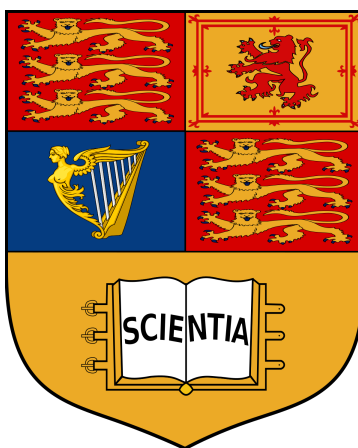


# Numerical Analysis - Concise Notes

MATH50003

Arnav Singh



Colour Code - **Definitions** are **green** in these notes, **Consequences** are **red** and **Causes** are **blue**

*Content from MATH40005 assumed to be known.*

Mathematics  
Imperial College London  
United Kingdom  
March 15, 2022

# Contents

|          |   |          |
|----------|---|----------|
| <b>I</b> | <b>Linear Algebra</b>                                 | <b>3</b> |
| 1        | Prelim  | 3        |
| 3        | Algebraic and Geometric multiplicities of eigenvalues | 4        |
| 4        | Direct Sums   | 4        |
| 5        | Quotient Spaces                                       | 5        |
| 6        | Triangularisation                                     | 5        |
| 7        | The Cayley-Hamilton Theorem                           | 5        |
| 8        | Polynomials   | 5        |
| 9        | The minimal polynomial of a linear map                | 6        |
| 10       | Primary Decomposition                                 | 7        |
| 11       | Jordan Canonical Form                                 | 7        |
| 12       | Cyclic Decomposition & Rational Canonical Form        | 9        |
| 13       | The Dual Space  | 10       |
| 14       | Inner Product Spaces                                  | 11       |
| 15       | Linear maps on inner product spaces                   | 13       |
| 16       | Bilinear & Quadratic Forms                            | 14       |

|           |  |           |
|-----------|--|-----------|
| <b>I</b>  | <b>Computing with Numbers</b>                            | <b>18</b> |
| <b>1</b>  | <b>Numbers</b>   | <b>18</b> |
| 1.1       | Binary Representation . . . . .                          | 18        |
| 1.2       | Integers . . . . .                                       | 18        |
| 1.2.1     | Signed Integer . . . . .                                 | 18        |
| 1.2.2     | Variable bit representation . . . . .                    | 18        |
| 1.2.3     | Division . . . . .                                       | 18        |
| 1.3       | Floating Point numbers . . . . .                         | 19        |
| 1.3.1     | IEEE Floating-point numbers . . . . .                    | 19        |
| 1.3.2     | Special normal numbers . . . . .                         | 19        |
| 1.3.3     | Special Numbers . . . . .                                | 20        |
| 1.4       | Arithmetic . . . . .                                     | 20        |
| 1.5       | Bounding errors in floating-point arithmetic . . . . .   | 20        |
| 1.5.1     | Arithmetic and Special numbers . . . . .                 | 21        |
| 1.5.2     | Special functions . . . . .                              | 21        |
| 1.6       | High-precision floating-point numbers . . . . .          | 21        |
| <b>2</b>  | <b>Differentiation</b>                                   | <b>21</b> |
| 2.1       | Finite-differences . . . . .                             | 21        |
| 2.1.1     | Bounding the error . . . . .                             | 22        |
| 2.2       | Dual numbers . . . . .                                   | 22        |
| 2.2.1     | Connection with differentiation . . . . .                | 22        |
| <b>II</b> | <b>Computing with Matrices</b>                           | <b>23</b> |
| <b>3</b>  | <b>Structured Matrices</b>                               | <b>23</b> |
| 3.1       | Dense vectors and matrices . . . . .                     | 23        |
| 3.2       | Triangular Matrices . . . . .                            | 24        |
| 3.3       | Banded Matrices . . . . .                                | 24        |
| 3.4       | Permutation Matrices . . . . .                           | 25        |
| 3.5       | Orthogonal Matrices . . . . .                            | 25        |
| 3.5.1     | Simple Roations . . . . .                                | 25        |
| 3.5.2     | Reflections . . . . .                                    | 26        |
| <b>4</b>  | <b>Decompositions and Least Squares</b>                  | <b>26</b> |
| 4.1       | QR and least squares . . . . .                           | 27        |
| 4.2       | Reduced QR and Gram-Schmidt . . . . .                    | 27        |
| 4.2.1     | Computing QR decomposition . . . . .                     | 27        |
| 4.3       | Householder reflections and QR . . . . .                 | 28        |
| 4.4       | PLU Decomposition . . . . .                              | 29        |
| 4.4.1     | Special "one-column" Lower triangular matrices . . . . . | 29        |
| 4.4.2     | LU Decomposition . . . . .                               | 29        |
| 4.4.3     | PLU Decomposition . . . . .                              | 30        |
| 4.5       | Cholesky Decomposition . . . . .                         | 30        |
| 4.6       | Timings . . . . .  | 31        |
| <b>5</b>  | <b>Singular Values and Conditioning</b>                  | <b>31</b> |
| 5.1       | Vector Norms . . . . .                                   | 31        |
| 5.2       | Matrix Norms . . . . .                                   | 31        |
| 5.3       | Singular Value Decomposition . . . . .                   | 32        |
| 5.4       | Condition numbers . . . . .                              | 32        |
| <b>6</b>  | <b>Differential equations via Finite differences</b>     | <b>33</b> |
| 6.1       | Indefinite integration . . . . .                         | 33        |
| 6.2       | Forward Euler . . . . .                                  | 34        |
| 6.3       | Backward Euler . . . . .                                 | 35        |
| 6.4       | Systems of equations . . . . .                           | 35        |
| 6.5       | Nonlinear problems . . . . .                             | 35        |
| 6.6       | Two-point boundary value problem . . . . .               | 35        |
| 6.7       | Convergence . . . . .                                    | 35        |
| 6.7.1     | Poisson . . . . .  | 36        |

## Part I

# Linear Algebra

## 1 Prelim

### Definition - Similair Matrices

$A, B \in M_n(F)$  similair ( $A \sim B$ ) if  $\exists$  invertible  $P \in M_n(F)$  s.t  $P^{-1}AP = B$

$\sim$  is an equivalence relation.

*Properties of Similair Matrices*

- Same Determinant
- Same Char. Poly.
- Same eigenvalues
- Same rank Same Trace

### Definition - Companion Matrix

Let  $p(x)$  a monic polynomial of degree  $r$ ;  $p(x) = x^r + a_{r-1}x^{r-1} + \dots + a_0$ .

Companion matrix of  $p(x)$ ;

$$C(p(x)) = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & -a_0 \\ 1 & 0 & 0 & \dots & 0 & -a_1 \\ 0 & 1 & 0 & \dots & 0 & -a_2 \\ & & & \dots & & \\ 0 & 0 & 0 & \dots & 1 & -a_{r-1} \end{pmatrix}$$

### Geometry

### Definition - Dot Product

$u = (u_1, \dots, u_n)$  and  $v = (v_1, \dots, v_n)$

$$u \cdot v = \sum_{i=1}^n u_i v_i$$

Length of  $u$ ,  $\|u\| = \sqrt{u \cdot u}$

Distance between  $u$  and  $v = \|u - v\|$

- $P$  orthogonal if  $P^T P = I$ ,  $(Pu \cdot Pv) = u \cdot v$
- $A$  symmetric if  $A^T = A$ ,  $(Au \cdot v = u \cdot Av)$

*Properties of dot product*

- linear in  $u, v$
- symmetric;  $u \cdot v = v \cdot u$
- $u \cdot v > 0, \forall u, v$

### 3 Algebraic and Geometric multiplicities of eigenvalues

#### Definition - Multiplicity of eigenvalues

For  $T : V \rightarrow V$  a linear map with char. poly.  $p(x)$  with roots  $\lambda$ , Then  $\exists a(\lambda) \in \mathbb{N}$  the **algebraic multiplicity** of  $\lambda$  s.t

$$p(x) = (x - \lambda)^{a(\lambda)} q(x)$$

where  $\lambda$  not a root of  $q(x)$

**Geometric multiplicity**  $g(\lambda) = \dim E_\lambda$ , for  $E_\lambda$  the eigenspace of  $T$

#### Theorem 3.2

$\dim V = n$ , Let  $T : V \rightarrow V$  a linear map with finite distinct eigenvalues  $\{\lambda_i\}_{i=1}^r$

Characteristic polynomial of  $T$  is

$$p(x) = \prod_{i=1}^r (x - \lambda_i)^{a(\lambda_i)}$$

so  $\sum_{i=1}^r a(\lambda_i) = n$ . Following are equivalent

- $T$  diagonalisable
- $\sum_{i=1}^r g(\lambda_i) = n$
- $g(\lambda_i) = a(\lambda_i) \forall i$  (This can be used to test for diagonalisability.)

### 4 Direct Sums

#### Define

For  $\{U_i\}_{i=1, \dots, k}$  subspaces of vector space  $V$ . Sum of these subspaces is:

$$U_1 + \dots + U_k = \{u_1 + \dots + u_k : u_i \in U_i, \forall i\}$$

#### Definition - Direct Sums

$V$  a vector space,  $\{V_i\}_{i=1, \dots, k}$  subspaces of vector space  $V$ .  $V$  a **direct sum of  $\{V_i\}$**  if:

$$V = V_1 \oplus \dots \oplus V_k$$

If  $\forall v \in V$  can be expressed as  $v = v_1 + \dots + v_k$  for unique vectors  $v_i \in V_i$

*Corollary*

$$V = V_1 \oplus \dots \oplus V_k \iff \dim V = \sum_{i=1}^k \dim V_i \text{ and if } B_i \text{ a basis for } V_i, B = \bigcup_i B_i \text{ is a basis for } V$$

#### Definition - Invariant subspaces

$T : V \rightarrow V$  a linear map,  $W$  a subspace of  $V$ .

$$W \text{ is } T\text{-invariant if } T(W) \subseteq W, T(W) = \{T(w) : w \in W\}$$

Write  $T_W : W \rightarrow W$  for the restriction of  $T$  to  $W$

#### Notation - Direct sums of matrices

$$A_1 \oplus \dots \oplus A_k = \begin{pmatrix} A_1 & & \\ & \ddots & \\ & & A_k \end{pmatrix}$$

## 5 Quotient Spaces

**Definition - Cosets**  $V$  a vector space over  $F$ , with  $W \leq V$  a subspace.

$$\text{Cosets } W + v \text{ for } v \in V \quad W + v := \{w + v : w \in W\}$$

### Quotient Space

Define  $V/W$  as a vector space of vectors  $W + v$  over  $F$

- Addition;  $(W + v_1) + (W + v_2) = W + v_1 + v_2$
- Scalar Multiplication;  $\lambda(W + v) = W + \lambda v$

Can verify this using vector space axioms.

*Dimension of  $V/W$*

$$\dim V/W = \dim V - \dim W$$

### Definition - Quotient Map

$T : V \rightarrow V$  a linear map,  $W$  a  $T$ -invariant subspace of  $V$ . Quotient map:  $\bar{T} : V/W \rightarrow V/W$  such that

$$\bar{T}(W + v) = W + T(v), \quad \forall v \in V$$

## 6 Triangularisation

*Lemma - Diagonal Matrices*

$$A = \begin{pmatrix} \lambda_1 & & & \\ 0 & \lambda_2 & & * \\ & & \ddots & \\ 0 & & & \lambda_n \end{pmatrix}, B = \begin{pmatrix} \mu_1 & & & \\ 0 & \mu_2 & & * \\ & & \ddots & \\ 0 & & & \mu_n \end{pmatrix}$$

- Characteristic polynomial of  $A = \prod_{i=1}^n (x - \lambda_i)$ , eigenvalues =  $\{\lambda_i\}$
- $\det A = \prod_{i=1}^n \lambda_i$
- $AB$  also upper triangular, with  $\text{diag}(AB) = \lambda_1 \mu_1, \dots, \lambda_n \mu_n$

### Theorem 6.2 - Triangularisation Theorem

$V$  an  $n$  dimensional vector space over  $F$ ,  $T : V \rightarrow V$  a linear map,

Where  $\chi(T) = \prod_{i=1}^n (x - \lambda_i)$ , where  $\lambda_i \in F \forall i \implies \exists$  basis  $B$  of  $V$  s.t  $[T]_B$  upper triangular

## 7 The Cayley-Hamilton Theorem

### Theorem. 7.1 - (Cayley-Hamilton Theorem)

$V$  a finite dimensional vector space over  $F$ .  $T : V \rightarrow V$  a linear map with char. poly.  $p(x)$

$$p(T) = 0$$

## 8 Polynomials

### Definition - Polynomials over a field

$F$  a field,  $p(x)$  over  $F$ , for  $p(x) = \sum_i a_i x^i$ ,  $F[x] = \{p(x) : a_i \in F\}$

