

# Expectation Maximization algorithm and it's applications for image processing

Seleznyov Mikhail

CMC MSU

January 27, 2021

# Contents

- 1 Introduction
- 2 What do we need EM for?
  - Latent Variable
  - Example
- 3 Algorithm
  - “Simple” EMA
  - “Classic” EMA
- 4 Bayesian Application
- 5 Variants of EM
- 6 References
- 7 Thanks

- What do we need EM for?
- How does it work?
- How can we apply it?

What is the Expectation-Maximization algorithm?

Expectation-Maximization algorithm is an iterative method, used to find maximum likelihood (or maximum a posteriori) estimates of parameters in statistical models, which depend on latent variables.

What is a latent variable?

A latent variable (from Latin *lateo* (lie hidden), opposed to observable variables) is a variable which can not be directly observed, but it's value can be inferred from observable variables.

Examples: intelligence, life quality, happyness.

Why do we need latent variables?

- They help to deal with missing values
- They are sometimes useful when we don't have a lot of data

Let's look at a real example.

## Example

Imagine that you are an analytic in a small company. Your director wants to hire some new workers. He gives you some information about candidates, and also about current staff. You have to make a prediction of how good a candidate will perform on the on-site interview.

	High school grade	University grade	IQ score	Phone Interview	Onsite interview
<i>John</i>	4.0	4.0	120	3/4	?
<i>Helen</i>	3.7	3.6	N/A	4/4	?
<i>Jack</i>	3.2	N/A	112	2/4	?
<i>Emma</i>	2.9	3.2	N/A	3/4	?
	High school grade	University grade	IQ score	Phone Interview	Onsite interview
<i>Sophia</i>	3.5	3.6	N/A	4/4	85/100
...					

Why can't we apply standard machine learning methods?

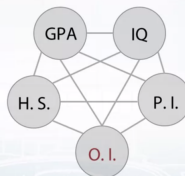
- There are missing values.
- We don't have much data (small company, not Google or Amazon).
- Director would also like your algorithm not to give a point estimate, but an interval, or at least a degree of confidence.



Let's try to build a statistical model.

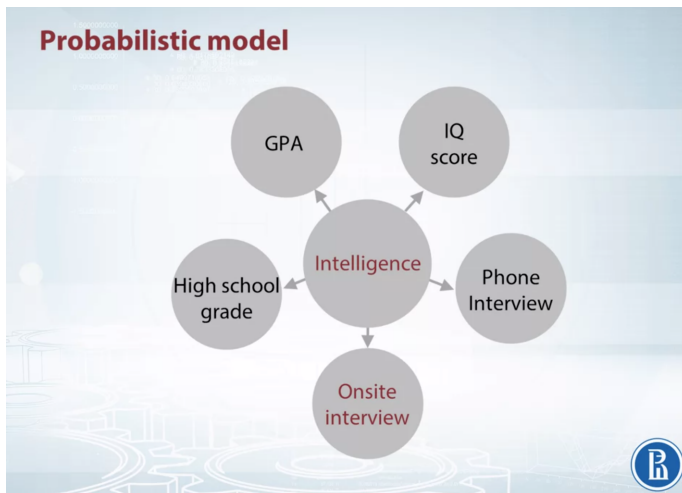
### Probabilistic model

High school	GPA	IQ	Phone Interview	Onsite Interview	Probability
1.0	1.0	1	0/4	1/100	0.001
1.0	1.0	1	0/4	2/100	0.0023
...	...	...	...		
4.0	4.0	180	4/4	100	0.000001

