# STAT 542: Homework 5

Ahmadreza Eslaminia (ae15)

Please make sure that your solutions are readable and the file size is reasonable. Typing the answers is highly encouraged.

## Problem 1.

[2pts] Suppose that $X \in \mathbb{R}^p$ is a random variable taking values of $\mu_1$ and $\mu_2$ with equal probability.

- Find an expression of $\mathbb{E}\left[XX^\top\right]$ in terms of $\mu_1$ and $\mu_2$.
- Can a statistician estimate $\mu_1$ and $\mu_2$ (up to permutation) by solving the singular value decomposition of the empirical second moment matrix $\frac{1}{n}\sum_{i=1}^n X_i X_i^\top$ (for iid samples $X_1, \ldots, X_n$ following the distribution of $X$) ? Why?

Hint: see p7 of s13_method_of_moments.pdf

## Solution 1.

### part 1

The expected value $E[XX^T]$ is calculated as the average of the outer products of the vectors $\mu_1$ and $\mu_2$, weighted by their respective probabilities. Since the random variable $X$ takes on the values $\mu_1$ and $\mu_2$ with equal probability, we have:

$$\begin{aligned} E[XX^T] &= \frac{1}{2}(\mu_1\mu_1^T) + \frac{1}{2}(\mu_2\mu_2^T) \\ &= \frac{1}{2}\mu_1\mu_1^T + \frac{1}{2}\mu_2\mu_2^T. \end{aligned}$$

Therefore, the expected value of $XX^T$ is the sum of the outer products of $\mu_1$ and $\mu_2$ scaled by $\frac{1}{2}$.

### Part 2

The empirical second moment matrix $M$ is computed from the sample data as:

$$M = \frac{1}{n} \sum_{i=1}^{n} X_i X_i^T$$

Applying SVD to $M$, we decompose it into $M = U \Sigma V^T$, where $U$ and $V$ are orthogonal matrices, and $\Sigma$ is a diagonal matrix of singular values.

Given that $X$ takes on the values $\mu_1$ and $\mu_2$ with equal probability, the weight for each is $\frac{1}{2}$. This makes $M$ the sum of two rank-1 matrices, each weighted by $\frac{1}{2}$, reflecting the probabilities of $\mu_1$ and $\mu_2$. Since $\mu_1$ and $\mu_2$ are distinct, $M$ is approximately a rank-2 matrix under the assumption of linear independence.

The non-zero singular values in $\Sigma$ correspond to the variances in the directions of $\mu_1$ and $\mu_2$. The associated columns of $U$, $u_1$ and $u_2$, should point in the directions of $\mu_1$ and $\mu_2$ or their linear combinations.

Because the SVD is unique up to the signs and order of the singular vectors and values, the exact mapping between $u_1$, $u_2$ and $\mu_1$, $\mu_2$ cannot be determined; this results in the estimation being up to permutation.

Since the weights are $\frac{1}{2}$, the singular vectors $u_1$ and $u_2$ will not need additional scaling to estimate $\mu_1$ and $\mu_2$. Thus, $u_1$ and $u_2$ directly provide the estimates for $\mu_1$ and $\mu_2$, up to permutation.

In conclusion, by performing SVD on the empirical second moment matrix, a statistician can indeed estimate $\mu_1$ and $\mu_2$, taking into account the equal probabilities and assuming linear independence of $\mu_1$ and $\mu_2$. This approach effectively utilizes the method of moments to estimate the parameters of the underlying distribution.

## Problem 2.

Consider 8 data points on the unit circle of the form $(\cos\theta, \sin\theta)$ where

$$\theta = \frac{m\pi}{2} \pm \epsilon, \quad m = 1, 2, 3, 4 \tag{1}$$

Suppose that we want to solve $K$-means clustering with $K = 4$.

2pts Show that if $\epsilon > 0$ is sufficiently small, the (global) minimum within-point scatter is achieved by pairs of points on the circle with angle difference $2\epsilon$.

1pts Set $\epsilon = 0.01$, and run Lloyd's algorithm to solve 4-means clustering with random initialization. Run the algorithm 10 times with random initializations and report the minimum value of within-point scatter achieved in the simulation.

Hint: the $R$ code for kmeans and within-cluster sum of squares is described in p21 of s12_unsupervised.pdf.