### Degree of polynomial

$\deg(p(x))$  = the highest power of  $x$  in  $p(x)$

### Euclidean Algorithm

$f, g \in F[x]$  with  $\deg(g) \geq 1$ , Then  $\exists q, r \in F[x]$  s.t

$$f = gq + r$$

for either  $r = 0$  or  $\deg(r) < \deg(g)$

**Definition - Greatest Common Divisor (GCD) of polynomials**

$f, g \in F[x] \setminus \{0\}$ , **Say**  $d \in F[x]$  **the gcd of**  $f, g$  **if:**

- (i)  $d|f$  and  $d|g$
- (ii) **if**  $e(x) \in F[x]$  **and**  $e|f$  **and**  $e|g$  **Then**  $e|d$

Say  $f, g$  are co-prime if  $\gcd(f, g) = 1$

**Corollary**

$$d = \gcd(f, g) \implies \exists r, s \in F[x] \text{ s.t. } d = rf + sg$$

**Definition - Irreducible polynomials**

$p(x) \in F[x]$  irreducible over  $F$  if  $\deg(p) \geq 1$  and  $p$  not factorisable over  $F$  as a product of  $\{f_i\} \in F$  s.t.  $\deg(f_i) \leq \deg(p)$

**Corollary**

$p(x) \in F[x]$  irreducible,  $\{g_i\} \in F[x]$ , if  $p|g_1 \dots g_r \implies p|g_i$  for some  $i$

**Theorem 8.7 - (Unique Factorization Theorem)**

$f(x) \in F[x]$  s.t.  $\deg(f) \geq 1$

$$f = p_1 \dots p_r$$

where each  $p_i \in F[x]$  irreducible. **Factorisation of**  $f$  **is unique up to scalar multiplication**

## 9 The minimal polynomial of a linear map

**Definition - Minimal polynomial**

Say  $m(x) \in F[x]$  a minimal polynomial for  $T : V \rightarrow V$  if

- (i)  $m(T) = 0$
- (ii)  $m(x)$  monic
- (iii)  $\deg(m)$  is as small as possible s.t (i) and (ii)

**Properties of the minimal polynomial**

- For  $T$  a linear map, its minimal polynomial  $m_T(x)$  is unique
- $p(x) \in F[x], p(T) = 0 \iff m_T(x)|p(x)$
- $m_T(x)|c_T(x)$  the char. poly. of  $T$
- $\lambda \in F$  a root of  $c_T(x) \implies \lambda$  a root of  $m_T(x)$
- $A, B \in M_n(F)$  s.t.  $A \sim B \implies m_A(x) = m_B(x)$

**Theorem 9.3**

$p(x) \in F[x]$  an irreducible factor of  $c_T(x) \implies p(x)|m_T(x)$

**Corollaries**

- $c_T(x) = c_{T_W}(x)c_{\bar{T}}(x)$
- $m_{T_W}(x)$  and  $m_{\bar{T}}(x)$  both divide  $m_T(x)$

## 10 Primary Decomposition

### Theorem 10.1 - (Primary Decomposition Theorem)

$V$  a finite dimensional vector space over  $F$ ,  $T : V \rightarrow V$  a linear map with  $m_T(x)$   
Let factorisation of  $m_T(x)$  into irreducible polynomials be:

$$m_T(x) = \prod_{i=1}^k f_i(x)^{n_i}$$

Where  $\{f_i(x)\}$  all distinct irreducible polynomials in  $F[x]$

For  $1 \leq i \leq k$ , define:

$$V_i = \ker(f_i(T)^{n_i})$$

Then

1.  $V = V_1 \oplus \cdots \oplus V_k$  (Call this the **primary decomposition** of  $V$  w.r.t  $T$ )
2. each  $V_i$  is  $T$ -invariant
3. each restriction  $T_{V_i}$  has minimal polynomial  $f_i(x)^{n_i}$

In the case where each  $f_i(x) = (x - \lambda_i)$

$$\implies m_T(x) = \prod_{i=1}^k (x - \lambda_i)^{n_i}$$

With  $\lambda_i$  distinct eigenvalues of  $T$  and  $V_i = \ker(T - \lambda_i I)^{n_i}$

We call  $V_i$  the **generalised  $\lambda_i$ -eigenspace of  $T$**

### Corollary

A linear map  $T : V \rightarrow V$  diagonalisable  $\iff m_T(x) = \prod_{i=1}^k (x - \lambda_i)$  a product of distinct linear factors

### Corollary

For  $T : V \rightarrow V$  a linear map, with  $g_1(x), g_2(x) \in F[x]$  coprime polynomials s.t  $g_1(T)g_2(T) = 0$

1. Then  $V = V_1 \oplus V_2$ , where  $V_i = \ker g_i(T), i = 1, 2$  with each  $V_i$  being  $T$ -invariant
2. Suppose  $m_T(x) = g_1(x)g_2(x) \implies m_{T_{V_i}}(x) = g_i(x), i = 1, 2$

## 11 Jordan Canonical Form

### Definition - Jordan Block

$F$  a field and let  $\lambda \in F$ . Define  $n \times n$  matrix:

$$J_n(\lambda) = \begin{pmatrix} \lambda & 1 & 0 & \cdots & 0 & 0 \\ 0 & \lambda & 1 & \cdots & 0 & 0 \\ 0 & 0 & \lambda & \cdots & 0 & 0 \\ & & & \cdots & & \\ 0 & 0 & 0 & \cdots & \lambda & 1 \\ 0 & 0 & 0 & \cdots & 0 & \lambda \end{pmatrix}$$

### Properties of the Jordan Blocks

1. characteristic and minimal polynomials of  $J_n = (x - \lambda)^n$
2.  $\lambda$  the only eigenvalue of  $J_n$ , with  $a(\lambda) = n, g(\lambda) = 1$
3.  $J_n - \lambda I = J_n(0)$ , multiplication by  $J_n - \lambda I$  sends basis vectors as such:

$$e_n \rightarrow e_{n-1} \rightarrow \cdots \rightarrow e_2 \rightarrow e_1 \rightarrow 0$$

4.  $(J_n - \lambda I)^n = 0$ , and for  $i < n$ ,  $\text{rank}((J_n - \lambda I)^i) = n - i$ . And under multiplication:

$$e_n \rightarrow e_{n-i}, e_{n-1} \rightarrow e_{n-i-1} \cdots$$



**Lemma**

Let  $A = A_1 \oplus \cdots \oplus A_k$  for each  $i$  let  $A_i$  have char. poly  $c_i(x)$  and min. poly.  $m_i(x)$ .

- $c_A(x) = \prod_{i=1}^k c_i(x)$
- $m_A(x) = \text{lcm}(m_1(x), \dots, m_k(x))$
- $\forall \lambda$  eigenvalues of  $A$ ,  $\dim E_\lambda(A) = \sum_{i=1}^k \dim E_\lambda(A_i)$
- $\forall q(x) \in F[x]$ ,  $q(A) = q(A_1) \oplus \cdots \oplus q(A_k)$

**Theorem 11.3 - (Jordan Canonical Form)**

$A \in M_n(F)$ , suppose  $c_A(x)$  a product of linear factors over  $F$ .

Then

1.  $A$  similar to matrix of form

$$J = J_{n_1}(\lambda_1) \oplus \cdots \oplus J_{n_k}(\lambda_k)$$

**This is the Jordan Canonical Form (JCF) of  $A$**

2. Matrix  $J$  from above, is uniquely determined by  $A$  up to order of Jordan blocks

**Computing the JCF**

JCF theorem says  $A \sim J$ , a JCF matrix.

$A \sim J \implies$  same characteristic polynomial, eigenvalues, geometric multiplicities, minimal polynomial and  $q(A) \sim q(J)$  for any polynomial  $q$ .

For each eigenvalue  $\lambda$ , collect all Jordan blocks as such;

$$J = \underbrace{(J_{n_1}(\lambda) \oplus \cdots \oplus J_{n_a}(\lambda))}_{\lambda\text{-blocks of } J} \oplus \underbrace{(J_{m_1}(\mu) \oplus \cdots \oplus J_{m_b}(\mu))}_{\mu\text{-blocks of } J} \oplus \cdots$$

**Properties of JCF**

$J$  as above,  $\lambda$  an eigenvalue;

1.  $n_1 + \cdots + n_a = a(\lambda)$
2.  $a = \text{number of } \lambda\text{-blocks} = g(\lambda)$
3.  $\max(n_1, \dots, n_a) = r$ , where  $(x - \lambda)^r$  the highest power of  $(x - \lambda)$  dividing  $m_A(x)$

**Theorem 11.6**

$T : V \rightarrow V$  a linear map s.t  $c_T(x)$  a product of linear factors  $\implies \exists$  basis  $B$  of  $V$  s.t  $[T]_B$  a JCF matrix

**Definition.- Nilpotent Matrix**

$A^k = 0$  for some  $k \in \mathbb{N}$

**Theorem 11.7**

$S : V \rightarrow V$  a nilpotent linear map  $\implies \exists$  basis  $B$  of  $V$  s.t

$$[S]_B = J_{n_1}(0) \oplus \cdots \oplus J_{n_k}(0)$$

### Computing a Jordan Basis

Finding the Jordan Basis  $B$  as above.

We have  $V = V_1 \oplus \cdots \oplus V_k$  by Primary Decomposition Theorem.

Take each restriction  $T_{V_i}$  each with 1 eigenvalue.

Let  $S_i = T_{V_i} - \lambda_i I$  so each  $S_i$  nilpotent.

**Step 1** - Compute subspaces

$$V \supset S(V) \supset S^2(V) \supset \cdots \supset S^r(V) \supset 0$$

$$S^{r+1}(V) = 0$$

**Step 2** - Find basis of  $S^r(V)$ , Using the following rules extend to basis of  $S^{r-1}(V)$ :

Given basis  $u_1, S(u_1), \dots, S^{m_1-1}(u_1), \dots, u_r, S(u_r), \dots, S^{m_r-1}(u_r)$

(1) for each  $i$  add vector  $v_i \in V$  s.t  $u_i = S(v_i)$

(2) note  $\ker(S)$  contains linearly independent vectors

$$S^{m_1-1}(u_1), \dots, S^{m_r-1}(u_r)$$

extend to basis of  $\ker(S)$  by adding vectors  $w_1, \dots, w_s$  with  $\dim \ker(S) = r + s$

Yielding

$$v_1, S(v_1), \dots, S^{m_1}(v_1), \dots, v_r, S(v_r), \dots, S^{m_r}(v_r), w_1, \dots, w_s$$

**Step 3** - Repeat successively finding Jordan bases of  $S^{r-2}, \dots, S(V), V$

## 12 Cyclic Decomposition & Rational Canonical Form

### Definition - Cyclic Subspaces

$V$  a finite dimensional vector space over  $F$ , and  $T : V \rightarrow V$  a linear map.

Let  $0 \neq v \in V$  and define

$$\begin{aligned} Z(v, T) &= \{f(T)(v) : f(x) \in F[x]\} \\ &= \text{Sp}(v, T(v), T^2(v), \dots) \end{aligned}$$

Say  $Z(v, T)$  the  $T$ -cyclic subspace of  $V$  generated by  $v$ .

$Z(v, T)$  is  $T$ -invariant. Write  $T_v$

### Definition - $T$ -annihilator of $v$ and $Z(v, T)$

Considering,  $v, T(v), T^2(v), \dots$  with  $T^k(v)$  first vector in span of previous ones

$$\implies T^k(v) = -a_0 v - a_1 T(v) - \cdots - a_{k-1} T^{k-1}(v)$$

$T$ -annihilator of  $v$  and  $Z(v, T)$  is

$$m_v(x) = x^k + a_{k-1}x^{k-1} + \cdots + a_0 \in F[x]$$

This is monic polynomial of smallest degree s.t  $m_v(T)(v) = 0$  also with  $m_v(T)(w) = 0 \forall w \in Z(v, T)$

### Theorem 12.2 (Cyclic Decomposition Theorem)

$V$  a finite dimensional vector space over  $F$

$T : V \rightarrow V$  a linear map. Suppose  $m_T(x) = f(x)^k$  for irreducible  $f(x) \in F[x]$

$\implies \exists v_1, \dots, v_r \in V$  s.t

$$V = Z(v_1, T) \oplus \cdots \oplus Z(v_r, T)$$

where

(1) each  $Z(v_i, T)$  has  $T$ -annihilator  $f(x)^{k_i}$  for  $1 \leq i \leq r$ ,  $k = k_1 \geq k_2 \geq \cdots \geq k_r$

