Mixture of Gaussian

Le Song

Machine Learning I
CSE 6740, Fall 2013
Why do we need density estimation?

- Learn the “shape” of the data cloud

- Assess the likelihood of seeing a particular data point
  - Is this a typical data point? (high density value)
  - Is this an abnormal data point / outlier? (low density value)

- Building block for more sophisticated learning algorithms
  - Classification, regression, graphical models ...
Parametric models

- Models which can be described by a fixed number of parameters

- Discrete case: eg. Bernoulli distribution
  \[ P(x|\theta) = \theta^x (1 - \theta)^{1-x} \]
  one parameter, \( \theta \in [0,1] \), which generate a family of models, \( \mathcal{F} = \{ P(x|\theta) | \theta \in [0,1] \} \),

- Continuous case: eg. Gaussian distribution in \( \mathbb{R}^n \)
  \[ p(x|\mu, \Sigma) = \frac{1}{\frac{1}{2}(2\pi)^{n/2} |\Sigma|} \exp \left( -\frac{1}{2} (x - \mu)^\top \Sigma^{-1} (x - \mu) \right) \]
  Two sets of parameters \( \{\mu, \Sigma\} \), which again generate a family of models, \( \mathcal{F} = \{ p(x|\mu, \Sigma) | \mu \in \mathbb{R}^n, \Sigma \in \mathbb{R}^{n\times n} and PSD \} \),
Nonparametric models

- Models which can not be described by a fixed number of parameters
  - “Nonparametric” does not mean there are no parameters
  - In fact, one can think of there are many many parameters

- Eg. Smooth density
  \[ F = \left\{ p(x) | p(x) \geq 0, \int_{\Omega} p(x)dx = 1, \int_{\Omega} (p''(x))^2 dx < \infty \right\} \]

- Eg. Histogram

- Eg. Kernel density estimator
Estimation of parametric models

- A very popular estimator is the **maximum likelihood estimator (MLE)**, which is simple and has good statistical properties.

- Assume that $m$ data points $\mathcal{D} = \{x^1, x^2, \ldots, x^m\}$ drawn independently and identically (iid) from some distribution $P^*(x)$.

- Want to fit the data with a model $P(x|\theta)$ with parameter $\theta$.

$$
\theta = \arg\max_\theta \log P(\mathcal{D}|\theta) \\
= \arg\max_\theta \log \prod_{i=1}^m P(x^i|\theta)
$$

**convex in many cases**
Estimating Gaussian distribution

- Gaussian distribution in $R$
  \[ p(x|\mu, \sigma) = \frac{1}{(2\pi)^\frac{p}{2} \sigma} \exp \left( -\frac{1}{2\sigma^2} (x - \mu)^2 \right) \]

- Need to estimate two sets of parameters $\mu, \sigma$

- Given $m$ iid samples
  \[ \mathcal{D} = \{x^1, x^2, ..., x^m\}, x^i \in R \]

- Likelihood of one data point:
  \[ p(x^i|\mu, \sigma) \propto \exp \left( -\frac{1}{2\sigma^2} (x^i - \mu)^2 \right) \]
MLE for Gaussian distribution

\[ l(\mu, \sigma; D) = -\frac{m}{2} \log 2\pi - \frac{m}{2} \log \sigma^2 - \sum_{i=1}^{m} \frac{(x^i - \mu)^2}{2\sigma^2} \]

\[
\frac{\partial l}{\partial \mu} = \frac{1}{\sigma^2} \sum_{i=1}^{m} (x^i - \mu) = 0
\]

\[
\Rightarrow \sum_{i} x^i = m\mu \Rightarrow \mu = \frac{1}{m} \sum_{i=1}^{m} x^i
\]

\[
\frac{\partial l}{\partial \sigma^2} = -\frac{m}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{i} (x^i - \mu)^2 = 0
\]

\[
\Rightarrow \sum_{i} (x^i - \mu)^2 = m\sigma^2 \Rightarrow \sigma^2 = \frac{1}{m} \sum_{i=1}^{m} (x^i - \mu)^2
\]
1-D Histogram

- Given $m$ iid samples $\mathcal{D} = \{x^1, x^2, ... x^m\}, x^i \in [0,1)$ from $P^*(x)$

- Split $[0,1)$ evenly into $n$ bins, $B_1 = \left[0, \frac{1}{n}\right), B_2 = \left[\frac{1}{n}, \frac{2}{n}\right), ... B_n = \left[\frac{n-1}{n}, 1\right)$, and Count the number of points, $c_1$ within $B_1$, $c_2$ within $B_2$...

- For a new test point $x$,

  $$p(x) = \sum_{j=1}^{n} \frac{nc_j}{m} I(x \in B_j)$$
Higher dimensional histogram

- Given $m$ iid samples $\mathcal{D} = \{x^1, x^2, \ldots x^m\}$, $x^i \in [0,1)^d$

- Split $[0,1)^d$ evenly into $n^d$ bins

\[
\begin{align*}
B_1 &= \left[0, \frac{1}{n}\right) \times \left[0, \frac{1}{n}\right) \ldots \times \left[0, \frac{1}{n}\right), \\
B_2 &= \left[\frac{1}{n}, \frac{2}{n}\right) \times \left[0, \frac{1}{n}\right) \ldots \times \left[0, \frac{1}{n}\right), \\
B_{n^d} &= \left[\frac{n-1}{n}, 1\right) \times \left[\frac{n-1}{n}, 1\right) \ldots \times \left[\frac{n-1}{n}, 1\right)
\end{align*}
\]

