The Expectation Maximization Algorithm

Gaurav Sharma

University of Rochester

The Expectation Maximization (EM) Algorithm

- An iterative algorithm for Maximum Likelihood (ML) parameter estimation
- Formulate Maximum likelihood problems using incomplete-data framework
  - Either in presence of missing data
  - Or when the model can be simplified by introducing latent variable
  - EM name given by Dempster, who gave the general formulation of the algorithm
  - Wide ranging applications in machine learning, signal processing, statistics, data mining, and many other fields
    - Learning parameters of finite mixture models (say, mixture of Gaussians)
    - Model estimation for hidden Markov models (Baum-Welch)

EM: Toy Example, Mixing Two Coins

- Random experiment: iid "mixing" of two coins
  - indexed by $j = 1, 2$, Coin j chosen with probability α_j
  - Coin characteristics: $p_j = \Pr(H \mid Coin = j) = \Pr(1 \mid Coin = j)$
  - Parameters $\boldsymbol{\theta} = [\alpha_1, \alpha_2, p_1, p_2]$, Constraint $\sum_j \alpha_j = 1$
- Observations: Series of outcomes of mixing experiment $\mathrm{x} = [x_1, x_2, \ldots x_N]$
- Want ML estimate of parameters

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p\left(\mathrm{x} \mid \boldsymbol{\theta}\right)$$

- What is $p\left(\mathrm{x} \mid \boldsymbol{\theta}\right)$?
  - For example, for a specific string $\mathrm{x} = 0010110001$?
- Recall ML estimation for Bernoulli $\mathcal{B}$ random variable
  - Expression for $p\left(\mathrm{x} \mid \boldsymbol{\theta}\right)$ was straightforward
  - Why can't we do the same here?

EM: Toy Example, Mixing Two Coins

- In addition to observation x, need information on which coin used for each outcome to write expression for likelihood
  - EM Approach: Complete data by introducing latent variables
    - Latent random variable Z^i as a 2×1 vector for the i^{th} outcome indicating which coin was used for the i^{th} toss
    - $Z^i_j = 1$ if j^{th} coin produced the i^{th} outcome
    - Z^i's iid vectors having one entry as 1 others 0 ($\equiv$ Bernoulli RV)
  - Complete likelihood is simple

$$p\left(\mathrm{x}, \mathrm{z} \mid \boldsymbol{\theta}\right) = \prod_{i=1}^{N} p\left(x_i, z^i \mid \boldsymbol{\theta}\right) = \prod_{i=1}^{N} \sum_j z^i_j \alpha_j \mathcal{B}(x_i, p_j)$$

$$\mathcal{B}(x, p) = p^x (1-p)^{1-x}$$

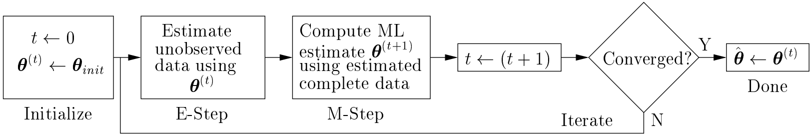
- How could you use the complete likelihood?

EM Intuition for Coin Mixing Problem

- Don't know z so also estimate it and iteratively update along with parameter estimates
- An approach:
  - Say you have initial estimate $\hat{\boldsymbol{\theta}}^0$ of the parameters $\boldsymbol{\theta}$
  - Initialize iteration count $t \leftarrow 0$
  - Estimate: Use current estimate of parameters $\hat{\boldsymbol{\theta}}^t$ to classify outcomes to obtain an estimate $\hat{\mathrm{z}}$ of z
  - Maximize: Update parameters $\hat{\boldsymbol{\theta}}^{t+1}$ to maximize $p(\mathrm{x}, \hat{\mathrm{z}} \mid \boldsymbol{\theta})$
  - Iterate till convergence
- EM is a refinement of this approach based on the same intuition
- Question: How well will this work for the coin mixing problem?
  - Are there parameters for which the approach fails?

