# Chapter 3

# Probabilistic Models for Clustering

**Hongbo Deng**
*University of Illinois at Urbana-Champaign*
*Urbana, IL*
hbdeng@illinois.edu

**Jiawei Han**
*University of Illinois at Urbana-Champaign*
*Urbana, IL*
hanj@illinois.edu



## 3.1 Introduction

Probabilistic model-based clustering techniques have been widely used and have shown promising results in many applications, ranging from image segmentation [71, 15], handwriting recognition [60], document clustering [36, 81], topic modeling [35, 14] to information retrieval [43]. Model-based clustering approaches attempt to optimize the fit between the observed data and some mathematical model using a probabilistic approach. Such methods are often based on the assumption that the data are generated by a mixture of underlying probability distributions. In practice, each cluster can be represented mathematically by a parametric probability distribution, such as a Gaussian or a Poisson distribution. Thus, the clustering problem is transformed into a parameter estimation problem since the entire data can be modeled by a mixture of $K$ component distributions.

Data points (or objects) that belong most likely to the same distribution can then easily be defined as clusters.

In this chapter, we introduce several fundamental models and algorithms for probabilistic clustering, including mixture models [45, 48, 25], EM algorithm [23, 46, 53, 16], and probabilistic topic models [35, 14]. For each probabilistic model, we will introduce its general framework of modeling, the probabilistic explanation, the standard algorithms to learn the model, and its applications. Mixture models are probabilistic models which are increasingly used to find the clusters for univariate and multivariate data. We therefore begin our discussion of mixture models in Section 3.2, in which the values of the discrete latent variables can be interpreted as the assignments of data points to specific components (i.e., clusters) of the mixture. To find maximum likelihood estimations in mixture models, a general and elegant technique, the Expectation-Maximization (EM) algorithm, is introduced in Section 3.3. We first use the Gaussian mixture model to motivate the EM algorithm in an informal way, and then give a more general view of the EM algorithm, which is a standard learning algorithm for many probabilistic models. In Section 3.4, we present two popular probabilistic topic models, i.e., probabilistic latent semantic analysis (PLSA) [35] and latent Dirichlet allocation (LDA) [14], for document clustering and analysis. Note that some of the methods (e.g., EM/mixture models) may be more appropriate for quantitative data, whereas others such as topic models, PLSI, and LDA, are used more commonly for text data. Finally, we give the conclusions and summary in Section 3.5.

## 3.2 Mixture Models

Mixture models for cluster analysis [75, 45, 48, 25, 47] have been addressed in a number of ways. The underlying assumption is that the observations to be clustered are drawn from one of several components, and the problem is to estimate the parameters of each component so as to best fit the data. Inferring the parameters of these components and identifying which component produced each observation leads to a clustering of the set of observations. In this section, we first give an overview of mixture modeling, then introduce two most common mixtures, Gaussian mixture model and Bernoulli mixture model, and finally discuss several issues about model selection.

### 3.2.1 Overview

Suppose we have a set of data points $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ consisting of $N$ observations of a $D$-dimensional random variable $\mathbf{x}$. The random variable $\mathbf{x}_n$ is assumed to be distributed according to a mixture of $K$ components. Each component (i.e., cluster) is mathematically represented by a parametric distribution. An individual distribution used to model a specific cluster is often referred to as a component distribution. The entire data set is therefore modeled by a mixture of these distributions. Formally, the mixture distribution, or probability density function, of $\mathbf{x}_n$ can be written as

$$p(\mathbf{x}_n) = \sum_{k=1}^{K} \pi_k p(\mathbf{x}_n|\theta_k) \tag{3.1}$$

where $\pi_1, \ldots, \pi_K$ are the *mixing probabilities* (i.e., mixing coefficients or weights), each $\theta_k$ is the set of parameters specifying the $k$th component, and $p(\mathbf{x}_n|\theta_k)$ is the component distribution. In order to be valid probabilities, the mixing probabilities $\{\pi_k\}$ must satisfy

$$0 \leq \pi_k \leq 1 \ (k = 1, \ldots, K), \ \textit{and} \ \sum_{k=1}^{K} \pi_k = 1.$$

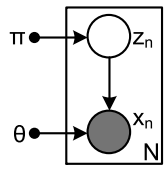



**FIGURE 3.1:** Graphical representation of a mixture model. Circles indicate random variables, and shaded and unshaded shapes indicate observed and latent (i.e., unobserved) variables.