(2)  $r$  and  $k_1, \dots, k_r$  uniquely determined by  $T$

**Corollary 12.3**

$T$  a finite dimensional vector space over  $F$   
 $\implies \exists$  basis  $B$  of  $V$  s.t

$$[T]_B = C(f(x)^{k_1}) \oplus \cdots \oplus C(f(x)^{k_r})$$

**Corollary 12.3**

$A \in M_n(F)$ , with  $m_A(x) = x^k$

$$\implies A \sim C(x^{k_1} \oplus \cdots \oplus C(x^{k_r}))$$

**Theorem 12.5 (Rational Canonical Form Theorem)**

$V$  be finite dimensional over field  $F$  with  $T : V \rightarrow V$  a linear map with

$$m_T(x) = \prod_{i=1}^t f_i(x)^{k_i}$$

with  $\{f_i(x)\}_{i=1}^t \in F[x]$  set of distinct irreducible polynomials  $\implies \exists$  basis  $B$  of  $V$  s.t

$$[T]_B = C(f_1(x)^{k_{11}}) \oplus \cdots \oplus C(f_1(x)^{k_{1r_1}}) \oplus \cdots \\ \oplus C(f_t(x)^{k_{t1}}) \oplus \cdots \oplus C(f_t(x)^{k_{tr_t}})$$

where for each  $i$

$$k_i = k_{i1} \geq \cdots \geq k_{ir_i}$$

with  $r_i$  and  $k_{i1}, \dots, k_{ir_i}$  uniquely determined by  $T$

**Corollary 12.6**

$A \in M_n(F)$  s.t  $m_A(x) = \prod_{i=1}^t f_i(x)^{k_i}$  distinct irreducible polynomials.

$$\implies A \sim C(f_1(x)^{k_{11}}) \oplus \cdots \oplus C(f_1(x)^{k_{1r_1}}) \oplus \cdots \oplus C(f_t(x)^{k_{t1}}) \oplus \cdots \oplus C(f_t(x)^{k_{tr_t}})$$

**Computing the RCF**

$T : V \rightarrow V$  we have

$$c_T(x) = \prod_{i=1}^t f_i(x)^{n_i}, \quad m_T(x) = \prod_{i=1}^t f_i(x)^{k_i}$$

$\{f_i(x)\}$  all distinct irreducible polynomials in  $F[x]$

enough to find;  $\text{rank}(f_i(T)^r) \forall i \in \{1, \dots, t\}, 1 \leq r \leq k_i$

## 13 The Dual Space

**Definition - Linear functional**

$V$  a vector space over  $F$

A **linear functional** on  $V$  a linear map  $\phi : V \rightarrow F$  s.t

$$\phi(\alpha v_1 + \beta v_2) = \alpha \phi(v_1) + \beta \phi(v_2) \quad \forall v_i \in V, \forall \alpha, \beta \in F$$

*Operations of linear functionals*

$$(i) (\phi_1 + \phi_2)(v) = \phi_1(v) + \phi_2(v), \quad \forall v \in V$$

$$(ii) (\lambda \phi)(v) = \lambda \phi(v), \quad \forall \lambda \in F, \forall v \in V$$

**Definition - The dual space**

$$V^* = \{\phi | \phi : V \text{ to } F \text{ a linear functional} \}$$

$V^*$  a vector space over  $F$  w.r.t above multiplication and addition.

**Dimension**

$\{v_i\}_i$  a basis of  $V$  with eigenvalues  $\{\lambda\}_i$

$\exists! \phi \in V^*$  sending  $v_i \rightarrow \lambda_i$

$$\phi(\sum \alpha_i v_i) = \sum \alpha_i \lambda_i$$

**Proposition 13.1**

Let  $n = \dim V$  with  $\{v_1, \dots, v_n\}$  a basis of  $V$   
 $\forall i$  define  $\phi_i \in V^*$  by

$$\phi_i(v_j) = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

$\implies \phi_i(\sum \alpha_j v_j) = \alpha_i \implies \{\phi_1, \dots, \phi_n\}$  a basis of  $V^*$  the **dual basis** of  $B$   
 $\dim V^* = n = \dim V$

**Definition - Annihilators**

$V$  a finite dimensional vector space over  $F$  and  $V^*$  the dual space.  $X \subset V$ . Say annihilator  $X^0$  of  $X$  :

$$X^0 = \{\phi \in V^* : \phi(x) = 0 \forall x \in X\}$$

$X^0$  a subspace of  $V^*$

**Proposition 13.2.**

$W$  subspace of  $V \implies \dim W^0 = \dim V - \dim W$

## 14 Inner Product Spaces

**Definition - Inner Product**

$F = \mathbb{R}$  or  $\mathbb{C}$ .  $V$  a vector space over  $F$

Inner product on  $V$  a map  $(u, v) : V \times V \rightarrow F$  satisfying

- (i)  $(\lambda_1 v_1 + \lambda_2 v_2, w) = \lambda_1(v_1, w) + \lambda_2(v_2, w)$
- (ii)  $(w, v) = \overline{(v, w)}$
- (iii)  $(v, v) > 0$  if  $v \neq 0$

$\forall v_i, v, w \in V$  and  $\lambda_i \in F$ . Call such a vector space  $V$  with inner product  $(,)$  an **inner product space**.

**Properties of Inner Product Space**

- right-linear for  $F = \mathbb{R}$ ;  $(v, \lambda_1 w_1 + \lambda_2 w_2) = \lambda_1(v, w_1) + \lambda_2(v, w_2)$
- $(v, v) \in \mathbb{R}$
- $(0, v) = 0 \forall v \in V$
- symmetry;  $F = \mathbb{R} \implies (w, v) = (v, w)$
- $(v, w) = (v, x) \forall v \in V \implies w = x$

**Matrix of an inner product**  $V$  a finite dimensional inner product space.  $B = \{v_1, \dots, v_n\}$  a basis.

Defining  $a_{ij} = (v_i, v_j)$ . So we have  $a_{ji} = \overline{a_{ij}}$

$F = \mathbb{R} \implies A$  symmetric

$F = \mathbb{C} \implies A$  hermitian

$$v, w \in V \implies (v, w) = [v]_B^T A [\bar{w}]_B$$

**Definition - Positive definite**

Hermitian matrix  $A$  positive-definite if  $x^T A \bar{x} > 0 \forall$  non-zero  $x \in F^n$

**Proposition 14.1**

For  $u, v, w \in V$  we have

- (i)  $|(u, v)| \leq \|u\| \|v\|$  (*Cauchy-Schwarz Inequality*)
- (ii)  $\|u + v\| \leq \|u\| + \|v\|$
- (iii)  $\|u - v\| \leq \|u - w\| + \|w - v\|$  (*Triangle inequalities*)

## Dual Space

Let  $V$  an inner product space over  $F = \mathbb{R}$  or  $\mathbb{C}$   
 $v \in V$  define

$$f_v : V \rightarrow F$$
$$f_v(w) = (w, v)$$

$\implies f_v$  linear functional  $\in V^*$

## Definition - $\bar{V}$

$\bar{V}$  has same vectors as  $V$

- Addition in  $\bar{V}$  same as  $V$
- Scalar multiplication;  $\lambda * v = \bar{\lambda}v$

## Proposition 14.2.

$V$  finite-dimensional. Define  $\pi : \bar{V} \rightarrow V^*$  as

$$\pi(v) = f_v \quad \forall v \in V$$

$\implies \pi$  a vector space isomorphism

## Definition - Orthogonality

$\{v_1, \dots, v_k\}$  orthogonal if  $(v_i, v_j) = 0 \quad \forall i, j \quad i \neq j$   
Orthonormal if also  $\|v_i\| = 1 \quad \forall i$

## Definition - $W^\perp$

$W \subseteq V$  define

$$W^\perp = \{u \in V : (u, w) = 0 \quad \forall w \in W\}$$

## Proposition

$V$  a finite dimensional inner product space.  $W \leq V$

$$\implies V = W \oplus W^\perp$$

## Theorem 14.5

$V$  a finite dimensional inner product space

- (i)  $V$  has orthonormal basis
- (ii) Any orthonormal set of vectors  $\{w_1, \dots, w_r\}$  can be extended to orthonormal basis of  $V$

## Gram-Schmidt Process

**Step 1** - Start with basis  $\{v_1, \dots, v_n\}$  of  $V$

**Step 2** - let  $u_1 = \frac{v_1}{\|v_1\|}$  define  $w_2 = v_2 - (v_2, u_1)u_1$   
 $\implies (w_2, u_1) = 0, \quad \text{let } u_2 = \frac{w_2}{\|w_2\|}$   
 $\implies \{u_1, u_2\}$  orthonormal

**Step 3** - Let

$$w_3 = v_3 - (v_3, u_1)u_1 - (v_3, u_2)u_2$$

$$\text{With } u_3 = \frac{w_3}{\|w_3\|} \implies \{u_1, u_2, u_3\}$$

**Step 4** - Continue, for  $i^{\text{th}}$  step

$$u_i = \frac{w_i}{\|w_i\|} \quad w_i = v_i - (v_i, u_1)u_1 - \dots - (v_i, u_{i-1})u_{i-1}$$

Yielding after  $n$  steps an orthonormal basis  $\{u_1, \dots, u_n\}$  with

$$\text{Sp}(u_1, \dots, u_i) = \text{Sp}(v_1, \dots, v_i) \quad \forall i \in \{1, \dots, n\}$$

## Projections

$V$  an inner product space.  $v, w \in V \setminus 0$

**Projection of  $v$  along  $w$**  defined to be  $\lambda w$  for  $\lambda \frac{(v, w)}{(w, w)}$ .

For  $W \leq V, v \in V$

define projection of  $V$  along  $W$  as follows:

$$V = W \oplus W^\perp$$

$$v = w + w' \quad \text{for unique } w \in W, w' \in W^\perp$$

Define **orthogonal projection** map along  $W$ .

$$\pi_W : V \rightarrow W$$

$$\pi_W(v) = w$$

### Proposition 14.7.

$V$  an inner product space.  $W \leq V$  with  $\pi_W$  orthogonal projection map along  $W$ .

- (i)  $v \in V \implies \pi_W$  vector in  $W$  closest to  $V$   
i.e for  $w \in W$ ,  $\|w - v\|$  minimum for  $w = \pi_W(v)$
- (ii)  $\text{dist}(v, w)$  denotes shortest distance from  $v$  to any vector in  $W$   
 $\implies \text{dist}(v, w) = \|v - \pi_W(v)\|$
- (iii)  $\{v_1, \dots, v_r\}$  orthonormal basis of  $W$   
 $\implies \pi_W(v) = \sum_{j=1}^r (v, v_j) v_j$

## Change of orthonormal basis

### Proposition 14.8

$V$  an inner product space.  $E = \{e_1, \dots, e_n\}$ ,  $F = \{f_1, \dots, f_n\}$  orthonormal basis of  $V$   
 $P = (p_{ij})$  change of basis matrix.

$$f_i = \sum_{j=1}^n p_{ji} e_j \implies P^T \bar{P} = I$$

### Definition

- $P \in M_n(\mathbb{R}) : P^T P = I \implies$  orthogonal matrix
- $P \in M_n(\mathbb{C}) : P^T \bar{P} = I \implies$  unitary matrix

### Properties of the above matrices

- (i) length-preserving maps of  $\mathbb{R}^n, \mathbb{C}^n$  (isometries)  
i.e  $\|Pv\| = \|v\| \quad \forall v$
- (ii) Set of all isometries form a group - *classical group*  
*orthogonal group*;  $O(n, \mathbb{R}) = \{P \in M_n(\mathbb{R}) : P^T P = I\}$   
*Unitary Group*;  $U(n, \mathbb{C}) = \{P \in M_n(\mathbb{C}) : P^T \bar{P} = I\}$

## 15 Linear maps on inner product spaces

### Proposition 15.1.

$V$  a finite dimensional inner product space.  $T : V \rightarrow V$  a linear map  
 $\implies \exists!$  linear map  $T^* : V \rightarrow V$  s.t  $\forall u, v \in V$

$$(T(u), v) = (u, T^*(v))$$

Say  $T^*$  - **adjoint of  $T$**

$T$  **self-adjoint** if  $T = T^*$

### Proposition 15.2.

$V$  an inner product space with orthonormal basis  $E = \{v_1, \dots, v_n\}$

$T : V \rightarrow V$  a linear map,  $A = [T]_E$

$\implies [T^*]_E = \bar{A}^T$  if field  $\mathbb{R} \implies A$  symmetric, if field  $\mathbb{C} \implies A$  hermitian

**Theorem 15.3. Spectral Theorem**

$V$  an inner product space.  $T : V \rightarrow V$  a self-adjoint linear map  $\implies V$  has orthonormal basis of  $T$ -eigenvectors.

