# 12

# Density Estimation with Gaussian Mixture Models

In earlier chapters, we covered three fundamental problems in machine learning: regression (Chapter 9), classification (Chapter 10) and dimensionality reduction (Chapter 11). In this chapter, we will have a look at a fourth pillar of machine learning: density estimation. On our journey, we introduce important concepts, such as the EM algorithm a latent variable perspective of density estimation with mixture models.

When we apply machine learning to data we often aim to represent data in some way. A straightforward way is to take the data points themselves as the representation of the data, see Figure 12.1 for an example. However, this approach may be unhelpful if the dataset is huge or if we are interested in representing characteristics of the data. In density estimation, we represent the data compactly using a density, e.g., a Gaussian or Beta distribution. For example, we may be looking for the mean and variance of a data set in order to represent the data compactly using a Gaussian distribution. The mean and variance can be found using tools we discussed in Section 9.2: maximum likelihood or maximum-a-posteriori estimation. We can then use the mean and variance of this Gaussian to represent the distribution underlying the data, i.e., we think of the dataset to be a typical realization from this distribution if we were to sample from it.

In practice, the Gaussian (or similarly all other distributions we encoun-

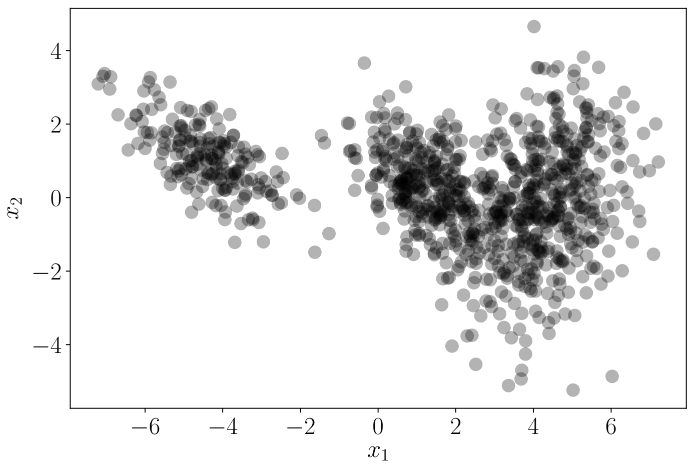



**Figure 12.1** Two-dimensional data set that cannot be meaningfully represented by a Gaussian.



Draft chapter (May 20, 2018) from "Mathematics for Machine Learning" ©2018 by Marc Peter Deisenroth, A Aldo Faisal, and Cheng Soon Ong. To be published by Cambridge University Press. Report errata and feedback to `http://mml-book.com`. Please do not post or distribute this file, please link to `http://mml-book.com`.

tered so far) have limited modeling capabilities. For example, a Gaussian approximation of the density that generated the data in Figure 12.1 would be a poor approximation. In the following, we will look at a more expressive family of distributions, which we can use for density estimation: *mixture models*.

mixture models

Mixture models can be used to describe a distribution $p(\boldsymbol{x})$ by a convex combination of $K$ simple (base) distributions

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k p_k(\boldsymbol{x}) \tag{12.1}$$

$$0 \leqslant \pi_k \leqslant 1\,, \quad \sum_{k=1}^{K} \pi_k = 1\,, \tag{12.2}$$

where the components $p_k$ are members of a family of basic distributions, e.g., Gaussians, Bernoullis or Gammas, and the $\pi_k$ are *mixture weights*. Mixture models are more expressive than the corresponding base distributions because they allow for multimodal data representations, i.e., they can describe datasets with multiple "clusters", see Figure 12.1 for an example of such a dataset.

mixture weights

In the following, we will focus on Gaussian mixture models (GMMs), where the basic distributions are Gaussians. For a given dataset, we aim to maximize the likelihood of the model parameters to train the GMM. For this purpose we will use results from Chapter 5, Section 7.2 and Chapter 6. However, unlike other application we discussed earlier (linear regression or PCA), we will not find a closed-form maximum likelihood solution. Instead, we will arrive at a set of dependent simultaneous equations, which we can only solve iteratively.

## 12.1 Gaussian Mixture Model

A *Gaussian mixture model* is a density model where we combine a finite number of $K$ Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ so that

Gaussian mixture model

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{12.3}$$

$$0 \leqslant \pi_k \leqslant 1\,, \quad \sum_{k=1}^{K} \pi_k = 1\,. \tag{12.4}$$

This convex combination of Gaussian distribution gives us significantly more flexibility for modeling complex densities than a simple Gaussian distribution (which we recover from (12.3) for $K = 1$). An illustration is given in Figure 12.2. Here, the mixture density is given as

$$p(x) = 0.5\mathcal{N}\big(x \mid -2, \tfrac{1}{2}\big) + 0.2\mathcal{N}\big(x \mid 1, 2\big) + 0.3\mathcal{N}\big(x \mid 4, 1\big)\,. \tag{12.5}$$

©2018 Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong. To be published by Cambridge University Press.

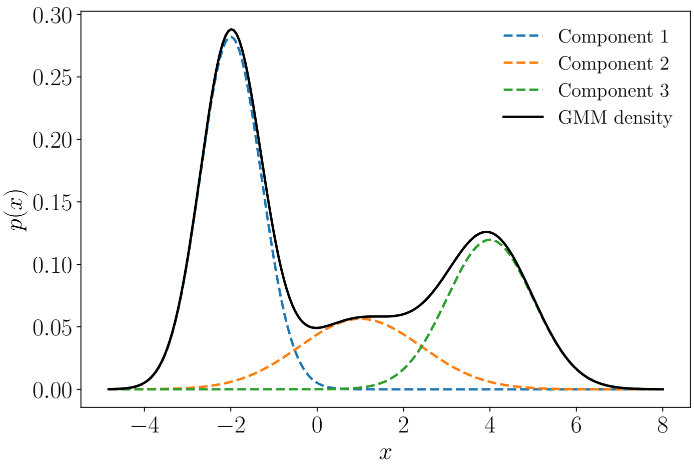



**Figure 12.2** Gaussian mixture model. The Gaussian mixture distribution (black) is composed of a convex combination of Gaussian distributions and is more expressive than any individual component. Dashed lines represent the weighted Gaussian components.

## 12.2 Parameter Learning via Maximum Likelihood

Assume we are given a data set $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ where $\boldsymbol{x}_n,\ n = 1, \ldots, N$ are drawn i.i.d. from an unknown distribution $p(\boldsymbol{x})$. Our objective is to find a good approximation/representation of this unknown distribution $p(\boldsymbol{x})$ by means of a Gaussian mixture model (GMM) with $K$ mixture components. The parameters of the GMM are the $K$ means $\boldsymbol{\mu}_k$, the covariances $\boldsymbol{\Sigma}_k$ and mixture weights $\pi_k$. We summarize all these free parameters in $\boldsymbol{\theta} := \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k,\ k = 1, \ldots, K\}$.

**Example (Initial setting)**

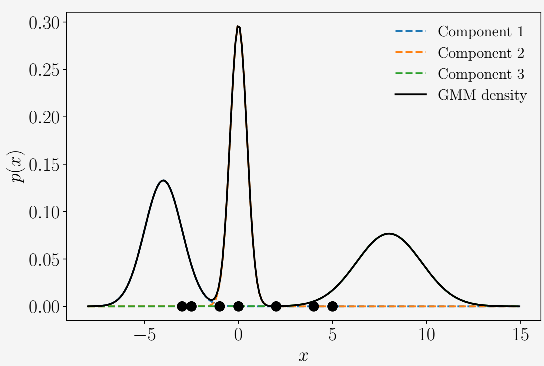



**Figure 12.3** Initial setting: GMM (black) with mixture three mixture components (dashed) and seven data points (discs).

