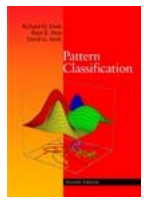
Chapter 10: Unsupervised Learning and Clustering

- Introduction
- Mixture Densities and Identifiability
- MI Estimates
- Application to Normal Mixtures



Introduction

- **Previously, all our training samples were labeled: these samples were said "supervised"**
- **We now investigate a number of "unsupervised" procedures which use unlabeled samples**
- **Collecting and Labeling a large set of sample patterns can be costly**
- **We can train with large amounts of (less expensive) unlabeled data, and only then use supervision to label the groupings found, this is appropriate for large "data mining" applications where the contents of a large database are not known beforehand**

- **This is also appropriate in many applications when the characteristics of the patterns can change slowly with time**
- **Improved performance can be achieved if classifiers running in a unsupervised mode are used**
- **We can use unsupervised methods to identify features that will then be useful for categorization**
- **We gain some insight into the nature (or structure) of the data**

Mixture Densities and Identifiability

- **We shall begin with the assumption that the functional forms for the underlying probability densities are known and that the only thing that must be learned is the value of an unknown parameter vector**
- **We make the following assumptions:**
  - **The samples come from a known number c of classes**
  - **The prior probabilities $P(\omega_j)$ for each class are known $(j = 1, \ldots, c)$**
  - **$P(x \mid \omega_j, \theta_j)$ $(j = 1, \ldots, c)$ are known**
  - **The values of the c parameter vectors $\theta_1, \theta_2, \ldots, \theta_c$ are unknown**

- The category labels are unknown

$$P(x \mid \theta) = \sum_{j=1}^{c} \underbrace{P(x \mid \omega_j, \theta_j)}_{\text{component densities}} \cdot \underbrace{P(\omega_j)}_{\text{mixing parameters}}$$

where $\theta = (\theta_1, \theta_2, ..., \theta_c)^t$

– This density function is called a mixture density

– Our goal will be to use samples drawn from this mixture density to estimate the unknown parameter vector θ. Once θ is known, we can decompose the mixture into its components and use a MAP classifier on the derived densities

– Definition

- A density $P(x \mid \theta)$ is said to be identifiable if $\theta \neq \theta'$ implies that there exists an x such that:

$$P(x \mid \theta) \neq P(x \mid \theta')$$

As a simple example, consider the case where x is binary and $P(x \mid \theta)$ is the mixture:

$$P(x \mid \theta) = \frac{1}{2}\theta_1^x (1-\theta_1)^{1-x} + \frac{1}{2}\theta_2^x (1-\theta_2)^{1-x}$$

$$= \begin{cases} \frac{1}{2}(\theta_1 + \theta_2) \text{ if } x = 1 \\ 1 - \frac{1}{2}(\theta_1 + \theta_2) \text{ if } x = 0 \end{cases}$$

Assume that:

$$P(x = 1 \mid \theta) = 0.6 \Rightarrow P(x = 0 \mid \theta) = 0.4$$

by replacing these probabilities values, we obtain:

$$\theta_1 + \theta_2 = 1.2$$

- Thus, we have a case in which the mixture distribution is completely unidentifiable, and therefore unsupervised learning is impossible
- In the discrete distributions, if there are too many components in the mixture, there may be more unknowns than independent equations, and identifiability can become a serious problem!
- While it can be shown that mixtures of normal densities are usually identifiable, the parameters in the simple mixture density

$$P(x \mid \theta) = \frac{P(\omega_1)}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}(x-\theta_1)^2\right] + \frac{P(\omega_2)}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}(x-\theta_2)^2\right]$$

Cannot be uniquely identified if $P(\omega 1) = P(\omega 2)$
(we cannot recover a unique θ even from an infinite amount of data!)

- $\theta = (\theta_1, \theta_2)$ and $\theta = (\theta_2, \theta_1)$ are two possible vectors that can be interchanged without affecting $P(x \mid \theta)$
- Identifiability can be a problem, we always assume that the densities we are dealing with are identifiable!

- **ML Estimates**
  - Suppose that we have a set $D = \{x_1, \ldots, x_n\}$ of n unlabeled samples drawn independently from the mixture densitv

$$p(x \mid \theta) = \sum_{j=1}^{c} p(x \mid \omega_j, \theta_j) P(\omega_j)$$

(θ is fixed but unknown!)

$$\hat{\theta} = \arg\max_{\theta} p(D \mid \theta) \text{ with } p(D \mid \theta) = \prod_{k=1}^{n} p(x_k \mid \theta)$$

The gradient of the log-likelihood is:

$$\nabla_{\theta_i} l = \sum_{k=1}^{n} P(\omega_i \mid x_k, \theta) \nabla_{\theta_i} \ln p(x_k \mid \omega_i, \theta_i)$$

