Intelligent Data Analysis

Martin Russell
School of Computer Science
Thursday, 14 May 2020

Exercise sheet – week 8 – Gaussian Mixture Models (GMMs)

1. Let $X = \{x_1, \ldots, x_N\}$be a set of real numbers.
   a. Show that the Maximum Likelihood estimate of the parameters m and v of a Gaussian probability density function for the set are given by:
   $m = \frac{1}{N}\sum_{n=1}^{N} x_n$, $v = \frac{1}{N}\sum_{n=1}^{N}(x_n - m)^2$. **[6 marks]**
   b. Are these values of and a local or global maximum? Justify your answer. **[4 marks]**

Solution:

a. The derivation of m was done in class. For v we want to maximise

$$P(X) = \prod_{n=1}^{N} p(x_n|m, v)$$

As a function of v. Note that

$$\log(P(X)) = \sum_{n=1}^{N} \log(p(x_n|m, v)) = \sum_{n=1}^{N} -\frac{1}{2}\log(2\pi v) - \frac{(x_n - m)^2}{2v}$$

Therefore,

$$\frac{d}{dv}\log(P(X)) = -\sum_{n=1}^{N} \frac{d}{dv}[-\frac{1}{2}log(2\pi v) - \frac{(x_n - m)^2}{2v}]$$

$$= -\sum_{n=1}^{N}[-\frac{1}{2} \times \frac{2\pi}{2\pi v} - \frac{2v \times 0 - 2(x_n - m)^2}{4v^2}]$$

Setting this to zero and multiplying by $2v^2$ gives

$$0 = \sum_{n=1}^{N}[v - (x_n - m)^2]$$

From which it follows that

$$v = \frac{1}{N}\sum_{n=1}^{N}(x_n - m)^2$$

b. They are a global optimum. Unlike the case with a Gaussian mixture model the solution is closed and has a unique solution. Of course, to demonstrate that the critical point is a maximum rather than a minimum you also need to look at the second derivative.

2. A 2-dimensional Gaussian PDF g has mean $m = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ and covariance matrix $C = \begin{bmatrix} 13 & -5.2 \\ -5.2 & 7 \end{bmatrix}$.

   The matrix C has eigenvalue decomposition $\mathrm{C} = \mathrm{UDU}^{\mathrm{T}}$, where:

$$U = \begin{bmatrix} cos\left(\frac{\pi}{3}\right) & -sin\left(\frac{\pi}{3}\right) \\ sin\left(\frac{\pi}{3}\right) & cos\left(\frac{\pi}{3}\right) \end{bmatrix}, D = \begin{bmatrix} 4 & 0 \\ 0 & 16 \end{bmatrix}.$$

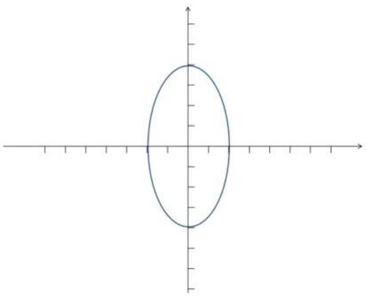
a. Sketch a 1-standard-deviation contour for g. **[4 marks].**

b. Calculate $g\left(\begin{bmatrix}1.5\\1.5\end{bmatrix}\right)$. Show all of your calculations. **[4 marks].**

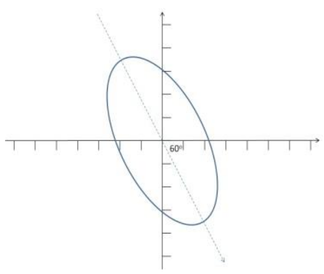
Solution:

a. First notice that the matrix U implements a rotation anti-clockwise around the origin through an angle of $\frac{\pi}{3}$.

Next use the fact that relative to the new basis (consisting of the eigenvectors of C, which are the columns of U), the covariance matrix is $D = \begin{bmatrix} 4 & 0 \\ 0 & 16 \end{bmatrix}$, and so the 1-standard deviation contour looks like figure (A) below. To get the required contour this needs to be rotated anti-clockwise though 60°, as in figure (B) below.



