Things to Remember

- a_{ij} is the ith row and jth column of A.
- $\pm 0.d_1 d_2 \ldots d_k \times 10^n$ is the decimal floating point representation of a number.
- Chopping is cheaper than rounding.

Error

- Error: $p - \hat{p}$
- Abs. Err: $|p - \hat{p}|$
- Rel. Err: $\frac{|p-\hat{p}|}{p}$ (for accuracy)

Significant Digits

An approximation $\hat{p}$ has t significant digits if:
$$\frac{|p-\hat{p}|}{|p|} \leq 5 \times 10^{-t}$$

Catastrophic Cancellation (Roundoff)

When subtracting nearly equal numbers, the relative error is large, and you lose a lot of significant digits (and accuracy).

How to Reduce Errors

- Reformat the formula to avoid roundoff
- Reduce num. of ops (avoid rounding)
  - Nested Arithmetic: Rewrite polynomials to reduce operations
    $x^3 - 6.1x^2 + 3.2x \to ((x-6.1)x + 3.2)x$

Alternative Quadratic Formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} = \frac{-2c}{b \mp \sqrt{b^2-4ac}}.$$

mind the signs

use the alt. form when $|-b|$ is close to $+\sqrt{b^2-4ac}$

Algorithms and Convergence

- Stable $\to$ errors grow linearly
- Unstable $\to$ errors grow exponentially

Rate of Convergence

- For sequences, if $\alpha_n \to \alpha$ and $|\alpha_n - \alpha| \leq k\beta_n$, $\beta_n \to 0$ then α_n is $\mathcal{O}(\beta_n)$
- For functions, if $\lim_{h\to 0} f(h) = L$ and $|f(h)| \leq kh^p$ then $f(h) = L + \mathcal{O}(h^p)$

Taylor Series

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n$$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!}$ $\quad \cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!}$

$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} + \cdots$

$(1+x)^{-p} = 1 - px + \frac{p(p+1)x^2}{2} - \frac{p(p+1)(p+2)x^3}{3!}$

The Error Term is the $(n+1)^{th}$ term.

Root Finding

- Find p such that $f(p) = 0$.

Generic Stopping Criterion

1. $\frac{|p_n - p_{n-1}|}{|p_n|} \leq \mathcal{E}; p_n \neq 0$: relative error
2. $|f(p_n)|| \leq \mathcal{E}$
   - Ensures small $f(p_n)$
   - p_n may differ significantly from p
3. Have a fixed number of iterations
4. (bisection) $\frac{b_n - a_n}{2} \leq \mathcal{E}$ **or** $|p_n - p_{n-1}| < \mathcal{E}$
   - Ensures p_n is within $\mathcal{E}$ of p
   - Does not ensure small $f(p_n)$

Bisection Method:

- **Conditions:** $f(x) \in C[a,b]$;
  $f(a)$ and $f(b)$ have opposite signs.
- **Midpoint:** $x = \frac{a+b}{2}$
- **Procedure:** Binary search for the root.
- **Error:** Guaranteed quadratic convergence
- **Error Formula:** $\frac{b-a}{2^n}$

Newton's Method

- Faster than bisection, quadratic. We follow the tangent line at p_{n-1} to its $x-$intercept.
- Requires $f'(p)$ to exist.
- Requires $f''(p)$ for quadratic convergence.
  1. Start with initial guess p_0 and p_1
  2. $p_n = p_{n-1} - \dfrac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f'(p_{n-1})f(p_{n-2})}$

Secant Method

- Does not require $f'(p)$ to exist.
- Faster than Bisection, order $\phi \approx 1.618$
  1. Start with initial guess p_0 and p_1
  2. $p_n = p_{n-1} - \dfrac{f(p_{n-1})(p_{n-1}p_{n-2})}{f(p_{n-1})f(p_{n-2})}$

Fixed Points

1. Start with initial guess p_0
2. Generate a sequence $p_n = g(p_{n-1})$
3. Stop when $|p_n - p_{n-1}| < \mathcal{E}$

- A fixed point of f is a point p such that $f(p) = p$.
- Converges if:
  1. $g : [a,b] \to [a,b]$ is continuous
  2. $\forall x \in [a,b] : |g'(x)| \leq k < 1$
  3. $f(x) = 0$ can be rewritten as $g(x) = x$
- **Error**: $\mathcal{O}(q^n)$, for some q, faster when q is small

Norms

Vector Norms

- $l_1 : \|x\|_1 = \sum x_i$
- $l_2 : \|x\|_2 = \sqrt{x_1^2 + \cdots + x_n^2}$ (Euclidean)
- $l_\infty : \|x\|_\infty = \max\{|x_1|, \cdots, |x_n|\}$ (∞)

Properties

- Scalability: $\|\alpha x\| = |\alpha|\|x\|$
- Triangle Inequality: $\|x+y\| \leq \|x\| + \|y\|$

