Intelligent Systems: Reasoning and Recognition

James L. Crowley

ENSIMAG 2 / MoSIG M1 Second Semester 2009/2010

Lesson 17 28 april 2010

Gaussian Mixture Models and Expectation-Maximization

Contents

Sources Bibliographiques :

"Neural Networks for Pattern Recognition", C. M. Bishop, Oxford Univ. Press, 1995.

"Pattern Recognition and Scene Analysis", R. E. Duda and P. E. Hart, Wiley, 1973.

Notation

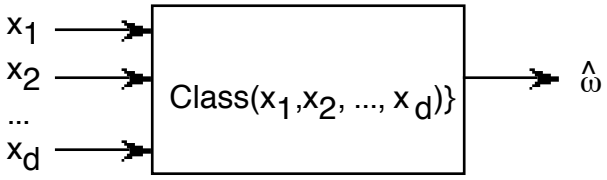
x	a variable
X	a random variable (unpredictable value)
$\vec{x}$	A vector of D variables.
$\vec{X}$	A vector of D random variables.
D	The number of dimensions for the vector $\vec{x}$ or $\vec{X}$
E	An observation. An event.
T_k	The class (tribe) k
k	Class index
K	Total number of classes
ω_k	The statement (assertion) that $E \in T_k$
$p(\omega_k) = p(E \in T_k)$	Probability that the observation E is a member of the class k. Note that $p(\omega_k)$ is lower case.
M_k	Number of examples for the class k. (think M = Mass)
M	Total number of examples. $M = \sum_{k=1}^{K} M_k$
$\{X_m^k\}$	A set of M_k examples for the class k. $\{X_m\} = \bigcup_{k=1,K} \{X_m^k\}$
$P(X)$	Probability density function for X
$P(\vec{X})$	Probability density function for $\vec{X}$
$P(\vec{X} \mid \omega_k)$	Probability density for $\vec{X}$ the class k. $\omega_k = E \in T_k$.
N	The number components in a Gaussian Mixture model

Gaussian Mixture model:

$$P(\vec{X}) = \sum_{n=1}^{M} \alpha_n \mathcal{N}(\vec{X}; \vec{\mu}_n, C_n)$$

Bayesian Classification (Reminder)

Our problem is to build a box that maps a set of features $\vec{X}$ from an Observation E into a class T_k from a set of K possible Classes.



Let ω_k be the proposition that the event belongs to class k: ω_k = E ∈ the class k

In order to minimize the number of mistakes, we choose the most probable ω_k

$$\hat{\omega}_k = \arg\max_k \left\{ \Pr(\omega_k \mid \vec{X}) \right\}$$

We will call on two tools for this:

1) Baye's Rule :

$$p(\omega_k \mid \vec{X}) = \frac{P(\vec{X} \mid \omega_k) p(\omega_k)}{P(\vec{X})}$$

2) Normal Density Functions

$$P(\vec{X} \mid \omega_k) = \frac{1}{(2\pi)^{\frac{D}{2}} \det(C_k)^{\frac{1}{2}}} e^{-\frac{1}{2}(\vec{X}-\vec{\mu}_k)^T C_k^{-1} (\vec{X}-\vec{\mu}_k)}$$

This week we will exam a method to estimate the Probability Density using a weighted sum of Normal (or Gaussian) Density functions

Gaussian Mixture Models

The "Central Limit Theorem" tells us that whenever an observation is the result of a sequence of N independent random events, the probability density of the features will tend toward a Normal or Gaussian density.

$$P(\vec{X}) = \mathcal{N}(\vec{X};\vec{\mu},C) = \frac{1}{(2\pi)^{\frac{D}{2}}\det(C)^{\frac{1}{2}}} e^{-\frac{1}{2}(\vec{X}-\vec{\mu})^T C^{-1}(\vec{X}-\vec{\mu})}$$

Unfortunately, this hypothesis does not always apply. A common case occurs when the event may come from one of a set of different "sources", each with its own density function.

In this case, the probability density is better represented as a weighted sum of normal densities.

$$P(\vec{X}) = \sum_{n=1}^{M} \alpha_n \mathcal{N}(\vec{X};\vec{\mu}_n,C_n)$$

Each normal density results from a different source. We can see the $\{\alpha_n\}$ as the relative probabilities for a set of independent "sources" for the event. The α_n coefficients represent the relative probability that event came from source "n".

$$\alpha_n = p(E \leftarrow Source(n))$$

Thus $$\sum_{n=1}^{N} \alpha_n = 1$$

Such a sum is referred to as a Gaussian Mixture Model. It can also be used to represent density functions where the Central Limit theorem does not apply or that have more complex forms. It can also be used to discover a set of subclasses within a global class.