**Corollary 15.4.**

- $A \in M_n(\mathbb{R}) \implies \exists$  orthogonal  $P$  s.t  $P^{-1}AP$  diagonal
- $A \in M_n(\mathbb{C}) \implies \exists$  unitary  $P$  s.t  $P^{-1}AP$  diagonal

**Lemma 15.5.**

$T : V \rightarrow V$  self-adjoint

- (i) eigenvalues of  $T$  real
- (ii) eigenvectors for distinct eigenvalues, orthogonal to each other
- (iii) If  $W \subseteq V$ ,  $T$ -invariant  $\implies W^\perp$  is also  $T$ -invariant

## 16 Bilinear & Quadratic Forms

**Definition. - Bi-linear form**

$V$  a vector space over  $F$

**Bi-linear form** on  $V$  a map;  $(, ) : V \times V \rightarrow F$  which is both right and left-linear.

i.e  $\forall \alpha, \beta \in F$

- $(\alpha v_1 + \beta v_2, w) = \alpha(v_1, w) + \beta(v_2, w)$
- $(v, \alpha w_1 + \beta w_2) = \alpha(v, w_1) + \beta(v, w_2)$

**General example**

$F$  a field,  $V = F^n$  with  $A \in M_n(F)$

$\implies (u, v) = u^T A v \quad \forall u, v \in V$  a bilinear form on  $V$

**Matrices**

$(, )$  a bilinear form on finite dimensional vector space  $V$ . With  $B = \{v_1, \dots, v_n\}$

$A$  matrix of  $(, )$  w.r.t  $B$ , So  $(a_{ij}) = (v_i, v_j) \implies \forall u, v \in V \quad (u, v) = [u]_B^T A [v]_B$

**Definition - Symmetric & Skew-symmetric**

Bilinear form  $(, )$  on  $V$  is

- **Symmetric** if  $(u, v) = (v, u) \quad \forall u, v \in V$
- **Skew symmetric** if  $(v, u) = -(u, v) \quad \forall u, v \in V$

**Definition - Characteristic of Field  $F$** 

$char$  of field  $F$  is the smallest  $n \in \mathbb{N}_+$  s.t  $n \cdot 1 = 0$ . if no such  $n$  exists say  $char(F) = 0$

**Lemma 16.1.**

$V$  a vector space over  $F$  with  $char(F) \neq 2$

$(, )$  skew-symmetric bilinear form on  $V \implies (v, v) = 0 \quad \forall v \in V$

$$(v, v) = -(v, v) \implies 2(v, v) = 0 \iff 2 = 0 \text{ or } (v, v) = 0$$

**Orthogonality****Theorem 16.2**

bilinear form  $(, )$  has property that

$$(v, w) = 0 \iff (w, v) = 0$$

$$\iff$$

$(, )$  skew-symmetric or symmetric

**Definition - Non-degenerate**

$(, )$  on  $V$  **non-degenerate** if  $V^\perp = \{0\}$ . Where  $V^\perp$  defined analogously w.r.t bilinear forms.

$$\forall u \in V, (u, v) = 0 \quad \forall v \in V \implies u = 0$$

$V^\perp = \{0\} \iff$  matrix of  $(,)$  w.r.t a basis is invertible.

### Dual Space

#### Proposition 16.3.

Suppose  $(,)$  non-degenerate bilinear form on a finite dimensional vector space  $V$ .

- (i)  $v \in V$  define  $f_v \in V^*$   
 $f_v(u) = (v, u) \quad \forall u \in V$   
 $\implies \phi : V \rightarrow V^*$  mapping  $v \mapsto f_v$  ( $v \in V$ ) an isomorphism
- (ii)  $\forall W \leq V$  we have  $\dim(W^\perp) = \dim(V) - \dim(W)$

### Bases

#### Definition

$A, B \in M_n(F)$  **congruent** if  $\exists$  invertible  $P \in M_n(F)$  s.t

$$B = P^T A P$$

$A, B$  congruent  $\implies$  bilinear forms  $(u, v)_1 = u^T A v$  and  $(u, v)_2 = u^T B v$  are **equivalent**

### Skew-symmetric bilinear forms

#### Theorem 16.4.

$V$  a finite dimensional vector space over  $F$  where  $\text{char}(F) \neq 2$

$(,)$  non-degenerate skew-symmetric bilinear form on  $V$ . Then

- (i)  $\dim(V)$  even
- (ii)  $\exists$  basis  $B = \{e_1, f_1, \dots, e_m, f_m\}$  of  $V$   
s.t matrix of  $(,)$  w.r.t  $B$  is a block-diagonal matrix

$$J_m = \underbrace{\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \oplus \dots \oplus \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}}_{m \text{ blocks}}$$

$$\begin{aligned} \text{So that } (e_i, f_i) &= -(f_i, e_i) = 1 \\ (e_i, e_j) &= (f_i, f_j) = (e_i, f_j) = (f_j, e_i) = 0 \quad \forall i \neq j \end{aligned}$$

#### Corollary 16.5.

If  $A$  invertible skew-symmetric  $n \times n$  matrix over  $F$  where  $\text{char}(F) \neq 2 \implies n$  even and  $A$  congruent to  $J_m$

### Symmetric bilinear forms

#### Theorem 16.6.

$V$  a finite dimensional vector space over  $F$  where  $\text{char}(F) \neq 2$

$(,)$  a non-degenerate symmetric bilinear form on  $V$

$\implies V$  has orthogonal basis  $B = \{v_1, \dots, v_n\}$

$$\begin{aligned} (v_i, v_j) &= 0 \quad \text{for } i \neq j \\ (v_i, v_i) &= \alpha_i \neq 0 \quad \forall i \end{aligned}$$

Matrix of  $(,)$  w.r.t  $B = \text{diag}(\alpha_1, \dots, \alpha_n)$

#### Corollary 16.7.

$A$  invertible symmetric matrix over  $F, \text{char}(F) \neq 2$

$\implies A$  congruent to diagonal matrix

#### Computing orthogonal basis for 16.6

1. find  $v_1$  s.t  $(v_1, v_1) \neq 0$
2. Compute  $v_1^\perp$  and find  $v_2 \in v_1^\perp$  s.t  $(v_2, v_2) \neq 0$
3. Compute  $Sp(v_1, v_2)^\perp$  and find  $v_3 \in Sp(v_1, v_2)^\perp$  s.t  $(v_3, v_3) \neq 0$
4. Continue until you get orthogonal basis



## Quadratic Form

Assume from now  $F$  s.t  $\text{char}(F) \neq 2$ ,  $V$  a finite dimensional vector space over  $F$

### Definition - Quadratic form

Quadratic form on  $V$  a map  $Q : V \rightarrow F$  of form

$$Q(v) = (v, v) \quad \forall v \in V$$

$(,)$  a symmetric bilinear form on  $V$

$Q$  non-degenerate if  $(,)$  non-degenerate.

### Remarks

(i) given  $Q$  we find  $(u, v) = \frac{1}{2}[Q(u+v) - Q(u) - Q(v)]$

(ii)  $V = F^n$  every symmetric bilinear forms s.t

$$(x, y) = x^T A y \quad \text{for } A = A^T, (x, y \in V)$$

For  $\mathbf{x} = (x_1, \dots, x_n)^T$

$$\begin{aligned} Q(x) &= x^T A x \\ &= \sum_{i,j} a_{ij} x_i x_j \\ &= \sum_{i=1}^n a_{ii} x_i^2 + 2 \sum_{i < j} a_{ij} x_i x_j \end{aligned}$$

A general homogeneous quadratic polynomial in  $x_1, \dots, x_n$  ( all terms of degree 2)

### Change of variables

### Definition - Equivalent Quadratic Forms

$V = F^n$ ,  $Q : V \rightarrow F$

$Q(x) = x^T A x \quad \forall x \in V, A$  symmetric

Take  $y = (y_1, \dots, y_n)^T$  s.t  $x = Py$  for  $P$  invertible

$$\implies Q(x) = y^T P^T A P y = Q'(y)$$

If such a  $P$  exists we say  $Q, Q'$  **equivalent**

*note:*

Congruent matrices  $A, P^T A P$

$$A \sim P^T A P \iff P \text{ orthogonal}$$

### Theorem 16.8.

$V = F^n$ ,  $Q : V \rightarrow F$  non-degenerate quadratic form

(i) if  $F = \mathbb{R} \implies Q$  equivalent to form

$$Q_0(x) = x_1^2 + \dots + x_n^2 \quad (x \in \mathbb{R}^n)$$

Has matrix  $I_n$

(ii) if  $F = \mathbb{Q} \implies Q$  equivalent to unique  $Q_{p,q}; p+q=n$

$$Q_{p,q}(x) = x_1^2 + \dots + x_p^2 - (x_{p+1}^2 + \dots + x_{p+q}^2) \quad (x \in \mathbb{Q}^n)$$

Has matrix  $I_{p,q} = \begin{pmatrix} I_p & 0 \\ 0 & -I_q \end{pmatrix}$

(iii) if  $F = \mathbb{Q} \implies \exists$  infinitely many inequivalent non-degenerate quadratic forms on  $\mathbb{Q}^n$

**Definition - isometry**

$f = (,)$  a non-degenerate symmetric/skew-symmetric bilinear form on finite dimensional vector space  $V$

**Isometry** of  $f$  a linear map  $T : V \rightarrow V$  s.t

$$(T(u), T(v)) = (u, v) \quad \forall u, v \in V$$

$T$  invertible since  $f$  non-degenerate.

**Definition - Isometry Group**

$$I(V, f) = \{T : T \text{ an isometry} \}$$

forms a subgroup of general linear group  $GL(V)$

**Equivalently;**

fix basis  $B$  of  $V$ ,  $A$  matrix of  $f$  w.r.t  $B$  if  $[T]_B = X \implies T \in I(V, f) \iff X^T A X = A$

$$\implies I(V, f) \cong \{X \in GL(n, F) : X^T A X = A\}$$

- $f$  skew-symmetric  $\implies$  there is only one form (up to equivalence) so we get one isometry group; Classical *symplectic group*  $Sp(V, f)$
- $f$  symmetric  $\implies$  there are many forms, forming the isometry groups; the classical *orthogonal groups*  $O(V, f)$

## Part I

# Computing with Numbers

## 1 Numbers

### 1.1 Binary Representation

#### Definition 1.1

$B_0, \dots, B_p \in \{0, 1\}$  denote  $x \in \mathbb{N}_0$  in **binary format**

$$(B_p \dots B_1 B_0)_2 := 2^p B_p + \dots + 2B_1 + B_0$$

For  $b_1, b_2, \dots \in \{0, 1\}$ , Denote  $x \in \mathbb{R}^+$  in binary format by:

$$(B_p \dots B_0 . b_1 b_2 b_3 \dots)_2 = (B_p \dots B_0)_2 + \frac{b_1}{2} + \frac{b_2}{2^2} + \frac{b_3}{2^3} + \dots$$

### 1.2 Integers

#### Definition 1.2 *Ring of integers modulo m*

$$\mathbb{Z}_m := \{0 \pmod{m}, 1 \pmod{m}, \dots, m-1 \pmod{m}\}$$

Integers with  $p$ -bits represent elements in  $\mathbb{Z}_{2^p}$

Integer arithmetic equivalent to arithmetic module  $2^p$

#### 1.2.1 Signed Integer

Use **Two's complement** convention.

$$\text{Integer is } \begin{cases} \text{negative,} & \text{if 1st bit} = 1 \\ \text{positive,} & \text{if 1st bit} = 0 \end{cases}$$

$2^p - y$  interpreted as  $-y$

e.g

$$11001001 = -55 \quad 01001001 = 73$$

#### *Overflow*

Given arithmetic is modulo  $2^p$  we often get overflow errors

```
typemax{Int8} + Int8(1) # returns typemin{Int8}      01111111
127 + 1 = -128                                         00000001+
                                                         -----
                                                         =10000000
```

#### 1.2.2 Variable bit representation

Can represent integers using a variable number of bits, hence avoiding overflow.

In Julia we have **BigInts** created by **big()**

#### 1.2.3 Division

We have 2 types of division

- (i) Integer division ( $\div$ )  
 $5 \div 2$  equivalent to `div(5,2)` rounds down returning 2
- (ii) Standard Division ( $/$ )  
Returns floating-point number  
 $5 / 2$   
Can also create rationals using (`//`)  
 $(1/2) + (3/4)$

Rational arithmetic often leads to overflow so combine it with **big()** often.

### 1.3 Floating Point numbers

Subset of real numbers representable using a fixed number of bits.

#### Definition 1.3 *Floating-point numbers*

Given integers

$\sigma$  - (Exponential shift)

$Q$  - (Number of exponent bits)

$S$  - (The precision)

Define set of floating-point numbers as

$$F_{\sigma,Q,S} := F_{\sigma,Q,S}^{normal} \cup F_{\sigma,Q,S}^{sub-normal} \cup F_{\sigma,Q,S}^{special}$$

With each component as such

$$\begin{aligned} F_{\sigma,Q,S}^{normal} &= \{\pm 2^{q-\sigma} \times (1.b_1b_2 \dots b_S)_2 : 1 \leq q < 2^Q - 1\} \\ F_{\sigma,Q,S}^{sub-normal} &= \{\pm 2^{1-\sigma} \times (0.b_1b_2b_3 \dots b_S)_2\}. \\ F_{\sigma,Q,S}^{special} &= \{-\infty, \infty, \text{NaN}\} \end{aligned}$$

Floating point numbers stored in  $1 + Q + S$  total bits as such

$$\textcolor{blue}{s}q_{Q-1} \dots q_0 \textcolor{violet}{b}_1 \dots \textcolor{violet}{b}_S$$

With first bit the **sign bit**: 0 positive, 1 negative

Bits  $q_{Q-1} \dots q_0$  the **exponent bits** - binary digits of unsigned integer  $q$

Bits  $b_1 \dots b_S$  the **significand bits**.