- Problem: too many bin! Not good for high dimensional data
  - If $n^d$ is larger than $m$, most bins are empty
  - Eg. $n = 10, d = 6$, you need approximately 1 million data points
1D-Kernel density estimation

Kernel density estimator

$$p(x) = \frac{1}{m} \sum_{i}^{m} \frac{1}{h} K \left( \frac{x^i - x}{h} \right)$$

Kernel function
- $K(u) \geq 0$,
- $\int K(u) du = 1$,
- $\int uK(u) = 0$,
- $\int u^2 K(u) du \leq \infty$

An example: Gaussian kernel

$$K(u) = \frac{1}{\sqrt{2\pi}} e^{-u^2/2}$$
Smoothing kernel functions

- An example: Gaussian kernel $K(u) = \frac{1}{\sqrt{2\pi}} e^{u^2/2}$

$K(u) = \frac{3}{4} (1 - u^2) I(|u| \leq 1)$

$K(u) = \frac{\pi}{4} \cos\left(\frac{\pi}{2} u\right) I(|u| \leq 1)$
Effect of the kernel bandwidth $h$
Two-dimensional example
Multimodal distributions

- What if we know the data consists of a few Gaussians
- What if we want to fit parametric models
Gaussian Mixture model

- A density model $p(X)$ may be multi-modal: model it as a mixture of uni-modal distributions (e.g. Gaussians)

- Consider a mixture of $K$ Gaussians
  
  $$ p(X) = \sum_{k=1}^{K} \pi_k \mathcal{N}(X | \mu_k, \Sigma_k) $$

- Learn $\pi_k \in (0,1), \mu_k, \Sigma_k$;
Image a generative process for data points

- For each data point $x^i$:
  - Randomly choose a mixture component, $z^i = \{1, 2, \ldots, K\}$, with probability $\pi_{z^i}$
  - Then sample the actual value of $x^i$ from a Gaussian distribution $\mathcal{N}(x | \mu_{z^i}, \Sigma_{z^i})$

- Joint distribution over $p(x, z)$
  $$p(x, z) = \pi_z \mathcal{N}(x | \mu_z, \Sigma_z)$$

- Marginal distribution $p(x)$
  $$p(x) = \sum_{z=1}^{K} p(x, z) = \sum_{z=1}^{K} p(x | z)p(z)$$
Why is learning hard?

- With latent variables \( z \), likelihood of the data becomes

\[
    l(\theta; D) = \log \prod_{i=1}^{m} \left( \sum_{z^i=1}^{K} p(x^i, z^i | \theta) \right)
\]

\[
= \log \prod_{i=1}^{m} \left( \sum_{z^i=1}^{K} p(x^i | \mu_{z^i}, \Sigma_{z^i}) p(z^i | \pi) \right)
\]

Nonconvex
Difficult!
EM algorithm

- A local search algorithm for finding parameters to maximize

\[ l(\theta; D) = \log \prod_{i=1}^{m} \left( \sum_{z_i=1}^{K} p(x^i, z^i | \theta) \right) \]

- Randomly initialize the parameters \( \pi_k \in (0,1), \mu_k, \Sigma_k \), and iterate the following two steps until convergence:
  
  - Expectation step (E-step)
  
  - Maximization step (M-step)
**E-step**

- **Expectation step**: computing the expected value of the sufficient statistics of the hidden variables ($z$) given current estimate of the parameters ($\pi, \mu, \Sigma$)

\[
\tau_k^i = q(z_k^i = 1|D, \mu, \Sigma) = \frac{\pi_k N(x_i | \mu_k, \Sigma_k)}{\sum_k \pi_k N(x_i | \mu_k, \Sigma_k)}
\]
M-step

- Maximization step: compute the parameters under current results of the expected complete log-likelihood

\[
\pi_k = \frac{\sum_i \tau_k^i}{m}
\]

\[
\mu_k = \frac{\sum_i \tau_k^i x^i}{\sum_i \tau_k^i}
\]

\[
\Sigma_k = \frac{\sum_i \tau_k^i (x^i - \mu_k)(x^i - \mu_k)^T}{\sum_i \tau_k^i}
\]
Expectation-Maximization Iterations

(a) $L = 1$

(b) $L = 2$

(c) $L = 5$

(d) $L = 20$

(e) $L = 5$

(f) $L = 20$
K-means vs EM for Gaussian mixture

• The EM algorithm for mixture of Gaussian is like a soft clustering algorithm

• K-means:
  
  “E-step”, we do hard assignment:
  $$z^i = \text{argmax}_k (x^i - \mu_k) \Sigma_k^{-1} (x^i - \mu_k)$$

  “M-step”, we update the means and covariance of cluster using maximum likelihood estimate:
  $$\mu_k = \frac{\sum_i \delta(z^i,k) x^i}{\sum_i \delta(z^i,k)}$$
  $$\Sigma_k = \frac{\sum_i \delta(z^i,k) (x^i - \mu_k)(x^i - \mu_k)^T}{\sum_i \delta(z^i,k)}$$
Theory underlying EM

- What are we doing?

Recall that according to MLE, we intend to learn the model parameter that would have maximize the likelihood of the data.

**Expectation step (E-step)**

- What distribution do we take expectation with? $q(z) = p(z|D, \theta)$
- What do we take expectation over? $f(\theta) = E_{q(z)}[\log p(x, z|\theta)]$

**Maximization step (M-step)**

- What do we maximize? $f(\theta)$
- What do we maximize with respect to? $\theta$