(A)



(B)

b. To calculate g$\left(\begin{bmatrix}1.5\\1.5\end{bmatrix}\right)$ use the standard formula for a multivariate Gaussian PDF:

$$g(x) = \frac{1}{\sqrt{(2\pi)^d|C|}} e^{-\frac{1}{2}(x-m)^T C^{-1}(x-m)}$$

with $d = 2, m = \begin{bmatrix}2\\2\end{bmatrix}$ and $C = \begin{bmatrix} 13 & -5.2 \\ -5.2 & 7 \end{bmatrix}$.

We have: $|C| = 64, C^{-1} = \begin{bmatrix} 0.109 & 0.081 \\ 0.081 & 0.203 \end{bmatrix}$, so

$$g\left(\begin{bmatrix}1.5\\1.5\end{bmatrix}\right) = \frac{1}{\sqrt{(2\pi)^2 \times 64}} e^{-\frac{1}{2}\left(\begin{bmatrix}1.5\\1.5\end{bmatrix}-\begin{bmatrix}2\\2\end{bmatrix}\right)^T \begin{bmatrix} 0.109 & 0.081 \\ 0.081 & 0.203 \end{bmatrix}\left(\begin{bmatrix}1.5\\1.5\end{bmatrix}-\begin{bmatrix}2\\2\end{bmatrix}\right)} = 0.0188$$

3. What are the similarities and differences between using a Mcomponent GMM to model a set of data points in N dimensional space compared to using a set of M centroids obtained through clustering?

[4 marks].

Solution:

Similarities:

a. Both methods model the data using a small set of carefully positioned data points. In clustering these are the M centroids $c_1, \dots, c_M$and in a GMM these are the M component means $\mu_1, \dots, \mu_M$.

Differences:

a. In clustering there is no further information. In a GMM, as well as the component means $\mu_1, \dots, \mu_M$ we have the component variances $\vartheta_1, \dots, \vartheta_M$, which indicate the spread of the data around the mean, and the component weights $w_1, \dots, w_M$ which indicate the proportion of the data that is modelled by the m^{th} component.

b. During training, in clustering a data point x is associated with its closest centroid c_x and only contributes to the re-estimation of the centroid c_x, so

$$c_{new} = \frac{1}{N_c} \sum_{\substack{c \text{ is closest} \\ \text{centroid to } x_n}} x_n$$

where N_c is the number of data points that have c as their closest centroid. In the E-M algorithm all of the data points contribute to the re-estimation of all of the centroids, but the extent to which x contributes to the reestimate of the m^{th} component depends on the posterior probability of the m^{th} component given x:

$$\mu_m^{new} = \frac{1}{P_m} \sum_{n=1}^{N} P(m|x_n) x_n$$

where $P(m|x_n)$ is the probability of the m^{th} GMM component given the data point x_n and $P_m = \sum_{n=1}^{N} P(m|x_n)$.

4. Why is the E-M algorithm necessary? Why doesn't the simple maximum likelihood parameter estimation procedure from question 1 apply to an -component Gaussian Mixture Model (GMM)? **[4 marks].**

Solution:

In the case of a single Gaussian PDF, all of the samples are assumed to come from the same distribution, and therefore the Maximum Likelihood technique from Q1 applies. In an *M* component GMM we don't know which component a sample x_n should be assigned to, so we don't know which of the *M* means and variances it should contribute to according to the equations in Q1. Therefore the solution involves two stages (1) decide the proportion of each sample that should be allocated to each component (this is the "E" step) and (2) use these proportions, together with the samples, to update the GMM parameters (the "M" step). It can be shown that the "proportion" of a sample x_n that should be allocated to the m^{th} GMM component is $P(m|x_n)$ - the posterior probability of the m^{th} component given x_n.

