Expectation-Maximization algorithm: Gaussian mixture models and Sigmoid belief networks

Caio Corro

Gaussian Mixture Model

Random variables

- $\mathcal{Y}$ taking values in $\{1...k\}$ — represents the choice of one (latent) cluster in k
- $\mathcal{X}$ taking values in $\mathbb{R}^d$ — represents the observed point

Generative story

1. $y \sim p_\theta(\mathcal{Y})$
2. $\boldsymbol{x} \sim p_\theta(\mathcal{X}|\mathcal{Y} = y)$

=> locally normalized models

Parameterization: $\theta = \{\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2\}$

- Prior distribution: $\boldsymbol{\lambda} \in \triangle(k)$, i.e. $p_\theta(\mathcal{Y} = y) = \lambda_y$.
- Conditional distribution: $\boldsymbol{\mu} \in \mathbb{R}^{k \times d}$ and $\boldsymbol{\sigma}^2 \in \mathbb{R}^{k \times d}_{++}$,
  i.e. $p_\theta(\boldsymbol{x}|y) = \prod_{i=1}^{j} p_\theta(x_i|y) = \prod_{i=1}^{j} f(x_i, \mu_{y,i}, \sigma^2_{y,i})$,
  where f is the PDF of univariate Gaussian distributions.

Gradient-based learning

Let $\mathcal{D} = \{\boldsymbol{x}\}_{i=1}^{n}$ be a training dataset of n datapoints.

Training objective

Maximize the log-likelihood of the training data (the evidence of the data)

$$\arg\max_{\theta\in\Theta} \sum_{\boldsymbol{x}\in\mathcal{D}} \log p_\theta(\boldsymbol{x}) = \arg\min_{\theta\in\Theta} - \sum_{\boldsymbol{x}\in\mathcal{D}} \log p_\theta(\boldsymbol{x}) = \arg\min_{\theta\in\Theta} - \sum_{\boldsymbol{x}\in\mathcal{D}} \log \underbrace{\sum_{y} p_\theta(y) p_\theta(\boldsymbol{x}|y)}_{\substack{\text{marginalization over}\\ \text{latent variables}}}$$

where θ is the set of all parameters the GMM and Θ the set of well-defined θ.

Training algorithm

- We can reparameterize the parameters so they are unconstrained
- We can simply use gradient descent on the objective with reparameterized variables! :)

Expectation-Maximization algorithm

Why another algorithm?

Gradient descent:

- Pros: trivial implementation via Pytorch
- Cons: you need to define a stepsize

Expectation-Maximization:

- Pros: no stepsize!
- Cons: you have to write the optimization code yourself

(there are other favors for EM, but outside the scope of this course)

Intuition of EM

- Derive a parameterized lower bound to maximization objective
- Interleave maximization of the lower bound parameters and the model parameters

EM objective

Evidence lower bound (ELBO)

$$\log \mathbb{E}_{p(\mathcal{Y})}[p(\boldsymbol{x}|\mathcal{Y})] \quad \geq \quad \mathbb{E}_{q(\mathcal{Y})}[\log p(\mathcal{Y})p(\boldsymbol{x}|\mathcal{Y})] + H[q(\mathcal{Y})]$$

where q is a proposal distribution and H the Shannon entropy.

EM objective

Evidence lower bound (ELBO)

$$\log \mathbb{E}_{p(\mathcal{Y})}[p(\boldsymbol{x}|\mathcal{Y})] \quad \geq \quad \mathbb{E}_{q(\mathcal{Y})}[\log p(\mathcal{Y})p(\boldsymbol{x}|\mathcal{Y})] + H[q(\mathcal{Y})]$$

where q is a proposal distribution and H the Shannon entropy.

Proposal distribution for GMMs

$$q_\phi(\mathcal{Y} = y|\mathcal{X} = \boldsymbol{x}) \quad = \quad \phi_y^{(\boldsymbol{x})}$$

where $\phi^{(\boldsymbol{x})} \in \triangle(k)$ for each $\boldsymbol{x}$ in the training set.