The Expectation Maximization (EM) Algorithm (Precursor)

- Iterate between
  - E-Step: Estimating "unobserved" data
  - M-Step: Maximum likelihood estimation of $\boldsymbol{\theta}$ from "completed" data



EM: Complete vs Incomplete Likelihood

- If we cannot even evaluate $p(\mathrm{x} \mid \boldsymbol{\theta})$ readily, ML estimation seems to be hard
- EM: Complete the data and iterate between "estimating the missing data" and maximizing the complete likelihood with the "estimated missing data"
  - Coin example
- Data is often "incomplete" in many practical situations
  - Missing data, partial observations, indirect observation
- EM addresses this class of problems
  - Provides an indirect approach for ML estimation

The Expectation Maximization (EM) Algorithm

- ML estimates of parameters

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p\left(\mathsf{x} \mid \boldsymbol{\theta}\right)$$

- The EM algorithm addresses scenarios where it is hard to evaluate/formulate

$$p\left(\mathsf{x} \mid \boldsymbol{\theta}\right)$$

but we can complete the observations to obtain y such that $p\left(\mathsf{y} \mid \boldsymbol{\theta}\right)$ is easy to evaluate and

$$\mathsf{x} = f\left(\mathsf{y}\right)$$

for some (typically many-to-one) mapping $f()$.

Likelihood for Incomplete Data from Complete Data

- Likelihood for observed "incomplete" data from "complete" data

$$p\left(\mathrm{x} \mid \boldsymbol{\theta}\right) = \int_{\mathrm{y}:f(\mathrm{y})=\mathrm{x}} p\left(\mathrm{y} \mid \boldsymbol{\theta}\right) dy$$

- Log likelihood

$$l_{\mathrm{x}}\left(\boldsymbol{\theta}\right) = \ln\left(p\left(\mathrm{x} \mid \boldsymbol{\theta}\right)\right)$$

EM Algorithm Intuition

- Idea: If complete data was available, would like to maximize $p\left(\mathrm{y} \mid \boldsymbol{\theta}\right) \equiv$ maximize $\ln\left(p\left(\mathrm{y} \mid \boldsymbol{\theta}\right)\right)$
- Since y is unavailable, maximize expectation of $\ln\left(p\left(\mathrm{y} \mid \boldsymbol{\theta}\right)\right)$ given data x and current estimate of parameters $\boldsymbol{\theta}^{(t)}$
- Two step procedure
  - E-Step: Compute the expectation

$$\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t) = E\left[\ln\left(p\left(\mathrm{y} \mid \boldsymbol{\theta}\right)\right) \mid \mathrm{x}, \boldsymbol{\theta}^t\right]$$

- M-Step: Update the parameters to maximize the expectation

$$\boldsymbol{\theta}^{t+1} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t)$$

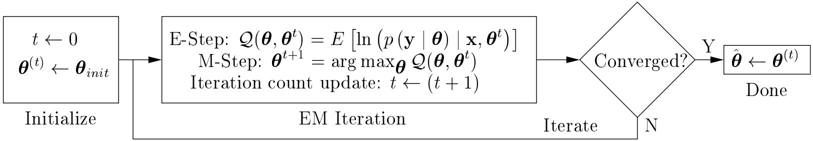
The Expectation Maximization (EM) Algorithm: Formal Statement

- Observed data $\mathbf{x}$, Full data $\mathbf{y}$
  - E-Step: Compute the expectation

$$\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t) = E\left[\ln\left(p\left(\mathrm{y} \mid \boldsymbol{\theta}\right)\right) \mid \mathrm{x}, \boldsymbol{\theta}^t\right]$$

- M-Step: Update parameter estimate to maximize the expectation

$$\boldsymbol{\theta}^{t+1} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t)$$



Formal EM algorithm for two coin mixing problem

- Recall, complete likelihood

$$p\left(\mathrm{x}, \mathrm{z} \mid \boldsymbol{\theta}\right) = \prod_{i=1}^{N} p\left(x_i, z^i \mid \boldsymbol{\theta}\right) = \prod_{i=1}^{N} \sum_j z_j^i \alpha_j \mathcal{B}(x_i, p_j)$$

