CHAPTER

12 BAYESIAN LEARNING: INFERENCE AND THE EM ALGORITHM

CHAPTER OUTLINE

Machine Learning. http://dx.doi.org/10.1016/B978-0-12-801522-3.00012-4

12.1 INTRODUCTION

The Bayesian approach to parameter inference was introduced in Chapter 3. In contrast to other methods for parameter estimation we have covered, the Bayesian method adopts a radically different viewpoint. The unknown set of parameters are treated as random variables instead of as a set of fixed (yet unknown) values. This was a revolutionary idea, at the time it was introduced by Bayes and later on by Laplace, as pointed out in Chapter 3. Even now, after more than two centuries, it may seem strange to assume that a physical phenomenon/mechanism is controlled by a set of random parameters. However, there is a subtle point here. Treating the underlying set of parameters as random variables, $\boldsymbol{\theta}$, we do not really imply a random nature for them. The associated randomness, in terms of the prior distribution $p(\boldsymbol{\theta})$, encapsulates our *uncertainty* about their values, prior to receiving any measurements/observations. Stated differently, the prior distribution represents our *belief* about the different possible values, although only one of them is actually true. From this perspective, probabilities are viewed in a more open-minded way, that is, as measures of uncertainty, as discussed in the beginning of Chapter 2.

Recall that parameter learning from data is an inverse problem. Basically, all we do is to deduce the "causes" (parameters) from the "effects" (observations). Bayes theorem can be seen as an inversion procedure expressed in a probabilistic context. Indeed, given the set of observations, say, $\mathcal{X}$, which are controlled by the unknown set of parameters, we write

$$p(\boldsymbol{\theta}|\mathcal{X}) = \frac{p(\mathcal{X}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{X})}.$$

All that is needed for the above inversion is to have a guess about $p(\boldsymbol{\theta})$. This term has brought a lot of controversy in the statistical community for a number of years. However, once a reasonable guess of the prior is available, a number of advantages associated with the Bayesian approach emerge, compared to the alternative route. The latter embraces methods that view the parameters deterministically as constants of unknown values, and they are also referred to as *frequentist* techniques. The term comes from the more classical view of probabilities as frequencies of occurrence of repeatable events. A typical example of this family of methods is the maximum likelihood approach, which estimates the values of the parameters by maximizing $p(\mathcal{X}|\boldsymbol{\theta})$; its value is solely controlled by the obtained observations in a sequence of experiments.

This is the first of two chapters dedicated to Bayesian learning. We present the main concepts and philosophy behind Bayesian inference. We introduce the expectation-maximization (EM) algorithm and apply it in some typical machine learning parametric modeling tasks, such as regression, mixture modeling, and mixture of experts. Finally, the exponential family of distributions is introduced and the notion of conjugate priors is discussed.

12.2 REGRESSION: A BAYESIAN PERSPECTIVE

The Bayesian inference treatment of the linear regression task was introduced in Chapter 3. In the current chapter, we go beyond the basic definitions and reveal and exploit various possibilities that the Bayesian philosophy offers to the study of this important machine learning task. Let us first summarize the findings of Chapter 3 and then start building upon them.[1]

[1] Recall our adopted notation: random variables and vectors are denoted with roman and their respected measured values/observations with Times Roman fonts.

Recall the (generalized) linear regression task, as it was introduced in previous chapters, that is,

$$\mathrm{y} = \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}) + \eta = \theta_0 + \sum_{k=1}^{K-1} \theta_k \phi_k(\mathbf{x}) + \eta, \tag{12.1}$$

where $\mathrm{y} \in \mathbb{R}$ is the output random variable, $\mathbf{x} \in \mathbb{R}^l$ is the input random vector, $\eta \in \mathbb{R}$ is the noise disturbance, $\boldsymbol{\theta} \in \mathbb{R}^K$ is the unknown parameter vector, and

$$\boldsymbol{\phi}(\mathbf{x}) := [\phi_1(\mathbf{x}), \ldots, \phi_{K-1}(\mathbf{x}), 1]^{\mathrm{T}}$$

where $\phi_k(\cdot),\ k = 1, \ldots, K-1$, are some (fixed) basis functions. As we already know, typical examples of such functions can be the Gaussian function, splines, monomials, and others. We are given a set of N output-input training points, $(y_n, \boldsymbol{x}_n)$, $n = 1, 2, \ldots, N$. In our current setting, we assume that the respective (unobserved) noise values, $\eta_n,\ n = 1, 2, \ldots, N$, are samples of jointly Gaussian distributed random variables with covariance matrix Σ_η, that is,

$$p(\boldsymbol{\eta}) = \frac{1}{(2\pi)^{N/2}|\Sigma_\eta|^{1/2}} \exp\left(-\frac{1}{2}\boldsymbol{\eta}^{\mathrm{T}}\Sigma_\eta^{-1}\boldsymbol{\eta}\right), \tag{12.2}$$

where $\boldsymbol{\eta} = [\eta_1, \eta_2, \ldots, \eta_N]^{\mathrm{T}}$.

12.2.1 THE MAXIMUM LIKELIHOOD ESTIMATOR

The maximum likelihood (ML) method was introduced in Chapter 3. According to the method, the unknown parameter is treated as a deterministic variable $\boldsymbol{\theta}$, which parameterizes the pdf describing the output vector of observations

$$\boldsymbol{y} = \Phi\boldsymbol{\theta} + \boldsymbol{\eta}, \tag{12.3}$$

where

$$\Phi = \begin{bmatrix} \boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_1) \\ \boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_2) \\ \vdots \\ \boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_N) \end{bmatrix}, \tag{12.4}$$

and

$$\boldsymbol{y} = [y_1, y_2, \ldots, y_N]^{\mathrm{T}}.$$

A simple replacement of X with Φ in (3.59) changes the ML estimate to

$$\hat{\boldsymbol{\theta}}_{\mathrm{ML}} = \left(\Phi^{\mathrm{T}}\Sigma_\eta^{-1}\Phi\right)^{-1}\Phi^{\mathrm{T}}\Sigma_\eta^{-1}\boldsymbol{y}. \tag{12.5}$$

For the simple case of uncorrelated noise samples of equal variance σ_η^2 ($\Sigma_\eta = \sigma_\eta^2 I$), Eq. (12.5) becomes identical to the least-squares (LS) solution

$$\hat{\boldsymbol{\theta}}_{\mathrm{ML}} = \left(\Phi^{\mathrm{T}}\Phi\right)^{-1}\Phi^{\mathrm{T}}\boldsymbol{y} = \hat{\boldsymbol{\theta}}_{\mathrm{LS}}. \tag{12.6}$$

A major drawback of the ML approach is that it is vulnerable to overfitting, becase no care is taken for complex models that try to"learn" the specificities of the particular training set, as already discussed in Chapter 3.

12.2.2 THE MAP ESTIMATOR

According to the maximum a posteriori probability (MAP) method, the unknown set of parameters is treated as a random vector Θ and its posterior, for a given set of output observations, $\boldsymbol{y}$, is expressed as

$$p(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\boldsymbol{y})}, \tag{12.7}$$

where $p(\boldsymbol{\theta})$ is the associated prior pdf. We have eliminated from the notation the dependence on $\mathcal{X}$, to make it look simpler. We emphasize that the input set, $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, is considered fixed, so all the randomness associated with $\boldsymbol{y}$ is due to the noise source. Assuming both the prior as well as the conditional pdfs to be Gaussians,[2] that is,

$$p(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\theta}_0, \Sigma_\theta), \tag{12.8}$$

and

$$p(\boldsymbol{y}|\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{y}|\Phi\boldsymbol{\theta}, \Sigma_\eta), \tag{12.9}$$

where (12.2), (12.3) have been used, the posterior $p(\boldsymbol{\theta}|\boldsymbol{y})$ turns out also to be Gaussian with mean vector

$$\boldsymbol{\mu}_{\theta|y} := \mathbb{E}[\Theta|\boldsymbol{y}] = \boldsymbol{\theta}_0 + \left(\Sigma_\theta^{-1} + \Phi^{\mathrm{T}}\Sigma_\eta^{-1}\Phi\right)^{-1}\Phi^{\mathrm{T}}\Sigma_\eta^{-1}\left(\boldsymbol{y} - \Phi\boldsymbol{\theta}_0\right). \tag{12.10}$$

Because the maximum of a Gaussian coincides with its mean, we have that

$$\hat{\boldsymbol{\theta}}_{\mathrm{MAP}} = \mathbb{E}[\Theta|\boldsymbol{y}]. \tag{12.11}$$

In the chapter's appendix, Section 12.9, an analytical proof of (12.10) is provided.[3] It suffices to replace in (12.140) $\boldsymbol{t} \to \boldsymbol{y}$, $\boldsymbol{z} \to \boldsymbol{\theta}$, $A \to \Phi$, $\Sigma_{t|z} \to \Sigma_\eta$, and $\Sigma_z \to \Sigma_\theta$. Note that the MAP estimate is a regularized version of $\hat{\boldsymbol{\theta}}_{\mathrm{ML}}$. Regularization is achieved via $\boldsymbol{\theta}_0$ and Σ_θ, which are imposed by the prior $p(\boldsymbol{\theta})$. If one assumes $\Sigma_\theta = \sigma_\theta^2 I$, $\Sigma_\eta = \sigma_\eta^2 I$, and $\boldsymbol{\theta}_0 = \mathbf{0}$, then (12.10) coincides with the solution of the regularized LS (ridge) regression,[4]

$$\hat{\boldsymbol{\theta}}_{\mathrm{MAP}} = \left(\lambda I + \Phi^{\mathrm{T}}\Phi\right)^{-1}\Phi^{\mathrm{T}}\boldsymbol{y}, \tag{12.12}$$

where we have set $\lambda := \frac{\sigma_\eta^2}{\sigma_\theta^2}$. We already know from Chapter 3 that the value of λ is critical to the performance of the estimator with respect to the mean-square error (MSE) performance. The main issue now becomes how to choose a good value for λ, or equivalently for Σ_θ, Σ_η in the more general case. In practice, the cross-validation method (Chapter 3) is employed; different values of λ are tested and the one that leads to the best MSE (or some other criterion) is selected. However, this is a computationally costly procedure, especially for complex models, where a large number of parameters is involved. Moreover, such a procedure forces us to use only a fraction of the available data for training, to reserve the rest for testing. The reader may wonder why we do not use the training data to optimize with

[2] Because in this chapter many random variables will be involved, we explicitly state the name of the variable to which we refer in $\mathcal{N}(\cdot|\cdot, \cdot)$.

[3] Because the appendix serves the needs of various parts of the book, each time involving different variables, one has to make the necessary notational substitutions.

[4] Recall from Section 3.8, that this is valid if either the data have been centered or the intercept (bias) is involved in the regularizing norm term.

respect to both the unknown parameter $\boldsymbol{\theta}$ as well as the regularization parameters. Let us consider as an example the simpler case of ridge regression, for centered data ($\theta_0 = 0$). The cost function comprises two terms, one that is data-dependent and measures the misfit, whereas the second one depends only on the unknown parameter

$$J(\boldsymbol{\theta}, \lambda) = \|\boldsymbol{y} - \Phi\boldsymbol{\theta}\|^2 + \lambda\|\boldsymbol{\theta}\|^2. \tag{12.13}$$

It is obvious that the only value of λ that leads to the minimum squared error fit *over the training data set* (empirical loss) is $\lambda = 0$. Any other value of λ would result in an estimate of $\boldsymbol{\theta}$ which scores larger values of the squared error term; this is natural, because for $\lambda \neq 0$ the optimization has to take care for the extra regularizing term, too. It is only when *test data sets* are employed, where values of $\lambda \neq 0$ lead to an overall decrease of the mean-square error (not the empirical one).

12.2.3 THE BAYESIAN APPROACH

The Bayesian approach to regression attempts to overcome the previously reported drawbacks, which are associated with the overfitting. All the involved parameters can be estimated on the training set. In this vein, the parameters will be treated as random variables. At the same time, because the main task now becomes that of inferring the pdf that describes the unknown set of parameters, instead of obtaining a single vector estimate, one has more information at her/his disposal. Having said that, it does not mean that Bayesian techniques are necessarily free from cross-validation; this will be needed to assess their overall performance. We will comment further on this in the Remarks at the end of Section 12.3.

As we know, the starting point is the same as that for MAP, and in particular (12.7). However, instead of taking just the maximum of the numerator in (12.7), we will make use of $p(\boldsymbol{\theta}|\boldsymbol{y})$ as a whole. Most of the secrets here lie in the denominator $p(\boldsymbol{y})$, which is basically the normalizing constant,

$$\boxed{p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta}.} \tag{12.14}$$

As we will soon see, there is much more information hidden in $p(\boldsymbol{y})$ that goes beyond the need of just computing $p(\boldsymbol{\theta}|\boldsymbol{y})$. The difficulty with (12.14) is that, in general, the evaluation of the integral cannot be performed analytically. In such cases, one has to resort to approximate techniques, to obtain the required information. To this end, a number of approaches are available, and a large part of this book is dedicated to their study. More specifically, the following methods have been proposed and will be considered:

- The Laplacian approximation method, presented in Section 12.3.
- The variational approximation method, presented in Section 13.2.
- The variational bound approximation method, presented in Section 13.8.
- Monte Carlo techniques for the evaluation of the integral, which are discussed in Chapter 14.
- Message passing algorithms, to be discussed in Chapter 15.

For the case under study in this section, where $p(\boldsymbol{y}|\boldsymbol{\theta})$ and $p(\boldsymbol{\theta})$ are both assumed to be Gaussians, $p(\boldsymbol{y})$ can be evaluated analytically; it turns out that the joint distribution $p(\boldsymbol{y}, \boldsymbol{\theta})$ is also Gaussian and hence the marginal $p(\boldsymbol{y})$ is Gaussian as well. All these are shown in detail in the appendix of Section 12.9, at the end of the chapter. Indeed, if we set in (12.143) and (12.148) of Section 9 $\boldsymbol{z} \to \boldsymbol{\theta}$, $\boldsymbol{t} \to \boldsymbol{y}$, and $A \to \Phi$, it turns out that for the regression model of (12.3) and the prior pdf in (12.8) as well as the noise model of (12.2), we obtain that,

$$p(\boldsymbol{y}) = \mathcal{N}\left(\boldsymbol{y}|\Phi\boldsymbol{\theta}_0, \Sigma_\eta + \Phi\Sigma_\theta\Phi^{\mathrm{T}}\right). \tag{12.15}$$

Moreover, the posterior $p(\boldsymbol{\theta}|\boldsymbol{y})$ is also Gaussian,

$$p(\boldsymbol{\theta}|\boldsymbol{y}) = \mathcal{N}\left(\boldsymbol{\theta}|\boldsymbol{\mu}_{\theta|y}, \Sigma_{\theta|y}\right), \tag{12.16}$$

where $\boldsymbol{\mu}_{\theta|y}$ is given by (12.10) and the covariance matrix results from (12.144), after the appropriate notational substitutions, that is,

$$\Sigma_{\theta|y} = \left(\Sigma_\theta^{-1} + \Phi^{\mathrm{T}}\Sigma_\eta^{-1}\Phi\right)^{-1}. \tag{12.17}$$

The posterior pdf in (12.16) encapsulates our knowledge about θ, after the observations $\boldsymbol{y}$ have been obtained. Hence, our uncertainty about θ has been reduced, which is the main reason that (12.16) is different from the prior pdf in (12.8); the latter represents only our initial guess. The covariance matrix in (12.17) provides the information about our uncertainty with respect to θ. If the Gaussian in (12.16) is very broad around its mean $\boldsymbol{\mu}_{\theta|y}$, it indicates that in spite of the reception of the observations still much uncertainty about θ remains. This can be due (a) to the nature of the problem, for example, high noise variance, as this is conveyed by Σ_η, (b) and/or to the number of observations, N, which may not be enough, (c) and/or to modeling inaccuracies, as this is conveyed by Φ in (12.17). The opposite comments are in order if the posterior pdf is sharply peaked around its mean.

As we have already stated in Chapter 3, the Bayesian philosophy provides the means for a direct inference of the output variable, which in many applications is the quantity of interest; given the input vector, the task is to predict the output. In such cases, estimating a value for the unknown θ is only the means to an end. To formulate the prediction task directly, without involving θ, one has to integrate the contribution of θ. Having learned the posterior $p(\boldsymbol{\theta}|\boldsymbol{y})$, then given a new input vector $\boldsymbol{x}$, for the regression model in (12.1), the conditional pdf of the output variable, y, given the set of observations is written as,

$$p(y|\boldsymbol{x},\boldsymbol{y}) = \int p(y|\boldsymbol{x},\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{y})\,\mathrm{d}\boldsymbol{\theta}. \tag{12.18}$$

Note that we have used $p(y|\boldsymbol{x},\boldsymbol{y},\boldsymbol{\theta}) = p(y|\boldsymbol{x},\boldsymbol{\theta})$ because y is conditionally independent of $\boldsymbol{y}$ given the value of $\boldsymbol{\theta}$. As it has already been stated, strictly speaking, the posterior should have been denoted as $p(\boldsymbol{\theta}|\boldsymbol{y};\mathcal{X})$ to indicate the dependence on the input training samples. However, the dependence on $\mathcal{X}$ has been suppressed to unclutter notation.

In the sequel, and in order to simplify algebra and focus on the concepts, we assume that the noise model in (12.2) is such that $\Sigma_\eta = \sigma_\eta^2 I$ and also $\Sigma_\theta = \sigma_\theta^2 I$ for the prior pdf in (12.8). Then, we have that

$$p(y|\boldsymbol{x},\boldsymbol{\theta}) = \mathcal{N}(y|\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}), \sigma_\eta^2),$$

and (12.17), (12.10) for the posterior covariance matrix and mean, respectively, become

$$\Sigma_{\theta|y} = \left(\frac{1}{\sigma_\theta^2}I + \frac{1}{\sigma_\eta^2}\Phi^{\mathrm{T}}\Phi\right)^{-1}, \tag{12.19}$$

$$\boldsymbol{\mu}_{\theta|y} = \boldsymbol{\theta}_0 + \frac{1}{\sigma_\eta^2}\left(\frac{1}{\sigma_\theta^2}I + \frac{1}{\sigma_\eta^2}\Phi^{\mathrm{T}}\Phi\right)^{-1}\Phi^{\mathrm{T}}\left(\boldsymbol{y} - \Phi\boldsymbol{\theta}_0\right). \tag{12.20}$$

The integration in (12.18) can now be carried out analytically as in (12.133), (12.134), and using (12.147), (12.148) in Section 12.9, with $\boldsymbol{z} \to \boldsymbol{\theta}$, $\boldsymbol{t} \to \boldsymbol{y}$, $A \to \boldsymbol{\phi}^T$, $\boldsymbol{\mu}_z \to \boldsymbol{\mu}_{\theta|y}$, $\Sigma_z \to \Sigma_{\theta|y}$, $\Sigma_{t|z} \to \sigma_\eta^2$, and we obtain that,

$$\boxed{p(y|\boldsymbol{x},\boldsymbol{y}) = \mathcal{N}\left(y|\mu_y, \sigma_y^2\right): \quad \text{Predictive Distribution,}} \tag{12.21}$$

where

$$\mu_y = \boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x})\boldsymbol{\mu}_{\theta|y}, \tag{12.22}$$

$$\begin{aligned}\sigma_y^2 &= \sigma_\eta^2 + \boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x})\Sigma_{\theta|y}\boldsymbol{\phi}(\boldsymbol{x}) \\ &= \sigma_\eta^2 + \boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x})\left(\frac{1}{\sigma_\theta^2}I + \frac{1}{\sigma_\eta^2}\Phi^{\mathrm{T}}\Phi\right)^{-1}\boldsymbol{\phi}(\boldsymbol{x}) \\ &= \sigma_\eta^2 + \sigma_\eta^2\sigma_\theta^2\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x})\left(\sigma_\eta^2 I + \sigma_\theta^2\Phi^{\mathrm{T}}\Phi\right)^{-1}\boldsymbol{\phi}(\boldsymbol{x}).\end{aligned} \tag{12.23}$$

Hence, given $\boldsymbol{x}$ one can predict the respective value of y using the most probable value, that is, μ_y in (12.22). Note that the same prediction value would result via the MAP estimate in (12.10) (or (12.12), if $\boldsymbol{\theta}_0$, also obtained via the ridge regression task). Have we then gained anything extra by adopting the Bayesian approach? The answer is in the affirmative. *More information concerning the predicted value is now available, because* (12.23) *quantifies the associated uncertainty*.

To investigate (12.23) further, let us simplify it by adopting the following approximation

$$R_\phi := \mathbb{E}[\boldsymbol{\phi}(\mathbf{x})\boldsymbol{\phi}^{\mathrm{T}}(\mathbf{x})] \simeq \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\phi}(\boldsymbol{x}_n)\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_n) = \frac{1}{N}\Phi^{\mathrm{T}}\Phi,$$

or

$$\Phi^{\mathrm{T}}\Phi \simeq NR_\phi, \tag{12.24}$$

where R_ϕ is the autocorrelation matrix of the random vector $\boldsymbol{\phi}(\mathbf{x})$. Employing (12.24) into (12.23), leads to

$$\sigma_y^2 \simeq \sigma_\eta^2\left(1 + \sigma_\theta^2\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x})\left(\sigma_\eta^2 I + N\sigma_\theta^2 R_\phi\right)^{-1}\boldsymbol{\phi}(\boldsymbol{x})\right), \tag{12.25}$$

which for large enough N becomes

$$\sigma_y^2 \simeq \sigma_\eta^2\left(1 + \frac{1}{N}\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x})R_\phi^{-1}\boldsymbol{\phi}(\boldsymbol{x})\right).$$

Thus, for a large number of observations, $\sigma_y^2 \longrightarrow \sigma_\eta^2$, and our uncertainty is contributed by the noise source, which cannot be reduced anymore. For smaller values of N, there is extra uncertainty associated with the parameter θ, measured by σ_θ^2 in (12.25).

So far in this section, we dealt with Gaussians, which led to tractable and analytically computed integrals. Are there ways to attack more general cases? Moreover, even in the case of Gaussian pdfs, we have assumed the covariance matrices Σ_θ, Σ_η to be known. In practice, they are not. Even if one assumes that Σ_η can be experimentally measured, there still remains Σ_θ. Can one select the related parameters via an optimization process? If the answer is yes, can this optimization be carried out on

the training set, or one would necessarily run into problems similar to the ones we faced with the regularization approach? We will indulge in all these challenges in the sections to follow.

Remarks 12.1.

- The MAP estimator is sometimes referred to as *Type I* estimator, to be distinguished from the Type II estimation method, which will be discussed in Remarks 12.2, in the next section.
- The posterior mean in (12.10) can be met in different variants, which are obtained via the application of the matrix inversion lemmas given in Appendix A.1. In Section 12.9, it is shown that (Eq. (12.149))

$$\boldsymbol{\mu}_{\theta|y} = \left(\Sigma_\theta^{-1} + \Phi^{\mathrm{T}}\Sigma_\eta^{-1}\Phi\right)^{-1}\left(\Phi^{\mathrm{T}}\Sigma_\eta^{-1}\boldsymbol{y} + \Sigma_\theta^{-1}\boldsymbol{\theta}_0\right) \tag{12.26}$$

or (Eq. (12.145))

$$\boldsymbol{\mu}_{\theta|y} = \boldsymbol{\theta}_0 + \Sigma_\theta\Phi^{\mathrm{T}}\left(\Sigma_\eta + \Phi\Sigma_\theta\Phi^{\mathrm{T}}\right)^{-1}(\boldsymbol{y} - \Phi\boldsymbol{\theta}_0)\,. \tag{12.27}$$

Also, using Woodbury's identity from Appendix A.1, we can readily see that

$$\Sigma_{\theta|y} = \Sigma_\theta - \Sigma_\theta\Phi^{\mathrm{T}}\left(\Sigma_\eta + \Phi\Sigma_\theta\Phi^{\mathrm{T}}\right)^{-1}\Phi\Sigma_\theta. \tag{12.28}$$

In practice, one uses the most computationally convenient form, depending on the dimensionality of the involved matrices to invert the one of lower dimension.

