Expectation Maximization Algorithm

David Rosenberg

New York University

October 29, 2016

Kullback-Leibler Divergence

- Let $p(x)$ and $q(x)$ be PMFs on $\mathcal{X}$.
- How can we measure how "different" p and q are?
- The **Kullback-Leibler** or **"KL" Diverence** is defined by

$$\mathrm{KL}(p\|q) = \sum_x p(x)\log\frac{p(x)}{q(x)}.$$

(Assumes $q(x)=0$ implies $p(x)=0$.)

- Can also write this as

$$\mathrm{KL}(p\|q) = \mathbb{E}_p \log\frac{p(X)}{q(X)},$$

where $X \sim p(x)$.

Gibbs Inequality ($\mathrm{KL}(p\|q) \geqslant 0$)

Theorem (Gibbs Inequality)

Let $p(x)$ and $q(x)$ be PMFs on $\mathcal{X}$. Then

$$KL(p\|q) \geqslant 0,$$

with equality iff $p(x) = q(x)$ for all $x \in \mathcal{X}$.

- KL divergence measures the "distance" between distributions.
- Note:
  - KL divergence **not a metric**.
  - KL divergence is **not symmetric**.

Jensen's Inequality

Theorem (Jensen's Inequality)

If $f : \mathcal{X} \to \mathbf{R}$ is a ***convex*** *function, and $X \in \mathcal{X}$ is a random variable, then*

$$\mathbb{E}f(X) \geqslant f(\mathbb{E}X).$$

Moreover, if f is ***strictly convex****, then equality implies that $X = \mathbb{E}X$ with probability 1 (i.e. X is a constant).*

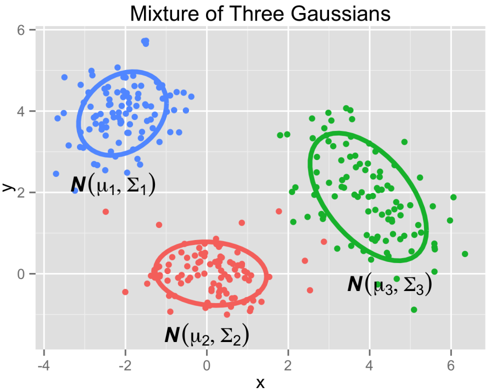
- e.g. $f(x) = x^2$ is convex. So $\mathbb{E}X^2 \geqslant (\mathbb{E}X)^2$. Thus

$$\mathrm{Var}X = \mathbb{E}X^2 - (\mathbb{E}X)^2 \geqslant 0.$$

- Jensen's inequality is used to prove Gibbs inequality ($\log(x)$ is strictly concave).

Gaussian Mixture Model ($k = 3$)

1. Choose $Z \in \{1,2,3\} \sim \text{Multi}\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$.
2. Choose $X \mid Z = z \sim \mathcal{N}(X \mid \mu_z, \Sigma_z)$.



Gaussian Mixture Model (k Components)

- GMM Parameters

$$\begin{aligned} \text{Cluster probabilities :} \quad & \pi = (\pi_1, \ldots, \pi_k) \\ \text{Cluster means :} \quad & \mu = (\mu_1, \ldots, \mu_k) \\ \text{Cluster covariance matrices:} \quad & \Sigma = (\Sigma_1, \ldots \Sigma_k) \end{aligned}$$

- Let $\theta = (\pi, \mu, \Sigma)$.
- Marginal log-likelihood

$$\log p(x \mid \theta) = \log \left\{ \sum_{z=1}^{k} \pi_z \mathcal{N}(x \mid \mu_z, \Sigma_z) \right\}$$

General Latent Variable Model

- Two sets of random variables: Z and X.
- Z consists of unobserved **hidden variables**.
- X consists of **observed variables**.
- Joint probability model parameterized by $\theta \in \Theta$:

$$p(x, z \mid \theta)$$

Notation abuse

Notation $p(x, z \mid \theta)$ suggests a Bayesian setting, in which θ is a r.v. However we are **not** assuming a Bayesian setting. $p(x, z \mid \theta)$ is just easier to read than $p_\theta(x, z)$, once θ gets more complicated.