For  $q = (q_{Q-1} \dots q_0)_2$

(i)  $1 \leq q < 2^Q - 1$  - Bits represent normal number

$$x = \pm 2^{q-\sigma} \times (1.b_1b_2b_3 \dots b_S)_2$$

(ii)  $q = 0$ . (All bits are 0) - Bits represent sub-normal number.

$$x = \pm 2^{1-\sigma} \times (0.b_1b_2b_3 \dots b_S)_2.$$

(iii)  $q = 2^Q - 1$  (All bits are 1) - Bits represent special number.  $\pm\infty$

#### 1.3.1 IEEE Floating-point numbers

##### Definition 1.4 *IEEE Floating-point numbers*

IEEE has 3 standard floating-point formats defined as such with corresponding types in Julia

|                            |                            |
|----------------------------|----------------------------|
| $F_{16} := F_{15,5,10}$    | Float16 – Double-precision |
| $F_{32} := F_{127,8,23}$   | Float32 – Single-precision |
| $F_{64} := F_{1023,11,52}$ | Float64 – Half-precision   |

Float64 - created by using decimals. e.g 1.0

Float32 - created by using f0 e.g 1f0

#### 1.3.2 Special normal numbers

##### Definition 1.5 *Machine epsilon*

Denoted:

$$\begin{aligned} \epsilon_{m,S} &:= 2^{-S} \\ \min |F_{\sigma,Q,S}^{normal}| &= 2^{1-\sigma} \end{aligned}$$

Largest (postive) normal number is

$$\max F_{\sigma,Q,S}^{normal} = 2^{2^Q-2-\sigma} (1.11 \dots 1)_2 = 2^{2^Q-2-\sigma} (2 - \epsilon_m)$$

### 1.3.3 Special Numbers

#### Definition 1.6 *Not a Number*

We have NaN represent "not a number"

### 1.4 Arithmetic

Arithmetic on floating-points exact up to rounding.

#### Definition 1.7 *Rounding*

$$\begin{aligned}\text{fl}_{\sigma_Q, S}^{\text{UP}} : \mathbb{R} &\rightarrow F_{\sigma, Q, S} \text{ rounds up} \\ \text{fl}_{\sigma_Q, S}^{\text{DOWN}} : \mathbb{R} &\rightarrow F_{\sigma, Q, S} \text{ rounds down} \\ \text{fl}_{\sigma_Q, S}^{\text{Nearest}} : \mathbb{R} &\rightarrow F_{\sigma, Q, S} \text{ rounds nearest}\end{aligned}$$

In case of tie, returns floating-point number whose least significand bit is equal to 0  
 $\text{fl}_{\text{nearest}}$  the default rounding mode. Exempt excess notation when implied by context.

Rounding modes in Julia we are going to use: **RoundUp**, **RoundDown**, **RoundNearest**  
Use **setrounding(Float\_, roundingmode)** to change mode in a chunk of code.

$$\begin{aligned}x \oplus y &:= \text{fl}(x + y) \\ x \ominus y &:= \text{fl}(x - y) \\ x \otimes y &:= \text{fl}(x * y) \\ x \oslash y &:= \text{fl}(x / y)\end{aligned}$$

Each of the above defined in IEEE arithmetic.

*Warning* These operations are not **associative**  $(x \oplus y) \oplus z \neq x \oplus (y \oplus z)$

### 1.5 Bounding errors in floating-point arithmetic

#### Definition 1.8 *Absolute/relative error*

if  $\tilde{x} = x + \delta_{rma} = x(1 + \delta_r)$

(i)  $|\delta_a|$  - **absolute error**

(ii)  $\delta_r$  - **relative error**

#### Definition 1.9 *Normalised Range*

Normalised range  $\mathcal{N}_{\sigma, Q, S} \subset \mathbb{R}$  - subset of reals, that lies between smallest and largest normal floating-point number:

$$\mathcal{N}_{\sigma, Q, S} := \{x : \min |F_{\sigma, Q, S}| \leq |x| \leq \max F_{\sigma, Q, S}\}$$

**Proposition.** - *Rounding arithmetic*

if  $x \in \mathcal{N} \implies$

$$\text{fl}^{\text{mode}}(x) = x(1 + \delta_x^{\text{mode}})$$

With relative error:

$$\begin{aligned}|\delta_x^{\text{nearest}}| &\leq \frac{\epsilon_m}{2} \\ |\delta_x^{\text{up/down}}| &< \epsilon_m.\end{aligned}$$

### 1.5.1 Arithmetic and Special numbers

We have the following identities

|          |        |             |         |            |         |
|----------|--------|-------------|---------|------------|---------|
| 1/0.0    | # Inf  | Inf*0       | # NaN   | NaN*0      | # NaN   |
| 1/(-0.0) | # -Inf | Inf+5       | # Inf   | NaN+5      | # NaN   |
| 0.0/0.0  | # NaN  | (-1)*Inf    | # -Inf  | 1/NaN      | # NaN   |
|          |        | 1/Inf       | # 0.0   | NaN == NaN | # false |
|          |        | 1/(-Inf)    | # -0.0  | NaN != NaN | # true  |
|          |        | Inf - Inf   | # NaN   |            |         |
|          |        | Inf == Inf  | # true  |            |         |
|          |        | Inf == -Inf | # false |            |         |

### 1.5.2 Special functions

Functions such as `cos`, `sin`, `exp` designed to have *relative accuracy*

e.g for  $s = \sin(x)$  we satisfy

$$s = \sin(x)(1 + \delta) \quad |\delta| < c\epsilon_m$$

for reasonable small  $c > 0$  given  $x \in F^{\text{normal}}$

## 1.6 High-precision floating-point numbers

Possible to set precision using `BigFloat` type created using `big()`

Use to find rigorous bound on a number.

e.g.

```
setprecision(4_000) # 4000 bit precision
setrounding(BigFloat, RoundDown) do
  big(1)/3
end, setrounding(BigFloat, RoundUp) do
  big(1)/3
end
```

[illegible]

## 2 Differentiation

Considering functions:

- (i) *Black-box function*  $f^{\text{FP}} : D \rightarrow F$ ,  $D \subset F \equiv F_{\sigma, Q, S}$   
 Only know function pointwise,  $F$  discrete  $\implies f^{\text{FP}}$  not differentiable rigorously.  
 Assume  $f^{\text{RP}}$  approximates a differentiable function  $f$  with controlled error.
- (ii) *Generic function*  
 A formula that can be evaluated on arbitrary types. e.g polynomial  $p(x) = p_0 + p_1x + \dots + p_nx^n$   
 Consider both differentiable  $f : D \rightarrow \mathbb{R}$ ,  $D \subset \mathbb{R}$  and floating point evaluated  $f^{\text{FP}} : D \cap F \rightarrow F$ , which is actually computed.
- (iii) *Graph Function*  
 Function built by composition of basic "kernels" with known differentiability properties.

## 2.1 Finite-differences

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} \implies f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

for sufficiently small  $h$

Approximation uses only *black-box* notion of function.

**Proposition.** - Bounding the derivative

$$\left| f'(x) - \frac{f(x+h) - f(x)}{h} \right| \leq \frac{M}{2}h$$

where  $M = \sup_{x \leq t \leq x+h} |f''(t)|$ . Given by Taylor's theorem.

Can also use left-side and central differences to compute derivatives.

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

### 2.1.1 Bounding the error

**Theorem 2.1** (*Finite differences error bound*)

$f$  twice-differentiable in neighbourhood of  $x$

Assume  $f^{\text{FP}} = f(x) + \delta_x^f$  has uniform absolute accuracy in that neighbourhood i.e  $|\delta_x^f| \leq c\epsilon_m$  for fixed constant  $c$ .

Take  $h = 2^{-n}$  for  $n \leq S$  (no. of Significant bits) and  $|x| < 1$

Finite difference approximation then satisfies

$$(f^{\text{FP}}(x+h) \ominus f^{\text{FP}}(x)) \oslash h = f'(x) + \delta_{x,h}^{\text{FD}}$$

Where

$$|\delta_{x,h}^{\text{FD}}| \leq \frac{|f'(x)|}{2}\epsilon_m + Mh + \frac{4c\epsilon_m}{h}$$

for  $M = \sup_{x \leq t \leq x+h} |f''(t)|$ .

3 terms in bound tell us behaviour.

**Heuristic** - (finite differences with floating point step.)

Choose  $h$  proportional to  $\sqrt{\epsilon_m}$

## 2.2 Dual numbers

**Definition 2.1** *Dual numbers*

Dual numbers,  $\mathbb{D}$  Commutative ring over reals generated by 1 and  $\epsilon$  with  $\epsilon^2 = 0$ , written  $a + b\epsilon$

### 2.2.1 Connection with differentiation

Dual numbers not prone to growth due to round-off errors.

**Theorem 2.2** (*Polynomials on dual numbers*)

$p$  a polynomial.

$$p(a + b\epsilon) = p(a) + b'p(a)\epsilon$$

**Definition 2.2** *Dual extension*

$f$  real-valued function differentiable at  $a$ , a dual extension at  $a$  if

$$f(a + b\epsilon) = f(a) + bf'(a)\epsilon$$

**Lemma** - (Product and Chain rule)

$f$  a dual extension at  $g(a)$ ,  $g$  a dual extension at  $a$

$$\implies q(x) := f(g(x)) \text{ a dual extension at } a$$

$f, g$  dual extensions at  $a$

$$\implies r(x) := f(x)g(x) \text{ a dual extension at } a$$

## Part II

# Computing with Matrices

## 3 Structured Matrices

Consider the following structures

- (i) *Dense*  
Considered unstructured, need to store all entries in vector or Matrix.  
Reduces directly to standard algebraic operations
- (ii) *Triangular*  
A matrix upper or lower triangular, can invert immediately with back-substitution  
Store as dense and ignore upper/lower entries in practice.
- (iii) *Banded*  
A matrix zero, apart from entries a fixed distance from diagonal.  
Have diagonal, bidiagonal and tridiagonal matrices.
- (iv) *Permutation*  
Permutation matrix permutes rows of a vector
- (v) *Orthogonal*  
 $Q$  orthogonal satisfies  $Q^T Q = I$ , hence easily inverted

### 3.1 Dense vectors and matrices

*Storage in memory*

- **Vector** of primitive type  
stored consecutively in memory.
- **Matrix** stored consecutively in memory  
going down column-by column. (column-major format)

|     |       |          |   |
|-----|-------|----------|---|
| A = | [1 2; | vec(A) = | 1 |
|     | 3 4;  |          | 3 |
|     | 5 6]  |          | 5 |
|     |       |          | 2 |
|     |       |          | 4 |
|     |       |          | 6 |

Transposing A done lazily, A' stores entries by row

**Matrix multiplication** done as expected **A\*x**

*Implemented 2 ways*

Using Traditional definition

Or going column-by-column

$$\begin{bmatrix} \sum_{j=1}^n a_{1,j} x_j \\ \vdots \\ \sum_{j=1}^n a_{m,j} x_j \end{bmatrix} \quad x_1 \mathbf{a}_1 + \cdots + x_n \mathbf{a}_n$$

Both are  $O(mn)$  operations, but column-by-column faster due to more efficient memory accessing.

**Solving a linear system** done by \

```
A = [1 2 3;      returns # 41.000000000000036
      1 2 4;      -17.000000000000014
      3 7 8]      1.0
b = [10; 11; 12]
A \ b
```



## 3.2 Triangular Matrices

Represented as dense square matrices, where we ignore entries above/below diagonal.

```
A = [1 2 3;
      4 5 6;
      7 8 9]
U = UpperTriangular(A)
# 1 2 3
   5 6
   9
L = LowerTriangular(A)
# 1
   4 5
   7 8 9
```

We have U,L both storing all the data of A

### Solving upper-triangular system

$$\begin{bmatrix} u_{11} & \cdots & u_{1n} \\ & \ddots & \vdots \\ & & u_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$$

by computing  $x_n, x_{n-1}, \dots, x_1$  by the back-substitution formula:

$$x_k = \frac{b_k - \sum_{j=k+1}^n u_{kj}x_j}{u_{kk}}$$

Multiplication and solving linear system  $O(n^2)$  for a triangular matrix.

## 3.3 Banded Matrices

### Definition 3.1 *Bandwidths*

Matrix  $A$  has

- **lower-bandwidth**,  $l$  if  $A[k, j] = 0 \forall k - j > l$
- **upper-bandwidth**,  $u$  if  $A[k, j] = 0 \forall j - k > u$
- **strictly lower-bandwidth** if it has lower-bandwidth  $l$  and  $\exists j$  such that  $A[j + l, j] \neq 0$
- **strictly upper-bandwidth** if it has upper-bandwidth  $u$  and  $\exists k$  such that  $A[k, k + u] \neq 0$

### Definition 3.2 *Diagonal*

Matrix diagonal if square and  $l = u = 0$  the bandwidths.

Stored as **Vectors** in Julia.

Perform multiplication and solving linear systems in  $O(n)$  operations.