Example 12.1. This example demonstrates the prediction task summarized in (12.22), (12.23). Data are generated based on the following nonlinear model,

$$y_n = \theta_0 + \theta_1 x_n + \theta_2 x_n^2 + \theta_3 x_n^3 + \theta_5 x_n^5 + \eta_n,\ n = 1, 2, \ldots, N,$$

where η_n are i.i.d. noise samples drawn form a zero mean Gaussian with variance σ_η^2. Samples x_n are equidistant points in the interval $[0, 2]$. The goal of the task is to predict the value y given a measured value x, using (12.22). The parameter values used to generate the data were equal to,

$$\theta_0 = 0.2,\ \theta_1 = -1,\ \theta_2 = 0.9,\ \theta_3 = 0.7,\ \theta_5 = -0.2.$$

(a) In the first set of experiments, a Gaussian prior for the unknown θ was used with mean $\boldsymbol{\theta}_0$ equal to the previous true set of parameters and $\Sigma_\theta = 0.1I$. Also, the true model structure was used to construct the matrix Φ. Figure 12.1a shows the points (y, x) in red together with the error bars, as measured by the computed σ_y^2, for the case of $N = 20$ training points and $\sigma_\eta^2 = 0.05$. Figure 12.1b demonstrates the obtained improvement when the training points are increased to $N = 500$, while keeping the values of the other two parameters unchanged. Figure 12.1c corresponds to the latter case, where the noise variance is increased to $\sigma_\eta^2 = 0.15$.

(b) In the second set of experiments, we kept the correct model, however, the mean of the prior was given a different value to that of the true model, namely,

$$\boldsymbol{\theta}_0 = [-10.54, 0.465, 0.0087, -0.093, -0.004]^{\mathrm{T}}.$$

Figure 12.2a corresponds to the case of $\sigma_\eta^2 = 0.05$, $N = 20$, and $\sigma_\theta^2 = 0.1$. Note the improvement that is obtained when increasing $\sigma_\theta^2 = 2$, shown in Figure 12.2b, while N and σ_η^2 remain the same as before; this is because the model takes into consideration our uncertainty about the prior mean being away form the true value. Figure 12.2c corresponds to $\sigma_\eta^2 = 0.05$, $N = 500$, and $\sigma_\theta^2 = 0.1$ and shows the advantage of using a large number of training points.

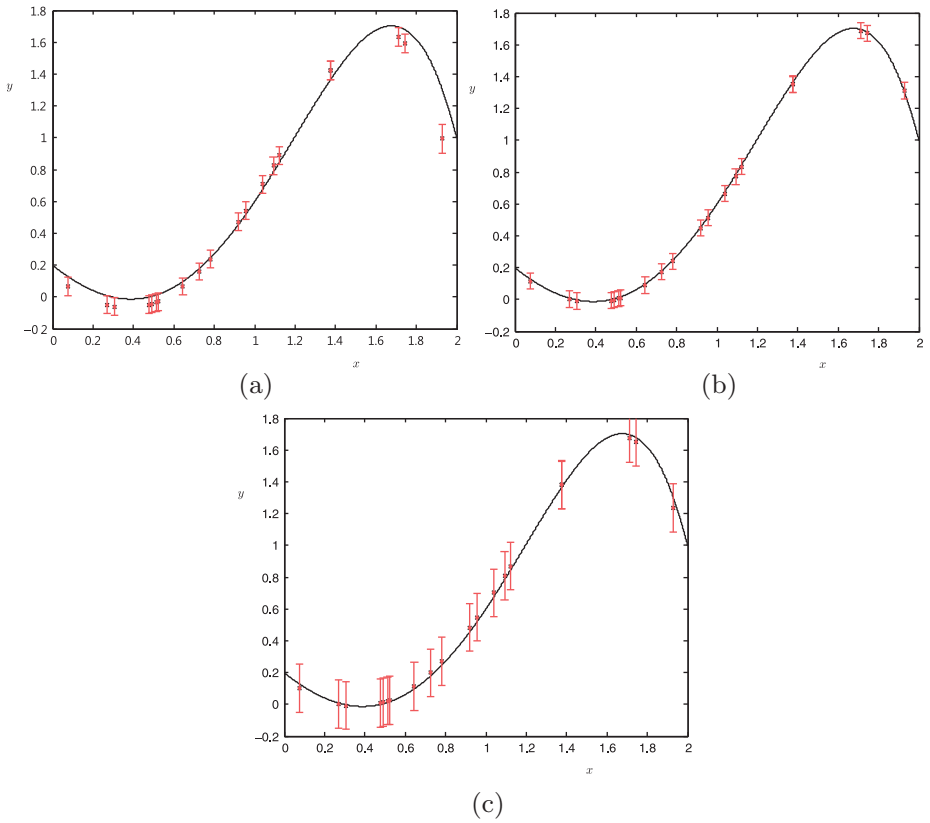

FIGURE 12.1

Each one of the red points, (y, x), indicates the prediction (y) corresponding to the input value, (x). The error bars are dictated by the computed variance, σ_y^2. The mean values used in the Gaussian prior are equal to the true values of the unknown model. (a) $\sigma_\eta^2 = 0.05$, $N = 20$, $\sigma_\theta^2 = 0.1$. (b) $\sigma_\eta^2 = 0.05$, $N = 500$, $\sigma_\theta^2 = 0.1$. (c) $\sigma_\eta^2 = 0.15$, $N = 500$, $\sigma_\theta^2 = 0.1$. Observe that the larger the data set, the better the predictions are and the larger the noise variance, the larger the error bars become.

(c) Figure 12.2d, corresponds to the case where the adopted model for prediction is the wrong one, that is,

$$y = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4 + \eta.$$

The used values were $\sigma_\eta^2 = 0.05$, $N = 500$, and $\sigma_\theta^2 = 2$. Observe that once a wrong model has been adopted, one must not have "high expectations" for good prediction performance.

12.3 THE EVIDENCE FUNCTION AND OCCAM'S RAZOR RULE

In the previous section, we made a comment about the importance of the marginal pdf $p(\boldsymbol{y})$. This section is fully dedicated to this quantity. In the notation used in (12.14), we did silently suppress the dependence on the adopted model. For example, the Gaussian assumption for the prior in (12.8) and for

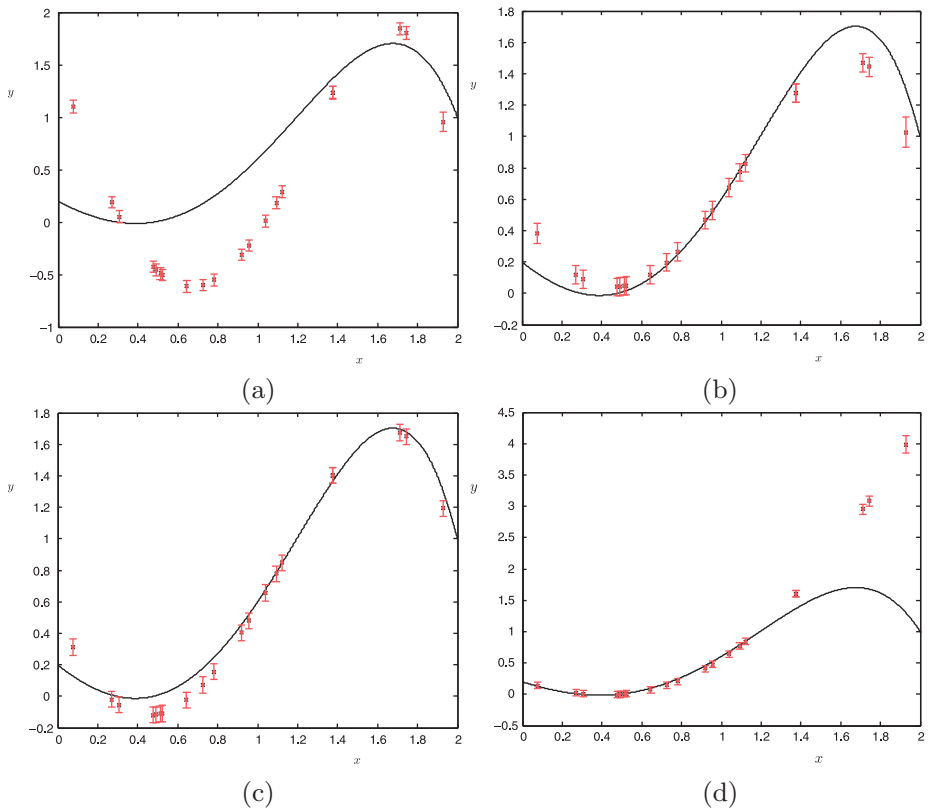

FIGURE 12.2

In this set of figures the mean values of the prior are different from that of the true model. (a) $\sigma_\eta^2 = 0.05$, $N = 20$, $\sigma_\theta^2 = 0.1$. (b) $\sigma_\eta^2 = 0.05$, $N = 20$, $\sigma_\theta^2 = 2$; observe the effect of using larger variance for the prior. (c) $\sigma_\eta^2 = 0.05$, $N = 500$, $\sigma_\theta^2 = 0.1$; observe the effect of the larger training data set. (d) The points correspond to a wrong model.

the conditional in (12.9) should have been reflected in the marginal as $p(\boldsymbol{y}; \boldsymbol{\phi}, \Sigma_\eta, \Sigma_\theta)$, because different Gaussians, different basis functions, and different orders, K, of the model can be used. Furthermore, non-Gaussian pdfs can also be adopted. In a more general setting, let us make the dependence on the model explicit as $p(\boldsymbol{y}|\mathcal{M}_i)$. Assuming the choice of a model to be random, then mobilizing Bayes theorem once more, we have

$$P(\mathcal{M}_i|\boldsymbol{y}) = \frac{P(\mathcal{M}_i)p(\boldsymbol{y}|\mathcal{M}_i)}{p(\boldsymbol{y})}, \tag{12.29}$$

where

$$p(\boldsymbol{y}) = \sum_i P(\mathcal{M}_i)p(\boldsymbol{y}|\mathcal{M}_i), \tag{12.30}$$

and $P(\mathcal{M}_i)$ is the prior probability of $\mathcal{M}_i$. $P(\mathcal{M}_i)$ provides a measure of the subjective prior over all possible models, which expresses our guess on how plausible a model is with respect to alternative

ones, prior to the data arrival. Because the denominator in (12.29) is independent of the model, one can obtain the most probable model, after observing $\boldsymbol{y}$, by maximizing the numerator. If one assigns to all possible models equal probabilities, then detecting the most probable model under the given set of observations becomes a task of maximizing $p(\boldsymbol{y}|\mathcal{M}_i)$. This is the reason that this pdf is known as the *evidence function* for the model or simply as the *evidence*. In practice, we content ourselves with using the most probable model, although an orthodox Bayesian would suggest to average all obtained quantities over all possible models, as in (12.30). In an ideal Bayesian setting, one does not choose among models; predictions are performed by summing over all possible models, each one weighted by the respective probability. However, in many practical problems we may have reasons to suggest that the evidence function is strongly peaked around a specific model; after all, such an assumption may simplify the task considerably.

We now turn our attention to what is hidden behind the optimization of $p(\boldsymbol{y}|\mathcal{M}_i)$ with respect to different models. Before we proceed, it is worth making a comment. A superficial first look may lead one to think whether this is any different from maximizing the likelihood $p(\boldsymbol{y};\boldsymbol{\theta})$, as it was done in Section 12.2.1. As a matter of fact, the two cases belong to two different worlds. ML maximizes with respect to a single (vector) parameter within an adopted model, and this is the weak point that makes ML vulnerable to overfitting. Maximizing the evidence is an optimization task with respect to the model itself, a wise alternative that guards us against overfitting, as we explain next.

From (12.14) we have

$$\boxed{p(\boldsymbol{y}|\mathcal{M}_i) = \int p(\boldsymbol{y}|\mathcal{M}_i, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{M}_i) d\boldsymbol{\theta} : \quad \text{Evidence Function.}} \tag{12.31}$$

Let us assume for simplicity that θ is a scalar, $\theta \in \mathbb{R}$, and that the integrand in (12.31), which according to the Bayes theorem is analogous to the posterior $p(\theta|\boldsymbol{y}, \mathcal{M}_i)$, peaks around a value; this is obviously the value that would result as the MAP estimate, $\hat{\theta}_{\text{MAP}}$. Figure 12.3 illustrates the respective graphs. Thus, (12.31) can be approximated by

$$p(\boldsymbol{y}|\mathcal{M}_i) \simeq p(\boldsymbol{y}|\mathcal{M}_i, \hat{\theta}_{\text{MAP}}) p(\hat{\theta}_{\text{MAP}}|\mathcal{M}_i) \Delta\theta_{\theta|y}. \tag{12.32}$$

To get a better feeling for each one of the factors involved in (12.32), let us also assume that the prior pdf is (almost) uniform with a width equal to $\Delta\theta$. Then, (12.32) is rewritten as

$$p(\boldsymbol{y}|\mathcal{M}_i) \simeq p(\boldsymbol{y}|\mathcal{M}_i, \hat{\theta}_{\text{MAP}}) \frac{\Delta\theta_{\theta|y}}{\Delta\theta}. \tag{12.33}$$

The first factor in the product on the right-hand side in (12.33) coincides with the likelihood function at its optimal value, because for this case of uniform prior, $\hat{\theta}_{\text{MAP}} = \hat{\theta}_{\text{ML}}$. In other words, this factor provides us with the best fit that model $\mathcal{M}_i$ can achieve on the given set of observations. However, now, in contrast to the ML method, the evidence function also depends on the second factor, $\frac{\Delta\theta_{\theta|y}}{\Delta\theta}$. As it has been pointed out in the insightful papers [14, 28, 30], this term accounts for the complexity of the model, and it is named the Occam factor for obvious reasons. Let us elaborate on this a bit more by following the reasoning given in [30].

The Occam factor penalizes those models, which are finely tuned to the received observations. As an example, if two different models $\mathcal{M}_i$ and $\mathcal{M}_j$ have a similar range of values for their prior pdfs, then if, say, $\Delta\theta_{\theta|y}(\mathcal{M}_i) \ll \Delta\theta_{\theta|y}(\mathcal{M}_j)$ then $\mathcal{M}_i$ will be penalized more; only a small range of values for θ survive (i.e., correspond to high probability values) after the reception of $\boldsymbol{y}$. So, if this fine-tuned

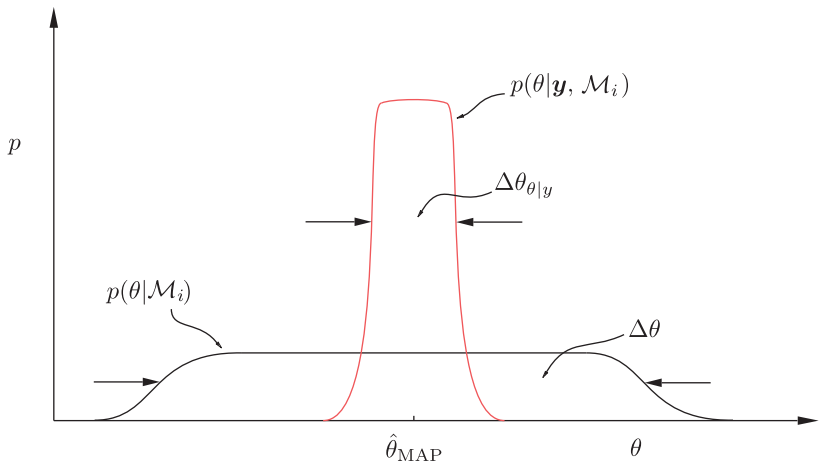

FIGURE 12.3

The posterior peaks around the value $\hat{\theta}_{\text{MAP}}$ and the posterior pdf can be approximated by $p(\hat{\theta}_{\text{MAP}}|\boldsymbol{y}; \mathcal{M}_i)$ over an interval of values equal to $\Delta\theta_{\theta|y}$.

(to the data) model, $\mathcal{M}_i$, had resulted in a large value of the ML term, it is not certain that the evidence would be maximized for it, because the Occam factor would be small. Which model, between the two, finally wins it depends on the product of the two involved terms. Soon we will see that the Occam term is also related to the number of parameters; that is, to the complexity of the adopted model.

Laplacian approximation and the evidence function

To investigate the evidence function for the general multiparameter case, we will employ the method of Laplacian approximation of a pdf. This is a general methodology that approximates any pdf *locally* in terms of a Gaussian one. To this end, define[5]

$$g(\boldsymbol{\theta}) = \ln\big(p(\boldsymbol{y}|\mathcal{M}_i, \boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{M}_i)\big). \tag{12.34}$$

Use Taylor's expansion around $\hat{\boldsymbol{\theta}}_{\text{MAP}}$ and keep terms up to the second order,

$$\begin{aligned} g(\boldsymbol{\theta}) &= g(\hat{\boldsymbol{\theta}}_{\text{MAP}}) + (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{\text{MAP}})^{\text{T}} \frac{\partial g(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_{\text{MAP}}} \\ &\quad + \frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{\text{MAP}})^{\text{T}} \frac{\partial^2 g(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^2}\Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_{\text{MAP}}} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{\text{MAP}}) \\ &= g(\hat{\boldsymbol{\theta}}_{\text{MAP}}) - \frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{\text{MAP}})^{\text{T}} \Sigma^{-1} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{\text{MAP}}), \end{aligned} \tag{12.35}$$

where

$$\Sigma^{-1} := -\frac{\partial^2 g(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^2}\Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_{\text{MAP}}},$$

[5] Similarly, to obtain the Laplacian approximation of a general pdf, $p(\boldsymbol{x})$, we set $g(\boldsymbol{x}) = \ln p(\boldsymbol{x})$.

which leads to the approximation,

$$p(\boldsymbol{y}|\mathcal{M}_i,\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{M}_i) \simeq p(\boldsymbol{y}|\mathcal{M}_i,\hat{\boldsymbol{\theta}}_{\text{MAP}})p(\hat{\boldsymbol{\theta}}_{\text{MAP}}|\mathcal{M}_i)\times \exp\left(-\frac{1}{2}(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}_{\text{MAP}})^{\text{T}}\Sigma^{-1}(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}_{\text{MAP}})\right). \tag{12.36}$$

Plugging (12.36) into the integral of (12.31) we obtain

$$p(\boldsymbol{y}|\mathcal{M}_i) = p(\boldsymbol{y}|\mathcal{M}_i,\hat{\boldsymbol{\theta}}_{\text{MAP}})p(\hat{\boldsymbol{\theta}}_{\text{MAP}}|\mathcal{M}_i)(2\pi)^{\frac{K}{2}}|\Sigma|^{1/2}, \tag{12.37}$$

and taking the logarithms we have

$$\underbrace{\ln p(\boldsymbol{y}|\mathcal{M}_i)}_{\text{Evidence}} = \underbrace{\ln p(\boldsymbol{y}|\mathcal{M}_i,\hat{\boldsymbol{\theta}}_{\text{MAP}})}_{\text{Best likelihood fit}} + \underbrace{\ln p(\hat{\boldsymbol{\theta}}_{\text{MAP}}|\mathcal{M}_i) + \frac{K}{2}\ln(2\pi) + \frac{1}{2}\ln|\Sigma|}_{\text{Occam factor}}. \tag{12.38}$$

The direct dependence of the Occam term on the complexity (number of basis functions) of the adopted model is now readily spotted. Moreover, the complexity-related Occam term depends on the prior pdf and the second derivatives (via Σ) of the posterior pdf, too; that is, it depends on how "sharp" the shape of the latter is in the K-dimensional space. In other words, the covariance term provides the "error bar" information. Hence, in a single equation, besides the number of parameters and the associated best-fit term, the evidence also takes into account information related to the associated variance; maximizing the evidence leads to the best trade-off. Figure 12.4 illustrates the essence behind the evidence maximization for model selection. If the model is too complex, it can fit well a wide range of data sets, and because

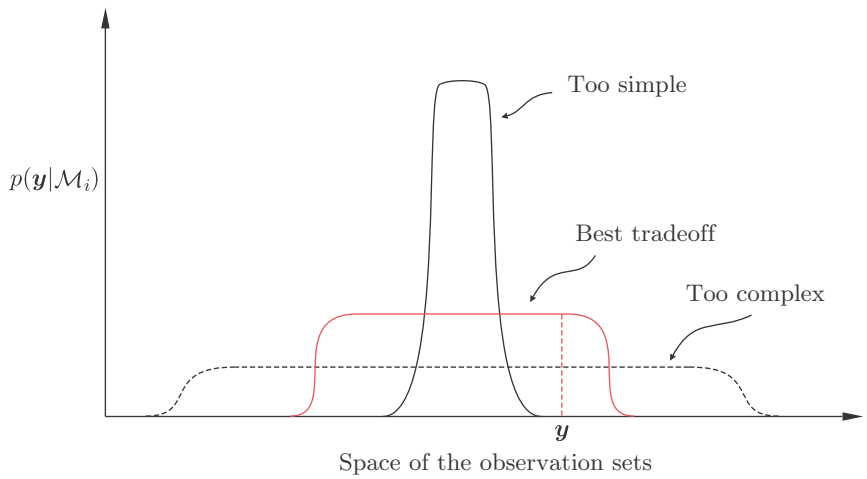

FIGURE 12.4

Too simple models can explain well a very small range of data. On the other hand, too complex models can explain a wide range of data; however, they do not provide any confidence because they assign low probability to all data sets. For the observation set, $\boldsymbol{y}$, the evidence is maximized for the model with intermediate complexity.

$p(\boldsymbol{y}|\mathcal{M}_i)$ has to integrate to one, its value for any value of $\boldsymbol{y}$ is expected to be low. The opposite is true for models that are too simple; such models can model well some data sets but not a wide range of them, and consequently, the evidence function peaks sharply around a value in the space of observation sets. Thus, selecting a data set at random it is rather unlikely that this has been generated by such a model. Having said that, it is important to emphasize, once more, the Occam term does not depend *solely* on the number of parameters; hence, complexity here should be interpreted in a more "open-minded" way. This robustness against overfitting, which is intrinsic in the Bayesian inference approach, is the consequence of integrating the parameters for any specific model in (12.31); this integration penalizes models of high complexity because such models can model a large range of data.

Historically, the Occam's razor rule in its Bayesian interpretation was first demonstrated in [14] and later on in [28, 39], although the foundations go back to the pioneering work of Sir Herald Jeffrey in the 1930s, [24]. Two insightful reviews on the Bayesian inference approach that are well worth reading are given in [23, 27].

Returning to (12.14) and assuming for simplicity that

$$p(\boldsymbol{y}|\boldsymbol{\theta}) = \mathcal{N}\left(\boldsymbol{y}|\Phi\boldsymbol{\theta}, \sigma_\eta^2 I\right) \quad \text{and} \quad p(\boldsymbol{\theta}) = \mathcal{N}\left(\boldsymbol{\theta}|\boldsymbol{\theta}_0, \sigma_\theta^2 I\right), \tag{12.39}$$

we can express the evidence as $p(\boldsymbol{y}; \sigma_\eta^2, \sigma_\theta^2)$, which, for this case, turns out to be Gaussian (appendix in Section 12.9); thus, it is available in closed form. Hence for this specific case, the model space is described via σ_η^2, σ_θ^2 and maximization of the evidence with respect to these (unknown) model parameters can take place iteratively, for the given set of observations stacked in $\boldsymbol{y}$, see, for example, [28]. However, being able to express the evidence in closed form is not the case in general. The EM algorithm, which is described in Section 12.5, is a popular way and a powerful tool to this end. One could also resort to the Laplacian approximation to approximate the involved pdfs as Gaussians, but this approximation turns out not always to be a good choice; furthermore, in high dimensional parameter spaces, the computations of the second order derivatives and the determinant can become burdensome [3].

Finally, let us make a final comment concerning the Laplacian approximation. In the discussion above, our goal was to get an approximation of the integral (normalizing constant/evidence) of $p(\boldsymbol{y}|\mathcal{M}_i, \boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{M}_i)$. However, if our interest were to approximate the pdf itself, we should be careful in selecting the normalizing constant, which by the nature of the Gaussian function leads to

$$p(\boldsymbol{y}|\mathcal{M}_i, \boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{M}_i) \simeq \frac{1}{(2\pi)^{K/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{\text{MAP}})^{\text{T}} \Sigma^{-1} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{\text{MAP}})\right).$$

This is also the case for the Laplacian approximation for any pdf $p(\cdot)$.

