Lecture 14: Unsupervised Learning

Introduction to Machine Learning [25737]

Sajjad Amini

Sharif University of Technology
Contents

1 Approach Definition

2 Principle Component Analysis
   - Interpretation Via Maximum Projection Spread
   - Interpretation Via Reconstruction

3 Clustering

4 Mixture Models
The material in the slides except cited are inspired from the following reference:

Section 1

Approach Definition
Unsupervised Learning

Principle Component Analysis

- Experience $E$: Set of $N$ samples $\mathcal{D} = \{x_n\}_{n=1}^{N}$
- Task $T$: Projecting data into low dimensional subspace which captures its main aspects
- Performance measure: Preserving data variations

Clustering

- Experience $E$: Set of $N$ samples $\mathcal{D} = \{x_n\}_{n=1}^{N}$
- Task $T$: Partition the input into regions that contains similar points.
- Performance measure in Compression: Compression loss
Section 2

Principle Component Analysis
Subsection 1

Interpretation Via Maximum Projection Spread
Assume $\mathbf{x} \in \mathbb{R}^D$ is a random variable and you have observed $N$ copies of it as $\{\mathbf{x}_i\}_{i=1}^N$ (Equivalently the dataset).

As before, we stack these copies into a Matrix $\mathbf{X}$ as:

$$
\mathbf{X} = \begin{bmatrix}
\mathbf{x}_1^T \\
\mathbf{x}_2^T \\
\vdots \\
\mathbf{x}_N^T
\end{bmatrix} = 
\begin{bmatrix}
x_1^1 & \ldots & x_1^D \\
x_2^1 & \ldots & x_2^D \\
\vdots & \ddots & \vdots \\
x_N^1 & \ldots & x_N^D
\end{bmatrix} 
\in \mathbb{R}^{N \times D}
$$

- Each column is a feature (covariate or predictor)
- Each row is an observation
Characterizing Dataset

The dataset point create a point cloud in $\mathbb{R}^D$ space.

- The expectation of this point cloud, calculated below, determines the center of point cloud.

$$
\mathbb{E}[\mathbf{x}] = \begin{bmatrix}
\mathbb{E}[x^1] \\
\vdots \\
\mathbb{E}[x^D]
\end{bmatrix}
$$

- The covariance matrix of this point cloud, calculated below, determines the spread of point cloud.

$$
\text{Cov}[\mathbf{x}] \triangleq \mathbb{E}[(\mathbf{x} - \mathbb{E}[\mathbf{x}])(\mathbf{x} - \mathbb{E}[\mathbf{x}])^T] = \mathbb{E}[\mathbf{x}\mathbf{x}^T] - \mathbb{E}[\mathbf{x}]\mathbb{E}[\mathbf{x}]^T = \Sigma
$$

$$
= \begin{bmatrix}
\text{Cov}[X_1, X_1] & \text{Cov}[X_1, X_2] & \cdots & \text{Cov}[X_1, X_D] \\
\text{Cov}[X_2, X_1] & \text{Cov}[X_2, X_2] & \cdots & \text{Cov}[X_2, X_D] \\
\vdots & \vdots & \ddots & \vdots \\
\text{Cov}[X_D, X_1] & \text{Cov}[X_D, X_2] & \cdots & \text{Cov}[X_D, X_D]
\end{bmatrix}
$$
Utilizing Empirical Distribution

The empirical distribution for dataset \( \{ \mathbf{x}_i \}_{i=1}^{N} \) is defined as:

\[
p_D(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^{N} \delta(\mathbf{x} - \mathbf{x}_n)
\]

We can use it to compute the empirical (sample) mean and empirical (sample) covariance matrix as:

\[
\mathbb{E}_D[\mathbf{x}] = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i = \bar{\mathbf{x}}
\]

\[
\text{Cov}_D[\mathbf{x}] = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i\mathbf{x}_i^T - \bar{\mathbf{x}}\bar{\mathbf{x}}^T = \mathbf{S}
\]
Characterizing Dataset [1]

Eliminating Summation Using Linear Algebra

For sample mean, we have:

\[
\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i = \frac{1}{N} \begin{bmatrix} x_1 & \cdots & x_N \end{bmatrix} \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}_{D \times N} = \frac{1}{N} X^T 1
\]