An obvious way of generating a random sample $\mathbf{x}_n$ with the mixture model, given by (3.1), is as follows. Let $z_n$ be a categorical random variable taking on the values $1,\ldots,K$ with probabilities $p(z_n = k) = \pi_k$ (also denoted as $p(z_{nk} = 1) = \pi_k$). Suppose that the conditional distribution of $\mathbf{x}_n$ given $z_n = k$ is $p(\mathbf{x}_n|\theta_k)$. Then the marginal distribution of $\mathbf{x}_n$ is obtained by summing the joint distribution over all possible values of $z_n$ to given $p(\mathbf{x}_n)$ as (3.1). In this context, the variable $z_n$ can be thought of as the component (or cluster) label of the random sample $\mathbf{x}_n$. Instead of using a single categorical variable $z_n$, we introduce a $K$-dimensional binary random vector $\mathbf{z}_n$ to denote the component label for $\mathbf{x}_n$. The $K$-dimensional random variable $\mathbf{z}_n$ has a 1-of-$K$ representation, in which one of the element $z_{nk} = (\mathbf{z}_n)_k$ equals to 1, and all other elements equal to 0, denoting the component of origin of $\mathbf{x}_n$ is equal to $k$. For example, if we have a variable with $K = 5$ clusters and a particular observation $\mathbf{x}_n$ of the variable happens to correspond to the cluster where $z_{n4} = 1$, then $\mathbf{z}_n$ will be represented by $\mathbf{z}_n = (0,0,0,1,0)^T$. Note that the values of $z_{nk}$ satisfy $z_{nk} \in \{0,1\}$ and $\sum_{k=1}^{K} z_{nk} = 1$. Because $\mathbf{z}_n$ uses a 1-of-$K$ representation, the marginal distribution over $\mathbf{z}_n$ is specified in terms of *mixing probabilities* $\pi_k$, such that

$$p(\mathbf{z}_n) = \pi_1^{z_{n1}}\pi_2^{z_{n2}}\ldots\pi_K^{z_{nK}} = \prod_{k=1}^{K}\pi_k^{z_{nk}}. \tag{3.2}$$

Similarly, the conditional distribution of $\mathbf{x}_n$ given $\mathbf{z}_n$ can be written in the form

$$p(\mathbf{x}_n|\mathbf{z}_n) = \prod_{k=1}^{K} p(\mathbf{x}_n|\theta_k)^{z_{nk}}. \tag{3.3}$$

The joint distribution is given by $p(\mathbf{z}_n)p(\mathbf{x}_n|\mathbf{z}_n)$, and the marginal distribution of $\mathbf{x}_n$ is obtained as

$$p(\mathbf{x}_n) = \sum_{\mathbf{z}_n} p(\mathbf{z}_n)p(\mathbf{x}_n|\mathbf{z}_n) = \sum_{k=1}^{K}\pi_k p(\mathbf{x}_n|\theta_k). \tag{3.4}$$

Thus, the above marginal distribution of $\mathbf{x}$ is an equivalent formulation of the mixture model involving an explicit latent variable. The graphical representation of the mixture model is shown in Figure 3.1. From the generative process point of view, a given set of data points could have been generated by repeating the following procedure $N$ times, once for each data point $\mathbf{x}_n$:

(a) Choose a hidden component (i.e., cluster) label $\mathbf{z}_n \sim Mult_K(1,\pi)$. This selects the $k$th component from which to draw point $\mathbf{x}_n$.

