4/29/15

EXPECTATION MAXIMIZATION (EM)

① EM for GMM Training/Fitting

k-Gaussians

Given: $D = \{\vec{x}_1, \ldots, \vec{x}_N\}$

Model: $Z \sim Cat(\vec{\pi})$

$X \mid Z = c \sim N(\vec{\mu}_c, \Sigma_c)$

Goal: Estimate parameters:

Θ (parameters)
- → Means: $\vec{\mu}_1, \vec{\mu}_2, \ldots, \vec{\mu}_k \in \mathbb{R}^d$
- → Cov.: $\Sigma_1, \ldots, \Sigma_k \in \mathbb{R}^{d \times d} \succeq 0$ [P.S.D]
- → Priors: $\pi_1, \ldots, \pi_k \geq 0 \quad \sum_{c=1}^{k} \pi_c = 1$

Estimator: Maximum (Marginal) Likelihood

$$\hat{\theta}_{MLE} = \underset{\theta}{\operatorname{argmax}} \sum_{i=1}^{N} \log P(\vec{X} = \vec{x}_i \mid \theta)$$

$$\sum_{i=1}^{N} \log \underbrace{\sum_{c=1}^{k}}_{\text{Problem!}} P(\vec{X} = \vec{x}_i, Z = c \mid \theta)$$

Idea: #1 What if someone told us the "GT" values of Z? Then it's easy!

$$\hat{\Theta}_{MLE} = \arg\max_{\Theta} \sum_{i=1}^{N} \left[\log P(\vec{X}=\vec{x}_i, \underbrace{Z=z_i}_{\text{GT provided}} \mid \Theta) \right]$$

$$= \arg\max_{\Theta} \sum_{i=1}^{N} \left[\log P(\vec{X}=\vec{x}_i \mid Z=z_i, \Theta) + \log P(Z=z_i \mid \Theta) \right]$$

all terms are "separable"

simple algebra

$$\pi_c = \frac{\#\text{ data-pts-assigned-to-Gaussian-}c\text{ in GT}}{N}$$

$$= \frac{\text{Count}(z_i = c)}{N}$$

$$\hat{\mu}_c = \frac{\sum_{z_i=c} \vec{x}_i}{\sum_{z_i=c} 1} \qquad \hat{\Sigma}_c = \frac{\sum_{z_i=c} (\vec{x}_i - \vec{\mu}_c)(\vec{x}_i - \vec{\mu}_c)^T}{\sum_{z_i=c} 1}$$

Idea #2: Even if some one told us "soft-GT" or soft assignment of pt to Gaussian, we would be fine.
Why? We ~~so~~ could do "soft" versions of above. (See next idea)

Idea #3 (Final EM alg for GMMs): Alternating Minimization w/ soft-assignment

$\longrightarrow$ Initialize $\Theta^{(0)} = \{ \pi_c^{(0)}, \vec{\mu}_c^{(0)}, \Sigma_c^{(0)} \}$

②

At iteration t

→ E(XPECTATION) - Step : Fix $\theta^{(t)}$; compute soft-assignment

$$a_{ic}^{(t)} = P(Z=c \mid \vec{X}=\vec{x}_i, \theta^{(t)})$$

$$= \frac{P(Z=c, \vec{X}=\vec{x}_i \mid \theta^{(t)})}{P(\vec{X}=\vec{x}_i \mid \theta^{(t)})}$$

$$= \frac{P(\vec{X}=\vec{x}_i \mid Z=c, \theta^{(t)})\, P(Z=c \mid \theta^{(t)})}{\sum_{k} P(\vec{X}=\vec{x}_i \mid Z=k, \theta^{(t)})\, P(Z=k \mid \theta^{(t)})}$$

prior → likelihood

$$= \frac{\pi_c^{(t)} \, N(\mu_c^{(t)}, \Sigma_c^{(t)})}{\underbrace{\sum_{k} \pi_k^{(t)} N(\mu_k^{(t)}, \Sigma_k^{(t)})}_{\text{Normalization}}}$$

"Just compute prior & likelihood from each gaussian at time (t) & renormalise"

→ M(AXIMIZATION) - Step : Fix $a_{ic}^{(t)}$; Compute $\theta^{(t+1)}$

$$\theta^{(t+1)} = \underset{\theta}{\operatorname{argmax}} \sum_{i=1}^{N} \sum_{c} a_{ic}^{(t)} \log P(\vec{X}=\vec{x}_i, Z=c \mid \theta)$$

$\equiv$ Learn from noisily / softly annotated dataset

Dataset part 1 [all instances labelled class 1]		Dataset part k [all instance labelled k]	
$(\vec{x}_1, 1)$	a_{11}	$(\vec{x}_1, k)$	a_{1k}
$(\vec{x}_2, 1)$	a_{21}	$(\vec{x}_2, k)$	a_{2k}
⋮	⋮	⋮	
$(x_N, 1)$	a_{N1}	$(\vec{x}_N, k)$	a_{Nk}

weights associated w/ each training pt

So we can easily derive estimators as:

$$\pi_c^{(t+1)} = \frac{\sum_{i=1}^{N} a_{ic}}{\underbrace{\sum_{c=1}^{K}\sum_{i=1}^{N} a_{ic}}_{N}}$$ } what ~~percentage~~ fraction of "mass" is associated with Gaussian C

$$\mu_c^{(t+1)} = \frac{\sum_i a_{ic} \vec{x}_i}{\sum_i a_{ic}}$$ } weighted mean

$$\Sigma_c^{(t+1)} = \frac{\sum_i a_{ic} (\vec{x}_i - \vec{\mu}_c)(\vec{x}_i - \vec{\mu}_c)^T}{\sum_i a_{ic}}$$ } weighted co-variance