- E Step:

$$\begin{aligned}
\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t) &= E\left[\ln\left(p\left(\mathrm{y} \mid \boldsymbol{\theta}\right)\right) \mid \mathrm{x}, \boldsymbol{\theta}^t\right] \\
&= \sum_{i=1}^{N} E\left[\ln\left(\sum_j z_j^i \alpha_j \mathcal{B}(x_i, p_j)\right) \mid \mathrm{x}^{(i)}, \theta^{(t)}\right] \\
&= \sum_{i=1}^{N} \sum_j \gamma_j^i \ln\left(\alpha_j \mathcal{B}(x_i, p_j)\right) \quad \text{Why?} \\
\gamma_j^i &= E\left[z_j^i \mid \mathrm{x}, \boldsymbol{\theta}^t\right] = Pr\left[z_j^{(i)} = 1 \mid \mathrm{x}^{(i)}, \theta^{(t)}\right]
\end{aligned}$$

γ_j^i is posterior prob. that i^{th} outcome came from j^{th} coin $(\mid \mathrm{x}, \boldsymbol{\theta}^t)$

Formal EM algorithm for two coin mixing problem

- E Step: Equivalent to estimating γ_j^i
  - From Bayes rule:

$$\gamma_j^i = Pr\left[z_j^{(i)} = 1 | \mathrm{x}^{(i)}, \theta^{(t)}\right] = \frac{Pr\left[\mathrm{x}^{(i)}, z_j^{(i)} = 1 | \theta^{(t)}\right]}{\sum_{z_i} Pr\left[\mathrm{x}^{(i)}, z_i | \theta^{(t)}\right]}$$

$$= \frac{\alpha_j^{(t)} \mathcal{B}(x_i, p_j^{(t)})}{\sum_{l=1}^{2} \alpha_l^{(t)} \mathcal{B}(x_i, p_l^{(t)})}$$

- Soft as opposed to "hard" categorization of outcomes to coins

$$\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t) = \sum_{i=1}^{N} \sum_{j} \gamma_j^i \ln\left(\alpha_j \mathcal{B}(x_i, p_j)\right)$$

$$= \sum_{i=1}^{N} \sum_{j} \gamma_j^i \left(\ln \alpha_j + x_i \ln p_j + (1 - x_i) \ln(1 - p_j)\right)$$

Formal EM algorithm for two coin mixing problem

- M Step: Maximize

$$\boldsymbol{\theta}^{t+1} \stackrel{\text{def}}{=} [\alpha_1^{t+1}, \alpha_2^{t+1}, p_1^{t+1}, p_2^{t+1}] = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t)$$

$$= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{N} \sum_{j} \gamma_j^i \left(\ln \alpha_j + x_i \ln p_j + (1 - x_i) \ln(1 - p_j) \right)$$

- Calculus of variations (constraints important)

$$\frac{\nabla \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t)}{\nabla \alpha_1} = 0 \equiv \sum_{i=1}^{N} \left(\gamma_1^i \frac{1}{\alpha_1} - (1 - \gamma_1^i) \frac{1}{1 - \alpha_1} \right) = 0$$

$$\alpha_j^{t+1} = \frac{1}{N} \sum_{i=1}^{N} \gamma_j^i \qquad\qquad p_j^{t+1} = \frac{\sum_{i=1}^{N} \gamma_j^i x_i}{\sum_{i=1}^{N} \gamma_j^i}$$

Formal EM algorithm for two coin mixing problem

- E Step: Estimate posterior probabilities γ_j^i

$$\begin{aligned}
\gamma_j^i &= \frac{\alpha_j^{(t)}\mathcal{B}(x_i, p_j^{(t)})}{\sum_{l=1}^2 \alpha_l^{(t)}\mathcal{B}(x_i, p_l^{(t)})} \\
&= \frac{\alpha_j^{(t)}\left(p_j^{(t)}\right)^{x_i}\left(1 - p_j^{(t)}\right)^{1-x_i}}{\sum_{l=1}^2 \alpha_l^{(t)}\left(p_l^{(t)}\right)^{x_i}\left(1 - p_l^{(t)}\right)^{1-x_i}}
\end{aligned}$$