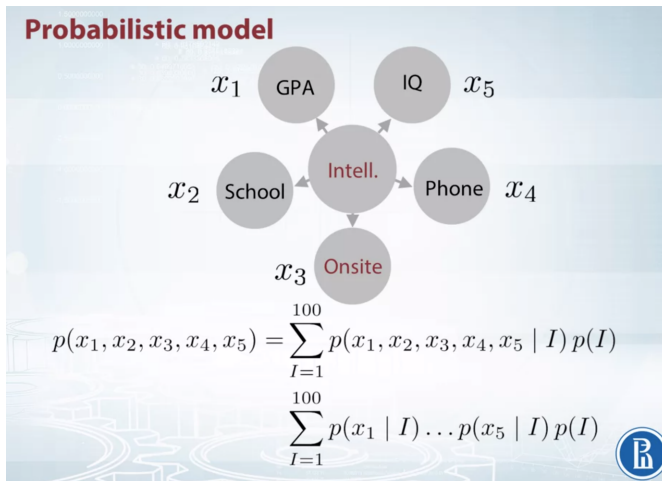


Obviously, we can't suppose, that the characteristics are independent. Moreover, it seems like everything is connected to everything here. We end up with millions of possible combinations.

Let's then introduce a latent variable — intelligence, and suppose that everything else depends on this variable.



We can now describe conditional distributions of observable variables depending on the intelligence.



Time to discuss the algorithm itself.

We work in a statistical model with latent variables. That is, we have a set of observed variables  $\mathbb{X} = \{X_1, \dots, X_n\}$ , a set of latent variables  $\mathbb{Z} = \{Z_1, \dots, Z_m\}$  and unknown parameters  $\theta \in \Theta$ .

Let's use  $\mathcal{X}$  and  $\mathcal{Z}$  to denote the sample space of observable and latent variables, respectively.

We will be looking for a maximum likelihood estimate:

$$L(X, \theta) = p(X | \theta) = \int_{\mathcal{Z}} p(X, Z | \theta) dZ \rightarrow \max_{\theta \in \Theta}$$

# Algorithm

**E-step:** The E-step of EM algorithm computes the expected value of log-likelihood  $l(X, Z, \theta)$  given the observed data  $X$  and the current parameter estimate  $\theta_{\text{old}}$ . Namely, we define

$$\begin{aligned} Q(\theta, \theta_{\text{old}}) &= \mathbb{E}[l(X, Z, \theta) \mid X, \theta_{\text{old}}] \\ &= \int_{\mathcal{Z}} l(X, Z \mid \theta) p(Z \mid X, \theta) dZ \end{aligned}$$

**M-step:** The M-step consists of maximizing the computed expectation  $Q(\theta, \theta_{\text{old}})$  over  $\theta$ . That is, we set

$$\theta_{\text{new}} = \operatorname{argmax}_{\theta \in \Theta} Q(\theta, \theta_{\text{old}})$$

# Why does it work?

Maximizing likelihood  $L(X, \theta)$  is equivalent to maximizing log-likelihood  $l(X, \theta) = \ln L(X, \theta)$ , since the logarithm function is concave. Usually it is easier to work with the second. We will write log-likelihood in different form, using Jensen's inequality. It states, that for any concave function  $f(x)$  the following holds:

$$\mathbb{E}f(\xi) \leq f(\mathbb{E}\xi)$$

# Keep calm and do maths

$$\begin{aligned} \ln p(X | \theta) &= \ln \int_{\mathcal{Z}} p(X, Z | \theta) dZ = \ln \int_{\mathcal{Z}} \frac{p(X, Z | \theta)}{p(Z | X, \theta_{\text{old}})} p(Z | X, \theta_{\text{old}}) dZ = \\ &\ln \mathbb{E} \left[ \frac{p(X, Z | \theta)}{p(Z | X, \theta_{\text{old}})} \middle| X, \theta_{\text{old}} \right] \geq \mathbb{E} \left[ \ln \frac{p(X, Z | \theta)}{p(Z | X, \theta_{\text{old}})} \middle| X, \theta_{\text{old}} \right] = \\ &\underbrace{\mathbb{E} \left[ \ln p(X, Z | \theta) \middle| X, \theta_{\text{old}} \right]}_{Q(\theta, \theta_{\text{old}})} - \underbrace{\mathbb{E} \left[ \ln p(Z | X, \theta_{\text{old}}) \middle| X, \theta_{\text{old}} \right]}_{\text{const}} \end{aligned}$$

where  $\theta_{\text{old}}$  is some known fixed value, for example, from previous iteration.

