Expectation Maximization*

Notes for the bAG seminar

Miguel de Benito

Universität Augsburg

Oct. 14th, 2014

Abstract. We revisit some of the ideas from Philipp's talk from the point of view of latent variables, then explain how an iterative algorithm, Expectation-Maximization, appears naturally for the estimation of the parameters. We apply it to mixtures of Gaussian and Bernoulli variables. We also say a few words about the Kullback-Leibler divergence to be able to show why EM works.

WARNING! These notes are sloppy, incomplete, inconsistent and sloppy. Also, they contain errors a.s. Did I mention they are sloppy? They are also a rip-off of [Bis06, Chapters 9 and 1.6].

Table of contents

*. This document has been written using the GNU TeXmacs text editor (see `www.texmacs.org`).

1. Why we do this

As before, when we studied K-means, we will model a random variable representing each data point as a mixture of known densities, though not necessarily Gaussians.

1. We would very much like to generalize/improve the algorithm we saw last week, as well as understand and prove its convergence. Furthermore, not using gradient descent frees us from the issue with the parameter for step length.

2. The technique we will develop can be used for *maximum a posteriori* estimation in order to fix the problem of overfitting without recourse to ad-hoc fixes or heuristics.

3. In a fully Bayesian framework we can use EM for automatic *model selection* (e.g. in our examples with mixtures, determining the right number K of different distributions).

4. An online version of EM runs in $O(1)$ time. See [Bis06, §9.4].

5. Understanding EM sets the grounds for a larger class of techniques for deterministic parameter estimation, known as variational inference.

6. Because we can.

2. Notation and conventions

Random variables take values in $\mathbb{R}$ or $\mathbb{R}^d$ and are denoted with capital letters X, Z. Their **realizations** are denoted with x, z. We will always use X for the RV originating the data. Z will be reserved for the **latent variables**, e.g. those "artificially" added to model our assumptions about the internals (e.g. hidden state) of the system studied. In our particular examples Z will take values in $\{e_1, \ldots, e_k\}$ (see §3). The **probability measure** is denoted by a capital P, **densities of random variables** by a small p. If X is a discrete RV, $P(X = x) = p_X(x)$. If no confusion can arise we will use the arguments of densities to distinguish among them: $p(x) = p_X(x)$ will be the density of X evaluated at x, $p(z|x) = p_{Z|X}(z|x) = p_{Z|X=x}(z)$ the density of Z "given $X = x$" at z, etc. (see §8.1 for a few words on conditioning on an event with probability zero). $\mathcal{N}(\mu, \Sigma)$ denotes a **Gaussian distribution** of mean μ and covariance matrix Σ, but if we add an argument x, then $\mathcal{N}(x|\mu, \Sigma)$ denotes its density, i.e.

$$\mathcal{N}(x|\mu, \Sigma) := \frac{1}{(2\pi)^{D/2}} \frac{1}{\sqrt{\det \Sigma}} \exp\left\{-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)\right\}.$$

We subsume all **distribution parameters** into the letter θ, which may refer to different parameters even in the same line, but is always understood as "the parameters for the distribution where θ is used". Conditioning wrt. θ is to be understood (for now) as a notational reminder that there are some parameters. We observe N **data points** in $\mathbb{R}^d$. They are i.i.d. observations of X, i.e. we have the **joint random variable** $\boldsymbol{X} = (X_1, ..., X_N)$ with **realization** $\boldsymbol{x} = (x_1, ..., x_N)$. $\boldsymbol{x}$ is the actual data we observe. Finally, C is a constant (hopefully) independent of the relevant quantities. It may change from line to line.

3. Gaussian Mixtures with latent variables

In a first example to connect with the previous talk about K-means, we postulate a convex combination of normal distributions (a **Gaussian mixture**) as the model for each of our data points:

$$\sum_{k=1}^{K} \pi_k \mathcal{N}(\mu_k, \Sigma_k), \tag{1}$$

for some given $K \in \mathbb{N}$ and $\sum_{k=1}^{K} \pi_k = 1$, $\pi_k \in [0, 1]$. Our final objective is of course to fit the parameters $\theta = (\pi_1, ..., \pi_k, \mu_1, ..., \mu_k, \Sigma_1, ..., \Sigma_k)$ to the data $\boldsymbol{x} = (x_1, ..., x_N)$, which is a realization of the joint $\boldsymbol{X} = (X_1, ..., X_N)$, but we will now arrive at this model from a new point of view.

We introduce the **latent variable** Z with values in $\{e_1, ..., e_K\}$ with $e_k \in \mathbb{R}^K$ basis vectors. For each data point, i.e. for each realization x_n of X, we will have an unobserved z_n. The event $Z = e_k$ should be interpreted as: the corresponding realization of X "comes from" the distribution $\mathcal{N}(\mu_k, \Sigma_k)$, i.e. we want to set

$$p(x | Z = e_k) = \mathcal{N}(x | \mu_k, \Sigma_k).$$

and we choose $p(Z = e_k) = \pi_k$, for some $\pi_k \in [0, 1]$ such that $\sum_{k=1}^{K} \pi_k = 1$, to be determined.

