

Introduction to Data Science 2022

Assignment 4

François Lauze, Thomas Hamelryck

Your solution to Assignment 4 must be uploaded to Absalon no later than Friday March 18th, 2022, 22:00.

Guidelines for the assignment:

- **The assignments in IDS must be completed and written individually.** This means that your code and report must be written completely by yourself.
- Upload your report as a single PDF file (no Word) named `firstname.lastname.pdf`.
- Upload your Python code. Upload it in a zip archive, even if there is only one file. Expected is either one or several `.py`. The zip archive must also contain the data files used in the assignments. *Remember that they should be on the current `.py` file directory and read from the current folder in your python script (i.e, no hard-coded path!)*

Total 50 points.

Data exploration with PCA

In the first part of the assignment we will work with the diatoms dataset, see the appendix below for details.

Exercise 1 (Plotting cell shapes, 10 points). Plot one of the cells by plotting the landmark points and interpolating between subsequent landmark points.

Next, plot all the cells on top of each other. Can you see any dataset tendencies from this plot?

When plotting, make sure to make the axes equal to get the right dimensions. If you use `matplotlib.pyplot` this can be done by typing `plt.axis('equal')`.

Deliverables. A plot of a cell, a plot of many cells, and a short description.

Exercise 2 (Visualizing variance in visual data, 10 points). Now, you will visualize the spatial variance of the cells by plotting some instances of the first three PCs. That is, if the mean of the data is given by m , you are going to plot the "cells"

$$\begin{array}{ccccc} m - 2\sigma_1 e_1 & m - \sigma_1 e_1 & m & m + \sigma_1 e_1 & m + 2\sigma_1 e_1 \\ m - 2\sigma_2 e_2 & m - \sigma_2 e_2 & m & m + \sigma_2 e_2 & m + 2\sigma_2 e_2 \\ m - 2\sigma_3 e_3 & m - \sigma_3 e_3 & m & m + \sigma_3 e_3 & m + 2\sigma_3 e_3. \end{array}$$

where the e_1, e_2 and e_3 are the eigenvectors defining the first three PCs, and σ_1, σ_2 and σ_3 denote the standard deviation of the data projected onto each of the first three PCs.

Plot the five cells corresponding to each PC in a single plot, and illustrate the temporal development with a changing colour. This can, for instance, be done by importing a colormap with `blues = plt.get_cmap('Blues')`, where `blues(x)` returns a different shade of blue for every number x between 0 and 1.

Describe the variance captured by the three components.

Deliverables. Three plots with sequences of cells showing the variance. A description of the three components.

Exercise 3 (Critical thinking, 10 points).

- a) (5 points) Assume that you perform each of the following preprocessing steps *prior* to performing PCA. What is the effect on the PCA result? Is it a good idea?
 - i) Centering
 - ii) Standardization
 - iii) Whitening
- b) (5 points) On the PCA toy dataset, run PCA and visualize the projection onto the first 2 PCs, as in the previous assignment. Repeat the procedure and leave out the last 2 datapoints. You should see a dramatic difference in result. Do you see the hidden structure? What happened? In this exercise, if you have not implemented PCA as part of Assignment 2, it is OK to use a pre-implemented version from e.g. `scikit-learn`.

Deliverables. a) Three short arguments, b) two dataset plots and an explanation.

Clustering II

This exercise continues Exercise 2 and Exercise 3 from the previous Assignment 3.

In Exercise 2 from Assignment 3, you were asked to perform PCA on `IDSWeedCropTrain.csv` and to visualize the data by projecting it on the first two 2 principal components. In Exercise 3 from Assignment 3, you were asked to cluster the training data in `IDSWeedCropTrain.csv` using k -means clustering with $k = 2$ using the first two data points in `IDSWeedCropTrain.csv` as starting points. Now we bring these two exercises together. You are suppose to visualize the cluster centres.

Exercise 4 (Clustering II, 10 points). Visualize the data in `IDSWeedCropTrain.csv` by projecting it onto its first principal components (as in Exercise 2 from Assignment 3). Colour the data points according to their class. Take the centres you found in Exercise 3 from Assignment 3 (2-means clustering of the input data in `IDSWeedCropTrain.csv`, the cluster centres initialized with the first two data points). Then project the centres onto the first two principal components found in the previous step and visualize them together with the data points (i.e., in the same plot). Briefly discuss whether you got meaningful clusters.