Thus,

$$l(X, \theta) \geq \underbrace{\mathbb{E} \left[ \ln p(X, Z | \theta) \middle| X, \theta_{\text{old}} \right]}_{Q(\theta, \theta_{\text{old}})} + \text{const}$$

One can notice, that if  $\theta = \theta_{\text{old}}$ , the inequality becomes an equality, since the term inside the expectation, to which we apply Jensen's inequality, becomes constant. So, if we denote right part as  $g(\theta | \theta_{\text{old}})$ , we'll have

$$l(X, \theta) \geq g(\theta | \theta_{\text{old}}), \quad l(X, \theta_{\text{old}}) = g(\theta_{\text{old}} | \theta_{\text{old}})$$

Therefore, any value of  $\theta$  that increases  $g(\theta | \theta_{\text{old}})$  beyond  $g(\theta_{\text{old}} | \theta_{\text{old}})$  must also increase  $l(X, \theta)$  beyond  $l(X, \theta_{\text{old}})$ .



◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡



$$l(X, \theta) = \mathcal{L}(q, \theta) + \text{KL}(q \parallel p_{\mathcal{Z}|\mathcal{X}})$$

The second term is called *Kullback-Leibler divergence*, or *relative entropy*. It is defined as follows:

$$\text{KL}(q \parallel p) = - \int q(x) \ln \frac{p(x)}{q(x)} dx = -\mathbb{E}_q \ln \frac{p(x)}{q(x)}$$

KL divergence is always nonnegative and equals to zero if and only if  $q(x) = p(x)$  almost everywhere. So, it could be used as a measure of distance between distributions.

Although, it is not a proper distance, since it is not symmetric and does not satisfy the triangle inequality.

$$l(X, \theta) = \mathcal{L}(q, \theta) + \text{KL}(q \parallel p_{Z|\mathcal{X}})$$

The first term is called *evidence lower bound*, or *ELBO*.

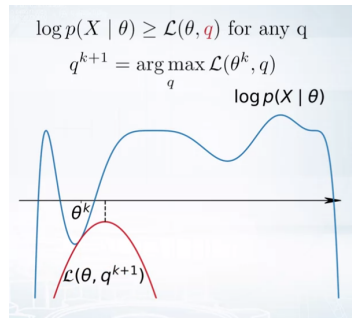
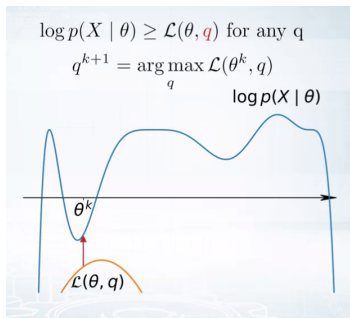
$$\mathcal{L}(q, \theta) = \int q(Z) \ln \frac{p(X, Z|\theta)}{q(Z)} dZ = \mathbb{E}_q \ln \frac{p(X, Z|\theta)}{q(Z)}$$

Since KL divergence is nonnegative,  $\mathcal{L}(q, \theta)$  is always less or equal than  $l(X, \theta)$ . Thus, maximizing ELBO, we implicitly maximize likelihood.

# Algorithm

- E-step:** This step maximizes  $\mathcal{L}(q, \theta_{\text{old}})$  with respect to distribution  $q(\cdot)$ , keeping  $\theta = \theta_{\text{old}}$ . Turns out, that it is equivalent to minimizing KL divergence. If we set  $q(Z) = p(Z | X, \theta)$ , then  $\text{KL}(q || p_{Z|X}) = 0$ , and we have  $\mathcal{L}(q, \theta_{\text{old}}) = l(X, \theta_{\text{old}})$ .
- M-step:** Here we keep  $q(Z)$  fixed and optimize  $\mathcal{L}(q, \theta)$  with respect to  $\theta$ . Since the lower bound increases, likelihood must also increase.