Remark: What we are doing here is determining the joint distribution $p_{X,Z}$ from the conditional $p_{X|Z}$ and the marginal p_Z. Although in this case this is no more than the product rule, in more complicated situations it is essential to study how joint densities factorize as products of conditional probabilities to reduce the complexity of the problem. We can depict the relationship between X, Z as in Figure 1.

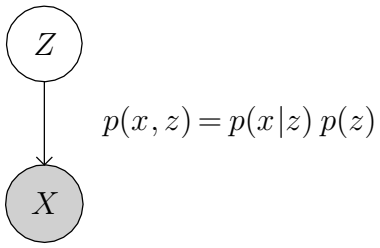

Figure 1. Graphical representation for our latent variable model.

Notice that using the fact that each realization $z = (z_1, \ldots, z_K)$ of Z is one of the e_k, we may write the distribution of Z as

$$p(z) = \prod_{k=1}^{K} \pi_k^{z_k},$$

where of course the π_k are unknown, and the conditional $X|Z = z$ as

$$p(x|z) = \prod_{k=1}^{K} \mathcal{N}(x|\mu_k, \Sigma_k)^{z_k}.$$

The density of X may now be written as the marginalization of the joint $p_{X,Z}$:

$$\begin{aligned} p(x) &= \sum_{z \in \{e^1, \ldots, e^K\}} p(z)\, p(x|z) \\ &= \sum_{k=1}^{K} \prod_{j=1}^{K} \pi_j^{\delta_{jk}} \prod_{l=1}^{K} \mathcal{N}(x|\mu_l, \Sigma_l)^{\delta_{lk}} \\ &= \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k). \end{aligned}$$

So we see that we recover the density (1) as we wished. From here we could proceed by attempting to maximize the likelihood of the data given the parameters, that is using the marginal:

$$\log p(\boldsymbol{x}|\theta) = \sum_{n=1}^{N} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k), \tag{2}$$

(recall that the X_n are i.i.d.), differentiating wrt. the parameters π_k, μ_k, Σ_k, equating to zero and solving for the parameters, but we will meet with the following two problems, already discussed in Philipp's talk, and explained in more detail below.

1. We don't obtain closed-form estimators when we differentiate (2), so we need to devise and justfiy an iterative scheme (e.g. Newton).

2. We suffer overfitting with collapsing variances (actually "ultrafitting"!).

So, why the did we introduce Z? We now have a joint probability distribution $p_{X,Z}$ to work with which may (as is the case with Gaussian distributions) be more easily tractable and lead to closed form estimators when maximizing its log likelihood (see §5). However, the values $\boldsymbol{z}$ are part of the problem, so we will end up computing an expectation wrt. the posterior $p_{Z|X}$. We will show that this provides an iterative scheme (which coincides with the Newton scheme for K-means) guaranteed to increase the log likelihood.

For comparison purposes here is how Expectation-Maximization looks for a mixture of Gaussians. You'll notice the exact analogy with K-means, where the *assignment step* is now the *E-step* and the *update step* is now the *M-step*. We will derive this algorithm in §5.

Algorithm No-questions-asked-EM for Gaussian mixtures

1. Initialize the parameters $\theta = (\mu_k, \Sigma_k, \pi_k)_{k=1}^K$.
2. **E-Step:** compute the posterior $p_{\boldsymbol{Z}|\boldsymbol{X}=\boldsymbol{x},\theta^{\text{old}}}(\boldsymbol{z})$ or *responsibilities*.
3. **M-Step:** maximize $E_{\boldsymbol{Z}|\boldsymbol{x},\theta}[\log p(\boldsymbol{x}, \boldsymbol{Z}|\theta)]$ wrt. the parameters μ_k, Σ_k, π_k to obtain new values $\mu_k^{\text{new}}, \Sigma_k^{\text{new}}, \pi_k^{\text{new}}$.
4. Evaluate log likelihood, check for convergence, go to 2 if necessary.

