1 Density Estimation

1.1 Slide 5

We see that when Δ is very small (top figure), the resulting density model is very spiky, with a lot of structure that is not present in the underlying distribution that generated the data set. Conversely, if Δ is too large (bottom figure) then the result is a model that is too smooth and that consequently fails to capture the bimodal property of the green curve. The best results are obtained for some intermediate value of Δ (middle figure). In principle, a histogram density model is also dependent on the choice of edge location for the bins, though this is typically much less significant than the value of Δ.

1.2 Slide 11

The scikit-learn toolkit defines a number of different kernels in addition to the Gaussian:

- Tophat kernel
$$k(x;h) \propto 1 \text{ if } x < h$$
- Epanechnikov kernel
$$k(x;h) \propto 1 - \frac{x^2}{h^2}$$
- Exponential kernel
$$k(x;h) \propto \exp(-x/h)$$
- Linear kernel
$$k(x;h) \propto 1 - x/h \text{ if } x < h$$
- Cosine kernel
$$k(x;h) \propto \cos\left(\frac{\pi x}{2h}\right) \text{ if } x < h$$

See `https://scikit-learn.org/stable/modules/density.html` for more details.

2 Mixture models

2.1 Slide 4

Note that in the definition of the GMM, $\mathbf{\Sigma}_k$ represents a general covariance matrix. We will see later how we can restrict the type of matrix to define different types of mixture model.

The values of z_k therefore satisfy $z_k \in \{0,1\}$ and $\sum_k z_k = 1$, and we see that there are K possible states for the vector $\mathbf{z}$ according to which element is nonzero.

2.2 Slide 5

What does this representation give us? We are now able to work with the joint distribution $p(\mathbf{x}, \mathbf{z})$ instead of the marginal distribution $p(\mathbf{x})$, and this will lead to significant simplifications, most notably through the introduction of the expectation-maximization (EM) algorithm.

2.3 Slide 7

We can sample from a Gaussian mixture model as follows. We first generate a value for $\mathbf{z}$, which we denote $\widehat{\mathbf{z}}$, from the marginal distribution $p(\mathbf{z})$ and then generate a value for $\mathbf{x}$ from the conditional distribution $p(\mathbf{x}|\widehat{\mathbf{z}})$. We can depict samples from the joint distribution $p(\mathbf{x}, \mathbf{z})$ by plotting points at the corresponding values of $\mathbf{x}$ and then colouring them according to the value of $\mathbf{z}$, in other words according to which Gaussian component was responsible for generating them, as shown in Figure (a).

We can also use this synthetic data set to illustrate the 'responsibilities' by evaluating, for every data point, the posterior probability for each component in the mixture distribution from which this data set was generated. In particular, we can represent the value of the responsibilities $\gamma(z_{nk})$ associated with data point $\mathbf{x}_n$ by plotting the corresponding point using proportions of red, blue, and green ink given by $\gamma(z_{nk})$ for $k = 1, 2, 3$, respectively, as shown in Figure (c). So, for instance, a data point for which $\gamma(z_{n1}) = 1$ will be coloured red, whereas one for which $\gamma(z_{n2}) = \gamma(z_{n3}) = 0.5$ will be coloured with equal proportions of blue and green ink and so will appear cyan. This should be compared with Figure (a) in which the data points were labelled using the true identity of the component from which they were generated.

The data set in (a) is said to be *complete*, whereas that in (b) is *incomplete.*

2.4 Slide 9

The maximization of the log likelihood function is not a well posed problem because such singularities will always be present and will occur whenever one of the Gaussian components 'collapses' onto a specific data point.

2.5 Slide 10

We see that the mean $\boldsymbol{\mu}_k$ for the k^{th} Gaussian component is obtained by taking a weighted mean of all of the points in the data set, in which the weighting factor for data point $\mathbf{x}_n$ is given by the posterior probability $\gamma(z_{nk})$ that component k was responsible for generating $\mathbf{x}_n$.

