w_i\}$ and $\{x_i\}$?

- Option 1: brute force, e.g., consider two nodes and weights: x_1, x_2, w_1, w_2 solving

$$\int_{-1}^{1} x^k dx = w_1 x_1^k + w_2 x_2^k, \quad k = 0, 1, 2, 3$$

  when n is large we don't want to do this.

- Option 2: use orthogonal polynomials

Definition 20.1 (Orthogonal Polynomial) — If f and g are functions on $[-1, 1]$ then an inner product can be defined

$$\langle f, g \rangle \coloneqq \int_{-1}^{1} f(x)g(x)\,dx$$

This is called an L^2 inner product. If $\langle f, g \rangle = 0$ for non-zero f, g, then $f \perp g$.

$\exists n+1$ polynomials $\{q_i\}_{i=0}^n$, each q_i is a polynomial of degree i,

$$\int_{-1}^{1} q_i(x)q_j(x)\,dx = \delta_{ij}$$

Lemma 20.2 (Lemma 1)
$\{q_i\}_{i=0}^n$ is a basis for space of polynomials of degree n or less.

Proof. $\{q_i\}_{i=0}^n$ being orthogonal $\implies$ they are linearly independent. Note that $\mathbb{P}^n$ has dimension $n+1$. Counting the degrees of freedom, e.g., cubic polynomial has 4 degree of freedoms. There are $n+1$ of the q_i's thus q_i's are a (orthonormal) basis. □

Lemma 20.3 (Lemma 2)
For each polynomials $p \in \mathbb{P}^{n-1}$, we have $\langle p, q_n \rangle = 0$, where q_n is the last vector in the set $\{q_i\}_{i=0}^n$.

Proof. $\{q_i\}_{i=0}^{n-1}$ is a basis for $\mathbb{P}^{n-1}$ by lemma 1. Thus, $p(x) = \sum_{i=0}^{n-1} c_i q_i(x)$ for some c_i. Therefore, $\langle p, q_n \rangle = \ldots = 0$ (inner product is linear). □

§21 | Lec 21: Nov 12, 2021

§21.1 Gaussian Quadrature (Cont'd)

Theorem 21.1 (Gaussian Quadrature)

Let $\{x_i\}_{i=1}^n$ be the n roots of n degree polynomials $q_n(x)$ where q_n is the last in the set $\{q_i\}_{i=0}^n$. We'll assume they are real and distinct. Let

$$w_i = \int_{-1}^{1} \prod_{j=1, j\neq i}^{n} \frac{x - x_j}{x_i - x_j} dx, \quad i = 1, 2, \ldots, n$$

where the integrand equals to $L_i(x)$ from Lagrangian interpolation polynomial. Then, $\sum_{i=1}^n w_i f(x_i)$ is exact for any $f \in \mathbb{P}^{2n-1}$.

Proof. Consider

1. $f \in \mathbb{P}^{n-1}$, there are n nodes $\{x_i\}_{i=1}^n \implies \exists P(x)$ to interpolate $f(x)$

$$P(x) = \sum_{i=1}^{n} f(x_i) L_i(x)$$

Both $P(x)$ and $f(x)$ are degree $n-1$ polynomials for $\{x_i\}_{i=1}^n$

$$\implies f(x) = \sum_{i=1}^{n} f(x_i) L_i(x) \quad \text{(uniqueness of L.I.P)}$$

Thus,

$$\implies \int_{-1}^{1} f(x)\, dx = \sum_{i=1}^{n} w_i f(x_i)$$

2. Next, assume f is a polynomial of degree $n \leq d \leq 2n-1$. Polynomial long division implies that

$$f(x) = Q(x) q_n(x) + R(x)$$

degree: $f : [n, 2n-1]$, $q_n : n$, $Q : [0, n-1]$, $R : [0, n-1]$

$$f(x_i) = Q(x_i) q_n(x_i) + R(x_i) = R(x_i) \tag{*}$$

By 1.,

$$\int_{-1}^{1} R(x)\, dx = \sum_{i=1}^{n} w_i R(x_i) \tag{**}$$

Then,

$$\begin{aligned} \int_{-1}^{1} f(x)\, dx &= \int_{-1}^{1} Q(x) q_n(x)\, dx + \int_{-1}^{1} R(x)\, dx \\ &= \int_{-1}^{1} R(x)\, dx \\ &= \sum_{i=1}^{n} w_i R(x_i) \\ &= \sum_{i=1}^{n} w_i f(x_i) \end{aligned}$$

□

Question 21.1. How to construct $\{q_1, q_2, \ldots, q_n\}$? Or more general, how to construct an orthonormal basis for a vector space?