### Definition 3.3 *Bidiagonal*

Matrix bidiagonal if square and has bandwidths

- $(l, u) = (1, 0) \implies$  lower-bidiagonal
- $(l, u) = (0, 1) \implies$  upper-bidiagonal

```
Bidiagonal([1,2,3], [4,5], :L)
```

```
# 1
  4 2
  5 3
```

```
Bidiagonal([1,2,3], [4,5], :U)
```

```
# 1 4
    2 5
    3
```

Multiplication and solving linear systems still  $O(n)$  operations.

### Definition 3.4 *Tridiagonal*

```
Tridiagonal([1,2], [3,4,5], [6,7])
```

```
# 3 6
  1 4 7
    2 5
```

Matrix tridiagonal if square and has bandwidths  $l = u = 1$

### 3.4 Permutation Matrices

Matrix representation of the symmetric group  $S_n$  acting on  $\mathbb{R}^n$   
 $\forall \sigma \in S_n$  a bijection between  $\{1, 2, \dots, n\}$  and itself.

*Cauchy Notation*

$$\begin{pmatrix} 1 & 2 & 3 & \cdots & n \\ \sigma_1 & \sigma_2 & \sigma_3 & \cdots & \sigma_n \end{pmatrix}$$

Where  $\{\sigma_1, \dots, \sigma_n\} = \{1, 2, \dots, n\}$

Inverse permutation given by  $\sigma^{-1}$ , found by swapping rows of cauchy notation and reordering.

*Permuting a vector*

$$\sigma = [\sigma_1, \dots, \sigma_n]^T$$

$$\mathbf{v}[\sigma] = \begin{bmatrix} v_{\sigma_1} \\ \vdots \\ v_{\sigma_n} \end{bmatrix}$$

Obviously  $\mathbf{v}[\sigma][\sigma^{-1}] = \mathbf{v}$

#### Definition 3.5 *Permutation Matrix*

Entries of  $P_\sigma$  given by

$$P_\sigma[k, j] = e_k^T P_\sigma e_j = e_k^T e_{\sigma_j^{-1}} = \delta_{k, \sigma_j^{-1}} = \delta_{\sigma_k, j}$$

where  $\delta_{k, j}$  is the Kronecker delta

Permutation matrix equal to identity matrix with rows permuted.

**Proposition** - Inverse of Permutation Matrix

$$P_\sigma^T = P_{\sigma^{-1}} = P_\sigma^{-1} \implies P_\sigma \text{ orthogonal}$$

### 3.5 Orthogonal Matrices

#### Definition 3.6 *Orthogonal Matrix*

Square matrix orthogonal if  $Q^T Q = Q Q^T = I$

Special cases

#### 3.5.1 Simple Rotations

#### Definition 3.7 *Simple Rotation*

$2 \times 2$  rotation matrix through angle  $\theta$

$$Q_\theta := \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$

#### Definition 3.8 *two-arg arctan*

two-argument arctan function gives angle  $\theta$  through point  $[a, b]^T$

$$\text{atan}(b, a) := \begin{cases} \text{atan} \frac{b}{a} & a > 0 \\ \text{atan} \frac{b}{a} + \pi & a < 0 \text{ and } b > 0 \\ \text{atan} \frac{b}{a} + \pi & a < 0 \text{ and } b < 0 \\ \pi/2 & a = 0 \text{ and } b > 0 \\ -\pi/2 & a = 0 \text{ and } b < 0 \end{cases}$$

$\text{atan}(-1, -2)$  # angle through  $[-2, -1]$

**Proposition** - Rotating vector to unit axis

$$Q = \frac{1}{\sqrt{a^2 + b^2}} \begin{bmatrix} a & b \\ -b & a \end{bmatrix}$$

Satisfies  $Q \begin{bmatrix} a \\ b \end{bmatrix} = \sqrt{a^2 + b^2} \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

### 3.5.2 Reflections

#### Definition 3.9 *Reflection Matrix*

Given vector  $\mathbf{v}$  satisfying  $\|\mathbf{v}\| = 1$ , reflection matrix is orthogonal matrix.

$$Q_{\mathbf{v}} := I - 2\mathbf{v}\mathbf{v}^T$$

Reflections in direction of  $\mathbf{v}$

**Proposition** - *Properties of reflection matrix*

- (i) 1. Symmetry:  $Q_v = Q_v^T$
- (ii) 2. Orthogonality:  $Q_v Q_v = I$
- (iii) 2.  $v$  is an eigenvector of  $Q_v$  with eigenvalue  $-1$
- (iv) 4.  $Q_v$  is a rank -1 perturbation of  $I$
- (v) 3.  $\det Q_v = -1$

#### Definition 3.10 *Householder reflection*

Given vector  $\mathbf{x}$  define Householder reflection.

$$Q_{\mathbf{x}}^{\pm, H} := Q_{\mathbf{w}}$$

For  $\mathbf{y} = \mp \|\mathbf{x}\| e_1 + x$ ,  $\mathbf{w} = \frac{\mathbf{y}}{\|\mathbf{y}\|}$

Default choice in sign is

$$Q_x^H := Q_x^{-\text{sign}(x_1), H}$$

**Lemma**

$$Q_x^{\pm, H} \mathbf{x} = \pm \|\mathbf{x}\| e_1$$

## 4 Decompositions and Least Squares

Consider decompositions of matrix into products of structured matrices.

1. *QR Decomposition* (For square or rectangular matrix  $A \in \mathbb{R}^{m \times n}$ ,  $m \geq n$ )

$$A = QR = \underbrace{[\mathbf{q}_1 | \cdots | \mathbf{q}_m]}_{m \times m} \underbrace{\begin{bmatrix} \times & \cdots & \times \\ & \ddots & \vdots \\ & & \times \\ & & 0 \\ & & \vdots \\ & & 0 \end{bmatrix}}_{m \times n}$$

$Q$  orthogonal and  $R$  right/upper-triangular

2. *Reduced QR Decomposition*

$$A = \hat{Q} \hat{R} = \underbrace{[\mathbf{q}_1 | \cdots | \mathbf{q}_m]}_{m \times m} \begin{bmatrix} \times & \cdots & \times \\ & \ddots & \vdots \\ & & \times \end{bmatrix}$$

$\hat{Q}$  has orthogonal columns, and  $\hat{R}$  upper-triangular.

3. *PLU Decomposition* (For square Matrix)

$$A = P^T L U$$

$P$  a permutation matrix,  $L$  lower triangular and  $U$  upper triangular

4. *Cholesky Decomposition* (For square, symmetric positive definite matrix ( $x^T A x > 0 \forall x \in \mathbb{R}^n, x \neq 0$ ))

$$A = LL^T$$

Useful as component pieces easily inverted on a computer.

$$\begin{aligned} A = P^T L U &\implies A^{-1} \mathbf{b} = U^{-1} L^{-1} P \mathbf{b} \\ A = Q R &\implies A^{-1} \mathbf{b} = R^{-1} Q^T \mathbf{b} \\ A = L L^T &\implies A^{-1} \mathbf{b} = L^{-1} L^{-1} \mathbf{b} \end{aligned}$$

## 4.1 QR and least squares

Consider matrices with more rows than columns.

QR decomposition contains reduced QR decomposition within it

$$A = QR = [\hat{Q} | \mathbf{q}_{n+1} | \dots | \mathbf{q}_m] \begin{bmatrix} \hat{R} \\ \mathbf{0}_{m-n \times n} \end{bmatrix} = \hat{Q} \hat{R}.$$

*Least squares problem*

Find  $\vec{x} \in \mathbb{R}^n$  s.t  $\|A\vec{x} - \vec{b}\|$  is minimised

For  $m = n$  and  $A$  invertible we simply have  $\vec{x} = A^{-1} \vec{b}$ .

$$\|A\mathbf{x} - \mathbf{b}\| = \|QR\mathbf{x} - \mathbf{b}\| = \|R\mathbf{x} - Q^T \mathbf{b}\| = \left\| \begin{bmatrix} \hat{R} \\ \mathbf{0}_{m-n \times n} \end{bmatrix} \mathbf{x} - \begin{bmatrix} \hat{Q}^T \\ \mathbf{q}_{n+1}^T \\ \vdots \\ \mathbf{q}_m^T \end{bmatrix} \mathbf{b} \right\|$$

To minimise this norm, suffices to minimise

$$\|\hat{R}\mathbf{x} - \hat{Q}^T \mathbf{b}\| \implies \mathbf{x} = \hat{R}^{-1} \hat{Q}^T \mathbf{b}$$

Provided column rank of  $A$  is full, we have  $\hat{R}$  invertible

## 4.2 Reduced QR and Gram-Schmidt

### 4.2.1 Computing QR decomposition

- (i) Write  $A = [\mathbf{a}_1 | \dots | \mathbf{a}_n]$ ,  $a_k \in \mathbb{R}^m$   
Assume  $A$  has full column rank,  $a_k$  all linearly independent.

Column span of first  $j$  columns in  $A$  same as first  $j$  columns in  $\hat{Q}$

$$\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_n) = \text{span}(\mathbf{q}_1, \dots, \mathbf{q}_n)$$

- (ii) if  $\mathbf{v} \in \text{span}(\mathbf{a}_1, \dots, \mathbf{a}_n) \implies \forall \mathbf{c} \in \mathbb{R}^j$

$$\begin{aligned} \mathbf{v} &= [\mathbf{a}_1 | \dots | \mathbf{a}_j] \mathbf{c} \\ &= [\mathbf{q}_1 | \dots | \mathbf{q}_j] \hat{R}[1:j, 1:j] \mathbf{c} \\ &\in \text{span}(\mathbf{q}_1, \dots, \mathbf{q}_n) \end{aligned}$$

- (iii) if  $\mathbf{w} \in \text{span}(\mathbf{q}_1, \dots, \mathbf{q}_n)$ , we have for  $\mathbf{d} \in \mathbb{R}^j$

$$\begin{aligned} \mathbf{w} &= [\mathbf{q}_1 | \dots | \mathbf{q}_j] \mathbf{d} \\ &= [\mathbf{a}_1 | \dots | \mathbf{a}_j] \hat{R}[1:j, 1:j]^{-1} \mathbf{d} \\ &\in \text{span}(\mathbf{a}_1, \dots, \mathbf{a}_j) \end{aligned}$$

We can find an orthogonal basis using Gram-Schmidt.

1. By assumption of full rank of  $A$

$$\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_n) = \text{span}(\mathbf{q}_1, \dots, \mathbf{q}_n)$$

2.  $\mathbf{q}_1, \dots, \mathbf{q}_n$  orthogonal

$$\mathbf{q}_k^T \mathbf{q}_l = \delta_{kl}$$

3. For  $k, l < j$ . Define

$$\mathbf{v}_j := \mathbf{a}_j - \sum_{k=1}^{j-1} \underbrace{\mathbf{q}_k^T \mathbf{a}_j}_{\mathbf{r}_{kj}} \mathbf{q}_k$$

4. For  $k < j$

$$\mathbf{q}_k^T \mathbf{v}_j = \mathbf{q}_k^T \mathbf{a}_j - \sum_{k=1}^{j-1} \underbrace{\mathbf{q}_k^T \mathbf{a}_j}_{\mathbf{r}_{kj}} \mathbf{q}_k^T \mathbf{q}_k = 0.$$

5. Define further

$$\mathbf{q}_j = \frac{\mathbf{v}_j}{\|\mathbf{v}_j\|}$$

Define  $\mathbf{r}_{jj} := \|\mathbf{v}_j\|$ , rearrange definition to have

$$\mathbf{a}_j = [\mathbf{q}_1 | \dots | \mathbf{q}_j] \begin{bmatrix} r_{1j} \\ \vdots \\ r_{jj} \end{bmatrix}$$

$$[\mathbf{a}_1 | \dots | \mathbf{a}_j] \begin{bmatrix} r_{11} & \dots & r_{1j} \\ & \ddots & \vdots \\ & & r_{jj} \end{bmatrix}$$

Compute reduced QR decomposition column-by-column  $\implies$  apply for  $j = n$  to complete decomposition.

### Complexity and Stability

We have a total complexity of  $O(mn^2)$  operations,

Gram-Schmidt algorithm is unstable, rounding errors in floating point accumulate,  $\implies$  lose orthogonality.

### 4.3 Householder reflections and QR

Consider multiplication by Householder reflection corresponding to first column.

$$Q_1 := Q_{a_1}^H$$

$$Q_1 A = \begin{bmatrix} \times & \times & \dots & \times \\ & \times & \dots & \times \\ & \vdots & \ddots & \vdots \\ & \times & \dots & \times \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ & \mathbf{a}_2^1 & \dots & \mathbf{a}_n^1 \end{bmatrix} \quad r_{1j} := (Q_1 \mathbf{a}_j)[1] \quad \mathbf{a}_1^j := (Q_1 \mathbf{a}_j)[2:m]$$

Note that  $r_{11} = -(a_1 1) \|a_1\|$  with all entries of  $\mathbf{a}_1^1$  zero.