4. The EM algorithm

The **key idea** is to use the following decomposition, true for any strictly positive probability density q:

$$\log p(\boldsymbol{x}|\theta) = \mathcal{L}(q,\theta) + \mathrm{KL}(q\|p_{\boldsymbol{Z}|\boldsymbol{x},\theta}), \tag{3}$$

where KL is the Kullback-Leibler divergence (see §8.3), $p_{\boldsymbol{Z}|\boldsymbol{x},\theta} = p_{\boldsymbol{Z}|\boldsymbol{X}=\boldsymbol{x},\theta}$ is the distribution of the posterior of $\boldsymbol{Z} = (Z_1, \dots, Z_N)$ given the dataset $\boldsymbol{x} \in \mathbb{R}^{N\times d}$ and the choice of parameters θ and

$$\begin{aligned} \mathcal{L}(q,\theta) &:= \sum_{\boldsymbol{z}'\in\{e_1,\dots,e_K\}^N} q(\boldsymbol{z}') \log \frac{p(\boldsymbol{x},\boldsymbol{z}'|\theta)}{q(\boldsymbol{z}')} \\ \mathrm{KL}(q\|p_{\boldsymbol{Z}|\boldsymbol{x}}) &:= -\sum_{\boldsymbol{z}'\in\{e_1,\dots,e_K\}^N} q(\boldsymbol{z}') \log \frac{p(\boldsymbol{z}'|\boldsymbol{x},\theta)}{q(\boldsymbol{z}')}. \end{aligned}$$

Indeed, adding both quantities and cancelling terms we have

$$\mathcal{L}(q,\theta) + \mathrm{KL}(q\|p_{\boldsymbol{Z}|\boldsymbol{X}}) = \sum_{\boldsymbol{z}'} q(\boldsymbol{z}') \log \frac{p(\boldsymbol{x},\boldsymbol{z}'|\theta)}{p(\boldsymbol{z}'|\boldsymbol{x},\theta)} = \underbrace{\sum_{\boldsymbol{z}'} q(\boldsymbol{z}')}_{=1} \log p(\boldsymbol{x}|\theta) = \log p(\boldsymbol{x}|\theta).$$

Notice that because $\mathrm{KL} \geqslant 0$, the functional $\mathcal{L}(q,\theta)$ is *always a lower bound* to $\log p(\boldsymbol{x}|\theta)$.

In each step of the algorithm we will optimize this bound: first wrt. q then wrt. θ.

The lower bound can be written as

$$\begin{aligned}\mathcal{L}(q,\theta) &= \sum_{\boldsymbol{z}'} q(\boldsymbol{z}')\log p(\boldsymbol{x},\boldsymbol{z}'|\theta) - \sum_{\boldsymbol{z}'} q(\boldsymbol{z}')\log q(\boldsymbol{z}') \\ &= E_q[\log p(\boldsymbol{x},\boldsymbol{z}'|\theta)] - \underbrace{E_q[\log q(\boldsymbol{z}'|\theta)]}_{=C}, \end{aligned} \tag{4}$$

where the constant C is the entropy of q (see §8.3) and $\log p(\boldsymbol{x}, \boldsymbol{z}'|\theta) = g(\boldsymbol{z}')$ a function of $\boldsymbol{z}'$ alone. This last line provides the intuition behind the decomposition: *if* we had the values $\boldsymbol{z}$ of the latent variable $\boldsymbol{Z} = (Z_1, \ldots, Z_N)$, we could evaluate the joint density on the complete data set $p(\boldsymbol{x}, \boldsymbol{z}|\theta)$. Under certain assumptions it would be an easy matter to maximize its log likelihood instead of the marginal $\log p(\boldsymbol{x}|\theta)$ and we wouldn't need the previous decomposition (e.g. assuming independence and a Gaussian mixture, the joint is a double product which the logarithm transforms into a double sum, see (6)). But we don't have the values $\boldsymbol{z}$ and all we actually know about them is contained in the posterior $p_{\boldsymbol{Z}|\boldsymbol{x},\theta}$. It is therefore sensible to compute the expected value of $\log p(\boldsymbol{x}, \boldsymbol{z}|\theta)$ under the marginal $p_{\boldsymbol{Z}|\boldsymbol{x},\theta}$, which is exactly the first term in the last equation.

Algorithm EM (Expectation-Maximization)

1. Initialize.
2. **E-step:** Fix $\theta = \theta^t$ in (3). Maximize the lower bound $\mathcal{L}$ wrt. the distribution q. This is achieved by minimizing KL because

   $$\mathcal{L}(q,\theta) = \log p(\boldsymbol{x}|\theta) - \mathrm{KL}(q\|p_{\boldsymbol{Z}|\boldsymbol{x}}),$$

   and $\log p(\boldsymbol{x}|\theta)$ is independent of q. This maximum is realized for the choice $q^t = p_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}$ because $\mathrm{KL}(q\|p_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}) \geqslant 0$ with equality iff $q = p_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}$, see (12). For this q^t, $\mathcal{L}(q^t, \theta^t) = \log p(\boldsymbol{x}|\theta^t)$ by construction, it is a lower bound and the gradients of both functions are parallel, so we have a situation like in Figure 2
   Finally, to prepare for the next step compute explicitly $\mathcal{L}$ as given by (4) and $g(\boldsymbol{z}) := \log p(\boldsymbol{x}, \boldsymbol{z}|\theta)$:

   $$\mathcal{L}(p_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}, \theta) = E_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}[g(\boldsymbol{Z})] + C. \tag{5}$$