$\Longrightarrow$ **Gram-Schmidt**! It's a method for orthonormalizing a set of vectors in an inner product space.

$$\text{proj}_u(\mathbf{v}) = \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\langle \mathbf{u}, \mathbf{u} \rangle} \mathbf{u},$$

Let $\{x_1, x_2, \ldots, x_n\}$ be linearly independent.

- Set $v_1 = x_1$
- For $i = 2, \ldots, n$ set $v_i = x_i - \sum_{j=1}^{i-1} \frac{\langle x_i, x_j \rangle}{\langle v_j, v_j \rangle} v_j$
- For $i = 1, \ldots, n$ normalize $q_i = \frac{v_i}{\|v_i\|}$ where $\|v_i\| = (\langle v_i, v_i \rangle)^{\frac{1}{2}}$.
- Output is $\{q_1, \ldots, q_n\}$ are orthonormal.

Consider $\mathbb{P}^n : \left\{1, x, x^2, \ldots, x^{n-1}, x^n\right\}$

$$\langle f, g \rangle = \int_{-1}^{1} fg \, dx$$

- $P_0(x) = 1$

$$P_1(x) = x - \frac{\langle x, 1 \rangle}{\langle 1, 1 \rangle} 1 = x - \frac{\int_{-1}^{1} x \, dx}{\int_{-1}^{1} dx} = x$$

- $P_2(x)$

$$\begin{aligned} P_2(x) &= x^2 - \frac{\langle x^2, 1 \rangle}{\langle 1, 1 \rangle} 1 - \frac{\langle x^2, x \rangle}{\langle x, x \rangle} x \\ &= x^2 - \frac{1}{3} \end{aligned}$$

- Similarly, $P_3(x) = x^3 - \frac{3}{5}x$
- $P_4(x) = x^4 - \frac{6}{7}x^2 + \frac{3}{35}$

These polynomials are known as Legendre polynomials. For the roots of the polynomial, we just need to look it up.

Theorem 21.2

Let $\{\phi_1, \ldots, \phi_n\}$ be a set of orthogonal polynomials on $[a, b]$, and let each ϕ_k has degree k. Then each ϕ_k has precisely k real roots which are simple.

Example 21.3

Approximate $\int_{-1}^{1} e^x \cos x \, dx$ using Gaussian quadrature with $n = 3$.

$$w_i = \int_{-1}^{1} \prod_{j=1, j \neq i}^{n} \frac{x - x_j}{x_i - x_j} \, dx$$

Search for the roots $r_{n,i}$ from a table and then

$$0.\overline{5} e^{0.774596692} \cos 0.774596692 + 0.\overline{8} \cos 0 + 0.\overline{5} e^{-0.774596692} \cos(-0.774596692) = 1.9333904$$

and the true value of the integral is 1.9334214. The absolute error is less than 3.2×10^{-5}.

§22 | Lec 22: Nov 15, 2021

§22.1 Remarks on Numerical Quadrature

Remark 22.1. If f is smooth, G.Q. with $n = 10, 20, 40, \ldots, \mathcal{O}(100)$ often sufficient.

Remark 22.2. In practice, it's best to use an existing implementation of G.Q. rather than writing one's own.

Remark 22.3. Trapezoidal/Simpson's rule is still quite effective and easy to implement.

Remark 22.4. Another simple/effective technique: interpolate $f(x)$ with cubic spline $s(x)$ and integrate.

Remark 22.5. We do not cover Romberg integration but the basic idea is to use Richardson extrapolation repeatedly.

