VARIATIONAL AUTOENCODERS

p_ϕ & q_θ → VARIATIONAL

TRADITIONAL

$q_\theta(x) = y$

$p_\phi(y) = \tilde{x}$

STOCHASTIC MAPPING

$q_\theta(y|x)$

$p_\phi(\tilde{x}|y)$

Max. Likelihood for the loss

prior to $p(y) \sim N(\phi, 1)$

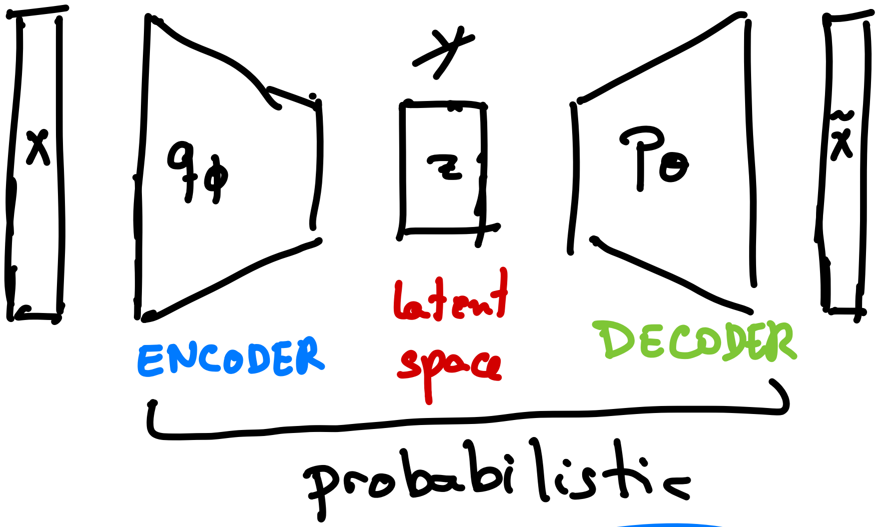
$KL(p_\phi(x|y) \mid \underbrace{N(\phi,1)}))$

LECTURE 5 28/10/2024

INTRODUCTION TO VAE

INTRINSIC DIMENSION ESTIMATION

DENSITY ESTIMATION



$$q_\phi(x) = z \rightarrow q_\phi(z|x)$$

$$p(z)$$

$$p_\phi(x|z)$$

Generative process

1. Sample z
2. Sample x from $p(x|z)$

$$p(x) = \int p(x|z) \underbrace{p(z)\, dz}_{\text{unknown}}$$

$$MC \rightarrow E_{z\sim p(z)}\left[p(x|z)\right] \sim p(x) = \frac{1}{K}\sum_{i}^{K} p(x|z_i)$$

Variational inference

$$\{ q_\phi(z) \}$$

Gaussian $\phi = \{\mu, \sigma^2\}$

$\downarrow$ Amortized

$$q_\phi(z|x)$$

$$\log p(x) = \log \int p(x|z)\, p(z)\, dz$$

$$\log \int \frac{q_\phi(z|x)}{q_\phi(z|x)} p(x|z)\, p(z)\, dz =$$

$$\log \mathbb{E}_{z \sim q_\phi(z|x)} \left[\frac{p(x|z)\, p(z)}{q_\phi(z|x)} \right] \geq$$

$$\mathbb{E}_{\substack{z\sim \\ q_\phi(z|x)}} \left(\left[\log \left(\frac{p(x|z)\, p(z)}{q_\phi(z|x)} \right) \right] \right) =$$

$$\mathbb{E}_{z \sim q_\phi(z|x)} \left(\log(p(x|z)) - \right.$$
$$\left. \mathbb{E}_{z \sim q_\phi(z|x)} \left[\log(q_\phi(z|x)) - \log p(z) \right] \right)$$

$$\log p(x) \geq \mathbb{E}_{z \sim q_\phi(z|x)} \left[\log(p(x|z)) \right]$$
$$- D_{KL}\left(q_\phi(z|x) \,\|\, p(z) \right)$$

$$\boxed{\log p(x) \geq E_{z \sim q_\phi(z|x)}\left[\log(p(x|z))\right] - D_{KL}\left(q_\phi(z|x) \,||\, p(z)\right)}$$

