Motivation of Expectation Maximization algorithm

Ask Question

19

15

In the EM algorithm approach we use Jensen's inequality to arrive at

$$\log p(x|\theta) \geq \int \log p(z, x|\theta) p(z|x, \theta^{(k)}) dz - \int \log p(z|x, \theta) p(z|x, \theta^{(k)}) dz$$

logp(x|θ)≥∫logp(z,x|θ)p(z|x,θ(k))dz−∫logp(z|x,θ)p(z|x,θ(k))dz

and define $\theta^{(k+1)}$ θ(k+1) by

$$\theta^{(k+1)} = \arg\max_{\theta} \int \log p(z, x|\theta) p(z|x, \theta^{(k)}) dz$$

θ(k+1)=argmaxθ∫logp(z,x|θ)p(z|x,θ(k))dz

Everything I read EM just plops it down but I've always felt uneasy by not having an explanation of why the EM algorithm arises naturally. I understand that loglog likelihood is typically dealt with to deal with addition instead of multiplication but the appearance of loglog in the definition of $\theta^{(k+1)}$ θ(k+1) feels unmotivated to me. Why should one consider loglog and not other monotonic functions? For various reasons I suspect that the "meaning" or "motivation" behind expectation maximization has some kind of explanation in terms of information theory and sufficient statistics. If there were such an explanation that would be much more satisifying than just an abstract algorithm.

mixture expectation-maximization

share cite
improve this question

edited Jul 20 '13 at 20:02

jpmuc
9,531 19 46

asked Jul 18 '13 at 11:05

user782220
412 4 12

migrated from math.stackexchange.com Jul 20 '13 at 15:49

This question came from our site for people studying math at any level and professionals in related fields.

3 What is the expectation maximization algorithm?, *Nature Biotechnology* **26**:897–899 (2008) has a nice picture that illustrates how the algorithm works. – chl ♦ Jul 20 '13 at 20:58

@chl: I have seen that article. The point I'm asking is that notice that nowhere does it explain why a non-log approach can't work – user782220 Jul 21 '13 at 1:11

add a comment

5 Answers

active oldest votes

10

The EM algorithm has different interpretations and can arise in different forms in different applications.

It all starts with the likelihood function $p(x|\theta)$ p(x|θ), or equivalently, the log-likelihood function $\log p(x|\theta)$ logp(x|θ) we would like to maximize. (We generally use logarithm as it simplifies the calculation: It is strictly monotone, concave, and $\log(ab) = \log a + \log b$ log(ab)=loga+logb.) In an ideal world, the value of pp depends only on the **model parameter** θθ, so we can search through the space of θθ and find one that maximizes pp.

However, in many interesting real-world applications things are more complicated, because not all the variables are observed. Yes, we might directly observe xx, but some other variables zz are unobserved. Because of the **missing variables** zz, we are in a kind of chicken-and-eggs situation: Without zz we cannot estimate the parameter θθ and without θθ we cannot infer what the value of zz may be.

It is where the EM algorithm comes into play. We start with an initial guess of the model parameters θθ and derive the expected values of the missing variables zz (i.e., the E step). When we have the values of zz, we can maximize the likelihood w.r.t. the parameters θθ (i.e., the M step, corresponding to the $\arg\max$ argmax equation in the problem statement). With this θθ we can derive the new expected values of zz (another E step), so on and so forth. In another word, in each step we assume one of the both, $z$zand θθ, is known. We repeat this iterative process until the likelihood cannot be increased anymore.

This is the EM algorithm in a nutshell. It is well-known that the likelihood will never decrease during this iterative EM process. But keep in mind that EM algorithm doesn't guarantee global optimum. That is, it might end up with a local optimum of the likelihood function.

The appearance of $\log$log in the equation of $\theta^{(k+1)}$ θ(k+1) is inevitable, because here the function you would like to maximize is written as a log-likelihood.

share cite improve this answer edited Jan 28 '14 at 12:13

answered Jul 20 '13 at 20:17

Weiwei
570 2 10

I don't see how this answers the question. – broncoAbiertoApr 14 '17 at 18:21

add a comment

9

Likelihood vs. log-likelihood

As has already been said, the $\log$log is introduced in maximum likelihood simply because it is generally easier

to optimize sums than products. The reason we don't consider other monotonic functions is that the logarithm is the **unique** function with the property of turning products into sums.

