Unsupervised Learning

Data: $\{x_j\}_{j=1}^n$ no label

Tasks:

1. Clustering 聚类

2. Dimension Reduction 降维

3. Density Estimation 目的是生成新的数据

Clustering

Model: $x_j = \bigcup_{k=1}^K c_k, c_p \bigcap c_q = \emptyset, \forall p \neq q$ where c_k is $k-th$ class.

Key: Distance measure

e.g. Diffusion distance： How to define distance on the graph

data --- network $G=(V,W)$ where $V=\{x_j\}, W=\{w_{ij}\}$ represent vertices and weight of edges, respectively. For example, set weight $w_{ij}=e^{-\frac{\|x_i-x_j\|^2}{\sigma^2}}$

e.g. 余弦度量

Objective function:

重心 $\alpha_k = \frac{1}{|c_k|}\sum_{x_j \in c_k} x_j$

$$J_1 = \{c_1, c_2, \dots, c_K\} = \sum_{k=1}^K \sum_{x_j \in c_k} \|x_j - \alpha_k\|^2$$

$$J_2 = \frac{1}{2}\sum_{k=1}^K \frac{1}{|c_k|} \sum_{x_i, x_j \in c_k} \|x_i - x_j\|^2$$

$$Lemma: J_1 = J_2$$

K-means algorithm (Hard Clustering)

(1) Given a clustering

$$\{x_j\} = \bigcup_{k=1}^K c_k$$

计算每个类的重心(at n-step)

$$\alpha_k^{(n)} = \frac{1}{|c_k^{(n)}|} \sum_{x_j \in c_k^{(n)}} x_j$$

(2) 重新聚类(update n+1-step)

$$\forall x_j, k(j) = \arg\min_k \|x_j - \alpha_k^{(n)}\|$$
$$x_j \in c_{k(j)}^{(n+1)}$$

Soft(probabilistic) clustering

$$\forall x_j, p_{jk} = prob\ of \{x_j \in c_k\}$$

Gaussian mixture model

We have two random variables$\{x, z\}$

$$\rho_1 \sim N(\mu_1, \sigma_1), \rho_2 \sim N(\mu_2, \sigma_2)$$
$$Prob\{x \in A|z = 1\} = \int_A \rho_1 dx, Prob\{x \in A|z = 2\} = \int_A \rho_2 dx$$

$$\begin{aligned} Prob\{x \in A\} &= Prob\{x \in A|z = 1\}Prob\{z = 1\} + Prob\{x \in A|z = 2\}Prob\{z = 2\} \\ &= \pi \int_A \rho_1 dx + (1-\pi) \int_A \rho_2 dx \end{aligned}$$
$$Prob\ density = \pi\rho_1(x) + (1-\pi)\rho_2(x)$$

We want to solve the inverse problem: Given $\{x_j\} \sim \text{mixture distribution}$, how to estimate the parameters of the mixture distributions? (**)

Likelihood *(for parameter estimation)*

e.g. (Single Gaussian)Estimate μ, σ

$$\{x_j\} \sim N(\mu, \sigma^2)$$
$$L(\mu, \sigma^2) = \Pi_{j=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_j-\mu)^2}{2\sigma^2}}$$
$$l(\mu, \sigma^2) = \log L(\mu, \sigma^2) = -\sum_{j=1}^n \frac{(x_j-\mu)^2}{2\sigma^2} - n\log\sqrt{2\pi\sigma^2}$$

Back to (**)

$$\log L(\theta) = \sum \log(\pi\rho_1(x_j) + (1-\pi)\rho_2(x_j))$$

It is not easy to solve θ, so we use EM algorithm.

EM Algorithm

n-step: $\pi^{(n)}, \mu_1^{(n)}, \sigma_1^{(n)}, \mu_2^{(n)}, \sigma_2^{(n)}$

n+1-step = ?