- M Step:

$$\alpha_j^{t+1} = \frac{1}{N}\sum_{i=1}^N \gamma_j^i$$

$$p_j^{t+1} = \frac{\sum_{i=1}^N \gamma_j^i x_i}{\sum_{i=1}^N \gamma_j^i}$$

EM algorithm for two coin mixing problem

- Final EM algorithm relatively simple and intuitive
  - E Step: Compute posterior probability that outcome i came from coin j
  - M Step: Update prob to posterior probability weighted mean fraction of heads in outcomes weight of i^{th} outcome is posterior probability that outcome i came from coin j
- Contrast with abstraction
  - E-Step: Compute the expectation

$$\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t) = E\left[\ln\left(p\left(\mathrm{y} \mid \boldsymbol{\theta}\right)\right) \mid \mathrm{x}, \boldsymbol{\theta}^t\right]$$

- M-Step: Update the parameters to maximize the expectation

$$\boldsymbol{\theta}^{t+1} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t)$$

- Simple intuitive structure of EM algorithm extends to many problems

Actual EM Application Examples

- WAMI to roadmap alignment
  - Formulating maximum likelihood directly was challenging
  - Different behavior of on-road vs spurious vehicles
  - Introduction of latent variable simplified things
    - Complete likelihood involving latent variables
    - Actual likelihood = marginal of complete likelihood over latent variables
- Color Barcodes
  - Similar story
  - Additional approximation/constraints from Physics
- Common theme in applications of EM
  - Incomplete Data/Latent Variable Problems

EM Algorithm for Channel-wise Color Barcodes

- Recall model relation

$$\mathrm{D}_{3\times 3}\mathrm{I}_{3\times N} \approx \mathrm{d}_{3\times N}$$

where $\mathrm{I}_{3\times N}$ is indicator variable indicating printing in $\{R, G, B\}$ channels for the N -pixels, $\mathrm{D}_{3\times 3}$ is the unknown channel cross-interference matrix, $\mathrm{d}_{3\times N}$ are the observed densities corresponding to the pixels.

EM Algorithm for Channel-wise Color Barcodes

- Probabilistic model formulation
  - Model noise in each pixel as iid zero mean Gaussian random variable

$$\mathrm{d}_{3\times N} = \mathrm{D}_{3\times 3}\mathrm{I}_{3\times N} + \boldsymbol{\eta}_{3\times N}$$

where $\boldsymbol{\eta}_{3\times N}$ is the noise in the N pixels.

- Incomplete data $\mathrm{x} \equiv \mathrm{d}$, Complete data $\mathrm{y} \equiv (\mathrm{d}, \mathrm{I})$, Parameters $\boldsymbol{\theta} \equiv \mathrm{D}$
- Complete data log likelihood

$$l_{\mathrm{d}}(\mathrm{D}) = -C\left\|\mathrm{d} - \mathrm{DI}\right\|^2$$

where C is a positive constant independent of D.

EM Algorithm for Channel-wise Color Barcodes

- E-Step:

$$\mathcal{Q}(\mathrm{D}, \mathrm{D}^{(t)}) = E\left[l_{\mathrm{d}}(\mathrm{D}) \mid \mathrm{d}, \mathrm{D}^{(t)}\right] = -C\left\|\mathrm{d} - \mathrm{D}E\left[\mathrm{I} \mid \mathrm{d}, \mathrm{D}^{(t)}\right]\right\|^2$$
$$= -C\left\|\mathrm{d} - \mathrm{D}\tilde{\mathrm{I}}\right\|^2$$
$$\tilde{\mathrm{I}} = E\left[\mathrm{I} \mid \mathrm{d}, \mathrm{D}^{(t)}\right] = \left(\mathrm{D}^{(t)}\right)^{-1}\mathrm{d} \tag{1}$$

used fact that noise is assumed to be zero mean

- M-Step:

$$\mathrm{D}^{(t+1)} = \arg\max_{\mathrm{D}} \mathcal{Q}(\mathrm{D}, \mathrm{D}^{(t)})$$
$$= \arg\min_{\mathrm{D}} \left\|\mathrm{d} - \mathrm{D}\tilde{\mathrm{I}}\right\|^2 \tag{2}$$