Throughout this chapter, we will have a simple running example that helps us illustrate and visualize important concepts.

We will look at a one-dimensional data set $\boldsymbol{X} = [-3, -2.5, -1, 0, 2, 4, 5]$ consisting of seven data points. We wish to find a GMM with $K = 3$ components that models the data. We initialize the individual components as

$$p_1(x) = \mathcal{N}\big(x \,|\, -4,\, 1\big) \tag{12.6}$$
$$p_2(x) = \mathcal{N}\big(x \,|\, 0,\, 0.2\big) \tag{12.7}$$
$$p_3(x) = \mathcal{N}\big(x \,|\, 8,\, 3\big) \tag{12.8}$$

and assign them equal weights $\pi_1 = \pi_2 = \pi_3 = \frac{1}{3}$. The corresponding model (and the data points) are shown in Figure 12.3.

In the following, we detail how to obtain a maximum likelihood estimate $\boldsymbol{\theta}_{\text{ML}}$ of the model parameters $\boldsymbol{\theta}$. We start by writing down the likelihood, i.e., the probability of the data given the parameters. We exploit our i.i.d. assumption, which leads to the factorized likelihood

$$p(\boldsymbol{X}|\boldsymbol{\theta}) = \prod_{n=1}^{N} p(\boldsymbol{x}_n|\boldsymbol{\theta})\,, \qquad p(\boldsymbol{x}_n|\boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big) \tag{12.9}$$

where every individual likelihood term $p(\boldsymbol{x}_n|\boldsymbol{\theta})$ is a Gaussian mixture density. Then, we obtain the log-likelihood as

$$\log p(\boldsymbol{X}|\boldsymbol{\theta}) = \sum_{n=1}^{N} \log p(\boldsymbol{x}_n|\boldsymbol{\theta}) = \underbrace{\sum_{n=1}^{N} \log \sum_{k=1}^{K} \pi_k \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)}_{=:\mathcal{L}}\,. \tag{12.10}$$

We aim to find parameters $\boldsymbol{\theta}^*_{\text{ML}}$ that maximize the log-likelihood $\mathcal{L}$ defined in (12.10). Our "normal" procedure would be to compute the gradient $d\mathcal{L}/d\boldsymbol{\theta}$ of the log-likelihood with respect to the model parameters $\boldsymbol{\theta}$, set it to $\boldsymbol{0}$ and solve for $\boldsymbol{\theta}$. However, unlike our previous examples for maximum likelihood estimation (e.g., when we discussed linear regression in Section 9.2), we cannot obtain a closed-form solution. If we were to consider a single Gaussian as the desired density, the sum over $k$ in (12.10) vanishes, and the log can be applied directly to the Gaussian component, such that we get

$$\log \mathcal{N}\big(\boldsymbol{x} \,|\, \boldsymbol{\mu},\, \boldsymbol{\Sigma}\big) = -\tfrac{D}{2}\log(2\pi) - \tfrac{1}{2}\log|\boldsymbol{\Sigma}| - \tfrac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\,. \tag{12.11}$$

This simple form allows us find closed-form maximum likelihood estimates of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, as discussed in Chapter 8. However, in (12.10), we cannot move the $\log$ into the sum over $k$ so that we cannot obtain a simple closed-form maximum likelihood solution. However, we can exploit an iterative scheme to find good model parameters $\boldsymbol{\theta}_{\text{ML}}$: the EM algorithm.

Generally, any optimum of a function exhibits the property that its gradient with respect to the parameters must vanish (necessary condition), see Chapter 7. In our case, we obtain the following necessary conditions

©2018 Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong. To be published by Cambridge University Press.

when we optimize the log-likelihood in (12.10) with respect to the GMM parameters $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k$:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}_k} = \mathbf{0} \iff \sum_{n=1}^{N} \frac{\partial \log p(\boldsymbol{x}_n|\boldsymbol{\theta})}{\partial \boldsymbol{\mu}_k} = \mathbf{0}\,, \tag{12.12}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\Sigma}_k} = \mathbf{0} \iff \sum_{n=1}^{N} \frac{\partial \log p(\boldsymbol{x}_n|\boldsymbol{\theta})}{\partial \boldsymbol{\Sigma}_k} = \mathbf{0}\,, \tag{12.13}$$

$$\frac{\partial \mathcal{L}}{\partial \pi_k} = \mathbf{0} \iff \sum_{n=1}^{N} \frac{\partial \log p(\boldsymbol{x}_n|\boldsymbol{\theta})}{\partial \pi_k} = \mathbf{0}\,. \tag{12.14}$$

For all three necessary conditions, by applying the chain rule (see Section 5.2.2), we require partial derivatives of the form

$$\frac{\partial \log p(\boldsymbol{x}_n|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \frac{1}{p(\boldsymbol{x}_n|\boldsymbol{\theta})} \frac{\partial p(\boldsymbol{x}_n|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \tag{12.15}$$

where $\boldsymbol{\theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k,\ k = 1, \dots, K\}$ comprises all model parameters and

$$\frac{1}{p(\boldsymbol{x}_n|\boldsymbol{\theta})} = \frac{1}{\sum_{j=1}^{K} \pi_j \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_j,\, \boldsymbol{\Sigma}_j\big)}\,. \tag{12.16}$$

5007 In the following, we will compute the partial derivatives (12.12)–(12.14).

**Theorem 12.1** (Update of the GMM Means)**.** *The update of the mean parameters $\boldsymbol{\mu}_k$, $k = 1, \dots, K$ of the GMM is given by*

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} r_{nk}\boldsymbol{x}_n}{\sum_{n=1}^{N} r_{nk}} = \frac{1}{N_k}\sum_{n=1}^{N} r_{nk}\boldsymbol{x}_n\,, \tag{12.17}$$

*where we define*

$$r_{nk} := \frac{\pi_k \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_j,\, \boldsymbol{\Sigma}_j\big)}\,, \tag{12.18}$$

$$N_k := \sum_{n=1}^{N} r_{nk}\,. \tag{12.19}$$

*Proof* From (12.15), we see that the gradient of the log-likelihood with respect to the mean parameters $\boldsymbol{\mu}_k$, $k = 1, \dots, K$ requires us to compute the partial derivative

$$\frac{\partial p(\boldsymbol{x}_n|\boldsymbol{\theta})}{\partial \boldsymbol{\mu}_k} = \sum_{j=1}^{K} \pi_j \frac{\partial \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_j,\, \boldsymbol{\Sigma}_j\big)}{\partial \boldsymbol{\mu}_k} = \pi_k \frac{\partial \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)}{\partial \boldsymbol{\mu}_k} \tag{12.20}$$

$$= \pi_k (\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big) \tag{12.21}$$

5008 where we exploited that only the $k$th mixture component depends on $\boldsymbol{\mu}_k$.