Choose such distribution  $q(\cdot)$ , which maximizes the ELBO  $\mathcal{L}(\theta^k, q)$ .  
If  $q = p_{Z|X}$ , then  $\mathcal{L}(\theta^k, q) = l(X, \theta^k) = \log p(X | \theta^k)$ .



1

The EM algorithm can also be used to compute the mode of the posterior distribution,  $p(\theta | X)$ , in a Bayesian setting where we are given a prior  $p(\theta)$  on an unknown parameter  $\theta$ .

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

$$\ln p(\theta | X) = \ln p(X | \theta) + \ln p(\theta) - \ln p(X)$$

The last term is just a constant, so it can be omitted during the optimization. If we now substitute

$l(X, \theta) = \mathcal{L}(q, \theta) + \text{KL}(q || p_{\mathcal{Z}|X})$ , we obtain

$$\ln p(\theta | X) = \mathcal{L}(q, \theta) + \text{KL}(q || p_{\mathcal{Z}|X}) + \ln p(\theta) - \ln p(X)$$

The E-step will remain the same, but in M-step we'll have to remember about the prior.



In Bayesian statistics all parameters are treated like random variables. So, we can go further and look not for a point estimate, but distribution over all possible values of  $\theta$ . In such case the difference between E-step and M-step disappears, since both times we choose a distribution (over parameters or latent variables) such that the value of lower bound is maximized. From that perspective, EMA looks much like a coordinate descent, because at each step we keep one group of parameters fixed and optimize with respect to the other.

If  $\mathcal{L}(p_Z, p_\theta)$  is a lower bound for  $l(X, \theta)$  for any distribution over latent variables  $p_Z$  and any distribution over parameters  $p_\theta$ , we can write algorithm as follows:

**E-step** Choose  $p_Z$  to maximize  $\mathcal{L}$ :

$$p_Z^{k+1} = \operatorname{argmax} \mathcal{L}(p_Z^k, p_\theta^k)$$

**M-step** Choose  $p_\theta$  to maximize  $\mathcal{L}$ :

$$p_\theta^{k+1} = \operatorname{argmax} \mathcal{L}(p_Z^{k+1}, p_\theta^k)$$

There is a common problem with Bayesian inference in complex statistical models: we often end up with intractable integrals. One way to cope with them is to choose the distribution  $q$  over latent variables (or parameters) from some simple parametric family of distributions. For example, in variational Bayesian methods people choose factorized distributions. Imagine that

$Z = (Z_1, Z_2, \dots, Z_t) \in \mathbb{R}^t$ . Then we look for  $q(Z)$  such that:

$$q(Z) = q_1(Z_1)q_2(Z_2) \dots q_t(Z_t) \approx p(Z | X, \theta)$$

The approximation can be interpreted in various ways. Usually it is expressed in terms of minimizing the KL divergence. Of course, here we can't find a precise solution, but we weren't able to find it anyway, so it's okay.

# Conditional EM

Sometimes the M-step is hard. It happens when we can't perform maximization in closed form. What if we abandon the idea of maximization and be satisfied with only *increasing the value*? This version was actually proposed in the original paper, and it is called *Generalized EM*. In order to make a algorithm, we have to specify, how we are gonna increase the value of the lower bound.

In *Conditional EM* the M-step is replaced with a sequence of conditional maximizations steps, each of which maximizes the ELBO with respect to some vector function of  $\theta$ ,  $g_s(\theta)$ ,  $s = 1, \dots, S$ . For example, we can partition vector  $\theta$  into subvectors  $(\theta_1, \dots, \theta_S)$ , and make  $S$  steps, maximizing lower bound wrt  $\theta_s$ .

This idea can be further developed into ECME and SAGEM.