Remarks 12.2.

- In the Bayesian approach, one makes all the modeling assumptions explicit, and it is then left to the rules of probability theory to provide the answers. One does not have to "worry" about the choice of an optimizing criterion, where different criteria lead to different estimators, and there is not an objective, systematic way to decide which criterion is best. On the other hand, in the Bayesian approach one has to make sure to select the prior that explains the data in the best possible way.
- The choice of the prior pdf is very critical in the performance of Bayesian methods and must be carried out in such a way to encapsulate prior knowledge as fully as possible. In practice, different alternatives can be adopted [3].

- *Subjective priors*. According to this path, we choose the prior $p(\boldsymbol{\theta})$ to make the manipulation of the integration tractable, by employing conjugate priors (Section 3.11.1) within the exponential family of pdfs. This family of pdfs will be presented in the next section.
- *Hierarchical priors*. Each one of the components of $\theta_k,\ k = 0, 2, \ldots, K-1$ of $\boldsymbol{\theta}$, is controlled by a different parameter; for example, all θ_k's may be assumed to be independent Gaussian variables, each one with a different variance. In turn, variances are considered to be random variables that follow a statistical distribution controlled by another set of deterministic (not random) parameters known as *hyperparameters*; thus, a hierarchy of priors is adopted. As we will see later on, hierarchical priors are often designed using conjugate pairs of pdfs.
- *Noninformative or objective priors.* The choice of the prior is done in such a way to embed as little extra information as possible and to exploit knowledge that is conveyed only by the available data. One way to construct such priors is to resort to information theoretic arguments. For example, one can estimate $p(\boldsymbol{\theta})$ by minimizing its Kullback-Leibler distance from $p(\boldsymbol{\theta}|\boldsymbol{y})$.

- The fact that the Bayesian approach allows the recovery of all the desired information from a single data set does not suggest that the method is "cross-validation" free. Maximizing the evidence, which at the same time guards against overfitting, *does not* necessarily mean that the performance of the designed estimator is optimized. This is more true in practice where, as we are going to see very soon, most often a bound of the evidence is optimized instead, to bypass computational obstacles. As it is always the case in life, the proof of the cake is in the eating. Thus, the final verdict should only come from the generalization ability of the designed estimator; that is, its ability to make reliable predictions using data unseen to it before. However, there is no reason to suggest that the evidence may be a reliable predictor of the generalization performance. This has been known and explicitly stated since the method's infancy stage, see, for example, [28]. The generalization performance depends heavily on whether the adopted prior matches the "true" distribution of the unknown parameters. This is nicely demonstrated with a toy example in [18]. It is shown that the Bayesian average is optimal only if the adopted prior coincides with the true one. The situation is less clear when this is not the case. A more theoretical treatment of the topic, when there is a mismatch between the true and the selected prior, can be found in [19]. Thus, to be able to assess the generalization performance of a model learned via Bayesian inference, cross-validation is required, unless an independent test set can be afforded, for example, [46].

  To avoid the need for cross-validation, an alternative way has been adopted by a number of authors. The cost function, to be minimized in (12.13), is built to quantify the generalization performance of an estimator; optimization then takes place concurrently for the unknown weights as well as the regularization parameter, see, for example, [16, 34]. In general, this leads to a nonconvex optimization task and such techniques have not, yet, been widely embraced by the machine learning community.
- The Laplacian approximation to the evidence function is closely related to the *Bayesian information criterion* (BIC) [41] for model selection, which is expressed as,

$$\ln p(\boldsymbol{y}|\mathcal{M}_i) \approx \ln p(\boldsymbol{y}|\mathcal{M}_i, \hat{\boldsymbol{\theta}}_{\text{MAP}}) - \frac{1}{2} K \ln N.$$

  BIC is obtained as a large N approximation to (12.38), assuming a broad enough Gaussian prior, and manipulating a bit on the determinant involved in the last term. For a discussion including other related criteria, see [3, 43].

- The Bayesian framework is also closely related to the minimum description length (MDL) methods. The log-evidence is associated with the number of bits in the shortest message that encodes the data via model $\mathcal{M}_i$, for example, [47].
- *Type II maximum likelihood*: Note that the evidence is the marginal likelihood function after integrating out the parameters Θ. To distinguish it from the MAP method, when the evidence function is maximized with respect to a set of unknown parameters, it is usually referred to as *generalized maximum likelihood* or *Type II maximum likelihood* and sometimes as *empirical Bayes*. Recall from Remarks 12.1, that the MAP was also named as Type I estimator.

12.4 EXPONENTIAL FAMILY OF PROBABILITY DISTRIBUTIONS

We will treat the topic of the exponential family of probability distributions in a general setting. Let $\mathbf{x} \in \mathbb{R}^l$ be a random vector and $\Theta \in \mathbb{R}^K$ a random (parameter) vector. We say that the parameterized pdf $p(\boldsymbol{x}|\boldsymbol{\theta})$ is of the exponential form if

$$p(\boldsymbol{x}|\boldsymbol{\theta}) = g(\boldsymbol{\theta}) f(\boldsymbol{x}) \exp\left(\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{u}(\boldsymbol{x})\right), \tag{12.40}$$

where

$$g(\boldsymbol{\theta}) = \frac{1}{\int f(\boldsymbol{x}) \exp\left(\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{u}(\boldsymbol{x})\right) d\boldsymbol{x}}, \tag{12.41}$$

is the normalizing constant of the pdf. A similar definition holds if $\mathbf{x}$ is a discrete random variable and the respective function represents the probability mass function $P(\boldsymbol{x}|\boldsymbol{\theta})$; in this case, the integration in (12.41) becomes a summation. The vector $\boldsymbol{\phi}(\boldsymbol{\theta})$ comprises the set of the so-called *natural parameters*, and f, $\boldsymbol{u}$ are functions defining the distribution. It is readily seen from the factorization theorem in Section 3.7 that $\boldsymbol{u}(\boldsymbol{x})$ is a *sufficient statistic* for the parameter $\boldsymbol{\theta}$. Note that an attribute of the exponential family is that the number of sufficient statistics, that is, the dimensionality of $\boldsymbol{u}$, is finite and remains independent of the number of observations. If $\boldsymbol{\phi}(\boldsymbol{\theta}) = \boldsymbol{\theta}$, then the exponential family is said to be in *canonical* form. A number of widely used distributions belongs to the exponential family, for example, the normal, exponential, gamma, chi-squared, Beta, Dirichlet, Bernoulli, binomial, multinomial distributions. Examples of distributions that do not belong in this family are the uniform with unknown bounds, the student's-t and most mixture distributions, for example, [42, 48].

An advantage of the exponential family is that one can find conjugate priors for Θ; that is, priors that lead to posteriors, $p(\boldsymbol{\theta}|\mathcal{X})$, of the same functional form as $p(\boldsymbol{\theta})$ (Section 3.11.1).

Given (12.40) its conjugate prior is given by

$$p(\boldsymbol{\theta}; \lambda, \boldsymbol{v}) = h(\lambda, \boldsymbol{v}) \big(g(\boldsymbol{\theta})\big)^{\lambda} \exp\left(\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{v}\right), \tag{12.42}$$

where $\lambda > 0$ and $\boldsymbol{v}$ are known as *hyperparameters*; that is, parameters that control other parameters. The factor $h(\lambda, \boldsymbol{v})$ is an appropriate normalizing constant. It is easy to see that defining the prior as in (12.42) and the likelihood function as in (12.40), the posterior $p(\boldsymbol{\theta}|\boldsymbol{x})$ is of the same form as in (12.42).

Before we give some examples, let us investigate a bit more the role played by λ and $\boldsymbol{v}$, as well as the presence of $g(\boldsymbol{\theta})$ and $\boldsymbol{\phi}(\boldsymbol{\theta})$ in both (12.42) and (12.40). Assume that $\mathbf{x}$ and Θ obey (12.40)–(12.42) and let $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ be a set of i.i.d. observations. Then,

$$p(\mathcal{X}|\boldsymbol{\theta}) = (g(\boldsymbol{\theta}))^N \prod_{n=1}^{N} f(\boldsymbol{x}_n) \exp\left(\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{\theta}) \sum_{i=1}^{N} \boldsymbol{u}(\boldsymbol{x}_i)\right), \tag{12.43}$$

and

$$p(\boldsymbol{\theta}|\mathcal{X}) \propto p(\mathcal{X}|\boldsymbol{\theta}) p(\boldsymbol{\theta}) \propto (g(\boldsymbol{\theta}))^{\lambda+N} \exp\left(\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{\theta}) \left(\boldsymbol{v} + \sum_{n=1}^{N} \boldsymbol{u}(\boldsymbol{x}_n)\right)\right). \tag{12.44}$$

In other words, the posterior has hyperparameters equal to

$$\tilde{\lambda} = \lambda + N, \;\; \tilde{\boldsymbol{v}} = \boldsymbol{v} + \sum_{n=1}^{N} \boldsymbol{u}(\boldsymbol{x}_n). \tag{12.45}$$

Interpreting (12.45), one can view λ as being the effective number of observations that, implicitly, the prior information contributes to the Bayesian learning process and $\boldsymbol{v}$ is the total amount of information that these (implicit) λ observations contribute to the sufficient statistic. Their exact values, basically, quantify the amount of prior knowledge that the designer wants to embed into the problem.

Example 12.2. *The Gaussian-gamma pair*: Let our random variable x be a scalar and assume that

$$p(x|\sigma^2) = \mathcal{N}(x|\mu, \sigma^2), \tag{12.46}$$

where μ is known and σ^2 is an unknown parameter. We will show that

1. $p(x|\sigma^2)$ belongs to the exponential family.

It is algebraically more convenient to work with the precision $\beta = \frac{1}{\sigma^2}$. Thus,

$$p(x|\beta) = \frac{\beta^{1/2}}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\beta(x-\mu)^2\right). \tag{12.47}$$

Thus, $p(x|\beta)$ belongs to the exponential family with

$$f(x) = \frac{1}{\sqrt{2\pi}}, \; \phi(\beta) = -\beta, \; u(x) = \frac{1}{2}(x-\mu)^2,$$

and

$$g(\beta) = \frac{1}{\int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\beta(x-\mu)^2\right) dx} = \beta^{1/2}.$$

2. The conjugate prior of (12.46) follows the gamma distribution.

The respective conjugate prior from (12.42) becomes

$$p(\beta; \lambda, v) = h(\lambda, v) \beta^{\frac{\lambda}{2}} \exp(-\beta v). \tag{12.48}$$

This has the form of

$$\text{Gamma}(\beta|a, b) = \frac{1}{\Gamma(a)} b^a \beta^{a-1} \exp(-b\beta), \tag{12.49}$$

with parameters (Chapter 2) $a = \frac{\lambda}{2} + 1$ and $b = v$ and the normalizing constant, $h(\lambda, v)$, being necessarily equal to $b^a/\Gamma(a)$. The function $\Gamma(a)$ is defined as

$$\Gamma(a) = \int_0^\infty x^{a-1} e^{-x} dx.$$

If we are given multiple observations $x_n,\ n = 1, 2, \ldots, N$, then the resulting posterior according to (12.44) and (12.45) will be a gamma distribution with

$$\tilde{b} = b + \frac{1}{2}\sum_{n=1}^{N}(x_n - \mu)^2 = b + \frac{N}{2}\hat{\sigma}^2_{\text{ML}},$$

where $\hat{\sigma}^2_{\text{ML}}$ denotes the maximum likelihood estimate of the variance (Problem 3.20). Hence, the physical meaning of b is that it quantifies our prior estimate about the unknown variance. This also ties nicely with what we have said in Section 3.7; $\hat{\sigma}^2_{\text{ML}}$ is a sufficient statistic for the variance, if this is the unknown parameter in a Gaussian. It can easily be shown that the conjugate prior with respect to μ, if σ^2 is known, is a Gaussian (Problem 12.4).

In case of a multivariate Gaussian of known mean $\boldsymbol{\mu}$ and unknown covariance matrix Σ (precision matrix $Q = \Sigma^{-1}$), it can also be shown that it is of the exponential form and its conjugate prior is given by the Wishart distribution,

$$\mathcal{W}(Q|W, v) = h|Q|^{\frac{v-l-1}{2}} \exp\left(-\frac{1}{2}\text{trace}\left\{W^{-1}Q\right\}\right), \tag{12.50}$$

where h is the normalizing constant (Problem 12.5) and W is an $l \times l$ matrix. The normalizing constant is given by

$$h = |W|^{-\frac{v}{2}}\left(2^{\frac{vl}{2}}\pi^{\frac{l(l-1)}{4}}\prod_{i=1}^{l}\Gamma\left(\frac{v+1-i}{2}\right)\right)^{-1}, \tag{12.51}$$

which admittedly is quite intimidating; however, in Bayesian learning we have the luxury of bypassing the computation of the normalizing factor in (12.50). Once we express a pdf in terms of Q as in (12.50), then the normalizing constant has to be given by (12.51). The Wishart distribution is a multivariate analogue of the gamma distribution.

Example 12.3. *The Gaussian Gaussian-gamma pair*: We will show that

1. $p(x; \mu, \sigma^2) = \mathcal{N}(x|\mu, \sigma^2)$ is also of an exponential form. Indeed, for this case we have

$$p(x; \mu, \sigma^2) = p(x; \mu, \beta^{-1}) = \frac{\beta^{1/2}\exp\left(-\beta\frac{\mu^2}{2}\right)}{\sqrt{2\pi}}\exp\left(\left[-\frac{\beta}{2}, \beta\mu\right]\begin{bmatrix} x^2 \\ x \end{bmatrix}\right).$$

Hence,

$$\boldsymbol{\theta} = [\beta, \mu]^{\text{T}},\ \boldsymbol{\phi}(\boldsymbol{\theta}) = \begin{bmatrix} -\frac{\beta}{2} \\ \beta\mu \end{bmatrix},\ \boldsymbol{u}(\boldsymbol{x}) = \begin{bmatrix} x^2 \\ x \end{bmatrix},$$

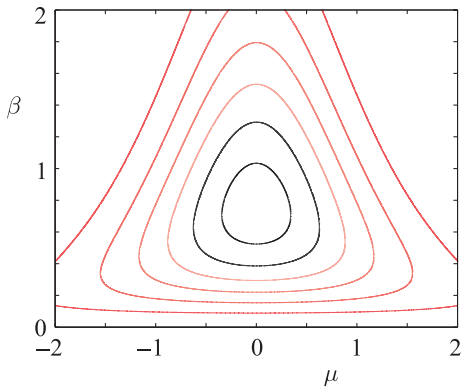

FIGURE 12.5

Contour plots of the Gauss-gamma distribution with parameter values $\lambda = 2, v_1 = 4, v_2 = 0$.

and performing the respective integration, we obtain

$$f(\boldsymbol{x}) = \frac{1}{\sqrt{2\pi}},\ g(\boldsymbol{\theta}) = \beta^{1/2} \exp\left(-\frac{\beta\mu^2}{2}\right),$$

which proves the claim.

2. The conjugate prior of $p(x|\mu, \sigma^2)$ is of a Gaussian-gamma form.

We have that

$$p(\mu, \beta; \lambda, \boldsymbol{v}) = h(\lambda, \boldsymbol{v})\beta^{\frac{\lambda}{2}} \exp\left(-\frac{\lambda\beta\mu^2}{2}\right) \exp\left(\left[-\frac{\beta}{2}, \beta\mu\right]\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}\right),$$

which after some trivial algebra (Problem 12.6) gives

$$p(\mu, \beta; \lambda, \boldsymbol{v}) = \mathcal{N}\left(\mu \middle| \frac{v_2}{\lambda}, (\lambda\beta)^{-1}\right) \text{Gamma}\left(\beta \middle| \frac{\lambda+1}{2}, \frac{v_1}{2} - \frac{v_2^2}{2\lambda}\right), \tag{12.52}$$

which is known as the Gaussian-gamma distribution with the Gaussian having mean value $\mu_0 = \frac{v_2}{\lambda}$ and variance $\sigma_\mu^2 = (\lambda\beta)^{-1}$ and the defining parameters of the gamma pdf are, $a = \frac{\lambda+1}{2}$ and $b = \frac{v_1}{2} - \frac{v_2^2}{2\lambda}$. Figure 12.5 shows the contour plot of the Gauss-gamma distribution of (12.6).

For the more general case of a multivariate Gaussian, $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \Sigma)$, it turns out that it is also of an exponential form and its conjugate prior is of the Gaussian-Wishart form (Problem 12.7), that is,

$$p\left(\boldsymbol{\mu}, Q; \boldsymbol{\mu}_0, \lambda, W, \nu\right) = \mathcal{N}\left(\boldsymbol{\mu}|\boldsymbol{\mu}_0, (\lambda Q)^{-1}\right) \mathcal{W}\left(Q|W, \nu\right),$$

where, $Q = \Sigma^{-1}$.

Example 12.4. We now turn our attention into discrete variables and we will show that the multinomial distribution is of an exponential form and that its conjugate prior is given by the Dirichlet distribution.

1. Let $z_1, z_2, \ldots, z_K$ be K mutually exclusive and exhaustive events. Let $P_1, P_2, \ldots, P_K$ be the respective probabilities, hence $\sum_{k=1}^{K} P_k = 1$. Let the experiment be repeated N times. Then, the probability of the joint event: z_1 occurred x_1 times, z_2 occurred x_2 times, and so on, is given by the multinomial distribution.

$$P(x_1, x_2, \ldots, x_K) = \begin{pmatrix} N \\ x_1 \ldots x_K \end{pmatrix} \prod_{k=1}^{K} P_k^{x_k}, \tag{12.53}$$

where

$$\begin{pmatrix} N \\ x_1 \ldots x_K \end{pmatrix} = \frac{N!}{x_1! x_2!, \ldots, x_K!}.$$

Defining $\boldsymbol{P} = [P_1, \ldots, P_K]^{\mathrm{T}}$, Eq. (12.53) can be rewritten as

$$\begin{aligned} P(x_1, \ldots, x_K | \boldsymbol{P}) &= \begin{pmatrix} N \\ x_1 \ldots x_K \end{pmatrix} \prod_{k=1}^{K} \exp\left(x_k \ln P_k\right) \\ &= \begin{pmatrix} N \\ x_1, \ldots, x_K \end{pmatrix} \exp\left(\sum_{k=1}^{K} x_k \ln P_k\right). \end{aligned} \tag{12.54}$$

Thus, the multinomial is of an exponential form with

$$\boldsymbol{\phi}(\boldsymbol{P}) = [\ln P_1, \ln P_2, \ldots, \ln P_K]^{\mathrm{T}},$$

$$\boldsymbol{u}(\boldsymbol{x}) = [x_1, x_2, \ldots, x_K]^{\mathrm{T}},$$

and because probabilities sum to one, we obtain

$$g(\boldsymbol{P}) = 1, \; f(\boldsymbol{x}) = \begin{pmatrix} N \\ x_1 \ldots x_K \end{pmatrix}.$$

2. The conjugate prior of (12.54) can then be written as

$$\begin{aligned} p(\boldsymbol{P}; \lambda, \boldsymbol{v}) &= h(\lambda, \boldsymbol{v}) \exp\left(\sum_{k=1}^{K} v_k \ln P_k\right) \\ &\propto \prod_{k=1}^{K} P_k^{v_k}, \end{aligned} \tag{12.55}$$

which is a Dirichlet pdf. If we let $v_k := a_k - 1$ we bring (12.55) in the more standard formulation

$$p(\boldsymbol{P}; \boldsymbol{a}) = \frac{\Gamma(\overline{a})}{\Gamma(a_1) \ldots \Gamma(a_K)} \prod_{k=1}^{K} P_k^{a_k - 1}, \; \sum_{k=1}^{K} P_k = 1, \tag{12.56}$$

where the normalization constant (Chapter 2) has been plugged in, with

$$\overline{a} := \sum_{k=1}^{K} a_k.$$

12.4.1 THE EXPONENTIAL FAMILY AND THE MAXIMUM ENTROPY METHOD

Besides the computational advantages associated with the exponential family, there is another reason that justifies its high popularity. Assume that we are given a set of observations, $x_n \in \mathcal{A}_x \subseteq \mathbb{R}$, $n = 1, 2, \ldots, N$, drawn from a distribution whose functional form is unknown. Our goal is to estimate the unknown pdf; however, we require that it will respect certain empirical expectations, which are computed from the available observations, that is,

$$\hat{\mu}_i := \frac{1}{N} \sum_{n=1}^{N} u_i(x_n), \; i \in \mathcal{I}, \tag{12.57}$$

where $\mathcal{I}$ is an index set and $u_i : \; \mathcal{A}_x \longmapsto \mathbb{R}, \; i \in \mathcal{I}$ are specific functions. For example, if $u_i(x) = x$, then $\hat{\mu}_i$ is the sample mean. In such cases, it is not sensible to adopt a parametric functional form for the pdf and try to optimize with respect to the unknown parameters, for example, via the maximum likelihood method; in general, we cannot know if an adopted functional form can comply with the available empirical expectations.

The maximum entropy (ME) method (sometimes called principle) offers a possible way to estimate the unknown pdf, subject to the set of the available constraints, [22]. According to this method, the cost function to be maximized is the *entropy* (Section 2.5.2) associated with the pdf, that is,

$$H := -\int_{\mathcal{A}_x} p(x) \ln p(x) \, \mathrm{d}x. \tag{12.58}$$

It is well known from Shannon's information theory that the entropy is a measure of uncertainty or randomness. Maximizing the entropy, with respect to $p(x)$, results to the most random pdf, subject to the available constraints. Seen from another point of view, such a procedure guarantees that the estimation of an unknown pdf is carried out by adopting the least number of assumptions; that is, only the available set of constraints. For our case, the maximum entropy estimation method is cast as follows:

$$\begin{aligned} &\text{maximize with respect to } p(x) && -\int_{\mathcal{A}_x} p(x) \ln p(x) dx, \\ &\text{subject to} && \mathbb{E}[u_i(\mathrm{x})] = \int_{\mathcal{A}_x} p(x) u_i(x) dx, \; i \in \mathcal{I}. \end{aligned} \tag{12.59}$$

In addition to the previous set of constraints, one has to consider the obvious one that guarantees that $p(x)$ integrates to one, that is,

$$\int_{\mathcal{A}_x} p(x) \, \mathrm{d}x = 1.$$

In the case of discrete variables, the involved integrations are replaced by summations. Solving the optimization task in (12.59), it turns out that (Problem 12.9)

$$\hat{p}(x) = C \exp\left(\sum_{i \in \mathcal{I}} \theta_i u_i(x)\right), \tag{12.60}$$

that is, the ME estimate is of an exponential form. The parameters $\theta_i,\ i \in \mathcal{I}$, are the Lagrange multipliers used in the optimization task and their values are determined via the constraints and are given in terms of the available empirical expectations, $\hat{\mu}_i,\ i \in \mathcal{I}$. If no constraint is used other than the obvious (normalizing) one and $\mathcal{A}_x = [a, b] \subset \mathbb{R}$, then the resulting pdf is the uniform distribution, $p(x) = C$; indeed, this is the most random one, because it shows no preference to any specific interval of values. If two constraints are used, such that $u_1(x) = x$ and $u_2(x) = x^2$, the resulting pdf is the Gaussian one, because the exponent is of a quadratic form (appendix in Section 12.9). In other words, the Gaussian is the most random pdf, subject to two constraints related to the mean and the variance. Note that although we focused on real-valued random variables, everything is trivially extended to vector-valued ones. An interesting discussion concerning maximum entropy method and alternative views of the problem is provided in [44].