We use our result from (12.21) in (12.15) and put everything together so that the desired partial derivative of $\mathcal{L}$ with respect to $\boldsymbol{\mu}_k$ is given as

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}_k} = \sum_{n=1}^{N} \frac{\partial \log p(\boldsymbol{x}_n|\boldsymbol{\theta})}{\partial \boldsymbol{\mu}_k} = \sum_{n=1}^{N} \frac{1}{p(\boldsymbol{x}_n|\boldsymbol{\theta})} \frac{\partial p(\boldsymbol{x}_n|\boldsymbol{\theta})}{\partial \boldsymbol{\mu}_k} \tag{12.22}$$

$$= \sum_{n=1}^{N} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} \underbrace{\boxed{\frac{\pi_k \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_j,\, \boldsymbol{\Sigma}_j\big)}}}_{=r_{nk}} \tag{12.23}$$

$$= \sum_{n=1}^{N} r_{nk} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} \,. \tag{12.24}$$

Here, we used the identity from (12.16) and the result of the partial derivative in (12.21) to get to the second row. The values $r_{nk}$ are often called *responsibilities*.

responsibilities

*Remark.* The responsibility $r_{nk}$ of the $k$th mixture component for data point $\boldsymbol{x}_n$ is proportional to the likelihood

$$p(\boldsymbol{x}_n|\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \pi_k \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big) \tag{12.25}$$

of the mixture component given the data point (the denominator in the definition of $r_{nk}$ is constant for all mixture components and serves as a normalizer). Therefore, mixture components have a high responsibility for a data point when the data point could be a plausible sample from that mixture component. ◇

From the definition of $r_{nk}$ in (12.18) it is clear that $[r_{n1}, \dots, r_{nK}]^\top$ is a probability vector, i.e., $\sum_k r_{nk} = 1$ with $r_{nk} \geqslant 0$. This probability vector distributes probability mass among the $K$ mixture component, and, intuitively, every $r_{nk}$ expresses the probability that $\boldsymbol{x}_n$ has been generated by the $k$th mixture component.

We now solve for $\boldsymbol{\mu}_k$ so that $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}_k} = \boldsymbol{0}^\top$ and obtain

$$\sum_{n=1}^{N} r_{nk} \boldsymbol{x}_n = \sum_{n=1}^{N} r_{nk} \boldsymbol{\mu}_k \iff \boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} r_{nk} \boldsymbol{x}_n}{\boxed{\sum_{n=1}^{N} r_{nk}}} = \frac{1}{\boxed{N_k}} \sum_{n=1}^{N} r_{nk} \boldsymbol{x}_n \,, \tag{12.26}$$

where we defined

$$N_k := \sum_{n=1}^{N} r_{nk} \,. \tag{12.27}$$

as the total responsibility of the $k$th mixture component across the entire dataset. This concludes the proof of Theorem 12.1. □

Intuitively, (12.17) can be interpreted of a Monte-Carlo estimate of the mean of weighted data points $\boldsymbol{x}_n$ where every $\boldsymbol{x}_n$ is weighted by the

©2018 Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong. To be published by Cambridge University Press.

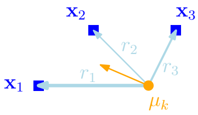
responsibility $r_{nk}$ of the $k$th cluster for $\boldsymbol{x}_n$. Therefore, the mean $\boldsymbol{\mu}_k$ is pulled toward a data point $\boldsymbol{x}_n$ with strength given by $r_{nk}$. Intuitively, the means are pulled stronger toward data points for which the corresponding mixture component has a high responsibility, i.e., a high likelihood. Figure 12.4 illustrates this. We can also interpret the mean update in (12.17) as the expected value of all data points under the distribution given by

$$\boldsymbol{r}_k := [r_{1k}, \dots, r_{Nk}]^\top / N_k\,, \tag{12.28}$$

which is a normalized probability vector, i.e.,

$$\boldsymbol{\mu}_k \leftarrow \mathbb{E}_{\boldsymbol{r}_k}[\boldsymbol{X}]\,. \tag{12.29}$$

**Figure 12.4** Update of the mean parameter of mixture component in a GMM. The mean $\boldsymbol{\mu}$ is being pulled toward individual data points with the weights given by the corresponding responsibilities. The mean update is then a weighted average of the data points.




### Example (Responsibilities)

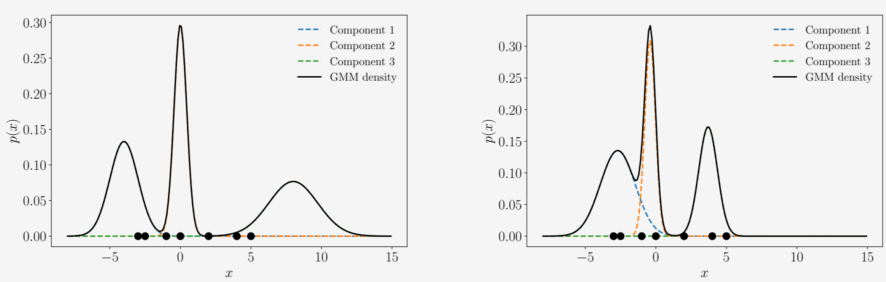
For our example from Figure 12.3 we compute the responsibilities $r_{nk}$

$$\begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 \\ 0.057 & 0.943 & 0.0 \\ 0.001 & 0.999 & 0.0 \\ 0.0 & 0.066 & 0.934 \\ 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix} \in \mathbb{R}^{N\times K}\,. \tag{12.30}$$

Here, the $n$th row tells us the responsibilities of all mixture components for $x_n$. The sum of all $K$ responsibilities for a data point (sum of every row) is 1. The $k$th column gives us an overview of the responsibility of the $k$th mixture component. We can see that the third mixture component (third column) is not responsible for any of the first four data points, but takes much responsibility of the remaining data points. The sum of all entries of a column gives us the values $N_k$, i.e., the total responsibility of the $k$th mixture component. In our example, we get $N_1 = 2.057, N_2 = 2.009, N_3 = 2.934$.

### Example (Mean Updates)

**Figure 12.5** Effect of updating the mean values in a GMM. (a) GMM before updating the mean values; (b) GMM after updating the mean values $\mu_k$ while retaining the variances and mixture weights.




(a) GMM density and individual components prior to updating the mean values.

(b) GMM density and individual components after updating the mean values.

In our example from Figure 12.3, the mean values are updated as follows:

$$\mu_1 : -4 \to -2.7 \tag{12.31}$$
$$\mu_2 : 0 \to -0.4 \tag{12.32}$$
$$\mu_3 : 8 \to 3.7 \tag{12.33}$$

Here, we see that the means of the first and third mixture component move toward the regime of the data, whereas the mean of the second component does not change so dramatically. Figure 12.5 illustrates this change, where Figure 12.6(a) shows the GMM density prior to updating the means and Figure 12.6(b) shows the GMM density after updating the mean values $\mu_k$.

The update of the mean parameters in (12.17) look fairly straightforward. However, note that the responsibilities $r_{nk}$ are a function of $\pi_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j$ for all $j = 1, \dots, K$, such that the updates in (12.17) depend on all parameters of the GMM, and a closed-form solution, which we obtained for linear regression in Section 9.2 or PCA in Section 11 cannot be obtained.

**Theorem 12.2** (Updates of the GMM Covariances)**.** *The update of the covariance parameters $\boldsymbol{\Sigma}_k$, $k = 1, \dots, K$ of the GMM is given by*

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top , \tag{12.34}$$

*where $r_{nk}$ and $N_k$ are defined in* (12.18) *and* (12.19)*, respectively.*

*Proof* To proof Theorem 12.2 our approach is to compute the partial derivatives of the log-likelihood $\mathcal{L}$ with respect to the covariances $\boldsymbol{\Sigma}_k$, set them to $\boldsymbol{0}$ and solve for $\boldsymbol{\Sigma}_k$. We start with our general approach

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\Sigma}_k} = \sum_{n=1}^{N} \frac{\partial \log p(\boldsymbol{x}_n | \boldsymbol{\theta})}{\partial \boldsymbol{\Sigma}_k} = \sum_{n=1}^{N} \frac{1}{p(\boldsymbol{x}_n | \boldsymbol{\theta})} \frac{\partial p(\boldsymbol{x}_n | \boldsymbol{\theta})}{\partial \boldsymbol{\Sigma}_k} . \tag{12.35}$$