For sample covariance matrix, we use forth method for matrix multiplication as:

\[
S = \frac{1}{N} \sum_{i=1}^{N} x_ix_i^T - \bar{x}\bar{x}^T = \frac{1}{N} \begin{bmatrix} x_1 & \cdots & x_N \end{bmatrix} \begin{bmatrix} -x_1^T \\ \vdots \\ -x_N^T \end{bmatrix} - \bar{x}\bar{x}^T
\]

\[
= \frac{1}{N} X^T X - \frac{1}{N^2} X^T 11^T X = \frac{1}{N} X^T \left( I - \frac{1}{N} 11^T \right) X
\]
Idempotent Matrix

Matrix $P$ is said to be idempotent matrix if $P^2 = P$

Projection Matrix

Matrix $Z$ is said to be projection matrix if it is symmetric and idempotent.

Working on $H$ Matrix

Matrix $H$ is a projection matrix because:

1. $H^T = (I - \frac{1}{N}11^T)^T = I - \frac{1}{N}11^T$
2. Idempotent property:

$$H^2 = HH = (I - \frac{1}{N}11^T)(I - \frac{1}{N}11^T) = I - \frac{2}{N}11^T + \frac{1}{N^2}11^T11^T$$

$$= I - \frac{2}{N}11^T + \frac{1}{N}11^T = H$$
Characterizing the Projection $\mathbf{H}$

Assume $\mathbf{v} \in \mathbb{R}^N$, then:

$$\mathbf{Hv} = \mathbf{v} - \frac{1}{N} \mathbf{1} \mathbf{1}^T \mathbf{v} = \mathbf{v} - \frac{1^T \mathbf{v}}{N} \mathbf{1} = \mathbf{v} - \bar{\mathbf{v}}$$

Thus $\mathbf{H}$ removes the mean of the vector from each coordinate. Equivalently $\mathbf{Hv} = 0$

Thus $\mathbf{H}$ is the projection onto the subspace of vectors with zero mean (Projection onto hyperplane which is orthogonal to $\mathbf{1}$ vector).

Re-writing $\mathbf{S}$

Based on the projection matrix $\mathbf{H}$, we have:

$$\mathbf{S} = \frac{1}{N} \mathbf{X}^T \mathbf{H} \mathbf{X} = \frac{1}{N} \mathbf{X}^T \mathbf{H}^2 \mathbf{X} = \frac{1}{N} \mathbf{X}^T \mathbf{H}^T \mathbf{H} \mathbf{X} = \frac{1}{N} (\mathbf{H} \mathbf{X})^T (\mathbf{H} \mathbf{X})$$

where $\mathbf{H} \mathbf{X}$ result in centered features.
Linear Combination of Features [1]

Original Formulation

Assume an arbitrary direction of \( u \in \mathbb{R}^D \), then consider the following value:

\[
\begin{align*}
\mathbf{u}^T\mathbf{\Sigma}\mathbf{u} &= \mathbf{u}^T \left[ \mathbb{E}[\mathbf{x}\mathbf{x}^T] - \mathbb{E}[\mathbf{x}]\mathbb{E}[\mathbf{x}^T] \right] \mathbf{u} \\
&\overset{(a)}{=} \mathbb{E}\left[\left( \mathbf{u}^T \mathbf{x} \right)\left( \mathbf{u}^T \mathbf{x} \right)^T \right] - \mathbb{E}\left[\mathbf{u}^T \mathbf{x} \right]\mathbb{E}\left[\left( \mathbf{u}^T \mathbf{x} \right)^T \right] \\
&= \mathbb{E}\left[\left( \mathbf{u}^T \mathbf{x} \right)^2 \right] - \mathbb{E}\left[\mathbf{u}^T \mathbf{x} \right]^2 = \text{var}(\mathbf{u}^T \mathbf{x})
\end{align*}
\]