1pts bonus For sufficiently small $\epsilon > 0$, describe a local minimum of Lloyd's algorithm which is not global optimal, and explain why it has such a property.

1pts bonus For sufficiently small $\epsilon > 0$, describe a local minimum of Lloyd's algorithm which is not a local minimum of Hartigan-Wong, and explain why it has such a property.

# Solution 2.

## Part 1

The within-cluster sum of squares (WCSS) for K-means aims to minimize the squared distances of all points in a cluster to their centroid. For two points on a unit circle separated by an angle $2\epsilon$, the minimum WCSS is achieved when the points are clustered together. We shall show that this configuration indeed provides the global minimum WCSS.

The position of the points on the unit circle can be described as:

$$A = (\cos(\frac{m\pi}{2} + \epsilon), \sin(\frac{m\pi}{2} + \epsilon))$$
$$A' = (\cos(\frac{m\pi}{2} - \epsilon), \sin(\frac{m\pi}{2} - \epsilon))$$

The centroid of each pair, for small $\epsilon$, can be approximated using trigonometric identities:

$$C = \left(\frac{\cos(\frac{m\pi}{2} + \epsilon) + \cos(\frac{m\pi}{2} - \epsilon)}{2}, \frac{\sin(\frac{m\pi}{2} + \epsilon) + \sin(\frac{m\pi}{2} - \epsilon)}{2}\right)$$

Simplifying with trigonometric formulas, we find that $C$ lies on the unit circle at an angle of $\frac{m\pi}{2}$.

The Euclidean distance from each point to the centroid is equal to the arc length for small $\epsilon$, which can be approximated by the straight line distance since the circle's curvature is negligible for small angles. This distance is $\sin(\epsilon)$, but for sufficiently small $\epsilon$, we can approximate $\sin(\epsilon) \approx \epsilon$.

Thus, the WCSS for each pair of points is:

$$\text{WCSS}_m = 2 \times (\epsilon)^2$$

Since $m$ takes on four values corresponding to four clusters, the total WCSS across all clusters is:

$$\text{Total WCSS} = 4 \times \text{WCSS}_m = 8 \times (\epsilon)^2$$

This is the global minimum WCSS because any other clustering of the points would result in at least one cluster centroid lying inside the unit circle, which would increase the distance from the points in that cluster to the centroid, resulting in a higher WCSS.

Therefore, for sufficiently small $\epsilon$, the minimum WCSS is indeed achieved by pairs of points on the circle with an angle difference of $2\epsilon$.

## Part 2

The following Python script performs K-means clustering and gives 0.0007999733336888611 as the minimum value of within-point scatter:

```
import numpy as np

def euclidean_distance(point1, point2):
    return np.sqrt(np.sum((point1 - point2) ** 2))

# Function to do K-means
def kmeans(data_points, k=4, num_iterations=100):
    # Randomly choose centroids
    centroids = data_points[np.random.choice(range(len(data_points)), k, replace=False)]

    for _ in range(num_iterations):
        # Assign clusters
        clusters = {}
        for x in data_points:
            distances = [euclidean_distance(x, centroid) for centroid in centroids]
            cluster_index = distances.index(min(distances))
            if cluster_index not in clusters:
                clusters[cluster_index] = []
            clusters[cluster_index].append(x)

        # Update centroids
        new_centroids = []
        for cluster_index in sorted(clusters):
            new_centroids.append(np.mean(clusters[cluster_index], axis=0))

        centroids = np.array(new_centroids)

    wcss = sum(
        sum(euclidean_distance(x, centroids[cluster_index])**2 for x in clusters[cluster_ind
        for cluster_index in clusters
    )

    return centroids, wcss

epsilon = 0.01

data_points = np.array([([np.cos(m * np.pi / 2 + epsilon), np.sin(m * np.pi / 2 + epsilon)],
                         [np.cos(m * np.pi / 2 - epsilon), np.sin(m * np.pi / 2 - epsilon)])
                        for m in range(1, 5)]).reshape(-1, 2)

best_wcss = np.inf
best_centroids = None
```

```
for _ in range(10):
    centroids, wcss = kmeans(data_points)
    if wcss < best_wcss:
        best_wcss = wcss
        best_centroids = centroids

print("Best WCSS:", best_wcss)
```

