

Homework 2

Numerical Analysis Fall 2024

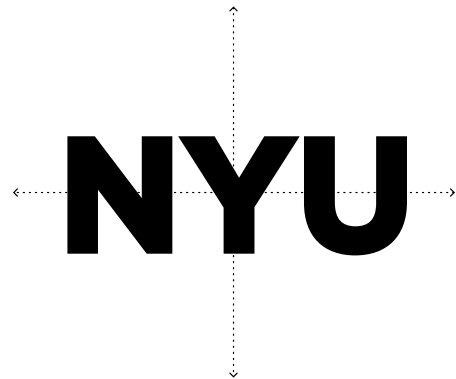
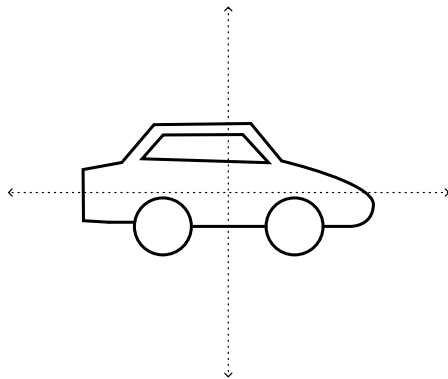
Instructions:

- Due 09/24 at 5:00pm on Gradescope.
- You must follow the submission policy in the syllabus

Problem 1. Consider the matrix

$$\mathbf{A} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 3 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \cos \pi/3 & -\sin \pi/3 \\ \sin \pi/3 & \cos \pi/3 \end{bmatrix}$$

- (a) Find an SVD $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ of \mathbf{A} . Hint: think about why the given factorization is not an SVD.
- (b) Draw what \mathbf{V}^T does to the following points (here a point is anything shown in black, and the dotted lines represent the x and y axes.):



- (c) Draw what $\mathbf{\Sigma}\mathbf{V}^T$ does to the points.
- (d) Draw what $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ does to the points.

Problem 2. Recall:

$$\|\mathbf{A}\|_2 = \max_{\mathbf{z} \neq 0} \frac{\|\mathbf{A}\mathbf{z}\|_2}{\|\mathbf{z}\|_2}, \quad \|\mathbf{A}\|_F^2 = \sum_{i,j} [\mathbf{A}]_{i,j}^2$$

We will prove the identity $\|\mathbf{AB}\|_F \leq \|\mathbf{A}\|_2 \|\mathbf{B}\|_F$.

- (a) Prove that $\|\mathbf{Ax}\|_2 \leq \|\mathbf{A}\|_2 \|\mathbf{x}\|_2$ for any vector \mathbf{x} .
- (b) Suppose \mathbf{X} is a $n \times m$ matrix. Write $\|\mathbf{X}\|_F$ in terms of the column-norms $\|[\mathbf{X}]_{:,i}\|_2$.
- (c) Use (a) and (b) to prove the identity $\|\mathbf{AB}\|_F \leq \|\mathbf{A}\|_2 \|\mathbf{B}\|_F$.

Problem 3. Let \mathbf{A} be a $m \times n$ matrix with SVD $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ where $\mathbf{\Sigma}$ contains the singular values $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$. We will prove that $\|\mathbf{A}\|_2 = \sigma_1$.

- (a) Prove that if $\mathbf{U}^T\mathbf{U} = \mathbf{I}$ then $\|\mathbf{U}\mathbf{z}\|_2 = \|\mathbf{z}\|_2$ for any vector \mathbf{z} .
- (b) Prove that if $\mathbf{V}^T\mathbf{V} = \mathbf{I}$, then $\mathbf{V}^T\mathbf{x} = \mathbf{0}$ if and only if $\mathbf{x} = \mathbf{0}$.
- (c) Prove that $\|\mathbf{A}\|_2 = \|\mathbf{\Sigma}\|_2$.
- (d) It remains to show that $\|\mathbf{\Sigma}\|_2 = \sigma_1$.
 - (i) Compute $\|\mathbf{\Sigma}\mathbf{x}\|_2^2$ for an arbitrary vector $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$.
 - (ii) Show that $\|\mathbf{\Sigma}\mathbf{x}\|_2^2 \leq \sigma_1^2 \|\mathbf{x}\|_2^2$
 - (iii) Show that there exists a vector \mathbf{x} such that $\|\mathbf{\Sigma}\mathbf{x}\|_2 = \sigma_1 \|\mathbf{x}\|_2$.
 - (iv) Conclude that $\|\mathbf{\Sigma}\|_2 = \sigma_1$.

