Easy Expectation Maximisation

Victor Btesh

March 5, 2021

1 Introduction and motivation

This small statistical computing project has the aim of providing a general purpose architecture to perform clustering on a set of data points and features. I am sure there exist better more complex packages to handle such problems but I need practice and having such a tool suits my current research needs. Finally, it's fun. Be advised, it will probably (for sure) be buggy, lack rigour and have unclear notation.

2 Models

This section details the type of model this package will be able to handle. The basic assumptions are that all data points are i.i.d, which is standard, but also that all features are independent given the hidden clustering variable. This allows for much simpler maximisation steps which can be standardised for each distribution. I aim to provide support for all standard distributions from the exponential family, starting with Poisson and Gaussian with known variance and unknown mean. Derivations will be below. I also want it to handle discrete time discrete space Markov chains, as it is a classic use case for this algorithm.

2.1 Joint distributions

Given two sets of variables, where h is unique, hidden and the cluster multinomial distribution with m possible outcomes and $v = \{v_1, v_2, ..., v_k\}$ are visible, and a parameter set θ, we have must have the following joint probability distribution:

$$p(h, v|\theta) = p(h|\theta) \prod_{j=1}^{k} p(v_j|h, \theta)$$

Yielding the following if we observed n i.i.d samples, where i are indices, not powers:

$$p(\mathbf{h}, \mathbf{v}|\theta) = \prod_{i=1}^{n} p(h^i|\theta) \prod_{j=1}^{k} p(v_j^i|h^i, \theta)$$

This is important as we want the log likelihood to have the following form:

$$\begin{aligned} \log p(h, v|\theta) &= \log \left[p(h|\theta) \prod_{j=1}^{k} p(v_i|h, \theta) \right] \\ &= \log p(h|\theta) + \sum_{j=1}^{k} \log p(v_j|h, \theta) \end{aligned}$$

Again, after observing n i.i.d. samples:

$$\log p(\mathbf{h}, \mathbf{v}|\theta) = \sum_{i=1}^{n} \left[\log p(h^i|\theta) + \sum_{j=1}^{k} \log p(v_j^i|h^i, \theta) \right]$$

2.2 Maximisation steps

This form of log likelihood is useful as the energy term, which we want to optimise in the maximisation step, will look as follows:

$$\begin{aligned}
\mathbb{E}(\theta) &= \sum_{i=1}^{n} \left\langle \log p(h^i|\theta) + \sum_{j=1}^{k} \log p(v_j^i|h^i, \theta) \right\rangle_{q^i(h)} \\
&= \sum_{i=1}^{n} \left[\left\langle \log p(h^i|\theta) \right\rangle_{q^i(h)} + \sum_{j=1}^{k} \left\langle \log p(v_j^i|h^i, \theta) \right\rangle_{q^i(h)} \right] \\
&= \sum_{i=1}^{n} \left\langle \log p(h^i|\theta) \right\rangle_{q^i(h)} + \sum_{i=1}^{n} \sum_{j=1}^{k} \left\langle \log p(v_j^i|h^i, \theta) \right\rangle_{q^i(h)} \\
&= \sum_{i=1}^{n} \left\langle \log p(h^i|\theta) \right\rangle_{q^i(h)} + \sum_{j=1}^{k} \sum_{i=1}^{n} \left\langle \log p(v_j^i|h^i, \theta) \right\rangle_{q^i(h)}
\end{aligned}$$

As a result, when we optimise w.r.t to each parameter in θ, for instance θ^h, the parameter of the hidden variable, all other terms can be ignored.

On $q^i(h)$, the variational distribution This is the variational distribution over hidden states h w.r.t to which we are performing the Expectation step. Two things are worthy to note here. First, we $q(h)$ actually represent $p(h|v, \theta^{old})$, which is the probability of being in a given cluster given observed variables and previous values of θ. This is because the expectation step is simply setting $q(h) = p(h|v, \theta^{old})$. The second thing is that it does not depend on the θ we are currently optimising but on θ^{old}, i.e. the θ optimised in the previous iteration of the algorithm. Rigorously, $q^i(h)$ should be written as $p^i(h|v, \theta^{old})$. Finally, the i index represents the fact that we compute the $p^i(h|v, \theta^{old})$ for each observations, therefore leading to different $q(h)$ for each sample.