ELBO

PARAMETERIZING

MNIST $\quad 28 \times 28 \to D = 784$

$0 \ldots 255$

$$p(x) = \text{Categorical} \qquad P \begin{pmatrix} p(x_1 = 0) \ldots & p(x_1 = 255) \\ \vdots & \\ p(x_{784} = 0 \ldots & \end{pmatrix}$$

$$p_\theta(x|z) = \text{Categorical}(x \mid \theta(z))$$

$$\theta(z) = \text{Softmax}(NN(z))$$

$q_\phi(z|x)$, $p(z)$

$$q_\phi(z|x) = \mathcal{N}(z; \mu_\phi(x), \text{diag}(\sigma^2_\phi(x)))$$

$$p(z) = \mathcal{N}(z; 0, 1)$$

$$(\mu_\phi(x); \sigma^2_\phi(x)) = NN(x)$$

Optimizing the ELBO

$$\nabla_{\{\theta,\phi\}}$$

$$\nabla_\theta \Rightarrow \quad \nabla_\theta \mathcal{L}(\theta,\phi,x) =$$

$$\nabla_\theta E_{q_\phi(z|x)}\left[\log p_\theta(x|z) - \log q_\phi(z|x)\right] =$$

$$E_{q_\phi(z|x)}\left[\nabla_\theta \log p_\theta(x|z) - \underbrace{\nabla_\theta \log q_\phi(z|x)}_{\emptyset}\right]$$

$$\nabla_\theta \mathcal{L}(\theta,\phi,x) = E_{q_\phi(z|x)}\left[\nabla_\theta \log p_\theta(x|z)\right]$$

$$\sim \nabla \log p(x|z)$$

∇_ϕ REPARAMETERIZATION TRICK

$z = g_\nu(\varepsilon)$ $\quad \varepsilon = \hat{q}(\varepsilon)$ $\quad \hat{q}$ it's independent on ϕ

$$\hat{\phi} = (\phi, \nu)$$

$$\mathcal{L}(\hat{\phi},\theta,x) = E_{\hat{q}(\varepsilon)}\left[\log\left(p(x, g_\nu(\varepsilon))\right) - \log q_\phi(g_\nu(\varepsilon))\right]$$

So $\nabla_{\hat{\phi}}$ -

$$E_{\hat{q}(\varepsilon)}\left[\underbrace{\nabla_{\hat{\phi}} \log (p(x, g_v(\varepsilon)) - \nabla_{\hat{\phi}} \log q_\phi(g_v(\varepsilon))}_{G(\varepsilon)}\right]$$

$$\nabla_{\hat{\phi}} \mathcal{L}(\hat{\phi}, \theta, x) = \frac{1}{S} \sum_{i}^{S} G(\varepsilon_i)$$