3. **M-step:** Fix now $q^t = p_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}$ coming from the E-step and maximize $\mathcal{L}(q^t, \theta)$ in (5) wrt. θ to obtain θ^{t+1}. This new choice of parameters yields a new lower bound $\mathcal{L}(q^t, \theta^{t+1}) \geqslant \mathcal{L}(q^t, \theta^t)$ and a yet greater increase in the target function:

   $$\begin{aligned}\log p(\boldsymbol{x}|\theta^{t+1}) &= \mathcal{L}(q^t,\theta^{t+1}) + \underbrace{\mathrm{KL}(q^t\|p_{\boldsymbol{Z}|\boldsymbol{x},\theta^{t+1}})}_{>0} \\ &> \mathcal{L}(q^t,\theta^t) + \underbrace{\mathrm{KL}(q^t\|p_{\boldsymbol{Z}|\boldsymbol{x},\theta^t})}_{=0} \\ &= \log p(\boldsymbol{x}|\theta^t),\end{aligned}$$

   where the first KL divergence is strictly positive because $q^t \neq p_{\boldsymbol{Z}|\boldsymbol{x},\theta^{t+1}}$. This means that this step always increases the log likelihood unless we were already at a maximum and $\theta^t = \theta^{t+1}$.
4. Check whether we have finished.

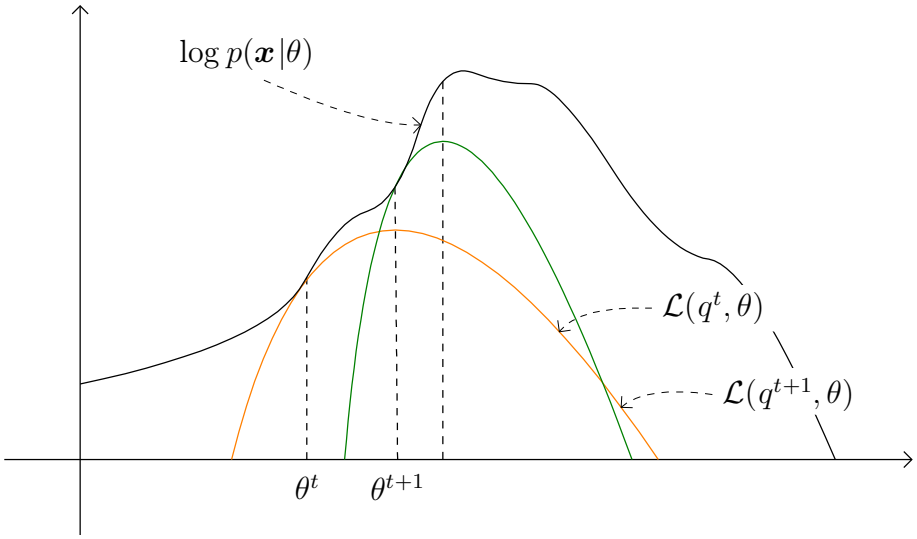

Figure 2. Example for a mixture in the exponential family, where the lower bound is always a concave function. Computing θ^{t+1} in the M-step increases the log likelihood. The E-step then computes the posterior q^{t+1} over the latent variables for the new parameters θ^{t+1}, yielding a new lower bound functional $\mathcal{L}(q^{t+1}, \theta)$, tangent to the log likelihood at θ^{t+1}.

5. Back to Gaussian Mixtures

Recall from §3 the expressions for p_Z and $p_{X|z}$. The joint density of (X, Z) is

$$p_{X,Z}(x, z) = \prod_{k=1}^{K} [\pi_k \mathcal{N}(x|\mu_k, \Sigma_k)]^{z_k}.$$

Suppose we had all the values $\boldsymbol{z} = (z_1, \ldots, z_N)$ in addition to the data $\boldsymbol{x} = (x_1, \ldots, x_N)$ (one talks of the **complete dataset**) and we wanted to maximize the log likelihood:

$$\begin{aligned} \log p(\boldsymbol{x}, \boldsymbol{z}|\theta) &= \log \prod_{n=1}^{N} \prod_{k=1}^{K} [\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)]^{z_{nk}} \\ &= \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} [\log \pi_k + \log \mathcal{N}(x_n|\mu_k, \Sigma_k)] \end{aligned} \tag{6}$$

where we now had to assume $\pi_k \in (0, 1)$ to be able to take the logarithm. The problem is we don't have the z_n! But EM saves the day:

E-step: Fix $\theta = \theta^t$. We know that the optimal q^t is the posterior over the latent variables, and we immediately see that it factorizes:

$$p_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}(\boldsymbol{z}) = \frac{p(\boldsymbol{x}|\boldsymbol{z}, \theta^t)\, p(\boldsymbol{z}|\theta^t)}{p(\boldsymbol{x}|\theta^t)} = \prod_{n=1}^{N} \underbrace{\frac{1}{p(x|\theta^t)} \prod_{k=1}^{K} [\pi_k^t \mathcal{N}(x|\mu_k^t, \Sigma_k^t)]^{z_k}}_{=: p_{Z|x}(z)}.$$

With this choice for q^t the new lower bound is the function of θ:

$$\mathcal{L}(p_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}, \theta) = E_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}[g(\boldsymbol{Z})] + C,$$

where $g(\boldsymbol{\zeta}) := \log p_{\boldsymbol{X},\boldsymbol{Z}|\theta}(\boldsymbol{x},\boldsymbol{\zeta}|\theta)$. Setting $z_{nk} = \mathbb{1}_{\{e_k\}}(Z_n)$, the expectation is

$$\begin{aligned} E_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}[g(\boldsymbol{Z})] &= E_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}\left[\sum_{n=1}^{N}\sum_{k=1}^{K} z_{nk}\left[\log\pi_k + \log\mathcal{N}(x_n|\mu_k,\Sigma_k)\right]\right] \\ &= \sum_{n=1}^{N}\sum_{k=1}^{K} \underbrace{E_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}[\mathbb{1}_{\{e_k\}}(Z_n)]}_{(*)}\left[\log\pi_k + \log\mathcal{N}(x_n|\mu_k,\Sigma_k)\right]. \end{aligned}$$

In order to compute $(*)$ we use that the posterior factorizes over n:[1]

$$\begin{aligned} E_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}[\mathbb{1}_{\{e_k\}}(Z_n)] &= \sum_{\boldsymbol{z}\in\{e_1,\dots,e_K\}^N} \mathbb{1}_{\{e_k\}}(z_n)\, p_{\boldsymbol{Z}|\boldsymbol{x},\theta^t}(\boldsymbol{z}) \\ &= \sum_{\boldsymbol{z}\in\{e_1,\dots,e_K\}^N} \mathbb{1}_{\{e_k\}}(z_n)\left(\prod_{j=1}^{N} p_{Z|x,\theta^t}(z_j)\right) \\ &= \underbrace{\sum_{z\in\{e_1,\dots,e_K\}} \mathbb{1}_{\{e_k\}}(z)\, p_{Z|x,\theta^t}(z)}_{=E_{Z|x,\theta^t}[\mathbb{1}_{\{e_k\}}(Z_n)]} \underbrace{\left(\prod_{j=1,j\neq n}^{N}\sum_{z\in\{e_1,\dots,e_K\}} p_{Z|x,\theta^t}(z)\right)}_{=1} \\ &= p_{Z|x,\theta^t}(e_k) \\ &= \frac{1}{p(x|\theta^t)}\,\pi_k^t\mathcal{N}(x|\mu_k^t,\Sigma_k^t) \\ &= \frac{\pi_k^t\mathcal{N}(x|\mu_k^t,\Sigma_k^t)}{\sum_{j=1}^{K}\pi_j^t\mathcal{N}(x|\mu_j^t,\Sigma_j^t)} =: \gamma_{nk}^t. \end{aligned}$$

Notice that this quantity is precisely the **responsibility** of cluster k for the point x_n as defined in (10) when maximizing the log likelihood of the marginal $p(\boldsymbol{x}|\theta)$. Substituting above, we have:

$$E_{\boldsymbol{Z}|\boldsymbol{x}}[\log p(\boldsymbol{x},\boldsymbol{z}|\theta^t)] = \sum_{n=1}^{N}\sum_{k=1}^{K}\gamma_{nk}^t\left[\log\pi_k^t + \log\mathcal{N}(x_n|\mu_k^t,\Sigma_k^t)\right], \tag{7}$$

and this is the quantity we maximize next.

M-step: We maximize (7) wrt. each of μ_k, Σ_k, π_k obtaining *closed formulas* for μ_k^{t+1} and we are in business for the next step.

1. Differentiate (7) wrt. μ_k and equate to zero to obtain: $0 = \sum_{n=1}^{N}\gamma_{nk}^t\,(\Sigma_k^t)^{-1}\,(x_n - \mu_k)$. Now solve for μ_k and multiply on the left by Σ_k^t:

$$\mu_k^{t+1} = \frac{1}{N_k^t}\sum_{n=1}^{N}\gamma_{nk}^t\,x_n, \text{ where } N_k^t = \sum_{n=1}^{N}\gamma_{nk}^t. \tag{8}$$

1. This is just an application of "Fubini's" theorem, and might be easier to see if we write integrals to make notation easier.