2.6 Slide 13

We can show that each update to the parameters resulting from an E step followed by an M step is guaranteed to increase the log likelihood function (see [Bishop, 2006] Section 9.4).

2.7 Slide 14

Here a mixture of two Gaussians is used, with centres initialised away from the data (so as to illustrate convergence: in a real application the centres would be initialised within the data), and with covariance matrices initialized to be proportional to the unit matrix. Plot (a) shows the data points in green, together with the initial configuration of the mixture model in which the one standard-deviation contours for the two Gaussian components are shown as blue and red circles. Plot (b) shows the result of the initial E step, in which each data point is depicted using a proportion of blue ink equal to the posterior probability of having been generated from the blue component, and a corresponding proportion of red ink given by the posterior probability of having been generated by the red component. Thus, points that have a significant probability for belonging to either cluster appear purple. The situation after the first M step is shown in plot (c), in which the mean of the blue Gaussian has moved to the mean of the data set, weighted by the probabilities of each data point belonging to the blue cluster, in other words it has moved to the centre of mass of the blue ink. Similarly, the covariance of the blue Gaussian is set equal to the covariance of the blue ink. Analogous results hold for the red component. Plots (d), (e), and (f) show the results after 2, 5, and 20 complete cycles of EM, respectively. In plot (f) the algorithm is close to convergence.

2.8 Slide 15

EM for Gaussian Mixtures.

1. Initialize the means $\boldsymbol{\mu}_k$, covariances $\boldsymbol{\Sigma}_k$ and mixing coefficients π_k, and evaluate the initial value of the log likelihood.

2. **E step**. Evaluate the responsibilities using the current parameter values

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\displaystyle\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}. \tag{2.1}$$

3. **M step**. Re-estimate the parameters using the current responsibilities

$$\boldsymbol{\mu}_k^{\text{new}} = \frac{1}{N_k}\sum_{n=1}^{N} \gamma(z_{nk})\mathbf{x}_n \tag{2.2}$$

$$\boldsymbol{\Sigma}_k^{\text{new}} = \frac{1}{N_k}\sum_{n=1}^{N} \gamma(z_{nk})\left(\mathbf{x}_n - \boldsymbol{\mu}_k^{\text{new}}\right)\left(\mathbf{x}_n - \boldsymbol{\mu}_k^{\text{new}}\right)^{\mathrm{T}} \tag{2.3}$$

$$\pi_k^{\text{new}} = \frac{N_k}{N} \tag{2.4}$$

where

$$N_k = \sum_{n=1}^{N} \gamma(z_{nk}). \tag{2.5}$$

4. Evaluate the log likelihood

$$\ln p(\mathbf{X}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi}) = \sum_{n=1}^{N} \ln\left\{\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right\} \tag{2.6}$$

and check for convergence of either the parameters or the log likelihood. If the convergence criterion is not satisfied return to step 2.

2.9 Slide 18

The exponential family is a broad class of probability distributions that includes the exponential, Gaussian, Bernoulli, t-distribution, gamma, and binomial distributions. These distributions have the form

$$p(\mathbf{x}|\boldsymbol{\eta}) = h(\mathbf{x})g(\boldsymbol{\eta})\exp\left\{\boldsymbol{\eta}^{\mathrm{T}}\mathbf{u}(\mathbf{x})\right\} \tag{2.7}$$

where $\mathbf{x}$ may be scalar or vector, and may be discrete or continuous. Here $\boldsymbol{\eta}$ are called the *natural parameters* of the distribution, and $\mathbf{u}(\mathbf{x})$ is some function of $\mathbf{x}$. The function $g(\boldsymbol{\eta})$ can be interpreted as the coefficient that ensures that the distribution is normalized and therefore satisfies

$$g(\boldsymbol{\eta})\int h(\mathbf{x})\exp\left\{\boldsymbol{\eta}^{\mathrm{T}}\mathbf{u}(\mathbf{x})\right\}\,\mathrm{d}\mathbf{x} = 1 \tag{2.8}$$

where the integration is replaced by summation if $\mathbf{x}$ is a discrete variable.