We already know $1/p(\boldsymbol{x}_n | \boldsymbol{\theta})$ from (12.16). To obtain the remaining partial derivative $\partial p(\boldsymbol{x}_n | \boldsymbol{\theta}) / \partial \boldsymbol{\Sigma}_k$, we write down the definition of the Gaussian distribution $p(\boldsymbol{x}_n | \boldsymbol{\theta})$, see (12.9), and drop all terms but the $k$th. We then obtain

$$\frac{\partial p(\boldsymbol{x}_n | \boldsymbol{\theta})}{\partial \boldsymbol{\Sigma}_k} = \frac{\partial}{\partial \boldsymbol{\Sigma}_k} \left( (2\pi)^{-\frac{D}{2}} |\boldsymbol{\Sigma}_k|^{-\frac{1}{2}} \exp\left(-\tfrac{1}{2} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)\right) \right) \tag{12.36}$$

$$= \pi_k (2\pi)^{-\frac{D}{2}} \left[ \frac{\partial}{\partial \boldsymbol{\Sigma}_k} |\boldsymbol{\Sigma}_k|^{-\frac{1}{2}} \exp\left(-\tfrac{1}{2} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)\right) \right. \tag{12.37}$$

©2018 Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong. To be published by Cambridge University Press.

$$+|\boldsymbol{\Sigma}_k|^{-\frac{1}{2}}\frac{\partial}{\partial\boldsymbol{\Sigma}_k}\exp\left(-\tfrac{1}{2}(\boldsymbol{x}_n-\boldsymbol{\mu}_k)^\top\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_n-\boldsymbol{\mu}_k)\right)\Big].\tag{12.38}$$

We now use the identities

$$\frac{\partial}{\partial\boldsymbol{\Sigma}_k}|\boldsymbol{\Sigma}_k|^{-\frac{1}{2}}=-\frac{1}{2}|\boldsymbol{\Sigma}_k|^{-\frac{1}{2}}\boldsymbol{\Sigma}_k^{-1},\tag{12.39}$$

$$\frac{\partial}{\partial\boldsymbol{\Sigma}_k}\left(-\frac{1}{2}(\boldsymbol{x}_n-\boldsymbol{\mu}_k)^\top\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_n-\boldsymbol{\mu}_k)\right)=\frac{1}{2}\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_n-\boldsymbol{\mu}_k)(\boldsymbol{x}_n-\boldsymbol{\mu}_k)^\top\boldsymbol{\Sigma}_k^{-1}\tag{12.40}$$

and obtain (after some re-arranging) the desired partial derivative required in (12.35) as

$$\begin{aligned}\frac{\partial p(\boldsymbol{x}_n|\boldsymbol{\theta})}{\partial\boldsymbol{\Sigma}_k}&=\pi_k\mathcal{N}\big(\boldsymbol{x}_n\,|\,\boldsymbol{\mu}_k,\,\boldsymbol{\Sigma}_k\big)\\&\quad\times\left[-\tfrac{1}{2}(\boldsymbol{\Sigma}_k^{-1}-\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_n-\boldsymbol{\mu}_k)(\boldsymbol{x}_n-\boldsymbol{\mu}_k)^\top\boldsymbol{\Sigma}_k^{-1})\right].\end{aligned}\tag{12.41}$$

Putting everything together, the partial derivative of the log-likelihood with respect to $\boldsymbol{\Sigma}_k$ is given by

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\Sigma}_k}=\sum_{n=1}^N\frac{\partial\log p(\boldsymbol{x}_n|\boldsymbol{\theta})}{\partial\boldsymbol{\Sigma}_k}=\sum_{n=1}^N\frac{1}{p(\boldsymbol{x}_n|\boldsymbol{\theta})}\frac{\partial p(\boldsymbol{x}_n|\boldsymbol{\theta})}{\partial\boldsymbol{\Sigma}_k}\tag{12.42}$$

$$\begin{aligned}&=\sum_{n=1}^N\underbrace{\frac{\pi_k\mathcal{N}\big(\boldsymbol{x}_n\,|\,\boldsymbol{\mu}_k,\,\boldsymbol{\Sigma}_k\big)}{\sum_{j=1}^K\pi_j\mathcal{N}\big(\boldsymbol{x}_n\,|\,\boldsymbol{\mu}_j,\,\boldsymbol{\Sigma}_j\big)}}_{=r_{nk}}\\&\quad\times\left[-\tfrac{1}{2}(\boldsymbol{\Sigma}_k^{-1}-\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_n-\boldsymbol{\mu}_k)(\boldsymbol{x}_n-\boldsymbol{\mu}_k)^\top\boldsymbol{\Sigma}_k^{-1})\right]\end{aligned}\tag{12.43}$$

$$=-\frac{1}{2}\sum_{n=1}^N r_{nk}(\boldsymbol{\Sigma}_k^{-1}-\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_n-\boldsymbol{\mu}_k)(\boldsymbol{x}_n-\boldsymbol{\mu}_k)^\top\boldsymbol{\Sigma}_k^{-1})\tag{12.44}$$

$$=-\frac{1}{2}\boldsymbol{\Sigma}_k^{-1}\underbrace{\sum_{n=1}^N r_{nk}}_{=N_k}+\frac{1}{2}\boldsymbol{\Sigma}_k^{-1}\left(\sum_{n=1}^N r_{nk}(\boldsymbol{x}_n-\boldsymbol{\mu}_k)(\boldsymbol{x}_n-\boldsymbol{\mu}_k)^\top\right)\boldsymbol{\Sigma}_k^{-1}.\tag{12.45}$$

We see that the responsibilities $r_{nk}$ also appear in this partial derivative. Setting this partial derivative to $\boldsymbol{0}$, we obtain the necessary optimality condition

$$N_k\boldsymbol{\Sigma}_k^{-1}=\boldsymbol{\Sigma}_k^{-1}\left(\sum_{n=1}^N r_{nk}(\boldsymbol{x}_n-\boldsymbol{\mu}_k)(\boldsymbol{x}_n-\boldsymbol{\mu}_k)^\top\right)\boldsymbol{\Sigma}_k^{-1}\tag{12.46}$$

$$\iff N_k\boldsymbol{I}=\left(\sum_{n=1}^N r_{nk}(\boldsymbol{x}_n-\boldsymbol{\mu}_k)(\boldsymbol{x}_n-\boldsymbol{\mu}_k)^\top\right)\boldsymbol{\Sigma}_k^{-1}\tag{12.47}$$

$$\iff \boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top \,, \tag{12.48}$$

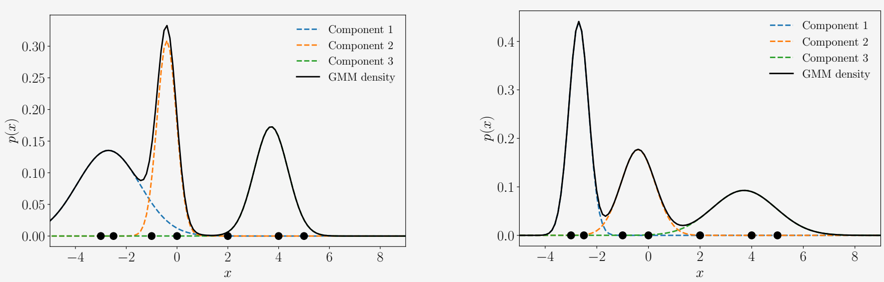
which gives us a simple update rule for $\boldsymbol{\Sigma}_k$ for $k = 1, \dots, K$ and proves Theorem 12.2. $\square$

Similar to the update of $\boldsymbol{\mu}_k$ in (12.17), we can interpret the update of the covariance in (12.34) as an expected value

$$\mathbb{E}_{\boldsymbol{r}_k}[\tilde{\boldsymbol{X}}_k^\top \tilde{\boldsymbol{X}}_k] \tag{12.49}$$

where $\tilde{\boldsymbol{X}}_k := [\boldsymbol{x}_1 - \boldsymbol{\mu}_k, \dots, \boldsymbol{x}_N - \boldsymbol{\mu}_k]^\top$ is the data matrix $\boldsymbol{X}$ centered at $\boldsymbol{\mu}_k$, and $\boldsymbol{r}_k$ is the probability vector defined in (12.28).

