SPR Day 15

Algorithms for (Penalized) ML estimation of parameters of Gaussian mixture model

Topics:

Review: ML estimation problem statement

Algorithm 1: Gradient Descent

Algorithm 2: Expectation Maximization (close analogy to K-means)

(Penalized) ML Estimation for Gaussian Mixture Model

Problem Statement

Given N observations $\{x_n\}_{n=1}^N$ each $x_n \in \mathbb{R}^D$

K number of clusters to find

Goal: Estimate all parameters of GMM w/ K clusters

$\{\mu_k\}_{k=1}^K$ cluster means
each $\mu_k \in \mathbb{R}^D$

$\{\Sigma_k\}_{k=1}^K$ cluster covariances
each Σ_k is $D \times D$ symmetric, pos. definite

$\{\pi_k\}_{k=1}^K$ cluster appearance probabilities
entire vector sums to one
$\sum_k \pi_k = 2, \quad \pi_k \geq 0$

Objective:

$$\max_{\pi \in \Delta^K, \mu \in \mathbb{R}^{K \times D}, \Sigma \text{ st } \Sigma_k \text{ is valid covariance}} \sum_n \log \text{GMMPDF}(x_n | \pi, \mu, \Sigma) - \lambda \sum_{k=1}^K \text{penalty}(\Sigma_k)$$

Example:

$$\text{penalty}(\Sigma_k) = -\sum_d \log \Sigma_{kdd}$$

log variance cannot be too small
avoids degenerate case where $\Sigma \to 0$ and likelihood $\to +\infty$

or, say Σ_k has a Gamma prior (1-d) or Wishart prior (M-dim) and do MAP estimation

ML Estimation for GMMs

Method 1: Gradient Descent

(not discussed in Bishop!)

Objective:

$$\max_{\substack{\pi \in \Delta^K \\ \mu \in \mathbb{R}^{K \times D} \\ \Sigma_k \in \text{valid covariance matrix}}} \underbrace{\sum_n \log \text{GMMPDF}(x_n | \pi, \mu, \Sigma) + \lambda \text{penalty}(\pi, \mu, \Sigma)}_{\mathcal{L}(\pi, \mu, \Sigma)}$$

Cannot do gradient descent when parameters are not admissable under any real value.

Why? Imagine $\pi_t \leftarrow \pi_{t-1} - \varepsilon \nabla_\pi \mathcal{L}(\pi)$

Our new π vector is no longer guaranteed to sum to one!

Solution: Reparameterize:

Let $s \in \mathbb{R}^K$, then define $\pi = \text{softmax}(s)$

Let $t \in \mathbb{R}^{K \times D}$, then define $\Sigma_k = \begin{bmatrix} e^{t_{k1}} & & \\ & \ddots & \\ & & e^{t_{kD}} \end{bmatrix}$

diagonal covariance

$e^{t_{kd}}$ ensures positive

Now can solve w/ 1st or 2nd order GD:

$$\max_{\substack{s \in \mathbb{R}^K \\ \mu \in \mathbb{R}^{K \times D} \\ t \in \mathbb{R}^{K \times D}}} \mathcal{L}(\text{softmax}(s), \mu, \exp(t))$$

$\exp(t)$: element wise exponential

How to compute gradients? Can do by hand, or via automatic differentiation libraries

Concerns: step size selection (as usual)

need to include penalty term to avoid pathologies as $\pi \to 0$

ML Estimation for GMMs:
Method 2: Expectation Maximization

Wish to solve:

$$\max_{\pi,\mu,\Sigma} \sum_n \underbrace{\log \sum_k \pi_k \, \text{NormPDF}(x_n | \mu_k, \Sigma_k)}_{\alpha_n(\pi,\mu,\Sigma)}$$

Let's consider what an optimal (π, μ, Σ) parameter would satisfy.

$$\vec{0} = \nabla_{\mu_j} \alpha = \sum_n \nabla_{\mu_j} \alpha_n(\pi,\mu,\Sigma)$$

$$= \sum_n \frac{1}{\sum_l \pi_l \, \text{NormPDF}(x_n|\mu_l,\Sigma_l)} \cdot \nabla_{\mu_j} \left[\sum_k \pi_k \, \text{NormPDF}(x_n|\mu_k,\Sigma_k) \right]$$

$$= \sum_n \frac{1}{\sum_l \pi_l \, \text{NormPDF}(x_n|\mu_l,\Sigma_l)} \cdot \pi_j \left[\nabla_{\mu_j} \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma_j|^{1/2}} e^{\overbrace{-\frac{1}{2}(x_n-\mu_j)^T \Sigma_j^{-1}(x_n-\mu_j)}^{f(\mu)}} \right]$$

$$= \sum_n \frac{1}{p(x_n)} \cdot \pi_j \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma_j|^{1/2}} e^{f(\mu)} \nabla_{\mu_j} [f(\mu)]$$

$$= \sum_n \frac{p(x_n, z_n = j)}{p(x_n)} \nabla_{\mu_j} \left[-\frac{1}{2}(x_n-\mu_j)^T \Sigma_j^{-1}(x_n-\mu_j) \right]$$

$$= \sum_n \frac{p(x_n, z_n = j)}{p(x_n)} \Sigma_j^{-1}(x_n - \mu_j)$$

posterior probability: $r_{nj} = p(z_{nj} = 1 | x_n)$, what Bishop calls γ_{nj}

Multiply both sides by Σ_j and rearrange

$$\Sigma_j \vec{0} = \sum_n r_{nj} \Sigma_j \Sigma_j^{-1}(x_n - \mu_j)$$

$$\vec{0} = \sum_n r_{nj}(x_n - \mu_j)$$

$$\hookrightarrow \mu_j = \frac{\sum_n r_{nj} x_n}{\sum_n r_{nj}}$$

At fixed point, $\mu_j \in \mathbb{R}^D$ each mean will be equal to weighted empirical mean of points assigned to cluster j

The Coordinate Ascent "E-M" Algorithm

So, we can work out optimal updates for all parameters given "responsibilities" $r_n \in \Delta^K$

where $r_{nj} \triangleq p(z_{nj}=1 \mid x_n)$

$$\mu_j = \frac{\sum_n r_{nj} x_n}{\sum_n r_{nj}}$$

Bishop PRML (9.17)

$$\Sigma_j = \frac{1}{\sum_n r_{nj}} \sum_n r_{nj} (x_n - \mu_j)(x_n - \mu_j)^T \quad (9.19)$$

may need to add small diagonal term to ensure positive definite

$$\pi_j = \frac{\sum_n r_{nj}}{N} \quad (9.22)$$

Leads to a coordinate ascent algorithm known as "Expectation-Maximization" (similar to k-means)

init: Some valid $\pi \in \Delta^K$, $\mu \in \mathbb{R}^{K \times D}$, $\Sigma : \{\Sigma_k\}_{k=1}^K$ each valid covariance

while not converged:

"E" step:

for each n in 1...N:

for each j in 1, 2, ... K:

$r_{nj} = \pi_j \text{NormPDF}(x_n \mid \mu_j, \Sigma_j)$

$r_n = \text{make\_sum\_to\_one}(r_{n1}, \ldots r_{nK})$

"M" step (no penalty term here):

for each j in 1...K:

Update μ_j with 9.17 given $\{r_n\}$

Σ_j with 9.19 "

π_j with 9.22 "

Advantages: No step sizes to select.