The exponential family has a number of properties that make statistical inference easier. More details can be found in Section 2.4 of [Bishop, 2006].

2.10 Slide 20

The general EM algorithm has the property that each cycle of EM will increase the incomplete-data log likelihood (unless it is already at a local maximum).

Given a joint distribution $p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$ over observed variables $\mathbf{X}$ and latent variables $\mathbf{Z}$, governed by parameters $\boldsymbol{\theta}$, the goal is to maximize the likelihood function $p(\mathbf{X}|\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$.

1. Choose an initial setting for the parameters $\boldsymbol{\theta}^{\mathrm{old}}$.
2. **E step** Evaluate $p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{\mathrm{old}})$.

3. **M step** Evaluate $\boldsymbol{\theta}^{\mathrm{new}}$ given by

$$\boldsymbol{\theta}^{\mathrm{new}} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\mathrm{old}}) \tag{2.9}$$

where

$$\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\mathrm{old}}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{\mathrm{old}}) \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}). \tag{2.10}$$

4. Check for convergence of either the log likelihood or the parameter values. If the convergence criterion is not satisfied, then let

$$\boldsymbol{\theta}^{\mathrm{old}} \leftarrow \boldsymbol{\theta}^{\mathrm{new}} \tag{2.11}$$

and return to step 2.

2.11 Slide 25

If we substitute $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{\mathrm{old}})$ into the relevant equation, we see that, after the E step, the lower bound takes the form

$$\begin{aligned}\mathcal{L}(q, \boldsymbol{\theta}) &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{\mathrm{old}}) \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}) - \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{\mathrm{old}}) \ln p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{\mathrm{old}}) \\ &= \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\mathrm{old}}) + \mathrm{const}\end{aligned} \tag{2.12}$$

where the constant is simply the negative entropy of the q distribution and is therefore independent of $\boldsymbol{\theta}$. Thus in the M step, the quantity that is being maximized is the expectation of the complete-data log likelihood, as we saw earlier in the case of mixtures of Gaussians. Note that the variable $\boldsymbol{\theta}$ over which we are optimizing appears only inside the logarithm. If the joint distribution $p(\mathbf{Z}, \mathbf{X}|\boldsymbol{\theta})$ comprises a member of the exponential family, or a product of such members, then we see that the logarithm will cancel the exponential and lead to an M step that will be typically much simpler than the maximization of the corresponding incomplete-data log likelihood function $p(\mathbf{X}|\boldsymbol{\theta})$.

3 Numerical Linear Algebra

For greater mathematical detail, including all the algorithmic details, refer to [Stoer and Bulirsch, 1983] and [Golub and van Loan, 1996].

4 Generative Topographic Mapping

4.1 Slide 9

The full derivation of an EM algorithm for probabilistic PCA can be found in Section 12.2.2 of [Bishop, 2006].

Another elegant feature of the EM approach is that we can take the limit $\sigma^2 \to 0$, corresponding to standard PCA, and still obtain a valid EM-like algorithm [Roweis, 1998].

Let us define $\widetilde{\mathbf{X}}$ to be a matrix of size $N \times D$ whose n^{th} row is given by the vector $\mathbf{x}_n - \overline{\mathbf{x}}$ and similarly define $\boldsymbol{\Omega}$ to be a matrix of size $D \times M$ whose n^{th} row is given by the vector $\mathbb{E}[\mathbf{z}_n]$. The E step of the EM algorithm for PCA then becomes

$$\boldsymbol{\Omega} = (\mathbf{W}_{\text{old}}^{\text{T}}\mathbf{W}_{\text{old}})^{-1}\mathbf{W}_{\text{old}}^{\text{T}}\widetilde{\mathbf{X}} \tag{4.1}$$

and the M step takes the form

$$\mathbf{W}_{\text{new}} = \widetilde{\mathbf{X}}^{\text{T}}\boldsymbol{\Omega}^{\text{T}}(\boldsymbol{\Omega}\boldsymbol{\Omega}^{\text{T}})^{-1}. \tag{4.2}$$