Vector Distances

- l_α distance: $\|x-y\|_\alpha$

Matrix Norms

- The Natural Norm $\|\cdot\|_*$ for $A, B \in \mathbb{R}^{n\times n}; \alpha \in \mathbb{R}$ is defined as a function that satisfies:
  1. $\|A\| \geq 0$
  2. $\|A\| = 0 \iff A = 0$
  3. $\|\alpha A\| = |\alpha|\|A\|$
  4. $\|A + B\| \leq \|A\| + \|B\|$
- **Def.** $\|A\|_* = \max_{\|x\|=1} \|Ax\|_*$ where $\|Ax\|$ is any vector norm.
- $l_\infty : \|A\|_\infty = \max_{1\leq i \leq n} \sum_{j=1}^n |a_{ij}|$ (row sum)

Special Properties

1. For any natural norm $\|\cdot\|_\alpha : \rho(A) \leq \|A\|_\alpha$
2. For $l_2 : \|A\|_2 = \sqrt{\rho(A^TA)}$

Vector Sequence Convergence

- $\{x^{(k)}\}$ converges to x for any small $\mathcal{E} > 0$ eventually every $x^{(k)}$ is within $\mathcal{E}$ of x

Eigenvalues and Eigenvectors

E.value (λ): Scalar s.t. $A\vec{x} = \lambda\vec{x}$
E.vector $(\vec{x})$: Nonzero vector only scaled by A **Spectral Radius**: $\rho(A) = \max\{|\lambda_i|\}$

Properties

1. $\det(A - \lambda I) = 0 \iff \lambda$ is an eigenvalue. Solve the characteristic polynomial for λ.
2. $\forall\lambda[(A - \lambda I)\vec{x} = 0 \iff \vec{x}$ is an eigenvector$]$
3. If $\rho A < 1$, A is convergent $\implies \lim_{k\to\infty} A^k = 0$

Linear Systems - Pivoting Strategies

If the pivot is small, large errors can occur. Pivoting helps maintain numerical stability.

Partial Pivoting

Choose the largest element in the **current column** (below or at the pivot) to avoid dividing by a small number.

1. For $k = 1 \ldots n-1$:
   - Find $r = \arg\max_{k\leq i\leq n}\{|a_{ik}|\}$
   - If $r \neq k$, swap rows: $E_k \leftrightarrow E_r$
   - Continue Gaussian Elimination as usual

Scaled Partial Pivoting

Handles rows with vastly different magnitudes by normalizing.

1. For each row $i = 1 \ldots n$, compute the scale factor: $s_i = \max_j |a_{ij}|$
2. For pivot column k, choose the row r such that $\frac{|a_{rk}|}{s_r}$ is maximal for $r \geq k$
3. If $r \neq k$, swap rows: $E_k \leftrightarrow E_r$
4. Proceed with Gaussian Elimination

Full Pivoting

Most stable but most expensive. Swap both rows and columns.

1. At step k, find the largest element $|a_{ij}|$ in the submatrix $A_{k:n,k:n}$
2. Swap row k with row i, and column k with column j
3. Update row and column permutations
4. Continue Gaussian Elimination

Linear Algebra

- To multiply $A \cdot B$, dot-product the rows of A by the columns of B.
- $AA^{-1} = A^{-1}A = I$
- To find A^{-1}, row reduce the aug. matrix $[A|I]$.
- A^T is A flipped over the main diagonal.

Determinant

- $\det(A) \neq 0 \implies \begin{cases} A^{-1} & \text{exists} \\ Ax = b & \text{has a unique solut} \end{cases}$
- **Cofactor Expansion (Laplace Expansion)**: $\det(A) = \sum_{j=1}^n a_{ij}(-1)^{i+j}\det(A_{ij})$

Matrix Factorization

LU Decomposition

If Gaussian elimination can be performed without row exchanges: $A = LU$, where L is lower triangular with unit diagonal entries and U is upper triangular.

To solve $Ax = b$:

1. Solve $Ly = b$ via forward substitution.
2. Solve $Ux = y$ via backward substitution.

Cost: $O(n^3)$ for factorization, $O(n^2)$ per solve.

Row Swaps: If row swaps are needed, introduce a permutation matrix $P : PA = LU \Rightarrow A = P^{-1}LU$, Then solve: $LUx = Pb$

Special Matrices

Permutation Matrices

- Formed by permuting rows of I_n, So there is exactly one entry of 1 per row and column.
- $P^{-1} = P^\top$
- PA permutes rows of A.

Singular

- A matrix A is singular if $\det(A) = 0$.
- Not invertible; $Ax = b$ has either no solution or infinitely many.

Banded Matrices

- Nonzero entries confined to a diagonal band.
- If $|i - j| > w \Rightarrow a_{ij} = 0$, bandwidth $= w$.
- Common in finite difference methods and sparse linear systems.

Tridiagonal Matrices

- Banded matrix with $w = 1$ (main ± 1 diagonals).
- Nonzero entries only on the main diagonal and the first sub/super diagonals.

Diagonally Dominant (DD / SDD)

- A is strictly diagonally dominant if: $|a_{ii}| > \sum_{j \neq i} |a_{ij}| \quad \forall i$
- A is weakly diagonally dominant of $|a_{ii}| \geq ...$
- Guarantees LU factorization without row swaps.
- Guaranteed convergence of Jacobi and G-S.