(1) E-step

$$
\begin{aligned}
\Delta_j &= \text{"} \; prob \; x_j \; comes \; from \; \rho_1 \; \text{"} \\
&= Prob\{z = 1|x_j\} \\
&= \frac{\pi^{(n)} \rho_1^{(n)}(x_j)}{\pi^{(n)} \rho_1^{(n)}(x_j) + (1 - \pi^{(n)}) \rho_2^{(n)}(x_j)}
\end{aligned}
$$

(2)M-step: update $\mu_1^{(n+1)}, \sigma_1^{(n+1)}, \mu_2^{(n+1)}, \sigma_2^{(n+1)}$

$$
\begin{aligned}
\log(L(\theta)) &= \sum_j \log(\Delta_j \rho_1(x_j) + (1 - \Delta_j) \rho_2(x_j)) \\
&\geq \sum_j \Delta_j \log(\rho_1(x_j)) + (1 - \Delta_j) \log(\rho_2(x_j)) (\text{Jensen ineq.}) \\
&= \text{expected} \log(L(\theta))
\end{aligned}
$$

$$
\theta^{(n+1)} = \arg\min_\theta \{\text{expected} \log(L(\theta))\}
$$

Result($\frac{\partial}{\partial \theta} Obj = 0$):

$$
\begin{aligned}
\mu_1^{(n+1)} &= \frac{\sum \Delta_j x_j}{\sum \Delta_j} \\
\mu_2^{(n+1)} &= \frac{\sum (1 - \Delta_j) x_j}{\sum (1 - \Delta_j)} \\
\sigma_1^{2\,(n+1)} &= \frac{\sum \Delta_j (x_j - \mu_1^{(n+1)})^2}{\sum \Delta_j} \\
\sigma_2^{2\,(n+1)} &= \frac{\sum (1 - \Delta_j)(x_j - \mu_2^{(n+1)})^2}{\sum (1 - \Delta_j)}
\end{aligned}
$$

$$
\pi^{(n+1)} = \frac{1}{n} \sum_{j=1}^{n} \Delta_j
$$

Dimension Reduction

Data: $\{x_j\} \subset R^d$

Linear

We want:

$$
\begin{gathered}
F : x \subset R^d \to z \subset R^{d'}, d' << d \\
G = F^{-1} : z \to x \\
x_j \xrightarrow{F} z_j \xrightarrow{G} \tilde{x}_j \\
\min_{F,G} \sum_j \|x_j - \tilde{x}_j\|^2
\end{gathered}
$$

e.g.

$$F(x) = \beta^\top x, \beta \in R^d$$
$$G(z) = \alpha z, \alpha \in R^d$$
$$\tilde{x} = G(F(x)) = \alpha\beta^\top x$$

$$L(\alpha, \beta) = \frac{1}{2}\sum \|x_j - \tilde{x}_j\|^2 = \frac{1}{2}\sum \|x_j - \alpha\beta^\top x_j\|^2$$
$$\nabla_\alpha L = -\sum (x_j - \alpha\beta^\top x_j)\beta^\top x_j = 0$$
$$\nabla_\beta L = -\sum (x_j - \alpha\beta^\top x_j)\alpha^\top x_j = 0$$
$$\alpha = \beta$$
$$(\sum x_j x_j^\top)\beta = \sum \beta\beta^\top x_j x_j^\top \beta = \beta \sum \beta^\top x_j x_j^\top \beta = \lambda\beta$$

Why $\alpha = \beta$?

Refer Deep Learning (Page 45-50)

Rewrite formula above (denote $X = \sum x_j x_j^\top$)

$$X\beta = \lambda\beta$$

The case is a special **PCA**(Principal Component Analysis). General PCA:

$$z = W_{\tilde{d}\times d} x \in R^{\tilde{d}}, \quad \tilde{x} = V^\top_{\tilde{d}\times d} z = V^\top W x \in R^d$$
$$L(w) = \sum \|x_j - \tilde{x}_j\|^2 = \sum \|x_j - W^\top W x_j\|^2 (V = W)$$

Consider $\tilde{W} = QW$ where $Q^\top Q = I$.

$$L(\tilde{W}) = \sum \|x_j - W^\top Q^\top Q W x_j\|^2 = \sum \|x_j - W^\top W x_j\|^2 = L(W)$$
$$\nabla_W L(W) = \sum (x_j - W^\top W x_j) x_j^\top W = 0$$
$$(\sum x_j x_j^\top) W^\top = \sum W^\top W x_j x_j^\top W \to XW^\top = \Lambda W^\top$$

Where Λ is the diagonal matrix of the largest $\tilde{d}$ eigenvalues with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{\tilde{d}}$.