Remark 22.6. Integrals over unbounded domains – we cannot approximate infinity in a computer. The idea is to transform variables to make integrals bounds finite.

Example 22.7

Consider

$$\int_0^\infty e^{-x^2} dx$$

Let $z = \frac{x}{1+x}$, $z(0) = 0$, and $z(\infty) = 1$.

$$\int_0^\infty e^{-x^2} dx = \int_0^1 \frac{1}{(1-z)^2} e^{-\left(\frac{z}{1-z}\right)^2} dz$$

§22.2 Direct Methods for Solving Linear System of Equations

Matrix equation: for $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{x}, \mathbf{b} \in \mathbb{R}^n$

$$\mathbf{Ax} = \mathbf{b}$$

From this point on, we will assume that $\det(A) \neq 0$. Also, this is equivalent to a linear system

$$\sum_{j=1}^{n} a_{ij} x_j = b_j, \quad 1 \leq i \leq n$$

Question 22.1. Why do we care about solving linear system?

Because they show up nearly everywhere in applied math.

Example 22.8

Solve for coefficients of cubic spline interpolant. Another example is Newton's method in higher dimension than $d = 1$. Third example is to solve (partial) differential equation.

We will transform the original form to upper triangular matrix $\mathbf{Ux} = \mathbf{y}$.

Fact 22.1. Every elementary row operation can be presented by applying an invertible matrix P.

Upper triangular matrices are easy to invert

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}$$

1. Start with the last equation because it has only one unknown.
2. Solve the second from last equation $(n-1)th$ using x_n solved for previously. This solves x_{n-1}.
3. Keep going up

§23 | Lec 23: Nov 17, 2021

§23.1 Gaussian Elimination

These statements are equivalent

- A is invertible
- Determinant of A is non-zero
- Nullspace of A is $\{0\}$ (trivial null space)
- Columns of A are linearly independent
- Rows of A are linearly independent
- and so on

We study direct methods for solving the matrix equation $Ax = b$ for x. Two basic issues in direct methods are

- How do we construct a solution method for solving $Ax = b$?
- What is computational complexity?
- Is the method efficient?

Gaussian Elimination: To find x s.t. $Ax = b$

1. Form augmented matrix $[A|b]$
2. Use row reduction to transform into upper triangular form

$$[A|b] \to [U|y]$$

3. Solve $Ux = y$ using back substitution.

General description:

1. Form an augmented matrix. If $A \in \mathbb{R}^{n\times n}$ and $x, b \in \mathbb{R}^n$ then $[A|b] \in \mathbb{R}^{n\times(n+1)}$
2. Use row reduction to transform into upper triangular form.
   - (non-zero) scalar multiplication: if we scale with zero the matrix is no longer invertible
   - scalar multiplication plus row addition
   - row swap

   These elementary row operations are invertible because they are reversible.

Row reduction on augmented matrix can be represented as

$$\begin{aligned} P_1(Ax) &= P_1 b \\ P_2\left(P_1 Ax\right) &= P_2 P_1 b \\ &\dots \\ P_n P_{n-1} \dots P_2 P_1 Ax &= P_n P_{n-1} \dots P_2 P_1 b \\ Ux &= y \end{aligned}$$

3. Solve $Ux = y$ using back substitution

For a 3×3 matrix A, "knock out" a_{21} and a_{31}. The corresponding E.R.O.s are

$$\lambda E_1 + E_2 \to E_2$$
$$\mu E_1 + E_3 \to E_3$$

where we choose λ, μ to "knock out" a_{21} and a_{31}. To do that we can choose $\lambda = -\frac{1}{a_{11}} a_{21}$, $\mu = -\frac{1}{a_{11}} a_{31}$. Then we just need to knock out the element below the diagonal 22.
Note: This can fail if elements along diagonal are zero. In this case we can pivot to a new diagonal element by performing row swapping (which is an E.R.O.). Fancy word for row swapping is **pivoting**.

§24 | Lec 24: Nov 19, 2021

§24.1 Gaussian Elimination with Pivoting

In general, swapping rows to avoid division by zero is called pivoting.