Since the gradient must vanish at the value of θ_i that maximizes l ($l = \sum_{k=1}^{n} \ln p(x_k \mid \theta)$) therefore, the ML estimate $\hat{\theta}_i$ must satisfy the conditions

$$\sum_{k=1}^{n} P(\omega_i \mid x_k, \hat{\theta}) \nabla_{\theta_i} \ln p(x_k \mid \omega_i, \hat{\theta}_i) = 0 \quad (i = 1, \ldots, c)$$

By including the prior probabilities as unknown variables, we finally obtain:

$$\hat{P}(\omega_i) = \frac{1}{n} \sum \hat{P}(\omega_i \mid x_k, \hat{\theta})$$

$$\text{and } \sum_{k=1}^{n} \hat{P}(\omega_i \mid x_k, \hat{\theta}) \nabla_{\theta_i} \ln p(x_k \mid \omega_i, \hat{\theta}_i) = 0$$

$$\text{where : } \hat{P}(\omega_i \mid x_k, \hat{\theta}) = \frac{p(x_k \mid \omega_i, \hat{\theta}_i) \hat{P}(\omega_i)}{\sum_{j=1}^{c} p(x_k \mid \omega_j, \hat{\theta}_j) \hat{P}(\omega_j)}$$

- **Applications to Normal Mixtures**

$p(x \mid \omega_i, \theta_i) \sim N(\mu_i, \Sigma_i)$

Case	μ_i	Σ_i	$P(\omega_i)$	c
1	?	x	x	x
2	?	?	?	x
3	?	?	?	?

Case 1 = Simplest case

– **Case 1: Unknown mean vectors**

$\mu_i = \theta_i \quad \forall\, i = 1, \ldots, c$

$$\ln p(x \mid \omega_i, \mu_i) = -\ln\left[(2\pi)^{d/2}\left|\sum_i\right|^{1/2}\right] - \frac{1}{2}(x-\mu_i)^t \sum_i^{-1}(x-\mu_i)$$

ML estimate of $\mu = (\mu_i)$ is:

$$\hat{\mu}_i = \frac{\sum_{k=1}^{n} P(\omega_i \mid x_k, \hat{\mu}) x_k}{\sum_{k=1}^{n} P(\omega_i \mid x_k, \hat{\mu})} \qquad (1)$$

$P(\omega_i \mid x_k, \hat{\mu})$ **is the fraction of those samples**

having value x_k that come from the ith class, and $\hat{\mu}_i$ is the average of the samples coming from the ith class.

- Unfortunately, equation (1) does not give $\hat{\mu}_i$ explicitly
- However, if we have some way of obtaining good initial estimates $\hat{\mu}_i(0)$ for the unknown means, therefore equation (1) can be seen as an iterative process for improving the estimates

$$\hat{\mu}_i(j+1) = \frac{\sum_{k=1}^{n} P(\omega_i \mid x_k, \hat{\mu}(j)) x_k}{\sum_{k=1}^{n} P(\omega_i \mid x_k, \hat{\mu}(j))}$$

- This is a gradient ascent for maximizing the log-likelihood function
- Example:

  Consider the simple two-component one-dimensional normal mixture

$$p(x \mid \mu_1, \mu_2) = \frac{1}{3\sqrt{2\pi}} \exp\left[-\frac{1}{2}(x-\mu_1)^2\right] + \frac{2}{3\sqrt{2\pi}} \exp\left[-\frac{1}{2}(x-\mu_2)^2\right]$$

  (2 clusters!)

  Let's set $\mu_1 = -2$, $\mu_2 = 2$ and draw 25 samples sequentially from this mixture. The log-likelihood function is:

$$l(\mu_1, \mu_2) = \sum_{k=1}^{n} \ln p(x_k \mid \mu_1, \mu_2)$$

The maximum value of l occurs at:

$$\hat{\mu}_1 = -2.130 \text{ and } \hat{\mu}_2 = 1.668$$

(which are not far from the true values: μ_1 = -2 and μ_2 = +2)

There is another pair $\hat{\mu}_1 = 2.085$ and $\hat{\mu}_2 = -1.257$ which has almost the same height as can be seen from the following figure:

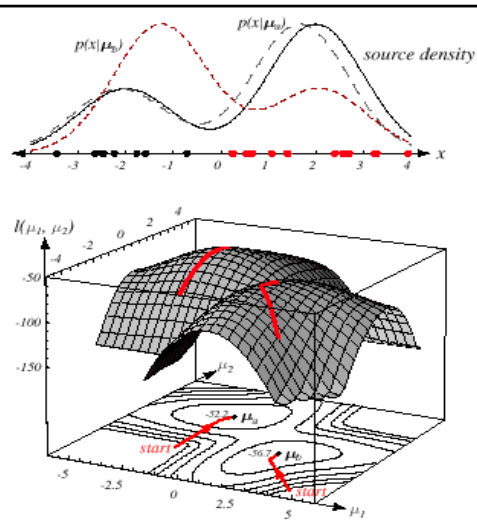


FIGURE 10.1. (Above) The source mixture density used to generate sample data, and two maximum-likelihood estimates based on the data in the table. (Bottom) Log-likelihood of a mixture model consisting of two univariate Gaussians as a function of their means, for the data in the table. Trajectories for the iterative maximum-likelihood estimation of the means of a two-Gaussian mixture model based on the data are shown as red lines. Two local optima (with log-likelihoods −52.2 and −56.7) correspond to the two density estimates shown above. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