Complete and Incomplete Data

- An observation of X is called an **incomplete data set**.
- An observation (X, Z) is called a **complete data set**.
    - We never have a complete data set for latent variable models.
    - But it's a useful construct.

- Suppose we have an incomplete data set $\mathcal{D} = (x_1, \ldots, x_n)$.
- To simplify notation, take X to represent the entire dataset

$$X = (X_1, \ldots, X_n),$$

and Z to represent the corresponding unobserved variables

$$Z = (Z_1, \ldots, Z_n).$$

Log-Likelihood

- The log-likelihood of θ for observation $X = x$ is

$$\log p(x \mid \theta) \quad = \quad \log\left\{\sum_z p(x, z \mid \theta)\right\}.$$

- (We write discrete case – everything same for continuous case.)
- For exponential families,
    - Without the sum "$\sum_z$", things simplify.
    - The log and the exp cancel out.
- **Assumption for the EM algorithm**:
    - Optimization for complete data is relatively easy

$$\underset{\theta \in \Theta}{\arg\max} \log p(x, z \mid \theta)$$

    - (We'll actually need slightly more than this.)

The EM Algorithm **Key Idea**

- Marginal log likelihood is hard to optimize:

$$\max_{\theta} \log \left\{ \sum_{z} p(x, z \mid \theta) \right\}$$

- Full log-likelihood would be easy to optimize:

$$\max_{\theta} \log p(x, z \mid \theta)$$

- What if we had a **distribution** $q(z)$ for the latent variables Z?
  - e.g. $q(z) = p(z \mid x, \theta)$
- Could maximize the **expected complete data log-likelihood**:

$$\max_{\theta} \sum_{z} q(z) \log p(x, z \mid \theta)$$

A Lower Bound for Marginal Likelihood

- Let $q(z)$ be any PMF on $\mathcal{Z}$, the support of Z:

$$\begin{aligned}
\log p(x \mid \theta) &= \log \sum_z p(x, z \mid \theta) \\
&= \log \sum_z q(z) \left[\frac{p(x, z \mid \theta)}{q(z)} \right] \\
&\geqslant \sum_z q(z) \log \left(\frac{p(x, z \mid \theta)}{q(z)} \right) \\
&=: \mathcal{L}(q, \theta).
\end{aligned}$$

- The inequality is by Jensen's, by concavity of the log.

Lower Bound and Expected Complete Log-Likelihood

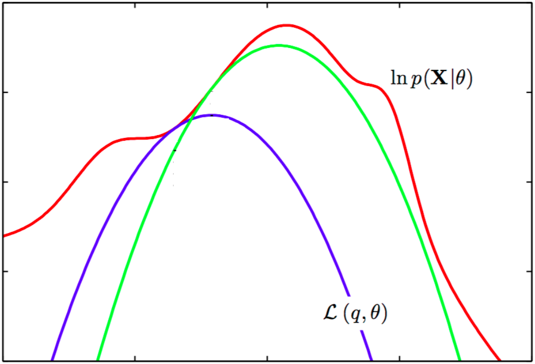
- Consider maximizing the lower bound $\mathcal{L}(q,\theta)$:

$$\begin{aligned}\mathcal{L}(q,\theta) &= \sum_z q(z)\log\left(\frac{p(x,z\mid\theta)}{q(z)}\right)\\ &= \underbrace{\sum_z q(z)\log p(x,z\mid\theta)}_{\mathbb{E}[\text{complete log-likelihood}]} - \underbrace{\sum_z q(z)\log q(z)}_{\text{no }\theta\text{ here}}\end{aligned}$$

- Maximizing $\mathcal{L}(q,\theta)$ equivalent to maximizing $\mathbb{E}$ [complete data log-likelihood].