$$z = \mu + \sigma \odot \varepsilon \qquad \varepsilon \sim \mathcal{N}(0, I)$$

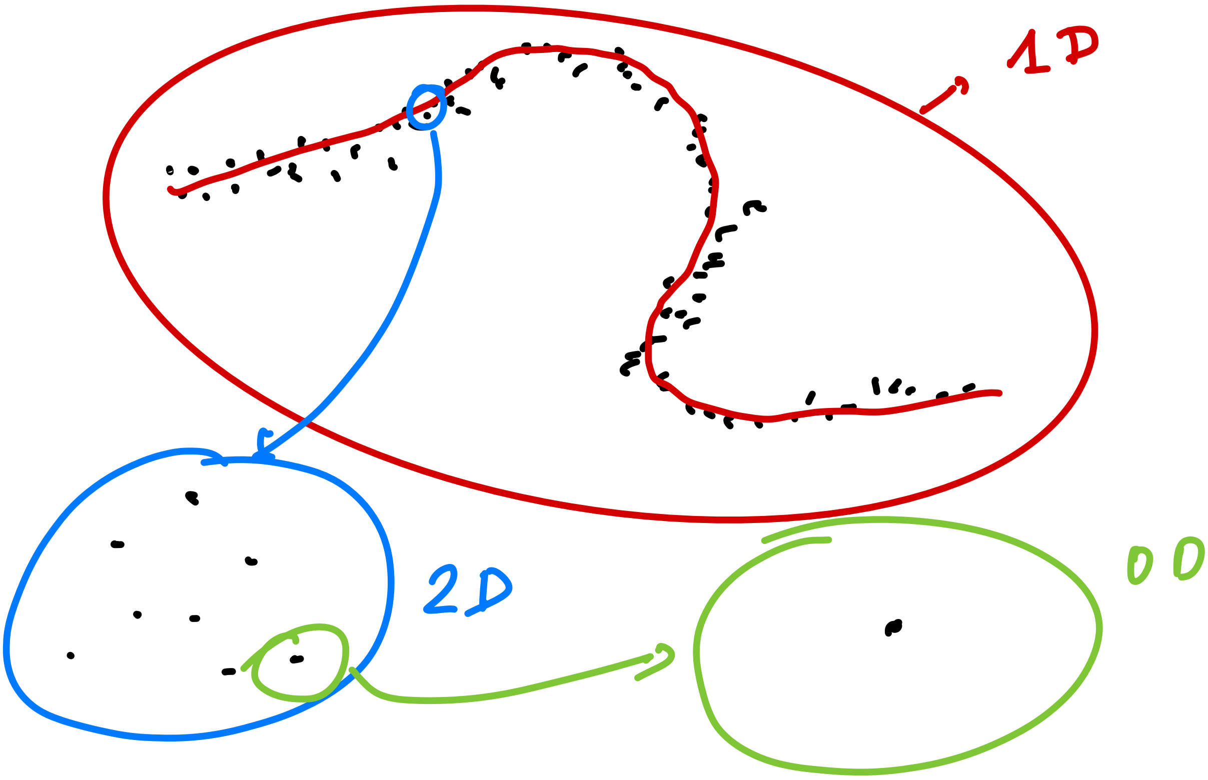
BUILDING A VAE

1. Given x_n apply the encoder $(\mu_\phi(x_n), \sigma_\phi^2(x_n))$
2. Use repar... trick to obtain $z_{\phi,n} = \mu_\phi(x_n) + \sigma_\phi^2(x_n) \cdot \varepsilon$
3. Apply decoder to obtain $\theta(z_{\phi,n})$
4. Compute the ELBO
5. Compute gradients and update ϕ, θ

INTRINSIC DIMENSION

ⓐ Minimum Number of variables needed for repre[se]nt the data with minimum information Loss

ⓑ The dimension of the manifold in which our data lies

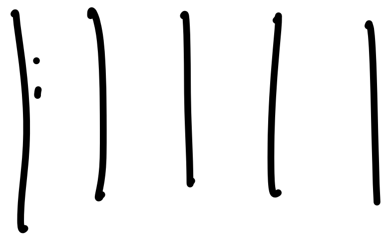


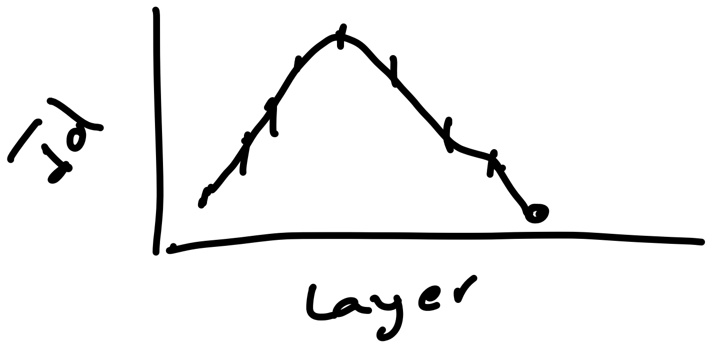
Intrinsic dimension is a scale-dependent property

