Gaussian Mixture Model and Applications with Expectation-Maximization

KK Thuwajit (kkuroma) and Khalid "Standard Deviation Male"

April 27, 2025

Contents

Preface

We aim to establish the mathematical derivations that will be use throughout the document here.

Theorem 1 (MLE of Categorical Sampling)**.** *Given $y_1, y_2, \ldots, y_n$ are sampled IID from $\{1, 2, \ldots, k\}$ such that $\mathbb{P}(y_i = j) = \phi_j$, $\sum_{j=1}^{k} = \phi_j$, the MLE for ϕ_i is given by*

$$\hat{\phi}_j = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}\{y_i = j\}$$

Proof. The likelihood of the parameters $\phi_1, \phi_2, \ldots, \phi_k$ given observations $y_1, y_2, \ldots, y_n$ can be expressed as

$$f(\phi_1, \phi_2, \ldots, \phi_k) = \prod_{i=1}^{n} \phi_{y_i} = \prod_{j=1}^{k} \phi_j^{\frac{1}{n} \sum_{i=1}^{n} \mathbb{I}\{y_i = j\}}$$

and consequently the log-likelihood

$$g(\phi_1, \phi_2, \ldots, \phi_k) = \frac{1}{n} \sum_{j=1}^{k} \log(\phi_j) \left(\sum_{i=1}^{n} \mathbb{I}\{y_i = j\} \right)$$

we are able to solve for the MLE using Lagrangian multipliers. Consider the objective function

$$\mathcal{L}(\phi_1, \phi_2, \ldots, \phi_k, \lambda) = \sum_{j=1}^{k} \log(\phi_j) \left(\sum_{i=1}^{n} \mathbb{I}\{y_i = j\} \right) + \lambda \left(1 - \sum_{j=1}^{k} \phi_j \right)$$

Taking the partial derivative for the gradient

$$\frac{\partial}{\partial \phi_l} \mathcal{L}(\phi_1, \phi_2, \ldots, \phi_k, \lambda) = \frac{\sum_{i=1}^{n} \mathbb{I}\{y_i = l\}}{\phi_l} - \lambda$$

and the second partial derivative for the hessian

$$\frac{\partial^2}{\partial \phi_l^2} \mathcal{L}(\phi_1, \phi_2, \ldots, \phi_k, \lambda) = -\frac{\sum_{i=1}^{n} \mathbb{I}\{y_i = l\}}{\phi_l^2} \text{ and } \frac{\partial}{\partial \phi_l} \frac{\partial}{\partial \phi_m} \mathcal{L}(\phi_1, \phi_2, \ldots, \phi_k, \lambda) = 0$$

We can see that the hessian is negative semi-definite, meaning setting the gradient as $\mathbf{0}$ or equivalently $\phi_l = \frac{\sum_{i=1}^{n} \mathbb{I}\{y_i = l\}}{\lambda}$ yields the maximum. Since

$$1 = \sum_{j=1}^{k} \phi_j = \sum_{j=1}^{k} \frac{\sum_{i=1}^{n} \mathbb{I}\{y_i = j\}}{\lambda} = \frac{n}{\lambda}$$

therefore, $\lambda = n$. Put together, the empirical prediction $\hat{\phi}_j = \frac{\sum_{i=1}^{n} \mathbb{I}\{y_i = j\}}{n}$ is the MLE, completing the proof □

Theorem 2 (MLE of Multivariate Gaussian)**.** *Given $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n \in \mathbb{R}^d$ are sampled IID from $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$, the MLE for $\boldsymbol{\mu}$ and Σ are given by*

$$\boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \text{ and } \Sigma = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top$$

Proof. The likelihood of the parameters $\boldsymbol{\mu}$ and Σ given observations $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ can be expressed as

$$f(\boldsymbol{\mu}, \Sigma) = \prod_{i=1}^{n} \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) \right)$$

and consequently the log-likelihood

$$g(\boldsymbol{\mu}, \Sigma) = -\frac{n}{2} \log |\Sigma| - \frac{nd}{2} \log(2\pi) - \frac{1}{2} \sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu})$$

Fancy footnote hehe

To find the MLE, we maximize $g(\boldsymbol{\mu}, \Sigma)$ with respect to $\boldsymbol{\mu}$ and Σ. First, optimizing with respect to $\boldsymbol{\mu}$, we take the gradient and hessian

$$\frac{\partial}{\partial \boldsymbol{\mu}} g(\boldsymbol{\mu}, \Sigma) = \Sigma^{-1} \sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu}) \text{ and } \frac{\partial^2}{\partial \boldsymbol{\mu}^2} g(\boldsymbol{\mu}, \Sigma) = -\Sigma^{-1}$$

