5. Linear independence

Outline

Linear independence

Basis

Orthonormal vectors

Gram–Schmidt algorithm

Linear dependence

- set of n-vectors $\{a_1, \ldots, a_k\}$ (with $k \geq 1$) is *linearly dependent* if

$$\beta_1 a_1 + \cdots + \beta_k a_k = 0$$

  holds for some $\beta_1, \ldots, \beta_k$, that are not all zero

- equivalent to: at least one a_i is a linear combination of the others
- we say '$a_1, \ldots, a_k$ are linearly dependent'

- $\{a_1\}$ is linearly dependent only if $a_1 = 0$
- $\{a_1, a_2\}$ is linearly dependent only if one a_i is a multiple of the other
- for more than two vectors, there is no simple to state condition

Example

- the vectors

$$a_1 = \begin{bmatrix} 0.2 \\ -7 \\ 8.6 \end{bmatrix}, \qquad a_2 = \begin{bmatrix} -0.1 \\ 2 \\ -1 \end{bmatrix}, \qquad a_3 = \begin{bmatrix} 0 \\ -1 \\ 2.2 \end{bmatrix}$$

  are linearly dependent, since $a_1 + 2a_2 - 3a_3 = 0$

- can express any of them as linear combination of the other two, *e.g.*,

$$a_2 = (-1/2)a_1 + (3/2)a_3$$

Linear independence

- set of n-vectors $\{a_1, \ldots, a_k\}$ (with $k \geq 1$) is *linearly independent* if it is not linearly dependent, *i.e.*,

$$\beta_1 a_1 + \cdots + \beta_k a_k = 0$$

holds only when $\beta_1 = \cdots = \beta_k = 0$

- we say '$a_1, \ldots, a_k$ are linearly independent'
- equivalent to: no a_i is a linear combination of the others
- example: the unit n-vectors $e_1, \ldots, e_n$ are linearly independent

Linear combinations of linearly independent vectors

- suppose x is linear combination of linearly independent vectors $a_1, \ldots, a_k$:

$$x = \beta_1 a_1 + \cdots + \beta_k a_k$$

- the coefficients $\beta_1, \ldots, \beta_k$ are *unique*, *i.e.*, if

$$x = \gamma_1 a_1 + \cdots + \gamma_k a_k$$

  then $\beta_i = \gamma_i$ for $i = 1, \ldots, k$

- this means that (in principle) we can deduce the coefficients from x
- to see why, note that

$$(\beta_1 - \gamma_1) a_1 + \cdots + (\beta_k - \gamma_k) a_k = 0$$

  and so (by linear independence) $\beta_1 - \gamma_1 = \cdots = \beta_k - \gamma_k = 0$

Outline

Linear independence

Basis

Orthonormal vectors

Gram–Schmidt algorithm

Independence-dimension inequality

- *a linearly independent set of n-vectors can have at most n elements*
- put another way: *any set of $n + 1$ or more n-vectors is linearly dependent*

Basis

- a set of n linearly independent n-vectors $a_1, \ldots, a_n$ is called a *basis*
- any n-vector b can be expressed as a linear combination of them:

  $$b = \beta_1 a_1 + \cdots + \beta_n a_n$$

  for some $\beta_1, \ldots, \beta_n$
- and these coefficients are unique
- formula above is called *expansion of b in the $a_1, \ldots, a_n$ basis*
- example: $e_1, \ldots, e_n$ is a basis, expansion of b is

  $$b = b_1 e_1 + \cdots + b_n e_n$$

Outline

Linear independence

Basis

Orthonormal vectors

Gram–Schmidt algorithm

Orthonormal vectors

- set of n-vectors $a_1, \ldots, a_k$ are *(mutually) orthogonal* if $a_i \perp a_j$ for $i \neq j$
- they are *normalized* if $\|a_i\| = 1$ for $i = 1, \ldots, k$
- they are *orthonormal* if both hold
- can be expressed using inner products as

$$a_i^T a_j = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

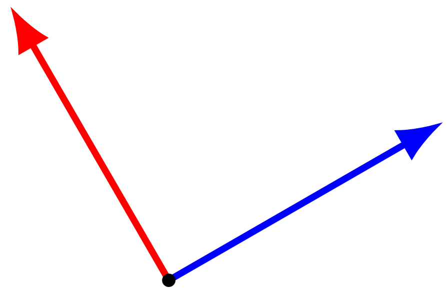
- orthonormal sets of vectors are linearly independent
- by independence-dimension inequality, must have $k \leq n$
- when $k = n$, $a_1, \ldots, a_n$ are an *orthonormal basis*