- Note: imposing non-negativity of D and $\tilde{\mathrm{I}}$ helps convergence to correct local minima

EM Algorithm: For Exponential Family PDFs

- Exponential family complete likelihood

$$p(\mathrm{y} \mid \boldsymbol{\theta}) = \frac{1}{a(\boldsymbol{\theta})} b(\mathrm{y}) \exp\left(\mathrm{c}^T(\boldsymbol{\theta}) \mathrm{s}(\mathrm{y})\right)$$

- $a(\boldsymbol{\theta})$ and $\mathrm{c}(\boldsymbol{\theta})$ are vectors that are functions of the parameters
- $\mathrm{s}(\mathrm{y}) =$ sufficient statistic
- Exponential family of distributions includes many distributions of common interest
  - Gaussian, Poisson, binomial, uniform, Rayleigh, etc

Example: Multivariate Gaussian as an Exponential Family PDF

- Recall: Multivariate Gaussian, Parameters $\boldsymbol{\theta} = (\boldsymbol{\mu}, \Sigma)$

$$p\left(\mathrm{y} \mid \boldsymbol{\theta}\right) = \frac{1}{\sqrt{(2\pi)^d \det(\Sigma)}} \exp\left(-\frac{(\mathrm{y}-\mu)^T \Sigma^{-1} (\mathrm{y}-\mu)}{2}\right)$$

- Multivariate Gaussian as an exponential family PDF
  - Observe that the exponent is a polynomial with terms involving y_i, $y_i y_j$, and constants
  - Can be expressed in the form

$$\begin{aligned} p\left(\mathrm{y} \mid \boldsymbol{\theta}\right) &= \frac{1}{\sqrt{(2\pi)^d \det(\Sigma)}} \exp\left(\sum_i \alpha_i y_i + \sum_{i,j} \beta_{ij} y_i y_j + \gamma\right) \\ &= \frac{1}{\mathrm{a}(\boldsymbol{\theta})} \exp\left(\mathrm{c}^T(\boldsymbol{\theta}) \mathrm{s}\left(\mathrm{y}\right)\right) \end{aligned}$$

EM Algorithm: For Exponential Family PDFs

- Expectation (E) step for exponential family

$$
\begin{aligned}
\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t) &= E\left[\ln\left(p\left(\mathrm{y} \mid \boldsymbol{\theta}\right)\right) \mid \mathrm{x}, \boldsymbol{\theta}^t\right] \\
&= E\left[\ln b(\mathrm{y}) + \left(\mathrm{c}^T(\boldsymbol{\theta})\mathrm{s}\left(\mathrm{y}\right)\right) - \ln a\left(\boldsymbol{\theta}\right) \mid \mathrm{x}, \boldsymbol{\theta}^t\right] \\
&= E\left[\ln b(\mathrm{y}) \mid \mathrm{x}, \boldsymbol{\theta}^t\right] + \mathrm{c}^T(\boldsymbol{\theta}) E\left[\mathrm{s}\left(\mathrm{y}\right) \mid \mathrm{x}, \boldsymbol{\theta}^t\right] - \ln a\left(\boldsymbol{\theta}\right)
\end{aligned}
$$

- Note $E\left[\mathrm{s}\left(\mathrm{y}\right) \mid \mathrm{x}, \boldsymbol{\theta}^t\right]$ is a conditional estimate of the sufficient statistic given the observed data and current estimated parameters
- EM sometimes called estimation maximization algorithm

EM Algorithm: For Exponential Family PDFs

- Recall: Expectation

$$\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^t) = E\left[\ln b(\mathrm{y}) \mid \mathrm{x}, \boldsymbol{\theta}^t\right] + \mathrm{c}^T(\boldsymbol{\theta}) E\left[\mathrm{s}(\mathrm{y}) \mid \mathrm{x}, \boldsymbol{\theta}^t\right] - \ln a(\boldsymbol{\theta})$$