2. Differentiate (7) wrt. Σ_k and equate to zero. After some computations and solving for Σ_k (here the constraint that Σ_k be symmetric positive definite complicates matters considerably, see e.g. [AO85]) we find:

$$\Sigma_k^{t+1} = \frac{1}{N_k^t} \sum_{n=1}^{N} \gamma_{nk}^t (x_n - \mu_k^t) \otimes (x_n - \mu_k^t).$$

3. Differentiate (7) wrt. π_k, with the constraint $\sum_{k=1}^K \pi_k = 1$ using a Lagrange multiplier (recall that now $\pi_k \in (0,1)$):

$$0 = \partial_{\pi_k} E_{\boldsymbol{Z}|\boldsymbol{x}}[\log p(\boldsymbol{x}, \boldsymbol{z}|\theta^t)] + \partial_{\pi_k} \lambda \left(\sum_k \pi_k - 1 \right) = \sum_{n=1}^{N} \frac{\gamma_{nk}^t}{\pi_k} + \lambda = \frac{N_k^t}{\pi_k} + \lambda.$$

Multiply by π_k and sum over K using the constraint on the π_k and the fact that $\sum_{k=1}^K \gamma_{nk}^t = 1$:

$$0 = \sum_{k=1}^{K} N_k^t + \lambda \sum_{k=1}^{K} \pi_k \Rightarrow \lambda = -N$$

Substituting above this yields

$$\pi_k^{t+1} = \frac{N_k^t}{N}.$$

Evaluate: We use the estimators obtained in the M-step to evaluate the log likelihood and check if we should stop.

6. An example with mixtures of Bernoullis

This model is called *latent class analysis* and is an exercise for the reader! We will use it for Hidden Markov Models over discrete variables, if we ever do those.

We now have D i.i.d. binary variables $B = (B_1, ..., B_D)$, with $B_i \sim \mathcal{B}(\mu_i)$, i.e.

$$p(b|\mu) = \prod_{i=1}^{D} \mu_i^{b_i} (1 - \mu_i)^{1 - b_i}.$$

Consider a mixture

$$X \sim \sum_{k=1}^{K} \pi_k B(\mu^k).$$

Compute mean, covariance and log likelihood for some data $\boldsymbol{x} = (x_1, ..., x_N)$.

Apply this to the MNIST data set of handwritten digits, see the Kaggle at [Kag12] or the original source at [LCB12].

7. Where to go from here

Some ideas:

- Implement **maximum a posteriori** estimation using EM. The idea is to maximize the log likelihood of the posterior over the parameters $\log p(\theta|\boldsymbol{x})$ to solve the problem of "collapsing variances". Using Bayes' rule and the decomposition (3) we have

$$\begin{aligned}\log p(\theta|\boldsymbol{x}) &= \log p(\theta,\boldsymbol{x}) - \log p(\boldsymbol{x}) \\ &= \log p(\boldsymbol{x}|\theta) + \log p(\theta) - \underbrace{\log p(\boldsymbol{x})}_{=C} \\ &= \mathcal{L}(q,\theta) + \mathrm{KL}(q\|p_{\boldsymbol{Z}|\boldsymbol{x},\theta}) + \log p(\theta) - C \\ &= \tilde{\mathcal{L}}(q,\theta) + \mathrm{KL}(q\|p_{\boldsymbol{Z}|\boldsymbol{x},\theta}),\end{aligned}$$

  and we now apply EM to this expression. The only necessary modification is to the estimators in the M step, due to the appearance of the prior in $\log p(\theta)$.

- Fully Bayesian use of EM including model selection.

- Expectation Propagation.

8. Appendix

8.1. A word about conditional densities.

If (X, Z) is a RV with real values, then

$$p(z|x) = p_{Z|X=x}(z) := \frac{p_{X,Z}(x,z)}{p_X(x)}$$

defines a probability density at every x such that $p(x) \neq 0$, since it is non-negative, measurable and

$$\int p(z|x)\,\mathrm{d}z = \frac{1}{p(x)}\int p(x,z)\,\mathrm{d}z = 1.$$

For discrete variables this agrees with the usual product rule, but for continuous variables needs interpreting, since $\{X = x\}$ is a null set, see e.g. [JP04, §12].

8.2. Problems related to the log likelihood approach.

Collapsing variance: Suppose that at some point in our algorithm for maximizing (2), whatever it is, we set $\mu_k = x_n$ for some n, k, then the k-th summand in (2) will be

$$\pi_k \mathcal{N}(x_n|\mu_k,\Sigma_k) = C\,\frac{1}{\sqrt{\det\Sigma_k}},$$

and we can obviously send this quantity to $+\infty$ by adjusting the covariance matrix. If the algorithm goes on trying to maximize the whole expression, it will do exactly that and will almost surely fail with a division by zero.