# Accelerated EM

EM is a fixed point algorithm. That is, it can be written as

$$\theta_{n+1} = \Phi(\theta_n), \text{ where } \Phi(\theta') = \underset{\theta \in \Theta}{\operatorname{argmax}} Q(\theta, \theta')$$

If sequence  $\{\theta_n\}$  converges towards some value  $\theta^*$  and  $\Phi$  is continuous, then  $\theta^*$  is a fix point for  $\Phi$ , i.e.  $\theta^* = \Phi(\theta^*)$ . If we further suppose, that  $\Phi$  is differentiable, we can approximate the sequence behaviour around point  $\theta^*$ , using Taylor's expansion:

$$\theta_{n+1} = \Phi(\theta_n) \approx \Phi(\theta^*) + \left. \frac{\partial \Phi}{\partial \theta} \right|_{\theta^*} (\theta_n - \theta^*) = \theta^* + \left. \frac{\partial \Phi}{\partial \theta} \right|_{\theta^*} (\theta_n - \theta^*)$$

If we denote  $S = I - \left. \frac{\partial \Phi}{\partial \theta} \right|_{\theta^*}$ , then  $\theta^* \approx \theta_n + S^{-1}(\theta_{n+1} - \theta_n)$ .

# Accelerated EM

If we knew the matrix  $S$  (it is sometimes called *the speed matrix*), we could approximate the solution in one iteration:  
 $\theta^* \approx \theta_0 - S^{-1}(\theta_1 - \theta_0)$ . There are ways to approximate the matrix  $S^{-1}$ , based on current parameter value  $\theta_n$ .

The scheme would look like this:

**E-step:** Compute  $Q(\theta, \theta_n)$  and approximate  $S_n^{-1} \approx S_{-1}$

**M-step:** Get an intermediate value  $\hat{\theta} = \underset{\theta \in \Theta}{\operatorname{argmax}} Q(\theta, \theta_n)$

**Update:** Set  $\theta_{n+1} = \theta_n + S^{-1}(\hat{\theta} - \theta_n)$

This is called *Aitken acceleration*.

# Accelerated EM

Frankly speaking, Aitken acceleration has a lot of problems. It is computationally hard to approximate the inverse of the speed matrix. We also lose the monotonic property — likelihood is not guaranteed to increase at each iteration.

There is another way of acceleration, which uses the idea of the conjugate gradients method. In vanilla gradient descend on each step we choose the gradient in current point as a direction of the next step. In conjugate gradients method we choose the direction depending on our previous steps. This involves the computation of so called generalized gradients.

# Accelerated EM

M. Jamshidian and R. I. Jennrich noticed, that  $\Phi(\theta) - \theta$  (where  $\Phi(\theta)$  is  $\arg \max_{\theta \in \Theta} \mathcal{L}(q, \theta)$ ) could serve as a good approximation of the generalized gradient. So, one could make use of it, replacing the maximization step with the update rule of form:

$$\theta_{n+1} = \theta_n + \lambda_n d_n$$

where  $d_n$  is the direction, composed from current generalized gradient approximation  $\Phi(\theta_n) - \theta_n$  and previous directions, and  $\lambda_n$  is the step size, typically computed from a line maximization of  $l(X, Z, \theta)$ .



# PX-EM

There is a version called Parameter Expanded Expectation Maximization. The new joint distribution  $q = q(X, Z | \theta, \alpha)$  is introduced, such that there is  $\alpha_0$ :  $q(X, Z | \theta, \alpha_0) = p(X, X | \theta)$ , and also there is a reduction function  $r(\theta, \alpha): \Theta \times \mathcal{A} \mapsto \Theta$ , for which

$$p(X | r(\theta, \alpha)) = \int_Z q(X, Z | \theta, \alpha) dZ$$

The M-step finds  $(\theta^*, \alpha^*)$  such that  $q(X | \theta^*, \alpha^*) \geq q(X | \theta_n, \alpha_0)$ . Then we obtain the next value  $\theta_{n+1} = r(\theta^*, \alpha^*)$ . Therefore  $p(X | \theta_{n+1}) = q(X | \theta^*, \alpha^*) \geq q(X | \theta_n, \alpha_0)$ .

In some way, PX-EM capitalizes on the fact that a large deviation between the estimate of  $\alpha$  and its known value  $\alpha_0$  is an indication that the parameter of interest  $\theta$  is poorly estimated. Hence, PX-EM adjusts the M-step for this deviation via the reduction function.

