21. Linear algebra III: Orthogonality and QR factorization

Last time

- Solving linear equations
- Elimination / row reduction
- LU factorization

Goals for today

- Geometry of Euclidean space
- Orthogonality
- Inner products
- Gram–Schmidt orthogonalization
- QR factorization

Systems of linear equations

- Recall: Want to solve linear systems $Ax = b$
- Usually requires significant computational effort

Systems of linear equations

- Recall: Want to solve linear systems $Ax = b$
- Usually requires significant computational effort

- Are there situations where it might be easy?

Systems of linear equations

- Recall: Want to solve linear systems $Ax = b$
- Usually requires significant computational effort

- Are there situations where it might be easy?
- When A is **diagonal**

Systems of linear equations

- Recall: Want to solve linear systems $Ax = b$
- Usually requires significant computational effort

- Are there situations where it might be easy?
- When A is **diagonal**
- Ease comes from strong kind of “independence” of columns:
  - **orthogonal** or **perpendicular**
- One column “does not affect” other columns

Length

- New concept: **length** in Euclidean space

Length

- New concept: **length** in Euclidean space
- Define **norm** (length) of vector $\mathbf{v} \in \mathbb{R}^n$ as

$$\|\mathbf{v}\|_2 := \sqrt{\sum_{i=1}^{n} v_i^2}$$

Length

- New concept: **length** in Euclidean space
- Define **norm** (length) of vector $\mathbf{v} \in \mathbb{R}^n$ as

$$\|\mathbf{v}\|_2 := \sqrt{\sum_{i=1}^{n} v_i^2}$$

- Can write as $\|\mathbf{v}\|^2 = \mathbf{v}^T \mathbf{v}$
- Think of $\mathbf{v}$ as "column vector", $n \times 1$ matrix
- Think of $\mathbf{v}^T$ as "row vector", $1 \times n$ matrix

Length

- New concept: **length** in Euclidean space
- Define **norm** (length) of vector $\mathbf{v} \in \mathbb{R}^n$ as

$$\|\mathbf{v}\|_2 := \sqrt{\sum_{i=1}^{n} v_i^2}$$

- Can write as $\|\mathbf{v}\|^2 = \mathbf{v}^T \mathbf{v}$
- Think of $\mathbf{v}$ as "column vector", $n \times 1$ matrix
- Think of $\mathbf{v}^T$ as "row vector", $1 \times n$ matrix
- **"Householder notation"**

Unit vectors

- Can now decompose vector v as

$$\mathbf{v} = \|\mathbf{v}\|\hat{\mathbf{v}}$$

- $\hat{\mathbf{v}}$ is **unit vector – direction** of $\mathbf{v}$

Orthogonality

- Want to characterise key concept: **orthogonality**
- Several ways to approach this
- Let’s think about **projections**

Projections

- Let u and v be vectors in $\mathbb{R}^2$
- Call $P_u(v)$ the (orthogonal) **projection** of v onto u
- Gives vector in direction of u, with length $\alpha_u(v)$

Projections

- Let u and v be vectors in $\mathbb{R}^2$
- Call $P_u(v)$ the (orthogonal) **projection** of v onto u
- Gives vector in direction of u, with length $\alpha_u(v)$
- (Orthogonal) "shadow" of v on the direction of u
- For orthogonal unit vectors i and j impose $P_i(i) = 1$ and $P_j(i) = 0$

Projections II

- For vector $u = u_1 i + u_2 j$ have $P_i(u) = u_1 i$
- Impose linearity:

$$P_u(v_1 i + v_2 j) = v_1 P_u(i) + v_2 P_u(j) = \|u\| \left[v_1 P_{\hat{u}}(i) + v_2 P_{\hat{u}}(j) \right]$$

- For two *unit* vectors, by symmetry $\alpha_i(\hat{u}) = \alpha_{\hat{u}}(i)$
- So $P_u(v) = (u_1 v_1 + u_2 v_2)\hat{u}$

Dot product and orthogonality

- Define **dot product** $u \cdot v := \alpha_u(v) = u_1 v_1 + u_2 v_2$
- In general $u \cdot v := \sum_{i=1^n} u_i v_i = u^T v$
- From above derivation $\hat{u} \cdot \hat{v} = \cos(\theta)$
- Where θ is **angle** between u and v

Dot product and orthogonality

- Define **dot product** $u \cdot v := \alpha_u(v) = u_1 v_1 + u_2 v_2$
- In general $u \cdot v := \sum_{i=1^n} u_i v_i = u^T v$
- From above derivation $\hat{u} \cdot \hat{v} = \cos(\theta)$
- Where θ is **angle** between u and v
- Say u and v are **orthogonal** if $u \cdot v = 0$

Orthogonal linear combinations

- Suppose $v_1, \ldots, v_n$ are all mutually orthogonal
- Suppose want to solve $x_1 v_1 + \cdots + x_n v_n = b$

Orthogonal linear combinations

- Suppose $v_1, \ldots, v_n$ are all mutually orthogonal
- Suppose want to solve $x_1 v_1 + \cdots + x_n v_n = b$
- It's now easy!:

$$x_i = b \cdot v_i$$

- Component in direction of v_i

Orthogonal linear combinations

- Suppose $v_1, \ldots, v_n$ are all mutually orthogonal
- Suppose want to solve $x_1 v_1 + \cdots + x_n v_n = b$
- It's now easy!:

$$x_i = b \cdot v_i$$

- Component in direction of v_i
- Orthogonal decomposition:
  $b = (b \cdot v_1) v_1 + \cdots + (b \cdot v_n) v_n$