It is sometimes convenient to group the parameters for each source into a single vector:

$$\vec{v}_n = (\alpha_n, \vec{\mu}_n, C_n)$$

For a feature vector of D dimensions, $\vec{v}_n$ has P = 1 + D + D(D+1)/2 coefficients.

The complete set of parameters is a vector with N·P coefficients.

$$\vec{v} = (\vec{v}_1, \vec{v}_2, ..., \vec{v}_n)$$

To estimate the $\{\alpha_n\}$ parameters we need the parameters $\{\mu_n, C_n\}$.
To estimate $\{\mu_n, C_n\}$ we need $\{\alpha_n\}$.
This leads to an iterative two-step process in which we alternately estimate $\{\mu_n, C_n\}$ and $\{\alpha_n\}$

To do this, we construct a table, h(m, n)

h(m, n) = Pr{the event E_m is from source n}

The iterative algorithm for this estimation is called EM: Expectation Maximisation.

A Quick Sketch of the Expectation Maximisation Algorithm

To illustrate the algorithm, let us consider a case where D=1.
Thus $\vec{v} = \{\alpha_n, \mu_n, \sigma_n^2\}$

We will estimate $\vec{v}$ for N Gaussians from a training set of M events $\{X_m\}$.
We will iteratively estimate $\vec{v}$. The i^{th} estimate will be $\vec{v}^{\,i)}$

We start by a first, initial guess $\nu^{(0)}$ and let i=0

Expectation step (E):
Calculate the table $h(m,n)^{(i)}$ using the training data.

$$h(m, n)^{(i)} = p(h_m = n \mid X_1, X_2, ..., X_M, \nu^{(i)})$$

$$h(m, n)^{(i)} = \frac{\alpha_n^{(i)} \mathcal{N}(X_m; \mu_n^{(i)}, \sigma_n^{(i)})}{\sum_{j=1}^{N} \alpha_j^{(i)} \mathcal{N}(X_m; \mu_j^{(i)}, \sigma_j^{(i)})}$$

Maximization Step: (M)
Calculate $\nu^{(i+1)}$ using $p(h_m \mid X_1, X_2, ..., X_M, \nu^{(i)})$

$$S_n^{(i+1)} = \sum_{m=1}^{M} h(m,n)^{(i)}$$

$$\alpha_n^{(i+1)} = \frac{1}{M} S_n^{(i+1)} = \frac{1}{M} \sum_{m=1}^{M} h(m,n)^{(i)}$$

$$\mu_n^{(i+1)} = \frac{1}{S_n^{(i+1)}} \sum_{m=1}^{M} h(m,n)^{(i)} X_m$$

$$\sigma^2{}_n^{(i+1)} = \frac{1}{S_n^{(i+1)}} \sum_{m=1}^{M} h(m,n)^{(i)} (X_m - \mu_n^{(i+1)})^2$$

Quit when the answer stops improving.
To properly derive EM, we need the notion of Maximum Likelihood.

Maximum Likelihood Estimation.

To keep the discussion simple, consider D=1 and assume that we have a training set $\{X_m\}$. We wish to estimate $P(x) = \mathcal{N}(x; \mu,\sigma)$.

For a normal density with D=1,

$$\vec{\nu} = (\mu, \sigma)$$

Our best estimate of $\nu = (\mu, \sigma)$ is that which maximizes the probabilty for the training data $\{X_m\}$

Let us define the Likelihood $L(\nu \mid X_1, X_2, ..., X_M)$
Assuming that the X_m are independent,

$$P(X_1, X_2 \mid \vec{\nu}) = P(X_1 \mid \vec{\nu}) \cdot P(X_2 \mid \vec{\nu})$$

in general for M events:

$$P(X_1, X_2, ..., X_M \mid \vec{\nu}) = P(\{X_m\} \mid \vec{\nu}) = \prod_{m=1}^{M} P(X_m \mid \vec{\nu})$$

We define the likelihood of v given $\{X_m\}$ as

$$L(\vec{\nu} \mid \{X_m\}) = P(\{X_m\} \mid \vec{\nu}) = \prod_{m=1}^{M} P(X_m \mid \vec{\nu})$$