Now consider,

$$Q_2 := \begin{bmatrix} 1 & \\ & Q_{\mathbf{a}_2^1}^H \end{bmatrix} = Q_{\begin{bmatrix} 0 \\ \mathbf{a}_2^1 \end{bmatrix}}^H$$

to achieve the following

$$Q_2 Q_1 A = \begin{bmatrix} \times & \times & \times & \dots & \times \\ & \times & \times & \dots & \times \\ & & \vdots & \ddots & \vdots \\ & & \times & \dots & \times \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & \dots & r_{1n} \\ & r_{22} & r_{23} & \dots & r_{2n} \\ & & \mathbf{a}_3^2 & \dots & \mathbf{a}_n^2 \end{bmatrix} \quad r_{2j} := (Q_2 \mathbf{a}_j^1)[1] \quad \mathbf{a}_j^2 := (Q_2 \mathbf{a}_j^1)[2:m-1]$$

**Inductively**, we get

Defining  $\mathbf{a}_j^0 := \mathbf{a}_j$  we have

$$Q_j := \begin{bmatrix} I_{j-1j-1} & \\ & Q_{\mathbf{a}_j^{j-1}}^H \end{bmatrix}$$

$$\mathbf{a}_j^k := (Q_k \mathbf{a}_j^{k-1})[2:m-k+1]$$

$$r_{kj} := (Q_k \mathbf{a}_j^{k-1})[1]$$

Then

$$Q_n \cdots Q_1 A = \underbrace{\begin{bmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \vdots \\ & & r_{nn} \\ & & 0 \\ & & \vdots \\ & & 0 \end{bmatrix}}_R$$

$$\implies A = \underbrace{Q_1 \cdots Q_n}_Q R.$$

## 4.4 PLU Decomposition

### 4.4.1 Special "one-column" Lower triangular matrices

Consider the following set of lower triangular matrices

$$\mathcal{L}_j := \left\{ I + \begin{bmatrix} \mathbf{0}_j \\ \mathbf{1}_j \end{bmatrix} \mathbf{r}_j^{1^{n-j}} \right\}$$

$$L_j = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & \ell_{j+1,j} & 1 & \\ & & \vdots & & \ddots \\ & & \ell_{n,j} & & & 1 \end{bmatrix}$$

With the following properties:

$$\bullet L_j^{-1} = I - \begin{bmatrix} \mathbf{0}_j \\ \mathbf{1}_j \end{bmatrix} \mathbf{e}_j^T = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -\ell_{j+1,j} & 1 & \\ & & \vdots & & \ddots \\ & & -\ell_{n,j} & & & 1 \end{bmatrix} \in \mathcal{L}_j$$

$$\bullet L_j L_k = I + \begin{bmatrix} \mathbf{0}_j \\ \mathbf{1}_j \end{bmatrix} \mathbf{e}_j^T + \begin{bmatrix} \mathbf{0}_k \\ \mathbf{1}_k \end{bmatrix} \mathbf{e}_k^T$$

- $\sigma$  a permutation leaving first  $j$  rows fixed ( $\sigma_\ell = \ell \forall \ell \leq j$ ) and  $L_j \in \mathcal{L}_\ell$

$$P_\sigma L_j = \tilde{L}_j P_\sigma \quad \tilde{L}_j \in \mathcal{L}_\ell$$

### 4.4.2 LU Decomposition

Similarly to QR decomposition we perform a triangularisation using  $L_j \in \mathcal{L}_\ell$ .

Taking the following definitions

$$L_j := I - \begin{bmatrix} \mathbf{0}_j \\ \frac{\mathbf{a}_{j+1}^j[2:n-j]}{\mathbf{a}_{j+1}^j[1]} \end{bmatrix} \mathbf{e}_j^T \quad \mathbf{a}_j^k := (L_k \mathbf{a}_j^{k-1})[2:n-k+1] \quad u_{kj} := (L_k \mathbf{a}_j^{k-1})[1]$$

$$\implies L_{n-1} \cdots L_1 A = \underbrace{\begin{bmatrix} u_{11} & \cdots & u_{1n} \\ & \ddots & \vdots \\ & & u_{nn} \end{bmatrix}}_U$$

$$A = \underbrace{L_1^{-1} \cdots L_{n-1}^{-1}}_L U \quad L_j = I + \begin{bmatrix} \mathbf{0}_j \\ \ell_{j+1,j} \\ \vdots \\ \ell_{n,j} \end{bmatrix} \mathbf{e}_j^T \implies L = \begin{bmatrix} 1 & & & & \\ -\ell_{21} & 1 & & & \\ -\ell_{31} & -\ell_{32} & 1 & & \\ \vdots & \vdots & \ddots & \ddots & \\ -\ell_{n1} & -\ell_{n2} & \cdots & -\ell_{n,n-1} & 1 \end{bmatrix}$$

#### 4.4.3 PLU Decomposition

Achieved by always pivoting when performing Gaussian elimination, swap largest in magnitude entry on the diagonal. This gives us

$$L_{n-1}P_{n-1} \dots P_2L_1P_1A = U$$

for  $P_j$  the permutation that leaves rows  $1 \rightarrow j-1$  fixed, swapping row  $j$  with row  $k \geq j$  whose entry is maximal in magnitude.

$$L_{n-1}P_{n-1} \dots P_2L_1P_1 = \underbrace{L_{n-1}\tilde{L}_{n-2} \dots \tilde{L}_1}_{L^{-1}} \underbrace{P_{n-1} \dots P_2P_1}_P$$

Tilde denotes combined actions of swapping permutations and lower-triangular matrices.

$$P_{n-1} \dots P_{j+1}L_j = \tilde{L}_jP_{n-1} \dots P_{j+1} \implies \tilde{L}_j = I + \begin{bmatrix} \mathbf{0}_j \\ \tilde{\ell}_{j+1,j} \\ \vdots \\ \tilde{\ell}_{n,j} \end{bmatrix} \mathbf{e}_j^\top \implies L = \begin{bmatrix} 1 & & & & & \\ -\tilde{\ell}_{21} & 1 & & & & \\ -\tilde{\ell}_{31} & -\tilde{\ell}_{32} & 1 & & & \\ \vdots & \vdots & \ddots & \ddots & & \\ -\tilde{\ell}_{n-1,1} & -\tilde{\ell}_{n-1,2} & \cdots & -\tilde{\ell}_{n-1,n-2} & 1 & \\ -\tilde{\ell}_{n1} & -\tilde{\ell}_{n2} & \cdots & -\tilde{\ell}_{n,n-2} & -\tilde{\ell}_{n,n-1} & 1 \end{bmatrix}$$

#### 4.5 Cholesky Decomposition

Form of Gaussian elimination (without pivoting) for **symmetric positive definite matrices**  
Substantially faster.

**Definition 4.1** (*Positive definite*)

A square matrix  $A \in \mathbb{R}^{n \times n}$  **positived definite** if  $\forall x \in \mathbb{R}^n, x \neq 0$  we have

$$x^T Ax > 0$$

**Proposition**

$A \in \mathbb{R}^{n \times n}$  positive definite and  $V \in \mathbb{R}^{n \times n}$  non-singular

$$\implies V^T AV \text{ pos. definite}$$

**Proposition**

$A \in \mathbb{R}^{n \times n}$  positive definite  $\implies$  diagonal entries  $a_{ii} > 0$

**Theorem 4.1** (*Subslice positive definite*)

$A \in \mathbb{R}^{n \times n}$  positive definite and  $k \in 1, \dots, n^m$  a vector of  $m$  integers, each integer appearing only once

$$\implies A[k, k] \in \mathbb{R}^{m \times m} \text{ pos. definite}$$

**Theorem 4.2** (*Cholesky and symmetric positive definite*)

Matrix  $A$  symmetric positive definite  $\iff$  has Cholesky Decomposition

$$A = LL^T$$

Where diagonals of  $L$  positive.

**Computing the Cholesky Decomposition**

Using the following definitions:

$$\begin{aligned} A_1 &:= A & \alpha_k &:= A_k[1, 1] \\ \mathbf{v}_k &:= A_k[2 : n - k + 1, 1] & A_{k+1} &:= A_k[2 : n - k + 1, 2 : n - k + 1] - \frac{\mathbf{v}_k \mathbf{v}_k^\top}{\alpha_k} \end{aligned}$$

$$\implies L = \begin{bmatrix} \sqrt{\alpha_1} & & & & \\ \frac{\mathbf{v}_1[1]}{\sqrt{\alpha_1}} & \sqrt{\alpha_2} & & & \\ \frac{\mathbf{v}_1[2]}{\sqrt{\alpha_1}} & \frac{\mathbf{v}_2[1]}{\sqrt{\alpha_2}} & \sqrt{\alpha_3} & & \\ \vdots & \vdots & \ddots & \ddots & \\ \frac{\mathbf{v}_1[n-1]}{\sqrt{\alpha_1}} & \frac{\mathbf{v}_2[n-2]}{\sqrt{\alpha_2}} & \dots & \frac{\mathbf{v}_{n-1}[1]}{\sqrt{\alpha_{n-1}}} & \sqrt{\alpha_n} \end{bmatrix}$$

## 4.6 Timings

Different decompositions have trade-offs between stability and speed.

```
n = 100                                # returns
A = Symmetric(rand(n,n)) + 100I
@btime cholesky(A);                     82.313 s
@btime lu(A);                           127.977 s
@btime qr(A);                           255.111 s
```

*Stability*

| Stable                          | Unstable                                      |
|---------------------------------|---|
| QR with Householder reflections | LU usually, unless diagonally dominant matrix |
| Cholesky for pos. def.          | PLU rarely unstable.                          |

Set of Matrices for which PLU unstable extremely small, often one doesn't run into them.

## 5 Singular Values and Conditioning

### 5.1 Vector Norms

**Definition 5.1** (*Vector-norm*)

Norm on  $\|\cdot\|$  on  $\mathbb{R}^n$  a function satisfying the following,  $\forall x, y \in \mathbb{R}^n, c \in \mathbb{R}$ :

- (i) Triangle inequality:  $\|x + y\| \leq \|x\| + \|y\|$
- (ii) Homogeneity:  $\|cx\| = |c|\|x\|$
- (iii) Positive-definiteness:  $\|x\| = 0 \iff x = 0$

**Definition 5.2** (*p-norm*)

For  $1 \leq p < \infty, x \in \mathbb{R}^n$

$$\|x\|_p := \left( \sum_{k=1}^n |x_k|^p \right)^{1/p}$$

$x_k$   $k$ -th entry of  $x$ .

$p = \infty$  we define

$$\|x\|_\infty := \max_k |x_k|$$

### 5.2 Matrix Norms

**Definition 5.3** (*Fröbenius norm*)

$A$  a  $m \times n$  matrix

$$\|A\|_F := \sqrt{\sum_{k=1}^m \sum_{j=1}^n A_{kj}^2}$$

Given by `norm(A)` in Julia.

`norm(A) == norm(vec(A))`

**Definition 5.4** (*Matrix-norm*)

$A \in \mathbb{R}^{n \times m}$  for 2 norms  $\|\cdot\|_X$  on  $\mathbb{R}^n$  and  $\|\cdot\|_Y$  on  $\mathbb{R}^m$

We have the **induced matrix norm**

$$\|A\|_{X \rightarrow Y} := \sup_{\mathbf{v}: \|\mathbf{v}\|_X = 1} \|A\mathbf{v}\|_Y = \sup_{\mathbf{x} \in \mathbb{R}^n, \mathbf{x} \neq 0} \frac{\|A\mathbf{x}\|_Y}{\|\mathbf{x}\|_X}$$

$$\|A\|_X := \|A\|_{X \rightarrow X}$$

$$\|A\|_1 = \max_j \|\mathbf{a}_j\|_1 \quad \|A\|_\infty = \max_k \|A[k, :]\|_1$$

Given by `opnorm(A, 1)`, `opnorm(A, Inf)` in Julia



### 5.3 Singular Value Decomposition

**Definition 5.5** (*Singular Value Decomposition*)

For  $A \in \mathbb{R}^{n \times n}$  with  $\text{rank}, r > 0$

Reduced singular value decomposition (SVD) is

$$A = U \Sigma V^T$$

$U \in \mathbb{R}^{m \times r}, V \in \mathbb{R}^{r \times n}$  that have orthonormal columns

$\Sigma \in \mathbb{R}^{r \times r}$  diagonal of singular values, all positive and decreasing  $\sigma_1 \leq \dots \leq \sigma_r > 0$

Full singular value decomposition (SVD) is

$$A = \tilde{U} \tilde{\Sigma} \tilde{V}^T$$

$\tilde{U} \in \mathbb{R}^{m \times m}, V \in \mathbb{R}^{n \times n}$  orthogonal matrices,

$\tilde{\Sigma} \in \mathbb{R}^{m \times n}$  has only diagonal entries.

