VAE

为什么VAE？

AE面临的生成挑战

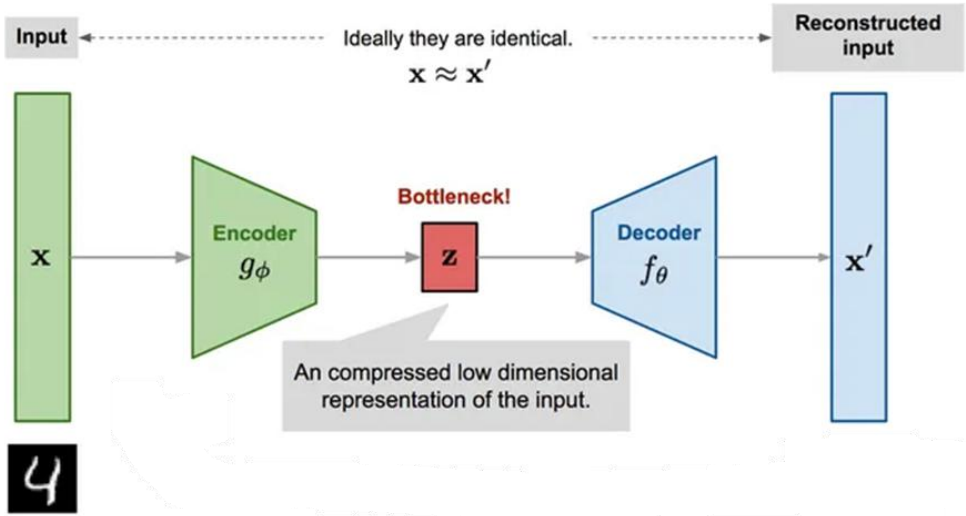
- VAE，旨在解决AE无法生成新样本的根本问题
  - 将AE从一个确定性的“压缩-解压”框架，升级为一个概率生成模型

- VAE核心目标是
  - 学习数据的光滑的、连续的、服从简单先验分布的潜在向量表示
    - 如，标准正态分布 $\mathrm{N}(0, \mathrm{I})$
  - 在此基础上，最大化数据对数似然log p(x)的证据下界ELBO，从而同时
    - 实现良好的重建
    - 有意义的潜在向量表示

AE概要

AE

- 将高维的原始数据映射到一个低维特征空间，然后重建回原始数据
  - g_ϕ 为Encoder网络（参数为 ϕ），f_θ 为Decoder网络（参数为 θ ）
  - Encoder：$\mathbf{z} = g_\phi(\mathbf{x})$，Decoder: $\mathbf{x}' = f_\theta(\mathbf{z}) = f_\theta\big(g_\phi(\mathbf{x})\big)$
  - 重建数据和原来数据近似一致，损失函数：$L_{\mathrm{AE}}(\theta, \phi) = \frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{x}^{(i)} - f_\theta\left(g_\phi\left(\mathbf{x}^{(i)}\right)\right)\right)^2$

