13. The Singular-Value Decomposition (SVD) and applications

Last time

- Least-squares problems
- Minimise $\|A\,\mathbf{x} - \mathbf{b}\|^2$
- Optimization
- Solution via normal equations and QR
- Action of a matrix

Goals for today

- Action of a matrix A
- Eigen-factorisation (eigen-decomposition) for symmetric, square matrix

Goals for today

- Action of a matrix A
- Eigen-factorisation (eigen-decomposition) for symmetric, square matrix

- Singular-Value Decomposition (SVD)
  - Factorisation $\mathsf{A} = \mathsf{U}\Sigma\mathsf{V}^{\mathsf{T}}$ for *any* $(m \times n)$ matrix
- Applications to understanding and compressing data

The Singular-Value Decomposition (SVD)

How does a matrix act?

- Consider an arbitrary $(m \times n)$ matrix A

How does a matrix act?

- Consider an arbitrary $(m \times n)$ matrix A
- What is the *geometrical* action of the matrix?

How does a matrix act?

- Consider an arbitrary $(m \times n)$ matrix A
- What is the *geometrical* action of the matrix?
- For 2×2 matrices we know that they can:
  - do nothing (identity)
  - reflect
  - shear
  - rotate

How does a matrix act?

- Consider an arbitrary $(m \times n)$ matrix A
- What is the *geometrical* action of the matrix?
- For 2×2 matrices we know that they can:
  - do nothing (identity)
  - reflect
  - shear
  - rotate
- In general they send **parallelograms** to **parallelograms**

How does a matrix act?

- Consider an arbitrary $(m \times n)$ matrix A
- What is the *geometrical* action of the matrix?
- For 2×2 matrices we know that they can:
  - do nothing (identity)
  - reflect
  - shear
  - rotate
- In general they send **parallelograms** to **parallelograms**
- What about in higher dimensions?

Direction of maximal stretching

- One approach to this question is an eigen-factorisation
- We look for special directions $\mathbf{v}$ such that $\mathrm{A}\mathbf{v} = \lambda\mathbf{v}$

Direction of maximal stretching

- One approach to this question is an eigen-factorisation
- We look for special directions $\mathbf{v}$ such that $\mathrm{A}\mathbf{v} = \lambda\mathbf{v}$
- But that works *only* for (some) square matrices

Direction of maximal stretching

- One approach to this question is an eigen-factorisation
- We look for special directions **v** such that $\mathrm{A}\mathbf{v} = \lambda\mathbf{v}$
- But that works *only* for (some) square matrices
- We want a general approach that applies to *any* matrix

Direction of maximal stretching

- One approach to this question is an eigen-factorisation
- We look for special directions **v** such that $A\mathbf{v} = \lambda\mathbf{v}$
- But that works *only* for (some) square matrices
- We want a general approach that applies to *any* matrix

- Idea: Look for different special directions: Those associated with different amounts of **stretching**

Direction of maximal stretching

- One approach to this question is an eigen-factorisation
- We look for special directions **v** such that $A\mathbf{v} = \lambda\mathbf{v}$
- But that works *only* for (some) square matrices
- We want a general approach that applies to *any* matrix

- Idea: Look for different special directions: Those associated with different amounts of **stretching**
- E.g. Which is the direction **x** with *maximal* **stretching**?

Direction of maximal stretching

- We need to solve

$$\operatorname{argmax}_{\mathbf{x}} \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|}$$

Direction of maximal stretching

- We need to solve

$$\mathrm{argmax}_{\mathbf{x}} \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|}$$

- Equivalently: $\mathrm{argmax}_{\|x\|=1} \|A\mathbf{x}\|$

Direction of maximal stretching

- We need to solve

$$\operatorname{argmax}_{\mathbf{x}} \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|}$$

- Equivalently: $\operatorname{argmax}_{\|x\|=1} \|A\mathbf{x}\|$

- Another optimisation problem
- Also solvable using linear algebra

Action of a matrix

- Let's study its action on the unit sphere $\mathbb{S}^{n-1}$ in $\mathbb{R}^{n}$
- $\mathbb{S}^{n-1} := \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|^2 = 1\}$

Action of a matrix

- Let's study its action on the unit sphere $\mathbb{S}^{n-1}$ in $\mathbb{R}^{n}$
- $\mathbb{S}^{n-1} := \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|^2 = 1\}$
- The image is a set in $\mathbb{R}^{m}$

