

Principal Component Analysis

Task 1. In this task, we will once again work with the MNIST training set as provided on Moodle. Choose three digit classes, e.g. 1, 2 and 3 and load $N=1000$ images from each of the classes to the workspace. Store the data in a floating point matrix X of shape $(784, 3*N)$ normalized to the number range $[0, 1]$. Furthermore, generate a color label matrix C of dimensions $(3*N, 3)$. Each row of C assigns an RGB color vector to the respective column of X as an indicator of the digit class. Choose $[0, 0, 1]$, $[0, 1, 0]$ and $[1, 0, 0]$ for the three digit classes.

- Compute the row-wise mean μ of X and subtract it from each column of X . Save the results as X_c .
- Use `np.linalg.svd` with `full_matrices=False` to compute the singular value decomposition $[U, \text{Sigma}, VT]$ of X_c . Make sure the matrices are sorted in descending order with respect to the singular values.
- Use `reshape` in order to convert μ and the first three columns of U to $(28, 28)$ -matrices. Plot the resulting images. What do you see?
- Compute the matrix $S = \text{np.dot}(\text{np.diag}(\text{Sigma}), VT)$. Note that this yields the same result as $S = \text{np.dot}(U.T, X_c)$. The S matrix contains the $3*N$ scores for the principal components 1 to 784. Create a 2D scatter plot with C as its color parameter in order to plot the scores for the first *two* principal components of the data.

Task 2. In this task, we consider the problem of choosing the number of principal vectors. Assuming that $\mathbf{X} \in \mathbb{R}^{p \times N}$ is the centered data matrix and $\mathbf{P} = \mathbf{U}_k \mathbf{U}_k^\top$ is the projector onto the k -dimensional principal subspace, the dimension k is chosen such that the fraction of overall energy contained in the projection error does not exceed ϵ , i.e.

$$\frac{\|\mathbf{X} - \mathbf{P}\mathbf{X}\|_F^2}{\|\mathbf{X}\|_F^2} = \frac{\sum_{i=1}^M \|\mathbf{x}_i - \mathbf{P}\mathbf{x}_i\|^2}{\sum_{i=1}^N \|\mathbf{x}_i\|^2} \leq \epsilon,$$

where ϵ is usually chosen to be between 0.01 and 0.2.

The MIT VisTex database as provided on Moodle consists of a set of 167 RGB texture images of sizes $(512, 512, 3)$. Download the ZIP file, unpack it and make yourself familiar with the directory structure.

- a) After preprocessing the entire image set (converting to normalized grayscale matrices), divide the images into non overlapping tiles of sizes $(64, 64)$ and create a centered data matrix X_c of size (p, N) from them, where $p=64*64$ and $N=167*(512/64)*(512/64)$.
- b) Compute the SVD of X_c and make sure the singular values are sorted in descending order.
- c) Plot the fraction of signal energy contained in the projection error¹ for the principal subspace dimensions 0 to p . How many principal vectors do you need to retain 80%, 90%, 95% or 99% of the original signal energy?
- d) Discuss: Can you imagine a scenario, where signal energy is a bad measure of useful information?

Helpful Python/Numpy functions

```
import imageio          contains imread
import matplotlib.pyplot contains plotting functionalities
```

¹Note that you do not need to evaluate any norms or projections. All you need is the result of subtask b)