GMM

16 November 2024 08:27

Gaussian Mixture Model

clustering — soft / hard

mixtures are a PROBABILISTICALLY GROUNDED way of soft clustering.

each cluster is a gaussian model.

A gaussian mixture is a LINEAR SUPERPOSITION OF MULTIPLE GAUSSIANS (we can assume a weight for each gaussian)

Say we've k gaussians w/ a weight π_k.

$$p(x) = \sum_{k=1}^{K} \pi_k \cdot N(x \mid \mu_k, \Sigma_k)$$ when $0 \le \pi_k \le 1$, $\sum \pi_k = 1$

$$\int p(x) = \int \sum \pi_k N(x \mid \mu_k, \Sigma_k) = \sum \pi_k \int N(x \mid \mu_k, \Sigma_k) = (1)\cdot(1) = 1$$ ($N(x \mid \mu_k, \Sigma_k)$ is the PDF)

are unimodal dist. not always practical.

if we take log likelihood:

$$\ln p(X \mid \mu, \Sigma, \pi) = \sum_{n=1}^{N} \ln p(x_n) = \sum_{n=1}^{N} \ln \left[\sum_{k=1}^{K} \pi_k \cdot N(x \mid \mu_k, \Sigma_k) \right]$$ X

No closed form solution.
⇒ MLE will not work.

EM works instead!

EM for GMM, where each G is 1D.

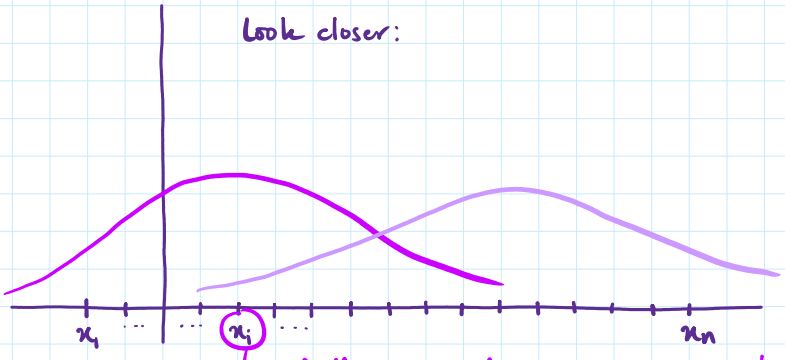
① Init step: π_j (weight), μ_j, σ_j^2 (defining characteristics of gaussian) for each gaussian j.

② E step: Calculate posterior distribution of latent variable (probability that this gaussian generated this data)

③ M step: Update parameter for each gaussian — π_j, μ_j, σ_j^2

repeat till convergence.

Look closer:



$x_1, x_2 \ldots x_n$ (observations)
$k = 2$ (2 gaussians)

If the source of x_i is known, estimating parameters is easy.
What if the source isn't known?
Then it's a latent variable!

Let the two gaussians be (μ_a, σ_a) and (μ_b, σ_b)
we'll look @ b but it's the same for a.

$$P(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left(\frac{-(x_i - \mu_b)^2}{2\sigma_b^2}\right)$$

We don't know the latent var. ⇒ we need (μ_b, σ_b^2) to get $P(b \mid x_i)$

$$P(b \mid x_i) = \frac{P(x_i \mid b) \cdot P(b)}{P(x_i \mid b) \cdot P(b) + P(x_i \mid a) \cdot P(a)}$$

So we take an EM based approach.

① Init step: Start w/ 2 random placings of gaussians — (μ_a, σ_a^2) and (μ_b, σ_b^2)

So we have an EM based approach:

① Init step: Start w/ 2 random placings of gaussians — (μ_a, σ_a^2) and (μ_b, σ_b^2)

do until convergence:

② Σ-Step: $p(b|x_i)$ — from b^0; $p(a|x_i)$ — from a^0] — generate new dataset using soft weight.

③ M-Step: adjust (μ_a, σ_a^2) & (μ_b, σ_b^2) till it fits

Zoom in.

Consider the joint probability $p(X = \bar{x}, \underbrace{z = k}_{\text{latent variable}})$

$p(\bar{x}, k)$ — probability of input vector $\bar{x}$ occurring together w/ class k.

Using bayes, $P(\bar{x}, k) = p(\bar{x}|k)\,p(k)$

↓

PDF of k^{th} gaussian component.

$p(\bar{x}|k) = N(x|\mu_k, \Sigma_k) \quad k \in [1, k]$

Let $p(k) = \pi_k \quad k \in [1, k]$

$\pi_k \approx \frac{N_k}{N}$ → no. of instances belonging to k / → total no. of instances

So $p(\bar{x}, k) = p(k)\,p(\bar{x}|k) = \pi_k \cdot N(x; \mu_k, \Sigma_k)$

So, from this, we get marginal probability of x:

$$p(x) = \sum p(x, k) = \sum \pi_k N(x; \mu_k, \Sigma_k)$$ — linear superposition of multiple gaussians.