On θ, the parameter matrix Note θ is matrix where each row corresponds to parameters for each variable associated with a specific cluster. If we have only one parameter per variable, as in a set of means for a Gaussians, then θ will be a c by $k + 1$ matrix where c is the number of clusters and k is the number of observed variables. If visible variables have multiple parameters, such as multinomial distributions, θ will be a c by $k + 1$ by o 3d matrix, where o is the number of parameters, i.e. outcomes, to learn for each cluster for a given observed variable. As observed variables can have different numbers of parameters, e.g. if we observe a multinomial and a Poisson random variable, as such we will treat each column in θ separately, guaranteeing that each θ^k is at a most a 2d matrix.

Optimisation should happen w.r.t to each entry θ. We are actually computing the gradient w.r.t θ for all variables. As discussed, we split θ so obtain one θ^k for each variable. We would then have for the prior probability of being in a given cluster:

$$\begin{aligned}\nabla_{\theta^h}\mathbb{E}(\theta) &= \nabla_{\theta^h}\left[\sum_{i=1}^{n}\left\langle \log p(h^i|\theta)\right\rangle_{q^i(h)}\right] \\ &= \sum_{i=1}^{n}\nabla_{\theta^h}\left\langle \log p(h^i|\theta)\right\rangle_{q^i(h)} \\ &= \sum_{i=1}^{n}\left\langle \nabla_{\theta^h}\log p(h^i|\theta)\right\rangle_{q^i(h)}\end{aligned}$$

What we get here is a generic expression that can be applied to any variable in the model. The gradient can be push inside the expectation as $q(h|v)$ does not depend on θ. As such, we can find generic expressions for gradient of each standard distribution w.r.t to a set of parameters, where each parameter in the set corresponds to one of the values of h, i.e. our clusters.

2.2.1 Prior distribution over hidden states

Log likelihood

$$\log p(h^i|v^i,\theta^h) = \log\theta^h$$

Where θ^h is the vector of probabilities for each cluster.

Optimal parameter value

$$\arg\max_{\theta^h}\mathbb{E}(\theta) = \frac{1}{n}\sum_{i=1}^{n}q^i(h)$$

Where θ^h is the vector of probabilities for each cluster and $q^i(h) = p(h^i|v^i,\theta^{old})$.

Derivation Let us continue here to derive the simplest case of the distribution over hidden states, which is the prior $p(h|\theta)$:

$$\begin{aligned}&= \sum_{i=1}^{n}\nabla_{\theta^h}\sum_{j=1}^{k}q^i(h)\log\theta_j^h \\ &= \sum_{i=1}^{n}\frac{q^i(h)}{\theta^h}\end{aligned}$$

After adding a Lagrange multiplier as $p(h|\theta)$, as a probability distribution must satisfy $\sum_{j=1}^{k}p(h_j|\theta) = 1$, which can be equivalently written as $\sum_{j=1}^{k}\theta_j^h = 1$, such that

$$\begin{aligned}f(\theta^h) &= \mathbb{E}(\theta) \\ g(\theta^h) &= \sum_{j=1}^{k}\theta_j^h - 1\end{aligned}$$

and

$$L(\theta^h,\lambda) = f(\theta^h) + \lambda g(\theta^h)$$

we can compute the gradient and maximise given the constraints:

$$\begin{aligned}\nabla L(\theta^h, \lambda) &= \nabla f(\theta^h) + \nabla \lambda g(\theta^h) \\ &= \sum_{i=1}^{n} \frac{q^i(h)}{\theta^h} + \lambda\end{aligned}$$

Optimising by setting it equal to 0, we get the following:

$$\theta^h = \frac{\sum_{i=1}^{n} q^i(h)}{\lambda} = \frac{1}{n} \sum_{i=1}^{n} q^i(h)$$

Where it can be shown that λ is a normalisation constraint, which here will be n, the number of observations. The expression is therefore the average of all probabilities in the variational distribution over the whole sample.

2.2.2 Multinomial distribution (categorical random variables)

This distribution should be used to describe categorical variables.

We call θ^m a generic $c \times o$ parameter matrix where c is the number of hidden clusters, o the number of categories/outcomes in the observed multinomial distribution and each entry θ^m_{co} is a probability observing outcome o given cluster c. In addition, let $p(v^i|h^i|\theta^m)$ refer to this generic multinomial distribution. Note that normalisation of θ^m is done row-wise.

Log likelihood

$$\log p(v^i|h^i, \theta_m) = \log \theta^m$$

Thus the log likelihood is simply the element wise log of θ^m.

Optimal parameter value

$$\arg\max_{\theta^m} \mathbb{E}(\theta) \propto \sum_{i=1}^{n} q^i(h) \mathbb{1}[v^i = v]$$

Intuitively, what this expression means is that to find the values of the parameters θ_m that maximise $\mathbb{E}(\theta)$, we must count all cases where a given visible outcome v has been observed weighted by the variational distribution.

