Math 5490 – Homework 4: Due March 22 by 11:59pm

Instructions:

- Complete the problems below, and submit your solutions and Python code by uploading them to the Google form: https://forms.gle/k5HPD8UuPdSRbEtu8
- Submit all your Python code in a single .py file using the function templates given in each problem. I will import your functions from this file and test your code.
- If you use LaTeX to write up your solutions, upload them as a pdf file. Students who use LaTeX to write up their solutions will receive bonus points on the homework assignment (equivalent to 1/3 of a letter grade bump).
- If you choose to handwrite your solutions and scan them, please either use a real scanner, or use a smartphone app that allows scanning with you smartphone camera. It is not acceptable to submit photos of your solutions, as these can be hard to read.

Problems:

1. Consider the weighted PCA energy

$$E_{\mathbf{w}}(V;\mathbf{x}_1,\ldots,\mathbf{x}_m) = \sum_{i=1}^m w_i \,\mathrm{dist}(\mathbf{x}_i, V)^2,$$

where $\mathbf{w} = (w_1, w_2, \ldots, w_m)$ are nonnegative numbers (weights), and V is a linear space.

(i) Show that the weighted energy $E_{\mathbf{w}}$ is minimized over k-dimensional subspaces $V \subset \mathbb{R}^n$ by setting

$$V = \mathrm{span}\{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k\},$$

where $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n$ are the orthonormal eigenvectors of the weighted covariance matrix

$$M_{\mathbf{w}} = \sum_{i=1}^m w_i \mathbf{x}_i \mathbf{x}_i^T,$$

with corresponding eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$.

(ii) Show that the weighted covariance matrix can also be expressed as

$$M_{\mathbf{w}} = X^T W X,$$

where W is the $m \times m$ diagonal matrix with diagonal entries $w_1, w_2, \ldots, w_m$, and

$$X = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_m \end{bmatrix}^T.$$

(iii) Show that the optimal energy is given by

$$E_{\mathbf{w}}(V;\mathbf{x}_1,\ldots,\mathbf{x}_m) = \sum_{i=k+1}^n \lambda_i.$$

(iv) Suppose we minimize $E_{\mathbf{w}}$ over affine spaces $A = \mathbf{a} + V$, so

$$E_{\mathbf{w}}(A; \mathbf{x}_1, \ldots, \mathbf{x}_m) = \sum_{i=1}^{m} w_i \operatorname{dist}(\mathbf{x}_i, A)^2,$$

Show that an optimal choice for $\mathbf{a}$ is the weighted centroid

$$\mathbf{a} = \frac{\sum_{i=1}^{m} w_i \mathbf{x}_i}{\sum_{i=1}^{m} w_i}.$$

2. We consider here the 2-means clustering algorithm in dimension $n = 1$. Let $x_1, x_2, \ldots, x_m \in \mathbb{R}$ and recall the 2-means energy is

$$E(c_1, c_2) = \sum_{i=1}^{m} \min\left\{(x_i - c_1)^2, (x_i - c_2)^2\right\}.$$

Throughout the question we assume that the x_i are ordered so that

$$x_1 \le x_2 \le \cdots \le x_m.$$

For $1 \le j \le m-1$ we define

$$\mu^-(j) = \frac{1}{j}\sum_{i=1}^{j} x_i, \quad \mu^+(j) = \frac{1}{m-j}\sum_{i=j+1}^{m} x_i,$$

and

$$F(j) = \sum_{i=1}^{j}(x_i - \mu^-(j))^2 + \sum_{i=j+1}^{m}(x_i - \mu^+(j))^2.$$

(i) Explain how $F(j)$ differs from the 2-means energy $E(c_1, c_2)$, and why minimizing $F(j)$ over $j = 1, \ldots, m-1$ and setting $c_1 = \mu_-(j_*)$ and $c_2 = \mu^+(j_*)$ will give a solution at least as good as the 2-means algorithm (here, j_* is a minimizer of $F(j)$).

(ii) By (i) we can replace the 2-means problem with minimizing $F(j)$. We will now show how to do this efficiently. In this part, show that

$$F(j) = \sum_{i=1}^{m} x_i^2 - j\mu^-(j)^2 - (m-j)\mu^+(j)^2.$$

Thus, minimizing $F(j)$ is equivalent to maximizing

$$G(j) = j\mu^-(j)^2 + (m-j)\mu^+(j)^2.$$

(iii) Show that we can maximize G (i.e., find j_* with $G(j) \le G(j_*)$ for all j) in $O(m \log m)$ computations. Hint: First show that

$$\mu^-(j+1) = \frac{j}{j+1}\mu^-(j) + \frac{x_{j+1}}{j+1},$$

and

$$\mu^+(j+1) = \frac{m-j}{m-j-1}\mu^+(j) - \frac{x_{j+1}}{m-j-1}.$$

and explain how these formulas allow you to compute all the values $G(1), G(2), \ldots, G(m-1)$ recursively in $O(m \log m)$ operations, at which point the maximum is found by brute force.

(iv) [Challenge] Implement the method described in the previous three parts in Python. Test it out on some synthetic 1D data. For example, you can try a mixture of two Gaussians with different means. This part of the homework is optional.