What do we take from this?

① GMM = weighted sum of K gaussian components.

② π_k is the selection probability of the k^{th} gaussian

③ Each of the K gaussians models a single class.

↓

k^{th} gaussian $N(x; \mu_k, \sigma_k)$ can be modeled as cond^n prob $p(x|k)$ — probability of x being in k^{th} sub class.

④ The sum of all joint subclass probabilities = marginal probability $p(x)$ of data value (x)

Classification via GMM

→ Set of unlabeled input X is given

↓

fit a GMM for all k classes.

↓

given an arbitrary x, compare $P(k|x)$ — probability that it's class k given x.

↓

Value of k yielding highest $p(k|x)$ is the class corresponding to x.

$$p(k|x) = \frac{P(x|k)}{P(x)} = \frac{\pi_k N(x; \mu_k, \sigma_k)}{\sum_{i=1}^{k} \pi_i \cdot N(x; \mu_i, \sigma_i)}$$

Whole process is super easy if we know GMM params.

↓

given x, just classify by assigning it to cluster k] — that yields max value of $p(z = k | X = x)$

assigning it to cluster k] of $p(z=k|x=x)$

MLE of GMM Parameters.

A GMM is fully described by parameter set:

$$\theta = \{\pi_k, \mu_k, \Sigma_k \ \forall \ k \in \{1 \dots k\}\}$$

(μ_k: mean of kth gaussian; π_k: weight of kth gaussian; Σ_k: variance of kth gaussian)

* we're not estimating k for the MLE.

Now, when we had a simple gaussian:

① we took joint log likelihood of all training data given by gaussian PDF.

② took gradient on both sides & set it to 0.

* a closed form soln has unknowns on LHS and knowns on RHS.

But that doesn't work w/ the GMM — we get an equation that has **NO closed form soln**

So we go for an ITERATIVE APPROXIMATION:

Let $x^{(1)}, x^{(2)}, x^{(3)} \dots x^{(n)}$ be the set of observed data points.

$$\underbrace{p(x^{(i)}|\theta)}_{\text{likelihood}} = \sum_{k=1}^{K} \pi_k \cdot N(x^{(i)}; \mu_k, \Sigma_k)$$

↳ we'll refer to it as $p(x^{(i)})$

All datapoints $(x^{(i)})$ are independent.

⇒ Joint probability = $p(x^{(1)}) \cdot p(x^{(2)}) \cdot p(x^{(3)}) \cdots p(x^{(n)})$

⇒ Joint loglikelihood = $\log\left(p(x^{(1)}) \cdot p(x^{(2)}) \cdot p(x^{(3)}) \cdots p(x^{(n)})\right)$

$$= \sum_{i=1}^{n} \log\left(p(x^{(i)})\right)$$

$$= \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \pi_k \cdot N(x^{(i)}; \mu_k, \Sigma_k)\right]$$

logarithm of a sum!

⇒ We've to identify $\mu_1, \Sigma_1, \mu_2, \Sigma_2 \dots$ that will maximize the log joint likelihood.] take gradient wrt param & solve.

Say we take μ_1 for gradient:

$$\nabla_{\mu_1} \log\left(p(x^{(1)}) \cdot p(x^{(2)}) \cdots p(x^{(n)})\right) = 0$$

$$\nabla_{\mu_1} \sum_{i=1}^{n} \log\left(p(x^{(i)})\right) = 0$$

$$\nabla_{\mu_1} \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \pi_k \cdot N(x^{(i)}; \mu_k, \Sigma_k)\right] = 0$$

grad is a linear operator so move it in.

$$\sum_{i=1}^{n} \nabla_{\mu_1} \log\left[\sum \pi_k \cdot N(x^{(i)}; \mu_k, \Sigma_k)\right]$$

$\frac{d}{dx} \log f(x) = \frac{1}{f(x)} \frac{\partial f}{\partial x}$

$$\sum_{\dots}^{n} \frac{\nabla_{\mu_1} \sum_{k=1}^{K} \pi_k \cdot N(x^{(i)}; \mu_k, \Sigma_k)}{\sum_{k=1}^{K} \pi_k \cdot N(x^{(i)}; \mu_k, \Sigma_k)} = 0$$

$\frac{\partial}{\partial x} \log f(x) = \frac{1}{f(x)} \frac{\partial f}{\partial x}$

$$\sum_{i=1} \frac{\nabla_{\mu_1} \sum_{k=1}^{K} \pi_k \cdot N(x^{(i)}; \mu_k, \Sigma_k)}{\sum_{k=1}^{K} \pi_k \cdot N(x^{(i)}; \mu_k, \Sigma_k)} = 0$$

If x_1, x_2 are independent, $\partial x_2 / \partial x_1 = 0$.
So the gradient wrt μ_1 will be 0 for all $k \neq 1$.