# Computational exercises for the lecture "Data Mining in Bioinformatics"

Organizer: Peter Meinicke
Submission Date: 31.05.2025

## Sheet 2: Principal Component Analysis (PCA)

## Task 1: Eigenvector computation

Repeat the experiment from task 3 of sheet 1 with a different data generation.

a) Now generate the data according to:
`x_data_mat = np.random.rand(n_points, n_dims) @ a_trans_mat`
with `a_trans_mat = np.array([[0.25, -0.433], [1.299, 0.75]])`.
What do you observe compared to the previous experiment?

b) As an alternative to the previous procedures for variance analysis, now consider an iterative scheme with the following two steps:

$$\mathbf{w}_i = \mathbf{C}\mathbf{v}_{i-1} \quad (1)$$
$$\mathbf{v}_i = \mathbf{w}_i / \|\mathbf{w}_i\| \quad (2)$$

for $i = 1, \ldots, m$. Start the iteration with a randomly chosen direction vector $\mathbf{v}_0$. What do you observe for a suitable choice of $m$ (not too small) when you start the scheme with a different random initialization? Repeat the whole scheme 10 times, starting each time with a different random direction.
How can the convergence of the method be determined or "monitored"? Hint: for the scalar product (`np.dot`) of two vectors $\mathbf{a}$ and $\mathbf{b}$ we have the following equivalence:

$$\mathbf{a}^T\mathbf{b} = \|\mathbf{a}\|\mathbf{b}\| \cos\phi$$

where $\phi$ denotes the angle between the two vectors. Monitor the angle between successive direction vectors during iteration!

c) Show in writing that the sum of the eigenvalues of the (estimated) covariance matrix, which is referred to as the *total variance*, is equal to the average squared distance of data points from the mean vector $\mathbf{m}$

$$\frac{1}{n}\sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{m}\|^2$$

Hint: the sum of the eigenvalues corresponds to the sum of the diagonal elements of the covariance matrix! What does a single eigenvalue of the covariance matrix represent accordingly?

d) Without using the function `np.linalg.eig`, how could you successively compute several eigenvectors using the iterative procedure from b)? Use the following equivalence

$$\mathbf{C} = \sum_{i=1}^{d} \lambda_i \mathbf{u}_i \mathbf{u}_i^T.$$

to eliminate the proportion of variance in the already found direction for the search of the next direction. Use `np.outer` for the outer product between two vectors. Summarize the complete calculation of eigenvalues/vectors in a function `power_method(x_data_mat, n_evs)`, where `n_evs` specifies the number of "leading" eigenvalues/vectors desired by the user.

## Task 2: Hidden structures

Load the datasets `hidden1.csv` and `hidden2.csv` from the the data folder on StudIP, each of which has 10 dimensions (matrix columns). Use the function `np.genfromtxt` for this. In the following, use `pyplot` for the visualizations with `import matplotlib.pyplot as plt` as the previous call.

a) **Histograms:** Examine the distributions of each dimension using `plt.hist` and vary the number of histogram bars. Do you see any interesting feature? Which type of distribution do the histograms seem to resemble (in most cases)?

b) **Scatterplots:** Visualize two dimensions of each of the data points from `hidden1.csv` with `plt.plot`. Write a script that displays all possible combinations of two dimensions as separate plots in a graph. Tip: Integrate the different scatterplots using `plt.subplot`. Do these scatterplots provide more information about the overall distribution, are there any hints for correlations or structures that may exist in the data?

c) **Eigenvalue spectrum:** Visualize the eigenvalues of the covariance matrix with bar graphs (see function `plt.bar`). Make sure the eigenvalues are in descending order from left to right. How many relevant principal component directions among the eigenvectors of the covariance matrix does the picture suggest?

d) **Principal components:** Visualize the two principal components that represent the largest proportion of variance with a scatterplot. As a reminder, the principal components are the dimensions that can be obtained from the linear mapping

$$\mathbf{z} = \mathbf{U}^T\mathbf{x}$$

of the mean-centered data variables, where $\mathbf{U}$ contains the eigenvectors of the covariance matrix as columns. I.e., the $i$-th principal component corresponds to the scalar product $\mathbf{u}_i^T\mathbf{x}$. ☺

## Submission instructions

Please submit your solution by **31th May** using the corresponding homework folder in StudIP. Upload a **zip** compressed file containing all documents and data that you use in your solution. The file name should indicate the author, for example *Peter_Meinicke_Sheet2.zip*.
With your solution you should provide the Python source code together with the documentation within a **Jupyter Notebook** answering all the questions above. Use Markdown cells for your answers and explanations well separated from the Python code.

For the programming in Python/numpy do not use any code or functions from special bioinformatics libraries or toolboxes! Make sure that your code is readable and understandable for others, i.e. use descriptive names for the variable definitions according to the conventions on **Sheet 0**, do not use implicit variables (hard-coded values without a variable) and make use of comments!
For reasons explained in the lecture, we do not recommend the use of **A.I. tools** (such as ChatGPT) but, in principle, it is not prohibited. However, if parts of the solutions or code were generated with A.I. tools, it is **mandatory to specify** to what extent (percentage) and for what purpose it was used. Not indicating the actual use of A.I. tools can lead to disqualification from the exercises. If you are unsure about the specification(s) please ask us and do not make any assumptions!