THU-70250043, Pattern Recognition (Spring 2018) **Homework: 3**

Gaussian Mixture Model and Expectation Maximization Algorithm

Lecturer: Changshui Zhang zcs@mail.tsinghua.edu.cn

Student: XXX xxx@mails.tsinghua.edu.cn

EM and Gradient Descent

In this problem you will investigate connections between the EM algorithm and gradient descent. Consider a GMM where $\Sigma_k = \sigma_k^2 I$, i.e., the covariances are spherical but of different spread. Moreover, suppose the mixture weight π_k is known. The log likelihood then is

$$l\left(\{\mu_k, \sigma_k^2\}_{k=1}^K\right) = \sum_{i=1}^n \log\left(\sum_{k=1}^K \pi_k \; N(x_i|\mu_k, \sigma_k^2 I)\right).$$

A maximization algorithm based on gradient descent is as follows:

- Initialize μ_k and σ_k^2, $k \in \{1, \cdots, K\}$. Set the iteration counter t=1.
- Repeat the following until convergence:
  - For $k = 1, \cdots, K$,
  $$\mu_k^{(t+1)} \leftarrow \mu_k^{(t)} + \eta_k^{(t)} \nabla_{\mu_k} l\left(\{\mu_k^{(t)}, (\sigma_k^2)^{(t)}\}_{k=1}^K\right)$$
  - For $k = 1, \cdots, K$,
  $$(\sigma_k^2)^{(t+1)} \leftarrow (\sigma_k^2)^{(t)} + s_k^{(t)} \nabla_{\sigma_k^2} l\left(\{\mu_k^{(t+1)}, (\sigma_k^2)^{(t)}\}_{k=1}^K\right)$$
  - Increase the iteration counter $t \leftarrow t+1$

Show that with properly chosen step size $\eta_k^{(t)}$ and $s_k^{(t)}$, the above gradient descent algorithm is equivalent to the following modified EM algorithm:

- Initialize μ_k and σ_k^2, $k \in \{1, \cdots, K\}$. Set the iteration counter t=1.
- Repeat the following until convergence:
  - E-step:
  $$\tilde{z}_{ik}^{(t+0.5)} \leftarrow Prob\left(x_i \in cluster_k|\{(\mu_j^{(t)}, (\sigma_j^2)^{(t)})\}_{j=1}^K, x_i\right),$$
  - M-step:
  $$\{\mu_k^{(t+1)}\}_{k=1}^K \leftarrow arg \max_{\{\mu_k\}_{k=1}^K} \sum_{i=1}^n \sum_{k=1}^K \tilde{z}_{ik}^{(t+0.5)} \left(\log \; N(x_i|\mu_k, (\sigma_k^2)^{(t)} I) + \log \pi_k\right)$$
  - E-step:
  $$\tilde{z}_{ik}^{(t+1)} \leftarrow Prob\left(x_i \in cluster_k|\{(\mu_j^{(t+1)}, (\sigma_j^2)^{(t)})\}_{j=1}^K, x_i\right),$$

– M-step:

$$\{(\sigma_k^2)^{(t+1)}\}_{k=1}^K \leftarrow arg \max_{\{\sigma_k\}_{k=1}^K} \sum_{i=1}^n \sum_{k=1}^K \tilde{z}_{ik}^{(t+1)} \left(\log\ N(x_i|\mu_k^{(t+1)}, \sigma_k^2 I) + \log \pi_k\right)$$

– Increase the iteration counter $t \leftarrow t + 1$

The main modification is inserting an extra E-step between the M-step for μ_k's and the M-step for σ_k^2's.

EM for MAP Estimation

The EM algorithm that we talked about in class was for solving a maximum likelihood estimation problem in which we wished to maximize

$$\prod_{i=1}^m p(x^{(i)};\theta) = \prod_{i=1}^m \sum_{z^{(i)}} p(x^{(i)}, z^{(i)};\theta) \tag{1}$$

where the $z^{(i)}$'s were latent random variables. Suppose we are working in a Bayesian framework, and wanted to find the MAP estimate of the parameters θ by maximizing

$$(\prod_{i=1}^m p(x^{(i)};\theta))p(\theta) = (\prod_{i=1}^m \sum_{z^{(i)}} p(x^{(i)}, z^{(i)}|\theta))p(\theta) \tag{2}$$

Here, $p(\theta)$ is our prior on the parameters. Generalize the EM algorithm to work for MAP estimation. You may assume that $log\ p(x, z|\theta)$ and $log\ p(\theta)$ are both concave in θ, so that the M-step is tractable if it requires only maximizing a linear combination of these quantities. (This roughly corresponds to assuming that MAP estimation is tractable when x, z is fully observed, just like in the frequentist case where we considered examples in which maximum likelihood estimation was easy if x, z was fully observed.)

Make sure your M-step is tractable, and also prove that $(\prod_{i=1}^m p(x^{(i)};\theta))p(\theta)$(viewed as a function of θ) monotonically increases with each iteration of your algorithm.

Programming

Points	ω_1			ω_2		
	x_1	x_2	x_3	x_1	x_2	x_3
1	0.42	-0.087	0.58	-0.4	0.58	0.089
2	-0.2	-3.3	-3.4	-0.31	0.27	-0.04
3	1.3	-0.32	1.7	0.38	0.055	-0.035
4	0.39	0.71	0.23	-0.15	0.53	0.011
5	-1.6	-5.3	-0.15	-0.35	0.47	0.034
6	-0.029	0.89	-4.7	0.17	0.69	0.1
7	-0.23	1.9	2.2	-0.011	0.55	-0.18
8	0.27	-0.3	-0.87	-0.27	0.61	0.12
9	-1.9	0.76	-2.1	-0.065	0.49	0.0012
10	0.87	-1.0	-2.6	-0.12	0.054	-0.063

Table 1: Data for Programming

Suppose we know that the ten data points in category ω_1 in the table above come from a three-dimensional Gaussian. Suppose, however, that we do not have access to the x_3 components for the even-numbered data points.

- Write an EM program to estimate the mean and covariance of the distribution. Start your estimate with $\mu_0 = 0$ and $\Sigma_0 = I$, the three-dimensional identity matrix.
- Compare your final estimate with that for the case when there is no missing data

Suppose we know that the ten data points in category ω_2 in the table above come from a three-dimensional uniform distribution $p(x|\omega_2) \sim U(x_l, x_u)$. Suppose, however, that we do not have access to the x_3 components for the even-numbered data points.

- Write an EM program to estimate the six scalars comprising x_l and x_u of the distribution. Start your estimate with $x_l = (-2, -2, -2)^t$ and $x_u = (+2, +2, +2)^t$.
- Compare your final estimate with that for the case when there is no missing data.