

Mixture density model

We assume that density is generated from k distributions:

$$p(x) = \sum_{j=1}^k w_j \varphi_j(x, \theta_j), \quad \sum_{j=1}^k w_j = 1, \quad w_j \geq 0,$$

Where:

w_j are weights of each distribution (we can call them “components”).

$\varphi_j(x, \theta_j)$ are densities of components (θ_j can be a vector of parameters).

We can interpret that density model as a two-step data generation process:

To generate data point x , first choose component j with probability w_j , then generate one observation from respective density $p(x | j) \equiv \varphi_j(x, \theta_j)$.

To solve we explicitly write down maximum likelihood:

$$\left\{ \begin{array}{l} L(w, \theta) = \ln \prod_{i=1}^{\ell} p(x_i) = \sum_{i=1}^{\ell} \ln \sum_{j=1}^k w_j \varphi_j(x_i, \theta_j) \rightarrow \max_{w, \theta} \\ \sum_{j=1}^k w_j = 1, \quad w_j \geq 0 \end{array} \right.$$

There is no general closed form solution, but there is general form of iterative EM-algorithm that shows how to solve that problem:

Theorem (necessary condition of extremum):

Vector $(w_j, \theta_j)_{j=1}^k$ gives local extremum to $L(w, \theta)$ if it satisfies the following system of iterative equations (that also depends on auxiliary variables g_{ij}):

$$\text{E-step: } g_{ij} = \frac{w_j \varphi_j(x_i, \theta_j)}{\sum_{s=1}^k w_s \varphi_j(x_i, \theta_s)}, \quad i = 1, \dots, \ell, \quad j = 1, \dots, k;$$

M-step:

$$\theta_j = \arg \max_{\theta} \sum_{i=1}^{\ell} g_{ij} \ln \varphi_j(x_i, \theta), \quad j = 1, \dots, k;$$

$$w_j = \frac{1}{\ell} \sum_{i=1}^{\ell} g_{ij}, \quad j = 1, \dots, k.$$

Algorithm for density separation

Input:

Dataset: $X = \{x_1, \dots, x_{\ell}\}$

Number of densities: k

Density shapes: $\varphi_1, \dots, \varphi_k$

Output:

Density parameters: $\theta_1, \dots, \theta_k$

Weights: $w_1, \dots, w_k$

Initialization:

Set: $w_j = \frac{1}{k}$

Parameters θ_j to any reasonable default values

Algorithm:

Repeat E and M steps:

$$E\text{-step: } g_{ij} := \frac{w_j \varphi_j(x_i, \theta_j)}{\sum_{s=1}^k w_s \varphi_j(x_i, \theta_s)}, \quad i = 1, \dots, \ell, \quad j = 1, \dots, k;$$

M-step:

$$\theta_j := \arg \max_{\theta} \sum_{i=1}^{\ell} g_{ij} \ln \varphi_j(x_i, \theta), \quad j = 1, \dots, k;$$

$$w_j := \frac{1}{\ell} \sum_{i=1}^{\ell} g_{ij}, \quad j = 1, \dots, k.$$

Until convergence θ_j, w_j and g_{ij} stabilize.

Example 1:

We model time until next session for a user of a website or an app. Often such data has a bimodal distribution with exponential density around zero and normal density around 24 hours. That clearly describes two processes:

Exponential: "Customer is on the website / app"

Normal: "Customer comes back next day around the same time"

Assumption:

For each customer data comes from mixture of exponential and normal densities.

$$p(x) = w_1 \lambda e^{-\lambda x} + w_2 \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Repeat E and M steps:

E-step:

$$g_{i1} := \frac{w_1 \lambda e^{-\lambda x_i}}{w_1 \lambda e^{-\lambda x_i} + w_2 \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}}}$$

$$g_{i2} := \frac{w_2 \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}}}{w_1 \lambda e^{-\lambda x_i} + w_2 \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}}}$$

M-step:

$$\lambda := \frac{\sum_{i=1}^{\ell} g_{i1}}{\sum_{i=1}^{\ell} g_{i1} x_i}$$

$$\mu := \frac{\sum_{i=1}^{\ell} g_{i2} x_i}{\sum_{i=1}^{\ell} g_{i2}}$$

$$\sigma^2 := \frac{\sum_{i=1}^{\ell} g_{i2} (x_i - \mu)^2}{\sum_{i=1}^{\ell} g_{i2}}$$

$$w_j := \frac{1}{\ell} \sum_{i=1}^{\ell} g_{ij}$$

Example 2:

Customers purchase products from m fixed categories. We observe a frequency matrix and notice that there are structural preferences of customers for specific groups of products. We assume then that whole dataset is described as mixture as multinomials. Each multinomial defines as set of customers with similar preferences.