Another way to motivate the logarithm is the following: Instead of maximizing the probability of the data under our model, we could equivalently try to minimize the Kullback-Leibler divergence between the data distribution, $p_{data}\,(x)$ pdata(x), and the model distribution, $p(x \mid \theta)$ p(x|θ),

$$D_{KL}[p_{data}\,(x) \,||\, p(x \mid \theta)] = \int p_{data}\,(x) \log \frac{p_{data}\,(x)}{p(x \mid \theta)}\, dx = const - \int p_{data}\,(x) \log p(x \mid \theta)\, dx.$$

DKL[pdata(x)||p(x|θ)]=∫pdata(x)logpdata(x)p(x|θ)dx=const−∫pdata(x)logp(x|θ)dx.

The first term on the right-hand side is constant in the parameters. If we have N N samples from the data distribution (our data points), we can approximate the second term with the average log-likelihood of the data,

$$\int p_{data}\,(x) \log p(x \mid \theta)\, dx \approx \frac{1}{N} \sum_n \log p(x_n \mid \theta).$$

∫pdata(x)logp(x|θ)dx≈1N∑nlogp(xn|θ).

An alternative view of EM

I am not sure this is going to be the kind of explanation you are looking for, but I found the following view of expectation maximization much more enlightening than its motivation via Jensen's inequality (you can find a detailed description in Neal & Hinton (1998) or in Chris Bishop's PRML book, Chapter 9.3).

It is not difficult to show that

$$\log p(x \mid \theta) = \int q(z \mid x) \log \frac{p(x, z \mid \theta)}{q(z \mid x)}\, dz + D_{KL}[q(z \mid x) \,||\, p(z \mid x, \theta)]$$

logp(x|θ)=∫q(z|x)logp(x,z|θ)q(z|x)dz+DKL[q(z|x)||p(z|x,θ)]

for any $q(z \mid x)$ q(z|x). If we call the first term on the right-hand side $F(q, \theta)$ F(q,θ), this implies that

$$F(q, \theta) = \int q(z \mid x) \log \frac{p(x, z \mid \theta)}{q(z \mid x)}\, dz = \log p(x \mid \theta) - D_{KL}[q(z \mid x) \,||\, p(z \mid x, \theta)].$$

F(q,θ)=∫q(z|x)logp(x,z|θ)q(z|x)dz=logp(x|θ)−DKL[q(z|x)||p(z|x,θ)].

Because the KL divergence is always positive, $F(q, \theta)$ F(q,θ) is a lower bound on the log-likelihood for every fixed qq. Now, EM can be viewed as alternately maximizing F F with respect to qq and θθ. In particular, by setting $q(z \mid x) = p(z \mid x, \theta)$ q(z|x)=p(z|x,θ) in the E-step, we minimize the KL divergence on the right-hand side and thus maximize F F.

share cite improve this answer

answered Jul 20 '13 at 22:56

Lucas
4,155 16 30

Thanks for the post! Though the given document doesn't say

logarithm is the unique function turning products into sums. It says logarithm is the only function that fulfills all three listed properties *at the same time*. – Weiwei Jul 22 '13 at 3:47

@Weiwei: Right, but the first condition mainly requires that the function is invertible. Of course, f(x) = 0 also implies f(x + y) = f(x)f(y), but this is an uninteresting case. The third condition asks that the derivative at 1 is 1, which is only true for the logarithm to base ee. Drop this constraint and you get logarithms to different bases, but still logarithms. – Lucas Jul 22 '13 at 7:49

add a comment

4

The paper that I found clarifying with respect to expectation-maximization is Bayesian K-Means as a "Maximization-Expectation" Algorithm (pdf) by Welling and Kurihara.

Suppose we have a probabilistic model $p(x, z, \theta)$ p(x,z,θ) with $x$xobservations, zz hidden random variables, and a total of θθparameters. We are given a dataset DD and are forced (by higher powers) to establish $p(z, \theta|D)$ p(z,θ|D).