Examples of orthonormal bases

- standard unit n-vectors $e_1, \ldots, e_n$
- the 3-vectors

$$\begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}, \qquad \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \qquad \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}$$

- the 2-vectors shown below



Orthonormal expansion

- if $a_1, \ldots, a_n$ is an orthonormal basis, we have for any n-vector x

$$x = (a_1^T x)a_1 + \cdots + (a_n^T x)a_n$$

- called *orthonormal expansion of* x (in the orthonormal basis)
- to verify formula, take inner product of both sides with a_i

Outline

Linear independence

Basis

Orthonormal vectors

Gram–Schmidt algorithm

Gram–Schmidt (orthogonalization) algorithm

- an algorithm to check if $a_1, \ldots, a_k$ are linearly independent
- we'll see later it has many other uses

Gram–Schmidt algorithm

---

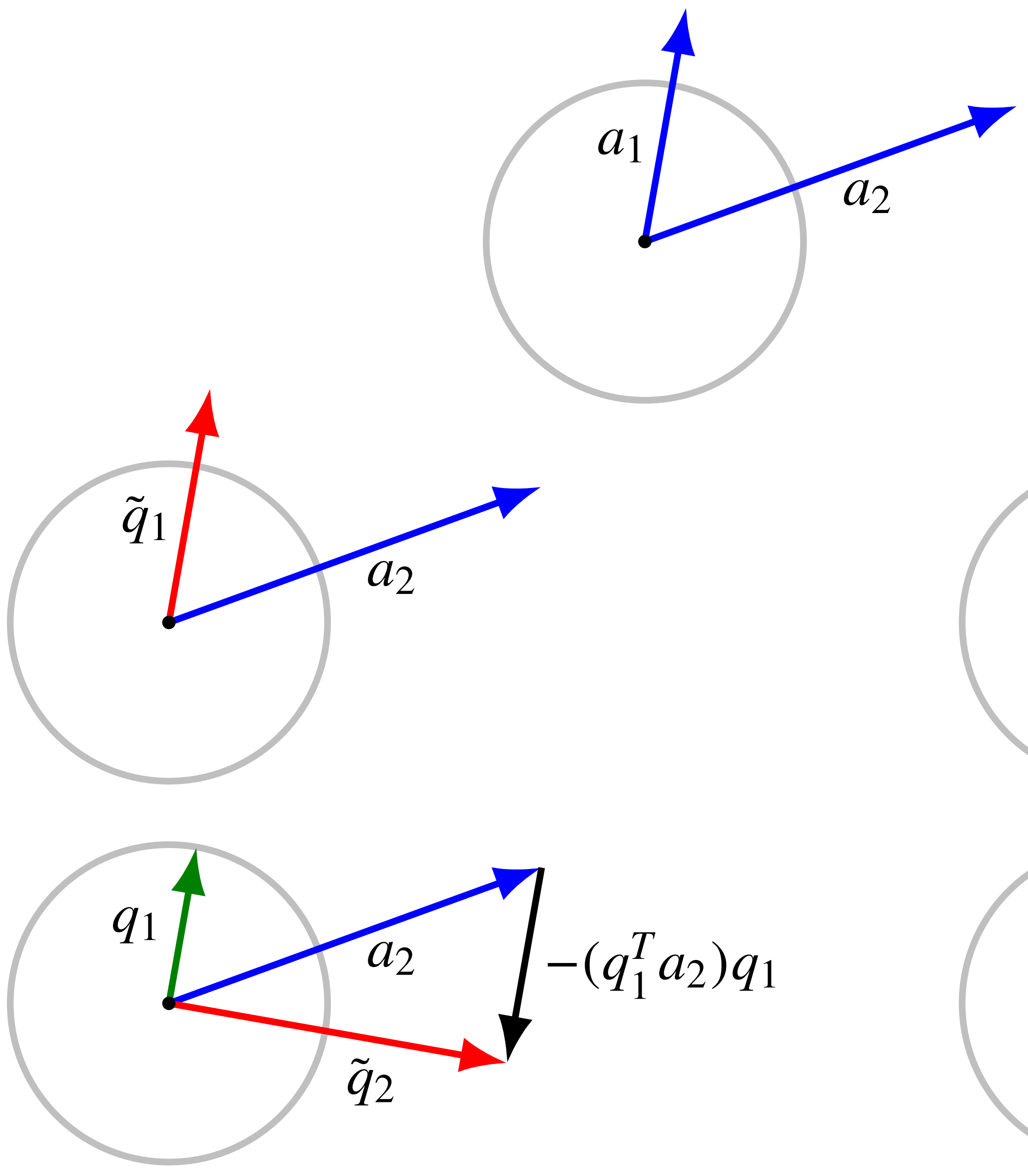
given n-vectors $a_1, \ldots, a_k$

for $i = 1, \ldots, k$

1. *Orthogonalization:* $\tilde{q}_i = a_i - (q_1^T a_i) q_1 - \cdots - (q_{i-1}^T a_i) q_{i-1}$
2. *Test for linear dependence:* if $\tilde{q}_i = 0$, quit
3. *Normalization:* $q_i = \tilde{q}_i / \|\tilde{q}_i\|$

- if G–S does not stop early (in step 2), $a_1, \ldots, a_k$ are linearly independent
- if G–S stops early in iteration $i = j$, then a_j is a linear combination of $a_1, \ldots, a_{j-1}$ (so $a_1, \ldots, a_k$ are linearly dependent)

Example



Analysis

let's show by induction that $q_1, \ldots, q_i$ are orthonormal

- assume it's true for $i - 1$
- orthogonalization step ensures that