Derivation The derivation here will be similar to the case of the hidden state as the distributions or of the same family. We call θ_m a generic m by c parameter matrix where m is the number of hidden clusters and c the number of categories/outcomes in the observed multinomial distributions.

2.2.3 Poisson random variables

This distribution should be used to model random variables that represent counts or the occurrence of events.

Log likelihood

$$\log p(v^i = v|h^i, \Lambda) \propto v \log \Lambda - \Lambda$$

Optimal parameter value

$$\arg\max_{\Lambda \in \theta} \mathbb{E}(\theta) = \frac{\sum_{i=1}^{n} q^i(h) v^i}{\sum_{i=1}^{n} q^i(h)}$$

Derivations

Log likelihood A Poisson random variable has only rate parameter generally labelled λ. Let us define $\Lambda \in \theta$ to be a vector of length c of rate parameters where each entry is the rate parameter corresponding to cluster c, represented by variable h. Let us remind ourselves then of the expression for the likelihood of observing a certain count given the clusters:

$$p(v^i = v|h^i, \Lambda) = \frac{\Lambda^v e^{-\Lambda}}{v!} \propto \Lambda^v e^{-\Lambda}$$

The log likelihood is then:

$$\begin{aligned}\log p(v^i = v|h^i, \Lambda) &= \log\left[\frac{\Lambda^v e^{-\Lambda}}{v!}\right] \\ &= \log \Lambda^v + \log e^{-\Lambda} - \log v! \\ &= v \log \Lambda - \Lambda - \log v!\end{aligned}$$

Optimisation Let us now find the expression for the values of $\Lambda \in \theta$ that maximise the energy $\mathbb{E}(\theta)$. We can start from the generic expression of the gradient of the energy w.r.t to the parameters Λ of a visible variable v.

$$\begin{aligned}\nabla_\Lambda \mathbb{E}(\theta) &= \nabla_\Lambda \left[\sum_{i=1}^{n} \left\langle \log p(v^i|h^i, \theta)\right\rangle_{q^i(h)}\right] \\ &= \nabla_\Lambda \left[\sum_{i=1}^{n} \left\langle v^i \log \Lambda - \Lambda - \log v^i!\right\rangle_{q^i(h)}\right] \\ &= \sum_{i=1}^{n} \nabla_\Lambda \sum_h q^i(h)(v^i \log \Lambda - \Lambda - \log v^i!) \\ &= \sum_{i=1}^{n} q^i(h) \nabla_\Lambda (v^i \log \Lambda - \Lambda - \log v^i!) \\ &= \sum_{i=1}^{n} q^i(h) \left(\frac{v^i}{\Lambda} - 1\right)\end{aligned}$$

We set this to 0 to optimise and get the following expression.

$$\begin{aligned}\sum_{i=1}^{n} q^i(h) \left(\frac{v^i}{\Lambda} - 1\right) &= 0 \\ \sum_{i=1}^{n} q^i(h) \frac{v^i}{\Lambda} &= \sum_{i=1}^{n} q^i(h) \\ \Lambda &= \frac{\sum_{i=1}^{n} q^i(h) v^i}{\sum_{i=1}^{n} q^i(h)}\end{aligned}$$

Poisson random variables are subject to the constraint $\Lambda > 0$. Here we can see that the output from the optimisation equation cannot be negative. Indeed, as v^i is a count, it will always be a positive integer and $q^i(h)$ is a multinomial probability distribution, which also will always be positive. We therefore do not need to use an optimisation constraint here, the maximum will always satisfy $\Lambda > 0$.

2.2.4 Normal distribution (Gaussian), known variance unknown mean

This distribution should be used to model the behaviour of continuous variables that can take any value in $\mathbb{R}$. Can also be used for positive or negative continuous quantities that do not describe event occurrence, e.g. height, scores on a scale, etc.