A Family of Lower Bounds

- Each q gives a different lower bound: $\log p(x \mid \theta) \geqslant \mathcal{L}(q, \theta)$
- Two lower bounds, as functions of θ:



From Bishop's *Pattern recognition and machine learning*, Figure 9.14.

EM: Coordinate Ascent on Lower Bound

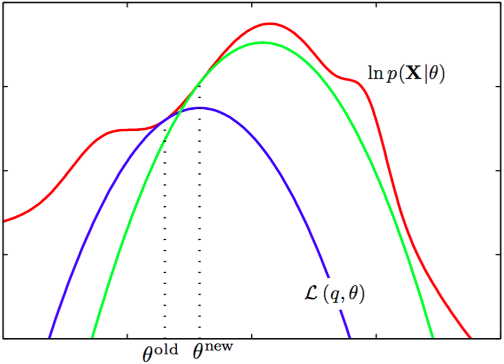
- In EM algorithm, we maximize the lower bound $\mathcal{L}(q,\theta)$:

$$\log p(x \mid \theta) \geqslant \mathcal{L}(q,\theta).$$

- EM Algorithm (high level):
  1. Choose initial θ^{old}.
  2. Let $q^* = \arg\max_q \mathcal{L}(q,\theta^{\text{old}})$
  3. Let $\theta^{\text{new}} = \arg\max_\theta \mathcal{L}(q^*,\theta)$.
  4. Go to step 2, until converged.
- Will show: $p(x \mid \theta^{\textbf{new}}) \geqslant p(x \mid \theta^{\textbf{old}})$
- **Get sequence of θ's with monotonically increasing likelihood.**

EM: Coordinate Ascent on Lower Bound

1. Start at θ^{old}. Find best lower bound at θ^{old}: $\mathcal{L}(q,\theta)$.
2. $\theta^{\text{new}} = \arg\max_{\theta} \mathcal{L}(q,\theta)$.



From Bishop's *Pattern recognition and machine learning*, Figure 9.14.

The Lower Bound

- Let's investigate the lower bound:

$$\begin{aligned}
\mathcal{L}(q,\theta) &= \sum_z q(z)\log\left(\frac{p(x,z\mid\theta)}{q(z)}\right) \\
&= \sum_z q(z)\log\left(\frac{p(z\mid x,\theta)p(x\mid\theta)}{q(z)}\right) \\
&= \sum_z q(z)\log\left(\frac{p(z\mid x,\theta)}{q(z)}\right)+\sum_z q(z)\log p(x\mid\theta) \\
&= -\mathrm{KL}[q(z),p(z\mid x,\theta)]+\log p(x\mid\theta)
\end{aligned}$$

- Amazing! We get back an equality for the marginal likelihood:

$$\log p(x\mid\theta)=\mathcal{L}(q,\theta)+\mathrm{KL}[q(z),p(z\mid x,\theta)]$$

The Best Lower Bound

- Find q maximizing

$$\mathcal{L}(q, \theta^{\text{old}}) = -\mathrm{KL}[q(z), p(z \mid x, \theta^{\text{old}})] + \underbrace{\log p(x \mid \theta^{\text{old}})}_{\text{no } q \text{ here}}?$$

- Recall $\mathrm{KL}(p \| q) \geqslant 0$, and $\mathrm{KL}(p \| p) = 0$.
- Best q is $q^*(z) = p(z \mid x, \theta^{\text{old}})$:

$$\mathcal{L}(q^*, \theta^{\text{old}}) = -\underbrace{\mathrm{KL}[p(z \mid x, \theta^{\text{old}}), p(z \mid x, \theta^{\text{old}})]}_{=0} + \log p(x \mid \theta^{\text{old}})$$

