Numerical Methods (CS 357)

Worksheet

Part 1. Objectives

- Apply eigenvalue computation to Markov chains
- Learn how to compute the SVD
- See the connection between the norm and the singular values
- Learn how to compute a rank-k best-approximation to a matrix

Part 2. The SVD and the 2-norm

A matrix A has the Singular Value Decomposition

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 7 & 0 \\ 0 & 2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{3}}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}.$$

What is the largest value that $\|Ax\|_2$ can attain for any x with $\|x\|_2 = 1$?

Part 3. SVD and rank-1 approximation

A matrix A has the Singular Value Decomposition

$$A = \begin{bmatrix} | & | & \cdots & | \\ u_1 & u_2 & \cdots & u_m \\ | & | & \cdots & | \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & \sigma_n \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} - & v_1 & - \\ - & v_2 & - \\ \vdots & \vdots & \vdots \\ - & v_n & - \end{bmatrix}$$

with $\sigma_1 \geq \sigma_2 \geq \cdots \sigma_n \geq 0$.

What is the matrix B that has rank 1 and minimizes $\|A - B\|_2$?

(A) $B = u_1 v_1^T$

(B) $B = \sigma_1 u_1 v_1^T$

(C) $B = v_1 u_1^T$

(D) $B = \sigma_1 v_1 u_1^T$

Part 4. Pseudoinverse

A matrix A has the Singular Value Decomposition

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 7 & 0 \\ 0 & 2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{3}}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}.$$

If you were to compute an SVD of the *pseudoinverse* of A, what would be the largest singular value in it?

Part 5. Rank-1 approximation

You are given a matrix `A`. Compute the rank-1 approximation to A as `A1` using the SVD of `A`. (Use `U, sigma, VT = numpy.linalg.svd(A)`. Note that the singular values `sigma` are returned as a vector.)

Can you find the 2-norm of `A` from the SVD of `A`? Assign it to `norm_A`. Print your prediction as well as the computed 2-norm to check.

Can you find the 2-norm of `A1` from the SVD of `A`? Assign it to `norm_A1`. Print your prediction as well as the computed 2-norm to check.

Can you find the 2-norm of `A-A1` from the SVD of `A`? Assign it to `norm_AmA1`. Print your prediction as well as the computed 2-norm to check.

INPUTS: `A`, an $n \times n$ matrix

OUTPUTS:

- `A1`, the rank-1 best-approximation to `A` in the 2-norm
- `norm_A`, which should equal $\|\mathrm{A}\|_2$
- `norm_A1`, which should equal $\|\mathrm{A1}\|_2$
- `norm_AmA1`, which should equal $\|\mathrm{A} - \mathrm{A1}\|_2$

```
import numpy as np
import numpy.linalg as la
```