## Part 3

Let's assume we have a local minimum clustering configuration where points are clustered with their diametric opposites rather than with their nearest neighbors on the circle. For instance, cluster $C_1$ contains points $\theta_m = \frac{m\pi}{2} + \epsilon$ and $\theta_{m+1} = \frac{(m+1)\pi}{2} - \epsilon$. If $m = 1$, the centroid of $C_1$, denoted as $C_1^*$, is the average of these points:

$$\begin{aligned} C_1^* &= \left( \frac{\cos\left(\frac{\pi}{2} + \epsilon\right) + \cos\left(\pi - \epsilon\right)}{2}, \frac{\sin\left(\frac{\pi}{2} + \epsilon\right) + \sin\left(\pi - \epsilon\right)}{2} \right) \\ &= \left( \frac{-\sin(\epsilon) - \cos(\epsilon)}{2}, \frac{\cos(\epsilon) + \sin(\epsilon)}{2} \right). \end{aligned}$$

The within-cluster sum of squares (WCSS) for the local minimum configuration can be written as:

$$WCSS_{local} = \|A - C_1^*\|^2 + \|B - C_1^*\|^2,$$

where $A = \left(\cos\left(\frac{\pi}{2} + \epsilon\right), \sin\left(\frac{\pi}{2} + \epsilon\right)\right)$ and $B = \left(\cos\left(\pi - \epsilon\right), \sin\left(\pi - \epsilon\right)\right)$.

To show that this choice is suboptimal, we calculate the derivative of WCSS with respect to $\epsilon$. The derivative at $\epsilon = 0$ should be zero if the configuration is at a local minimum. The derivative is given by:

$$\begin{aligned} \frac{d(WCSS_{local})}{d\epsilon} &= 2\left(-\frac{1}{2}\cos(\epsilon) + \frac{1}{2}\sin(\epsilon)\right)\left(-\sin(\epsilon) - \cos(\epsilon)\right) \\ &\quad + 2\left(-\frac{1}{2}\sin(\epsilon) - \frac{1}{2}\cos(\epsilon)\right)\left(\cos(\epsilon) - \sin(\epsilon)\right). \end{aligned}$$

Evaluating this derivative at $\epsilon = 0$, we find:

$$\begin{aligned}\left.\frac{d(WCSS_{local})}{d\epsilon}\right|_{\epsilon=0} &= 2\left(-\frac{1}{2}+0\right)(0-1)+2\left(0-\frac{1}{2}\right)(1-0)\\ &= 2\left(-\frac{1}{2}\right)(-1)+2\left(-\frac{1}{2}\right)(1)\\ &= 1-1\\ &= 0.\end{aligned}$$

Thus, the derivative of WCSS at $\epsilon = 0$ is zero, indicating that we are at a stationary point. This supports the claim that the configuration associated with $C_1^*$ could be a local minimum. We know that this can not be the global minimum because the global is in part 1 which is less than the cost function in this scenario.

## Part 4

For the Hartigan-Wong algorithm, we aim to minimize the within-cluster sum of squares (WCSS), defined as follows:

$$W(C) = \sum_{i=1}^{K} \sum_{x \in C_i} \|x - \mu_i\|^2$$

In contrast to Lloyd's algorithm, which assigns each point to the closest centroid by minimizing the distance $\|x - \mu_\ell\|$, the Hartigan-Wong algorithm considers the net effect on the within-cluster sum of squares when moving a point from one cluster to another.

Let's consider the scenario where the data points are located on the unit circle with angles $\theta = \frac{m\pi}{2} \pm \epsilon$, where $m = 1, 2, 3, 4$, and $\epsilon$ is sufficiently small. We have eight points in total, and the aim is to cluster them into $K = 4$ clusters.

Suppose in the initial clustering, points with angles $\frac{\pi}{2} - \epsilon$ and $\frac{3\pi}{2} + \epsilon$ are placed in cluster $C_1$, with their centroid $\mu_1$ nearly at $(0, 0)$ due to the small $\epsilon$. Similarly, other clusters $C_2$, $C_3$, $C_4$ are formed with two points each, symmetrically placed around the circle.

