Math 5467 – Homework 2

Instructions:

- Complete the problems below, and submit your solutions by uploading them to your shared Google form for HW2.
- 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.
- If you choose to write 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 images of your solutions, as these can be hard to read.

Problems:

1. Consider the weighted PCA energy

$$E_w(L) = \sum_{i=1}^{m} w_i \|x_i - \mathrm{Proj}_L x_i\|^2,$$

where $w_1, w_2, \ldots, w_m$ are nonnegative numbers (weights).

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

$$L = \mathrm{span}\{p_1, p_2, \ldots, p_k\},$$

where $p_1, p_2, \ldots, p_n$ are the orthonormal eigenvectors of the covariance matrix

$$M_w = \sum_{i=1}^{m} w_i x_i x_i^T,$$

with corresponding eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, given in decreasing order.

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

$$M_w = X^T W X,$$

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

$$X = \begin{bmatrix} x_1 & x_2 & \cdots & x_m \end{bmatrix}^T.$$

(iii) Show that the optimal energy is given by

$$E_w(L) = \sum_{i=k+1}^{n} \lambda_i.$$

(iv) Suppose we minimize E_w over affine spaces $x_0 + L$, so

$$E_w(x_0, L) = \sum_{i=1}^{m} w_i \|x_i - x_0 - \mathrm{Proj}_L(x_i - x_0)\|^2.$$

Show that an optimal choice for x_0 is the weighted centroid

$$x_0 = \frac{\sum_{i=1}^m w_i x_i}{\sum_{i=1}^m w_i}.$$

2. The k-means clustering algorithm is sensitive to outliers, since it uses the squared Euclidean distance. We consider the robust k-means energy

$$E_{robust}(c_1, c_2, \ldots, c_k) = \sum_{i=1}^m \min_{1 \le j \le k} \|x_i - c_j\|. \tag{1}$$

The robust k-means algorithm attempts to minimize (1). We start with some randomized initial values for the means $c_1^0, c_2^0, \ldots, c_k^0$, and iterate the steps below until convergence.

(a) Update the clusters

$$\Omega_j^t = \left\{ x_i \,:\, \|x_i - c_j^t\| = \min_{1 \le \ell \le k} \|x_i - c_\ell^t\| \right\}. \tag{2}$$

(b) Update the cluster centers

$$c_j^{t+1} \in \operatorname*{arg\,min}_{y \in \mathbb{R}^n} \sum_{x \in \Omega_j^t} \|x - y\|. \tag{3}$$

Complete the following exercises.

(i) Show that the Robust k-means algorithm descends on the energy E_{robust}.

(ii) The cluster center (3) does not admit a closed form expression and is sometimes inconvenient to work with in practice. Consider changing the Euclidean norm in (1) to the ℓ^1-norm $\|x\|_1 = \sum_{i=1}^n |x(i)|$, and define

$$E_{\ell^1}(c_1, c_2, \ldots, c_k) = \sum_{i=1}^m \min_{1 \le j \le k} \|x_i - c_j\|_1.$$

Formulate both steps of the k-means algorithm so that it descends on E_{ℓ^1}. Show that the cluster centers c_j^{t+1} are the coordinatewise medians of the points $x \in \Omega_j^t$, which are simple to compute.

(iii) Can you think of any reasons why the Euclidean norm would be preferred over the ℓ^1 norm in the k-means energy?

3. 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 $1 \leq j \leq m-1$ 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 $G(j) \leq G(j_*)$ 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.