12.5 LATENT VARIABLES AND THE EM ALGORITHM

At the end of Section 12.3, it was pointed out that the evidence function associated with the regression task in Eq. (12.3), assuming that $p(\boldsymbol{y}|\boldsymbol{\theta})$ and $p(\boldsymbol{\theta})$ are Gaussians of the form given in (12.39), is also Gaussian parameterized via a set of parameters, $\boldsymbol{\xi}$, where for this case $\boldsymbol{\xi} = [\sigma_\eta^2, \sigma_\theta^2]$, and we can write $p(\boldsymbol{y}; \boldsymbol{\xi})$. Maximizing the evidence with respect to $\boldsymbol{\xi}$ becomes a typical ML task. However, in general, such closed-form expressions for the evidence function are not possible, and the integration in (12.14) is intractable. The main source of difficulty is the fact that our regression model is described via two random variables, that is, $\mathbf{y}$ and $\boldsymbol{\theta}$, yet only one of them, $\mathbf{y}$, can be directly observed. The other one, $\boldsymbol{\theta}$, cannot be observed, and this is the reason that the Bayesian philosophy tries to integrate it out of the joint pdf $p(\boldsymbol{y}, \boldsymbol{\theta})$. If $\boldsymbol{\theta}$ could be observed, then the unknown set of parameters $\boldsymbol{\xi}$ could be obtained by maximizing the likelihood $p(\boldsymbol{y}, \boldsymbol{\theta}; \boldsymbol{\xi})$, given a set of (joint) observations $(\boldsymbol{y}, \boldsymbol{\theta})$. Because it cannot be observed, the random variable $\boldsymbol{\theta}$ is known as *latent* or *hidden* variable.

Although we introduced the notion of latent variables via our familiar regression task, latent variables occur very often in a number of problems in probability and statistics. In a number of cases, from a larger set of jointly distributed random variables, only some can be observed and the rest remain hidden. Moreover, it is often useful to *build* hidden variables into a model by design. These variables are meant to represent latent causes that influence the observed variables and their introduction may facilitate the analysis.

12.5.1 THE EXPECTATION-MAXIMIZATION ALGORITHM

The *expectation-maximization* (EM) algorithm is an elegant algorithmic tool to maximize the likelihood function for problems with latent variables. We will state the problem in a general formulation, and then we will apply it to different tasks, including regression.

Let $\mathbf{x}$ be a random vector and let $\mathcal{X}$ be the respective set of observations. Let $\mathcal{X}^l := \{\boldsymbol{x}_1^l, \ldots, \boldsymbol{x}_N^l\}$ be the corresponding set of latent variables; these can be either of a discrete or of a continuous nature. Each observation in $\mathcal{X}$ is associated with a latent vector $\boldsymbol{x}^l$ in $\mathcal{X}^l$. We will refer to the set $\left\{\mathcal{X}, \mathcal{X}^l\right\}$

as the *complete* data set and to the set of observations $\mathcal{X}$ as the *incomplete* one. Let, also, their joint distribution be parameterized in terms of a set of unknown parameters, $\boldsymbol{\xi}$.[6] We further assume that, although $\mathcal{X}^l$ cannot be observed, the posterior distribution $p(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi})$ ($P(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi})$ for the discrete case) is fully specified, given the values in $\boldsymbol{\xi}$ and the observations in $\mathcal{X}$. This is a critical assumption for the EM algorithm. If the posterior pdf is not known, then one has to resort to variants of the EM, which attempt to approximate it. We will come to such schemes in Section 13.2.

If the complete log-likelihood $p(\mathcal{X},\mathcal{X}^l;\boldsymbol{\xi})$ were available, then the problem would be a typical ML one. However, because no observations for the latent variables are available, the EM algorithm considers the *expectation* of the complete log-likelihood with respect to the latent variables associated with $\mathcal{X}^l$; this operation is possible, because the posterior distribution $p(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi})$ is known, provided that $\boldsymbol{\xi}$ is known. To this end, the EM algorithm builds on an iterative philosophy, initialized by an arbitrary value $\boldsymbol{\xi}^{(0)}$. Then it proceeds along the following steps (see Problem 12.11 for a justification).

The EM Algorithm

1. Expectation E-step: at the $(j+1)$ iteration, compute $p(\mathcal{X}^l|\mathcal{X},\boldsymbol{\xi}^{(j)})$ and

$$\boxed{\mathcal{Q}(\boldsymbol{\xi},\boldsymbol{\xi}^{(j)}) = \mathbb{E}\Big[\ln p(\mathcal{X},\mathcal{X}^l;\boldsymbol{\xi})\Big],} \tag{12.61}$$

   where the expectation is taken with respect to $p(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi}^{(j)})$.
2. Maximization M-step: Determine $\boldsymbol{\xi}^{(j+1)}$ so that

$$\boxed{\boldsymbol{\xi}^{(j+1)} = \arg\max_{\boldsymbol{\xi}} \mathcal{Q}(\boldsymbol{\xi},\boldsymbol{\xi}^{(j)}).} \tag{12.62}$$

3. Check for convergence according to a criterion. If it is not satisfied go to step 1.

A possible convergence criterion is to check whether $\|\boldsymbol{\xi}^{(j+1)} - \boldsymbol{\xi}^{(j)}\| < \epsilon$, for some user-defined constant ϵ. The use of the EM algorithm presupposes that working with the joint pdf $p(\mathcal{X},\mathcal{X}^l;\boldsymbol{\xi})$ is computationally tractable. This is, for example, the case when working within the exponential family of pdfs, where the E-step may require only the computation of a few statistics of the latent variables.

Remarks 12.3.

- The EM algorithm was proposed and given its name in the seminal 1977 paper by Arthur Dempster, Nan Laird, and Donald Rubin [12]. The paper generalized previously published results, as for example, [2, 40] and had a significant impact as a powerful tool in statistics. The complete convergence proof was given in [49]. See, for example, [31] for a related discussion.
- It can be shown that the EM algorithm converges to a (in general, local) maximum of $p(\mathcal{X};\boldsymbol{\xi})$, which was our original goal. The likelihood never decreases. The convergence is slower than the quadratic convergence of Newton-type searching techniques, although near an optimal point a speed up may be possible. However, the convergence of the algorithm is smooth and its complexity more attractive to Newton-type schemes, with no matrix inversions involved. The keen reader may obtain more information in, for example, [15, 32, 35, 45].

[6] In the context of this chapter, we keep the notation $\boldsymbol{\theta}$ for parameters treated as random variables and $\boldsymbol{\xi}$ for deterministic ones.

- The EM algorithm can be modified to obtain the MAP estimate. To this end, the M-step is changed to

$$\boldsymbol{\xi}^{(j+1)} = \arg\max_{\boldsymbol{\xi}} \left\{ \mathcal{Q}(\boldsymbol{\xi}, \boldsymbol{\xi}^{(j)}) + \ln p(\boldsymbol{\xi}) \right\},$$

  where $p(\boldsymbol{\xi})$ is the prior pdf associated with $\boldsymbol{\xi}$, if it is considered to be a random vector.
- The EM algorithm can be sensitive to the choice of the initial point $\boldsymbol{\xi}^{(0)}$. In practice, one can run the algorithm a number of times, starting from different initial points and keep the best of the results. Other initialization procedures have also been used, depending on the application.
- *Missing data*: The EM algorithm can also be used to cope with cases where some of the values from the observed training data are missing. Missing values can be treated as hidden variables and maximization of the likelihood can be done by marginalizing over them. Such a procedure makes sense only if data are *missing at random*; that is, the cause of missing data is a random event and does not depend on the values of the unobserved samples.

12.5.2 THE EM ALGORITHM: A LOWER BOUND MAXIMIZATION VIEW

Let us consider the functional[7]

$$\boxed{\mathcal{F}(q, \boldsymbol{\xi}) := \int q(\mathcal{X}^l) \ln \frac{p(\mathcal{X}, \mathcal{X}^l; \boldsymbol{\xi})}{q(\mathcal{X}^l)} \, \mathrm{d}\mathcal{X}^l,} \tag{12.63}$$

where $q(\mathcal{X}^l)$ is any nonnegative function that integrates to one; that is, it is a pdf defined over the latent variables. The functional $\mathcal{F}(\cdot, \cdot)$, depends on $\boldsymbol{\xi}$ and on $q(\cdot)$, and its definition bears a strong similarity with the notion of free energy, used in statistical physics. Indeed, (12.63) can be written as,

$$\mathcal{F}(q, \boldsymbol{\xi}) = \int q(\mathcal{X}^l) \ln p(\mathcal{X}, \mathcal{X}^l; \boldsymbol{\xi}) \, \mathrm{d}\mathcal{X}^l + H,$$

where

$$H = -\int q(\mathcal{X}^l) \ln q(\mathcal{X}^l) \, \mathrm{d}\mathcal{X}^l,$$

is the entropy associated with $q(\mathcal{X}^l)$. If one defines $-\ln p(\mathcal{X}, \mathcal{X}^l; \boldsymbol{\xi})$ as the *energy* of the system, $(\mathcal{X}, \mathcal{X}^l)$, then $\mathcal{F}(q, \boldsymbol{\xi})$ represents the negative of the so-called *free energy* [36].

Elaborating on (12.63), we get

$$\begin{aligned} \mathcal{F}(q, \boldsymbol{\xi}) &= \int q(\mathcal{X}^l) \ln \frac{p(\mathcal{X}^l | \mathcal{X}; \boldsymbol{\xi}) p(\mathcal{X}; \boldsymbol{\xi})}{q(\mathcal{X}^l)} d\mathcal{X}^l, \\ &= \int q(\mathcal{X}^l) \ln \frac{p(\mathcal{X}^l | \mathcal{X}; \boldsymbol{\xi})}{q(\mathcal{X}^l)} d\mathcal{X}^l + \ln p(\mathcal{X}; \boldsymbol{\xi}), \end{aligned} \tag{12.64}$$

where the latter results because $p(\mathcal{X}; \boldsymbol{\xi})$ does not depend on $q(\mathcal{X}^l)$. The first term on the right-hand side is the negative of the so-called *Kullback-Leibler divergence* between $q(\mathcal{X}^l)$ and $p(\mathcal{X}^l | \mathcal{X}; \xi)$, which we will denote as $\mathrm{KL}(q \parallel p)$. Thus, finally we get

[7] A functional is an operator that takes as input a function and returns a real value. It is a generalization of our familiar functions, where now the inputs are also functions.

$$\ln p(\mathcal{X};\boldsymbol{\xi}) = \mathcal{F}(q,\boldsymbol{\xi}) + \mathrm{KL}(q \parallel p). \tag{12.65}$$

Because $\mathrm{KL}(q \parallel p) \geq 0$ (Problem 12.12) it turns out that

$$\boxed{\ln p(\mathcal{X};\boldsymbol{\xi}) \geq \mathcal{F}(q,\boldsymbol{\xi}).} \tag{12.66}$$

$\mathcal{F}(q,\boldsymbol{\xi})$ is a lower bound of the log-likelihood function, and the bound becomes tight if $\mathrm{KL}(q \parallel p) = 0$, which is true, *if and only if*, $q(\mathcal{X}^l) = p(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi})$. The previous findings pave the way of maximizing $\ln p(\mathcal{X};\xi)$ by trying to maximize its lower bound. Note that maximization of $\mathcal{F}(\cdot,\cdot)$ involves two terms, namely $q,\ \boldsymbol{\xi}$. We will adopt a procedure that belongs to a more general class of optimization algorithms known as *alternating optimization*. Such an approach naturally imposes an iterative procedure that starts from an arbitrary $\boldsymbol{\xi}^{(0)}$ and the $(j+1)$ iteration comprises the following steps:

- Step 1: Holding $\boldsymbol{\xi}^{(j)}$ fixed, optimize with respect to q. This step tightens the lower bound in (12.66). This is achieved if $\mathrm{KL}(q \parallel p) = 0$, and it can only happen if

$$q^{(j+1)}(\mathcal{X}^l) = p(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi}^{(j)}), \tag{12.67}$$

  that is, if we set $q(\mathcal{X}^l)$ equal to the posterior given $\mathcal{X}$ and $\boldsymbol{\xi}^{(j)}$; as (12.65) suggests, this makes the bound tight, that is,

$$\ln p(\mathcal{X};\boldsymbol{\xi}^{(j)}) = \mathcal{F}\left(p(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi}^{(j)}),\boldsymbol{\xi}^{(j)}\right). \tag{12.68}$$

- Step 2: Fixing $q^{(j+1)}$, insert it in the place of q in (12.66), and because the bound holds for any q, maximize with respect to $\boldsymbol{\xi}$, that is,

$$\boldsymbol{\xi}^{(j+1)} = \arg\max_{\boldsymbol{\xi}} \mathcal{F}\left(p\left(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi}^{(j)}\right),\boldsymbol{\xi}\right).$$

It is now readily seen that we have re-derived the EM algorithm. Indeed, from the definition of $\mathcal{F}(\cdot,\cdot)$ in (12.63) we obtain that

$$\mathcal{F}\left(p(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi}^{(j)}),\boldsymbol{\xi}\right) = \mathcal{Q}(\boldsymbol{\xi},\boldsymbol{\xi}^{(j)}) - \int p(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi}^{(j)}) \ln p(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi}^{(j)}) d\mathcal{X}^l, \tag{12.69}$$

where $\mathcal{Q}(\boldsymbol{\xi},\boldsymbol{\xi}^{(j)})$ is the same as in (12.61) and the second term on the right-hand side is independent of $\boldsymbol{\xi}$; this latter term is equal to the entropy associated with $q^{(j+1)}(\mathcal{X}^l)$. The rederivation of the EM via this path makes it clear that the quantity that is maximized is the log-likelihood, $\ln p(\mathcal{X};\boldsymbol{\xi})$, and that its value is guaranteed not to decrease after each combined iteration step. Figure 12.6 illustrates schematically the two EM steps comprising the $(j+1)$th iteration.

Needless to say that the EM algorithm is not a panacea. We will soon seek variants to deal with cases where the posterior cannot be given in an analytic form. Moreover, there are still cases where the M-step can be computationally intractable. To this end, several variants have been proposed, see, for example, [33, 36].

Remarks 12.4.

- *Online versions of EM*: We have already pointed out that in many cases of large data applications, online versions are the preferable choice in practice. The EM algorithm is no exception, and a number of related versions have been proposed. In [36], an online EM algorithm is proposed based on the lower bound interpretation. In [6], stochastic approximation arguments have been employed. In [26], a comparative study of different techniques is reported.

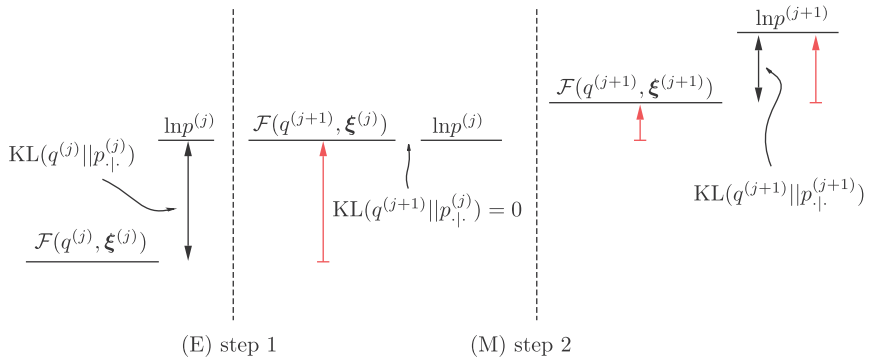

FIGURE 12.6

The E-step adjusts $q^{(j)} := q^{(j)}(\mathcal{X}^l)$ so that its Kullback-Leibler (KL) distance from $p_{\cdot|\cdot}^{(j)} := p(\mathcal{X}^l|\mathcal{X}; \boldsymbol{\xi}^{(j)})$ becomes zero. The M-step maximizes with respect to $\boldsymbol{\xi}$.

- Often in practice, carrying out the expectation step may be intractable. Later on in the next chapter, we will see variational methods as a way to overcome this obstacle. An alternative path is to employ Monte Carlo sampling techniques (Chapter 14) to generate samples from the involved distributions and approximate the expectation with the computation of the respective sample mean see, for example, [8, 11].
- The EM algorithm can also be seen as a special case of a more general class of methods, which are based on optimizing a bound instead of the original cost. These techniques come under the name of *minorize-maximization, or mazorize-minimization* (MM) methods; see, for example, [21].

12.6 LINEAR REGRESSION AND THE EM ALGORITHM

The Bayesian viewpoint to the regression task was considered in Section 12.2.3 via the Gaussian model assumption for $p(\boldsymbol{y}|\boldsymbol{\theta})$ and $p(\boldsymbol{\theta})$, given in (12.9) and (12.8), which subsequently led to a Gaussian posterior for $p(\boldsymbol{\theta}|\boldsymbol{y})$, given in (12.16). In the current section, and for the sake of presentation simplicity, we will adopt the special case of diagonal covariance matrices, that is, $\Sigma_\eta = \sigma_\eta^2 I$, $\Sigma_\theta = \sigma_\theta^2 I$, and $\boldsymbol{\theta}_0 = \mathbf{0}$.

Our goal now becomes to consider σ_η^2 and σ_θ^2 as (non-random) parameters and to obtain their values by maximizing the corresponding evidence function in (12.15). To this end, we will use the EM algorithm. We will treat $\boldsymbol{y}$ as the observed variables (corresponding to $\mathcal{X}$) and $\boldsymbol{\theta}$ as the set of latent variables (corresponding to $\mathcal{X}^l$). A prerequisite in order to apply the EM procedure is the knowledge of the posterior, which for this case is known, given the values of the parameters. As we have done many times in the book so far, we will work with the precision variables, and the parameter vector becomes

$$\boldsymbol{\xi} = [\alpha, \beta]^{\mathrm{T}}, \ \alpha = \frac{1}{\sigma_\theta^2} \quad \text{and} \quad \beta = \frac{1}{\sigma_\eta^2}.$$

The EM algorithm is initialized with some arbitrary positive values $\alpha^{(0)}$ and $\beta^{(0)}$. Then the algorithm at the $(j+1)$ iteration step, where $\alpha^{(j)}$ and $\beta^{(j)}$ are assumed known, proceeds as follows:

- E-Step: Compute the posterior $p(\boldsymbol{\theta}|\boldsymbol{y};\boldsymbol{\xi}^{(j)})$, which according to (12.16) and for $\boldsymbol{\theta}_0 = \mathbf{0}$ is fully specified if we compute its mean and covariance matrix, using (12.19, 12.20), that is,

$$\boxed{\begin{aligned}\Sigma_{\theta|y}^{(j)} &= \left(\alpha^{(j)} I + \beta^{(j)}\Phi^{\mathrm{T}}\Phi\right)^{-1}, \qquad (12.70)\\ \boldsymbol{\mu}_{\theta|y}^{(j)} &= \beta^{(j)}\Sigma_{\theta|y}^{(j)}\Phi^{\mathrm{T}}\boldsymbol{y}. \qquad (12.71)\end{aligned}}$$

Compute the expected value of the log-likelihood associated with the complete data set; this is given by,

$$\ln p(\boldsymbol{y},\boldsymbol{\theta};\boldsymbol{\xi}) := \ln p(\boldsymbol{y},\boldsymbol{\theta};\alpha,\beta) = \ln\Big(p(\boldsymbol{y}|\boldsymbol{\theta};\beta)p(\boldsymbol{\theta};\alpha)\Big),$$

or

$$\begin{aligned}\ln p(\boldsymbol{y},\boldsymbol{\theta};\alpha,\beta) = \frac{N}{2}\ln\beta + \frac{K}{2}\ln\alpha - \frac{\beta}{2}\|\boldsymbol{y}-\Phi\boldsymbol{\theta}\|^2 - \frac{\alpha}{2}\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\theta}\\ - \left(\frac{N}{2}+\frac{K}{2}\right)\ln(2\pi). \qquad (12.72)\end{aligned}$$

Treating the latent parameters as random variables, the expected value of (12.72), with respect to $\boldsymbol{\theta}$, is carried out via the Gaussian posterior defined by (12.70) and (12.71). To this end, the following steps are adopted.

1. To compute $\mathbb{E}[\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\theta}]$, recall the definition of the respective covariance matrix,

$$\Sigma_{\theta|y}^{(j)} = \mathbb{E}\left[(\boldsymbol{\theta}-\boldsymbol{\mu}_{\theta|y}^{(j)})(\boldsymbol{\theta}-\boldsymbol{\mu}_{\theta|y}^{(j)})^{\mathrm{T}}\right] \qquad (12.73)$$

or

$$\mathbb{E}[\boldsymbol{\theta}\boldsymbol{\theta}^{\mathrm{T}}] = \Sigma_{\theta|y}^{(j)} + \boldsymbol{\mu}_{\theta|y}^{(j)}\boldsymbol{\mu}_{\theta|y}^{(j)T}, \qquad (12.74)$$

which results in

$$\begin{aligned}A := \mathbb{E}[\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\theta}] &= \mathbb{E}[\mathrm{trace}\{\boldsymbol{\theta}\boldsymbol{\theta}^{\mathrm{T}}\}]\\ &= \mathrm{trace}\{\boldsymbol{\mu}_{\theta|y}^{(j)}\boldsymbol{\mu}_{\theta|y}^{(j)T} + \Sigma_{\theta|y}^{(j)}\}\\ &= \|\boldsymbol{\mu}_{\theta|y}^{(j)}\|^2 + \mathrm{trace}\{\Sigma_{\theta|y}^{(j)}\}. \qquad (12.75)\end{aligned}$$

2. To compute $\mathbb{E}[\|\boldsymbol{y}-\Phi\boldsymbol{\theta}\|^2]$, define $\boldsymbol{\psi} := \boldsymbol{y}-\Phi\boldsymbol{\theta}$, and use the previous rationale to compute $\mathbb{E}[\boldsymbol{\psi}^T\boldsymbol{\psi}]$, which leads to (Problem 12.13)

$$B := \mathbb{E}[\|\boldsymbol{y}-\Phi\boldsymbol{\theta}\|^2] = \|\boldsymbol{y}-\Phi\boldsymbol{\mu}_{\theta|y}^{(j)}\|^2 + \mathrm{trace}\{\Phi\Sigma_{\theta|y}^{(j)}\Phi^{\mathrm{T}}\}. \qquad (12.76)$$

Hence,

$$\mathcal{Q}(\alpha,\beta;\alpha^{(j)},\beta^{(j)}) = \frac{N}{2}\ln\beta + \frac{K}{2}\ln\alpha - \frac{\beta}{2}B - \frac{\alpha}{2}A - \left(\frac{N}{2}+\frac{K}{2}\right)\ln(2\pi). \qquad (12.77)$$

- M-Step: Compute

$$\begin{aligned}\alpha^{(j+1)} &: \frac{\partial}{\partial\alpha}\mathcal{Q}(\alpha,\beta;\alpha^{(j)},\beta^{(j)}) = 0,\\ \beta^{(j+1)} &: \frac{\partial}{\partial\beta}\mathcal{Q}(\alpha,\beta;\alpha^{(j)},\beta^{(j)}) = 0,\end{aligned}$$

which trivially lead to

$$\alpha^{(j+1)} = \frac{K}{\|\boldsymbol{\mu}_{\theta|y}^{(j)}\|^2 + \text{trace}\{\Sigma_{\theta|y}^{(j)}\}}, \tag{12.78}$$

$$\beta^{(j+1)} = \frac{N}{\|\boldsymbol{y} - \Phi\boldsymbol{\mu}_{\theta|y}^{(j)}\|^2 + \text{trace}\{\Phi\Sigma_{\theta|y}^{(j)}\Phi^{\mathrm{T}}\}}. \tag{12.79}$$

Once the algorithm converges, the resulting values for α and β are used to completely specify the involved pdfs, which can be used either to obtain an estimate of $\hat{\boldsymbol{\theta}}$, for example, $\hat{\boldsymbol{\theta}} = \mathbb{E}[\boldsymbol{\theta}|\boldsymbol{y}]$, or make predictions via (12.21).

Example 12.5. In this example, the generalized linear regression model of Example 12.1 is reconsidered. The goal is to use the EM algorithm of Section 12.6, as summarized by the recursions (12.70), (12.71), (12.78), and (12.79). The variance of the Gaussian noise used in the model to generate the data was set equal to $\sigma_\eta^2 = 0.05$. The number of training points was $N = 500$. For the EM algorithm, both α and β were initialized to one. The correct dimensionality for the unknown parameter vector was used. The recovered values after the convergence of the EM were $\alpha = 1.32$ corresponding to $\sigma_\theta^2 = 0.756$ and $\beta = 19.96$ corresponding to $\sigma_\eta^2 = 0.0501$. Note that the latter is very close to the true variance of the noise. Then, predictions of the output variable y were performed at 20 points, using (12.22) and the value of $\boldsymbol{\mu}_{\theta|y}$ recovered by the EM algorithm, via (12.71).