- Observe that $E\left[\ln b(\mathrm{y}) \mid \mathrm{x}, \boldsymbol{\theta}^t\right]$ does not depend on $\boldsymbol{\theta}$
- EM simplifies to
  - E-Step (equivalent):

$$\mathrm{s}^{t+1} = E\left[\mathrm{s}(\mathrm{y}) \mid \mathrm{x}, \boldsymbol{\theta}^t\right]$$

- M-Step:

$$\boldsymbol{\theta}^{t+1} = \arg\max_{\boldsymbol{\theta}} \mathrm{c}^T(\boldsymbol{\theta}) \mathrm{s}^{t+1} - \ln a(\boldsymbol{\theta})$$

Convergence of EM Algorithm

- Locally convergent
  - A contraction map
- Not necessarily to right point!

Gaussian Mixture Models

- $\mathsf{X} = \{\mathsf{x}^{(1)}, \mathsf{x}^{(2)}, \ldots, \mathsf{x}^{(N)}\}$ an N i.i.d random vectors that follows K -component Gaussian mixture distribution
- Probability density function:

$$p(\mathsf{x}^{(i)}|\boldsymbol{\theta}) = \sum_{j=1}^{K} \alpha_j \, \mathcal{N}(\mathsf{x}^{(i)}|\mu_j, \Sigma_j) \tag{3}$$

- $\mathcal{N}(.)$ denotes the normal distribution
- α_j denotes the mixing coefficient of the j -th Gaussian
- μ_j and Σ_j are the mean and covariance matrix of j -th Gaussian

Gaussian Mixture Model: Example

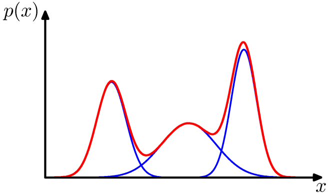


Figure: Mixture of 3 1D Gaussians (pdf)

GMM: ML Parameter Estimate

- Likelihood function:

$$L(\mathsf{X}, \boldsymbol{\theta}) = \sum_{i=1}^{N} \log \left(\sum_{j=1}^{K} \alpha_j \mathcal{N}(\mathsf{x}^{(i)} | \mu_j, \Sigma_j) \right) \tag{4}$$

- Goal: Compute the ML estimate of parameters $\boldsymbol{\theta} = \{\alpha_j, \mu_j, \Sigma_j\}_{j=1}^{K}$ that maximizes $L(\mathsf{X}, \boldsymbol{\theta})$

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} L(\mathsf{X}, \boldsymbol{\theta}) \tag{5}$$

- Analytical solution is not possible
  - Use iterative EM algorithm
  - X is treated as incomplete data
  - Introduce latent variables to simplify the optimization problem

EM for Gaussian Mixture Models

- Latent variable $\mathrm{z}^{(i)} = [z_1^{(i)}, \ldots, z_K^{(i)}]$ for each data vector $\mathrm{x}^{(i)}$
  - $\mathrm{z}^{(i)}$ indicates which component produced $\mathrm{x}^{(i)}$
  - If $\mathrm{x}^{(i)}$ is produced by the m -th mixture component, then $z_m{}^{(i)} = 1$ and $z_p{}^{(i)} = 0$, $\forall p \neq m$
- Complete data, $\mathrm{Y} = \{\mathrm{X}, \mathrm{Z}\}$ and complete log-likelihood:

$$
\begin{aligned}
L_c(\boldsymbol{\theta}, \mathrm{X}, \mathrm{Z}) &= \sum_{i=1}^{n} \log \left[\sum_{j=1}^{K} z_j^{(i)} \alpha_j \mathcal{N}(\mathrm{x}^{(i)} | \mu_j, \Sigma_j) \right] \\
&= \sum_{i=1}^{n} \sum_{j=1}^{K} z_j^{(i)} \log \left[\alpha_j \mathcal{N}(\mathrm{x}^{(i)} | \mu_j, \Sigma_j) \right] \qquad (6)
\end{aligned}
$$

- Why is the last step in the above equation valid?
- Actual value of $\mathrm{z}_j^{(i)}$ is unknown
- Use EM framework