Problem 4. Download the numpy data file from this link: https://drive.google.com/file/d/18Xf0029XXm3ENWfe3Z0a--Cf6tjyCViz/view?usp=drive_link

Use the following code to import the file into numpy.

```
import numpy as np
import matplotlib.pyplot as plt

im = np.load('change this path/CIMS.npy')
```

If you are using google colab, you can copy the CIMS.npy file to your own drive and then

```
from google.colab import drive
drive.mount('/content/gdrive')

im = np.load('gdrive/MyDrive/change this path/CIMS.npy')
```

In both cases, form a matrix from the image data.

```
A = np.mean(im, axis=2)
```

Here we obtain \mathbf{A} by averaging the red, green, and blue channels of the image. This results in a black and white image.

- (a) Plot the image using `plt.imshow`. You may want to use the colormap 'Greys_r' so that it looks like a greyscale image.

- (b) Compute the reduced SVD of \mathbf{A} . You can use `full_matrices=False` to get the reduced SVD. This will be much faster than computing the full SVD.

For each $k = 1, 10, 100, 200$, make a plot of the best rank- k approximation \mathbf{A}_k to \mathbf{A} (i.e. via truncated SVD). Label each plot with the rank k as well as the relative error $\|\mathbf{A} - \mathbf{A}_k\|_F / \|\mathbf{A}\|_F$

- (c) Remark on the quality of the plots.

How many numbers are required to store \mathbf{A} ? How many numbers are required to store the rank- k truncated SVD (as a factorization)?

Problem 5. Computing the SVD is expensive, but randomization can help us!

- (a) Randomized numerical linear algebra (RandNLA) is the study of the use of randomness in numerical linear algebra algorithms. One of the most famous randNLA algorithms is the randomized SVD. A simple version for approximating the SVD of a $m \times n$ matrix \mathbf{A} can be described in several lines:

- Choose a $n \times k$ matrix \mathbf{R} with standard normal random entries
- Compute $\mathbf{X} = \mathbf{A}\mathbf{R}$
- Compute $\mathbf{Q}, \mathbf{\Sigma} = \text{REDUCED-SVD}(\mathbf{X})$
- $\hat{\mathbf{U}}, \hat{\mathbf{\Sigma}}, \hat{\mathbf{V}}^T = \text{REDUCED-SVD}(\mathbf{Q}^T \mathbf{A})$:
- Return approximate SVD of \mathbf{A} : $(\mathbf{Q}\hat{\mathbf{U}})\hat{\mathbf{\Sigma}}\hat{\mathbf{V}}^T$

Implement this algorithm with the same matrix \mathbf{A} as in Problem 3. To generate the random matrix, you can use `np.random.randn(n, k)`.

Again make sure to use `full_matrices=False` when computing the SVD of $\mathbf{Q}^T \mathbf{A}$. Compare this to long the whole randomized SVD took (all of the steps) with $k = 100$ against the time to compute the exact SVD in the previous problem.

- (b) Prove that the factors $\mathbf{Q}\hat{\mathbf{U}}$, $\hat{\mathbf{\Sigma}}$ and $\hat{\mathbf{V}}^T$ have the same properties as a SVD; i.e. $\mathbf{Q}\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$ have orthonormal columns and $\hat{\mathbf{\Sigma}}$ is diagonal with non-negative entries.
- (c) Make a plot of the rank $k = 100$ truncated SVD (from problem 3) and the $k = 100$ randomized SVD. Show the relative errors $\|\mathbf{A} - (\mathbf{Q}\hat{\mathbf{U}})\hat{\mathbf{\Sigma}}\hat{\mathbf{V}}^T\|_F / \|\mathbf{A}\|_F$ for each.
- (d) How long did this algorithm take to run vs. the reduced SVD in problem 3? Why was it so much faster? Hint: what are the dimensions of the matrices which we take the SVD of using this approach?