For  $\sigma_k = 0$  if  $k > r$

$$\tilde{\Sigma} = \begin{array}{c|c} \text{if } m > n & \text{if } m < n \\ \hline \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_n & & \\ & & 0 & & \\ & & \vdots & & \\ & & 0 & & \end{bmatrix} & \begin{bmatrix} \sigma_1 & & & & & \\ & \ddots & & & & \\ & & \sigma_m & 0 & \dots & 0 \end{bmatrix} \end{array}$$

**Proposition** - Gram matrix kernel

**Gram-matrix:**  $A^T A$  Kernel of  $A$  also kernel of  $A^A$

**Proposition** - Gram matrix diagonalisation

Gram-matrix satisfies

$$A^T A = Q \Lambda Q^T$$

$Q$  orthogonal and eigenvalues  $\lambda_k$  non-negative

**Theorem 5.1** (*SVD existence*)

$\forall A \in \mathbb{R}^{m \times n}$  has a SVD.

**Corollary**

$A \in \mathbb{R}^{n \times n}$  invertible

$$\implies \|A\|_2 = \sigma_1, \quad \|A^{-1}\|_2 = \sigma_n^{-1}$$

**Theorem 5.2** (*Best low rank approximation*)

$$A_k := [\mathbf{u}_1 | \dots | \mathbf{u}_k] \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix} [\mathbf{v}_1 | \dots | \mathbf{v}_k]^T$$

The best 2-norm approximation of  $A$  by a rank  $k$  matrix.

We have  $\forall$  matrices  $B$  of rank  $k$ ,  $\|A - A_k\|_2 \leq \|A - B\|_2$

### 5.4 Condition numbers

**Proposition**

$|\epsilon_i| \leq \epsilon$  and  $n\epsilon < 1$ , then

$$\prod_{k=1}^n (1 + \epsilon_i) = 1 + \theta_n$$

for constant  $\theta_n$  s.t  $|\theta_n| \leq \frac{n\epsilon}{1-n\epsilon}$

**Lemma.** - Dot product backward error

$\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$

$$\text{dot}(\mathbf{x}, \mathbf{y}) = (\mathbf{x} + \delta \mathbf{x})^T \mathbf{y}$$

Where we have  $|\delta \mathbf{x}| \leq \frac{n\epsilon_m}{2-n\epsilon_m} |\mathbf{x}|$ ,  $|\mathbf{x}|$  absolute value of each entry.

**Theorem 5.3** (Matrix-vector backward error)

$A \in \mathbb{R}^{m \times n}, \mathbf{x} \in \mathbb{R}^n$

$$\text{mul}(A, \mathbf{x}) = (A + \delta A) \mathbf{x}$$

Where  $|\delta A| \leq \frac{n\epsilon_m}{2-n\epsilon_m} \|A\| \implies$

$$\|\delta A\|_1 \leq \frac{n\epsilon_m}{2-n\epsilon_m} \|A\|_1$$

$$\|\delta A\|_2 \leq \frac{\sqrt{\min(m, n)n\epsilon_m}}{2-n\epsilon_m} \|A\|_2$$

$$\|\delta A\|_\infty \leq \frac{n\epsilon_m}{2-n\epsilon_m} \|A\|_\infty$$

**Definition 5.6** (Condition number)

$A$  a square matrix.

**Condition number** (in  $p$ -norm)

$$\kappa_p(A) := \|A\|_p \|A^{-1}\|_p$$

Under the 2-norm:

$$\kappa_2(A) = \frac{\sigma_1}{\sigma_n}$$

**Theorem 5.4** (relative-error for matrix-vector)

Worst-case relative error in  $A\mathbf{x} \approx (A + \delta A)\mathbf{x}$

$$\frac{\|\delta A\mathbf{x}\|}{\|A\mathbf{x}\|} \leq \kappa(A)\epsilon$$

if we have relative perturbation error  $\|\delta A\| = \|A\|\epsilon$

We know for floating point arithmetic the error is bounded by

$$\kappa(A) \frac{n\epsilon_m}{2-n\epsilon_m}$$

## 6 Differential equations via Finite differences

### 6.1 Indefinite integration

For simple differential equation on interval  $[a, b]$

$$u(a) = c$$

$$u'(x) = f(x)$$

We have, for  $u_k \approx u(x_k), k=1, \dots, n-1$

$$f(x_k) = u'(x_k) \approx \frac{u_{k+1} - u_k}{h} = f(x_k)$$

As a linear system

$$\underbrace{\frac{1}{h} \begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}}_{D_h \in \mathbb{R}^{n-1 \times n}} \mathbf{u}^f = \underbrace{\begin{bmatrix} f(x_1) \\ \vdots \\ f(x_{n-1}) \end{bmatrix}}_{\mathbf{f}^f}$$

Super-script  $^f$  denotes forward differences.

$D_h$  not square  $\implies$  need to add extra row from the initial condition  $\mathbf{e}^T \mathbf{u}^f = c$

$$\begin{bmatrix} \mathbf{e}_1^T \\ D_h \end{bmatrix} \mathbf{u}^f = \underbrace{\begin{bmatrix} 1 & & & \\ -1/h & 1/h & & \\ & & \ddots & \ddots \\ & & & -1/h & 1/h \end{bmatrix}}_{L_h} \mathbf{u}^f = \begin{bmatrix} c \\ \mathbf{f}^f \end{bmatrix}$$

Lower-triangular bidiagonal system  $\implies$  solved using forward substitution in  $O(n)$

Can choose either central or backwards-difference formulae too.

*Central differences*

Take  $m_k = \frac{x_{k+1} - x_k}{2} \implies u'(m_k) \approx \frac{u_{k+1} - u_k}{h} = f(m_k)$

$$\frac{1}{h} \underbrace{\begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}}_{D_h} \mathbf{u}^m = \underbrace{\begin{bmatrix} f(m_1) \\ \vdots \\ f(m_{n-1}) \end{bmatrix}}_{\mathbf{f}^m}$$

*Convergence*

We see experimentally that the error for solutions from forward differences is  $O(n^{-1})$  while for central differences it is a faster  $O(n^{-2})$  convergence.

Both appearing to be stable.

## 6.2 Forward Euler

Consider scalar linear time-evolution for  $0 \leq t \leq T$

$$\begin{aligned} u(0) &= c \\ u'(t) - a(t)u(t) &= f(t) \end{aligned}$$

Label  $n$ -point grid as  $t_k = (k-1)h$ ,  $h = \frac{T}{n-1}$

**Definition 6.1** (*Restriction Matrices*)

Define  $n-1 \times n$  **restriction matrices** as

$$I_n^f := \begin{bmatrix} 1 & & & \\ & \ddots & & \\ & & 1 & 0 \\ 0 & 1 & & \end{bmatrix}$$

$$I_n^b := \begin{bmatrix} & & & \\ & \ddots & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}$$

Can replace discretisation using finite differences.  $\frac{u_{k+1} - u_k}{h} - a(t_k)u_k = f(u_k)$

Giving us the linear system

$$\begin{bmatrix} \mathbf{e}_1^T \\ D_h - I_n^f A_n \end{bmatrix} \mathbf{u}^f = \underbrace{\begin{bmatrix} 1 & & & \\ -a(t_1) - 1/h & 1/h & & \\ & \ddots & \ddots & \\ & & -a(t_{n-1}) - 1/h & 1/h \end{bmatrix}}_L \mathbf{u}^f = \begin{bmatrix} c \\ I_n^f \mathbf{f} \end{bmatrix}$$

Where we have

$$A_n = \begin{bmatrix} a(t_1) & & \\ & \ddots & \\ & & a(t_n) \end{bmatrix} \quad \mathbf{f} = \begin{bmatrix} f(t_1) \\ \vdots \\ f(t_n) \end{bmatrix}$$

### 6.3 Backward Euler

Simply replace forward-difference with backward-difference  $\frac{u_k - u_{k-1}}{h} - a(t_k)u_k = f(u_k)$   
 Giving us our system:

$$\begin{bmatrix} \mathbf{e}_1^T \\ D_h - I_n^b A_n \end{bmatrix} \mathbf{u}^f = \underbrace{\begin{bmatrix} 1 & & & \\ -1/h & 1/h - a(t_2) & & \\ & \ddots & \ddots & \\ & & -1/h & 1/h - a(t_n) \end{bmatrix}}_L \mathbf{u}^b = \begin{bmatrix} c \\ I_n^b \mathbf{f} \end{bmatrix}$$

Still bidiagonal forward-substitution

$$\begin{aligned} u_1 &= c \\ (1 - ha(t_{k+1}))u_{k+1} &= u_k + hf(t_{k+1}) \\ u_{k+1} &= (1 - ha(t_{k+1}))^{-1}(u_k + hf(t_{k+1})) \end{aligned}$$

### 6.4 Systems of equations

Solving systems of the form

$$\begin{aligned} \mathbf{u}(0) &= c \\ \mathbf{u}'(t) - A(t)\mathbf{u}(t) &= f(t) \end{aligned}$$

For  $\mathbf{u}, \mathbf{f} : [0, T] \rightarrow \mathbb{R}^d$  and  $A : [0, T] \rightarrow \mathbb{R}^{d \times d}$

Once again discretise at the grid  $t_k$  approximating  $\mathbf{u}(t_k) \approx \mathbf{u}_k \in \mathbb{R}^d$

*Forward-Euler*

$$\begin{aligned} \mathbf{u}_1 &= c \\ \mathbf{u}_{k+1} &= (I - hA(t_{k+1}))^{-1}(\mathbf{u}_k + h\mathbf{f}(t_{k+1})) \end{aligned}$$

### 6.5 Nonlinear problems

Forward-euler extends naturally to nonlinear equations.

$$\mathbf{u}' = f(t, \mathbf{u}(t))$$

Becomes:

$$\mathbf{u}_{k+1} = \mathbf{u}_k + hf(t_k, \mathbf{u}_k)$$

### 6.6 Two-point boundary value problem

Consider one discretisation, since symmetric

$$u''(x) \approx \frac{u_{k-1} - 2u_k + u_{k+1}}{h^2}$$

So we use the  $n - 1 \times n + 1$  matrix

$$D^2h := \frac{1}{h^2} \begin{bmatrix} 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \end{bmatrix}$$

### 6.7 Convergence

**Definition 6.2** (*Toeplitz*)

**Toeplitz matrix** has constant diagonals

$$T[k, j] = a_{k-j}$$

**Proposition.** - (*Bidiagonal Toeplitz inverse*)

Inverse of  $n \times n$  bidiagonal Toeplitz matrix is

$$\begin{bmatrix} 1 & & & & \\ -\ell & 1 & & & \\ & -\ell & 1 & & \\ & & \ddots & \ddots & \\ & & & -\ell & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & & & & \\ \ell & 1 & & & \\ \ell^2 & \ell & 1 & & \\ \vdots & \ddots & \ddots & \ddots & \\ \ell^{n-1} & \dots & 2 & \ell & 1 \end{bmatrix}$$

**Theorem 6.1** (*Forward/Backward Euler Convergence*)

Consider equation

$$u(0) = c, \quad u'(t) + au(t) = f(t)$$

Denote

$$\mathbf{u} := \begin{bmatrix} u(t_1) \\ \vdots \\ u(t_n) \end{bmatrix}$$

Assume  $u$  twice differentiable with uniformly bounded 2nd derivative.

$\Rightarrow$  error for forwardEuler is

$$\|\mathbf{u}^f - \mathbf{u}\|_\infty, \|\mathbf{u}^b - \mathbf{u}\|_\infty = O(n^{-1})$$

### 6.7.1 Poisson

For 2D problems consider Poisson. First stage is to row-reduce to get a summetric tridiagonal pos. def. matrix

$$\begin{bmatrix} 1 & & & & \\ -1/h^2 & 1 & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 & -1/h^2 \\ & & & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & & \\ 1/h^2 & -2/h^2 & 1/h^2 & & \\ & \ddots & \ddots & \ddots & \\ & & 1/h^2 & -2/h^2 & 1/h^2 \\ & & & & 1 \end{bmatrix} = \begin{bmatrix} 1 & & & & \\ 0 & -2/h^2 & 1/h^2 & & \\ & \ddots & \ddots & \ddots & \\ & & 1/h^2 & -2/h^2 & 0 \\ & & & & 1 \end{bmatrix}$$

Consider right-hand side, aside from first and last row, we have

$$\underbrace{\frac{1}{h^2} \begin{bmatrix} -2 & 1 & & \\ 1 & -2 & \ddots & \\ & \ddots & \ddots & 1 \\ & & 1 & -2 \end{bmatrix}}_{\Delta} \begin{bmatrix} u_2 \\ \vdots \\ u_{n-1} \end{bmatrix} = \underbrace{\begin{bmatrix} f(x_2) - c_0/h^2 \\ f(x_3) \\ \vdots \\ f(x_{n-2}) \\ f(x_{n-1}) - c_1/h^2 \end{bmatrix}}_{\mathbf{f}^p}$$

**Theorem 6.2** (*Poisson Convergence*)

Suppose  $u$  four-times differentiable with uniformly bounded fourth-derivative

$\Rightarrow$  finite difference approximation to Poisson convergence like  $O(n^2)$