Since Σ has to be positive semi-definite to be a covariance, so does Σ^{-1}, which implies the gradient to zero or

$$\Sigma^{-1} \sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu}) = 0 \implies \sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu}) = 0 \implies n\boldsymbol{\mu} = \sum_{i=1}^{n} \mathbf{x}_i \implies \boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i$$

yields the maximum. Next, optimizing with respect to Σ, plugging $\hat{\boldsymbol{\mu}}$ back in, the log-likelihood simplifies to

$$g(\boldsymbol{\mu}, \Sigma) = -\frac{n}{2} \log |\Sigma| - \frac{1}{2} \sum_{i=1}^{n} (\mathbf{x}_i - \hat{\boldsymbol{\mu}})^\top \Sigma^{-1} (\mathbf{x}_i - \hat{\boldsymbol{\mu}})$$

Here, we use the trace trick:

$$-\frac{n}{2} \log |\Sigma| - \frac{1}{2} \sum_{i=1}^{n} \operatorname{Tr}\left((\mathbf{x}_i - \hat{\boldsymbol{\mu}})^\top \Sigma^{-1} (\mathbf{x}_i - \hat{\boldsymbol{\mu}}) \right)$$

Taking the derivative with respect to Σ^{-1} using the these identities

$$\frac{\partial}{\partial \Sigma^{-1}} \log |\Sigma| = \Sigma \quad \text{and} \quad \frac{\partial}{\partial \Sigma^{-1}} \operatorname{tr}(A\Sigma^{-1}) = -A$$

we obtain

$$\frac{\partial}{\partial \Sigma^{-1}} g(\boldsymbol{\mu}, \Sigma) = \frac{n}{2} \Sigma - \frac{1}{2} \sum_{i=1}^{n} (\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^\top$$

Setting the gradient to zero, we get

$$n\Sigma = \sum_{i=1}^{n} (\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^\top \implies \Sigma = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^\top$$

Put together, the empirical predictions are indeed the MLE, thus completing the proof. □

1. Gaussian Mixture Model

1.1 Defining Gaussian Mixture

Definition 1 (Gaussian Mixture Problem)**.** For $j = 1, 2, \ldots, k$ suppose there exists some fixed, hidden parameters $\phi_j \in \mathbb{R}, \boldsymbol{\mu}_j \in \mathbb{R}^d, \Sigma_j \in \mathbb{R}^{d\times d}$ such that $\sum_{j=1}^{k} \phi_j = 1$ and Σ_j is a valid covariance matrix, we consider the following procedure

- For $i = 1, 2, \ldots, n$ sample IID y_i from $\{1, 2, \ldots, k\}$ such that $\mathbb{P}(y_i) = \phi_j$
- For $i = 1, 2, \ldots, n$ sample IID $\mathbf{x}_i$ from $\mathcal{N}(\boldsymbol{\mu}_{y_i}, \Sigma_{y_i})$

There are two versions of the problems:

- **Supervised**: for $i = 1, 2, \ldots, n$ we are given $\mathbf{x}_i, y_i$. Find the estimate for $\phi_j, \boldsymbol{\mu}_j, \Sigma_j$
- **Unsupervised**: for $i = 1, 2, \ldots, n$ we are given only $\mathbf{x}_i$. Find the estimate for $\phi_j, \boldsymbol{\mu}_j, \Sigma_j$

This is assuming that there exists k different Gaussian clusters, and $\mathbf{x}_i$ is sampled from one of them with probability specified by ϕ_j. Notice that the supervised version of the problem essentially boils down to running the regular Multivariate Gaussian problem for each class y_i can take. We are more interested in the unsupervised problem, as the MLE could not be solved in a closed form.

1.2 Potential Applications of Gaussian Mixture

We present two applications of the unsupervised version of the Gaussian Mixture defined above. The exact implementations will be explained in later sessions.

- **Unsupervised classification**: the resulting estimate for the Gaussian clusters' mean and covariance allows estimates for the unknown label y_i to be retrieved from $\mathbf{x}_i$.
- **Conditional generation**: the Gaussian Mixture model allows data outside of the given set to be sampled from the resulting Gaussian distributions, conditioned on one of the possible classes.

1.3 Gaussian Mixture as MLE Problem

Given access to only the data vectors $\mathbf{x}_i$ for $i = 1, 2, \ldots, n$, one may model the likelihood of the parameters $\{\phi_j, \boldsymbol{\mu}_j, \Sigma_j\}_{j=1}^{k}$ as follows:

$$
\begin{aligned}
f(\{\phi_j, \boldsymbol{\mu}_j, \Sigma_j\}_{j=1}^{k}) &= \prod_{i=1}^{n} \sum_{z=1}^{k} \mathbb{P}(Z = k) p(\mathbf{x}_n; \boldsymbol{\mu}_k, \Sigma_k) \\
&= \prod_{i=1}^{n} \sum_{z=1}^{k} \prod_{i=1}^{n} \frac{\phi_z}{(2\pi)^{d/2}|\Sigma_z|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_z)^\top \Sigma_z^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_z)\right)
\end{aligned}
$$

This results in a double summation once a logarithm is taken, particularly with the term inside logarithm being a summation. While a closed form MLE solution doesn't exist for this expression, the use of the EM algorithm explained over the next sections will help us iteratively solve for a good approximate.