New objective

$$\begin{aligned} & \max_{\theta \in \Theta} \sum_{\boldsymbol{x} \in \mathcal{D}} \log \sum_{y} p_\theta(y) p_\theta(\boldsymbol{x}|y) \\ \geq \quad & \max_{\theta \in \Theta, \phi \in \Phi} \sum_{\boldsymbol{x} \in \mathcal{D}} \mathbb{E}_{q_\phi(\mathcal{Y}|\boldsymbol{x})}[\log p_\theta(y) p_\theta(\boldsymbol{x}|y)] + H[q_\phi(\mathcal{Y}|\boldsymbol{x})] \end{aligned}$$

The two steps of EM

EM objective

$$\max_{\theta\in\Theta,\phi\in\Phi} \sum_{\boldsymbol{x}\in\mathcal{D}} \mathbb{E}_{q_\phi(\mathcal{Y}|\boldsymbol{x})}[\log p_\theta(y)p_\theta(\boldsymbol{x}|y)] + H[q_\phi(\mathcal{Y}|\boldsymbol{x})]$$
$$= \max_{\theta\in\Theta,\phi\in\Phi} \sum_{\boldsymbol{x}\in\mathcal{D}} \textsc{Elbo}(\boldsymbol{x},\theta,\phi)$$

EM algorithm

Compute a sequence of parameters $\phi^{(1)},\theta^{(1)},\phi^{(2)},\theta^{(2)},\phi^{(3)},\theta^{(3)}\ldots$ as follows:

- E step: $\phi^{(t+1)} = \arg\max_{\phi\in\Phi} \sum_{i=1}^n \textsc{Elbo}(\boldsymbol{x},\theta^{(t)},\phi)$
- M step: $\theta^{(t+1)} = \arg\max_{\theta\in\Theta} \sum_{i=1}^n \textsc{Elbo}(\boldsymbol{x},\theta,\phi^{(t+1)})$

Comments:

- the new parameters ϕ computed in the E step are immediately used for the M step
- this is just a block coordinate ascent algorithm

Expectation step 1/4

$$\phi^{(t+1)} = \operatorname*{arg\,max}_{\phi \in \boldsymbol{\Phi}} \sum_{\boldsymbol{x} \in \mathcal{D}} \textsc{Elbo}(\boldsymbol{x}, \theta^{(t)}, \phi^{(t)})$$

Parameter decomposition

The parameters of the proposal distribution decomposes over training examples, so we can compute each one independently:

$$\phi^{(t+1)}(\boldsymbol{x}) = \operatorname*{arg\,max}_{\phi \in \triangle(k)} \textsc{Elbo}(\boldsymbol{x}, \theta^{(t)}, \boldsymbol{\phi})$$

Intuition

For a given set of parameters θ, the E step computes the best proposal distribution possible (i.e. so the bound is the best possible bound)

Expectation step 2/4

KL divergence

$$D_{\mathrm{KL}}[q(\mathcal{Y})|p(\mathcal{Y})] = \sum_{y} q(y) \log \frac{q(y)}{p(y)}$$

- non-negative, i.e. $D_{\mathrm{KL}}[q(\mathcal{Y})|p(\mathcal{Y})] \geq 0$
- non-necessarily symmetric, i.e. $D_{\mathrm{KL}}[q(\mathcal{Y})|p(\mathcal{Y})] \neq D_{\mathrm{KL}}[p(\mathcal{Y})|q(\mathcal{Y})]$
- null iff the two distributions are equal: $D_{\mathrm{KL}}[q(\mathcal{Y})|p(\mathcal{Y})] = 0 \iff q(y) = p(y), \forall y$

Expectation step 3/4

Evidence lower bound (ELBO)

$$\log \mathbb{E}_{p(\mathcal{Y})}[p(\boldsymbol{x}|\mathcal{Y})] \quad \geq \quad \mathbb{E}_{q(\mathcal{Y})}[\log p(\mathcal{Y})p(\boldsymbol{x}|\mathcal{Y})] + H[q(\mathcal{Y})]$$

where q is a proposal distribution and H the Shannon entropy.