① Decide d

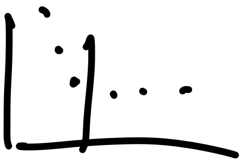
② Information content

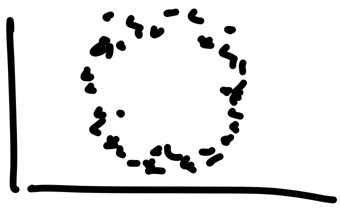
Id

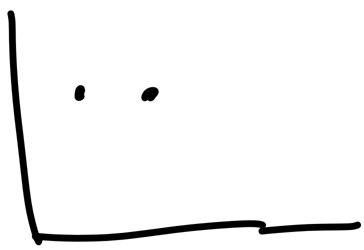


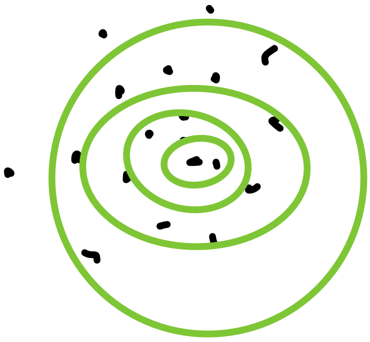


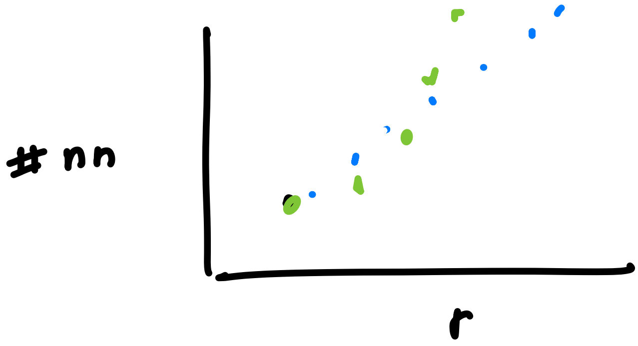
PCA → Spectrum

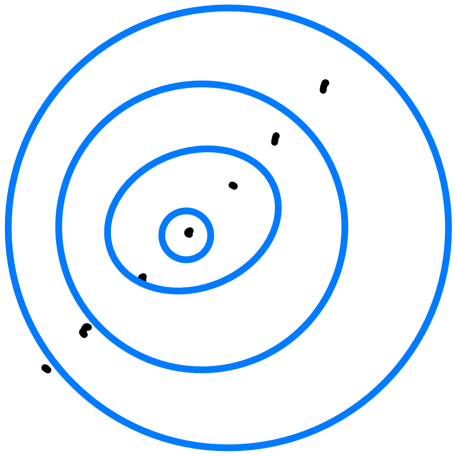












$$\boxed{\#nn \propto r^d}$$

$$\#nn = \rho \cdot r^d$$

$$\log(\#) = \log(\rho) + d \log(r)$$

for each data point

a) Compute the distances from its K-NN

b) Plot the log of the rank as function of the logarithm of the distances

c) linear fit

d) slope will be d

$$\log(\#) = \underline{\log(\rho)} + d \log(r)$$

$i \quad r_i^1 \quad r_i^2 \quad \dots \quad r_i^k$

1

$\log(1) \quad \log(r_i^1)$

$\log(2) \quad \log(r_i^2)$

Id from the ρ

two-NN

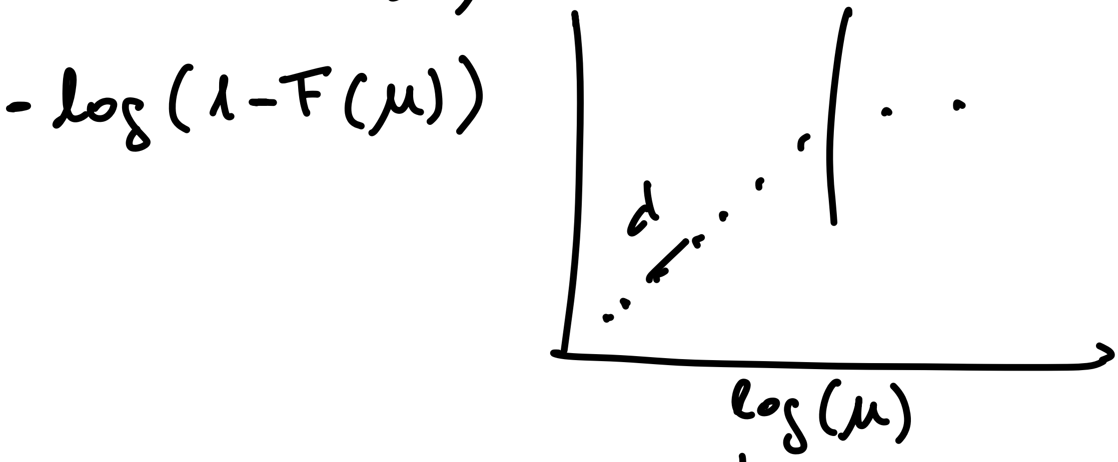
$$\mu_i = \frac{r_2^i}{r_1^i} \in [1, \infty) \qquad \text{i.i.d}$$

$$p(\mu) = \underbrace{\mu^{-d-1} \cdot d}_{\text{pareto}}$$

ρ within the second NN ~ constant

ⓐ $F(\mu) = \int_1^{\mu} \mu^{*\,-d-1} \cdot d \cdot d\mu^* = 1 - \mu^{-d}$

$$\frac{-\log(1-F(\mu))}{\log(\mu)} = d$$



ⓑ ML $\qquad p(\mu) = \mu^{-d-1} d$

$$\log \mathcal{L} = \sum_i \log\left(\mu_i^{-d-1} \cdot d\right) =$$

$$N \log(d) - (d+1) \sum_i \log(\mu_i)$$

$$\frac{\partial \log \mathcal{L}}{\partial d} = \frac{N}{d} - \sum_i \log(\mu_i) = \emptyset$$

$$d = \frac{N}{\sum_i \log(\mu_i)}$$

- Max. Likelihood
- DANCO

Density Estimation

$P(y \mid x)$ supervised

$\boxed{p(x)}$ Unsupervised

Dim. Reduction

$$x \longrightarrow y \in \mathbb{R}^d$$

Id estimation

manifold learning

Obtain $p(x)$ task density estimation

Parametric : Assume a functional form
Learn the parameters by fitting the data

Non-parametric : There's no assumption about the functional form

Parametric : Powerful but "rigid"