EM for Gaussian Mixture Models: E-Step

- Complete data log-likelihood:

$$L_c(\boldsymbol{\theta}, \mathsf{X}, \mathsf{Z}) = \sum_{i=1}^{n} \sum_{j=1}^{K} z_j^{(i)} \log \left[\alpha_j \mathcal{N}(\mathsf{x}^{(i)} | \mu_j, \Sigma_j) \right] \tag{7}$$

- E-Step:

$$\begin{aligned} Q(\boldsymbol{\theta} | \boldsymbol{\theta}^{(t)}) =& E\left[L_c(\boldsymbol{\theta}, \mathsf{X}, \mathsf{Z}) | \mathsf{X}, \boldsymbol{\theta}^{(t)} \right] \\ =& \sum_{i=1}^{N} \sum_{j=1}^{K} \gamma_j^{(i)} \log \left[\alpha_j \mathcal{N}(\mathsf{x}^{(i)} | \mu_j, \Sigma_j) \right] \end{aligned} \tag{8}$$

- $\gamma_j^{(i)} = E\left[z_j^{(i)} | \mathsf{X}, \boldsymbol{\theta}^{(t)} \right] = Pr\left[z_j^{(i)} = 1 | \mathsf{x}^{(i)}, \boldsymbol{\theta}^{(t)} \right]$. Posterior probability that i^{th} observation came from j^{th} mixture component
- $\boldsymbol{\theta}^{(t)}$ is the current estimate of parameters

EM for Gaussian Mixture Models: M-Step

- Maximize conditional expectation $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \sum_{i=1}^{N}\sum_{j=1}^{K}\gamma_j^{(i)} log\left[\alpha_j \mathcal{N}(\mathrm{x}^{(i)}|\mu_j, \Sigma_j)\right] \tag{9}$$

- M-Step: Re-estimate parameters $\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$
  - Simple calculus (with constraints)

$$\alpha_j^{(t+1)} = \frac{\sum_{i=1}^{N}\gamma_j^{(i)}}{N} \qquad \mu_j^{(t+1)} = \frac{\sum_{i=1}^{N}\gamma_j^{(i)}\ \mathrm{x}^{(i)}}{\sum_{i=1}^{N}\gamma_j^{(i)}} \tag{10}$$

$$\Sigma_j^{(t+1)} = \frac{\sum_{i=1}^{N}\gamma_j^{(i)}\ (\mathrm{x}^{(i)} - \mu_j^{(t+1)})(\mathrm{x}^{(i)} - \mu_j^{(t+1)})^T}{\sum_{i=1}^{N}\gamma_j^{(i)}} \tag{11}$$

EM for Gaussian Mixture Models: Final Algo. Summary

- Initial parameter estimate $\boldsymbol{\theta}^{(0)}$
- Iteratively apply alternate E and M-steps
  - E-step: Estimate posterior probabilities $\gamma_j^{(i)}$ and the expected value of complete data likelihood,

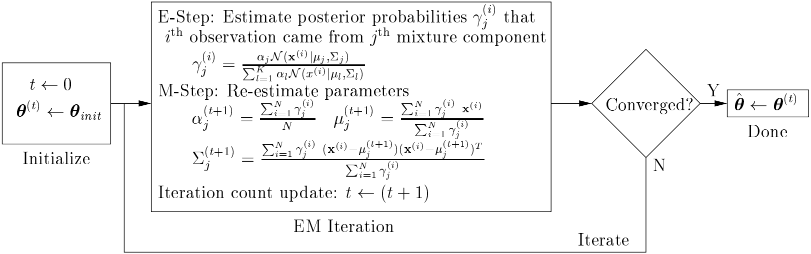
$$\gamma_j^{(i)} = \frac{\alpha_j \mathcal{N}(\mathrm{x}^{(i)} | \mu_j, \Sigma_j)}{\sum_{l=1}^{K} \alpha_l \mathcal{N}(x^{(i)} | \mu_l, \Sigma_l)} \tag{12}$$