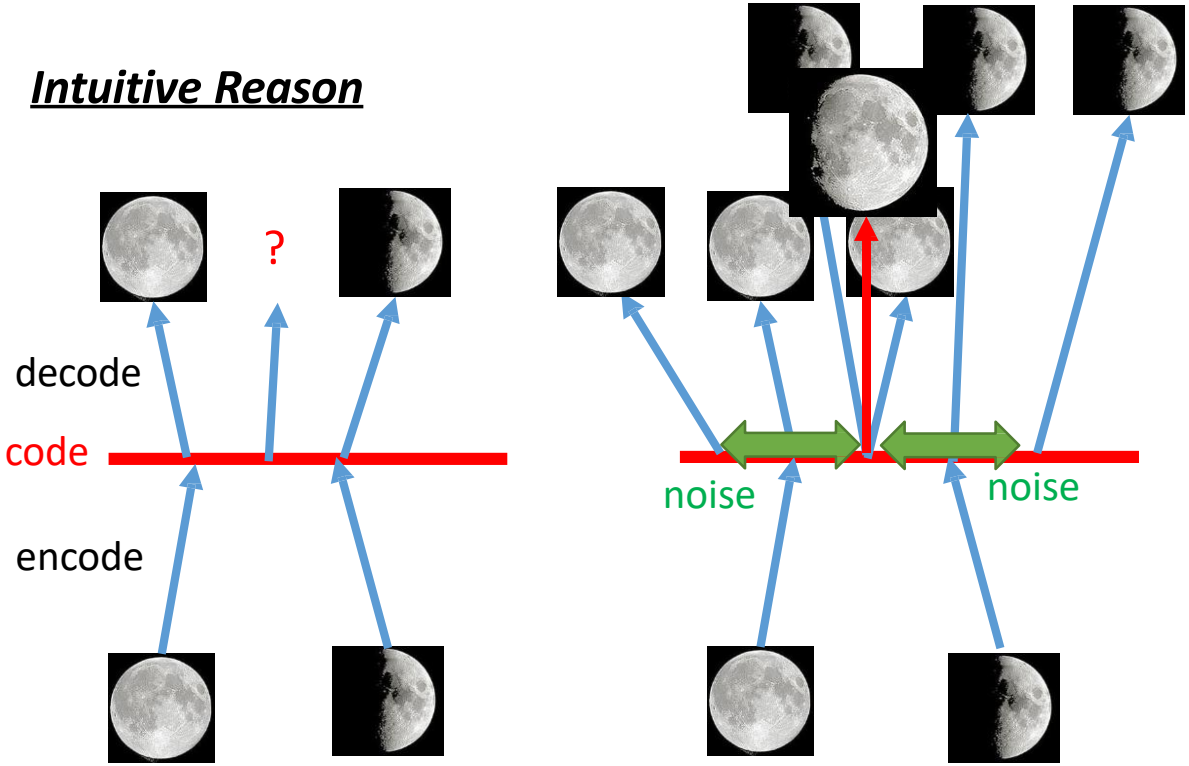


AE的缺陷

- 缺乏生成能力
  - AE主要用于数据重构，学习数据的压缩表示
  - 虽然可以生成数据，但通常不是生成新样本的良好选择
- 模型的概率解释不足
  - 没有内置的概率解释。潜在空间的点没有明确的概率分布，难以进行概率推断
- 潜在空间表示不连续
  - 潜在空间可能会导致非连续的样本表示，这在生成新样本时可能会出现问题
- 缺少正则化
  - 可能容易过拟合，因为没有内置机制来规范潜在空间的结构
- 应用
  - 适用于特征提取、降维和去噪等任务

AE的生成问题

- code插值再解码

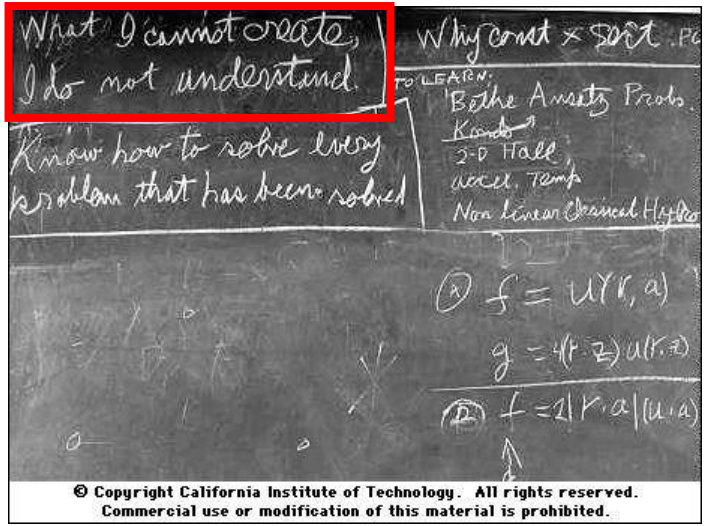


如何描述数据规律？

高斯混合模型

生成才是真正的理解

- Generative Models: https://openai.com/blog/generative-models/



What I cannot create, I do not understand.