Figure 12.7a shows the predictions together with the associated error bars, computed from (12.23) using the values of σ_η^2 and σ_θ^2 obtained via the EM algorithm. Figure 12.7b shows the convergence curve for σ_η^2 as a function of the number of iterations of the EM algorithm.

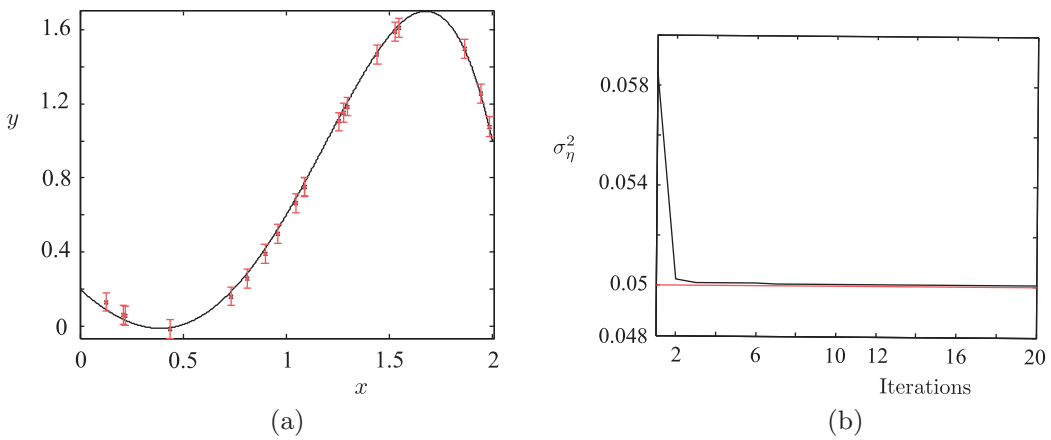

FIGURE 12.7

(a) The original graph from which the training points were sampled. In red, the respective predictions $\hat{y}$ and associated error bars for twenty randomly chosen points are shown. (b) The convergence curve for σ_η^2 as a function of the iterations of the EM algorithm. The red line corresponds to the true value.

12.7 GAUSSIAN MIXTURE MODELS

So far, we have seen a number of pdfs that can be used to model the distribution of an unknown random vector $\mathbf{x} \in \mathbb{R}^l$. However, all these models restrict the pdf to a specific functional term. Mixture modeling provides the freedom to model the unknown pdf, $p(\boldsymbol{x})$, as a linear combination of different distributions, that is,

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} P_k p(\boldsymbol{x}|k), \tag{12.80}$$

where P_k is the parameter weighting the specific contributing pdf, $p(\boldsymbol{x}|k)$. To guarantee that $p(\boldsymbol{x})$ is a pdf, the weighting parameters must be nonnegative and add to one ($\sum_{k=1}^{K} P_k = 1$). The physical interpretation of (12.80) is that we are given a set of K distributions, $p(\boldsymbol{x}|k),\ k = 1, 2, \ldots, K$. Each observation, $\boldsymbol{x}_n, n = 1, 2, \ldots, N$, is drawn from one of these K distributions, but we are not told from which one. All we know is a set of parameters, $P_k,\ 1, 2, \ldots, K$, each one providing the probability that a sample has been drawn from the corresponding pdf, $p(\boldsymbol{x}|k)$. It can be shown that for a large enough number of *mixtures*, K, and appropriate choice of the involved parameters, one can approximate arbitrarily close any continuous pdf.

Mixture modeling is a typical task involving hidden variables; that is, the labels, k, of the pdf from which an obtained observation has originated. In practice, each $p(\boldsymbol{x}|k)$ is chosen from a known pdf family, parameterized via a set of parameters, and (12.80) can be rewritten as

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} P_k p(\boldsymbol{x}|k; \boldsymbol{\xi}_k), \tag{12.81}$$

and the task is to estimate $(P_k, \boldsymbol{\xi}_k),\ k = 1, 2, \ldots, K$, based on a set of observations $\boldsymbol{x}_n,\ n = 1, 2, \ldots, N$. The set of observations, $\mathcal{X} = \{\boldsymbol{x}_n,\ n = 1, \ldots, N\}$, forms the incomplete set while the complete set $\{\mathcal{X}, \mathcal{K}\}$ comprises the samples $(\boldsymbol{x}_n, k_n),\ n = 1, \ldots, N$, with k_n being the label of the distribution (pdf) from which $\boldsymbol{x}_n$ was drawn. Parameter estimation for such a problem naturally lends itself to be treated via the EM algorithm. We will demonstrate the procedure via the use of Gaussian mixtures.

Let

$$p(\boldsymbol{x}|k; \boldsymbol{\xi}_k) = p(\boldsymbol{x}|k; \boldsymbol{\mu}_k, \Sigma_k) = \mathcal{N}\left(\boldsymbol{x}|\boldsymbol{\mu}_k, \Sigma_k\right),$$

where for simplicity we will assume that $\Sigma_k = \sigma_k^2 I,\ k = 1, \ldots, K$. We will further assume the observations to be i.i.d. For such a modeling, the following hold true:

- The log-likelihood of the *complete* data set is given by

$$\ln p\left(\mathcal{X}, \mathcal{K}; \boldsymbol{\Xi}, \boldsymbol{P}\right) = \sum_{n=1}^{N} \ln p(\boldsymbol{x}_n, k_n; \boldsymbol{\xi}_{k_n}) = \sum_{n=1}^{N} \ln\left(p(\boldsymbol{x}_n|k_n; \boldsymbol{\xi}_{k_n}) P_{k_n}\right). \tag{12.82}$$

  We have used the notation,

$$\boldsymbol{\Xi} = [\boldsymbol{\xi}_1^{\mathrm{T}}, \ldots, \boldsymbol{\xi}_K^{\mathrm{T}}]^{\mathrm{T}},\ \ \boldsymbol{P} = [P_1, P_2, \ldots, P_K]^{\mathrm{T}},\ \text{ and }\ \boldsymbol{\xi}_k = [\boldsymbol{\mu}_k^{\mathrm{T}}, \sigma_k^2]^{\mathrm{T}}.$$

- The posterior probabilities of the hidden discrete variables are given by

$$P(k|\boldsymbol{x}; \boldsymbol{\Xi}, \boldsymbol{P}) = \frac{p(\boldsymbol{x}|k; \boldsymbol{\xi}_k) P_k}{p(\boldsymbol{x}; \boldsymbol{\Xi}, \boldsymbol{P})}, \tag{12.83}$$

where

$$p(\boldsymbol{x};\boldsymbol{\Xi},\boldsymbol{P})=\sum_{k=1}^{K}P_k p(\boldsymbol{x}|k;\boldsymbol{\xi}_k). \tag{12.84}$$

We have now all the ingredients required by the EM algorithm. Starting from $\boldsymbol{\Xi}^{(0)}$ and $\boldsymbol{P}^{(0)}$, the $(j+1)$ iteration comprises the following steps:

- E-step: Using (12.83, 12.84) compute

$$\boxed{P(k|\boldsymbol{x}_n;\boldsymbol{\Xi}^{(j)},\boldsymbol{P}^{(j)})=\frac{p(\boldsymbol{x}_n|k;\boldsymbol{\xi}_k^{(j)})P_k^{(j)}}{\sum_{k=1}^{K}P_k^{(j)}p(\boldsymbol{x}_n|k;\boldsymbol{\xi}_k^{(j)})},\ n=1,2,\ldots,N,} \tag{12.85}$$

which in turn defines

$$\begin{aligned}\mathcal{Q}(\boldsymbol{\Xi},\boldsymbol{P};\boldsymbol{\Xi}^{(j)},\boldsymbol{P}^{(j)})&=\sum_{n=1}^{N}\mathbb{E}\left[\ln\left(p(\boldsymbol{x}_n|\mathrm{k}_n;\boldsymbol{\xi}_{k_n})P_{k_n}\right)\right]\\&:=\sum_{n=1}^{N}\sum_{k=1}^{K}P(k|\boldsymbol{x}_n;\boldsymbol{\Xi}^{(j)},\boldsymbol{P}^{(j)})\Big(\ln P_k-\frac{l}{2}\ln\sigma_k^2\\&\quad-\frac{1}{2\sigma_k^2}\|\boldsymbol{x}_n-\boldsymbol{\mu}_k\|^2\Big)+C,\end{aligned} \tag{12.86}$$

where C includes all the terms corresponding to the normalization constant. Note that we have finally relaxed the notation from k_n to k, because we sum up over all k, which does not depend on n.

- M-step: Maximization of $\mathcal{Q}(\boldsymbol{\Xi},\boldsymbol{P};\boldsymbol{\Xi}^{(j)},\boldsymbol{P}^{(j)})$ with respect to all the involved parameters results in the following set of recursions (Problem 12.14):
  Set for notational convenience,

$$\gamma_{kn}:=P(k|\boldsymbol{x}_n;\boldsymbol{\Xi}^{(j)},\boldsymbol{P}^{(j)}).$$

Then,

$$\boldsymbol{\mu}_k^{(j+1)}=\frac{\sum_{n=1}^{N}\gamma_{kn}\boldsymbol{x}_n}{\sum_{n=1}^{N}\gamma_{kn}}, \tag{12.87}$$

$$\sigma_k^{2(j+1)}=\frac{\sum_{n=1}^{N}\gamma_{kn}\|\boldsymbol{x}_n-\boldsymbol{\mu}_k^{(j+1)}\|^2}{l\sum_{n=1}^{N}\gamma_{kn}}, \tag{12.88}$$

$$P_k^{(j+1)}=\frac{1}{N}\sum_{n=1}^{N}\gamma_{kn}. \tag{12.89}$$

Iterations continue until a convergence criterion is met. The extension to the case of a general covariance matrix is straightforward by replacing (12.88) by

$$\Sigma_k^{(j+1)}=\frac{\sum_{n=1}^{N}\gamma_{kn}(\boldsymbol{x}_n-\boldsymbol{\mu}_k^{(j+1)})(\boldsymbol{x}_n-\boldsymbol{\mu}_k^{(j+1)})^{\mathrm{T}}}{\sum_{n=1}^{N}\gamma_{kn}}.$$

Remarks 12.5.

- To get good initialization for the EM algorithm, sometimes a simpler clustering algorithm, for example, the k-means (Section 12.7.1, [43]), is run to provide an initial estimate of the means and shapes of clusters (covariance matrices), by associating each mixture with a cluster in the input space. Another simpler way is to select K points randomly from the data set. A more elaborate technique, which is commonly used, is to select them randomly but in such a way to make sure that the whole data set is represented in a balanced way, see, for example, [1].
- The number of mixtures, K, is usually determined by cross-validation (Chapter 3); see, also, [17].
- The mixing parameters, $P_k,\ k = 1, \ldots, K$, should be initialized by keeping in mind that they are probabilities and they have to add to one.
- One of the problems, that may be encountered in practice in the Gaussian mixture task, is when one of the mixture components is centered at (or very close to) one of the data points, for example, $\boldsymbol{\mu}_k^{(j+1)} = \boldsymbol{x}_n$, for some values of k and n. In such a case, the exponent term of the respective Gaussian becomes one and the contribution of this particular component in the log-likelihood is equal to $(2\pi\sigma_k^2)^{-l/2}$. If, in addition, σ_k is very small, this will lead the likelihood to a large value, although this is not indicative that the true model has been learned. Soon, we will see that the use of priors can alleviate such problems.
- *Identifiability*: A further issue associated with the EM algorithm, in the context of distribution mixtures, is that the obtained solution in the parameter space is not unique. For the case of K mixtures, for each solution (point in the parameter space) there are $K! - 1$ other points which give rise to the same distribution. For example, let us fit a model of two Gaussians in the one-dimensional space, which will result in estimates for the respective mean values, $\hat{\mu}_1$ and $\hat{\mu}_2$. However, in the corresponding parameter space, there is an uncertainty on whether these values define the point $\boldsymbol{\mu}_a = [\hat{\mu}_1, \hat{\mu}_2]^{\mathrm{T}}$ or the point $\boldsymbol{\mu}_b = [\hat{\mu}_2, \hat{\mu}_1]$. Both of these points give rise to the same distribution. We say that the parameters in our model are not identifiable. A parameter (vector), which defines a family of distributions, $p(\boldsymbol{x}; \boldsymbol{\theta})$, is said to be *identifiable* if $p(\boldsymbol{x}; \boldsymbol{\theta}_1) \neq p(\boldsymbol{x}; \boldsymbol{\theta}_2)$ for $\boldsymbol{\theta}_1 \neq \boldsymbol{\theta}_2$, see, e.g, [7]. Although in our context, where our interest is in computing $p(\boldsymbol{x})$, unidentifiability does not cause any problems, this can be an issue in cases where the focus of interest lies on the parameters, see, for example, [38].
- *Mixtures of Student's-t distributions*: A significant shortcoming of mixtures based on normal distributions is their vulnerability to outliers. Recently, the replacement of normal distributions with the heavier-tailed Student's-t distributions (see Section 13.5) has been proposed as a way to mitigate these shortcomings and a related treatment of the resulting model under an expectation-maximization algorithmic framework has been conducted. Although the steps get a bit more involved, the ideas explored so far transfer nicely in this case too, see, for example, [9, 10, 37, 39].

Example 12.6. The goal of this example is to demonstrate the application of the EM algorithm in the context of the Gaussian mixture modeling. The data are generated according to three Gaussians in the two-dimensional space, with parameters,

$$\boldsymbol{\mu}_1 = [10, 3]^{\mathrm{T}}, \quad \boldsymbol{\mu}_2 = [1, 1]^{\mathrm{T}}, \quad \boldsymbol{\mu}_3 = [5, 4]^{\mathrm{T}},$$

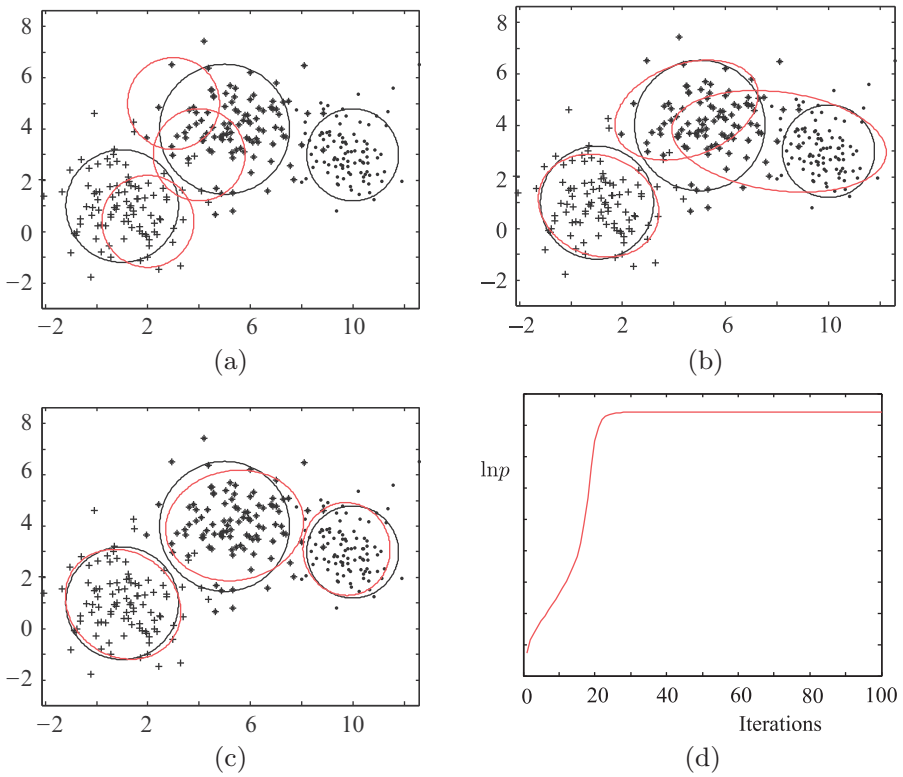

FIGURE 12.8

The curves (ellipses) indicate the 80% probability regions. The gray curves correspond to the true Gaussian clusters, of Example 12.6. The red curves correspond to (a) the initial values for the mean, covariance matrices and probabilities, (b) to the recovered by the EM algorithm mixtures after five iterations, and (c) after 30 iterations. (d) The log-likelihood as a function of the number of iterations.

and covariance matrices,

$$\Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \Sigma_2 = \begin{bmatrix} 1.5 & 0 \\ 0 & 1.5 \end{bmatrix}, \quad \Sigma_3 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix},$$

respectively. The number of the generated points is 300 with 100 points per mixture. The points are shown in Figure 12.8, together with the gray circles indicating the 80% probability regions, for each one of the clusters. The EM algorithm, comprising the steps (12.85) and (12.87)-(12.89) was run, with the following initial values,

$$\boldsymbol{\mu}_1^{(0)} = [3,5]^{\mathrm{T}}, \quad \boldsymbol{\mu}_2^{(0)} = [2,0.4]^{\mathrm{T}}, \quad \boldsymbol{\mu}_3^{(0)} = [4,3]^{\mathrm{T}},$$

and

$$\Sigma_1^{(0)} = \Sigma_2^{(0)} = \Sigma_3^{(0)} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

The probabilities were initialized to their true values $P_1^{(0)} = P_2^{(0)} = P_3^{(0)} = 1/3$. The red curves in Figure 12.8 correspond to the mixtures recovered by the EM algorithm at (a) the initial estimates (12.8a), (b) after five iterations (12.8b) and (c) after convergence, (12.8c). Figure 12.8d shows the log-likelihood as a function of the number of iterations.

Figure 12.9 corresponds to a different setup. This time, the mean values were initialized at points very far from the true ones, that is,

$$\boldsymbol{\mu}_1^{(0)} = [10, 13]^{\mathrm{T}}, \quad \boldsymbol{\mu}_2^{(0)} = [11, 12]^{\mathrm{T}}, \quad \boldsymbol{\mu}_3^{(0)} = [13, 11]^{\mathrm{T}},$$

while the covariances and probabilities were initialized as before. Observe that in this case, the EM algorithm fails to capture the true nature of the problem, having been trapped in a local minimum.

12.7.1 GAUSSIAN MIXTURE MODELING AND CLUSTERING

Clustering or unsupervised learning is an important part of machine learning, which is not treated in this book. Extensive coverage of clustering is given in, for example, [43]. However, the mixture modeling

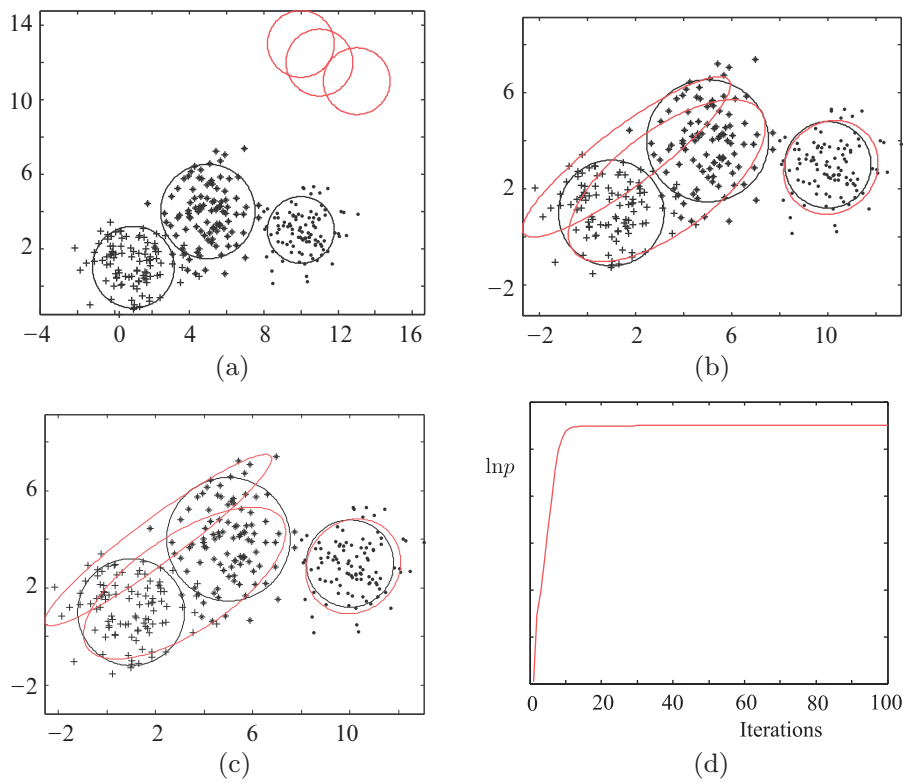

FIGURE 12.9

This is the counterpart of Figure 12.8, where now the initial values for the means are very far from the true ones. In this case, the EM fails to recover the true nature of the mixtures and has been trapped in a local minimum.

task via the EM offers us a good excuse to say a few words. Without going into formal definitions, the task of clustering is to assign a number of points, $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$, into K groups or clusters. Points that are assigned to the same cluster must be more "similar" than points which are assigned to different clusters. Some clustering algorithms need the number of clusters, K, to be provided by the user as an input variable. Other schemes treat it as a free parameter to be recovered from the data by the algorithm. The other major issue in clustering is to quantify "similarity." Different definitions end up with different clusterings. A clustering is a specific allocation of the points to clusters. In general, assigning points to clusters according to an optimality criterion is an NP-hard task, see, for example, [43]. Thus, in general, any clustering algorithm provides a suboptimal solution.

Gaussian mixture modeling is among the popular clustering algorithms. The main assumption is that the points, which belong to the same cluster, are distributed according to the same Gaussian distribution (this is how similarity is defined in this case), of unknown mean and covariance matrix. Each mixture component defines a different cluster. Thus, the goal is to run the EM algorithm over the available data points to provide, after convergence, the posterior probabilities $P(k|\boldsymbol{x}_n),\ k = 1, 2, \ldots, K$, $n = 1, 2, \ldots, N$, where each k corresponds to a cluster. Then, each point is assigned to cluster k according to the rule,

$$\text{assign } \boldsymbol{x}_n \text{ to cluster } k = \arg\min_i P(i|\boldsymbol{x}_n),\ i = 1, 2, \ldots, K.$$

The EM algorithm for clustering can be considered to be a refined version of a more primitive scheme, known as the *k-means* or *isodata* algorithm. In the EM algorithm, the posterior probability of each point, $\boldsymbol{x}_n$, with respect to each one of the clusters, k, is computed recursively. Moreover, the mean value $\boldsymbol{\mu}_k$, of the points associated with cluster k, is computed as a weighted average of *all* the training points (12.87). In contrast, in the k-means algorithm, at each iteration the posterior probability, gets a binary value in $\{1, 0\}$; for each point, $\boldsymbol{x}_n$, the Euclidean distance from all the currently available estimates of the mean values is computed, and the posterior probability is estimated according to the following rule,

$$P(k|\boldsymbol{x}_n) = \begin{cases} 1, & \text{if } ||\boldsymbol{x}_n - \boldsymbol{\mu}_k||^2 < ||\boldsymbol{x}_n - \boldsymbol{\mu}_j||^2,\ j \neq k, \\ 0, & \text{otherwise.} \end{cases}$$

The k-means algorithm is not concerned about covariance matrices. Despite its simplicity, it is not an exaggeration to say that it is the most well-known clustering algorithm, and a number of theoretical papers and improved versions have been proposed over the years, see, for example, [43]. Due to its popularity, we will take the liberty to state it in Algorithm 12.1.

Algorithm 12.1 (The k-means or isodata clustering algorithm).

- Initialize
  - Select the number of clusters K.
  - Set $\boldsymbol{\mu}_k,\ k = 1, 2, \ldots, K$, to arbitrary values.
- **For** $n = 1, 2, \ldots, N$, **Do**
  - Determine the closest cluster mean, say, $\boldsymbol{\mu}_k$, to $\boldsymbol{x}_n$.
  - Set $b(n) = k$.
- **End For**
- **For** $k = 1, 2, \ldots, K$, **Do**
  - Update $\boldsymbol{\mu}_k,\ k = 1, 2, \ldots, K$, as the mean of all the points with $b(n) = k,\ n = 1, 2, \ldots, N$.

- **End For**
- Until no change in $\boldsymbol{\mu}_k$, $k = 1, 2, \ldots, K$, occurs between two successive iterations.