# Stochastic EM

By definition,

$$Q(\theta, \theta_{\text{old}}) = \mathbb{E} \left[ \frac{p(X, Z | \theta)}{p(Z | X, \theta_{\text{old}})} \middle| X, \theta_{\text{old}} \right] = \mathbb{E} \left[ p(X | \theta) \middle| X, \theta_{\text{old}} \right]$$

It is a conditional expectation, the best estimate of complete log-likelihood in a sense of minimizing mean squared error. Maybe, we don't need the best estimate? We had already tried substituting maximization with increasing. Let's now replace the expectation with sampling.

**Simulation:** Compute  $p(Z | X, \theta_n)$ , draw a sample  $Z^{(n)}$  from  $p(Z | X, \theta_n)$ .

**Maximization:** Set  $\theta_{n+1} = \underset{\theta \in \Theta}{\operatorname{argmax}} p(Z^{(n)} | \theta)$ .

# Stochastic EM

By construction, the resulting sequence  $\{\theta_n\}$  is an homogeneous Markov chain which, under mild regularity conditions, converges to a stationary pdf. This means in particular that  $\{\theta_n\}$  doesn't converge to a unique value! Various schemes can be used to derive a pointwise limit, such as averaging the estimates over iterations once stationarity has been reached. It was established in some specific cases that the stationary pdf concentrates around the likelihood maximizer with a variance inversely proportional to the sample size. However, in cases where several local maximizers exist, one may expect a multimodal behavior.

# SAEM

Stochastic Approximation type EM. The SAEM algorithm is a simple hybridation of EM and SEM that provides a pointwise convergence as opposed to the erratic behavior of SEM. Given a current estimate  $\theta_n$ , SAEM performs a standard EM iteration in addition to the SEM iteration. The parameter is then updated as a weighted mean of both contributions, yielding:

$$\theta_{n+1} = (1 - \gamma_{n+1})\theta_{n+1}^{\text{EM}} + \gamma_{n+1}\theta_{n+1}^{\text{SEM}}$$




where  $0 \leq \gamma_n \leq 1$ . Of course, to apply SAEM, the standard EM needs to be tractable. The sequence  $\gamma_n$  is typically chosen so as to decrease from 1 to 0, in such a way that the algorithm is equivalent to SEM in the early iterations, and then becomes more similar to EM. It is established that SAEM converges almost surely towards a local likelihood maximizer (thus avoiding saddle points) under the

assumption that  $\gamma_n$  decreases to 0 with  $\lim_{n \rightarrow +\infty} \frac{\gamma_n}{\gamma_{n+1}} = 1$  and  $\sum_{n=1}^{+\infty} \gamma_n < +\infty$ .

# MCME

Monte Carlo EM. At least formally, MCEM turns out to be a generalization of SEM. In the SEM simulation step, we draw  $m$  independent samples  $z_n^{(1)}, z_n^{(2)}, \dots, z_n^{(m)}$  and then maximize the following function:  $\hat{Q}(\theta, \theta_n) = \frac{1}{m} \sum_{j=1}^m \log p(z_n^{(j)} | \theta)$ . In general, it converges almost surely to the standard EM auxiliary function thanks to the law of large numbers. Choosing a large value for  $m$  justifies calling this Monte Carlo something. In this case,  $\hat{Q}$  may be seen as an empirical approximation of the standard EM auxiliary function, and the algorithm is expected to behave similarly to EM. On the other hand, choosing a small value for  $m$  is not forbidden, if not advised (in particular, for computational reasons). A possible strategy consists of increasing progressively the parameter  $m$ , yielding a “simulated annealing” MCEM which is close in spirit to SAEM.

# References

-  A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum Likelihood from Incomplete Data via the EM Algorithm, December 8, 1976
-  Martin Haugh, The EM Algorithm, Spring 2015
-  Alexis Roche, EM algorithm and variants: an informal tutorial, Spring 2003

Thanks for attention!