(b) Sample a data point $\mathbf{x}_n$ from the $k$th component according to the conditional distribution $p(\mathbf{x}_n|\theta_k)$.

Because we have represented the marginal distribution in the form $p(\mathbf{x}_n) = \sum_{\mathbf{z}_n} p(\mathbf{x}_n,\mathbf{z}_n)$, it follows that for every observed data point $\mathbf{x}_n$ there is a corresponding latent variable $\mathbf{z}_n$. Using Bayes' theorem, we can obtain the conditional probability of $z_{nk} = 1$ given $\mathbf{x}_n$ as

$$p(z_{nk} = 1|\mathbf{x}_n) = \frac{p(z_{nk} = 1)p(\mathbf{x}_n|z_{nk} = 1)}{\sum_{j=1}^{K} p(z_{nj} = 1)p(\mathbf{x}_n|z_{nj} = 1)} = \frac{\pi_k p(\mathbf{x}_n|\theta_k)}{\sum_{j=1}^{K}\pi_j p(\mathbf{x}_n|\theta_j)}, \tag{3.5}$$

where $\pi_k$ (i.e., $p(z_{nk}=1)$) is the *prior probability* that data point $\mathbf{x}_n$ was generated from component $k$, and $p(z_{nk}=1|\mathbf{x}_n)$ is the *posterior probability* that the observed data point $\mathbf{x}_n$ came from component $k$. In the following, we shall use $\gamma(z_{nk})$ to denote $p(z_{nk}=1|\mathbf{x}_n)$, which can also be viewed as the *responsibility* that component $k$ takes for explaining the observation $\mathbf{x}_n$.

In this formulation of the mixture model, we need to infer a set of parameters from the observation, including the *mixing probabilities* $\{\pi_k\}$ and the parameters for the *component distributions* $\{\theta_k\}$. The number of components $K$ is considered fixed, but of course in many applications, the value of $K$ is unknown and has to be inferred from the available data [25]. Thus, the overall parameter of the mixture model is $\Theta = \{\pi_1,\ldots,\pi_K,\theta_1,\ldots,\theta_K\}$. If we assume that the data points are drawn independently from the distribution, then we can write the probability of generating all the data points in the form

$$p(\mathbf{X}|\Theta) = \prod_{n=1}^{N}\sum_{k=1}^{K}\pi_k p(\mathbf{x}_n|\theta_k), \tag{3.6}$$

or in a logarithm form

$$\log p(\mathbf{X}|\Theta) = \sum_{n=1}^{N}\log\sum_{k=1}^{K}\pi_k p(\mathbf{x}_n|\theta_k). \tag{3.7}$$

In statistics, maximum likelihood estimation (MLE) [23, 8, 41] is an important statistical approach for parameter estimation,

$$\Theta_{ML} = \arg\max_{\Theta}\{\log p(\mathbf{X}|\Theta)\},$$

which considers the best estimate as the one that maximizes the probability of generating all the observations. Sometimes we have a priori information $p(\Theta)$ about the parameters, and it can be incorporated into the mixture models. Thus, the maximum a posteriori (MAP) estimation [70, 27] is used instead,

$$\Theta_{MAP} = \arg\max_{\Theta}\{\log p(\mathbf{X}|\Theta) + \log p(\Theta)\},$$

which considers the best estimate as the one that maximizes the posterior probability of $\Theta$ given the observed data. MLE and MAP give a unified approach to parameter estimation, which is well-defined in the case of the normal distribution and many other problems.