The k-means algorithm can also be derived as a limiting case of the EM-scheme, for example, [36]. Note that both the EM algorithm as well as the k-means one can only recover *compact clusters*. In other words, if the points are distributed in ring-shaped clusters, then this type of clustering algorithms is not appropriate.

Figure 12.10a shows the data points generated by two Gaussians; 200 points from each one. The points are shown by red and gray colors, depending on the Gaussian that generated them. Of course in clustering, the data points are given to the algorithm without the "color" (labeling). It is up to the algorithm to make the partition in clusters. For both, the EM and the k-means algorithms, the correct number of clusters ($K = 2$) was given. The k-means was initialized with zero mean values. Figure 12.10b shows the clusters formed by the k-means and in 12.10d the clusters formed

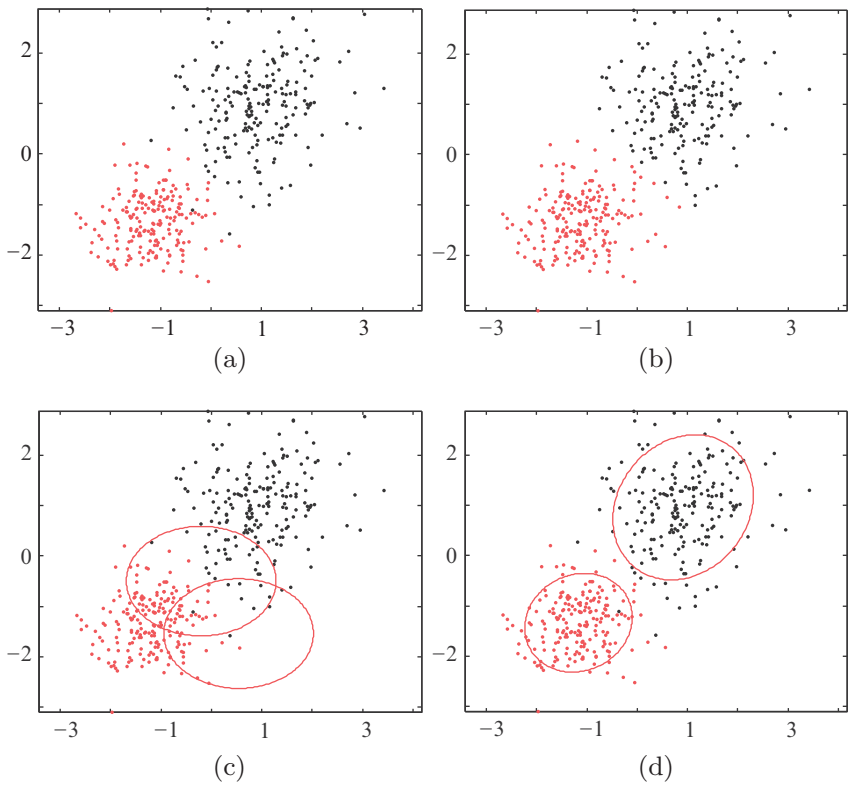

FIGURE 12.10

(a) The data points generated by two Gaussians (red and gray), (b) the recovered clusters by the k-means (red and gray), (c) The 80% probability curves for the initialization of the the EM algorithm, and (d) the final obtained by the EM algorithm Gaussians with the respective clusters.

by the EM algorithm. Figure 12.10c shows the Gaussians that were used for the initialization of the EM.

Figure 12.11 shows the respective sequence of figures, which corresponds to points obtained by the same Gaussians; however, now, there is an imbalance to the number of the points, where only 20 points spring from the first one and 200 points from the second. Observe that the k-means has a problem in recovering the true clustering structure; it attempts to make the two clusters more equally sized. A number of techniques and versions of the basic k-means scheme have been proposed to overcome its drawbacks; see, [43]. Finally, it must be stressed that both, the EM and the k-means algorithms, will always recover as many clusters as the user-defined input variable, K, dictates. In the case of the EM algorithm, this drawback is overcome when the variational EM algorithm is used, as will be discussed in Section 13.4.

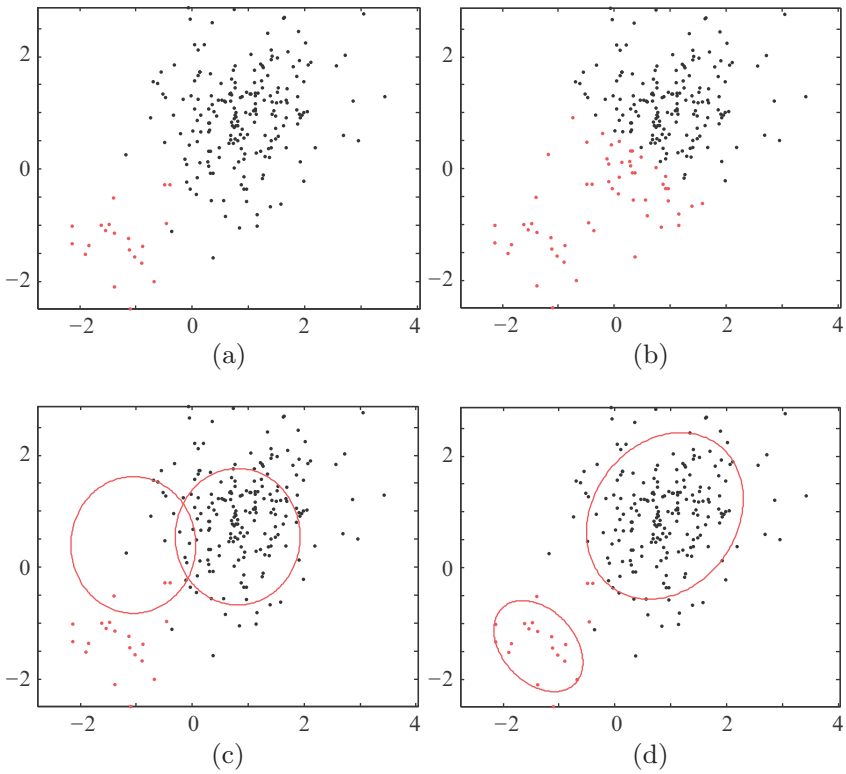

FIGURE 12.11

(a) The data points generated by two Gaussians (red and gray). One of the clusters consists of only 20 points and the other one of 200 points. (b) The recovered clusters by the k-means (red and gray). Observe that the algorithm has not identified the correct clusters, by assigning more points to the "smaller" one. (c) The 80% probability curves for the initialization of the the EM algorithm, and (d) the final Gaussians, obtained by the EM algorithm, with the respective clusters.

12.8 COMBINING LEARNING MODELS: A PROBABILISTIC POINT OF VIEW

The idea of combining different learners to boost the overall performance, by exploiting their individual characteristics was introduced in Section 7.9. We now return to this task via probabilistic arguments. Our starting consideration is that the data are distributed in different regions of the input space. Thus, it seems reasonable to fit different learning models, one for each region. This idea reminds us of the decision trees treated in Chapter 7. There, axis-aligned (linear) splits of the input space were performed. Here, the input space will be split via hyperplanes (generalizations to more general hypersurfaces are also possible) in a general position. Moreover, the main difference lies in the fact that in CARTS, the splits were of the hard-type decision rule. In the current setting, we adopt a more relaxed attitude and we are going to consider soft-type probabilistic splits, at the expense of some loss in interpretability.

The basic concept of the combining scheme of this section is illustrated in Figure 12.12. It is common to refer to each one of the K learners as an *expert*. At the heart of our modeling approach lie the so-called *gating* functions, $g_k(\boldsymbol{x})$, $k = 1, 2, \ldots, K$, which control the importance of each expert towards the final decision. These are optimally tuned during the training phase, together with the set of parameters, $\boldsymbol{\theta}_k$, $k = 1, 2, \ldots, K$, which parameterize the experts, respectively. In the general case, the gating functions are functions of the input variables. We refer to this type of modeling as *mixture of experts*. In contrast, the special type of combination, where these are parameters and not functions, that is, $g_k(\boldsymbol{x}) = g_k$, will be referred to as *mixing of learners*. We will focus on the latter case and present the method in the context of the regression and classification tasks, using linear models.

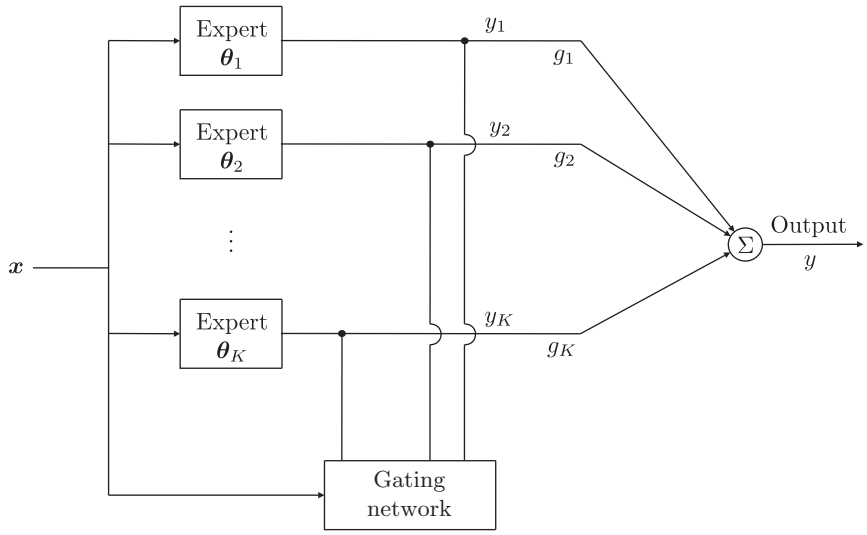

FIGURE 12.12

A block diagram of a mixture of experts. The output of each expert is weighted according to the outputs of the gating network. In the general case, these weights are considered functions of the input.

12.8.1 MIXING LINEAR REGRESSION MODELS

Our starting point is that each model is a linear regression model, $\boldsymbol{\theta}_k,\ k = 1, 2, \ldots, K$, where the dimensionality of the input space has been assumed to increase by one, to account for the intercepts and the output variables are related to the input according to our familiar equation,

$$\mathrm{y}_k = \boldsymbol{\theta}_k^T \boldsymbol{x} + \eta, \tag{12.90}$$

where, η is a Gaussian noise source, with variance σ_η^2 and it is assumed to be common for all models; extension to more general cases can be obtained in a straightforward way. For generalized linear models, $\boldsymbol{x}$ can simply be replaced by the nonlinear mapping, $\boldsymbol{\phi}(\boldsymbol{x})$. We assume that the gating parameters are interpreted as probabilities, and they will be denoted as $g_k = P_k$.

Under the previous assumptions, the following mixing model is adopted,

$$p(y; \boldsymbol{\Xi}, \boldsymbol{P}) = \sum_{k=1}^{K} P_k \mathcal{N}(y|\boldsymbol{\theta}_k^{\mathrm{T}} \boldsymbol{x}, \sigma_\eta^2), \tag{12.91}$$

where

$$\boldsymbol{\Xi} := [\boldsymbol{\theta}_1^T, \ldots, \boldsymbol{\theta}_K^T, \sigma_\eta^2]^T,\ \ \boldsymbol{P} := [P_1, \ldots, P_K]^T, \tag{12.92}$$

are the vectors of the unknown parameters, to be estimated during the training phase using the set of training points, $(y_n, \boldsymbol{x}_n),\ n = 1, 2, \ldots, N$. Because each model is designed to be "in charge" of one region in space, the corresponding parameters should be trained using input samples that originate from the respective region; note, however, that the regions are not known and have to be learned during training, as well. This is in analogy with the task of Gaussian mixture modeling; recall that during training, each observation was associated with a specific mixture component via the use of a hidden variable. In the current setting, each input sample will be associated with a specific learner. Thus, our current task is a close relative of the one treated in Section 12.7 and we could follow similar steps to derive our results. However, for the sake of variety, a slightly different route will be taken. This will also prove useful later on and at the same time fits slightly better with the jargon used for the current formulation. Instead of the indices, k_n, used in Gaussian mixture modeling, we will introduce a new set of *hidden* variables, $z_{nk} \in \{0, 1\},\ k = 1, 2, \ldots, K,\ n = 1, 2, \ldots, N$. If $z_{nk} = 1$, then sample $\boldsymbol{x}_n$ is processed by expert k. At the same time, for each n, z_{nk} becomes equal to one only for a single value of k and zero for the rest. We are now ready to write down the likelihood for the *complete* training data set[8], (y_n, z_{nk}), that is,

$$p(\boldsymbol{y}, Z; \boldsymbol{\Xi}, \boldsymbol{P}) = \prod_{n=1}^{N} \prod_{k=1}^{K} \left(P_k \mathcal{N}\left(y_n|\boldsymbol{\theta}_k^T \boldsymbol{x}_n, \sigma_\eta^2\right)\right)^{z_{nk}}, \tag{12.93}$$

where $\boldsymbol{y}$ is the vector of the output observations and Z the matrix of the respective hidden variables. The log-likelihood is readily obtained as,

$$\ln p(\boldsymbol{y}, Z; \boldsymbol{\Xi}, \boldsymbol{P}) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \ln \left(P_k \mathcal{N}\left(y_n|\boldsymbol{\theta}_k^T \boldsymbol{x}_n, \sigma_\eta^2\right)\right). \tag{12.94}$$

[8] Strictly speaking, the data set depends also on $\boldsymbol{x}_n$; to simplify notation, we only give y_n, because this is the one that is treated as a random variable.

We can now state the steps for the EM algorithm. Starting from some initial conditions, $\boldsymbol{\Xi}^{(0)}$, $\boldsymbol{P}^{(0)}$, the $j+1$ iteration is given by:

- E-Step:

$$\begin{aligned}\mathcal{Q}(\boldsymbol{\Xi},\boldsymbol{P};\boldsymbol{\Xi}^{(j)},\boldsymbol{P}^{(j)}) &= \mathbb{E}_Z\left[\ln p(\boldsymbol{y},\mathrm{Z};\boldsymbol{\Xi},\boldsymbol{P})\right]\\ &= \sum_{n=1}^{N}\sum_{k=1}^{K}\mathbb{E}[\mathrm{z}_{nk}]\ln\left(P_k\mathcal{N}\left(y_n|\boldsymbol{\theta}_k^T\boldsymbol{x}_n,\sigma_\eta^2\right)\right).\end{aligned}$$

  However,

$$\begin{aligned}\mathbb{E}[\mathrm{z}_{nk}] &= P(k|y_n;\boldsymbol{\Xi}^{(j)},\boldsymbol{P}^{(j)}),\\ &= \frac{P_k^{(j)}\mathcal{N}(y_n|\boldsymbol{\theta}_k^{(j)T}\boldsymbol{x}_n,\sigma_\eta^2)}{\sum_{i=1}^{K}P_i^{(j)}\mathcal{N}(y_n|\boldsymbol{\theta}_i^{(j)T}\boldsymbol{x}_n,\sigma_\eta^2)},\end{aligned}\tag{12.95}$$

  or

$$\begin{aligned}\mathcal{Q}(\boldsymbol{\Xi},\boldsymbol{P};\boldsymbol{\Xi}^{(j)},\boldsymbol{P}^{(j)}) = \sum_{n=1}^{N}\sum_{k=1}^{K}\gamma_{nk}\Big(\ln P_k - \frac{1}{2}\ln\sigma_\eta^2 -\\ \frac{1}{2\sigma_\eta^2}\left(y_n-\boldsymbol{\theta}_k^{\mathrm{T}}\boldsymbol{x}_n\right)^2\Big)+C,\end{aligned}\tag{12.96}$$

  where C is a constant not affecting the optimization and

$$\gamma_{nk} := P(k|y_n;\boldsymbol{\Xi}^{(j)},\boldsymbol{P}^{(j)}).$$

  As expected, (12.96) looks like (12.86).

- M-Step: The step comprises the computation of the unknown parameters via three different optimization problems.
  Gating parameters: Following exactly similar steps as for (12.89), we obtain

$$\boxed{P_k^{(j+1)} = \frac{1}{N}\sum_{n=1}^{N}\gamma_{nk}.}\tag{12.97}$$

  Learners' parameters: For each $k=1,2,\ldots,K$, we have

$$\mathcal{Q}(\boldsymbol{\Xi},\boldsymbol{P};\boldsymbol{\Xi}^{(j)},\boldsymbol{P}^{(j)}) = -\sum_{n=1}^{N}\frac{\gamma_{nk}}{2\sigma_\eta^2}\left(y_n-\boldsymbol{\theta}_k^{\mathrm{T}}\boldsymbol{x}_n\right)^2 + C_1,\tag{12.98}$$

  where C_1 includes all terms that do not depend on $\boldsymbol{\theta}_k$. Taking the gradient and equating to zero we readily obtain,

$$\sum_{n=1}^{N}\gamma_{nk}\boldsymbol{x}_n\left(y_n-\boldsymbol{x}_n^{\mathrm{T}}\boldsymbol{\theta}_k\right)=\boldsymbol{0},$$

  or, employing the input data matrix, $X^{\mathrm{T}} := [\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N]$,

$$X^T\Gamma_k\left(\boldsymbol{y}-X\boldsymbol{\theta}_k\right)=\boldsymbol{0},$$

with

$$\Gamma_k := \text{diag}\{\gamma_{1k}, \ldots, \gamma_{Nk}\},$$

and finally

$$\boxed{\boldsymbol{\theta}_k^{(j+1)} = \left(X^{\mathrm{T}}\Gamma_k X\right)^{-1} X^{\mathrm{T}}\Gamma_k \boldsymbol{y},\ k = 1, 2, \ldots, K.} \tag{12.99}$$

Equation (12.99) is the solution to a weighted LS problem, similar in form as the one met in Section 7.6, while dealing with the logistic regression. Note that the weighting matrix involves the posterior probabilities associated with the kth expert.

Noise variance: We have that,

$$\mathcal{Q}(\boldsymbol{\Xi}, \boldsymbol{P}; \boldsymbol{\Xi}^{(j)}, \boldsymbol{P}^{(j)}) = \sum_{n=1}^{N}\sum_{k=1}^{K}\gamma_{nk}\Big(-\frac{1}{2}\ln\sigma_\eta^2 - \frac{1}{2\sigma_\eta^2}\left(y_n - \boldsymbol{\theta}_k^{T(j+1)}\boldsymbol{x}_n\right)^2\Big) + C_2, \tag{12.100}$$

whose optimization with respect to σ_η^2, leads to

$$\boxed{(\sigma_\eta^{(j+1)})^2 = \frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K}\gamma_{nk}\left(y_n - \boldsymbol{\theta}_k^{T(j+1)}\boldsymbol{x}_n\right)^2.} \tag{12.101}$$

.

Mixture of experts

In mixture of experts, [25], the gating parameters are expressed in a parametric form, as functions of the input variables, $\boldsymbol{x}$. A common choice is to assume that

$$g_k(\boldsymbol{x}) := P_k(\boldsymbol{x}) = \frac{\exp\left(\boldsymbol{w}_k^{\mathrm{T}}\boldsymbol{x}\right)}{\sum_{i=1}^{K}\exp\left(\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{x}\right)}. \tag{12.102}$$

Referring to Figure 12.12, the gating weights are the outputs of the gating network, which is also excited by the same inputs as the experts. In the neural networks context, as we will see in Chapter 18, we can consider the gating network as a neural network, with activation function given by (12.102), which is known as the *softmax* activation, [5]. Note that (12.102) is of exactly the same form as (7.40), used in the multiclass logistic regression. Under such a setting, P_k in (12.96) is replaced in $P_k(\boldsymbol{x})$, and the respective M-step becomes equivalent with optimizing with respect to $\boldsymbol{w}_k,\ k = 1, 2, \ldots, K$, the following

$$\mathcal{Q}(\boldsymbol{\Xi}, \boldsymbol{P}; \boldsymbol{\Xi}^{(j)}, \boldsymbol{P}^{(j)}) = \sum_{n=1}^{N}\sum_{k=1}^{K}\gamma_{nk}\ln P_k(\boldsymbol{x}) + C_3. \tag{12.103}$$

Observe that (12.103) is of the same form as (7.42), used for the multiclass logistic regression, and optimization follows similar steps; see also, for example, [20].

A mixture of experts has been used in a number of applications with a typical one being that of inverse problems, where from the output, one has to deduce the input. However, in many cases, this

is a one-to-many task and the mixture of experts is useful to model the choice among these "many" options. For example, in [4], mixture of experts are used for tracking people in video recordings, where the mapping from the image to pose is not unique, due to occlusion.

Hierarchical mixture of experts

A direct generalization of the mixture of experts concept is to add more levels of gating functions in a hierarchical fashion, giving rise to what is known as a *hierarchial mixture of experts* (HME). The idea is illustrated in the block diagram of Figure 12.13. This architecture resembles that of trees, having the experts as leaves, the gating networks as nonterminal nodes, and the output (summing) node as the root one. A hierarchical mixture of experts divides the space into a nested set of regions, with the information combined among the experts under the control of the hierarchically placed gating networks. This hierarchy conforms with the more general idea of *conquer and divide* strategies.

Compared to decision trees, an HME evolves around soft decision rules, in contrast to the hard ones that are employed in CARTs. A hard decision, usually, leads to a loss of information. Once a decision is taken, it cannot change later on. In contrast, soft decision rules provide the luxury to the network to preserve information until a final decision is taken. For example, according to a hard decision rule, if a sample is located close to a decision surface, it will be labeled according to the label on which side it lies. However, in a soft decision rule, the information, related to the position of the point with respect to the decision surface, will be retained until the stage at which the final decision must be made, by taking into consideration more information that becomes available as the processing develops.

Note that training a mixture of experts can also be carried via a different path, by optimizing a cost function without it being necessary to employ probabilistic arguments; see, for example, [20].

12.8.2 MIXING LOGISTIC REGRESSION MODELS

Following Section 12.8.1, the combination rationale can also be applied to classification tasks. To this end, we employ the two-class logistic regression model for each one of the experts and the combination rule, given the input value $\boldsymbol{x}$, is now written as,

$$P(y;\Xi,\boldsymbol{P})=\sum_{k=1}^{K}P_k s_k^y(1-s_k)^{1-y}, \tag{12.104}$$

where the definition of logistic regression from Section 7.6 has been used, with the labels $y\in\{0,1\}$ corresponding to the two classes, ω_1 and ω_2 respectively, and

$$s_k:=\sigma(\boldsymbol{\theta}_k^{\mathrm{T}}\boldsymbol{x}), \tag{12.105}$$

denotes the output of the kth expert. As in Section 12.8.1, Ξ is the set of the unknown parameters and $\boldsymbol{P}$ the corresponding set of the gating network. Mobilizing similar arguments as for the case of linear regression, we can easily state that the likelihood of the complete data set is given by

$$P(\boldsymbol{y},Z;\Xi,\boldsymbol{P})=\prod_{n=1}^{N}\prod_{k=1}^{K}\left(P_k s_{nk}^{y_n}(1-s_{nk})^{1-y_n}\right)^{z_{nk}}, \tag{12.106}$$

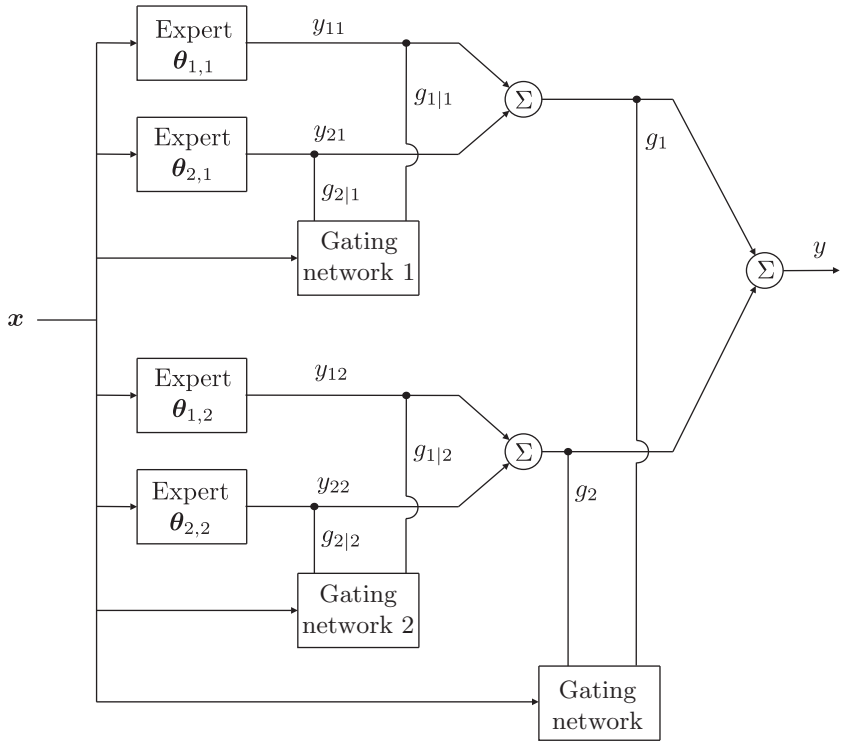

FIGURE 12.13

A block diagram of a hierarchical mixture of experts with two levels of hierarchy.

where $s_{nk} := \sigma(\boldsymbol{\theta}_k^{\mathrm{T}} \boldsymbol{x}_n)$ and $\boldsymbol{y}$ the set of labels , y_n, $n = 1, 2, \ldots, N$, of the training samples. Following the standard arguments of the EM algorithm applied on the respective log-likelihood function, it is readily shown that the E-step at the jth iteration is given by,

$$Q(\Xi, \boldsymbol{P}; \Xi^{(j)}, \boldsymbol{P}^{(j)}) = \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma_{nk} \Big(\ln P_k + y_n \ln s_{nk} + (1 - y_n) \ln(1 - s_{nk}) \Big), \tag{12.107}$$

where,

$$\gamma_{nk} = \mathbb{E}[\mathrm{z}_{nk}] = P(k|y_n, \Xi^{(j)}, \boldsymbol{P}^{(j)}) = \frac{P_k^{(j)} s_{nk}^{y_n} (1 - s_{nk})^{1 - y_n}}{\sum_{i=1}^{K} P_i^{(j)} s_{ni}^{y_n} (1 - s_{ni})^{1 - y_n}}. \tag{12.108}$$

Note that in (12.108), the notation $s_{nk}^{(j)}, s_{ni}^{(j)}$ should have been used, but we tried to unclutter it slightly.