Richard Feynman

https://www.quora.com/What-did-Richard-Feynman-mean-when-he-said-What-I-cannot-create-I-do-not-understand

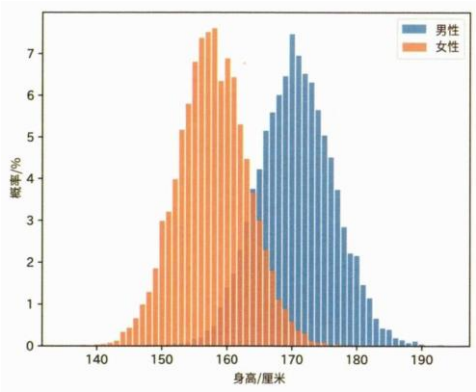
如何对复杂的数据分布进行建模和理解？

- 无监督学习中的两个核心任务
  - 密度估计（Density Estimation）
    - 给定数据集，能否学习概率分布 $p(x)$ 来描述这些数据规律？
      - 该模型应该能表明，任意一个新的数据x属于这个分布的可能性$p(x)$有多大
  - 表示学习（Representation Learning）
    - 能否为每个数据点找到一个更有意义的、通常是低维的＂编码＂或＂表示＂？
      - 该表示能捕捉到数据背后的核心结构，如它的类别，或者在某个连续谱系上的位置

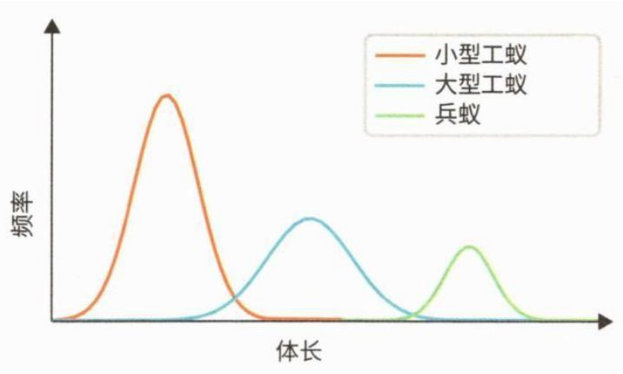
- 混合模型的方案
  - GMM的方案
    - 数据是由少数几个（有限的）简单的分布（高斯分布）混合而成
  - VAE则认为
    - 数据是由一个无限丰富的、连续的“潜在空间”（Latent Space）通过复杂的非线性变换，混合生成

离散高斯混合模型

- 17 岁男性和 17 岁女性的身高分布
  - 数据来自 2016 年日本学校健康统计调查
  - 已知
    - 每个性别（类别）的身高分布
  - 求解
    - 类别的分布概率（参数：均值和方差）
- 蚂蚁体长的分布呈现多峰性



