Notes on Expected Maximization

Dan Beatty

April 26, 2007

Big deal: extending the application of maximum-likelihood techniques to permit the "learning of parameters governing a distribution from training points."

- Uncorrupted cases could use $\hat{\vec{\theta}}$ acquired from MLE.
- Iteratively converge on the likelihood for a given data set via Expectation Maximization or Baum-Welch
- Features can be in terms of good features and bad features: $D = \{\vec{x_1}, \ldots, \vec{x_n}\}$ or $D = D_g \cup D_b$.

Thus one function to contemplate comes to the front:

$$Q(\vec{\theta}; \vec{\theta^i}) = E_{D_b}[\ln p(D_g, D_b; \vec{\theta}) | D_g; \vec{\theta^i}] \tag{1}$$

- $Q(\vec{\theta}; \theta^i)$ is a function of $\vec{\theta}$ and $\vec{\theta^i}$
- $E_{D_b}[\ln p(D_g, D_b; \vec{\theta}) | D_g; \vec{\theta^i}]$ is the expected value is over the missing features. The expected value hinges on $\vec{\theta^i}$ are the true parameters.
- $\vec{\theta^i}$ is the current (best) estimate for the full distribution;
- $\vec{\theta}$ is a candidate vector for an improved estimate
- D_b gets marginalized with respect to $\vec{\theta^i}$.
- The goal of the EM algorithm is select from the candidate $\vec{\theta}$ from a set of $\vec{\theta}$s, and iterate it to $\vec{\theta^{i+1}}$ which yields the greatest $Q(\vec{\theta}; \vec{\theta^i})$
- Samples are assumed iid.

Claims about algorithm **??**:

- Useful when the optimization of $Q(\cdot; \cdot)$ is simpler than computing $l(\cdot)$
- To marginalize bad data, and increase marginalization monotonically.

Algorithm 1 Expectation Maximization

```
initialize $\vec{\theta}^0$, $T$, and $i \leftarrow 0$
repeat
    $i \leftarrow i + 1$
    E Step: compute $Q(\vec{\theta}; \vec{\theta}^i)$
    M step: $\theta^{i+1} \rightarrow \arg\max_\theta Q(\vec{\theta}; \vec{\theta}^i)$
until $Q(\vec{\theta}^{i+1}; \vec{\theta}^i) - Q(\vec{\theta}^i; \vec{\theta}^{i-1}) \leq T$
return $\hat{\vec{\theta}} \rightarrow \vec{\theta}^{i+1}$
```

1 Information Theory Involved

Basic information theory was viewed earlier in the course. Entropy in context of a discrete distribution is "a measure of the randomness or unpredictability of a sequence of symbols drawn" [**?**, 630] from such a distribution. The units of entropy depend on the number system used, but otherwise is a unit-less value. Entropy depends on the probabilities of the discrete items in the distribution, and not on the items themselves.

$$H = -\sum_{i=1}^{m} P_i \log_2 P_i = E[\log \frac{1}{P}] \tag{2}$$

The relative entropy also known as Kullback-Leibler distance is a measure between two probabilities over the same variable.

$$D_{KL}(p(x), q(x)) = \sum_x q(x) \ln \frac{q(x)}{p(x)} \tag{3}$$

$$D_{KL}(p(x), q(x)) = \int_{-\infty}^{\infty} q(x) \ln \frac{q(x)}{p(x)} \tag{4}$$

If there are two distributions, then there is a possibility of the distributions have information in common. A few exceptions arise due to the mutual information. Mutual information is the reduction of uncertainty about one variable due to information about another.

$$I(p; q) = H(p) - H(p|q) = \sum_{x,y} r(x, y) \log_2 \frac{r(x, y)}{p(x)q(y)} \tag{5}$$

where

- $r(x, y)$ is the joint distribution of finding x, y.
- $p(x)$ and $q(y)$ are the probabilities of x and y in their respective distributions.
- Exceptions to metric rules includes that

  $$p(x) = q(y) \rightarrow I(x; y) = 0 \tag{6}$$

  is not guaranteed.

note moon book
A.P. Book

2 MLE and EM

Expectation and maximization: An EM algorithm finds maximum likelihood (ML) estimates of parameters in probabilistic variables that are not directly observed but inferred from observed and directly measured variables (latent variables). EM alternates between an expectant step (that computes the expectation of the likelihood by including latent variables) and a maximization step (M-step that maximizes the expected likelihood found on the E step). The process is repeated by performing another E step using the parameters found on the M-step. EM is frequently used in data clustering in “Computer Vision” and “Machine Learning”.

2.1 Specification of the EM procedure

Let $Y \rightarrow$ incomplete consisting of values of observable variables. $X \rightarrow$ the missing data X and Y together form the incomplete data.

2.2 Step 1: Estimate unobservable data

Let $p \rightarrow$ the joint probability distribution of the complete data with parameters given by the vector $\theta \rightarrow p(y, x|\theta) \rightarrow$ likelihood of the complete data. Then the conditional distribution of the missing data is given by

$$p(x|y, \theta) = \frac{p(y|x, \theta)}{p(y|\theta)} = \frac{p(y|x, \theta)p(x|\theta)}{\int p(y|\hat{x}, \theta)p(\hat{x}|\theta)d\hat{x}} \tag{7}$$

using Bayes rule and total probability. The above formulation only the needs the knowledge of $p(y|x, \theta)$, the likelihood of the observation given the unobservable data, and the probability of the unobservable data, $p(x|\theta)$.