Switching to Empirical Distribution

Using empirical distribution, we have the empirical variance for \( \{\mathbf{u}^T \mathbf{x}_i\}_{i=1}^N \):

\[
\begin{align*}
\mathbf{u}^T\mathbf{S}\mathbf{u} &= \mathbf{u}^T \left[ \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i^T \mathbf{x}_i - \overline{\mathbf{x}}\overline{\mathbf{x}}^T \right] \mathbf{u} = \frac{1}{N} \sum_{i=1}^N \left( \mathbf{u}^T \mathbf{x}_i \right)\left( \mathbf{x}_i^T \mathbf{u} \right) - \left( \mathbf{u}^T \overline{\mathbf{x}} \right)\left( \overline{\mathbf{x}}^T \mathbf{u} \right) \\
&= \frac{1}{N} \sum_{i=1}^N \left( \mathbf{u}^T \mathbf{x}_i \right)^2 - \left( \mathbf{u}^T \overline{\mathbf{x}} \right)^2 = \frac{1}{N} \sum_{i=1}^N \left( \mathbf{u}^T \mathbf{x}_i \right)^2 - \left( \overline{\mathbf{u}^T \mathbf{x}} \right)^2 = s^2
\end{align*}
\]
Intuition Behind Principle Component Analysis [1]

**Intuition**
Finding the direction \( \mathbf{u} \) which result in the high projection value spread measured by project value variance

**Extreme Cases**
- Zero variance: The projection of all points onto \( \mathbf{u} \) is equal (The points are in the hyper-plane whose normal vector is \( \mathbf{u} \)).
- Large variance: The points are spread along the \( \mathbf{u} \) direction.

**Objective**
Fining the direction that maximize the projection variance (or equivalently projection spread)
Formulation
The problem for PCA can be formulated as:

$$\max_{u \in \mathbb{R}^D} u^T Su$$

The maximum value of objective function is infinity, thus we need to constrained $u$ as:

$$\max_{u \in \mathbb{R}^D} u^T Su \text{ subject to } \|u\|_2 = 1$$
Spectral Theorem

Eigenvalues and Eigenvectors of Symmetric Matrices

Based on *Spectral Theorem*, for symmetric matrix $S$ we have:

- All eigenvalues are real
- Eigenvectors are orthonormal ($U$ is orthogonal thus $P^{-1} = P^T$)

Then we have:

$$S = P \Lambda P^T = \begin{bmatrix} p_1 & p_2 & \cdots & p_n \end{bmatrix} \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix} \begin{bmatrix} -p_1^T \\ -p_2^T \\ \vdots \\ -p_n^T \end{bmatrix}$$

$$= \sum_{i=1}^{n} \lambda_i p_i p_i^T$$

Covariance Matrices

Covariance matrices are positive semi-definite, equivalently, all their eigenvalues are non-negative ($u^T \Sigma u \geq 0, \forall u$ and $u^T Su \geq 0, \forall u$).
Characterizing $S$ using Spectral theorem, we can write $S$ as:

$$S = P \Lambda P^T,$$

where

$$P = \begin{bmatrix} p_1 & p_2 & \cdots & p_D \end{bmatrix},$$

$$\Lambda = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_D \end{bmatrix},$$

and

$$P^T P = I,$$

with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_D \geq 0$. 

(Sajjad Amini, IML-S14, Principle Component Analysis, 18/45)
Transforming Using Eigenvectors

Assume we define \( y = P^T x \in \mathbb{R}^D \) and \( \bar{x} = 0 \), then:

\[
\bar{y} = P^T x = P^T \bar{x} = 0
\]

Thus the sample covariance matrix for \( y \) is:

\[
S^y = \frac{1}{N} \sum_{i=1}^{N} (P^T x_i)(P^T x_i)^T = P^T \left( \frac{1}{N} \sum_{i=1}^{N} x_i x_i^T \right) P = P^T S P = D
\]

Thus we take one step through whitening:

\[
\text{cov}(Y^i, Y^j) = \begin{cases} 
0 & i \neq j \\
\lambda_i & i = j
\end{cases}
\]
Finding Maximum Spread Direction [1]

Assume the maximum spread direction is $u$ and consider the following definition:

$$b = P^T u \Rightarrow u = P b$$

Now we measure the spread as:

$$u^T S u = (P b)^T (P D P^T) (P u) = b^T (P P^T) D (P^T P) b = \sum_{j=1}^{D} \lambda_j b_j^2 \leq \lambda_1 \sum_{j=1}^{D} b_j^2$$

On the other hand, for $\|b\|^2$, we have:

$$\|b\|^2 = \|P^T u\|^2 = (P^T u)^T (P^T u) = u^T (P P^T) u = \|u\|^2 = 1$$

Thus:

$$\forall u \in \mathbb{R}^D : u^T S u \leq \lambda_1$$
Finding Maximum Spread Direction