**Example (Variance Updates)**




(a) GMM density and individual components prior to updating the variances.

(b) GMM density and individual components after updating the variances.

**Figure 12.6** Effect of updating the variances in a GMM. (a) GMM before updating the variances; (b) GMM after updating the variances while retaining the means and mixture weights.

In our example from Figure 12.3, the variances are updated as follows:

$$\sigma_1^2 : 1 \to 0.14 \tag{12.50}$$
$$\sigma_2^2 : 0.2 \to 0.44 \tag{12.51}$$
$$\sigma_3^2 : 3 \to 1.53 \tag{12.52}$$

Here, we see that the variances of the first and third component shrink significantly, the variance of the second component increases slightly.

Figure 12.6 illustrates this setting. Figure 12.7(a) is identical (but zoomed in) to Figure 12.6(b) and shows the GMM density and its individual components prior to updating the variances. Figure 12.7(b) shows the GMM density after updating the variances.

Similar to the update of the mean parameters, we can interpret (12.34) as a Monte-Carlo estimate of the weighted covariance of data points $\boldsymbol{x}_n$ associated with the $k$th mixture component, where the weights are the responsibilities $r_{nk}$.

As for the updates of the mean parameters, this update depends on all

©2018 Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong. To be published by Cambridge University Press.

$\pi_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j, j = 1, \ldots, K$, through the responsibilities $r_{nk}$, which prohibits a closed-form solution.

**Theorem 12.3** (Update of the GMM Mixture Weights)**.** *The mixture weights of the GMM are updated as*

$$\pi_k = \frac{N_k}{N} \tag{12.53}$$

*for $k = 1, \ldots, K$, where $N$ is the number of data points and $N_k$ is defined in* (12.19).

*Proof* To find the partial derivative of the log-likelihood with respect to the weight parameters $\pi_k, k = 1, \ldots, K$ we take the constraint $\sum_k \pi_k = 1$ into account by using Lagrange multipliers (see Section 7.2). The Lagrangian is

$$L = \mathcal{L} + \lambda \left( \sum_{k=1}^{K} \pi_k - 1 \right) \tag{12.54}$$

$$= \sum_{n=1}^{N} \log \sum_{k=1}^{K} \pi_k \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big) + \lambda \left( \sum_{k=1}^{K} \pi_k - 1 \right) \tag{12.55}$$

where $\mathcal{L}$ is the log-likelihood from (12.10) and the second term encodes for the equality constraint that all the mixture weights need to sum up to 1. We obtain the partial derivatives with respect to $\pi_k$ and the Lagrange multiplier $\lambda$

$$\frac{\partial L}{\partial \pi_k} = \sum_{n=1}^{N} \frac{\mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)}{\sum_{j=1}^{K} \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_j,\, \boldsymbol{\Sigma}_j\big)} + \lambda = \frac{N_k}{\pi_k} + \lambda\,, \tag{12.56}$$

$$\frac{\partial L}{\partial \lambda} = \sum_{k=1}^{K} \pi_k - 1\,. \tag{12.57}$$

Setting both partial derivatives to **0** (necessary condition for optimum) yields the system of equations

$$\pi_k = -\frac{N_k}{\lambda}\,, \tag{12.58}$$

$$1 = \sum_{k=1}^{K} \pi_k\,. \tag{12.59}$$

Using (12.59) in (12.58) and solving for $\pi_k$, we obtain

$$\sum_{k=1}^{K} \pi_k = 1 \iff -\sum_{k=1}^{K} \frac{N_k}{\lambda} = 1 \iff -\frac{N}{\lambda} = 1 \iff \lambda = -N\,. \tag{12.60}$$

This allows us to substitute $-N$ for $\lambda$ in (12.58) to obtain

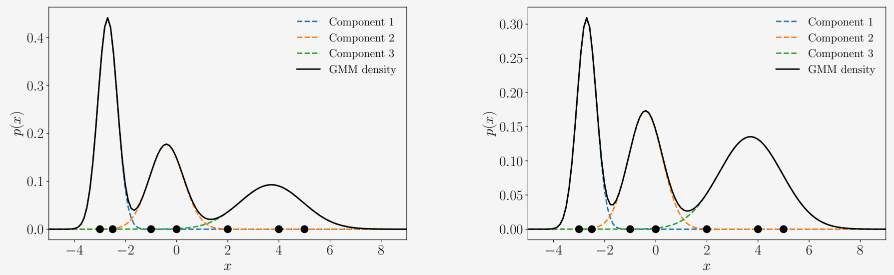
$$\pi_k = \frac{N_k}{N}, \tag{12.61}$$

which gives us the update for the weight parameters $\pi_k$ and proves Theorem 12.3. $\square$

We can identify the mixture weight in (12.53) as the ratio of the total responsibility of the $k$th cluster and the number of data points. Since $N = \sum_k N_k$ the number of data points can also be interpreted as the total responsibility of all mixture components together, such that $\pi_k$ is the relative importance of the $k$th mixture component for the dataset.

*Remark.* Since $N_k = \sum_{i=1}^{N} r_{nk}$, the update equation (12.53) for the mixture weights $\pi_k$ also depends on all $\pi_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j, j = 1, \ldots, K$ via the responsibilities $r_{nk}$. $\diamondsuit$

**Example (Weight Parameter Updates)**




(a) GMM density and individual components prior to updating the mixture weights.

(b) GMM density and individual components after updating the mixture weights.

**Figure 12.7** Effect of updating the mixture weights in a GMM. (a) GMM before updating the mixture weights; (b) GMM after updating the mixture weights while retaining the means and variances. Note the different scalings of the vertical axes.

In our running example from Figure 12.3, the mixture weights are updated as follows:

$$\pi_1 : \tfrac{1}{3} \to 0.29 \tag{12.62}$$
$$\pi_2 : \tfrac{1}{3} \to 0.29 \tag{12.63}$$
$$\pi_3 : \tfrac{1}{3} \to 0.42 \tag{12.64}$$

Here we see that the third component gets more weight/importance, while the other components become slightly less important. Figure 12.7 illustrates the effect of updating the mixture weights. Figure 12.8(a) is identical to Figure 12.7(b) and shows the GMM density and its individual components prior to updating the mixture weights. Figure 12.8(b) shows the GMM density after updating the mixture weights.

Overall, having updated the means, the variances and the weights once,

©2018 Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong. To be published by Cambridge University Press.

we obtain the GMM shown in Figure 12.8(b). Compared with the initialization shown in Figure 12.3, we can see that the parameter updates caused the GMM density to shift some of its mass toward the data points.

After updating the means, variances and weights once, the GMM fit in Figure 12.8(b) is already remarkably better than its initialization from Figure 12.3. This is also evidenced by the log-likelihood values, which increased from $28.3$ (initialization) to $14.4$ (after one complete update cycle).