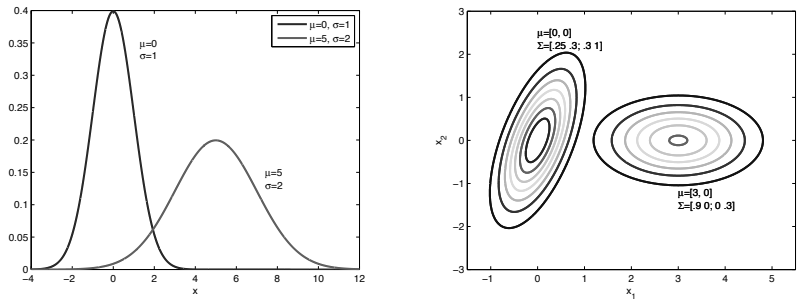
As mentioned previously, each cluster is mathematically represented by a parametric distribution. In principle, the mixtures can be constructed with any types of components, and we could still have a perfectly good mixture model. In practice, a lot of effort is given to parametric mixture models, where all components are from the same parametric family of distributions, but with different parameters. For example, they might all be Gaussians with different means and variances, or all Poisson distributions with different means, or all power laws with different exponents. In the following section, we will introduce the two most common mixtures, mixture of Gaussian (continuous) and mixture of Bernoulli (discrete) distributions.

### 3.2.2 Gaussian Mixture Model

The most well-known mixture model is the Gaussian mixture model (GMM), where each component is a Gaussian distribution. Recently, GMM has been widely used for clustering in many applications, such as speaker identification and verification [62, 61], image segmentation [15, 56], and object tracking [71].

The Gaussian, also known as the normal distribution, is a widely used model for the distribution of continuous variables. In the case of a single variable $x$, the Gaussian distribution can be written in the form

$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}}\exp\{-\frac{1}{2\sigma^2}(x-\mu)^2\}, \tag{3.8}$$




(a) (b)

**FIGURE 3.2 (See color insert):** (a) Plots of the univariate Gaussian distribution given by (3.8) for various parameters of $\mu$ and $\sigma$, and (b) contours of the multivariate (2-D) Gaussian distribution given by (3.9) for various parameters of $\mu$ and $\Sigma$.

where $\mu$ is the mean and $\sigma^2$ is the variance. Figure 3.2(a) shows plots of the Gaussian distribution for various values of the parameters. For a $D$-dimensional vector $\mathbf{x}$, the multivariate Gaussian distribution takes the form

$$\mathcal{N}(\mathbf{x}|\mu,\Sigma) = \frac{1}{(2\pi)^{D/2}}\frac{1}{|\Sigma|^{1/2}}\exp\{-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)\}, \tag{3.9}$$

where $\mu$ is a $D$-dimensional mean vector, $\Sigma$ is a $D \times D$ covariance matrix, and $|\Sigma|$ denotes the determinant of $\Sigma$. Figure 3.2(b) shows contours of the Gaussian distribution for various values of the parameters.

In the Gaussian mixture model, each component is represented by the parameters of a multivariate Gaussian distribution $p(\mathbf{x}_k|\theta_k) = \mathcal{N}(\mathbf{x}_n|\mu_k,\Sigma_k)$. Based on (3.1), the Gaussian mixture distribution can be written as a linear superposition of Gaussians in the form

$$p(\mathbf{x}_n|\Theta) = p(\mathbf{x}_n|\pi,\mu,\Sigma) = \sum_{k=1}^{K}\pi_k\mathcal{N}(\mathbf{x}_n|\mu_k,\Sigma_k). \tag{3.10}$$

For a given set of observations $\mathbf{X}$, the log-likelihood function is given by

$$l(\Theta) = \log p(\mathbf{X}|\Theta) = \sum_{n=1}^{N}\log p(\mathbf{x}_n|\Theta) = \sum_{n=1}^{N}\log\sum_{k=1}^{K}\pi_k\mathcal{N}(\mathbf{x}_n|\mu_k,\Sigma_k). \tag{3.11}$$