The figure shows synthetic data illustrating the EM algorithm for PCA defined by (4.1) and (4.2). (a) A data set $\mathbf{X}$ with the data points shown in green, together with the true principal components (shown as eigenvectors scaled by the square roots of the eigenvalues). (b) Initial configuration of the principal subspace defined by $\mathbf{W}$, shown in red, together with the projections of the latent points $\mathbf{Z}$ into the data space, given by $\mathbf{Z}\mathbf{W}^{\text{T}}$, shown in cyan. (c) After one M step, the latent space has been updated with $\mathbf{Z}$ held fixed. (d) After the successive E step, the values of $\mathbf{Z}$ have been updated, giving orthogonal projections, with $\mathbf{W}$ held fixed. (e) After the second M step. (f) After the second E step.

4.2 Slide 18

The M-step consists of maximising the expectation of the complete-data log likelihood

$$\langle \mathcal{L}_{\text{comp}}(\mathbf{W}, \sigma) \rangle = \sum_{n=1}^{N}\sum_{j=1}^{M} R_{jn}^{(m)}(\mathbf{W}^{(m)}, \sigma^{(m)}) \ln\{p(\mathbf{x}_n|\mathbf{z}_j, \mathbf{W}, \sigma)\}, \tag{4.3}$$

which gives the following equation for $\mathbf{W}$:

$$\sum_{n=1}^{N}\sum_{j=1}^{M} R_{jn}^{(m)}(\mathbf{W}^{(m)}, \sigma^{(m)})\{\mathbf{W}^{(m+1)}\boldsymbol{\phi}(\mathbf{z}_j) - \mathbf{x}_n\}\boldsymbol{\phi}^T(\mathbf{z}_j) = 0. \tag{4.4}$$

This can be written in matrix form as

$$\boldsymbol{\Phi}^T\mathbf{G}^{(m)}\boldsymbol{\Phi}(\mathbf{W}^{(m+1)})^T = \boldsymbol{\Phi}^T\mathbf{R}^{(m)}\mathbf{X}, \tag{4.5}$$

where $\boldsymbol{\Phi}$ is the $M \times K$ RBF *design matrix* with elements $\Phi_{ji} = \phi_i(\mathbf{z}_j)$, $\mathbf{X}$ is the $N \times d$ data matrix, $\mathbf{R}$ is an $M \times N$ responsibility matrix with elements R_{jn}, and $\mathbf{G}$ is an $M \times M$ diagonal matrix with elements

$$G_{jj} = \sum_{n=1}^{N} R_{jn}(\mathbf{W}, \sigma). \tag{4.6}$$

4.3 Slide 20

To use GTM for visualisation, it is necessary to map data points $\mathbf{x}_n$ to corresponding points in latent space. As with PPCA, we do this by using Bayes' theorem to compute the posterior density $p(\mathbf{z}|\mathbf{x}_n)$. With our choice of prior distribution, this is given by a sum of delta functions centred at the lattice points $\mathbf{z}_j$ with weights given by the responsibilities R_{jn}. However, in order to visualise a whole dataset in a single plot, we need to find a statistic to summarise this distribution. One convenient statistic to use is the mean:

$$\langle \mathbf{z}|\mathbf{x}_n, \mathbf{W}, \sigma\rangle = \sum_{j=1}^{M} R_{jn}\mathbf{z}_j. \tag{4.7}$$

References

[Bishop, 2006] Bishop, C. M. 2006. *Pattern recognition and machine learning.* Springer.

[Golub and van Loan, 1996] Golub, G. H. and C. F. van Loan 1996. *Matrix Computations.* Baltimore: Johns Hopkins University Press.

[Roweis, 1998] Roweis, S. 1998. EM algorithms for PCA and SPCA. *Advances in neural information processing systems*, 626–632.

[Stoer and Bulirsch, 1983] Stoer, J. and R. Bulirsch 1983. *Introduction to Numerical Analysis.* New York: Springer-Verlag.