Action of a matrix

- Let's study its action on the unit sphere $\mathbb{S}^{n-1}$ in $\mathbb{R}^{n}$
- $\mathbb{S}^{n-1} := \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|^2 = 1\}$
- The image is a set in $\mathbb{R}^{m}$

- A visualisation suggests that $A(\mathbb{S}^{n-1})$ is an **ellipsoid** (hyper-ellipse) – a stretched and rotated sphere

Action of a matrix

- Let's study its action on the unit sphere $\mathbb{S}^{n-1}$ in $\mathbb{R}^{n}$
- $\mathbb{S}^{n-1} := \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|^2 = 1\}$
- The image is a set in $\mathbb{R}^{m}$

- A visualisation suggests that $A(\mathbb{S}^{n-1})$ is an **ellipsoid** (hyper-ellipse) – a stretched and rotated sphere

- Key quantities:
  - Lengths σ_i of semi-axes – stretches
  - Directions $\mathbf{u}_i$ of stretches

Collaboration I

Understanding the action of a matrix

Assume for now that A is square and invertible.

1. Suppose that $\mathbf{y}$ lies in the *image* of the sphere under A. What can you say about $\mathbf{y}$?
2. Can you expand this equation to get something more useful?

Action of a matrix II

- Let's think about the image of $\mathbb{S}^{n-1}$ under A

Action of a matrix II

- Let's think about the image of $\mathbb{S}^{n-1}$ under A
- Suppose $\mathbf{x} \in \mathbb{S}^{n-1}$
- Define $\mathbf{y} := \mathsf{A}\mathbf{x}$ to be its image

Action of a matrix II

- Let's think about the image of $\mathbb{S}^{n-1}$ under A
- Suppose $\mathbf{x} \in \mathbb{S}^{n-1}$
- Define $\mathbf{y} := \mathsf{A}\mathbf{x}$ to be its image
- Suppose that A is square and invertible for now

Action of a matrix II

- Let's think about the image of $\mathbb{S}^{n-1}$ under A
- Suppose $\mathbf{x} \in \mathbb{S}^{n-1}$
- Define $\mathbf{y} := \mathsf{A}\mathbf{x}$ to be its image
- Suppose that A is square and invertible for now

- Since $\|x\|^2 = 1$ we have $\quad \|\mathsf{A}^{-1}\mathbf{y}\|^2 = 1$
- So $\quad \mathbf{y}^\mathsf{T}(\mathsf{A}^{-1})^\mathsf{T}\mathsf{A}^{-1}\mathbf{y} = 1$

Action of a matrix II

- Let's think about the image of $\mathbb{S}^{n-1}$ under A
- Suppose $\mathbf{x} \in \mathbb{S}^{n-1}$
- Define $\mathbf{y} := \mathsf{A}\mathbf{x}$ to be its image
- Suppose that A is square and invertible for now

- Since $\|x\|^2 = 1$ we have $\quad \|\mathsf{A}^{-1}\mathbf{y}\|^2 = 1$
- So $\quad \mathbf{y}^\mathsf{T}(\mathsf{A}^{-1})^\mathsf{T}\mathsf{A}^{-1}\mathbf{y} = 1$
- Hence $\quad \mathbf{y}^\mathsf{T}\mathsf{S}\mathbf{y} = 1$
- Where $\mathsf{S} := (\mathsf{A}^{-1})^\mathsf{T}(\mathsf{A}^{-1})$ is symmetric

Spectral theorem for symmetric matrices

- Recall the **spectral theorem** for symmetric matrices

Spectral theorem for symmetric matrices

- Recall the **spectral theorem** for symmetric matrices
- It tells us that S has a *basis* of orthogonal eigenvectors, $\mathbf{q}_i$, for S:

$$\mathsf{S}\mathbf{q}_i = \lambda_i \mathbf{q}_i$$

Collaboration II

Eigen-factorisation

1. Use the spectral theorerm to find an equation involving S and a certain matrix that you can construct.
2. Hence write down a matrix factorisation of S, i.e. $S = \ldots$.
3. Can you use that to say something about the image vector $\mathbf{y}$?