Log likelihood

$$\log p(v^i = v|h^i, \theta^\mu) \propto -\frac{1}{2\sigma^2}(v - \theta^\mu)^2$$

Optimal parameter value

$$\arg\max_{\theta^\mu} \mathbb{E}(\theta) = \frac{\sum_{i=1}^n q^i(h) v^i}{\sum_{i=1}^n q^i(h)}$$

Derivations

Log likelihood We will deal with the case when we assume knowledge of the variance, e.g. approximated by the sample variance, but expect means to differ between clusters. Therefore, we only have one set of parameters to learn here, which we will call θ^μ and is a 1d vector of length c, where each entry will be the mean for each cluster. Let us remind ourselves of the form of the normal likelihood of observing v, with a sample variance σ^2 and possible means θ^μ depending on the cluster:

$$p(v^i = v|h^i, \theta^\mu) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(v-\theta^\mu)^2} \propto e^{-\frac{1}{2\sigma^2}(v-\theta^\mu)^2}$$

The log likelihood is then

$$\begin{aligned}
\log p(v^i = v|h^i, \theta^\mu) &= \log \left[\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(v-\theta^\mu)^2} \right] \\
&= \log \frac{1}{\sqrt{2\pi\sigma^2}} + \log e^{-\frac{1}{2\sigma^2}(v-\theta^\mu)^2} \\
&= \log \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2\sigma^2}(v - \theta^\mu)^2 \\
&= -\frac{1}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}(v - \theta^\mu)^2 \\
&\propto -\frac{1}{2\sigma^2}(v - \theta^\mu)^2
\end{aligned}$$

Optimisation Starting from the generic form of the gradient of the energy, $\mathbb{E}(\theta)$, w.r.t to the parameters, θ^μ, of a visible variable, v, all other terms disappear and we have the following:

$$\nabla_{\theta^\mu} \mathbb{E}(\theta) = \nabla_{\theta^\mu} \left[\sum_{i=1}^n \left\langle \log p(v^i|h^i, \theta) \right\rangle_{q^i(h)} \right]$$

replacing the log likelihood by the expression found above

$$
\begin{aligned}
&= \nabla_{\theta^\mu} \left[\sum_{i=1}^{n} \left\langle \log p(v^i|h^i,\theta) \right\rangle_{q^i(h)} \right] \\
&= \nabla_{\theta^\mu} \left[\sum_{i=1}^{n} \left\langle -\frac{1}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}(v-\theta^\mu)^2 \right\rangle_{q^i(h)} \right] \\
&= -\sum_{i=1}^{n} \nabla_{\theta^\mu} \left\langle \frac{1}{2}\log(2\pi\sigma^2) + \frac{1}{2\sigma^2}(v-\theta^\mu)^2 \right\rangle_{q^i(h)} \\
&= -\sum_{i=1}^{n} \nabla_{\theta^\mu} \sum_h q^i(h) \left[\frac{1}{2}\log(2\pi\sigma^2) + \frac{1}{2\sigma^2}(v-\theta^\mu)^2 \right] \\
&= -\sum_{i=1}^{n} q^i(h) \nabla_{\theta^\mu} \left[\frac{1}{2}\log(2\pi\sigma^2) + \frac{1}{2\sigma^2}(v-\theta^\mu)^2 \right] \\
&= \sum_{i=1}^{n} q^i(h) \frac{1}{\sigma^2}(v^i - \theta^\mu)
\end{aligned}
$$

Notation is unclear here, but the sum over h disappears because when taking the gradient, each value for h that is not the one we are differentiating w.r.t. amounts to 0.

We can now optimise by setting this to 0. Note that we do not use a Lagrange multiplier as the mean of the normal distribution can take any value on the real line.

$$
\begin{aligned}
\sum_{i=1}^{n} q^i(h) \frac{1}{\sigma^2}(v^i - \theta^\mu) &= 0 \\
\sum_{i=1}^{n} q^i(h) \frac{1}{\sigma^2}\theta^\mu &= \sum_{i=1}^{n} q^i(h) \frac{1}{\sigma^2} v^i \\
\frac{1}{\sigma^2}\theta^\mu \sum_{i=1}^{n} q^i(h) &= \sum_{i=1}^{n} q^i(h) \frac{1}{\sigma^2} v^i \\
\theta^\mu \sum_{i=1}^{n} q^i(h) &= \sum_{i=1}^{n} q^i(h) v^i \\
\theta^\mu &= \frac{\sum_{i=1}^{n} q^i(h) v^i}{\sum_{i=1}^{n} q^i(h)}
\end{aligned}
$$

2.3 Expectation steps

The below equation depicts the evidence lower bound for the marginal log likelihood as the sum between the negative KL divergence between the variational distribution and the probability distribution of the hidden states given the observed variables.

$$\log p(v|\theta) \geq -\mathrm{KL}[p(h|v,\theta)||q(h)] + \log p(v|\theta)$$

As the KL divergence is maximal, i.e. equal to 0, when both distribution are the same, we get the following when the variational distribution, $q(h)$, is set to $p(h|v,\theta)$:

$$\begin{aligned}\log p(v|\theta) &\geq -\mathrm{KL}[p(h|v,\theta)||p(h|v,\theta)] + \log p(v|\theta)\\ \log p(v|\theta) &\geq -0 + \log p(v|\theta)\\ \log p(v|\theta) &= \log p(v|\theta)\end{aligned}$$

Thus we can see that the bound is equal when $q(h) = p(h|v,\theta)$. Therefore the expectation step is simply to set the variational distribution to be equal to the posterior. We do not use the normalisation constraint, as explained in the appendix.