We see:

\[ \forall u \in \mathbb{R}^D : u^T Su \leq \lambda_1 \]

Now check the variance for \( u = p_1 \):

\[
b = \begin{bmatrix}
- & p_1^T & - \\
- & p_2^T & - \\
\vdots & & \\
- & p_m^T & - \\
\end{bmatrix}
\]

\[
p_1 = \begin{bmatrix}
p_1^T p_1 \\
p_2^T p_1 \\
\vdots \\
p_m^T p_1 \\
\end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}
\]

Thus:

\[
p_1^T S p_1 = b^T D b = \sum_{j=1}^{D} \lambda_j b_j = \lambda_1
\]

And \( u = p_1 \) is the direction of maximum spread.
Assume $\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_D$ to be the eigenvectors of $\mathbf{S}$ matrix corresponding to eigenvalues sorted in the descending order. Then, we have seen:

$$\mathbf{p}_1 \in \arg\max_{\|\mathbf{u}\|=1} \mathbf{u}^T \mathbf{S} \mathbf{u}$$

We can show the following in an almost similar way:

$$\mathbf{p}_2 \in \arg\max_{\|\mathbf{u}\|=1, \mathbf{u} \perp \mathbf{p}_1} \mathbf{u}^T \mathbf{S} \mathbf{u}$$

$$\mathbf{p}_3 \in \arg\max_{\|\mathbf{u}\|=1, \mathbf{u} \perp \mathbf{p}_i, i=1,2} \mathbf{u}^T \mathbf{S} \mathbf{u}$$

$$\vdots$$

$$\mathbf{p}_j \in \arg\max_{\|\mathbf{u}\|=1, \mathbf{u} \perp \mathbf{p}_k, k=1,\ldots,(j-1)} \mathbf{u}^T \mathbf{S} \mathbf{u}$$
Subsection 2

Interpretation Via Reconstruction
PCA Interpretation Using Reconstruction

Assume we have a high-dimensional data $\mathbf{x} \in \mathbb{R}^D$ and we want to project it to a low dimensional subspace $\mathbf{z} \in \mathbb{R}^L$ such that low dimensional representation is a good representation. To approach a mathematical formulation, we need:

- A projection (encoding) operator: $\mathbf{z} = \text{Encode}(\mathbf{x}; \theta)$
- An un-projection (decoding) operator: $\hat{\mathbf{x}} = \text{Decode}(\mathbf{z}; \theta)$
- A goodness measure: $\|\mathbf{x} - \hat{\mathbf{x}}\|^2$
Parameters

- Representation in the low dimensional space $z \in \mathbb{R}^L$
- Basis functions for reconstruction $\hat{x} = \sum_{i=1}^{L} z_i w_i$ such that:

$$w_i^T w_j = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

Or equivalently if $W = [w_1, w_2, \ldots, w_L] \in \mathbb{R}^{D \times L}$ then:

$$W^T W = \begin{bmatrix} w_1^T \\ w_2^T \\ \vdots \\ w_L^T \end{bmatrix} \begin{bmatrix} w_1 & w_2 & \ldots & w_L \end{bmatrix} = \begin{bmatrix} w_1^T w_1 & w_1^T w_2 & \ldots & w_1^T w_L \\ w_2^T w_1 & w_2^T w_2 & \ldots & w_2^T w_L \\ \vdots & \vdots & \ddots & \vdots \\ w_L^T w_1 & w_L^T w_2 & \ldots & w_L^T w_L \end{bmatrix} = I$$
PCA Interpretation Using Reconstruction

You are given a dataset \( \{x_i\}_{i=1}^N \) in \( \mathbb{R}^D \). You should design \( W \in \mathbb{R}^{D \times L} \) and \( \{z_i\}_{i=1}^N \) using the following problem:

\[
\min_{W, \{z_k\}} \frac{1}{N} \sum_{i=1}^N \|x_i - Wz_i\|_2^2
\]
Basic Problem \( L = 1 \)