- This mixture of normal densities is identifiable
- When the mixture density is not identifiable, the ML solution is not unique

– **Case 2: All parameters unknown**

- No constraints are placed on the covariance matrix

Let $p(x \mid \mu, \sigma^2)$ be the two-component normal mixture:

$$p(x \mid \mu,\sigma^2) = \frac{1}{2\sqrt{2\pi}.\sigma}\exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right] + \frac{1}{2\sqrt{2\pi}}\exp\left[-\frac{1}{2}x^2\right]$$

Suppose $\mu = x_1$, therefore:

$$p(x_1 \mid \mu,\sigma^2) \geq \frac{1}{2\sqrt{2\pi}\,\sigma} + \frac{1}{2\sqrt{2\pi}}\exp\left[-\frac{1}{2}x_1^2\right]$$

For the rest of the samples:

$$p(x_k \mid \mu,\sigma^2) \geq \frac{1}{2\sqrt{2\pi}}\exp\left[-\frac{1}{2}x_k^2\right]$$

Finally,

$$p(x_1,\ldots,x_n \mid \mu,\sigma^2) \geq \underbrace{\left\{\frac{1}{\sigma} + \exp\left[-\frac{1}{2}x_1^2\right]\right\}}_{\left(\text{this term} \to \infty \atop \sigma \to 0\right)} \frac{1}{(2\sqrt{2\pi})^n}\exp\left[-\frac{1}{2}\sum_{k=2}^{n}x_k^2\right]$$

The likelihood is therefore large and the maximum-likelihood solution becomes singular.

- **Adding an assumption**
  Consider the largest of the finite local maxima of the likelihood function and use the ML estimation.
  We obtain the following:

Iterative scheme

$$\hat{P}(\omega_i) = \frac{1}{n}\sum_{k=1}^{n}\hat{P}(\omega_i \mid x_k, \hat{\theta})$$

$$\hat{\mu}_i = \frac{\sum_{k=1}^{n}\hat{P}(\omega_i \mid x_k, \hat{\theta})x_k}{\sum_{k=1}^{n}\hat{P}(\omega_i \mid x_k, \hat{\theta})}$$

$$\hat{\Sigma}_i = \frac{\sum_{k=1}^{n}\hat{P}(\omega_i \mid x_k, \hat{\theta})(x_k - \hat{\mu}_i)(x_k - \hat{\mu}_i)^t}{\sum_{k=1}^{n}\hat{P}(\omega_i \mid x_k, \hat{\theta})}$$

Where:

$$\hat{P}(\omega_i \mid x_k, \hat{\theta}) = \frac{|\Sigma_i|^{-1/2}\exp\left[-\frac{1}{2}(x_k - \hat{\mu}_i)^t\hat{\Sigma}_i^{-1}(x_k - \hat{\mu}_i)\right]\hat{P}(\omega_i)}{\sum_{j=1}^{c}|\hat{\Sigma}_j|^{-1/2}\exp\left[-\frac{1}{2}(x_k - \hat{\mu}_j)^t\hat{\Sigma}_j^{-1}(x_k - \hat{\mu}_j)\right]\hat{P}(\omega_j)}$$

– K-Means Clustering

- **Goal: find the c mean vectors $\mu_1, \mu_2, \ldots, \mu_c$**
- **Replace the squared Mahalanobis distance**

$(x_k - \hat{\mu}_i)^t\hat{\Sigma}_i^{-1}(x_k - \hat{\mu}_i)$ by the squared Euclidean distance $\|x_k - \hat{\mu}_i\|^2$

- **Find the mean $\hat{\mu}_m$ nearest to x_k and approximate $\hat{P}(\omega_i \mid x_k, \hat{\theta})$ as: $\hat{P}(\omega_i \mid x_k, \theta) \cong \begin{cases} 1 & \text{if } i = m \\ 0 & \text{otherwise} \end{cases}$**

- Use the iterative scheme to find $\hat{\mu}_1, \hat{\mu}_2, \ldots, \hat{\mu}_c$
- if n is the known number of patterns and c the desired number of clusters, the k-means algorithm is:

```
Begin
    initialize n, c, μ1, μ2, …, μc(randomly
    selected)
          do classify n samples according to
                nearest μi
                recompute μi
          until no change in μI
    return μ1, μ2, …, μc
End
```

Exercise 2 p.594 (Textbook)