Labs

Set 13 (DM562)

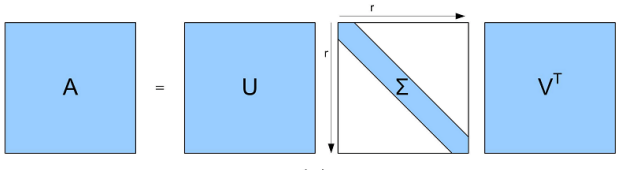
DM562 Scientific Programming
DM857 Introduction to Programming
DS830 Introduction to Programming

Singular Value Decomposition and Image Compression

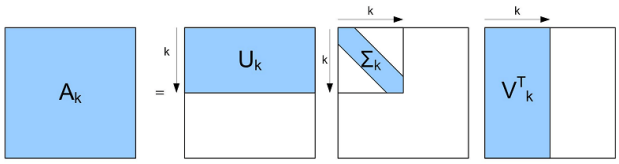
The notion of singular value decomposition (SVD) from linear algebra has found applications in many fields, especially in signal processing and data science as a dimensionality reduction technique. This is because SVD provides an answer to the problem of finding the matrix A_k of rank k that is closest to a given matrix A of rank r for any $k < r$.

For the aims of this lab we don't need to go into the details of the mathematical definition of SVD or "closest". It suffices to know that:

(a) Given a real $r \times n$ matrix A its SVD is $A = U\Sigma V^*$ where Σ is a rectangular diagonal matrix.



(b) The first k singular values (truncate U, Σ, V^* to $k < m$), are a decomposition of the matrix of rank k closest to A.



To compute the SVD of a matrix, you can use the `svd`[1] function from module `linalg` of `numpy`. This library offers a number of classes and functions for representing and operating on matrices[2]. Other implementations of SVD are available from `scipy`[3] or `sklearn`[4].

Image compression is a type of data compression applied to digital images to reduce their cost for storage or transmission. Since a digital image is essentially a matrix where each element stores the colour of the corresponding pixel, it can be compressed by computing its SVD and then taking a number k of singular values. This compression comes at a cost: by discarding some singular values we lose some information about the original image.

[1] `https://numpy.org/doc/stable/reference/generated/numpy.linalg.svd.html`
[2] `https://numpy.org/doc/stable/reference/generated/numpy.matrix.html`
[3] `https://docs.scipy.org/doc/scipy/reference/generated/scipy.linalg.svd.html`
[4] `https://scikit-learn.org/stable/modules/classes.html#module-sklearn.decomposition`

Write a program that write a Python program that applies this compression method with varying values for k. To load an image and store its pixels as a list, you can use the `Image`[5] class from module `PIL` (aka "Pillow").

```
from PIL import Image
image = Image.open('some_image.jpg').convert('L')
```

To display an image stored in a list, you can use the `imshow`[6] function from `matplotlib`.

[5] `https://pillow.readthedocs.io/en/stable/reference/Image.html`
[6] https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.imshow.html