ELBO gap

$$\log \mathbb{E}_{p(\mathcal{Y})}[p(\boldsymbol{x}|\mathcal{Y})] - \mathbb{E}_{q(\mathcal{Y})}[\log p(\mathcal{Y})p(\boldsymbol{x}|\mathcal{Y})] + H[q(\mathcal{Y})] \quad = \quad D_{\mathrm{KL}}[q(\mathcal{Y})|p(\mathcal{Y}|\boldsymbol{x})]$$

where D_{KL} is the Kullback-Leibler divergence.

Expectation step 4/4

$$\phi^{(t+1)}(\boldsymbol{x}) = \underset{\phi \in \triangle(k)}{\arg\max}\ \textsc{Elbo}(\boldsymbol{x}, \theta^{(t)}, \boldsymbol{\phi})$$

How to solve the E step?

By the ELBO gap, we know that the objective is maximized if the proposal distribution is equal to the posterior distribution!

$$q(y|\boldsymbol{x}) = p(y|\boldsymbol{x}) = \frac{p(y)p(\boldsymbol{x}|y)}{\sum_{y'} p(y')p(\boldsymbol{x}|y')}$$

=> very easy to compute the optimal parameters ϕ in the E step

Maximization step

$$\theta^{(t+1)} = \arg\max_{\theta \in \Theta} \sum_{\boldsymbol{x} \in \mathcal{D}} \textsc{Elbo}(\boldsymbol{x}, \theta, \phi^{(t+1)})$$

How to solve the M step?

Ignore the constraints on the variance parameters, and simply compute the closed form expression using first order optimality methods!

Solution: looks like "weighted" means and variances.

Exercises

1. Compute the ELBO gap
2. Derive the E step solution using KKT conditions instead of the ELBO gap — is the result expected?
3. Compute the closed form solution for the M step (for the E step it's too trivial)

Sigmoid belief network (two layers only)

Random variables

- $\mathcal{Y}$ taking values in $[0,1]^k$ (latent)
- $\mathcal{X}$ taking values in $[0,1]^d$ (observed)

Parameterization $\theta = \{\boldsymbol{a}, \boldsymbol{B}, \boldsymbol{c}\}$

$$p_\theta(\boldsymbol{x}, \boldsymbol{y}) = p_\theta(\boldsymbol{y}) p_\theta(\boldsymbol{x}|\boldsymbol{y}) = \prod_{i=1}^{k} p_\theta(y_i) \prod_{i=1}^{d} p_\theta(x_i|\boldsymbol{y}) = \prod_{i=1}^{k} \frac{\exp(y_i a_i)}{1+\exp(a_i)} \prod_{i=1}^{d} \frac{\exp(x_i(\boldsymbol{B}_i \boldsymbol{y} + c_i))}{1+\exp(\boldsymbol{B}_i \boldsymbol{y} + c_i)}$$

where $\boldsymbol{a} \in \mathbb{R}^k$, $\boldsymbol{B} \in \mathbb{R}^{d\times k}$ and $\boldsymbol{c} \in \mathbb{R}^d$.

Generative story

1. $\boldsymbol{y} \sim p_\theta(\mathcal{Y})$
2. $\boldsymbol{x} \sim p_\theta(\mathcal{X}|\mathcal{Y} = \boldsymbol{y})$

$=>$ sampling from independant Bernoullis

SBN training

Can we directly use gradient ascent to?

Evidence of a training datapoint $\boldsymbol{x} \in \mathcal{D}$:

$$\log p_\theta(\boldsymbol{x}) = \log \sum_{\boldsymbol{y}} p_\theta(\boldsymbol{y}) p_\theta(\boldsymbol{x}|\boldsymbol{y})$$

$=>$ sum over 2^k values for $\boldsymbol{y}$, **intractable!!!**

We can't even compute the objective, not even mentioning the gradient

SBN training

Can we directly use gradient ascent to?