Non-Linear: Auto-encoder

Use NN represent F, G, input R^d, output$R^{\tilde{d}}$

Objective: $L(\theta) = \sum_{j=1}^n \|x_j - \tilde{x}_j\|^2$

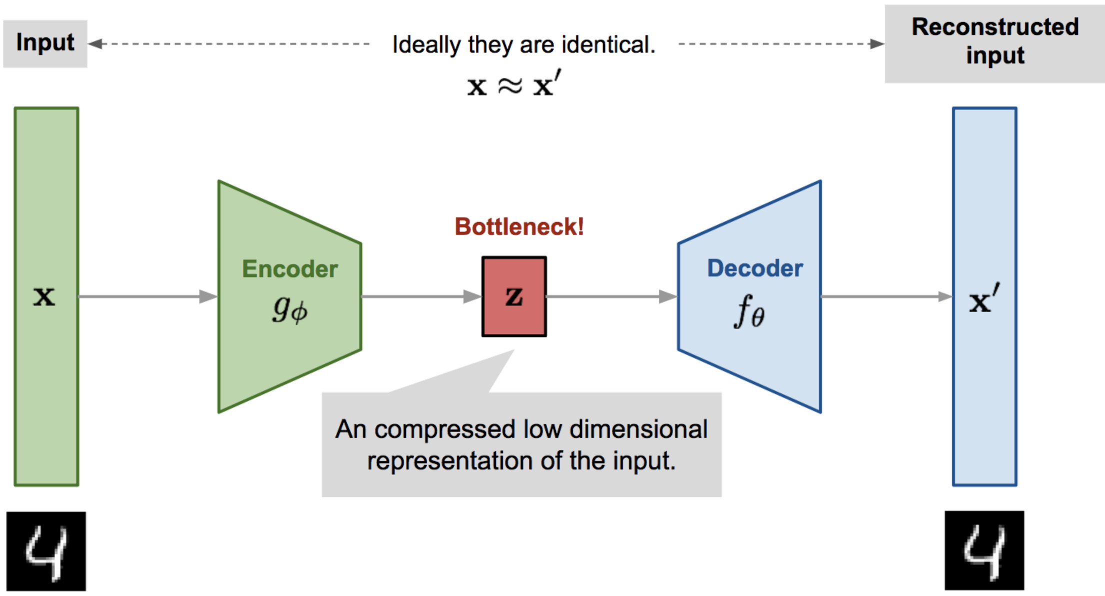

Figure source: https://lilianweng.github.io/lil-log/2018/08/12/from-autoencoder-to-beta-vae.html

Density Estimation

Data: $\{x_j\}$

Objective: $\{x_j\} \sim \mu$ want to find μ

Basic idea: histogram

$$\rho(x) \approx \frac{1}{n} \sum \frac{1}{h} H(\frac{x - x_j}{h})$$
$$\int H(x)dx = 1$$
$$H(x) = \delta(x) \text{ for example}$$

High-dimension case

$$R^{\tilde{d}} \to R^d$$
$$\exists G, \forall f, \int f(G(z))d\mu^* = \int f d\mu \approx \frac{1}{n} \sum_{j=1}^{n} f(x_j)$$

$$\sup_{\|f\|_{lip} \leq 1} |\int f(G(z))d\mu^* - \frac{1}{n} \sum f(x_j)| = L(\theta)$$

Where $\|\cdot\|_{lip}$ is Lipschitz norm. (W-GAN)