Eigen-factorisation

- Let's build an orthogonal matrix Q whose columns are the orthonormal basis vectors $\mathbf{q}_i$:

Eigen-factorisation

- Let's build an orthogonal matrix Q whose columns are the orthonormal basis vectors $\mathbf{q}_i$:

$$\mathrm{Q} := \left[\mathbf{q}_1 \middle| \mathbf{q}_2 \middle| \cdots \middle| \mathbf{q}_n\right]$$

Eigen-factorisation

- Let's build an orthogonal matrix Q whose columns are the orthonormal basis vectors $\mathbf{q}_i$:

$$\mathrm{Q} := \left[\mathbf{q}_1 \middle| \mathbf{q}_2 \middle| \cdots \middle| \mathbf{q}_n\right]$$

- Since $\mathrm{S}\mathbf{q}_i = \lambda_i \mathbf{q}_i$ we get $\mathrm{SQ} = \mathrm{Q}\Lambda$
- With

$$\Lambda := \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_m \end{bmatrix}$$

Eigen-factorisation

- Let's build an orthogonal matrix Q whose columns are the orthonormal basis vectors $\mathbf{q}_i$:

$$\mathsf{Q} := \left[\mathbf{q}_1 \middle| \mathbf{q}_2 \middle| \cdots \middle| \mathbf{q}_n\right]$$

- Since $\mathsf{S}\mathbf{q}_i = \lambda_i \mathbf{q}_i$ we get $\mathsf{S}\mathsf{Q} = \mathsf{Q}\Lambda$
- With

$$\Lambda := \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_m \end{bmatrix}$$

- Hence $\mathsf{S} = \mathsf{Q}\Lambda\mathsf{Q}^\mathsf{T}$ – the **eigen-factorisation**

Action of a matrix III

- So $\mathbf{y}^{\mathsf{T}} \mathrm{Q} \Lambda \mathrm{Q}^{\mathsf{T}} \mathbf{y} = 1$

Action of a matrix III

- So $\mathbf{y}^{\mathsf{T}} \mathrm{Q} \Lambda \mathrm{Q}^{\mathsf{T}} \mathbf{y} = 1$
- Hence $\mathbf{z}^{\mathsf{T}} \Lambda \mathbf{z} = 1$

  where $\mathbf{z} := \mathrm{Q}^{\mathsf{T}} \mathbf{y}$

Action of a matrix III

- So $\mathbf{y}^\mathsf{T} \mathtt{Q} \Lambda \mathtt{Q}^\mathsf{T} \mathbf{y} = 1$
- Hence $\mathbf{z}^\mathsf{T} \Lambda \mathbf{z} = 1$

  where $\mathbf{z} := \mathtt{Q}^\mathsf{T} \mathbf{y}$
- So $\lambda_1 z_1^2 + \cdots + \lambda_n z_n^2 = 1$

Action of a matrix III

- So $\mathbf{y}^{\mathsf{T}} \mathtt{Q} \Lambda \mathtt{Q}^{\mathsf{T}} \mathbf{y} = 1$
- Hence $\mathbf{z}^{\mathsf{T}} \Lambda \mathbf{z} = 1$

  where $\mathbf{z} := \mathtt{Q}^{\mathsf{T}} \mathbf{y}$
- So $\lambda_1 z_1^2 + \cdots + \lambda_n z_n^2 = 1$

- Can show $(\mathtt{A}^{\mathsf{T}} \mathtt{A})^{-1}$ has eigenvalues $\lambda_i > 0$ (exercise)

Action of a matrix III

- So $\mathbf{y}^\mathsf{T} \mathsf{Q} \Lambda \mathsf{Q}^\mathsf{T} \mathbf{y} = 1$
- Hence $\mathbf{z}^\mathsf{T} \Lambda \mathbf{z} = 1$

  where $\mathbf{z} := \mathsf{Q}^\mathsf{T} \mathbf{y}$
- So $\lambda_1 z_1^2 + \cdots + \lambda_n z_n^2 = 1$

- Can show $(\mathsf{A}^\mathsf{T} \mathsf{A})^{-1}$ has eigenvalues $\lambda_i > 0$ (exercise)
- Hence $\mathbf{z}$ lies on an **ellipse**, with semi-axis lengths $\sigma_i := \sqrt{\lambda_i}$