5. Let $X = \{x_1, x_2, x_3, x_4\}$, where x_1= 1, x_2 = 7, x_3 = 5, x_4 = 4. Suppose that:

- g_1 is a Gaussian PDF with mean m_1 = 2 and variance v_1 = 2, and
- g_2 is a Gaussian PDF with mean m_2 = 3 and variance v_2 = 2, and
- g is the Gaussian Mixture Model $g(x) = 0.3 \times g_1(x) + 0.7 \times g_2(x)$.

a. Calculate the new values of the means m_1 and m_2 after the application of one iteration of the E-M algorithm with the samples X. **[8 marks]**

b. Are the new values of means and guaranteed to correspond to a global or local maximum of the likelihood function? **[4 marks].**

Solution:

a. We need to calculate the (posterior) probabilities $P(1|x_n)$ and $P(2|x_n)$ for each data point x_n, where $P(m|x_n)$ is the posterior probability of the m^{th} component given the data point x_n.

First note that, for example:

$$g_1(x_n) = \frac{1}{\sqrt{2\pi v_1}} e^{-\frac{(x_n - m_1)^2}{2v_1}}$$

So,

$$g_1(x_1) = g_1(1) = \frac{1}{\sqrt{2\pi \times 2}} e^{-\frac{(1-2)^2}{2\times 2}} = \frac{1}{2\sqrt{\pi}} e^{-\frac{1}{4}} = 0.22$$

Similarly,

$$g_1(x_2) = 0.0005, g_1(x_3) = 0.03, g_1(x_4) = 0.104$$
$$g_2(x_1) = 0.04, g_2(x_2) = 0.03, g_2(x_3) = 0.03, g_2(x_4) = 0.035$$

Now apply Bayes' rule to obtain the posterior probabilities:

$$P(1|x_1) = \frac{g_1(x_1)w_1}{g_1(x_1)w_1 + g_2(x_1)w_2} = \frac{0.22 \times 0.3}{0.22 \times 0.3 + 0.04 \times 0.7} = 0.7$$

Similarly,

$$P(1|x_2) = 0.008, P(1|x_3) = 0.3, P(1|x_4) = 0.56$$
$$P(2|x_1) = 0.3, P(2|x_2) = 0.99, P(2|x_3) = 0.7, P(2|x_4) = 0.44$$

Therefore,

$$\overline{m_1} = \frac{P(1|x_1)x_1 + P(1|x_2)x_2 + P(1|x_3)x_3 + P(1|x_4)x_4}{P(1|x_1) + P(1|x_2) + P(1|x_3) + P(1|x_4)}$$
$$= \frac{0.7 \times 1 + 0.008 \times 7 + 0.3 \times 5 + 0.56 \times 4}{0.7 + 0.008 + 0.3 + 0.56} = 2.86$$

Similarly, $\overline{m_2} = 5.14$.

b. All that is guaranteed is that the probability of the training set of samples given the new estimates of the means will be greater than or equal to their probability given the old means. If the process is continued through multiple iterations, then the values of m_1 and m_2 will converge and will not be significantly changed by further applications of the E-M algorithm. Even then, the parameters are only guaranteed to correspond to a local maximum of the likelihood function. This is because at each iteration of the E-M algorithm all that is guaranteed is an increase in the likelihood function, therefore when the parameters reach a local maximum they cannot escape from it.

The E-M algorithm is slightly cleverer than a "standard" gradient ascent algorithm. In standard gradient ascent, give a parameter θ, we would calculate the derivative $\frac{\partial L}{\partial \theta}$ of the likelihood L w.r.t. θ and define $\bar{\theta} = \theta + \rho \frac{\partial L}{\partial \theta}$, where ρ is the learning rate, which we have to choose. In E-M we replace θ with its update $\bar{\theta}$ - there is no need to worry about a learning rate. However, the derivation of the E-M algorithm relies on standard calculus - calculate the derivative and set it to zero. When the E-M algorithm reaches a local maximum, the derivative will be zero and the process is stuck.

[Total marks 38]