CSE 422 Autumn 26 HW 4

Note: For all problems, you will be expected to explain your reasoning. Providing an expression without reasoning will not receive any partial credit.

Problem 1: Clustering Mixture Models with PCA

A *mixture model* is a very popular class of modeling assumption that captures collections of heterogeneous data points. An important special case of these mixture models is that of *Gaussian mixture models* (GMMs).

A mixture of k (spherical) Gaussians (a.k.a. a k-Gaussian mixture model, or k-GMM for short) is a distribution over $\mathbb{R}^d$ specified by k mean vectors $\mu_1, \ldots, \mu_k \in \mathbb{R}^d$, and k mixing weights $w_1, \ldots, w_k$ which satisfy that $w_i \geq 0$ for all $i = 1, \ldots, k$, and $\sum_{i=1}^{k} w_i = 1$. In other words, the mixing weights specify a probability distribution over k elements. To sample from a k-GMM, we perform the following 2-step procedure:

1. Choose j from $\{1, \ldots, k\}$ with probability w_i, and then
2. Sample a point $X \sim \mathcal{N}(\mu_j, I)$.

We call $\mathcal{N}(\mu_j, I)$ the *j-th component* of the mixture. For most of the problem, we will consider the $k = 2$ setting of this problem, so our distribution will be a mixture of two Gaussians. We'll also assume that the mixing weights are *uniform*, so $w_j = 1/k$ for all $j = 1, \ldots, k$, or for the case of $k = 2$, $w_1 = w_2 = 1/2$.

(a) [6 points] Consider a uniform mixture of two Gaussians D with means u and $-u$ satisfying $\|\mu\|_2 = 1$. What is the population covariance Σ of this data distribution? Explain how this suggests to use PCA to find a 1-dimensional space that contains u, given exact access to Σ.

(b) [8 points] When we are given sample access to D, we will never be able to get exact access to its covariance. So, assume we are given $\widehat{\Sigma} = \Sigma + E$, where E is a symmetric error matrix with *spectral norm* at most ε, that is,

$$\|Ev\|_2 \leq \varepsilon \text{ , for all } v \text{ with } \|v\|_2 = 1.$$

Show that PCA on $\widehat{\Sigma}$ recovers the projector Π onto a 1-dimensional subspace so that $\|\Pi u\|_2^2 \geq 1 - O(\varepsilon)$.

(c) [6 points] Explain informally how to use PCA to obtain a good clustering procedure, which, given a fresh sample from D, guesses which component it came from with good probability. What kind of guarantee can you hope to obtain?

Problem 2: Interpreting PCA for genetic data

The following four plots are extracted from PCA on real genetic data from the 1000 Genomes Project.

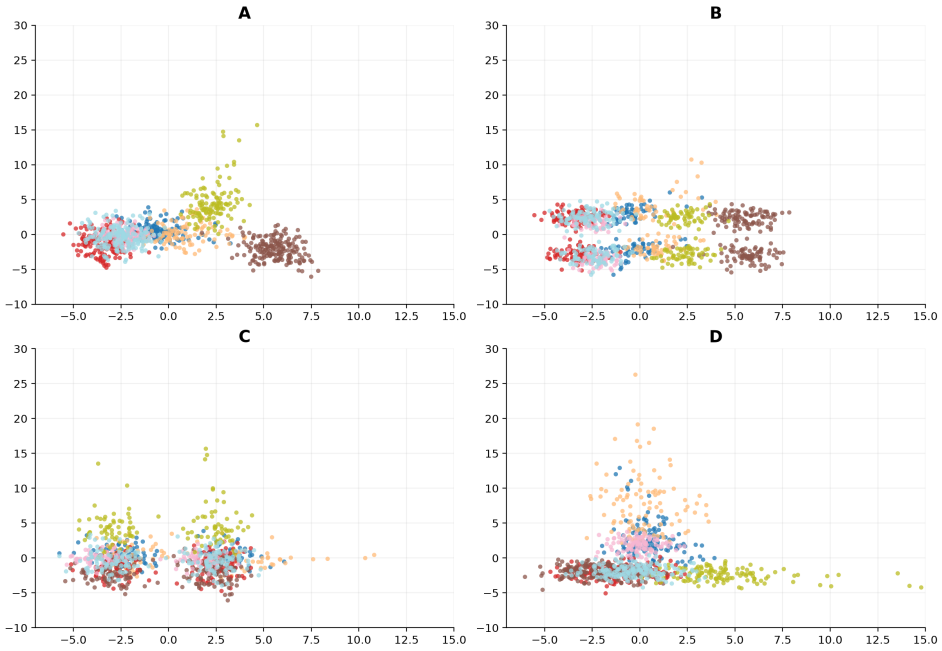


Figure 1: Genetic data plots

The plots were generated as follows. The dataset tracks d genes across a large set of individuals. Every individual is associated with a length-d vector X_i, where in the j-th coordinate of that vector, we put a 1 if they have a mutation in the j-th gene tracked in the dataset, and a 0 otherwise. We then centered the data and computed the first, second, third, and sixth principal components (all of which are unique). Call them v_1, v_2, v_3, and v_6. Each plot has x and y axes which is a projection of the data points onto one of these four principal components (PCs). That is, the points in each plot consist of one 2D point (x, y) for each original point X_i where $x = \langle v_\ell, X_i \rangle$ and $y = \langle v_k, X_i \rangle$, for some $\ell, k \in \{1, 2, 3, 6\}$.

The problem is: the professor who generated the plots was super lazy, and failed to tell you which PC is being used in the axes in each plot! And he's got a super bad memory too: all he remembers is that the *energy* of the points in each plot on the top row is strictly larger than the energy of the plot directly below it. Here, the energy of a collection of 2D points is the sum of the squared norms of the points:

$$\text{Energy}((x_1, y_1), \ldots, (x_n, y_n)) = \sum_{i=1}^{n} \left(x_i^2 + y_i^2\right) \ .$$

For each plot, your objective is to, using this information, recover which PC generated the x and y axis for the plot. Explain your reasoning for how you determined them. The colors of the data points correspond to the individual's nationality, but you won't need to know that information for this problem—it's just supposed to be visually useful for tracking stuff.

Problem 3: On de-randomizing power method

Power method, as described in class, is the following method. Suppose we are given a matrix A with eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_d$ satisfying $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_d|$. Let its top eigenvector be $x \in \mathbb{R}^d$, for $\|x\|_2 = 1$. Note that this eigenvector is unique up to sign. Then power method starts with a random unit vector $u_0 \in \mathbb{R}^d$, and iteratively computes

$$v_t = Av_{t-1} = A^t u_0 \ .$$

What we argued in class is that if we let $u_t = v_t/\|v_t\|_2$, then for t not too large, we should expect that $|\langle v_t, x\rangle| \approx 1$, that is, v_t is very close to x.

Crucially, this depends on initializing u randomly, and we argued that generically, this is necessary. In this homework, we will explore a restricted setting where there is a deterministic algorithm. Specifically, we will consider the setting where A is additionally assumed to be entrywise non-negative, that is, $A_{ij} \geq 0$ for all i, j.

(a) [10 points] First, argue that even without the non-negativity assumption, there exists a random family of entrywise non-negative unit vectors u so that if you initialize power method using u chosen randomly from this family, then with high probability, power method will still converge to the top eigenvector of A.

(b) [10 points] Show that this implies that this implies that if A satisfies the non-negativity assumption, then there is a deterministic initialization of power method that is guaranteed to converge to the top eigenvector x.