To find maximum likelihood solutions that are valid at local maxima, we compute the derivatives of $\log p(\mathbf{X}|\pi,\mu,\Sigma)$ with respect to $\pi_k$, $\mu_k$, and $\Sigma_k$, respectively. The derivative with respect to the mean $\mu_k$ is given by

$$\frac{\partial l}{\partial\mu_k} = \sum_{n=1}^{N}\frac{\pi_k\mathcal{N}(\mathbf{x}_n|\mu_k,\Sigma_k)}{\sum_{j=1}^{K}\pi_j\mathcal{N}(\mathbf{x}_n|\mu_j,\Sigma_j)}\Sigma_k^{-1}(\mathbf{x}_n-\mu_k) = \sum_{n=1}^{N}\gamma(z_{nk})\Sigma_k^{-1}(\mathbf{x}_n-\mu_k),$$

where we have used (3.9) and (3.5) for the Gaussian distribution and the responsibilities (i.e., posterior probabilities), respectively. Setting this derivative to zero and multiplying by $\Sigma_k$, we obtain

$$\mu_k = \frac{\sum_{n=1}^{N}\gamma(z_{nk})\mathbf{x}_n}{\sum_{n=1}^{N}\gamma(z_{nk})} \tag{3.12}$$

and

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}_n|\mu_j, \Sigma_j)}. \tag{3.13}$$

We can interpret that the mean $\mu_k$ for the $k$th Gaussian component is obtained by taking a weighted mean of all the points in the data set, in which the weighting factor corresponds to the posterior probability $\gamma(z_{nk})$ that component $k$ was responsible for generating $\mathbf{x}_n$.

Similarly, we set the derivative of $\log p(\mathbf{X}|\pi, \mu, \Sigma)$ with respect to $\Sigma_k$ to zero, and we obtain

$$\Sigma_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})(\mathbf{x}_n - \mu_k)(\mathbf{x}_n - \mu_k)^T}{\sum_{n=1}^{N} \gamma(z_{nk})}. \tag{3.14}$$

Similar to (3.12), each data point is weighted by the conditional probability generated by the corresponding component and with the denominator given by the effective number of points associated with the corresponding component.

The derivative of $\log p(\mathbf{X}|\pi, \mu, \Sigma)$ with respect to the mixing probabilities $\pi_k$ requires a little more work, because the values of $\pi_k$ are constrained to be positive and sum to one. This constraint can be handled using a Lagrange multiplier and maximizing the following quantity

$$\log p(\mathbf{X}|\pi, \mu, \Sigma) + \lambda(\sum_{k=1}^{K} \pi_k - 1).$$

After simplifying and rearranging we obtain

$$\pi_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk})}{N} \tag{3.15}$$

so that the mixing probabilities for the $k$th component are given by the average responsibility that the component takes for explaining the data points.

It is worth noting that the Equations (3.12), (3.14), and (3.15) are not the closed-form solution for the parameters of the mixture model. The reason is that these equations are intimately coupled with Equation (3.13). More specifically, the responsibilities $\gamma(z_{nk})$ given by Equation (3.13) depend on all the parameters of the mixture model, while all the results of (3.12), (3.14), and (3.15) rely on $\gamma(z_{nk})$. Therefore, maximizing the log likelihood function for a Gaussian mixture model turns out to be a very complex problem. An elegant and powerful method for finding maximum likelihood solutions for models with latent variables is called the *Expectation-Maximization* algorithm or EM algorithm [23].