We talk about collapsing variance because Σ being symmetric positive definite, it may be diagonalized and the new diagonal entries are the variances of the X_i in the new coordinates.

We have at least two solutions to this problem, the hacky and the not-so-hacky:

1. Hack hack: use some heuristics in the code to avoid the situation $\mu_k = x_n$ for any k, n. For instance, randomly reset μ_k if there is any $n \in \{1, ..., N\}$ such that $|x_n - \mu_k| < \varepsilon$ for some fixed $\varepsilon > 0$. Ugly!

2. Penalize: postulate a prior on the parameters p_Θ and add a term $\log p_\Theta(\theta)$ (called **capacity**) to penalize lower probabilities of θ. Some good choice of p_Θ should fix the issue. We can justify this idea within the framework presented in §4, as sketched in §7.

Non-closed estimators: Maximizing (2) is a matter of differentiating wrt. each parameter in θ, setting equal to 0 and solving for the parameter, then checking the second derivatives. Consider for example the mean μ_k:

$$\partial_{\mu_k} \log p(\boldsymbol{x}|\theta) = \text{...compute compute...} = \sum_{n=1}^{N} \gamma_{nk} \Sigma_k^{-1} (x_n - \mu_k).$$

Solving for μ_k yields:

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} x_n, \text{ where } N_k = \sum_{n=1}^{N} \gamma_{nk}, \tag{9}$$

and we defined the **responsibility**

$$\gamma_{nk} = P(Z_n = e_k | X_n = x_n) \tag{10}$$

to be the *posterior probability* of having z_n given the data x_n. You can see that (9) is not a closed formula for μ_k since on the rhs. γ_{nk} depends itself on μ_k. Indeed, recalling our previous computations for the joint and marginal and using the product rule twice we have explicitly

$$p(z|x) = \frac{p(x,z)}{p(x)} = \frac{p(x|z)\, p(z)}{p(x)} = \frac{\prod_{k=1}^{K} \mathcal{N}(x|\mu_k, \Sigma_k)^{z_k} \prod_{k=1}^{K} \pi_k^{z_k}}{\sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k)},$$

and consequently:

$$\gamma_{nk} = p(z_n = e_k | x_n) = \frac{\mathcal{N}(x_n|\mu_k, \Sigma_k)\, \pi_k}{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}.$$

Analogously, the other two estimators are also in non-closed form (they are obtained in the same way as in §5):

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk}(x_n - \mu_k) \otimes (x_n - \mu_k) \text{ and } \pi_k = \frac{N_k}{N}. \tag{11}$$

We already saw a solution with Philipp to this problem: the expressions (9), (11) for the estimators for μ_k, Σ_k, π_k feature the variables themselves at the rhs., which suggests that we use a Newton-like iterative scheme. Our different modelling guides us to Expectation-Maximization.

Model order: This is actually not a problem of the log likelihood estimation *per se*, but of the mixture model. We have always assumed that we know some value for the number of components K, but for many applications this will not be the case.

8.3. A few sketchy ideas from information theory.

We want to know how much information about a random variable X is transmitted when we send a specific value x. This should be a monotonically decreasing function of the probability of the event because of the following simple ideas:

- News of event with probability one are no information.
- News of event with low probability are "high" information.

Information content: $h(x) := -\log p(x)$.

Entropy of a RV:

$$E_X[h(X)] = -\int_{\mathbb{R}} \log\left(p(x)\right) p(x)\, \mathrm{d}x,$$

where we set $p \log p = 0$ if $p = 0$ by continuity $p \ln p \to 0$ for $p \to 0$. This integral converges (!).

Noiseless decoding theorem. (Shanon 1948) *Entropy is a lower bound on the number of bits (resp. nats) needed to transmit the state of a random variable.*

Kullback-Leibler divergence: We have approximated an unknown distribution p of some random variable X (assumed to be $p > 0$) with a distribution q. How well did we do? Define the KL-*divergence* as

$$\begin{aligned} \mathrm{KL}(p\|q) &:= -\int \log q(x)\, p(x)\, \mathrm{d}x - \underbrace{\left(-\int \log p(x)\, p(x)\, \mathrm{d}x\right)}_{=H[X]} \\ &= -\int \log \frac{q(x)}{p(x)}\, p(x)\, \mathrm{d}x. \end{aligned}$$

Intuition: KL is the *additional* amount of information needing to be transmitted to specify X by using q instead of p. $\mathrm{KL}(\cdot\|\cdot)$ is not a distance since it is obviously not symmetric. We have however:

Non-negativity of KL.