Action of a matrix III

- So $\mathbf{y}^\mathsf{T} \mathrm{Q} \Lambda \mathrm{Q}^\mathsf{T} \mathbf{y} = 1$
- Hence $\mathbf{z}^\mathsf{T} \Lambda \mathbf{z} = 1$

  where $\mathbf{z} := \mathrm{Q}^\mathsf{T} \mathbf{y}$
- So $\lambda_1 z_1^2 + \cdots + \lambda_n z_n^2 = 1$

- Can show $(\mathrm{A}^\mathsf{T} \mathrm{A})^{-1}$ has eigenvalues $\lambda_i > 0$ (exercise)
- Hence $\mathbf{z}$ lies on an **ellipse**, with semi-axis lengths $\sigma_i := \sqrt{\lambda_i}$
- Thus $\mathbf{y}$ lies on a rotated ellipse !

Singular values and singular vectors

- σ_i are **singular values** of A – this is a terrible name!

Singular values and singular vectors

- σ_i are **singular values** of A – this is a terrible name!

- The $\mathbf{u}_i$, the columns of Q, are the **left singular vectors**
- I.e. the directions in which the ellipse is stretched

Singular values and singular vectors

- σ_i are **singular values** of A – this is a terrible name!

- The $\mathbf{u}_i$, the columns of Q, are the **left singular vectors**
- I.e. the directions in which the ellipse is stretched

- $\mathbf{v}_i$ such that $A\mathbf{v}_i = \sigma_i \mathbf{u}_i$ are the **right singular vectors**

Singular values and singular vectors

- σ_i are **singular values** of A – this is a terrible name!

- The $\mathbf{u}_i$, the columns of Q, are the **left singular vectors**
- I.e. the directions in which the ellipse is stretched

- $\mathbf{v}_i$ such that A$\mathbf{v}_i = \sigma_i \mathbf{u}_i$ are the **right singular vectors**
- We arrange the singular values in decreasing order: $\sigma_1 \geq \sigma_2 \geq \cdots$

Singular-value decomposition (SVD)

- *Any* $(m \times n)$ matrix A has an SVD:

$$\mathsf{A} = \mathsf{U}\Sigma\mathsf{V}^\mathsf{T} = \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^\mathsf{T}$$

  - the columns of U are $\mathbf{u}_i$
  - the columns of V are $\mathbf{v}_i$
  - Σ is diagonal matrix with σ_i on diagonal

Singular-value decomposition (SVD)

- *Any* $(m \times n)$ matrix A has an SVD:

$$\mathsf{A} = \mathsf{U}\Sigma\mathsf{V}^{\mathsf{T}} = \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^{\mathsf{T}}$$

  - the columns of U are $\mathbf{u}_i$
  - the columns of V are $\mathbf{v}_i$
  - Σ is diagonal matrix with σ_i on diagonal

- Note: only square, diagonalisable matrices have eigen-factorisations
- But *any* matrix has an SVD

Singular-value decomposition (SVD)

- *Any* $(m \times n)$ matrix A has an SVD:

$$\mathsf{A} = \mathsf{U}\Sigma\mathsf{V}^{\mathsf{T}} = \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^{\mathsf{T}}$$

  - the columns of U are $\mathbf{u}_i$
  - the columns of V are $\mathbf{v}_i$
  - Σ is diagonal matrix with σ_i on diagonal

- Note: only square, diagonalisable matrices have eigen-factorisations
- But *any* matrix has an SVD
- The SVD is often more useful than an eigen-factorisation

Collaboration III

Calculating the SVD

1. How could you *calculate* the SVD, using what we have seen so far in the course?

Calculating the SVD

- We can *calculate* the SVD
- There are more efficient algorithms that are beyond the scope of the course

Calculating the SVD

- We can *calculate* the SVD
- There are more efficient algorithms that are beyond the scope of the course

- Form $\mathsf{B} := \mathsf{A}^T\mathsf{A}$

Calculating the SVD

- We can *calculate* the SVD
- There are more efficient algorithms that are beyond the scope of the course

- Form $B := A^T A$
- Calculate the eigenvalues and eigenvectors of the symmetric matrix B

Calculating the SVD

- We can *calculate* the SVD
- There are more efficient algorithms that are beyond the scope of the course

- Form $B := A^T A$
- Calculate the eigenvalues and eigenvectors of the symmetric matrix B
- Do the same for $C := AA^T$