1. Gibbs sampling

We can approximate $p(z, \theta|D)$ p(z,θ|D) by sampling. Gibbs sampling gives $p(z, \theta|D)$ p(z,θ|D) by alternating:

$$\theta \sim p(\theta|z, D)$$
$$z \sim p(z|\theta, D)$$

θ~p(θ|z,D)z~p(z|θ,D)

2. Variational Bayes

Instead, we can try to establish a distribution $q(\theta)$ q(θ) and $q(z)$q(z) and minimize the difference with the distribution we are after $p(\theta, z|D)$ p(θ,z|D). The difference between distributions has a convenient fancy name, the KL-divergence. To minimize $KL[q(\theta)q(z)||p(\theta, z|D)]$ KL[q(θ)q(z)||p(θ,z|D)] we update:

$$q(\theta) \propto \exp(E[\log p(\theta, z, D)]_{q(z)})$$
$$q(z) \propto \exp(E[\log p(\theta, z, D)]_{q(\theta)})$$

q(θ)∝exp(E[logp(θ,z,D)]q(z))q(z)∝exp(E[logp(θ,z,D)]q(θ))

3. Expectation-Maximization

To come up with full-fledged probability distributions for both zz and θθ might be considered extreme. Why don't we instead consider a point estimate for one of these and keep the other nice and nuanced. In EM the parameter θθis established as the one being unworthy of a full distribution, and set to its MAP (Maximum A Posteriori) value, θ^* θ∗.

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \; E[\log p(\theta, z, D)]_{q(z)}$$
$$q(z) = p(z|\theta^*, D)$$

$$\theta* = \text{argmax}_\theta E[\log p(\theta, z, D)] \quad q(z)q(z) = p(z|\theta*, D)$$

Here $\theta^* \in \text{argmax}$ θ∗∈argmax would actually be a better notation: the argmax operator can return multiple values. But let's not nitpick. Compared to variational Bayes you see that correcting for the $\log$log by $\exp$ exp doesn't change the result, so that is not necessary anymore.

4. Maximization-Expectation

There is no reason to treat zz as a spoiled child. We can just as well use point estimates z^* z∗ for our hidden variables and give the parameters θθ the luxury of a full distribution.

$$z^* = \text{argmax}_z \; E[\log p(\theta, z, D)]_{q(\theta)}$$

$$q(\theta) = p(\theta|z^*, D)$$

$$z* = \text{argmax}_z E[\log p(\theta, z, D)] \quad q(\theta)q(\theta) = p(\theta|z*, D)$$

If our hidden variables zz are indicator variables, we suddenly have a computationally cheap method to perform inference on the number of clusters. This is in other words: model selection (or automatic relevance detection or imagine another fancy name).

5. Iterated conditional modes

Of course, the poster child of approximate inference is to use point estimates for both the parameters θθ as well as the observations zz.

$$\theta^* = \text{argmax}_\theta \; p(\theta, z^*, D)$$

$$z^* = \text{argmax}_z \; p(\theta^*, z, D)$$

$$\theta* = \text{argmax}_\theta p(\theta, z*, D) \quad z* = \text{argmax}_z p(\theta*, z, D)$$

To see how Maximization-Expectation plays out I highly recommend the article. In my opinion, the strength of this article is however not the application to a kk-means alternative, but this lucid and concise exposition of approximation.

share cite improve this answer

answered Dec 23 '14 at 19:06

Anne van Rossum
308 3 13

(+1) this is a beautiful summary of all methods. – kedarps Apr 20 '17 at 17:54

add a comment

4

There is a useful optimisation technique underlying the EM algorithm. However, it's usually expressed in the language of probability theory so it's hard to see that at the core is a method that has nothing to do with probability and expectation.

Consider the problem of maximising

$$g(x) = \sum_i \exp(f_i(x))$$
g(x)=∑iexp(fi(x))

(or equivalently $\log g(x)$ logg(x)) with respect to xx. If you write down an expression for $g'(x)$ g′(x) and set it equal to zero you will often end up with a transcendental equation to solve. These can be nasty.

Now suppose that the f_i fi play well together in the sense that linear combinations of them give you something easy to optimise. For example, if all of the $f_i(x)$ fi(x) are quadratic in xx then a linear combination of the $f_i(x)$ fi(x) will also be quadratic, and hence easy to optimise.