Preserving length

- Alternative point of view: Preserve length
- Look for linear transformation Q preserving length
- So $\|Q\mathbf{u}\| = \|\mathbf{u}\|$ for all u

Preserving length

- Alternative point of view: Preserve length
- Look for linear transformation Q preserving length
- So $\|Q\mathbf{u}\| = \|\mathbf{u}\|$ for all u
- So $(Q\mathbf{u})^T(Q\mathbf{u}) = \mathbf{u}^T\mathbf{u}$

Preserving length

- Alternative point of view: Preserve length
- Look for linear transformation Q preserving length
- So $\|Q\mathbf{u}\| = \|\mathbf{u}\|$ for all u
- So $(Q\mathbf{u})^T(Q\mathbf{u}) = \mathbf{u}^T\mathbf{u}$
- So $\mathbf{u}^T Q^T Q\mathbf{u} = \mathbf{u}^T\mathbf{u}$

Preserving length

- Alternative point of view: Preserve length
- Look for linear transformation Q preserving length
- So $\|Q\mathbf{u}\| = \|\mathbf{u}\|$ for all u
- So $(Q\mathbf{u})^T(Q\mathbf{u}) = \mathbf{u}^T\mathbf{u}$
- So $\mathbf{u}^T Q^T Q\mathbf{u} = \mathbf{u}^T\mathbf{u}$
- Hence $Q^T Q = I$, identity matrix
- So columns $\mathbf{q}_i$ of Q satisfy $\mathbf{q}_i \cdot \mathbf{q}_j = \delta_{ij}$
- i.e. columns are **orthonormal**

Orthogonalization

- Suppose have set of linearly independent vectors
- Can we create an *orthogonal* set of vectors from them?

Orthogonalization

- Suppose have set of linearly independent vectors
- Can we create an *orthogonal* set of vectors from them?
- For two vectors u and v, take u as one of the vectors

Orthogonalization

- Suppose have set of linearly independent vectors
- Can we create an *orthogonal* set of vectors from them?
- For two vectors u and v, take u as one of the vectors
- Make an orthogonal decomposition of v as

  v = (part of v in direction of u) + (the rest)

Orthogonalization II

- Define $q_1 := (v \cdot u)\hat{u}$ = part of v in direction of u
- Define $q_2 := v - a$
- Then
  $q_1 \cdot q_2 = q_1 \cdot v - q_1 \cdot q_1 = (v \cdot u)(u \cdot v) - (v \cdot u)^2 = 0$

Orthogonalization II

- Define $q_1 := (v \cdot u)\hat{u}$ = part of v in direction of u
- Define $q_2 := v - a$
- Then $q_1 \cdot q_2 = q_1 \cdot v - q_1 \cdot q_1 = (v \cdot u)(u \cdot v) - (v \cdot u)^2 = 0$
- i.e. q_1 and q_2 are indeed orthogonal

Orthogonalization III

- Generalize to n vectors:
- Define $u_1 = v_1; \quad q_1 := \frac{u_1}{\|u_1\|}$
- Define $u_2 = v_2 - (v_2 \cdot q_1)q_1; \quad q_2 := \frac{u_2}{\|u_2\|}$
- Define $u_3 = v_3 - (v_3 \cdot q_2)q_2 - (v_3 \cdot q_1)q_1; \quad q_3 := \frac{u_3}{\|u_3\|}$

QR factorization

- Above algorithm is **Gram–Schmidt** algorithm

QR factorization

- Above algorithm is **Gram–Schmidt** algorithm
- Produces a set of n orthogonal vectors q_i
- Now express columns of A in terms of q_i:

$$\begin{aligned} v_1 &= (v_1 \cdot q_1) q_1 \\ v_2 &= (v_2 \cdot q_1) q_1 + (v_2 \cdot q_2) q_2 \\ v_3 &= (v_3 \cdot q_1) q_1 + (v_3 \cdot q_2) q_2 + (v_3 \cdot q_3) q_3 \end{aligned}$$

- Gives $A = QR$
- With orthogonal Q and upper-triangular R

QR factorization

- Above algorithm is **Gram–Schmidt** algorithm
- Produces a set of n orthogonal vectors q_i
- Now express columns of A in terms of q_i:

$$
\begin{aligned}
v_1 &= (v_1 \cdot q_1) q_1 \\
v_2 &= (v_2 \cdot q_1) q_1 + (v_2 \cdot q_2) q_2 \\
v_3 &= (v_3 \cdot q_1) q_1 + (v_3 \cdot q_2) q_2 + (v_3 \cdot q_3) q_3
\end{aligned}
$$

- Gives $A = QR$
- With orthogonal Q and upper-triangular R
- However, **not numerically stable**

Solving linear equations

- Gives another method to solve $Ax = b$:

Solving linear equations

- Gives another method to solve $Ax = b$:
- Factorize $A = QR$
- Solve $Qy = b$ as $y_i = (b \cdot q_i)q_i$
- Then solve $Rx = y$ by substitution

Summary

- Key concepts: **length**, **angle**, **orthogonality**
- Described by inner product $a \cdot b$
- Can orthogonalise set of vectors using Gram–Schmidt algorithm
- Gives QR factorization of a matrix