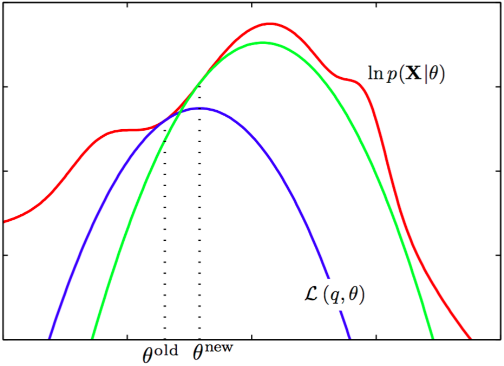
- Summary:

$$\begin{aligned} \log p(x \mid \theta^{\text{old}}) &= \mathcal{L}(q^*, \theta^{\text{old}}) \quad (\text{tangent at } \theta^{\text{old}}). \\ \log p(x \mid \theta) &\geqslant \mathcal{L}(q^*, \theta) \quad \forall \theta \end{aligned}$$

General EM Algorithm

1. Choose initial θ^{old}.
2. **Expectation Step**
   - Let $q^*(z) = p(z \mid x, \theta^{\text{old}})$.
   - Let
$$J(\theta) = \mathcal{L}(q^*, \theta) = \sum_z q^*(z) \log\left(\frac{p(x, z \mid \theta)}{q^*(z)}\right)$$
   - Note that $J(\theta)$ is an **expectation** w.r.t. $Z \sim q^*(z)$.
3. **Maximization Step**
$$\theta^{\text{new}} = \underset{\theta}{\arg\max}\, J(\theta).$$
4. Go to step 2, until converged.

EM Gives Monotonically Increasing Likelihood: By Picture



From Bishop's *Pattern recognition and machine learning*, Figure 9.14.

EM Gives Monotonically Increasing Likelihood: By Math

1. Start at θ^{old}.
2. Choose $q^*(z) = \arg\max_q \mathcal{L}(q, \theta^{\text{old}})$. We've shown

$$\log p(x \mid \theta^{\text{old}}) = \mathcal{L}(q^*, \theta^{\text{old}})$$

3. Choose $\theta^{\text{new}} = \arg\max_\theta \mathcal{L}(q^*, \theta^{\text{old}})$. So

$$\mathcal{L}(q^*, \theta^{\text{new}}) \quad \geqslant \quad \mathcal{L}(q^*, \theta^{\text{old}}).$$

Putting it together, we get

$$\begin{array}{rcll} \log p(x \mid \theta^{\text{new}}) & \geqslant & \mathcal{L}(q^*, \theta^{\text{new}}) & \mathcal{L} \text{ is a lower bound} \\ & \geqslant & \mathcal{L}(q^*, \theta^{\text{old}}) & \text{By definition of } \theta^{\text{new}} \\ & = & \log p(x \mid \theta^{\text{old}}) & \text{Bound is tight at } \theta^{\text{old}}. \end{array}$$

From Bishop's *Pattern recognition and machine learning*, Figure 9.14.

EM Gives Monotonically Increasing Likelihood: And so?

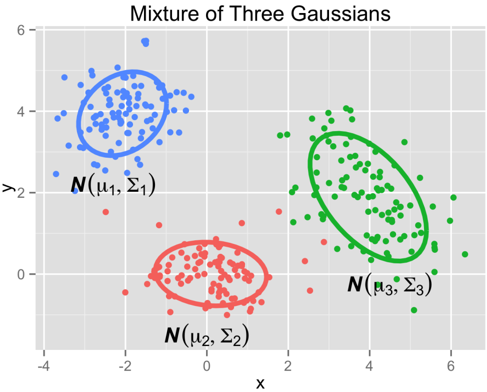
- Let θ_n be value of EM algorithm after n steps.
- Are there conditions for which
  - θ_n converges to the maximum likelihood?
  - θ_n converges to a local maximum?
  - θ_n converges to a stationary point of likelihood?
  - θ_n converges?
- There are conditions for each of these (to happen and not to happen).
- See "On the Convergence Properties of the EM Algorithm" by C. F. Jeff Wu, *The Annals of Statistics*, Mar. 1983.
  - `http://web.stanford.edu/class/ee378b/papers/wu-em.pdf`
- In practice, can run EM multiple times with random starts.