2. Expectation-Maximization (EM) Algorithm

This section aims to define a class of algorithms known as the Expectation-Maximization (EM) algorithm. These algorithms are designed to provide an approximate solution to MLE problems where, because of an unknown set of "latent" random variables, don't lead to a closed form solution.

Definition 2 (Distribution Learning with Hidden Latent Variables)**.** Suppose there exists

- A distribution $\mathcal{P}$ from $\mathbb{R}^d$ with a joint pdf p and an unknown parameter θ_*
- A distribution $\mathcal{Q}_*$ from $\mathbb{R}^s$ with a pdf q_*.

Note that $\mathcal{P}$ and $\mathcal{Q}_*$ can either be explicitly given or not. Hidden behind the scenes, we sample IID:

- For $i = 1, 2, \ldots, n$, $\mathbf{z}_i \sim \mathcal{Q}_*$ (that is, following the pdf of $q_*(\mathbf{z}_i)$)
- For $i = 1, 2, \ldots, n$, $\mathbf{x}_i|\mathbf{z}_i \sim \mathcal{P}_{\theta_*}$ (that is, following the pdf of $p(\mathbf{x}_i|\mathbf{z}_i; \theta_*)$)

Given access to only $\mathbf{x}_i$ for $i = 1, 2, \ldots, n$, find an estimate for θ_* and q_* (the latter if not explicitly given).

2.1 The Evidence Lower Bound (ELBO)

Like the expression for Gaussian Mixture Models, we may model the likelihood of θ and q as follows:

$$\begin{aligned}
f(\theta, q) &= \prod_{i=1}^{n} \int p(\mathbf{x}_i, \mathbf{z}; \theta) d\mathbf{z} \\
&= \prod_{i=1}^{n} \int q(\mathbf{z}) p(\mathbf{x}_i|\mathbf{z}; \theta) d\mathbf{z} \\
&= \prod_{i=1}^{n} \mathbb{E}_{\mathbf{z}} \left[p(\mathbf{x}_i|\mathbf{z}; \theta) \right] \\
g(\theta, q) &= \log(f(\theta, q)) \\
&= \sum_{i=1}^{n} \log \left(\mathbb{E}_{\mathbf{z}} \left[p(\mathbf{x}_i|\mathbf{z}; \theta) \right] \right) \quad (*) \\
&\geq \sum_{i=1}^{n} \mathbb{E}_{\mathbf{z}} \left[\log \left(p(\mathbf{x}_i|\mathbf{z}; \theta) \right) \right] \quad (**) \\
&= \sum_{i=1}^{n} \int q(\mathbf{z}) \log(p(\mathbf{x}_i|\mathbf{z}; \theta)) d\mathbf{z}
\end{aligned}$$

We're able to go from $(*)$ to $(**)$ using Jensen's inequality, as log is a convex function. This allows us to minimize the lower bound of $g(\theta, q)$ which indirectly optimizes the log-likelihood itself. This lower bound term is therefore dubbed the **evidence lower bound** (ELBO):

$$\text{ELBO}(\mathbf{x}, \theta, q) = \int q(\mathbf{z}) \log(p(\mathbf{x}|\mathbf{z}; \theta)) d\mathbf{z}$$

Finding the optimal prior: For the tightest bound with Jensen inequality to hold, the expectation must be taken over a constant, meaning $p(\mathbf{x}_i|\mathbf{z}; \theta) = c$ over some constant c. By Bayes' law,

$$\begin{aligned}
p(\mathbf{x}_i|\mathbf{z}; \theta) &= c \\
\frac{p(\mathbf{x}_i, \mathbf{z}; \theta)}{q(\mathbf{z})} &= c \\
q(\mathbf{z}) &\propto p(\mathbf{x}_i, \mathbf{z}; \theta)
\end{aligned}$$

And since $\int_{\mathbf{z}} q(\mathbf{z}) d\mathbf{z} = 1$,

$$\begin{aligned} q(\mathbf{z}) &= \frac{q(\mathbf{z})}{\int_{\mathbf{z}} q(\mathbf{z}) d\mathbf{z}} \\ &= \frac{p(\mathbf{x}_i, \mathbf{z}; \theta)}{\int_{\mathbf{z}} p(\mathbf{x}_i, \mathbf{z}; \theta)} \\ &= \frac{p(\mathbf{x}_i, \mathbf{z}; \theta)}{p(\mathbf{x}_i, \theta)} \\ &= p(\mathbf{z}|\mathbf{x}_i; \theta) \end{aligned}$$

With this, we may re-write the lower bound of our log-likelihood as follows:

$$g(\theta, q) \geq \sum_{i=1}^{n} \text{ELBO}(\mathbf{x}_i, \theta, q)$$

Moreover, we may isolate θ from the expression by summing up all

2.2 Defining EM Algorithms

2.3 EM Algorithm for Gaussian Mixture Models

2.4 EM Algorithm for Variational Inference

3. Unsupervised Classification

3.1 K-means Clustering

3.2 Gaussian Mixture Classification

3.3 Comparison

4. Conditional Generation

4.1 Pure Gaussian Mixture Models

4.2 Gaussian Mixture Variational Autoencoders

4.3 Comparison