Now, if we consider a point $x$ in cluster $C_2$ that is only marginally closer to the centroid $\mu_1$ than to its own centroid $\mu_2$, we might contemplate moving $x$ from $C_2$ to $C_1$. For Lloyd's algorithm, this move would be made as $x$ is closer to $\mu_1$ than $\mu_2$. However, for the Hartigan-Wong algorithm, we must consider the overall change in within-cluster sum of squares, $\Delta W$.

The change in within-cluster sum of squares for moving $x$ from $C_2$ to $C_1$ can be calculated as follows:

$$\Delta W = W(C_2 \setminus \{x\}) + W(C_1 \cup \{x\}) - W(C_2) - W(C_1)$$

If this reassignment would cause a substantial increase in the within-cluster sum of squares for $C_1$ because $x$ is not well aligned with the other point in $C_1$, then $\Delta W$ could be positive, even though $\|x-\mu_1\| < \|x-\mu_2\|$. This is particularly true if $x$ is an outlier in $C_1$ after the move, pulling the centroid $\mu_1$ away from the other point in $C_1$. Therefore, in the Hartigan-Wong algorithm, the point $x$ would remain in $C_2$, resulting in a local minimum that is not observed with Lloyd's algorithm.

This specific instance on the given dataset illustrates a case where the Hartigan-Wong algorithm finds a locally optimal clustering by balancing the overall variance, instead of simply assigning points to the nearest centroid as Lloyd's algorithm does.

# Problem 3.

Consider a Gaussian 2-mixture model $P_X = \frac{1}{2}\mathcal{N}(\mu_1, 1) + \frac{1}{2}\mathcal{N}(\mu_2, 1)$. Note that here the weights and the variances of each class are know, so the only unknown parameter is $\theta = (\mu_1, \mu_2) \in \Theta = \mathbb{R}^2$. Let $X_1, \ldots, X_n \in \mathbb{R}$ be observed samples, and $Z_1, \ldots, Z_n \in \{1, 2\}$ be the unobserved class labels.

1pts Give a precise expression of the distribution $p(Z_i = \cdot \mid X_i, \theta)$ using the logistic function, for any given $X_i$ and $\theta$.

1pts Give an explicit expression of $Q(\beta, \alpha) := \mathbb{E}_{p(Z^n|X^n,\alpha)}[\log p(Z^n, X^n \mid \beta)]$ in terms of $X^n$ and $\alpha, \beta \in \Theta$, where we defined $Z^n := (Z_1, \ldots, Z_n)$ and $X^n := (X_1, \ldots, X_n)$.

1pts bonus Give an explicit expression of $\arg\max_{\beta\in\Theta} Q(\beta, \alpha)$ in terms of $X^n$ and $\alpha$.

# Solution 3.

## part 1

Given the Gaussian mixture model and observed samples $X_1, ..., X_n$, we apply Bayes' theorem to find the posterior probability:

$$p(Z_i = z|X_i, \theta) = \frac{p(X_i|Z_i = z, \theta) \cdot p(Z_i = z)}{p(X_i|\theta)}$$

For our model, we have:

$$p(X_i|Z_i = z, \theta) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(X_i-\mu_z)^2}$$

Since the priors $p(Z_i = 1)$ and $p(Z_i = 2)$ are both $\frac{1}{2}$, the posterior probabilities simplify to:

$$p(Z_i = 1|X_i, \theta) = \frac{e^{-\frac{1}{2}(X_i-\mu_1)^2}}{e^{-\frac{1}{2}(X_i-\mu_1)^2} + e^{-\frac{1}{2}(X_i-\mu_2)^2}}$$

$$p(Z_i = 2|X_i, \theta) = \frac{e^{-\frac{1}{2}(X_i-\mu_2)^2}}{e^{-\frac{1}{2}(X_i-\mu_1)^2} + e^{-\frac{1}{2}(X_i-\mu_2)^2}}$$

Let $\sigma(x)$ denote the logistic function:

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Taking the log ratio of $p(Z_i = 1|X_i, \theta)$ to $p(Z_i = 2|X_i, \theta)$, we get:

$$\log\left(\frac{p(Z_i = 1|X_i, \theta)}{p(Z_i = 2|X_i, \theta)}\right) = (X_i(\mu_2 - \mu_1) - \frac{1}{2}(\mu_2^2 - \mu_1^2))$$