Theorem 24.1

Let $A \in \mathbb{R}^{n\times n}$, then

$\det(A) \neq 0 \iff$ Gaussian elimination with row interchanges can be performed on A without failure

Example 24.2

consider

$$\begin{pmatrix} 4 & 1 & 1 \\ 1 & \frac{1}{4} & 1 \\ 1 & 1 & 3 \end{pmatrix} \implies \begin{pmatrix} 4 & 1 & 1 \\ 0 & 0 & \frac{3}{4} \\ 0 & \frac{3}{4} & \frac{11}{4} \end{pmatrix} \implies \begin{pmatrix} 4 & 1 & 1 \\ 0 & \frac{3}{4} & \frac{11}{4} \\ 0 & 0 & \frac{3}{4} \end{pmatrix} \text{ – invertible}$$

Consider

$$\begin{pmatrix} 4 & 1 & 1 \\ 1 & \frac{1}{4} & 1 \\ 1 & \frac{1}{4} & 3 \end{pmatrix} \implies \begin{pmatrix} 4 & 1 & 1 \\ 0 & 0 & \frac{3}{4} \\ 0 & 0 & \frac{11}{4} \end{pmatrix} \text{ – stuck with zero diagonals (failure/singular)}$$

Input: invertible matrix $A \in \mathbb{R}^{n\times n}$. For $i = 1, 2, \ldots, n-1$: let $p\,(i \leq p \leq n)$ be the smallest integer s.t. $a_{pi} \neq 0$. If $p \neq i$, perform E.R.O $E_i \leftrightarrow E_p$. For $j = i+1, i+2, \ldots, n$, set $\lambda_{ij} = -a_{ji}/a_{ii}$. Perform E.R.O: $E_j + \lambda_{ji}E_i \to E_j$.
Output: upper triangulate matrix U.

§24.2 Computational Complexity of G.E.

Cost of upper-triangularization: To transform A to U, the answer is: $\frac{n^3}{3} + n^2 - \frac{n}{3}$ multiplication/divisions $\frac{n^3}{3} + \frac{n^2}{2} - \frac{5n}{6}$ additions/subtractions. Remember that $\frac{2}{3}n^3$ FLOPs for large n.
Cost of back-substitution: Let U be a upper triangular, then solving $Ux = y$ with back substitution requires $\approx n^2$ FLOPs.
Recall two types of red flags:

1. Division by small numbers
2. Subtracting two numbers that are close

In Gaussian elimination, we are essentially doing both.

Remark 24.3. This strategy is called partial pivoting. There exists other strategies such as scaled partial pivoting and complete pivoting.

§24.3 Matrix Decomposition

Eigenvalue Decomposition: Until specified later, we assume now we are using exact arithmetic. Recall from linear algebra, $A \in \mathbb{R}^{n\times n}$ is called normal if it commutes with its transpose:

$$AA^\top = A^\top A$$

Theorem 24.4

If $A \in R^{n\times n}$ is normal then

$$A = UDU^\top \qquad (*)$$

where $D = \text{diag}(\lambda_i)$ is diagonal and U is orthogonal ($U^{-1} = U^\top$).

LU Decomposition: Recall that for Gaussian Elimination, row reduction can be represented by E.R.O that can be represented as multiplications of matrices. Thus, G.E. is equivalent to doing

$$P_{n-1}P_{n-2}\dots P_3P_2P_1A = U$$

Here are some facts about the E.R.O. matrices

Fact 24.1.

1. Each P_j is invertible, because we can always undo E.R.O.s.
2. If no row swapping is performed ,then each P_j is lower triangular.
3. The inverse of a lower triangular matrix is lower triangular (same holds for upper triangular matrix).
4. If L_1 and L_2 are both lower triangular, then their product is also lower triangular (same holds for upper triangular)

Therefore, $P_{n-1}P_{n-2}\dots P_3P_2P_1$ is lower triangular.

Let $L^{-1} = P_{n-1}P_{n-2}\dots P_3P_2P_1$. So $A = LU$.