Evidence of a training datapoint $\boldsymbol{x} \in \mathcal{D}$:

$$\log p_\theta(\boldsymbol{x}) = \log \sum_{\boldsymbol{y}} p_\theta(\boldsymbol{y}) p_\theta(\boldsymbol{x}|\boldsymbol{y})$$

$=>$ sum over 2^k values for $\boldsymbol{y}$, **intractable!!!**
We can't even compute the objective, not even mentioning the gradient

Can we apply EM?

- The proposal distribution $q_\phi(\boldsymbol{y}|\boldsymbol{x})$ would require 2^k parameters per training point
- The E step closed-form expression requires summing over 2^k values (denominator in Bayes rule)

$=>$ **intractable again :(**

Mean Field Theory (MFT)

Mean Field assumption

Assume independence between dimension of latent space in the proposal distribution:

$$q_\phi(\boldsymbol{z}|\boldsymbol{x}) = \prod_{i=1}^{k} (\phi_i^{(\boldsymbol{x})})^{z_i} (1 - \phi^{(\boldsymbol{x})})_i^{1-z_i}$$

where $\phi^{(\boldsymbol{x})} \in [0,1]^k$ are the parameters of the proposal distribution associated with observation $\boldsymbol{x}$.

What does it changes?

- Cons: (probably) not possible to have a tight ELBO anymore (gap $= 0$)
- Pros: can make computation tractable

ELBO for SBNs

$$
\begin{aligned}
\log p_\theta(\boldsymbol{x}) &= \log \sum_{\boldsymbol{y} \in Y} p_\theta(\boldsymbol{y}) p_\theta(\boldsymbol{x}|\boldsymbol{y}) \\
&\geq \underbrace{\mathbb{E}_{q_\phi(\mathcal{Y}|\boldsymbol{x})}[\log p_\theta(\mathcal{Y})]}_{\textbf{(a)}} + \underbrace{\mathbb{E}_{q_\phi(\mathcal{Y}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\mathcal{Y})]}_{\textbf{(b)}} + \underbrace{H^{\mathrm{S}}[q_\phi(\mathcal{Y}|\boldsymbol{x})]}_{\textbf{(c)}}
\end{aligned}
$$

ELBO for SBNs

$$
\begin{aligned}
\log p_\theta(\boldsymbol{x}) &= \log \sum_{\boldsymbol{y} \in Y} p_\theta(\boldsymbol{y}) p_\theta(\boldsymbol{x}|\boldsymbol{y}) \\
&\geq \underbrace{\mathbb{E}_{q_\phi(\mathcal{Y}|\boldsymbol{x})}[\log p_\theta(\mathcal{Y})]}_{\textbf{(a)}} + \underbrace{\mathbb{E}_{q_\phi(\mathcal{Y}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\mathcal{Y})]}_{\textbf{(b)}} + \underbrace{H^{\mathrm{S}}[q_\phi(\mathcal{Y}|\boldsymbol{x})]}_{\textbf{(c)}} \\
&\geq \underbrace{\langle \boldsymbol{a}, \phi^{(\boldsymbol{x})} \rangle - \sum_{i=1}^{k} \log(1 + \exp(a_i))}_{\textbf{(a)}}
\end{aligned}
$$

ELBO for SBNs

$$\log p_\theta(\boldsymbol{x}) = \log \sum_{\boldsymbol{y}\in Y} p_\theta(\boldsymbol{y}) p_\theta(\boldsymbol{x}|\boldsymbol{y})$$

$$\geq \underbrace{\mathbb{E}_{q_\phi(\mathcal{Y}|\boldsymbol{x})}[\log p_\theta(\mathcal{Y})]}_{\textbf{(a)}} + \underbrace{\mathbb{E}_{q_\phi(\mathcal{Y}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\mathcal{Y})]}_{\textbf{(b)}} + \underbrace{H^{\mathrm{S}}[q_\phi(\mathcal{Y}|\boldsymbol{x})]}_{\textbf{(c)}}$$

$$\geq \underbrace{\langle \boldsymbol{a}, \phi^{(\boldsymbol{x})} \rangle - \sum_{i=1}^{k} \log(1 + \exp(a_i))}_{\textbf{(a)}}$$

$$\underbrace{\langle \boldsymbol{x}, \boldsymbol{B}\phi^{(\boldsymbol{x})} \rangle + \langle \boldsymbol{x}, \boldsymbol{c} \rangle - \sum_{i=1}^{d} \log \left(1 + \exp(c_i) \prod_{j=1}^{k} (1 - \phi_j^{(\boldsymbol{x})} + \phi_j^{(\boldsymbol{x})} \exp(B_{i,j})) \right)}_{\text{lower bound on } \textbf{(b)}}$$