$$\mathrm{KL}(p\|q)\geqslant 0 \text{ with equality iff } p=q. \tag{12}$$

Proof. This is a direct application of Jensen's inequality (recall that for φ convex and μ finite this is: $\varphi(\int f\,\mathrm{d}\mu)\leqslant\int\varphi(f)\,\mathrm{d}\mu$, when things make sense, blah blah...). By the convexity of the $-\log$ (recall that we assume $p>0$):

$$\mathrm{KL}(p\|q)=\int -\log\frac{q(x)}{p(x)}\,p(x)\,\mathrm{d}x\geqslant -\log\int\frac{q(x)}{p(x)}\,p(x)\,\mathrm{d}x=-\log\int q(x)\,\mathrm{d}x=0,$$

where we used that q is a probability distribution. □

An example using KL: Suppose we stick to one family of distributions $q=q(x|\theta)$. We would like to estimate the best θ by minimizing KL:

$$\mathrm{KL}(p\|q)=-\int\log\frac{q(x)}{p(x)}\,p(x)\,\mathrm{d}x=-E_p[\log q]+C,$$

but this is not possible since p is unknown. However, the expectation wrt. p may be approximated (!) by a finite sum over the observed data $x_1,\dots,x_N$:

$$\begin{aligned}\mathrm{KL}(p\|q) &\backsimeq \frac{1}{N}\sum_{n=1}^{N}\left[-\log q(x_n|\theta)+\log p(x_n)\right]\\ &= \frac{1}{N}\sum_{n=1}^{N}-\log q(x_n|\theta)+C,\end{aligned}$$

and this quantity may be now minimized. Notice that the first term is the negative log likelihood of $q(\cdot|\theta)$ under the observed data $\boldsymbol{x}=(x_1,\dots,x_n)$.

8.4. Maximization of the log likelihood for the complete dataset.

Assuming we have the complete dataset, i.e. all of $\boldsymbol{x}$ and $\boldsymbol{z}$ (which we don't) finding maxima wrt. to the parameters is easy. For the means we compute

$$\partial_{\mu_k}\log p(\boldsymbol{x},\boldsymbol{z}|\theta)=\sum_{n\in C_k}\partial_{\mu_k}\log\mathcal{N}(x_n|\mu_k,\Sigma_k)=\sum_{n\in C_k}\Sigma_k^{-1}\,(x_n-\mu_k).$$

Equating to zero and solving for μ_k we find that

$$\overline{\mu}_k=\frac{1}{\#C_k}\sum_{n\in C_k}x_n$$

is a critical point, and it is a maximum by the properties of Σ_k. Analogously, differentiation wrt. Σ_k yields (with quite some effort since we have the constraint that Σ_k be symmetric positive definite, see e.g. [AO85])

$$\overline{\Sigma}_k = \frac{1}{\#C_k} \sum_{n \in C_k} (x_n - \overline{\mu}_k) \otimes (x_n - \overline{\mu}_k).$$

Finally, for the weights π_k we have to include the constraint $\sum_k \pi_k = 1$, which we do with a Lagrange multiplier (recall that now $\pi_k \in (0,1)$, so this is no issue):

$$0 = \partial_{\pi_k} \log p(\boldsymbol{x}, \boldsymbol{z} | \theta) + \partial_{\pi_k} \lambda \left(\sum_k \pi_k - 1 \right) = \sum_{n \in C_k} \frac{1}{\pi_k} + \lambda = \frac{\#C_k}{\pi_k} + \lambda.$$

We multiply by π_k and sum over K using the constraint on π_k: $\lambda\, \pi_k = -\#C_k \Rightarrow \lambda \sum_{k=1}^K \pi_k = -\sum_{k=1}^K \#C_k = -N$, so we have $\lambda = -N$. Substituting above this yields

$$\pi_k = \frac{\#C_k}{N}.$$

9. References

You can read a much better version of all this stuff in [Bis06, Chapter 9]. Another reference, explaining less but with more applications is [Mur12, Chapter 11]. For more on probabilistic graphical models, read [Mur12, Chapter 10] or [KF09, Chapter 3]. A $\TeX_{\text{MACS}}$ file with an implementation of EM for Gaussians is also available.

[AO85] T. W. Anderson and I. Olkin. Maximum-likelihood estimation of the parameters of a multivariate normal distribution. *Linear Algebra and its Applications*, 70:147–171, oct 1985.

[Bis06] Christopher M. Bishop. *Pattern recognition and machine learning*. Information science and statistics. Springer, 1 edition, aug 2006.

[JP04] Jean Jacod and Philip Protter. *Probability essentials*. Universitext. Springer Berlin Heidelberg, 2 edition, jan 2004.

[Kag12] Kaggle.com. Classify handwritten digits using the famous MNIST data. jul 2012.

[KF09] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. Adaptive Computation and Machine Learning. MIT Press, 2009.

[LCB12] Yan LeCun, Corinna Cortes and Christopher J.C. Burges. The MNIST database of handwritten digits. 2012.

[Mur12] Kevin P. Murphy. *Machine learning: a probabilistic perspective*. MIT Press, aug 2012.