## 12.3 EM Algorithm

Unfortunately, the updates in (12.17), (12.34), and (12.53) do not constitute a closed-form solution for the updates of the parameters $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k$ of the mixture model because the responsibilities $r_{nk}$ depend on those parameters in a complex way. However, the results suggest a simple *iterative scheme* for finding a solution to the parameters estimation problem via maximum likelihood. The Expectation Maximization algorithm (*EM algorithm*) was proposed by Dempster et al. (1977) and is a general iterative scheme for learning parameters (maximum likelihood or MAP) in mixture models and, more generally, latent-variable models.

EM algorithm

In our example of the Gaussian mixture model, we choose initial values for $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k$ and alternate until convergence between

- *E-step:* Evaluate the responsibilities $r_{nk}$ (posterior probability of data point $i$ belonging to mixture component $k$)
- *M-step:* Use the updated responsibilities to re-estimate the parameters $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k$

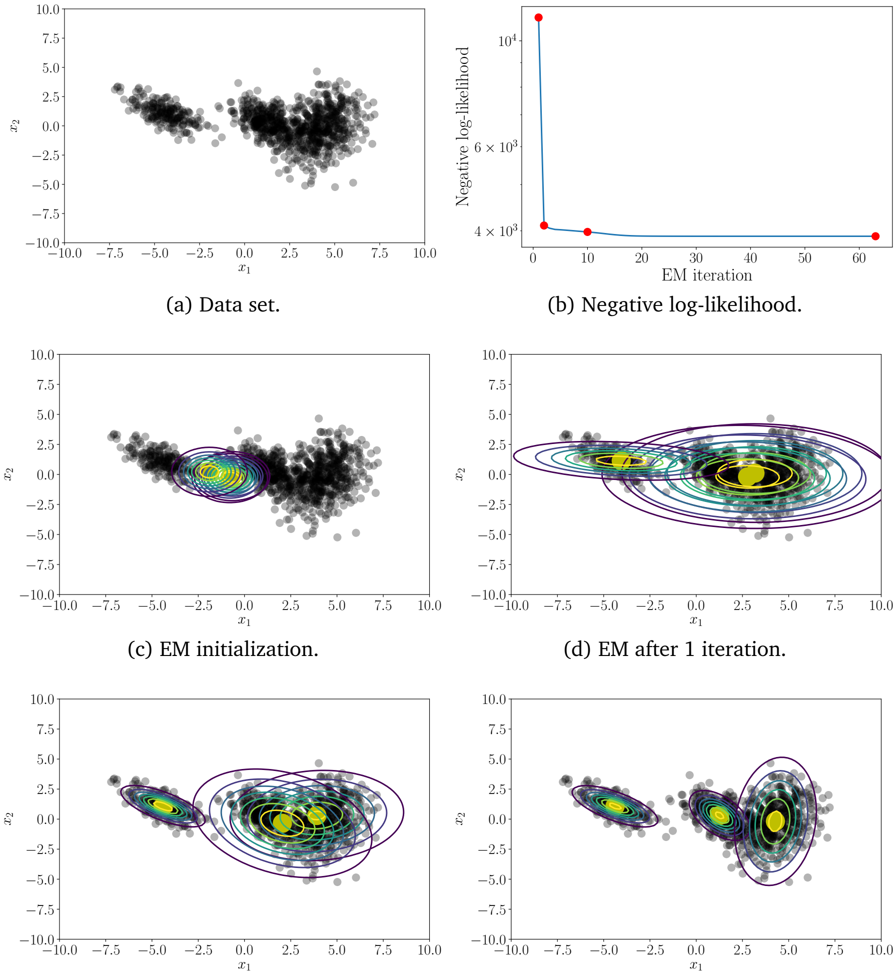
Every step in the EM algorithm increases the log-likelihood function (Neal and Hinton, 1999). For convergence, we can check the log-likelihood or the parameters directly. A concrete instantiation of the EM algorithm for estimating the parameters of a GMM is as follows:

1. Initialize $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k$
2. *E-step:* Evaluate responsibilities $r_{nk}$ for every data point $\boldsymbol{x}_n$ using current parameters $\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$:

$$r_{nk} = \frac{\pi_k \mathcal{N}\big(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)}{\sum_j \pi_j \mathcal{N}\big(\boldsymbol{x}_n \mid \boldsymbol{\mu}_j,\, \boldsymbol{\Sigma}_j\big)} \tag{12.65}$$

3. *M-step:* Re-estimate parameters $\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$ using the current responsibilities $r_{nk}$ (from E-step):

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} \boldsymbol{x}_n \tag{12.66}$$




(a) Data set. (b) Negative log-likelihood.

(c) EM initialization. (d) EM after 1 iteration.

(e) EM after 10 iterations (f) EM after 63 iterations

**Figure 12.8** Illustration of the EM algorithm for fitting a Gaussian mixture model with three components to a two-dimensional data set. (a) Data set; (b) Negative log likelihood (lower is better) as a function of the EM iterations. The red dots indicate the iterations for which the corresponding GMM fits are shown in (c)–(f). The yellow discs indicate the mean of the Gaussian distribution.

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k}\sum_{n=1}^{N} r_{nk}(\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^\top \tag{12.67}$$

$$\pi_k = \frac{N_k}{N} \tag{12.68}$$

**Example (GMM Fit)**

When we run EM on our example from Figure 12.3, we obtain the final result shown in Figure 12.9(a) after five iterations, and Figure 12.9(b) shows how the negative log-likelihood evolves as a function of the EM iterations. The final GMM is given as

$$\begin{aligned} p(x) = {} & 0.29\mathcal{N}\big(x \mid -2.75,\ 0.06\big) + 0.28\mathcal{N}\big(x \mid -0.50,\ 0.25\big) \\ & + 0.43\mathcal{N}\big(x \mid 3.64,\ 1.63\big)\,. \end{aligned} \tag{12.69}$$

©2018 Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong. To be published by Cambridge University Press.

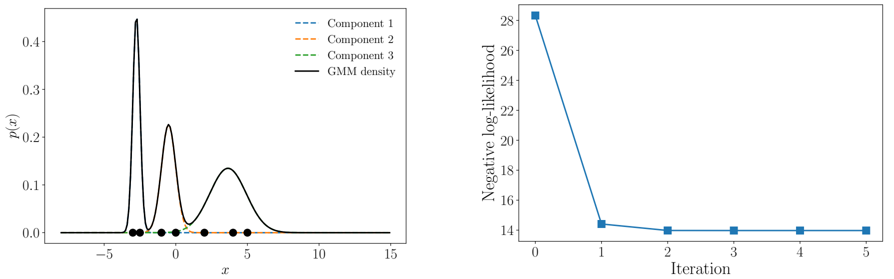



**Figure 12.9** EM algorithm applied to the GMM from Figure 12.2.

(a) Final GMM fit. After five iterations, the EM algorithm converges and returns this mixture density.

(b) Negative log-likelihood as a function of the EM iterations.

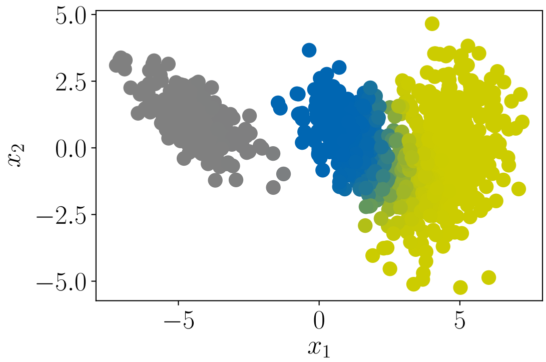



**Figure 12.10** Data set colored according to the responsibilities of the mixture components when EM converges. While a single mixture component is clearly responsible for the data on the left, the overlap of the two data clusters on the right could have been generated by two mixture components.

Figure 12.8 illustrates a few steps of the EM algorithm when applied to the two-dimensional dataset shown in Figure 12.1 with $K = 3$ mixture components.