Symmetric Positive Definite (SPD)

- A is positive definite if $\forall x \neq 0 : x^T A x > 0$
- All eigenvalues are positive.
- All leading principal minors are positive. $\forall k \det(A_{1:k,1:k}) > 0$
- Cholesky factorization: $A = LL^T$ lets us solve $Ax = b$ in $O(n^2)$ time.
- Also: $A = LDL^T$

Iterative Methods for Linear Systems

Convergent Matrix Theorem

The following statements are equivalent:

(i) A is convergent
(ii) $\rho(A) < 1$ (nec + suf for Jacobi and G-S)
(iii) $\forall x : \lim_{n\to\infty} A^n x = 0$
(iv) $\forall \alpha : \lim_{n\to\infty} \|A^n\|_\alpha = 0$

Jacobi Method $A = D + L + U$

$$x^{(k+1)} = \underbrace{D^{-1}(L+U)}_{T_J} x^{(k)} + \underbrace{D^{-1}b}_{C_J}$$

- Requires $a_{ii} \neq 0$. Always permute so a_{ii} big.
- Uses previous iteration values for all components.
- Converges if A **strictly** diagonally dominant or SPD.

Gauss-Seidel Method $A = D + L + U$

$$x^{(k+1)} = \underbrace{(D+L)^{-1}U}_{T_{GS}} x^{(k)} + \underbrace{(D+L)^{-1}Lb}_{C_{GS}}$$

- Iteration uses most recent updates:
- Often converges faster than Jacobi.
- Also converges under **strict** diagonal dominance or SPD.

Numerical Interpolation

Lagrange Interpolation

Constructs a polynomial $P(x)$ of degree $\leq n$ through points $(x_0, y_0), \ldots, (x_n, y_n)$:

$$P(x) = \sum_{j=0}^{n} y_j L_j(x)$$
$$L_j(x) = \prod_{\substack{0 \leq i \leq n \\ i \neq j}} \frac{x - x_i}{x_j - x_i}$$

Error: If $f \in C^{n+1}[a,b]$, then

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

for some $\xi \in [a,b]$

Newton's Divided Differences

Efficient and updatable polynomial form:

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

Recursive definition:

- zeroth: $f[x_0] = f(x_0)$
- first: $f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$
- kth: $f[x_i, \ldots, x_{i+k}] = \frac{f[x_{i+1}, \ldots, x_{i+k}] - f[x_i, \ldots, x_{i+k-1}]}{x_{i+k} - x_i}$

Neville's Method

Recursive algorithm to evaluate $P(x)$ at a point:

$$P_{i,j}(x) = \frac{(x - x_i)P_{i+1,j}(x) - (x - x_j)P_{i,j-1}(x)}{x_j - x_i}$$

Returns $P(x)$ only — not the polynomial form.

Hermite Interpolation

Matches both values and derivatives: - Duplicate nodes in divided difference table. - Derivative at a node: $f[x_i, x_i] = f'(x_i)$.

Cubic Spline Interpolation

Piecewise cubic $S_i(x)$ defined on $[x_i, x_{i+1}]$:

- $S(x)$, $S'(x)$, and $S''(x)$ are continuous.
- **Natural spline:** $S''(x_0) = S''(x_n) = 0$.
- Solve a tridiagonal linear system for coefficients.

Parametric Curves

For 2D/3D data: interpolate $x(t)$, $y(t)$, $z(t)$ independently. Used in animation and CAD. Preserves geometric continuity.

Numerical Integration

Trapezoidal Rule

- Approximates $f(x)$ with a linear polynomial over $[a,b]$:

$$\int_a^b f(x)\,dx \approx \frac{h}{2}[f(x_0) + f(x_1)]$$

- Composite version over n subintervals ($h = \frac{b-a}{n}$):

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

- Error: $-\frac{(b-a)^3}{12n^2} f^{(2)}(\xi)$ for some $\xi \in [a,b]$

Simpson's Rule

- Approximates $f(x)$ with a quadratic polynomial:

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

- Composite version (even n):

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

- Error: $-\frac{(b-a)^5}{180n^4} f^{(4)}(\xi)$ for some $\xi \in [a,b]$

ODE Initial Value Problems

Euler's Method

$$w_{n+1} = w_n + hf(t_n, w_n)$$

Error: $O(h)$

Modified Euler Method (Heun's)

$$w_{n+1} = w_n + \frac{h}{2}[f(t_n, w_n) + f(t_{n+1}, w_n + hf(t$$

Error: $O(h^2)$

Midpoint Method

$$w_{n+1} = w_n + hf\left(t_n + \frac{h}{2}, w_n + \frac{h}{2}f(t_n, w_n)\right)$$

Error: $O(h^2)$

Runge-Kutta Method (RK4)

$$k_1 = hf(t_n, w_n)$$
$$k_2 = hf\left(t_n + \frac{h}{2}, w_n + \frac{k_1}{2}\right)$$
$$k_3 = hf\left(t_n + \frac{h}{2}, w_n + \frac{k_2}{2}\right)$$
$$k_4 = hf(t_n + h, w_n + k_3)$$
$$w_{n+1} = w_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

Error: $O(h^4)$