**Simplifying the Loss**

In this case, the loss function is:

\[
\mathcal{L}(\mathbf{w}_1, \{z_k^1\}) = \frac{1}{N} \sum_{i=1}^{N} \| \mathbf{x}_i - z_i^1 \mathbf{w}_1 \|^2 = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{x}_i - z_i^1 \mathbf{w}_1)^T (\mathbf{x}_i - z_i^1 \mathbf{w}_1)
\]

\[
= \frac{1}{N} \sum_{i=1}^{N} \left[ \mathbf{x}_i^T \mathbf{x}_i - 2z_i^1 \mathbf{w}_1^T \mathbf{x}_i + (z_i^1)^2 \mathbf{w}_1^T \mathbf{w}_1 \right] = 1
\]

\[
= \frac{1}{N} \sum_{i=1}^{N} \left[ \mathbf{x}_i^T \mathbf{x}_i - 2z_i^1 \mathbf{w}_1^T \mathbf{x}_i + (z_i^1)^2 \right]
\]
Basic Problem \( L = 1 \)

**Derivative w.r.t. Representation**

\[
\frac{\partial L(w_1, \{z_k^1\})}{\partial z_n^1} = \frac{1}{N} \left[ -2w_1^T x_n + 2z_n^1 \right] = 0 \Rightarrow z_n^1 = w_1^T x_n
\]

**Updating Loss Function**

\[
L(w_1) = \frac{1}{N} \sum_{i=1}^{N} [x_n^T x_n - (z_i^1)^2] = \text{const} - \frac{1}{N} \sum_{i=1}^{N} (z_i^1)^2
\]

Dropping the constant term, we have:

\[
L(w_1) = -\frac{1}{N} \sum_{i=1}^{N} (z_i^1)^2 = -\frac{1}{N} \sum_{i=1}^{N} w_1^T x_i x_i^T w_1 = -w_1^T S w_1
\]

Note that in the above, we assumed the empirical mean vector to be zero \( \bar{x} = 0 \)
Basic Problem $L = 1$

Solving for $w_1$

We have the following optimization problem:

$$\min_{w_1} \ w_1^T Sw_1 \ \text{subject to} \ w_1^T w_1 = 1$$

Thus we form the Lagrangian as:

$$\tilde{L}(w_1) = w_1^T Sw_1 - \lambda_1 (w_1^T w_1 - 1)$$

The partial derivative for the Lagrangian is:

$$\frac{\partial}{\partial w_1} \tilde{L}(w_1) = 2Sw_1 - 2\lambda_1 w_1 = 0 \Rightarrow Sw_1 = \lambda_1 w_1$$

Thus $(\lambda_1, w_1)$ is a pair of (eigenvalue,eigenvector). But which of them?

$$w_1^T Sw_1 = w_1^T w_1 = \lambda_1$$

Thus $w_1$ is the direction of eigenvector corresponding to largest eigenvalue.
Assume we want to find $W = [w_1, \ldots, w_L]$ and $z = [z^1, \ldots, z^L]$. Then we have the following problem:

$$
\mathcal{L}(W, \{z_k\}) = \frac{1}{N} \sum_{i=1}^{N} \|x_i - \sum_{j=1}^{L} z^j_i w_j\|^2
$$

And the solution is:

$$
w_i = p_i, \quad i = 1, \ldots, L
$$

$$
z^j_i = p^T_j x_i, \quad \begin{cases} 
 i = 1, \ldots, N \\
 j = 1, \ldots, L
\end{cases}
$$

where $\{p_i\}$ is the set of eigenvector for $S$ matrix corresponding to eigenvalues sorted in descending order.
Encoding and Decoding

**Encoding**

\[ \mathbb{R}^L \ni z = \text{Encode}(x, W) = W^T x = \begin{bmatrix} w_1^T \\ w_2^T \\ \vdots \\ w_L^T \end{bmatrix} \begin{bmatrix} x \\ w_2^T x \\ \vdots \\ w_L^T x \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_L \end{bmatrix} \]

**Decoding**

\[ \mathbb{R}^D \ni \hat{x} = \text{Decode}(x, W) = W z = \begin{bmatrix} w_1 & w_2 & \ldots & w_L \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_L \end{bmatrix} = \sum_i z^i w_i \]
Section 3

Clustering
Clustering Problem

Clustering

- Experience $E$: Set of $N$ samples $\mathcal{D} = \{\mathbf{x}_n\}_{n=1}^N$
- Task $T$: Partition the input into regions that contains similar points.
- Performance measure in Compression: Compression loss

Figure: Sample GMM distribution
Section 4

Mixture Models
Mixture Models

One way to create more complex probability models is to take a convex combination of simple distributions. This is called a mixture model. This has the form $p(y|\theta) = \sum_{k=1}^{K} \pi_k p_c(y|\theta_k)$ where:

- $p_c(\cdot|\theta_k)$ is the $k$-th mixture component
- $\{\pi_k\}_{k=1}^{K}$ are mixture weights with the following constraints:
  - $0 \leq \pi_k \leq 1, k = 1, \ldots, K$
  - $\sum_{k=1}^{K} \pi_k = 1$

Mixture Models - Generative Story

Suppose latent variable $z$ to be a categorical RV and distributed as $p(z|\theta) = \text{Cat}(z|\pi)$ and conditional $p(y|z = k, \theta) = p_c(y|\theta_k)$. We can interpret mixture models as follows:

- We sample a specific component.
- We generate $y$ using sampled value of $z$.

Using the above procedure, we have:

$$p(y|\theta) = \sum_{k=1}^{K} p(z = k|\theta)p(y|z = k, \theta) = \sum_{k=1}^{K} \pi_k p(y|\theta_k)$$
Gaussian Mixture Model

Gaussian Mixture Model (GMM) or Mixture of Gaussian (MoG) is defined as:

\[ p(y|\theta) = \sum_{k=1}^{K} \pi_k N(y|\mu_k, \Sigma_k) \]

**Figure:** Sample GMM distribution
Problem Formulation

- Observed data samples \( \{x_i\}_{i=1}^{n} \)
- Unobserved mixture element corresponding to each data sample \( \{z_i\}_{i=1}^{N} \)

Using the above two formulation, the complete dataset likelihood is:

\[
p(\mathcal{D}|\theta) = p(\{x_i\}, \{z_i\}|\theta)
\]

The marginal likelihood of dataset is:

\[
p(\{x_i\}|\theta) = \sum_{\{z_i\}} p(\{x_i\}, \{z_i\}|\theta)
\]

and the maximum likelihood estimation for \( \theta = \{\theta_1, \ldots, \theta_K, \pi\} \) can be calculated as:

\[
\hat{\theta}_{mle} = \arg\max_{\theta} p(\{x_i\}|\theta)
\]
Challenge

As the scale of the problem increases (dimension of $x$ and number of dataset sample $N$), it becomes computationally intractable to exactly evaluate (or even optimize) the marginal likelihood.

Solution

One solution is to use expectation maximization algorithm as:

- Initialize $\theta$ randomly (or by using problem-specific heuristics) as $\theta^{(0)}$
- For $t = 1, 2, \ldots, T$, repeat:
  
  - **E-step**: Compute posterior distribution of $\{z_i\}$ given $\{x_i\}$ and $\theta^{(t-1)}$ as:
    
    $$q^{(t)}(\{z_i\}) = p(\{z_i\}|\{x_i\}, \theta^{(t-1)})$$
  
  - **M-step**: Find $\theta^{(t)}$ as the maximizer of complete log-likelihood with respect to $q^{(t)}(\{z_i\})$ as:

$$\theta^{(t)} = \arg\max_{\theta} \mathbb{E}_{q^{(t)}} [\log p(\{x_i\}, \{z_i\}|\theta)] = \arg\max_{\theta} \sum_{\{z_i\}} q^{(t)}(\{z_i\}) \log p(\{x_i\}, \{z_i\}|\theta)$$
General Mixture Model

For a general mixture model, the samples are generated using the following distribution:

\[ p(x|\theta) = \sum_{k=1}^{K} \pi_k p_c(x|\theta_k) \]

where we have:

\[ \theta = \left\{ \pi = \begin{bmatrix} \pi_1 \\ \vdots \\ \pi_K \end{bmatrix}, \theta_1, \ldots, \theta_K \right\} \]

and \( z \sim \text{Cat}(\pi) \)
General Mixture Model

Complete log-Likelihood Formulation

\[
\log p(\{x_i\}, \{z_i\} | \theta) = \log \prod_{i=1}^{N} p(x_i, z_i | \theta) = \log \prod_{i=1}^{N} p(x_i | z_i, \theta)p(z_i | \theta)
\]

On the other hand, we have:

\[
p(x_i | z_i, \theta) = p_c(x_i | \theta z_i)
\]

\[
p(z_i | \theta) = \pi_{z_i}
\]