Calculating the SVD

- We can *calculate* the SVD
- There are more efficient algorithms that are beyond the scope of the course

- Form $B := A^T A$
- Calculate the eigenvalues and eigenvectors of the symmetric matrix B
- Do the same for $C := AA^T$
- **Exercise**: Relate these to the singular values and singular vectors of A

Rank of a matrix

- The SVD gives as information about the action of a matrix

Rank of a matrix

- The SVD gives as information about the action of a matrix
- E.g. the **rank** of a matrix is the dimension of the *image* of $\mathbb{R}^n$ (i.e. the column rank)

Rank of a matrix

- The SVD gives as information about the action of a matrix
- E.g. the **rank** of a matrix is the dimension of the *image* of $\mathbb{R}^n$ (i.e. the column rank)
- This is given by the *number of non-zero singular values* !

Rank of a matrix

- The SVD gives as information about the action of a matrix
- E.g. the **rank** of a matrix is the dimension of the *image* of $\mathbb{R}^n$ (i.e. the column rank)
- This is given by the *number of non-zero singular values* !
- Usually floating-point round-off error will make the singular values non-zero
- So we need to look at the **numerical rank**

– the number of “non-zero” singular values above a threshold

Best-fitting subspace

- Suppose that we want to find the best-fitting plane, or dimension-k hyperplane, to a cloud of data

Best-fitting subspace

- Suppose that we want to find the best-fitting plane, or dimension-k hyperplane, to a cloud of data
- It turns out that the SVD solves this problem *immediately*!

Best-fitting subspace

- Suppose that we want to find the best-fitting plane, or dimension-k hyperplane, to a cloud of data
- It turns out that the SVD solves this problem *immediately*!
- We first **de-mean** the data (subtract the mean from each component)
- Then the SVD gives the **best approximation** to a matrix A by a matrix of rank k as

$$\mathsf{A}_k := \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^{\mathsf{T}}$$

Best-fitting subspace

- Suppose that we want to find the best-fitting plane, or dimension-k hyperplane, to a cloud of data
- It turns out that the SVD solves this problem *immediately*!
- We first **de-mean** the data (subtract the mean from each component)
- Then the SVD gives the **best approximation** to a matrix A by a matrix of rank k as

$$\mathsf{A}_k := \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^\mathsf{T}$$

- For a data matrix the SVD gives the best approximation by a k-dimensional subspace

Principal Component Analysis (PCA)

- Given an $(m \times n)$ **data matrix**

Principal Component Analysis (PCA)

- Given an $(m \times n)$ **data matrix**
- How many “important directions” are there in the data?
- i.e. what is the “dimension” of the data

Principal Component Analysis (PCA)

- Given an $(m \times n)$ **data matrix**
- How many “important directions” are there in the data?
- i.e. what is the “dimension” of the data

- Idea: We want to ignore “noise”

Principal Component Analysis (PCA)

- Given an $(m \times n)$ **data matrix**
- How many “important directions” are there in the data?
- i.e. what is the “dimension” of the data

- Idea: We want to ignore “noise”
- This is basically asking for the numerical rank!

PCA II

- Calculate the SVD

PCA II

- Calculate the SVD
- Plot the "importance" of each direction

  σ_i or σ_i^2 or $\frac{\sigma_i^2}{\sum_j \sigma_j^2}$

  as a function of i

PCA II

- Calculate the SVD
- Plot the "importance" of each direction

  σ_i or σ_i^2 or $\frac{\sigma_i^2}{\sum_j \sigma_j^2}$

  as a function of i
- Called a "scree plot"
- It often has an "elbow" after which the σ_i are small
- This helps to choose the k for the low-rank approximation

$$\mathsf{A}_k := \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^{\mathsf{T}}$$

Summary

- The SVD decomposes the action of a matrix
  - rotation + stretch + rotation
- *Any* $(m \times n)$ matrix A has an SVD

$$\mathsf{A} = \mathsf{U}\Sigma\mathsf{V}^{\mathsf{T}} = \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^{\mathsf{T}}$$

- The SVD is closely related to the eigen-factorisation of the symmetric matrices $\mathsf{A}^{\mathsf{T}}\mathsf{A}$ and $\mathsf{A}\mathsf{A}^{\mathsf{T}}$
- Applications to analysing and compressing data