Non-parametric :

Gaussian Mixture Model (Parametric):

$$\hat{p}(x) \sim p(x)$$

$$\hat{p}(x) = \sum_i^K \pi_i \, \varphi(x, \theta_i)$$

$$\theta_i = (\mu_i, \Sigma_i)$$

1D → μ, σ_i^2

Max. Likelihood

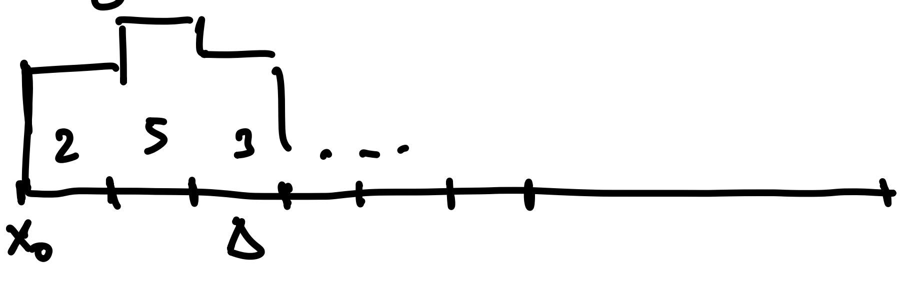
$$\mathcal{L} = \prod_l^N \left(\sum_i^K \pi_i \, \varphi(x^l; \theta_i) \right)$$

$$\log \mathcal{L} = \sum_l^N \log \left(\sum_i^K \pi_i \, \varphi(x^l; \theta_i) \right)$$

Expectation - Maximization

Non-parametric density estimation

- Histograms



x_{min} ; Δ ; N_{bins} or x_{max}

$$p(x) \sim \hat{p}(x_j) = \frac{n_j}{N \Delta}$$

$$\mathcal{E}(\hat{p}(x_j)) \propto \frac{\hat{p}(x_j)}{\sqrt{n_j}}$$

$$\Delta \sim \frac{2\,IQR}{\sqrt[3]{N}}$$ Freedman-diaconis Rule

$$IQR = X_{(3/4)} - X_{(1/4)}$$

$$p(x) \sim \hat{p}(x) = \sum_{l} f(x^l, h)\,\frac{1}{N h}$$

kernel density estimation

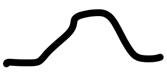
$$p(x) \sim \hat{p}(x) = \sum_{l}^{N} K(x_l, h)\,\frac{1}{N h}$$

K = Uniform 

= P. arzen windows

K = Gaussian $K(x, x_l, h) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x - x^l}{h}\right)^2\right)$

K  triangl

Cosine 

Epenechnikov  cubic h → Silverman's Rule

$$h = 0.9 \min\left(\sigma, \frac{IQR}{1.34}\right) \cdot N^{-1/5}$$

Derived assuming gaussianity

$$\left.\begin{matrix} h \nearrow p_1(x) \\ h \searrow p_2(x) \end{matrix}\right\} \Delta = KL(p_1 \| p_2)$$

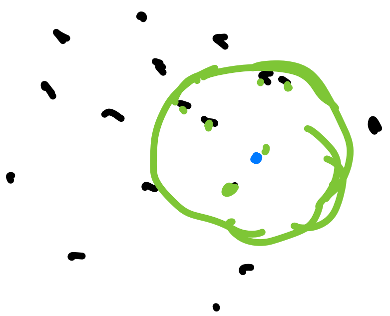
~~as min~~

$$h_{opt} = \underset{h}{\operatorname{argmin}} \; \Delta(h)$$

K-Nearest Neighbor density estimator

$$p(x) = \frac{K}{N V_K} = \frac{k}{V^d \, r_k^d \, N}$$

Volume of the hypersphere of dimension d



r_k → distance from the k-th NN

$$\varepsilon(p) = \frac{p}{\sqrt{k}}$$