In the M-step, minimization with respect to P_k is of the same form as as it was for the regression task, and it leads to

$$P_k^{(j+1)} = \frac{1}{N} \sum_{n=1}^{N} \gamma_{nk}. \tag{12.109}$$

To obtain the parameters for the experts, one has to resort to an iterative scheme. Observe that the only differences of (12.107) with (7.31) are (a) the presence of the term involving P_k, (b) the summation over k and (c) the existence of the multiplicative factors γ_{nk}. The first two make no difference in the optimization with respect to a single $\boldsymbol{\theta}_k$ and the latter is just a constant. Hence the optimization is similar to the one used for the two-class logistic regression in Section 7.6, with the gradient and the Hessian matrices the same except for the multiplicative factors (and the sign because, there, the negative log-likelihood was considered). The extension to multiclass case is straightforward and follows similar steps.

Example 12.7. This example demonstrates the application of a mixture of two linear regression models to a synthetic data set. The input and the output are scalars, x_n, y_n. Figure 12.14a shows the setup. The data in the input space reside in different parts of the space and in each region the input-output relation is of a different form. The goal is to estimate the two linear functions, $\theta_{1,k}x + \theta_{0,k}$, $k \in \{1, 2\}$. The EM algorithm of Subsection 12.8.1 was initialized with the true value of the noise variance σ_η^2.

Figures 12.14a, 12.14b and 12.14c show the resulting linear models after the 1st, the 7th, and finally the 15th iterations, respectively. Figure 12.14d, shows the resulting posteriors $P(k|y_n, x_n)$ (measured by the length of the bar) associated with each learner, as a function of x_n. After convergence, they are of a bimodal nature, depending on where each input sample resides in the input space. In this way, significant probability mass is assigned even to regions where data points do not exist. A smoother and more accurate, from a generalization point of view, estimate results if we let the gating parameters to be functions of the input variables themselves.

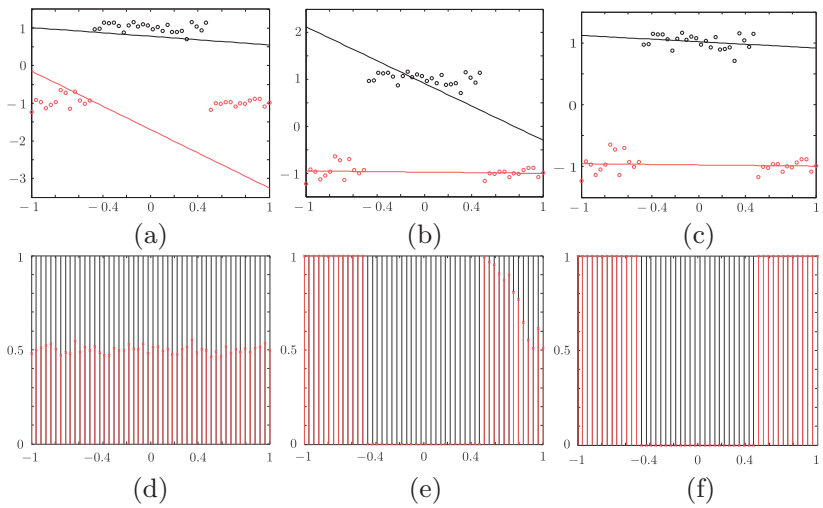

FIGURE 12.14

The two fitted lines as estimated by the EM algorithm after (a) the 1st, (b) the 7th, and (c) the 15th iterations. Figures (d)-(f) show the corresponding posterior probabilities, for each one of the training points x_n. The length of each segment is equal to the value of the respective probability.

PROBLEMS

12.1 Show that if

$$p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_z, \Sigma_z),$$

and

$$p(\boldsymbol{t}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{t}|A\boldsymbol{z}, \Sigma_{t|z}),$$

then

$$\mathbb{E}[\boldsymbol{z}|\boldsymbol{t}] = (\Sigma_z^{-1} + A^T\Sigma_{t|z}^{-1}A)^{-1}(A^T\Sigma_{t|z}^{-1}\boldsymbol{t} + \Sigma_z^{-1}\boldsymbol{\mu}_z).$$

12.2 Prove that the binomial and beta distributions are conjugate pairs with respect to the mean value.

12.3 Show that the normalizing constant C in the Dirichlet pdf

$$\text{Dir}(\boldsymbol{x}|\boldsymbol{a}) = C\prod_{k=1}^{K} x_k^{a_k-1}, \ \sum_{k=1}^{K} x_k = 1,$$

is given by

$$C = \frac{\Gamma(a_1 + a_2 + \cdots + a_K)}{\Gamma(a_1)\Gamma(a_2)\ldots\Gamma(a_K)}.$$

Hint. Use the property $\Gamma(a+1) = a\Gamma(a)$.

(a) Use induction. Because the proposition is true for $k = 2$ (beta distribution), assume that it is true for $k = K-1$, and prove that it will be true for $k = K$.

(b) Note that due to the constraint $\sum_{k=1}^{K} x_k = 1$, only $K-1$ of the variables are independent. So, basically the Dirichlet pdf implies that

$$p(x_1, x_2, \ldots, x_{K-1}) = C\prod_{k=1}^{K-1} x_k^{a_k-1}\left(1 - \sum_{k=1}^{K-1} x_k\right)^{a_K-1}.$$

12.4 Show that $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \Sigma)$ for known Σ is of an exponential form and that its conjugate prior is also Gaussian.

12.5 Show that the conjugate prior of the multivariate Gaussian with respect to the precision matrix, Q, is a Wishart distribution.

12.6 Show that the conjugate prior of the univariate Gaussian $\mathcal{N}(x|\mu, \sigma^2)$ with respect to the mean and the precision $\beta = \frac{1}{\sigma^2}$, is the Gaussian-gamma product

$$p(\mu, \beta; \lambda, \boldsymbol{v}) = \mathcal{N}\left(\mu\Big|\frac{v_2}{\lambda}, (\lambda\beta)^{-1}\right)\text{Gamma}\left(\beta\Big|\frac{\lambda+1}{2}, \frac{v_1}{2} - \frac{v_2^2}{2\lambda}\right),$$

where $\boldsymbol{v} := [v_1, v_2]^{\text{T}}$.

12.7 Show that the multivariate Gaussian $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, Q^{-1})$ has as a conjugate prior, with respect to the mean and the precision matrix, Q, the Gaussian-Wishart product.

12.8 Show that the distribution

$$P(x|\mu) = \mu^x(1-\mu)^{1-x},\ x \in \{0,1\},$$

is of an exponential form and derive its conjugate prior with respect to μ.

12.9 Show that estimating an unknown pdf by maximizing the respective entropy, subject to a set of empirical expectations, results in a pdf that belongs to the exponential family.

12.10 Let $\mathbf{x} \in \mathbb{R}^l$ be a random vector following the normal $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \Sigma)$. Consider $\boldsymbol{x}_n,\ n = 1, 2, \ldots, N$, to be i.i.d. observations. If the prior for $\boldsymbol{\mu}$ follows $\mathcal{N}(\boldsymbol{\mu}|\boldsymbol{\mu}_0, \Sigma_0)$, show that the posterior $p(\boldsymbol{\mu}|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$ is normal $\mathcal{N}(\boldsymbol{\mu}|\tilde{\boldsymbol{\mu}}, \tilde{\Sigma})$ with

$$\tilde{\Sigma}^{-1} = \Sigma_0^{-1} + N\Sigma^{-1},$$

and

$$\tilde{\mu} = \tilde{\Sigma}(\Sigma_0^{-1}\boldsymbol{\mu}_0 + N\Sigma^{-1}\bar{\boldsymbol{x}}),$$

where $\bar{\boldsymbol{x}} = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{x}_n$.

12.11 If $\mathcal{X}$ is the set of observed variables and $\mathcal{X}^l$ the set of the corresponding latent ones, show that

$$\frac{\partial \ln p(\mathcal{X};\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \mathbb{E}\left[\frac{\partial \ln p(\mathcal{X}, \mathcal{X}^l;\boldsymbol{\xi})}{\partial \boldsymbol{\xi}}\right],$$

where $\mathbb{E}[\cdot]$ is with respect to $p(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi})$ and $\boldsymbol{\xi}$ is an unknown vector parameter. Note that if one fixes the value of $\boldsymbol{\xi}$ in $p(\mathcal{X}^l|\mathcal{X};\boldsymbol{\xi})$, then one has obtained the M-step of the EM algorithm.

12.12 Show that the Kullback-Leibler divergence $\mathrm{KL}(p \parallel q)$ is a nonnegative quantity.

Hint. Recall that $\ln(\cdot)$ is a concave function and use Jensen's inequality, that is,

$$f\left(\int g(\boldsymbol{x})p(\boldsymbol{x})d\boldsymbol{x}\right) \leq \int f(g(\boldsymbol{x}))p(\boldsymbol{x})d\boldsymbol{x},$$

where $p(\boldsymbol{x})$ is a pdf and f is a convex function.

12.13 Let $\boldsymbol{y} \in \mathbb{R}^N$, $\theta \in \mathbb{R}^l$ and Φ a matrix of appropriate dimensions. Derive the expected value of $\|\boldsymbol{y} - \Phi\theta\|^2$ with respect to θ, given $\mathbb{E}[\theta]$ and the corresponding covariance matrix Σ_θ.

12.14 Derive recursions (12.87)–(12.89).

MATLAB Exercises

12.15 Sample $N = 20$ equally spaced points x_n in the interval $[0, 2]$. Create the output samples, y_n, according to the nonlinear model of Example 12.1, where the noise variance is set equal to $\sigma_\eta^2 = 0.05$.

(a) Let the parameters of the Gaussian prior be $\boldsymbol{\theta}_0 = [0.2, -1, 0.9, 0.7, -0.2]^T$ and $\Sigma_\theta = 0.1I$. Compute the covariance matrix and the mean of the posterior Gaussian distribution using Eq. (12.19) and Eq. (12.20), respectively. Then, select randomly $K = 20$ points x_k in the interval $[0, 2]$. Compute the predictions for the mean values, μ_y, and the associated variances, σ_y^2, utilizing Eq. (12.22) and Eq. (12.23), respectively. Plot the graph of the true function together with the predicted mean values, μ_y, and use MATLAB's "errorbar" function to show the confidence intervals on these predictions. Repeat the experiment again, using $N = 500$ points, and try different values of σ_η^2, to notice the change on the estimated confidence intervals.

(b) Repeat the previous experiment using a randomly chosen value for $\boldsymbol{\theta}_0$ and different values on the parameters, for example $\sigma_\eta^2 = 0.05$, or $\sigma_\eta^2 = 0.15$, $\sigma_\theta^2 = 0.1$, or $\sigma_\theta^2 = 2$, and $N = 500$ or $N = 20$.

(c) Repeat the experiment once more using the alternative model as in (c) of Example 12.1.

12.16 Consider Example 12.1 as before. Sample $N = 500$ equally spaced points x_n in the interval $[0, 2]$. Create the output samples, y_n, according to the nonlinear model of the example, where the noise variance is set equal to $\sigma_\eta^2 = 0.05$. Implement the linear regression EM algorithm of Section 12.6. Assume the correct number of parameters. Then repeat Example 12.5. After the convergence of the EM, sample ten points, x_k, randomly, in the same interval as before and compute the predictive means μ_y and variances σ_y^2. Plot the true signal curve, the predictive means μ_y, and the respective confidence intervals using Matlab's "errorbar" function. Repeat the EM run, using different initial values and an incorrect number of parameters. Comment on the results.

12.17 Generate 100 data points from each of the three two-dimensional Gaussian distributions of Example 12.6. Plot the data points along with the confidence ellipsoids for each Gaussian with coverage probability 80%. Implement the Gaussian mixture model via the EM algorithm, whose steps are described in Eqs. (12.85)–(12.89). Moreover, compute the log-likelihood function in every iteration of the EM algorithm using Eq. (12.82).

(a) In separate figures (always containing the data), plot the ellipsoids of the Gaussian distributions estimated by the EM algorithm during iterations $j = 1$, $j = 5$, and $j = 30$, and the log-likelihood function versus the number of iterations.

(b) Repeat the same experiment after bringing the cluster means closer together. Compare the results.

12.18 Generate 100 data points from each of the following two-dimensional Gaussian distributions with parameters

$$\boldsymbol{\mu}_1^T = [0.9, 1.02]^T, \quad \boldsymbol{\mu}_2^T = [-1.2, -1.3]^T$$

and

$$\Sigma_1 = \begin{bmatrix} 0.5 & 0.081 \\ 0.081 & 0.7 \end{bmatrix}, \quad \Sigma_2 = \begin{bmatrix} 0.4 & 0.02 \\ 0.02 & 0.3 \end{bmatrix}.$$

Plot the data points using different colors for the two Gaussian distributions. Implement the k-means algorithm, presented in Algorithm 12.1.

(a) Run the k-means algorithm for $K = 2$ and plot the results. Run, also, the Gaussian mixtures EM of the previous exercise and plot the 80% probability confidence ellipsoids to compare the results.

(b) Now, sample $N_1 = 100$ and $N_2 = 20$ points from each distribution and repeat the experiment to reproduce the results of Figure 12.11d.

(c) Try different configurations and play with K different than the true number of clusters. Comment on the results.

(d) Play with different initialization points and also try points which are too far from the true mean values of the clusters. Comment on the results.

12.19 Generate 50 equidistant input data points in the interval $[-1, 1]$. Assume two linear regression models, the first with scale 0.005 and intercept -1 and the second with scale 0.018 and

intercept 1. Generate observations from these two models by using the first model for the input points in the interval $[-0.5, 0.5]$, and the second model for the inputs in the interval $[-1, -0.5] \cup [0.5, 1]$. Also, add Gaussian noise of zero-mean and variance 0.01. Next, implement the EM algorithm developed in Section 12.8.1. Initialize the noise precision β to its true value. For iterations 1, 5, and 30, plot the data points and the estimated linear functions $\theta_{1,k}x + \theta_{0,k}$, $k \in \{1, 2\}$ of the models, to reproduce the results of Figure 12.14.

12.9 APPENDIX TO CHAPTER 12

In this appendix, a number of results concerning the Gaussian pdf are derived. The reader is advised to work on these derivations to get familiar with the tools that are heavily used in Bayesian inference. Because the derived formulas are applicable to different parts of the book and to different variables, we denote the involved random vectors as $\mathbf{z}$ and $\mathbf{t}$ and one can substitute notation accordingly, depending on the notational needs for each case.

12.9.1 PDFs WITH EXPONENT OF QUADRATIC FORM

Let

$$p(z) = \exp\big(F(z)\big), \tag{12.110}$$

where

$$F(z) = -\frac{1}{2}z^{\mathrm{T}}Qz + z^{\mathrm{T}}\boldsymbol{p} + C, \tag{12.111}$$

where C is a constant and $Q = Q^{\mathrm{T}}$ and invertible. We rewrite (12.111) as

$$\begin{aligned} F(z) &= -\frac{1}{2}z^{\mathrm{T}}Qz + z^{\mathrm{T}}QQ^{-1}\boldsymbol{p} + C \\ &= -\frac{1}{2}\left(z - Q^{-1}\boldsymbol{p}\right)^{\mathrm{T}} Q\left(z - Q^{-1}\boldsymbol{p}\right) + \frac{1}{2}\boldsymbol{p}^{\mathrm{T}}Q^{-1}\boldsymbol{p} + C. \end{aligned} \tag{12.112}$$

From (12.112, 12.110) we get

$$p(z) = \exp\left(\frac{1}{2}\boldsymbol{p}^{\mathrm{T}}Q^{-1}\boldsymbol{p} + C\right)\exp\left(-\frac{1}{2}(z-\boldsymbol{\mu})^{\mathrm{T}}\Sigma^{-1}(z-\boldsymbol{\mu})\right),$$

where

$$\boldsymbol{\mu} = Q^{-1}\boldsymbol{p}, \tag{12.113}$$

and

$$\Sigma = Q^{-1}. \tag{12.114}$$

Because $p(z)$ has to integrate to one then, necessarily,

$$\exp\left(\frac{1}{2}\boldsymbol{p}^{\mathrm{T}}Q^{-1}\boldsymbol{p} + C\right) = \frac{1}{(2\pi)^{l/2}|\Sigma|^{1/2}},$$

where l is the dimensionality of the space where z lies.

12.9.2 THE CONDITIONAL FROM THE JOINT GAUSSIAN PDF

Let $\mathbf{z} \in \mathbb{R}^{l_1}$ and $\mathbf{t} \in \mathbb{R}^{l_2}$ be two jointly Gaussian vectors, with $l = l_1 + l_2$. Let

$$\boldsymbol{\Phi} = \begin{bmatrix} \mathbf{z} \\ \mathbf{t} \end{bmatrix}, \quad \mathbb{E}[\boldsymbol{\Phi}] = \begin{bmatrix} \boldsymbol{\mu}_z \\ \boldsymbol{\mu}_t \end{bmatrix} := \boldsymbol{\mu}_\phi.$$

Then

$$p(\boldsymbol{z}, \boldsymbol{t}) = p(\boldsymbol{\phi}) = \frac{1}{(2\pi)^{l/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{\phi} - \boldsymbol{\mu}_\phi)^{\mathrm{T}} \Sigma_\phi^{-1} (\boldsymbol{\phi} - \boldsymbol{\mu}_\phi)\right), \tag{12.115}$$

where

$$\Sigma_\phi = \mathbb{E}\left[(\boldsymbol{\Phi} - \boldsymbol{\mu}_\phi)(\boldsymbol{\Phi} - \boldsymbol{\mu}_\phi)^T\right] = \begin{bmatrix} \Sigma_z & \Sigma_{zt} \\ \Sigma_{tz} & \Sigma_t \end{bmatrix}, \tag{12.116}$$

with

$$\Sigma_z = \mathbb{E}\left[(\mathbf{z} - \boldsymbol{\mu}_z)(\mathbf{z} - \boldsymbol{\mu}_z)^T\right], \ \Sigma_t = \mathbb{E}\left[(\mathbf{t} - \boldsymbol{\mu}_t)(\mathbf{t} - \boldsymbol{\mu}_t)^T\right], \tag{12.117}$$

and

$$\Sigma_{zt} = \mathbb{E}\left[(\mathbf{z} - \boldsymbol{\mu}_z)(\mathbf{t} - \boldsymbol{\mu}_t)^T\right] = \Sigma_{tz}^T. \tag{12.118}$$

We will prove that $p(\boldsymbol{z}|\boldsymbol{t})$ is also Gaussian. In a similar way, one proves the same for $p(\boldsymbol{t}|\boldsymbol{z})$. To this end, we will show that (12.115) is quadratic with respect to $\boldsymbol{z}$. Indeed,

$$Q_\phi := \Sigma_\phi^{-1} = \begin{bmatrix} \Sigma_z & \Sigma_{zt} \\ \Sigma_{tz} & \Sigma_t \end{bmatrix}^{-1} := \begin{bmatrix} Q_z & Q_{zt} \\ Q_{tz} & Q_t \end{bmatrix}, \tag{12.119}$$

where Q_z is $l_1 \times l_1$, Q_t is $l_2 \times l_2$ and $Q_{zt} = Q_{tz}^{\mathrm{T}}$ are $l_1 \times l_2$ matrices. The exponent in (12.115) becomes

$$\mathrm{EXP} = -\frac{1}{2}[\boldsymbol{z}^{\mathrm{T}} - \boldsymbol{\mu}_z^{\mathrm{T}}, \boldsymbol{t}^{\mathrm{T}} - \boldsymbol{\mu}_t^{\mathrm{T}}] \begin{bmatrix} Q_z & Q_{zt} \\ Q_{tz} & Q_t \end{bmatrix} \begin{bmatrix} \boldsymbol{z} - \boldsymbol{\mu}_z \\ \boldsymbol{t} - \boldsymbol{\mu}_t \end{bmatrix}, \tag{12.120}$$

which after some trivial algebra becomes

$$\mathrm{EXP} = -\frac{1}{2}\boldsymbol{z}^{\mathrm{T}} Q_z \boldsymbol{z} + \boldsymbol{z}^{\mathrm{T}} Q_z \boldsymbol{\mu}_z - \boldsymbol{z}^{\mathrm{T}} Q_{zt}(\boldsymbol{t} - \boldsymbol{\mu}_t) + C(\boldsymbol{t}), \tag{12.121}$$

where

$$C(\boldsymbol{t}) = -\frac{1}{2}(\boldsymbol{t} - \boldsymbol{\mu}_t)^{\mathrm{T}} Q_t (\boldsymbol{t} - \boldsymbol{\mu}_t) + \boldsymbol{\mu}_z^{\mathrm{T}} Q_{zt}(\boldsymbol{t} - \boldsymbol{\mu}_t) - \frac{1}{2}\boldsymbol{\mu}_z^{\mathrm{T}} Q_z \boldsymbol{\mu}_z, \tag{12.122}$$

which is considered constant, because in the conditional $p(\boldsymbol{z}|\boldsymbol{t})$ we fix the values of $\boldsymbol{t}$. It is readily seen that the exponent is quadratic with respect to $\boldsymbol{z}$ and of the form given in (12.111),

$$\mathrm{EXP} = -\frac{1}{2}\boldsymbol{z}^{\mathrm{T}} Q_z \boldsymbol{z} + \boldsymbol{z}^{\mathrm{T}}\left(Q_z \boldsymbol{\mu}_z - Q_{zt}(\boldsymbol{t} - \boldsymbol{\mu}_t)\right) + C(\boldsymbol{t}). \tag{12.123}$$

Hence, combining with (12.113, 12.114) we get for the conditional mean

$$\boldsymbol{\mu}_{z|t} := \mathbb{E}[\mathbf{z}|\boldsymbol{t}] = \boldsymbol{\mu}_z - Q_z^{-1} Q_{zt}(\boldsymbol{t} - \boldsymbol{\mu}_t) \tag{12.124}$$

and the conditional covariance

$$\Sigma_{z|t} = Q_z^{-1}. \tag{12.125}$$

It suffices to compute Q_z and Q_{zt} in terms of the known Σ_z, Σ_t, Σ_{zt}. From (12.119) and using the matrix inversion lemmas (Appendix A.1) we get

$$Q_z = \Sigma_z^{-1} + \Sigma_z^{-1}\Sigma_{zt}\left(\Sigma_t - \Sigma_{tz}\Sigma_z^{-1}\Sigma_{zt}\right)^{-1}\Sigma_{tz}\Sigma_z^{-1} \tag{12.126}$$

$$= (\Sigma_z - \Sigma_{zt}\Sigma_t^{-1}\Sigma_{tz})^{-1} \tag{12.127}$$

$$Q_{zt} = -\Sigma_z^{-1}\Sigma_{zt}\left(\Sigma_t - \Sigma_{tz}\Sigma_z^{-1}\Sigma_{zt}\right)^{-1} \tag{12.128}$$

$$= -\left(\Sigma_z - \Sigma_{zt}\Sigma_t^{-1}\Sigma_{tz}\right)^{-1}\Sigma_{zt}\Sigma_t^{-1}. \tag{12.129}$$

Thus, from (12.125) and (12.127) we obtain

$$\Sigma_{z|t} = Q_z^{-1} = \Sigma_z - \Sigma_{zt}\Sigma_t^{-1}\Sigma_{tz}, \tag{12.130}$$

and combining (12.127) and (12.129) with (12.124) we obtain

$$\boldsymbol{\mu}_{z|t} = \boldsymbol{\mu}_z + \left(\Sigma_z - \Sigma_{zt}\Sigma_t^{-1}\Sigma_{tz}\right)\left(\Sigma_z - \Sigma_{zt}\Sigma_t^{-1}\Sigma_{tz}\right)^{-1}\Sigma_{zt}\Sigma_t^{-1}(\boldsymbol{t} - \boldsymbol{\mu}_t),$$

or

$$\boldsymbol{\mu}_{z|t} = \boldsymbol{\mu}_z + \Sigma_{zt}\Sigma_t^{-1}(\boldsymbol{t} - \boldsymbol{\mu}_t). \tag{12.131}$$

12.9.3 THE MARGINAL FROM THE JOINT GAUSSIAN PDF

Our next goal is to compute the marginal pdf of either of the two involved jointly Gaussian variables, for example,

$$p(\boldsymbol{t}) = \int p(\boldsymbol{z}, \boldsymbol{t})d\boldsymbol{z},$$

and show that it is also Gaussian.