Our objective is to estimate $\hat{\nu}$ to maximise $L(\hat{\nu} \mid \{X_m\})$

$$\hat{\nu} = \arg\max_{\nu}\{L(\hat{\nu} \mid \{X_m\})\} = \arg\max_{\nu}\{\prod_{m=1}^{M} P(X_m \mid \hat{\nu})\}$$

Because we will use $P(\vec{X}) = \mathcal{N}(\vec{X}; \vec{\mu}, C) = \frac{1}{(2\pi)^{\frac{D}{2}} \det(C)^{\frac{1}{2}}} e^{-\frac{1}{2}(\vec{X}-\vec{\mu})^T C^{-1} (\vec{X}-\vec{\mu})}$

it is easier to work with the Log likelihood:

$$\mathcal{L}(v) = Log\{L(\hat{\nu} \mid \{X_m\}) = Log\{P(\{X_m\} \mid \hat{\nu})\} = \sum_{m=1}^{M} Log\{P(X_m \mid \hat{\nu})\}$$

$P(X_m \mid \hat{\nu})$ is a simple Normal, then it is sufficient to maximize the sum.

MLE for a Univariate Gaussian Density functions

For D=1, $\mathcal{N}$(X; μ,σ) the paremeter vector is $\vec{v}$ = (μ, σ)

To estimate μ,σ using MLE, define the log likelihood.

$$\mathcal{L}(\vec{v}) = Log\{P(X_m \mid \vec{v})\} = -\frac{1}{2} Log\{2\pi\sigma^2\} - \frac{1}{2\sigma^2}(X_m - \mu)^2$$

The maximum Log Likelihood occurs when the derivative is zero.

$$\frac{\partial l(v)}{\partial \mu} = \sum_{m=1}^{M} \frac{1}{\sigma^2}(X_m - \mu) = 0$$

$$\frac{\partial l(\vec{v})}{\partial \sigma^2} = -\frac{1}{2\sigma^2} + \frac{(X_m - \mu)^2}{2\sigma^4} = 0$$

We formulate this as the gradient

$$\nabla_{\mu,\sigma}\mathcal{L}(\vec{v}) = \begin{pmatrix} \frac{\partial l(v)}{\partial \mu} \\ \frac{\partial l(\vec{v})}{\partial \sigma^2} \end{pmatrix} = \begin{pmatrix} \sum_{m=1}^{M} \frac{1}{\sigma^2}(X_m - \mu) \\ -\frac{1}{2\sigma^2} + \frac{(X_m - \mu)^2}{2\sigma^4} \end{pmatrix} = 0$$

with a little algebra:

$$\sum_{m=1}^{M} \frac{1}{\sigma^2}(X_m - \hat{\mu}) = 0.$$

$$\frac{1}{\sigma^2} \sum_{m=1}^{M} X_m = \frac{1}{\sigma^2} \sum_{m=1}^{M} \hat{\mu}$$

$$\sum_{m=1}^{M} X_m = M\,\hat{\mu}$$

$$\hat{\mu} = \frac{1}{M} \sum_{m=1}^{M} X_m$$

In the same way

$$\frac{\partial l(\nu)}{\partial \sigma^2} = -\frac{1}{2\sigma^2} + \frac{(X_m - \mu)^2}{2\sigma^4} = 0$$

$$\sum_{m=1}^{M} -\frac{1}{2\sigma^2} + \frac{(X_m - \mu)^2}{2\sigma^4} = 0$$

$$\sum_{m=1}^{M} \frac{1}{2\sigma^2} = \sum_{m=1}^{M} \frac{(X_m - \mu)^2}{2\sigma^4}$$

$$\frac{1}{2\sigma^2} \sum_{m=1}^{M} 1 = \frac{1}{2\sigma^2} \sum_{m=1}^{M} \frac{(X_m - \mu)^2}{\sigma^2}$$

$$\sum_{m=1}^{M} 1 = \sum_{m=1}^{M} \frac{(X_m - \mu)^2}{\sigma^2}$$

$$M = \frac{1}{\sigma^2} \sum_{m=1}^{M} (X_m - \mu)^2 \quad \Longrightarrow \sigma^2 = \frac{1}{M} \sum_{m=1}^{M} (X_m - \mu)^2$$