Mixture of multinomials:

$x_i = \langle n_1^{(i)}, \dots, n_m^{(i)} \rangle$, which is vector of observed frequencies of dimension m

Dataset is composed of x_i put in a matrix of size (l, m) (l rows, m columns).

Multinomial density for each component j is described by a vector of probabilities $\theta_j = \langle p_1^{(j)}, \dots, p_m^{(j)} \rangle$:

$$\varphi(x_i, \theta_j) = A(n_1^{(i)}, \dots, n_m^{(i)}) [p_1^{(j)}]^{n_1^{(i)}} \cdot \dots \cdot [p_m^{(j)}]^{n_m^{(i)}}$$

Where $A(n_1^{(i)}, \dots, n_m^{(i)})$ is a normalizing constant we can luckily ignore.

Our goal is to find θ_j for $j = 1, \dots, k$ as we assume there are k components (that will be an input to the algorithm).

Next we solve explicitly argmax part (we ignore $A(\dots)$):

$$\theta_j := \arg \max_{\theta} \sum_{i=1}^{\ell} g_{ij} \ln \left([p_1^{(j)}]^{n_1^{(i)}} \cdot \dots \cdot [p_m^{(j)}]^{n_m^{(i)}} \right)$$

We put g_{ij} into logarithms

$$\theta_j := \arg \max_{\theta} \sum_{i=1}^{\ell} \ln \left([p_1^{(j)}]^{g_{ij} n_1^{(i)}} \cdot \dots \cdot [p_m^{(j)}]^{g_{ij} n_m^{(i)}} \right)$$

We translate sum of logarithms into product:

$$\theta_j := \arg \max_{\theta} \ln \left([p_1^{(j)}]^{\sum g_{ij} n_1^{(i)}} \cdot \dots \cdot [p_m^{(j)}]^{\sum g_{ij} n_m^{(i)}} \right)$$

And get rid of logarithm

$$\theta_j := \arg \max_{\theta} [p_1^{(j)}]^{\sum g_{ij} n_1^{(i)}} \cdot \dots \cdot [p_m^{(j)}]^{\sum g_{ij} n_m^{(i)}}$$

That is equivalent to find mode of Dirichlet distribution, which has a closed form solution (and can be easily done using Lagrange multipliers or checked on Wikipedia):

$$p_r^{(j)} = \frac{\sum_{i=1}^l g_{ij} n_r^{(i)}}{\sum_{s=1}^m \sum_{i=1}^l g_{ij} n_s^{(i)}}$$

Where $p_r^{(j)}$ is r th component of $\theta_j = \langle p_1^{(j)}, \dots, p_m^{(j)} \rangle$

Full algorithm for multinomials:

Repeat E and M steps:

E-step:

$$g_{ij} := \frac{w_j \varphi(x_i, \theta_j)}{\sum_{s=1}^k w_s \varphi(x_i, \theta_s)}, \quad i = 1, \dots, \ell, \quad j = 1, \dots, k;$$

$$\varphi(x_i, \theta_j) = A(n_1^{(i)}, \dots, n_m^{(i)}) [p_1^{(j)}]^{n_1^{(i)}} \dots [p_m^{(j)}]^{n_m^{(i)}}$$

$$\theta_j = \langle p_1^{(j)}, \dots, p_m^{(j)} \rangle$$

$$x_i = \langle n_1^{(i)}, \dots, n_m^{(i)} \rangle$$

M-step:

$$p_r^{(j)} = \frac{\sum_{i=1}^l g_{ij} n_r^{(i)}}{\sum_{s=1}^m \sum_{i=1}^l g_{ij} n_s^{(i)}}, \quad r = 1, \dots, m; \quad j = 1, \dots, k;$$

$$w_j := \frac{1}{\ell} \sum_{i=1}^{\ell} g_{ij}, \quad j = 1, \dots, k.$$

Until convergence: θ_j, w_j and g_{ij} stabilize.