Similar arguments will follow for the computation of $p(\boldsymbol{z})$. The exponent of the joint pdf from (12.122), (12.123), and (12.124) becomes

$$\begin{aligned}\text{EXP} = &-\frac{1}{2}(\boldsymbol{z} - \boldsymbol{\mu}_{z|t})^{\mathrm{T}}Q_z(\boldsymbol{z} - \boldsymbol{\mu}_{z|t}) + \frac{1}{2}\boldsymbol{\mu}_{z|t}^{\mathrm{T}}Q_z\boldsymbol{\mu}_{z|t} \\ &-\frac{1}{2}(\boldsymbol{t} - \boldsymbol{\mu}_t)^{\mathrm{T}}Q_t(\boldsymbol{t} - \boldsymbol{\mu}_t) + \boldsymbol{\mu}_z^{\mathrm{T}}Q_{zt}(\boldsymbol{t} - \boldsymbol{\mu}_t) - \frac{1}{2}\boldsymbol{\mu}_z^{\mathrm{T}}Q_z\boldsymbol{\mu}_z \\ = &-\frac{1}{2}(\boldsymbol{z} - \boldsymbol{\mu}_{z|t})^{\mathrm{T}}Q_z(\boldsymbol{z} - \boldsymbol{\mu}_{z|t}) + F(\boldsymbol{t}).\end{aligned} \tag{12.132}$$

Observe that both terms in (12.132) are functions of $\boldsymbol{t}$. However, combining (12.115), (12.120), (12.132) we get

$$p(\boldsymbol{t}) = \int p(\boldsymbol{z}, \boldsymbol{t})d\boldsymbol{z} \propto \exp\left(F(\boldsymbol{t})\right), \tag{12.133}$$

because

$$\int \exp\left(-\frac{1}{2}(z-\boldsymbol{\mu}_{z|t})^T Q_z (z-\boldsymbol{\mu}_{z|t})\right) dz,$$

being the integral of the exponent of a Gaussian will depend on $|Q_z|$ only and it is independent of $\boldsymbol{\mu}_{z|t}$, which is a function of $\boldsymbol{t}$. Hence, it suffices to check in (12.133) if $F(\boldsymbol{t})$ is quadratic in $\boldsymbol{t}$. From the respective definition in (12.132) we obtain

$$F(\boldsymbol{t}) = -\frac{1}{2}(\boldsymbol{t}-\boldsymbol{\mu}_t)^{\mathrm{T}} Q_t (\boldsymbol{t}-\boldsymbol{\mu}_t) + \boldsymbol{\mu}_z^{\mathrm{T}} Q_{zt} (\boldsymbol{t}-\boldsymbol{\mu}_t) + \frac{1}{2}\boldsymbol{\mu}_{z|t}^{\mathrm{T}} Q_z \boldsymbol{\mu}_{z|t} - \frac{1}{2}\boldsymbol{\mu}_z^{\mathrm{T}} Q_z \boldsymbol{\mu}_z, \tag{12.134}$$

and using (12.124) we can readily check that $p(\boldsymbol{t})$ is of quadratic form. However, we need not manipulate further (12.134) to find the respective mean and covariance matrix. Because the original joint pdf was specified, then from (12.117) $\boldsymbol{\mu}_t$ and Σ_t are readily available. No doubt, these are the same values that would result by manipulating (12.134), and the reader can verify it as an exercise.

12.9.4 THE POSTERIOR FROM GAUSSIAN PRIOR AND CONDITIONAL PDFs

Let

$$p(z) = \mathcal{N}(z|\boldsymbol{\mu}_z, \Sigma_z), \tag{12.135}$$

and

$$p(\boldsymbol{t}|z) = \mathcal{N}(\boldsymbol{t}|Az, \Sigma_{t|z}). \tag{12.136}$$

The goal is to show that $p(z|\boldsymbol{t})$ and $p(\boldsymbol{t})$ are also Gaussians.

Path 1: We will first show that $\mathbf{z}$ and $\mathbf{t}$ are jointly Gaussian. We have that

$$p(z,\boldsymbol{t}) = p(\boldsymbol{t}|z)p(z), \tag{12.137}$$

which from (12.135, 12.136) turns out to be

$$p(z,\boldsymbol{t}) \propto \exp\big(F(z,\boldsymbol{t})\big),$$

where

$$\begin{aligned} F(z,\boldsymbol{t}) = &-\frac{1}{2}(\boldsymbol{t}-Az)^{\mathrm{T}} \Sigma_{t|z}^{-1} (\boldsymbol{t}-Az) \\ &-\frac{1}{2}(z-\boldsymbol{\mu}_z)^{\mathrm{T}} \Sigma_z^{-1} (z-\boldsymbol{\mu}_z). \end{aligned} \tag{12.138}$$

Then, after some simple algebraic manipulations we have,

$$\begin{aligned} F(z,\boldsymbol{t}) = &-\frac{1}{2}\boldsymbol{t}^T \Sigma_{t|z}^{-1} \boldsymbol{t} - \frac{1}{2} z^T (\Sigma_z^{-1} + A^T \Sigma_{t|z}^{-1} A) z \\ &+ \boldsymbol{t}^T \Sigma_{t|z}^{-1} A z + z^T \Sigma_z^{-1} \boldsymbol{\mu}_z - \frac{1}{2}\boldsymbol{\mu}_z^T \Sigma_z^{-1} \boldsymbol{\mu}_z, \end{aligned}$$

or

$$F(z,\boldsymbol{t}) = -\frac{1}{2}[z^T, \boldsymbol{t}^T] \begin{bmatrix} \Sigma_z^{-1} + A^T \Sigma_{t|z}^{-1} A & -A^T \Sigma_{t|z}^{-1} \\ -\Sigma_{t|z}^{-1} A & \Sigma_{t|z}^{-1} \end{bmatrix} \begin{bmatrix} z \\ \boldsymbol{t} \end{bmatrix}$$

$$+[\boldsymbol{z}^T, \boldsymbol{t}^T] \begin{bmatrix} \Sigma_z^{-1}\boldsymbol{\mu}_z \\ 0 \end{bmatrix} + C, \tag{12.139}$$

which is obviously in the quadratic form of (12.111) for the joint variables

$$\boldsymbol{\phi} = \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{t} \end{bmatrix}.$$

Then we already know that $p(\boldsymbol{z}|\boldsymbol{t})$ will also be Gaussian, with mean and covariance matrix resulting by combining (12.124), (12.125),

$$\boldsymbol{\mu}_{z|t} = \boldsymbol{\mu}_z + (\Sigma_z^{-1} + A^{\mathrm{T}}\Sigma_{t|z}^{-1}A)^{-1}A^{\mathrm{T}}\Sigma_{t|z}^{-1}(\boldsymbol{t} - \boldsymbol{\mu}_t) \tag{12.140}$$

$$\Sigma_{z|t} = (\Sigma_z^{-1} + A^{\mathrm{T}}\Sigma_{t|z}^{-1}A)^{-1}. \tag{12.141}$$

It suffices to compute $\boldsymbol{\mu}_t$. From (12.139) and (12.113) we have

$$\mathbb{E}[\boldsymbol{\phi}] = \begin{bmatrix} \boldsymbol{\mu}_z \\ \boldsymbol{\mu}_t \end{bmatrix} = \begin{bmatrix} \Sigma_z^{-1} + A^T\Sigma_{t|z}^{-1}A & -A^T\Sigma_{t|z}^{-1} \\ -\Sigma_{t|z}^{-1}A & \Sigma_{t|z}^{-1} \end{bmatrix}^{-1} \begin{bmatrix} \Sigma_z^{-1}\boldsymbol{\mu}_z \\ 0 \end{bmatrix},$$

or taking into consideration the matrix inversion lemma with respect to the low partition and some algebraic manipulations, we obtain

$$\begin{bmatrix} \boldsymbol{\mu}_z \\ \boldsymbol{\mu}_t \end{bmatrix} = \begin{bmatrix} \Sigma_z & \Sigma_z A^T \\ A\Sigma_z & \Sigma_{t|z} + A\Sigma_z A^T \end{bmatrix} \begin{bmatrix} \Sigma_z^{-1}\boldsymbol{\mu}_z \\ 0 \end{bmatrix}, \tag{12.142}$$

or

$$\boldsymbol{\mu}_t = A\boldsymbol{\mu}_z$$

$$\boldsymbol{\mu}_{z|t} = \boldsymbol{\mu}_z + \left(\Sigma_z^{-1} + A^{\mathrm{T}}\Sigma_{t|z}^{-1}A\right)^{-1} A^{\mathrm{T}}\Sigma_{t|z}^{-1}(\boldsymbol{t} - A\boldsymbol{\mu}_z) \tag{12.143}$$

$$\Sigma_{z|t} = \left(\Sigma_z^{-1} + A^{\mathrm{T}}\Sigma_{t|z}^{-1}A\right)^{-1}. \tag{12.144}$$

Using the matrix inversion lemma from Appendix A.1,

$$\left(A^{-1} + B^{\mathrm{T}}C^{-1}B\right)^{-1}B^{\mathrm{T}}C^{-1} = AB^{\mathrm{T}}\left(BAB^{\mathrm{T}} + C\right)^{-1}$$

we get

$$\boldsymbol{\mu}_{z|t} = \boldsymbol{\mu}_z + \Sigma_z A^{\mathrm{T}}(\Sigma_{t|z} + A\Sigma_z A^{\mathrm{T}})^{-1}(\boldsymbol{t} - A\boldsymbol{\mu}_z). \tag{12.145}$$

Also, by applying Woodbury's identity (Appendix A.1) in (12.144) we obtain an alternative expression for $\Sigma_{z|t}$,

$$\Sigma_{z|t} = \Sigma_z - \Sigma_z A^{\mathrm{T}}(\Sigma_{t|z} + A\Sigma_z A^{\mathrm{T}})^{-1}A\Sigma_z. \tag{12.146}$$

It now remains to derive the marginal $p(\boldsymbol{t})$. Because $\mathbf{z}$, $\mathbf{t}$ are jointly Gaussian, $p(\boldsymbol{t})$ is also Gaussian with mean

$$\boldsymbol{\mu}_t = A\boldsymbol{\mu}_z, \tag{12.147}$$

and using $\Sigma_\phi := Q_\phi^{-1}$ in (12.142)

$$\Sigma_t = \Sigma_{t|z} + A\Sigma_z A^{\mathrm{T}}. \tag{12.148}$$

Finally, one can show via the use of the matrix inversion lemmas that $\boldsymbol{\mu}_{z|t}$ in (12.143) is equal to (Problem 12.1)

$$\boldsymbol{\mu}_{z|t} = \left(\Sigma_z^{-1} + A^{\mathrm{T}}\Sigma_{t|z}^{-1}A\right)^{-1}\left(A^{\mathrm{T}}\Sigma_{t|z}^{-1}\boldsymbol{t} + \Sigma_z^{-1}\boldsymbol{\mu}_z\right). \tag{12.149}$$

Path 2: There is another more direct path to obtain $p(\boldsymbol{z}|\boldsymbol{t})$, if one wants to bypass the joint distribution. From (12.135, 12.136) we obtain that

$$p(\boldsymbol{z}|\boldsymbol{t}) \propto \exp\left(-\frac{1}{2}\boldsymbol{z}^{\mathrm{T}}\Sigma_z^{-1}\boldsymbol{z} + \boldsymbol{z}^{\mathrm{T}}\Sigma_z^{-1}\boldsymbol{\mu}_z\right) \times \exp\left(-\frac{1}{2}\boldsymbol{z}^{\mathrm{T}}A^{\mathrm{T}}\Sigma_{t|z}^{-1}A\boldsymbol{z} + \boldsymbol{z}^{\mathrm{T}}A^{\mathrm{T}}\Sigma_{t|z}^{-1}\boldsymbol{t}\right),$$

or

$$\begin{aligned} p(\boldsymbol{z}|\boldsymbol{t}) \propto \exp\Big(&-\frac{1}{2}\boldsymbol{z}^{\mathrm{T}}(\Sigma_z^{-1} + A^{\mathrm{T}}\Sigma_{t|z}^{-1}A)\boldsymbol{z} \\ &+ \boldsymbol{z}^{\mathrm{T}}\left(\Sigma_z^{-1}\boldsymbol{\mu}_z + A^{\mathrm{T}}\Sigma_{t|z}^{-1}\boldsymbol{t}\right)\Big) \\ := \exp&\left(-\frac{1}{2}\boldsymbol{z}^{\mathrm{T}}Q_{z|t}\boldsymbol{z} + \boldsymbol{z}^{\mathrm{T}}\boldsymbol{p}_{z|t}\right), \end{aligned}$$

which is of quadratic form and using (12.113, 12.114) leads to the conclusion that

$$\Sigma_{z|t} = Q_{z|t}^{-1} = \left(\Sigma_z^{-1} + A^{\mathrm{T}}\Sigma_{t|z}^{-1}A\right)^{-1}$$

and

$$\begin{aligned} \boldsymbol{\mu}_{z|t} &= (\Sigma_z^{-1} + A^{\mathrm{T}}\Sigma_{t|z}^{-1}A)^{-1}\boldsymbol{p}_{z|t} \\ &= (\Sigma_z^{-1} + A^{\mathrm{T}}\Sigma_{t|z}^{-1}A)^{-1}(A^{\mathrm{T}}\Sigma_{t|z}^{-1}\boldsymbol{t} + \Sigma_z^{-1}\boldsymbol{\mu}_z). \end{aligned}$$

Remarks 12.6.

- Let

  $$Q_z = \Sigma_z^{-1},\ \boldsymbol{p}_z = Q_z\boldsymbol{\mu}_z,\ Q_{t|z} = \Sigma_{t|z}^{-1},\ \boldsymbol{p}_{t|z} = A^{\mathrm{T}}Q_{t|z}\boldsymbol{t}.$$

  Then observe that the posterior $p(\boldsymbol{z}|\boldsymbol{t})$ is normal with

  $$Q_{z|t} = Q_z + A^{\mathrm{T}}Q_{t|z}A, \tag{12.150}$$

  and

  $$\boldsymbol{p}_{z|t} = \boldsymbol{p}_z + \boldsymbol{p}_{t|z}. \tag{12.151}$$

  Observe in (12.150) and (12.151) that if we express the involved normal pdfs in terms of their precision matrices and the respective $\boldsymbol{p}$ vectors, as in (12.111), then these parameters are added to

give the respective parameters of the posterior. This property complies with what we have said in Section 12.4 concerning the conjugate priors of the exponential family.

REFERENCES

[1] D. Arthur, S. Vassilvitskii, *k*-means++: the advantages of careful seeding, in: Proceedings 18th ACM-SIAM Symposium on Discrete algorithms, SODA, 2007, pp. 1027-1035.

[2] L.E. Baum, T. Petrie, G. Soules, N. Weiss, A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains, Ann. Math. Stat. 41 (1970) 164-171.

[3] M.J. Beal, Variational algorithms for approximate Bayesian inference, Ph.D. Thesis, University College London, 2003.

[4] L. Bo, C. Sminchisescu, A. Kanaujia, D. Metaxas, Fast algorithms for large scale conditional 3D prediction, in: Proceedings International Conference to Computer Vision and Pattern Recognition, CVPR, Anchorage, AK, 2008.

[5] J.S. Bridle, Probabilistic interpretation of feedforward classification network outputs with relationship to statistical pattern recognition, in: F. Fougelman-Soulie, J. Heurault (Eds.), Nuero-Computing: Algorithms, Architectures and Applications, Springer Verlag, 1990.

[6] O. Cappe, E. Mouline, Online EM algorithm for latent data models, J. R. Stat. Soc. B 71(3) (2009) 593-613.

[7] G. Casella, R.L. Berger, Statistical Inference, second ed., Duxbury, 2002.

[8] G. Celeux, J. Diebolt, The SEM algorithm: A probabilistic teacher derived from the EM algorithm for the mixture problem, Comput. Stat. Quart. 2 (1985) 73-82.

[9] S.P. Chatzis, D.I. Kosmopoulos, T.A. Varvarigou, Signal modeling and classification using a robust latent space model based on *t*-distributions, IEEE Trans. Signal Process. 56(3) (2008) 949-963.

[10] S.P. Chatzis, D. Kosmopoulos, T.A. Varvarigou, Robust sequential data modeling using an outlier tolerant hidden Markov model, IEEE Trans. Pattern Anal. Machine Intell. 31(9) (2009) 1657-1669.

[11] B. Delyon, M. Lavielle, E. Moulines, Convergence of a stochastic approximation version of the EM algorithm, Ann. Stat. 27(1) (1999) 94-128.

[12] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. R. Stat. Soc. B 39(1) (1977) 1-38.

[13] R.P. Feyman, Statistical Mechanics: A Set of Lectures, Addison-Wesley, Reading, MA, 1998.

[14] S.F. Gull, Bayesian inductive inference and maximum entropy, in: G.J. Erickson, C.R. Smith (Eds.), Maximum Entropy and Bayesian Methods in Science and Engineering, Kluwer, 1988.

[15] M.R. Gupta, Y. Chen, Theory and Use of the EM Algorithm, Found. Trends Signal Process. 4(3) (2010) 223-299.

[16] L.K. Hansen, C.E. Rasmussen, Pruning from adaptive regularization, Neural Comput. 6 (1993) 1223-1232.

[17] L.K. Hansen, J. Larsen, Unsupervised learning and generalization, in: IEEE International Conference on Neural Networks, 1996, pp. 25-30.

[18] L.K. Hansen, Bayesian averaging is well-temperated, in: S. Solla (Ed.), Proceedings Neural Information Processing, NIPS, MIT Press, 2000, pp. 265-271.

[19] D. Haussler, M. Kearns, R. Schapire, Bounds on the sample complexity of Bayesian learning using information theory and the VC dimension, Machine Learn. 14 (1994) 83-113.

[20] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice Hall, 1999.

[21] D.R. Hunter, K. Lange, A tutorial on MM algorithms, Amer. Statist. 58 (2004) 30-37.

[22] E.T. Jaynes, On the rationale of the maximum entropy methods, Proc. IEEE 70(9) (1982) 939-952.

[23] E.T. Jaynes, Bayesian Methods-an introductory tutorial, in: J.H. Justice (Ed.), Maximum Entropy and Bayesian Methods in Science and Engineering, Cambridge University Press, 1986.

[24] H. Jeffreys, Theory of Probability, Oxford University Press, 1992.
[25] M.I. Jordan, R.A. Jacobs, Hierarchical mixture of experts and the EM algorithm, Neural Comput. 6 (1994) 181-214.
[26] P. Liang, D. Klein, Online EM for unsupervised models, in: Proceeding of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, NAACL, 2009, pp. 611-619.
[27] T.J. Loredo, From Laplace to supernova SN 1987A: Bayesian inference in astrophysics, in: P. Fougere (Ed.), Maximum entropy and Bayesian methods, Kluwer, 1990, pp. 81-143.
[28] D.J.C. McKay, Bayesian interpolation, Neural Comput. 4(3) (1992) 417-447.
[29] D.J.C. MacKay, The evidence framework applied to classification networks, Neural Comput. 4 (1992) 720-736.
[30] D.J.C. McKay, Probable networks and plausible predictions – a review of practical Bayesian methods for supervised neural networks. Network Comput. Neural Syst. 6 (1995) 469-505.
[31] X.L. Meng, D. Van Dyk, The EM algorithman old folk-song sung to a fast new tune, J. R. Stat. Soc. B 59(3) (1997) 511-567.
[32] G.J. McLachlan, K.E. Basford, Mixture Models. Inference and Applications to Clustering, Marcel Dekker, 1988.
[33] X.L. Meng, D.B. Rubin, Maximum likelihood estimation via the ECM algorithm: a generalization framework, Biometrika 80 (1993) 267-278.
[34] J.E. Moody, Note on generalization, regularization, and architecture selection in nonlinear learning systems, in: Proceedings, IEEE Workshop on Neural Networks for Signal Processing, Princeton, NJ, USA, 1991, pp. 1-10.
[35] T. Moon, The expectation maximization algorithm, Signal Process. Mag. 13(6) (1996) 47-60.
[36] R.M. Neal, G.E. Hinton, A new view of the EM algorithm that justifies incremental, sparse and other variants, in: M.J. Jordan (Ed.), Learning in Graphical Models, Kluwer Academic Publishers, 1998, pp. 355-369.
[37] S. Shoham, Robust clustering by deterministic agglomeration EM of mixtures of multivariate *t* distributions, Pattern Recognit. 35(5) (2002) 1127-1142.
[38] M. Stephens, Dealing with label-switching in mixture models, J. R. Statist. Soc. B 62 (2000) 795-809.
[39] M. Svensen, C.M. Bishop, Robust Bayesian mixture modeling, Neurocomputing 64 (2005) 235-252.
[40] R. Sundberg, Maximum likelihood theory for incomplete data from an exponential family, Scand. J. Statist. 1(2) (1974) 49-58.
[41] G. Schwarz, Estimating the dimension of a model, Ann. Stat. 6 (1978) 461-464.
[42] J. Shao, Mathematical Statistics: Exercises and Solutions, Springer, 2005.
[43] S. Theodoridis, K. Koutroumbas, Pattern Recognition, fourth ed., Academic Press, 2009.
[44] Y. Tikochinsky, N.Z. Tishby, R.D. Levin, Alternative approach to maximum-entropy inference, Phys. Rev. A 30(5) (1985) 2638-2644.
[45] D.M. Titterington, A.F.M. Smith, U.E. Makov, Statistical Analysis of Finite Mixture Distributions, John Wiley & Sons, 1985.
[46] A. Vehtari, J. Lampinen, Bayesian model assessment and comparison using cross-validation predictive densities, Neural Comput. 14 (10) (2002) 2439-2468.
[47] C.S. Wallace, P.R. Freeman, Estimation and inference by compact coding, J. R. Stat. Soc. B 493 (1987) 240-265.
[48] R.L. Wolpert, Exponential families, Technical Report, University of Duke, www.stat.duke.edu/courses/Spring11/sta114/lec/expofam.pdf, 2011.
[49] C. Wu, On the convergence properties of the EM algorithm, Ann. Stat. 11(1) (1983) 95-103.