2.3 Step 2: Maximize log-likelihood of the complete data set

In this step, the EM algorithm improves on an initial estimate θ_0 by new estimates $\theta_1, ..., \theta_n$, iteratively. An individual re-estimation step has the following form:

$$\theta_{n+1} \arg\max_{\theta} E_x[\log p(y, x|\theta)|y] \tag{8}$$

where $E_x[\cdot]$ denotes the conditional expectation of $\log p(y, x|\theta)$ with θ in the conditional distribution of x fixed at θ_n. $\therefore \theta_{n+1}$ is the value that maximizes (M) the expectation (E) of the complete data likelihood given the observed variables.

$$p(x|\theta) = p(x, y|\theta) = p(y|x, \theta)p(x|\theta) \tag{9}$$

which is the joint density function. Now we define a new likelihood function

$$L(\theta|Z) = L(\theta|X, Y) = p(X, Y, \theta) \tag{10}$$

This leads to the complete data likelihood. The EM algorithm finds the expected value of the complete data log-likelihood i.e., $\log p(x, y|\theta)$ with, respect to the unknown data Y given the observed data X and the current estimates of the parameters. $\therefore$ we define $Q(\theta, \theta^{i-1})$, where θ^{i-1}, where θ^{i-1} are the current parameter estimates used to evaluate the expectation and θ are the parameters that are optimized to increase Q.

$$\therefore Q\{\theta, \theta^{i-1}\} = E[\log p(x, y|\theta)|X, \theta^{i-1}] \tag{11}$$

Here x, θ^{i-1} are constants, θ is a variable to be adjusted, and Y is a random variable ruled by a distribution $f(y|x, \theta^{i-1})$:

$$\therefore Q\{\theta, \theta^{i-1}\} = \infty \tag{12}$$

$$E[\log p(x, y|\theta)|X, \theta^{i-1}] = \infty \tag{13}$$

In the M step of the EM algorithm

$$\theta^{(i)} = \arg\max_{\theta} Q(\theta, \theta^{i-1}) \tag{14}$$

is found by maximizing the expectation computing in the E step.

3 Lessons from Moon

E Step Compute $Q(\vec{\theta}|\vec{\theta}^{(k)})$

$$Q(\vec{\theta}|\vec{\theta}^{(k)}) = E[\log f(\vec{x}|\vec{y}, \vec{\theta}^{k})] \tag{15}$$

Condition the likelihood of the complete data.

- Fixed
- Conditions the expectation function

M Step: Let $\vec{\theta}^{(k+1)}$ be that value of $\vec{\theta}$ which versions $Q(\vec{\theta}|\vec{\theta}^{k})$.

$$\vec{\theta}^{(k+1)} = \arg\max_{\theta} Q(\vec{\theta}|\vec{\theta}^{k})$$

Maximization is about the conditioner of the complete data.

Exponential Family

Definition A family of distributions with probabilities mass function of density $f_x(\vec{x}|\vec{\theta})$ is said to be a k parameter exponential family if $f_x(x|\theta)$ has the form

$$f_x(x|\theta) = c(\theta)a(x)\exp[\sum_{i=1}^{k} \pi_i(\theta)t_i(x)]$$

$$c(\theta) = \frac{1}{\sum_x a(x)\exp[\sum_{i=1}^{k} \pi_i(\theta)t_i(x)]}$$

Applied to EM:

- A vector of sufficient statistics

$$\vec{t}(\vec{x}) = [t_1(\vec{x}), ..., t_q\vec{x}]^T$$

- $\vec{\theta}$ is a vector of parameters for the family.

E-Step Rewritten:

$$Q(\vec{\theta}|\vec{\theta}^k) = E[\log a(x)|\vec{y},\vec{\theta}^k] + \pi(\vec{\theta})^T E[\vec{t}(\vec{x})|\vec{y},\vec{\theta}^k] + logc(\vec{\theta}) \tag{16}$$

$$\vec{t}^{[k+1]} = E[\log a(\vec{x})|\vec{y},\vec{\theta}^{[k]}] \tag{17}$$

M-Step Rewritten:

$$E[\log(a(\vec{x}))|\vec{y},\vec{\theta}^{[k]}] + \pi(\vec{\theta})\vec{t}^{[k+1]} + \log c(\vec{\theta}) \tag{18}$$

Gaussian application to derivation

$$p(\vec{y}) = \sum_{i=1}^{M} \alpha_i p_i(\vec{y}_j|\vec{\mu}_i,\mathbf{\Sigma}_i) \tag{19}$$

$$\vec{\theta} = \{\alpha_1,...,\alpha_M,\vec{\mu}_1,...,\vec{\mu}_M,\mathbf{\Sigma}_1,..,\mathbf{\Sigma}\} \tag{20}$$

E - Step

$$a_{ij}^p = \frac{\alpha_j^p p(\vec{y}_i^{(p)}|\vec{\mu}_j^{(p)}\mathbf{\Sigma}_j^{(p)})}{\sum_{j=1}^M \alpha_j^p p(\vec{y}_i^{(p)}|\vec{\mu}_j^{(p)}\mathbf{\Sigma}_j^{(p)})} \tag{21}$$

M-Step

$$\vec{\mu}_j^{(p+1)} = \frac{\sum_{i=1}^N a_{ij}^p \vec{y}_i}{\sum_{i=1}^N a_{ij}^{(p)}} \tag{22}$$

$$\mathbf{\Sigma}_j^{(p+1)} = \frac{\sum_{i=1}^N a_{ij}(\vec{y}i - \vec{\mu}_j)(\vec{y}i - \vec{\mu}_j)^T}{\sum_{i=1}^N a_{ij}^{(p)}} \tag{23}$$

$$\alpha_j^{(p+1)} = |a_j| \tag{24}$$

- $\vec{a}_j$ is a column vector from matrix $\mathbf{A}$
- $\vec{y}_j$ is a sample vector
- N is the number of samples
- M is the number of attributes known and unknown.