Figure 12.10 visualizes the final responsibilities of the mixture components for the data points. It becomes clear that there are data points that cannot be uniquely assigned to a single (either blue or yellow) component, such that the responsibilities of these two clusters for those points are around $0.5$.

## 12.4 Latent Variable Perspective

We can look at the GMM from the perspective of a latent variable model. Let us assume that we have a mixture model with $K$ components and that a data point $\boldsymbol{x}$ is generated by exactly one component. We can then use an indicator variable $z_k \in \{0, 1\}$ that indicates whether the $k$th mixture component generated that data point so that

$$p(\boldsymbol{x}|z_k = 1) = \mathcal{N}\big(\boldsymbol{x} \,|\, \boldsymbol{\mu}_k, \, \boldsymbol{\Sigma}_k\big) \,. \tag{12.70}$$

We define $\boldsymbol{z} := [z_1, \dots, z_K]^\top \in \mathbb{R}^K$ as a vector consisting of $K-1$ many 0s and exactly one 1. Because of this, it also holds that $\sum_{k=1}^{K} z_k = 1$. Therefore, $\boldsymbol{z}$ is a *one-hot encoding* (also: 1-*of*-$K$ *representation*). This allows us to write the conditional distribution as

one-hot encoding

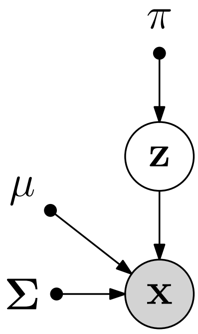
1-of-$K$ representation

$$p(\boldsymbol{x}|\boldsymbol{z}) = \prod_{k=1}^{K} \mathcal{N}\big(\boldsymbol{x} \mid \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)^{z_k}, \tag{12.71}$$

where $z_k \in \{0, 1\}$. Thus far, we assumed that the indicator variables $z_k$ are known. However, in practice, this is not the case, and we place a prior distribution on $\boldsymbol{z}$.

In the following, we will discuss the prior $p(\boldsymbol{z})$, the marginal $p(\boldsymbol{x}, \boldsymbol{z})$ and the posterior $p(\boldsymbol{z}|\boldsymbol{x})$ for the case of observing a single data point $\boldsymbol{x}$ (the corresponding graphical model is shown in Figure 12.11) before extending the concepts to the general case where the dataset consists of $N$ data points.

**Figure 12.11** Graphical model for a GMM with a single data point.




### 12.4.1 Prior

Given that we do not know which mixture component generated the data point, we treat the indicators $\boldsymbol{z}$ as a latent variable and place a prior $p(\boldsymbol{z})$ on it. Then the prior $p(z_k = 1) = \pi_k$ describes the probability that the $k$th mixture component generated data point $\boldsymbol{x}$. To ensure that our probability distribution is normalized, we require that $\sum_{k=1}^{K} \pi_k = 1$. We summarize the prior $p(\boldsymbol{z}) = \boldsymbol{\pi}$ in the probability vector $\boldsymbol{\pi} = [\pi_1, \dots, \pi_K]^\top$. Because of the one-hot encoding of $\boldsymbol{z}$, we can write the prior in a less obvious form

$$p(\boldsymbol{z}) = \prod_{k=1}^{K} \pi_k^{z_k}, \quad z_k \in \{0, 1\}, \tag{12.72}$$

but this form will become handy later on.

*Remark* (Sampling from a GMM). The construction of this latent variable model lends itself to a very simple sampling procedure to generate data:

1. Sample $\boldsymbol{z} \sim p(\boldsymbol{z}|\boldsymbol{\pi})$
2. Sample $\boldsymbol{x} \sim p(\boldsymbol{x}|\boldsymbol{z})$

In the first step, we would select a mixture component at random according to $\boldsymbol{\pi}$; in the second step we would draw a sample from a single mixture component. ♢

©2018 Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong. To be published by Cambridge University Press.

### 12.4.2 Marginal

If we marginalize out the latent variables $\boldsymbol{z}$ (by summing over all possible one-hot encodings), we obtain the marginal distribution

$$p(\boldsymbol{x}) = \sum_{\boldsymbol{z}} p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\pi}) = \sum_{\boldsymbol{z}} \prod_{k=1}^{K} \left(\pi_k \mathcal{N}\big(\boldsymbol{x} \mid \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)\right)^{z_k} \tag{12.73}$$

$$= \pi_1 \mathcal{N}\big(\boldsymbol{x} \mid \boldsymbol{\mu}_1,\, \boldsymbol{\Sigma}_1\big) + \cdots + \pi_K \mathcal{N}\big(\boldsymbol{x} \mid \boldsymbol{\mu}_K,\, \boldsymbol{\Sigma}_K\big) \tag{12.74}$$

$$= \sum_{k=1}^{K} \pi_k \mathcal{N}\big(\boldsymbol{x} \mid \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)\,, \tag{12.75}$$

The marginal distribution $p(\boldsymbol{x})$ is a Gaussian mixture model.

which is identical to the GMM we introduced in (12.3). Therefore, the latent variable model with latent indicators $z_k$ is an equivalent way of thinking about a Gaussian mixture model.

### 12.4.3 Posterior

Let us have a brief look at the posterior distribution on the latent variable $\boldsymbol{z}$. According to Bayes' theorem, the posterior is

$$p(\boldsymbol{z}|\boldsymbol{x}) = \frac{p(\boldsymbol{x}, \boldsymbol{z})}{p(\boldsymbol{x})}\,, \tag{12.76}$$

where $p(\boldsymbol{x})$ is given in (12.75). With (12.71) and (12.72) we get the joint distribution as

$$p(\boldsymbol{x}, \boldsymbol{z}) = p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z}) = \prod_{k=1}^{K} \mathcal{N}\big(\boldsymbol{x} \mid \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)^{z_k} \prod_{k=1}^{K} \pi_k^{z_k} \tag{12.77}$$

$$= \prod_{k=1}^{K} \left(\pi_k \mathcal{N}\big(\boldsymbol{x} \mid \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)\right)^{z_k}\,. \tag{12.78}$$

Here, we identify $p(\boldsymbol{x}|\boldsymbol{z}) = \prod_{k=1}^{K} \mathcal{N}\big(\boldsymbol{x} \mid \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)^{z_k}$ as the likelihood. This yields the posterior distribution for the $k$th indicator variable $z_k$

$$p(z_k = 1|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|z_k = 1)p(z_k = 1)}{\sum_{j=1}^{K} p(z_j = 1)p(\boldsymbol{x}|z_j = 1)} = \frac{\pi_k \mathcal{N}\big(\boldsymbol{x} \mid \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\big(\boldsymbol{x} \mid \boldsymbol{\mu}_j,\, \boldsymbol{\Sigma}_j\big)}\,, \tag{12.79}$$

which we identify as the responsibility of the $k$th mixture component for data point $\boldsymbol{x}$. Note that we omitted the explicit conditioning on the GMM parameters $\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$ where $k = 1, \ldots, K$.

### 12.4.4 Extension to a Full Dataset

Thus far, we only discussed the case where the dataset consists only of a single data point $\boldsymbol{x}$. However, the concepts can be directly extended to the case of $N$ data points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$, which we collect in the data matrix $\boldsymbol{X}$.