Setting $a = \mu_2 - \mu_1$ and $b = -\frac{1}{2}(\mu_2^2 - \mu_1^2)$, the posterior probability can be written using the logistic function as:

$$p(Z_i = 1|X_i, \theta) = \sigma(aX_i + b)$$

$$p(Z_i = 2|X_i, \theta) = 1 - \sigma(aX_i + b)$$

To write this in a compact form for both classes $z \in \{1, 2\}$:

$$p(Z_i = z|X_i, \theta) = \sigma((2z - 3)(aX_i + b))$$

## part 2

The complete-data log-likelihood for a single observation $(X_i, Z_i)$ is given by:

$$\log p(X_i, Z_i|\beta) = Z_i \log\left(\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}(X_i-\mu_1)^2}\right) + (1 - Z_i)\log\left(\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}(X_i-\mu_2)^2}\right)$$

where $Z_i$ is a binary latent variable indicating the component from which $X_i$ is drawn.

Given the Gaussian mixture model $P_X$, we define $Q(\beta, \alpha)$ as the sum over all data points of the expected complete-data log-likelihood:

$$Q(\beta, \alpha) = \sum_{i=1}^{n} [\tau_i \log p(X_i, Z_i = 1|\beta) + (1 - \tau_i)\log p(X_i, Z_i = 0|\beta)]$$

where $\tau_i = p(Z_i = 1|X_i, \alpha)$ represents the posterior probability (responsibility) that the $i$-th observation $X_i$ was generated from the first component of the mixture model.

By plugging in the complete-data log-likelihood we have:

$$
\begin{aligned}
Q(\beta, \alpha) &= \sum_{i=1}^{n} \left[ \tau_i \log \left( \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(X_i - \mu_1)^2} \right) + (1 - \tau_i) \log \left( \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(X_i - \mu_2)^2} \right) \right] \\
&= \sum_{i=1}^{n} \left[ \tau_i \left( \log \left( \frac{1}{\sqrt{2\pi}} \right) - \frac{1}{2}(X_i - \mu_1)^2 \right) + (1 - \tau_i) \left( \log \left( \frac{1}{\sqrt{2\pi}} \right) - \frac{1}{2}(X_i - \mu_2)^2 \right) \right]
\end{aligned}
$$

## Part 3

We aim to maximize the $Q(\beta, \alpha)$ function with respect to the parameters $\beta$. Since the term $\log \left( \frac{1}{\sqrt{2\pi}} \right)$ is constant with respect to $\mu_1$ and $\mu_2$, it does not affect the maximization with respect to these parameters and can be omitted. Thus, we simplify the expression for $Q(\beta, \alpha)$:

$$
Q(\beta, \alpha) = -\frac{1}{2} \sum_{i=1}^{n} \left[ \tau_i (X_i - \mu_1)^2 + (1 - \tau_i)(X_i - \mu_2)^2 \right]
$$

where $\tau_i$ represents the posterior probability for the $i$-th data point and $X_i$ are the observed samples.

Optimization for $\mu_1$

Taking the derivative of $Q$ with respect to $\mu_1$ and setting it to zero for optimization yields:

$$
\frac{\partial Q}{\partial \mu_1} = \sum_{i=1}^{n} \tau_i (X_i - \mu_1) = 0
$$

Solving for $\mu_1$, we obtain:

$$
\mu_1 = \frac{\sum_{i=1}^{n} \tau_i X_i}{\sum_{i=1}^{n} \tau_i}
$$

Optimization for $\mu_2$

Similarly, for $\mu_2$, we take the derivative of $Q$ with respect to $\mu_2$, set it to zero, and solve:

$$
\frac{\partial Q}{\partial \mu_2} = \sum_{i=1}^{n} (1 - \tau_i)(X_i - \mu_2) = 0
$$

This gives us:

$$
\mu_2 = \frac{\sum_{i=1}^{n} (1 - \tau_i) X_i}{\sum_{i=1}^{n} (1 - \tau_i)}
$$

By solving these optimization problems, we find the values of $\mu_1$ and $\mu_2$ that make the function $Q(\beta, \alpha)$ reach its maximum, given the current estimate $\alpha$.