Homework: Derive EM for GMM from General EM Algorithm

- Subsequent slides may help set things up.
- Key skills:
  - MLE for multivariate Gaussian distributions.
  - Lagrange multipliers

Gaussian Mixture Model ($k = 3$)

1. Choose $Z \in \{1, 2, 3\} \sim \text{Multi}\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$.
2. Choose $X \mid Z = z \sim \mathcal{N}(X \mid \mu_z, \Sigma_z)$.



Gaussian Mixture Model (k Components)

- GMM Parameters

$$\begin{aligned} \text{Cluster probabilities}: &\quad \pi = (\pi_1, \ldots, \pi_k) \\ \text{Cluster means}: &\quad \mu = (\mu_1, \ldots, \mu_k) \\ \text{Cluster covariance matrices}: &\quad \Sigma = (\Sigma_1, \ldots \Sigma_k) \end{aligned}$$

- Let $\theta = (\pi, \mu, \Sigma)$.
- Marginal log-likelihood

$$\log p(x \mid \theta) = \log \left\{ \sum_{z=1}^{k} \pi_z \mathcal{N}(x \mid \mu_z, \Sigma_z) \right\}$$

$q^*(z)$ = Soft Assignments

- At each step, we take

$$q^*(z) = p(z \mid x, \theta^{\text{old}})$$

.

- This corresponds to "soft assignments" we had last time:

$$\begin{aligned}\gamma_i^j &= \mathbb{P}(Z = j \mid X = x_i) \\ &= \frac{\pi_j \mathcal{N}(x_i \mid \mu_j, \Sigma_j)}{\sum_{c=1}^k \pi_c \mathcal{N}(x_i \mid \mu_c, \Sigma_c)}\end{aligned}$$

Expectation Step

- The complete log-likelihood is

$$\begin{aligned}\log p(x,z\mid\theta) &= \sum_{i=1}^{n}\log\left[\pi_z \mathcal{N}(x_i\mid\mu_z,\Sigma_z)\right]\\ &= \sum_{i=1}^{n}\left(\log\pi_z+\underbrace{\log\mathcal{N}(x_i\mid\mu_z,\Sigma_z)}_{\text{simplifies nicely}}\right)\end{aligned}$$

- Take the expected complete log-likelihood w.r.t. q^*:

$$\begin{aligned}J(\theta) &= \sum_{z} q^*(z)\log p(x,z\mid\theta)\\ &= \sum_{i=1}^{n}\sum_{j=1}^{k}\gamma_i^j\left[\log\pi_j+\log\mathcal{N}(x_i\mid\mu_j,\Sigma_j)\right]\end{aligned}$$

Maximization Step

- Find θ^* maximizing $J(\theta)$. Result is what we had last time:

$$\begin{aligned}
\mu_c^{\text{new}} &= \frac{1}{n_c}\sum_{i=1}^{n}\gamma_i^c x_i \\
\Sigma_c^{\text{new}} &= \frac{1}{n_c}\sum_{i=1}^{n}\gamma_i^c \left(x_i-\mu_{\text{MLE}}\right)\left(x_i-\mu_{\text{MLE}}\right)^T \\
\pi_c^{\text{new}} &= \frac{n_c}{n},
\end{aligned}$$

for each $c = 1, \ldots, k$.

Machine Learning

1. Look at other course notes at this level.
    - Every course covers different subset of topics.
    - Different perspectives. (e.g. Bayesian / Probabilistic)
2. Read on some "second semester" topics
    - LDA / Topic Models (DS-GA 1005?)
    - Sequence models: Hidden Markov Models / MEMMs / CRFs (DS-GA 1005)
    - Bayesian methods
    - Collaborative Filtering / Recommendations
    - Ranking
    - Bandit problems (Thompson sampling / UCB methods)
    - Gaussian processes

Other Stuff To Learn

- Statistics
- Data Structures & Algorithms (Theoretical)
- Some production programming language (e.g. Java, C++)