Given this supposition, it'd be cool if, in order to optimise $\log g(x) = \log \sum_i \exp(f_i(x))$ logg(x)=log∑iexp(fi(x)) we could somehow shuffle the $\log$log past the $\sum$ ∑ so it could meet the $\exp$ exps and eliminate them. Then the f_i fi could play together. But we can't do that.

Let's do the next best thing. We'll make another function $h$hthat is similar to gg. And we'll make it out of linear combinations of the f_i fi.

Let's say x_0x0 is a guess for an optimal value. We'd like to improve this. Let's find another function hh that matches $g$gand its derivative at x_0x0, i.e. $g(x_0) = h(x_0)$ g(x0)=h(x0) and $g'(x_0) = h'(x_0)$ g′(x0)=h′(x0). If you plot a graph of hh in a small neighbourhood of x_0x0 it's going to look similar to gg.

You can show that

$$g'(x) = \sum_i f_i'(x) \exp(f_i(x)).$$
g′(x)=∑ifi′(x)exp(fi(x)).

We want something that matches this at x_0x0. There's a natural choice:

$$h(x) = \text{constant} + \sum_i f_i(x) \exp(f_i(x_0)).$$
h(x)=constant+∑ifi(x)exp(fi(x0)).

You can see they match at $x = x_0$ x=x0. We get

$$h'(x) = \sum_i f_i'(x) \exp(f_i(x_0)).$$
h′(x)=∑ifi′(x)exp(fi(x0)).

As x_0x0 is a constant we have a simple linear combination of the f_i fi whose derivative matches gg. We just have to choose the constant in hh to make $g(x_0) = h(x_0)$ g(x0)=h(x0).

So starting with x_0x0, we form $h(x)$ h(x) and optimise that. Because it's similar to $g(x)$ g(x) in the neighbourhood of x_0x0we hope the optimum of hh is similar to the

optimum of g. Once you have a new estimate, construct the next hh and repeat.

I hope this has motivated the choice of hh. This is exactly the procedure that takes place in EM.

But there's one more important point. Using Jensen's inequality you can show that $h(x) \leq g(x)$ h(x)≤g(x). This means that when you optimise $h(x)$ h(x) you always get an xx that makes gg bigger compared to $g(x_0)$ g(x0). So even though $h$hwas motivated by its *local* similarity to gg, it's safe to *globally* maximise hh at each iteration. The hope I mentioned above isn't required.

This also gives a clue to when to use EM: when linear combinations of the arguments to the $\exp$ exp function are easier to optimise. For example when they're quadratic - as happens when working with mixtures of Gaussians. This is particularly relevant to statistics where many of the standard distributions are from exponential families.

share cite improve this answer answered Oct 20 '16 at 1:51

Dan Piponi
142 6

add a comment

3

As you said, I will not go into technical details. There are quite a few very nice tutorials. One of my favourites are Andrew Ng's lecture notes. Take a look also at the references here.

1. EM is naturally motivated in mixture models and models with hidden factors in general. Take for example the case of Gaussian mixture models (GMM). Here we model the density of the observations as a weighted sum of K K gaussians:

$$p(x) = \sum_{i=1}^{K} \pi_i N(x|\mu_i, \Sigma_i)$$

p(x)=∑i=1Kπ*i*N(x|μi,Σi)

where π_i π*i* is the probability that the sample xx was caused/generated by the ith component, μ_i μi is the mean of the distribution, and Σ_i Σi is the covariance matrix. The way to understand this expression is the following: each data sample has been generated/caused by one component, but we do not know which one. The approach is then to express the uncertainty in terms of probability (π_i π*i* represents the chances that the ith component can account for that sample), and take the weighted sum. As a concrete example, imagine you want to cluster text documents. The idea is to assume that each document belong to a topic (science, sports,...) which you do not know beforehand!. The possible topics are hidden variables. Then you are given a bunch of documents, and by counting n-grams or whatever features you extract, you want to then find those clusters and see to which cluster each document

belongs to. EM is a procedure which attacks this problem step-wise: the expectation step attempts to improve the assignments of the samples it has achieved so far. The maximization step you improve the parameters of the mixture, in other words, the form of the clusters.

2. The point is not using monotonic functions but convex functions. And the reason is the Jensen's inequality which ensures that the estimates of the EM algorithm will improve at every step.