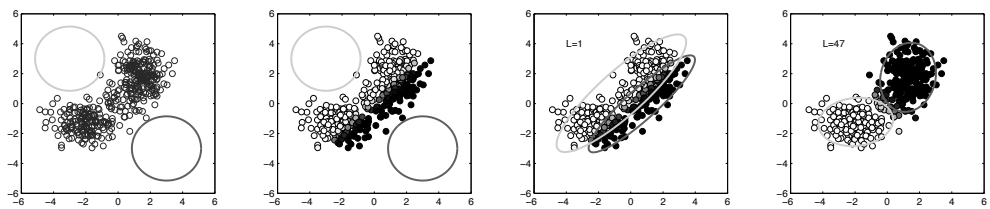
However, the above equations do suggest an iterative solution for the Gaussian mixture model, which turns out to be an instance of the EM algorithm for the particular case of the Gaussian mixture model. Here we shall give such an iterative algorithm in the context of Gaussian mixture model, and later we shall give a general framework of EM algorithm in Section 3.3. This algorithm is started by initializing with guesses about the parameters, including the means, covariances, and mixing probabilities. Then we alternative between two updating steps, the *expectation* step and *maximization* step. In the *expectation* step, or E-step, we use the current parameters to calculate the responsibilities (i.e., posterior probabilities) according to (3.13). In the *maximization* step, or M-step, we maximize the log-likelihood with the updated responsibilities, and reestimate the means, covariances, and mixing coefficients using (3.12), (3.14), and (3.15). In Section 3.3, we shall show that each iteration of the EM algorithm is guaranteed to increase the log-likelihood. In practice, the EM algorithm is converged when the change in the log-likelihood or the parameter values fall below some threshold. We summarize the EM algorithm for Gaussian mixtures in Algorithm 11.

As illustrated in Figure 3.3, the EM algorithm for a mixture of two Gaussian components is applied to a random generated data set. Plot (a) shows the data points together with the random initialization of the mixture model in which the two Gaussian components are shown. Plot (b) shows

**Algorithm 11** EM for Gaussian Mixtures

Given a set of data points and a Gaussian mixture model, the goal is to maximize the log-likelihood with respect to the parameters.

1: Initialize the means $\mu_k^0$, covariances $\Sigma_k^0$, and mixing probabilities $\pi_k^0$.
2: **E-step**: Calculate the responsibilities $\gamma(z_{nk})$ using the current parameters based on Equation (3.13).
3: **M-step**: Update the parameters using the current responsibilities. Note that we first update the new means using (3.12), then use these new values to calculate the covariances using (3.14), and finally reestimate the mixing probabilities using (3.15).
4: Compute the log-likelihood using (3.10) and check for convergence of the algorithm. If the convergence criterion is not satisfied, then repeat steps 2–4; otherwise, return the final parameters.




(a) (b) (c) (d)

**FIGURE 3.3:** Illustration of the EM algorithm for two Gaussian components.

the result of the initial E-step, in which each data point is depicted using a proportion of white ink and black ink according to the posterior probability of having been generated by the corresponding component. Plot (c) shows the situation after the first M-step, in which the means and covariances of both components have changed. Plot (d) shows the results after 47 cycles of EM, which is close to convergence.

Compared with $K$-means algorithm, the EM algorithm for GMM takes many more iterations to reach convergence [8, 55]. There will be multiple local maxima of the log-likelihood which depends on different initialization, and EM is not guaranteed to find the largest of these maxima. In order to find a suitable initialization and speed up the convergence for a Gaussian mixture model, it is common to run the $K$-means algorithm [8] and choose the means and covariances of the clusters, and the fractions of data points assigned to the respective clusters, for initializing $\mu_k^0$, $\Sigma_k^0$ and $\pi_k^0$, respectively. Another problem of the EM algorithm for GMM is the singularity of the likelihood function in which a Gaussian component collapses onto a particular data point. It is reasonable to avoid the singularities by using some suitable heuristics [8, 55], for instance, by detecting when a Gaussian component is collapsing and resetting its mean and covariance, and then continuing with the optimization.

### 3.2.3 Bernoulli Mixture Model

Although most research in mixture models has focused on distributions over continuous variables described by mixtures of Gaussians, there are many clustering tasks for which binary or discrete mixture models are better fitted. We now discuss mixtures of discrete binary variables described by Bernoulli distributions, which is also known as *latent class analysis* [42, 48].