$$\tilde{q}_i \perp q_1, \ldots, \tilde{q}_i \perp q_{i-1}$$

- to see this, take inner product of both sides with q_j, $j < i$

$$\begin{aligned} q_j^T \tilde{q}_i &= q_j^T a_i - (q_1^T a_i)(q_j^T q_1) - \cdots - (q_{i-1}^T a_i)(q_j^T q_{i-1}) \\ &= q_j^T a_i - q_j^T a_i = 0 \end{aligned}$$

- so $q_i \perp q_1, \ldots, q_i \perp q_{i-1}$
- normalization step ensures that $\|q_i\| = 1$

Analysis

assuming G–S has not terminated before iteration i

- a_i is a linear combination of $q_1, \ldots, q_i$:

$$a_i = \|\tilde{q}_i\| q_i + (q_1^T a_i) q_1 + \cdots + (q_{i-1}^T a_i) q_{i-1}$$

- q_i is a linear combination of $a_1, \ldots, a_i$: by induction on i,

$$q_i = (1/\|\tilde{q}_i\|) \left(a_i - (q_1^T a_i) q_1 - \cdots - (q_{i-1}^T a_i) q_{i-1} \right)$$

  and (by induction assumption) each $q_1, \ldots, q_{i-1}$ is a linear combination of $a_1, \ldots, a_{i-1}$

Early termination

suppose G–S terminates in step j

- a_j is linear combination of $q_1, \ldots, q_{j-1}$

$$a_j = (q_1^T a_j)q_1 + \cdots + (q_{j-1}^T a_j)q_{j-1}$$

- and each of $q_1, \ldots, q_{j-1}$ is linear combination of $a_1, \ldots, a_{j-1}$
- so a_j is a linear combination of $a_1, \ldots, a_{j-1}$

Complexity of Gram–Schmidt algorithm

- step 1 of iteration i requires $i-1$ inner products,

$$q_1^T a_i, \ldots, q_{i-1}^T a_i$$

  which costs $(i-1)(2n-1)$ flops

- $2n(i-1)$ flops to compute $\tilde{q}_i$
- $3n$ flops to compute $\|\tilde{q}_i\|$ and q_i
- total is

$$\sum_{i=1}^{k} ((4n-1)(i-1)+3n) = (4n-1)\frac{k(k-1)}{2} + 3nk \approx 2nk^2$$

  using $\sum_{i=1}^{k}(i-1) = k(k-1)/2$