Let us derive a generic expression for this. From Bayes' rule and our models' constraints, we have:

$$\begin{aligned}p(h^i|v^i,\theta) &= \frac{p(v^i,h^i|\theta)}{p(v^i|\theta)}\\ &= \frac{p(h^i|\theta)\prod_{j=1}^k p(v_j^i|h^i,\theta)}{\sum_h p(h^i|\theta)\prod_{j=1}^k p(v_j^i|h^i,\theta)}\\ &\propto p(h^i|\theta)\prod_{j=1}^k p(v_j^i|h^i,\theta)\end{aligned}$$

Taking the log on both side yields:

$$\begin{aligned}\log p(h^i|v^i,\theta) &\propto \log\left[p(h^i|\theta)\prod_{j=1}^k p(v_j^i|h^i,\theta)\right]\\ &= \log p(h^i|\theta) + \sum_{j=1}^k \log p(v_j^i|h^i,\theta)\end{aligned}$$

This expression is easy to deal with as all log probabilities are well separated in their own addition term. We can use the log of the distribution in each case, which are well known for distributions in the exponential family. Also, we use the log likelihoods up to proportionality as we are then normalising $p(h^i|v^i,\theta)$.

3 Appendix

3.1 Numerical stability

Normalisation of small log likelihoods As the expectation steps requires us to compute log likelihoods for the joint distribution of each observation without allowing us easy compute the normalisation constant, we need to come up with a solution to deal with very negative log values so that they do not go to 0 when exponentiated. Let us remind ourselves of the issue. We have, for the expectation step, the following quantity to compute:

$$\begin{aligned}p(h^i|v^i,\theta) &= \frac{p(v^i,h^i|\theta)}{p(v^i|\theta)}\\ &= \frac{p(h^i|\theta)\prod_{j=1}^k p(v_j^i|h^i,\theta)}{\sum_h p(h^i|\theta)\prod_{j=1}^k p(v_j^i|h^i,\theta)}\end{aligned}$$

Taking the log on both side yields:

$$\begin{aligned}
\log p(h^i|v^i,\theta) &= \log\left[\frac{p(h^i|\theta)\prod_{j=1}^{k} p(v_j^i|h^i,\theta)}{\sum_h p(h^i|\theta)\prod_{j=1}^{k} p(v_j^i|h^i,\theta)}\right] \\
&= \log\left[p(h^i|\theta)\prod_{j=1}^{k} p(v_j^i|h^i,\theta)\right] - \log\left[\sum_h p(h^i|\theta)\prod_{j=1}^{k} p(v_j^i|h^i,\theta)\right] \\
&= \log p(h^i|\theta) + \sum_{j=1}^{k}\log p(v_j^i|h^i,\theta) - \log\left[\sum_h p(h^i|\theta)\prod_{j=1}^{k} p(v_j^i|h^i,\theta)\right]
\end{aligned}$$

The first term is the log likelihood and can be easily computed by pushing the log inside and adding all inner terms. However, as we are marginalising h out, the normalisation constant is a sum and is not easily computable. It would not be a problem for cases where we do not have a lot of features but we still need a method that can deal with cases in which the probabilities get too small to be numerically approximated.

The solution is to find a way to not have to compute the normalising constant and "normalise" the log likelihood. A thread on StackExchange provides an elegant solution. The thread can be found here: https://stats.stackexchange.com/questions/66616/converting-normalizing-very-small-likelihood-values-to-probability.

The idea is to subtract the maximum logarithm from all log in the likelihood array. This will not change the relative weights of each case and will render exponentiation much easier. The index of the maximum will be 0 which will yield 1 when exponentiated, with a lower log values indices being closer to 0.

The next step is to define an underflow threshold when using a programming language that does not automatically output 0 for values that underflow, which is not the case for python.

Therefore, if we have a log likelihood array, $\mathbf{L}$, with m entries $\lambda_1, \lambda_2, ..., \lambda_m$, and maximum entry we perform the following operation:

$$\mathbf{L}_{new} = \mathbf{L} - \max \mathbf{L}$$

We can then exponentiate L_{new} element-wise and then normalise.