  - M-step: Re-estimate parameters $\boldsymbol{\theta}^{(t+1)}$

$$\alpha_j^{(t+1)} = \frac{\sum_{i=1}^{N} \gamma_j^{(i)}}{N} \qquad \mu_j^{(t+1)} = \frac{\sum_{i=1}^{N} \gamma_j^{(i)} \ \mathrm{x}^{(i)}}{\sum_{i=1}^{N} \gamma_j^{(i)}} \tag{13}$$

$$\Sigma_j^{(t+1)} = \frac{\sum_{i=1}^{N} \gamma_j^{(i)} \ (\mathrm{x}^{(i)} - \mu_j^{(t+1)})(\mathrm{x}^{(i)} - \mu_j^{(t+1)})^T}{\sum_{i=1}^{N} \gamma_j^{(i)}} \tag{14}$$

EM Algorithm for Gaussian Mixture Models



Convergence of EM Algorithm

- Proof of convergence of EM algorithm (Dempster [1])
- Further extended by (Wu et al [5])
  - Minor corrections to Dempster's proof
  - Conditions under which EM converges to stationary points and local maxima
- EM has linear rate of convergence (Redner & Walker [4])
- EM shows asymptotic superlinear rate of convergence ([6, 3, 2])
  - Connection between first order Gradient ascent and EM algorithm
    - EM has faster convergence than gradient ascent, for well separated Gaussians
    - Superlinear rate of convergence, as overlap among Gaussian components goes to zero

Generalized EM and Convergence

- Generalized EM (GEM): Any algorithm that chooses $\boldsymbol{\theta}^{(t+1)}$ to increase $Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)})$ instead of maximizing.
- Specifically a GEM chooses any $\boldsymbol{\theta}^{(t+1)} \in \Omega$ (parameter space) such that,
$$Q(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}) \geq Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) \tag{15}$$
- GEM algorithm $\boldsymbol{\theta}^{(t+1)} = M(\boldsymbol{\theta}^{(t)})$.
  - an implicit mapping: $\boldsymbol{\theta} \to M(\boldsymbol{\theta})$ from parameter space Ω to itself
  - A fixed point $\boldsymbol{\theta}^*$ satisfies $M(\boldsymbol{\theta}^*) = \boldsymbol{\theta}^*$

Convergence Rate

- Linear Convergence Rate
  - A sequence $\{x^{(t)}\}$ is said to be linearly convergent if, there exists a number $r \in (0, 1)$ such that,

$$\lim_{t \to \infty} \frac{\left\| x^{(t+1)} - x^* \right\|}{\left\| x^{(t)} - x^* \right\|} = r \tag{16}$$

    - Rate of convergence r and speed of convergence $(1 - r)$
    - Superlinear if $r = 0$

Expectation Maximization: Summary

- EM provides a framework for addressing "missing data" problems
  - Iteratively optimize likelihood of observed "incomplete" data by "completing" it
    - Alternation between computation of expectation of unobserved variables given current parameter values and determination of maximizing parameters for "full data" likelihood
- EM algorithm is guaranteed to converge monotonically to a stationary point
- EM for GMM:
  - Fast convergence for well-separated Gaussians with similar sizes
  - Convergence slows down due to:
    - strong Overlap among Gaussian components
    - large dynamic range of the mixing coefficient ($\alpha_j < 0.1$)

References I

[1] A.P. Dempster, N.M. Laird, and D.B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 39.1 (1977), pp. 1–38.

[2] J. Ma and S. Fu. "On the correct convergence of the EM algorithm for Gaussian mixtures". In: *Pattern Recognition* 38.12 (2005), pp. 2602–2611.

[3] J. Ma, L. Xu, and M.I. Jordan. "Asymptotic convergence rate of the EM algorithm for Gaussian mixtures". In: *Neural Computation* 12.12 (2000), pp. 2881–2907.

[4] R. Redner and H. Walker. "Mixture Densities, maximum likelihood, and the EM Algorithm". In: *SIAM Review* 26.2 (1984), pp. 195–239.

[5] C. F. Jeff Wu. "On the convergence properties of the EM algorithm". In: *Annals of Statistics* 11 (1983), pp. 95–103.

References II

[6] L. Xu and M.I. Jordan. "On convergence properties of the EM algorithm for Gaussian mixtures". In: *Neural computation* 8.1 (1996), pp. 129–151.