Deliverables. Description of software used; projection of the two cluster centres (i.e., two two-dimensional vectors), a 2D plot visualizing the data and the cluster centres, short discussion of results

Exercise 5 (Clustering III, 10 points). Perform 2-means clustering on the `occupancy_data.csv` and report the center values as you did in exercise 3 of Assignment 3 with the crop dataset. Initialise you 2-means the same way. Visualise the data in `occupancy_data.csv` by projecting it onto its first principal components (as in Exercise 2 from Assignment 3). Colour the data points according to their class. Take the centres you found above and project them onto the first two principal components found in the previous step and visualize them together with the data points (i.e., in the same plot). Briefly discuss whether you got meaningful clusters.

Deliverables. Description of software used; projection of the two cluster centres (i.e., two two-dimensional vectors), a 2D plot visualizing the data and the cluster centres, short discussion of results

Exercise 6 (Virtual Chocolate Box (no points)). Using Matplotlib3d, redo the previous visualisation for the data in `occupancy_data.csv` by projecting it onto their 3 first principal components, colouring each point by the corresponding cluster and display the k-means data centres. Then display the data again (separately) but this time colour it by the true class labels of the data. *Deliverables.*

Description of the software. Short discussion of results.

Appendix: Data material

Diatoms

Diatoms are single celled algae, which can be classified into types (taxa) dependent on different features of e.g. their shape. We will perform principal component analysis on outlines of diatoms to explore their shape. The paper Jalba et al. [2006] (found on Absalon) contains more information about diatoms (the pages 338 and 339).

Figure 6 in the paper shows examples of diatoms and outlines of diatoms.

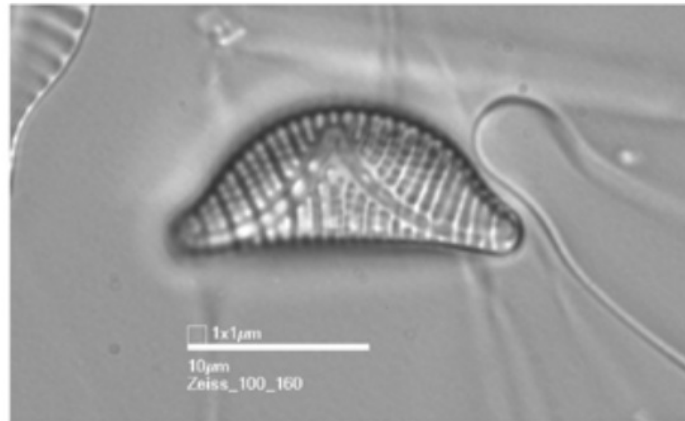


Figure 1: An example diatom.

The dataset we will explore contains 780 outlines of diatoms represented by points sampled along the outline of each diatom. The dataset is a modified version of the data used in the paper. We have redistributed the points so that there is the same number of points along each outline, and the positions of the points on the outlines are approximately the same for all diatoms.

Each outline is represented by the coordinates of 90 points in the plane. The 90 x - and y -coordinates for each outline can be considered a vector in \mathbb{R}^n , $n = 180$. When we refer to the mean of the dataset and compute PCA below, we will consider the diatoms observations in \mathbb{R}^{180} . The mean will for example also be a vector in \mathbb{R}^{180} .

The file `diatoms.txt` contains the outlines. It can be opened using the command

```
data = np.loadtxt('diatoms.txt')
```

The points are organized so that the x - and y -coordinates come in pairs, ie. each observation is on the form

$$x_1, y_1, \dots, x_{90}, y_{90}.$$

Please follow this convention when you work out your solution.

PCA toy dataset

This synthetic dataset can be opened with the command

```
data = np.loadtxt('pca_toydata.txt').
```

The dataset consists of 501 data points. It contains hidden structure – can you find it? (Hint: do not go and look for a complicated one, and PCA is not nice with it!)

References

- A. Jalba, M. H. F. Wilkinson, and J. B. T. M. Roerdink. Shape representation and recognition through morphological curvature scale spaces. *IEEE Trans. Image Processing*, 15(2):331–341, 2006.