Every data point $\boldsymbol{x}_n$ possesses its own latent variable

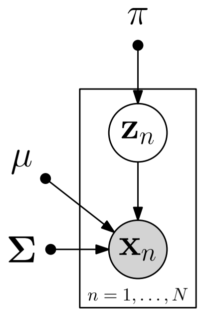
$$\boldsymbol{z}_n = [z_{n1}, \ldots, z_{nK}]^\top \in \mathbb{R}^K \,. \tag{12.80}$$

Previously (when we only considered a single data point $\boldsymbol{x}$) we omitted the index $n$, but now this becomes important. We collect all of these latent variables in the matrix $\boldsymbol{Z}$. We share the same prior distribution $\boldsymbol{\pi}$ across all data points. The corresponding graphical model is shown in Figure 12.12, where we use the plate notation.

The likelihood $p(\boldsymbol{X}|\boldsymbol{Z})$ factorizes over the data points, such that the joint distribution (12.78) is given as

$$p(\boldsymbol{X}, \boldsymbol{Z}) = p(\boldsymbol{X}|\boldsymbol{Z})p(\boldsymbol{Z}) = \prod_{n=1}^{N}\prod_{k=1}^{K}\big(\pi_k \mathcal{N}\big(\boldsymbol{x}_n \,|\, \boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k\big)\big)^{z_{nk}} \,. \tag{12.81}$$

**Figure 12.12** Graphical model for a GMM with $N$ data points.




Generally, the posterior distribution $p(z_k = 1|\boldsymbol{x}_n)$ is the probability that the $k$th mixture component generated data point $\boldsymbol{x}_n$ and corresponds to the responsibility $r_{nk}$ we introduced in (12.18). Now, the responsibilities also have not only an intuitive but also a mathematically justified interpretation as posterior probabilities.

### ***12.4.5 EM Algorithm Revisited***

The EM algorithm that we introduced as an iterative scheme for maximum likelihood estimation can be derived in a principled way from the latent variable perspective. Given a current setting $\boldsymbol{\theta}^{(t)}$ of model parameters, the E-step calculates the expected log-likelihood

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \mathbb{E}_{\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta}^{(t)}}[\log p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})] = \int \log p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{(t)})d\boldsymbol{Z} \,, \tag{12.82}$$

where the expectation of the log-joint distribution of latent variables $\boldsymbol{Z}$ and observations $\boldsymbol{X}$ is taken with respect to the posterior $p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{(t)})$ of the latent variables. The M-step selects an updated set of model parameters $\boldsymbol{\theta}^{(t+1)}$ by maximizing (12.82).

Although an EM iteration does increase the log-likelihood, there are no guarantees that EM converges to the maximum likelihood solution. It is possible that the EM algorithm converges to a local maximum of the log-likelihood. Different initializations of the parameters $\boldsymbol{\theta}$ could be used in multiple EM runs to reduce the risk of ending up in a bad local optimum. We do not go into further details here, but refer to the excellent expositions by Rogers and Girolami (2016) and Bishop (2006).

## 12.5 Further Reading

The GMM can be considered a generative model in the sense that it is straightforward to generate new data using ancestral sampling (Bishop,

©2018 Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong. To be published by Cambridge University Press.

2006). For given GMM parameters $\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$, $k = 1, \dots, K$, we sample an index $k$ from the probability vector $[\pi_1, \dots, \pi_K]^\top$ and then sample a data point $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$. If we repeat this $N$ times, we obtain a dataset that has been generated by a GMM. Figure 12.1 was generated using this procedure.

Throughout this chapter, we assumed that the number of components $K$ is known. In practice, this is often not the case. However, we could use cross-validation, as discussed in Section 8.2, to find good models.

Gaussian mixture models are closely related to the $K$-means clustering algorithm. $K$-means also uses the EM algorithm to assign data points to clusters. If we treat the means in the GMM as cluster centers and ignore the covariances, we arrive at $K$-means. As also nicely described by MacKay (2003), $K$-means makes a “hard” assignments of data points to cluster centers $\boldsymbol{\mu}_k$, whereas a GMM makes a “soft” assignment via the responsibilities.

We only touched upon the latent variable perspective of GMMs and the EM algorithm. Note that EM can be used for parameter learning in general latent variable models, e.g., nonlinear state-space models (Ghahramani and Roweis, 1999; Roweis and Ghahramani, 1999) and for reinforcement learning as discussed by Barber (2012). Therefore, the latent variable perspective of a GMM is useful to derive the corresponding EM algorithm in a principled way (Bishop, 2006; Barber, 2012; Murphy, 2012).

We only discussed maximum likelihood estimation (via the EM algorithm) for finding GMM parameters. The standard criticisms of maximum likelihood also apply here:

- As in linear regression, maximum likelihood can suffer from severe overfitting. In the GMM case, this happens when the mean of a mixture component is identical to a data point and the covariance tends to $\mathbf{0}$. Then, the likelihood approaches infinity. Bishop (2006) and Barber (2012) discuss this issue in detail.
- We only obtain a point estimate of the parameters $\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$ for $k = 1, \dots, K$, which does not give any indication of uncertainty in the parameter values. A Bayesian approach would place a prior on the parameters, which can be used to obtain a posterior distribution on the parameters. This posterior allows us to compute the model evidence (marginal likelihood), which can be used for model comparison, which gives us a principled way to determine the number of mixture components. Unfortunately, closed-form inference is not possible in this setting because there is no conjugate prior for this model. However, approximations, such as variational inference, can be used to obtain an approximate posterior (Bishop, 2006).

In this chapter, we discussed mixture models for density estimation. There is a plethora of density estimation techniques available. In practice we often use histograms and kernel density estimation.

Histograms

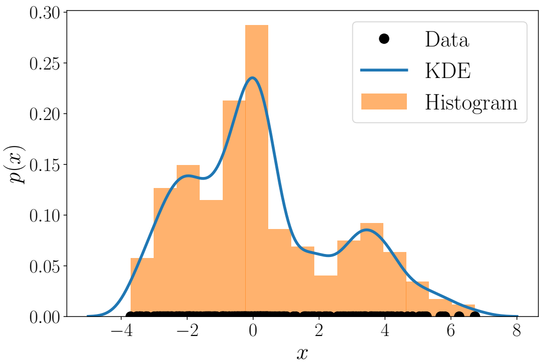



**Figure 12.13** Histogram (orange bars) and kernel density estimation (blue line). The kernel density estimator (with a Gaussian kernel) produces a smooth estimate of the underlying density, whereas the histogram is simply an unsmoothed count measure of how many data points (black) fall into a single bin.

5182 *Histograms* provide a non-parametric way to represent continuous den-
5183 sities and have been proposed by Pearson (1895). A histogram is con-
5184 structed by "binning" the data space and count how many data points fall
5185 into each bin. Then a bar is drawn at the center of each bin, and the height
5186 of the bar is proportional to the number of data points within that bin.

Kernel density estimation

*Kernel density estimation*, independently proposed by Rosenblatt (1956) and Parzen (1962), is a nonparametric way for density estimation. Given $N$ i.i.d. samples, the kernel density estimator represents the underlying distribution as

$$p(\boldsymbol{x}) = \frac{1}{Nh}\sum_{n=1}^{N} k\left(\frac{\boldsymbol{x}-\boldsymbol{x}_n}{h}\right), \tag{12.83}$$

5187 where $k$ is a kernel function, i.e., a non-negative function that integrates
5188 to 1 and $h > 0$ is a smoothing/bandwidth parameter. Note that we place a
5189 kernel on every single data point $\boldsymbol{x}_n$ in the dataset. Commonly used ker-
5190 nel functions are the uniform distribution and the Gaussian distribution.
5191 Kernel density estimates are closely related to histograms, but by choosing
5192 a suitable kernel, we can guarantee smoothness of the density estimate.
5193 Figure 12.13 illustrates the difference between a histogram and a kernel
5194 density estimator (with a Gaussian-shaped kernel) for a given data set of
5195 250 data points.

©2018 Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong. To be published by Cambridge University Press.