ELBO for SBNs

$$\log p_\theta(\boldsymbol{x}) = \log \sum_{\boldsymbol{y} \in Y} p_\theta(\boldsymbol{y}) p_\theta(\boldsymbol{x}|\boldsymbol{y})$$

$$\geq \underbrace{\mathbb{E}_{q_\phi(\mathcal{Y}|\boldsymbol{x})}[\log p_\theta(\mathcal{Y})]}_{\textbf{(a)}} + \underbrace{\mathbb{E}_{q_\phi(\mathcal{Y}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\mathcal{Y})]}_{\textbf{(b)}} + \underbrace{H^{\mathrm{S}}[q_\phi(\mathcal{Y}|\boldsymbol{x})]}_{\textbf{(c)}}$$

$$\geq \underbrace{\langle \boldsymbol{a}, \phi^{(\boldsymbol{x})} \rangle - \sum_{i=1}^{k} \log(1 + \exp(a_i))}_{\textbf{(a)}}$$

$$\underbrace{\langle \boldsymbol{x}, \boldsymbol{B}\phi^{(\boldsymbol{x})} \rangle + \langle \boldsymbol{x}, \boldsymbol{c} \rangle - \sum_{i=1}^{d} \log \left(1 + \exp(c_i) \prod_{j=1}^{k} (1 - \phi_j^{(\boldsymbol{x})} + \phi_j^{(\boldsymbol{x})} \exp(B_{i,j})) \right)}_{\text{lower bound on } \textbf{(b)}}$$

$$\underbrace{\sum_{i=1}^{k} \left(\phi_i^{(\boldsymbol{x})} \log \phi_i^{(\boldsymbol{x})} + (1 - \phi_i^{(\boldsymbol{x})}) \log(1 - \phi_i^{(\boldsymbol{x})}) \right)}_{\textbf{(c)}}$$

Mean Field EM for SBN

Algorithm

- E step: maximize the $\textsc{Elbo}$:
  1. write down the first-order optimality condition for ϕ
  2. solve the resulting problem with iterative equation solving method
- M step: one step of gradient ascent on the $\textsc{Elbo}$ wrt model parameters θ

..

Difference with GMMs

- Cannot use the $\textsc{Elbo}$ gap and Bayes rule in EM
- No closed-form expression for step E or M

Upper bound on the evidence

Question

As we cannot close the Elbo gap in the E step,
how to evaluate the quality of the resulting bound?

Upper bound on the evidence

Question

As we cannot close the Elbo gap in the E step,
how to evaluate the quality of the resulting bound?

Variational formulation of the sigmoid

$$\sigma(u) = \inf_{\epsilon \in [0,1]} \exp(\epsilon u - H^{\mathrm{FD}}[\epsilon])$$

where $H^{\mathrm{FD}}[\epsilon] = -\epsilon \log \epsilon - (1-\epsilon)\log(1-\epsilon)$ is the Fermi-Dirac entropy.

Upper bound

For any $\boldsymbol{\epsilon} \in [0,1]^d$:

$$\log p_\theta(\boldsymbol{x}) \leq \left(\begin{array}{l} -\sum_{i=1}^d H^{\mathrm{FD}}[\epsilon_1] + \sum_{i=1}^d \epsilon_i c_i (2x_i - 1) \\ +\sum_{j=1}^k \left(1 - \sigma(a_i) + \sigma(a_i) \exp(\sum_{i=1}^d \epsilon_i \boldsymbol{B}_{i,j}(2x_i - 1))\right) \end{array} \right)$$

(not trivial to find the best variational parameters for this bound!)