Thus we have:

\[
\log p(\{x_i\}, \{z_i\} | \theta) = \sum_{i=1}^{N} \left( \log \pi_{z_i} + \log p_c(x_i | \theta z_i) \right)
\]

\[
= \sum_{i=1}^{N} \sum_{k=1}^{K} \delta_{k, z_i} \left( \log \pi_k + \log p_c(x_i | \theta_k) \right)
\]
E-step

\[
p(\{z_i\}|\{x_i\}, \theta) = \prod_{i=1}^{N} p(z_i| x_i, \theta)
\]

To compute \(p(z_i | x_i, \theta)\), we use Bayes rule as:

\[
p(z_i = k | x_i, \theta) = \frac{p(x_i | z_i = k, \theta)p(z_i = k | \theta)}{\sum_{l=1}^{K} p(x_i | z_i = l, \theta)p(z_i = l | \theta)} = \frac{\pi_k p_c(x_i | \theta_k)}{\sum_{l=1}^{K} \pi_l p_c(x_i | \theta_l)}
\]

Thus we have:

\[
q^{(t)}(\{z_i\}) \prod_{i=1}^{N} q^{(t)}_i(z_i), \quad q^{(t)}_i(z_i) = p(z_i | x_i, \theta^{(t-1)})
\]
General Mixture Model

**M-step**

\[ \mathbb{E}_{q(t)} \left( \sum_{i=1}^{N} \sum_{k=1}^{K} \delta_{k, z_i} \left( \log \pi_k + \log p_c(x_i | \theta_k) \right) \right) \]

\[ = \sum_{i=1}^{N} \sum_{k=1}^{K} \mathbb{E}_{q(t)} \left[ \delta_{k, z_i} \left( \log \pi_k + \log p_c(x_i | \theta_k) \right) \right] \]

\[ = \sum_{i=1}^{N} \sum_{k=1}^{K} \mathbb{E}_{q(t)} \left[ \delta_{k, z_i} \left( \log \pi_k + \log p_c(x_i | \theta_k) \right) \right] \]

\[ = \sum_{i=1}^{N} \sum_{k=1}^{K} q_i^{(t)}(k) \left( \log \pi_k + \log p_c(x_i | \theta_k) \right) \]

Now we should maximize the above over all parameters \( \theta \).
General Mixture Model

M-step

The optimization problem for different parameters is:

\[ \hat{\theta}_k^{(t)} = \arg\max_{\theta_k} \sum_{i=1}^{N} q_i^{(t)}(k) \log p_c(x_i | \theta_k) \]

\[ \hat{\pi}^{(t)} = \arg\max_{\pi} \sum_{i=1}^{N} q_i^{(t)}(k) \log \pi_k, \text{ subject to } \sum_{k=1}^{K} \pi_k = 1, \pi_k \leq 0 \]

The second optimization problem result in the following answer:

\[ \hat{\pi}_k^{(t)} = \frac{1}{N} \sum_{i=1}^{N} q_i^{(t)}(k) \]
Multivariate Gaussian as $p_c$

Algorithm

The algorithm is as follows:

- Initialize $\{\mu_k^{(0)}, \Sigma_k^{(0)}\}_{k=1}^K$ randomly and $\pi^{(0)} = \frac{1}{K} \mathbf{1}$.

- For $t = 1, 2, \ldots, T$, repeat:
  - **E-step:**
    $$q_i^{(t)}(z_i = k) = \frac{\pi_k^{(t-1)} p_c(x_i | \mu_k^{(t-1)}, \Sigma_k^{(t-1)})}{\sum_{l=1}^K \pi_l^{(t-1)} p_c(x_i | \mu_l^{(t-1)}, \Sigma_l^{(t-1)})}, \quad \left\{ \begin{array}{l} k = 1, \ldots, K \\ i = 1, \ldots, N \end{array} \right.$$  
  
  - **M-step:**
    $$\pi_k^{(t)} = \frac{1}{N} \sum_{i=1}^N q_i^{(t)}(k)$$
    $$\mu_k^{(t)} = \frac{1}{N \pi_k^{(t)}} \sum_{i=1}^N q_i^{(t)}(k) x_i$$
    $$\Sigma_k^{(t)} = \frac{1}{N \pi_k^{(t)}} \sum_{i=1}^N q_i^{(t)} \left( x_i - \mu_k^{(t)} \right) \left( x_i - \mu_k^{(t)} \right)^T$$