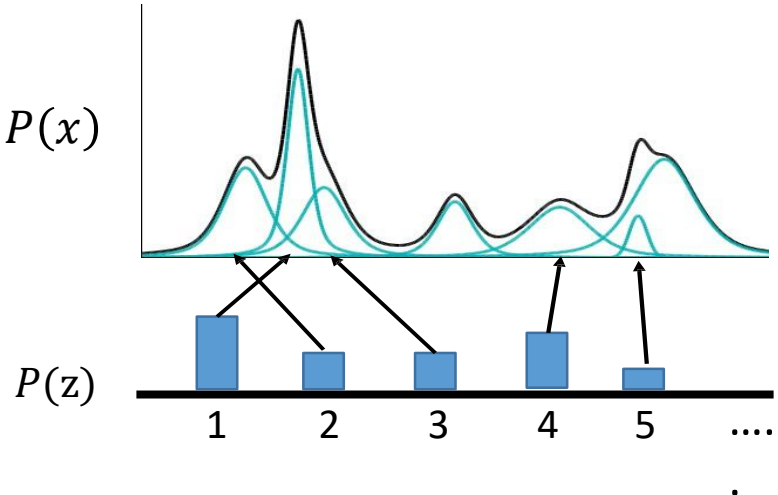
- 隐变量z
  - 代表类别的变量
  - 有限类别，离散的变量



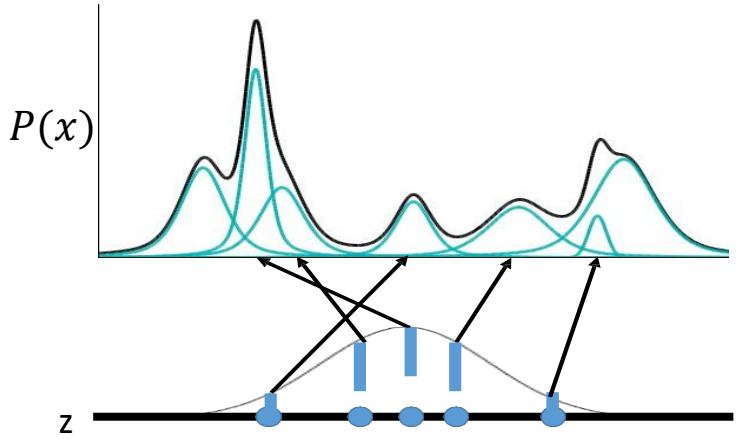
隐变量模型的两种范式

高斯混合模型

- *离散形式*
  - $z \sim P(z),\ \sum_z P(z) = 1$
  - $x \mid z \sim N(\mu^z, \Sigma^z)$
  - $P(x) = \sum_z P(z)P(x \mid z)$,



- 连续形式
  - $z \sim N(0, I)$
  - $x \mid z \sim N(\mu(z), \sigma(z))$
  - $P(x) = \int_z P(z)P(x \mid z)dz$



极大似然估计

混合高斯模型（GMM）

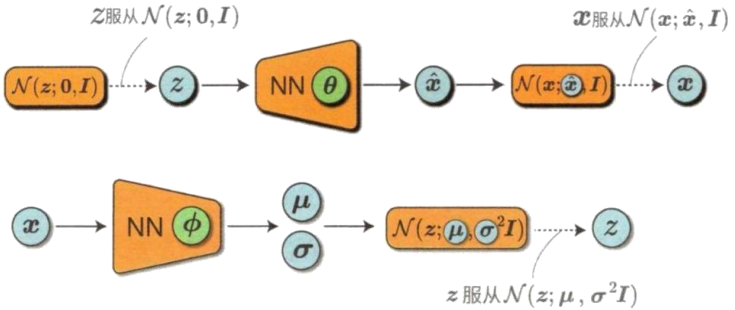
- GMM假设数据分布 $p(x)$ 是 K 个高斯分布的加权和

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x \mid \mu_k, \Sigma_k)$$

- 生成过程
  - 引入一个离散的隐变量 z，$p(z = k) = \pi_k$
  - 给定 $z = k$，数据点 x 从第 k 个高斯分布中采样：$p(x \mid z = k) = \mathcal{N}(x \mid \mu_k, \Sigma_k)$
- （EM）算法
  - E步：计算每个数据点 x_i 来自每个高斯分量 k 的后验概率（称为“责任”），即 $p(z_i = k \mid x_i)$【选择$p(z)$，最小化KL散度，更新ELBO 】
  - M步：使用这些“责任”作为权重，更新每个高斯分量的参数 π_k, μ_k, Σ_k 【最大化ELBO】

变分自编码器 (VAE)

- VAE假设数据由连续隐变量 z 通过神经网络生成
- $p_\theta(x) = \int p_\theta(x \mid z)p(z)dz$
  - 从简单的先验分布（如 $\mathcal{N}(0, I)$ ）中采样连续的隐变量 z
  - 将z输入由参数θ定义的解码器网络，其输出定义 x 分布 $p_\theta(x \mid z)$（如高斯分布均值和方差）



- 学习算法（后验 $p_\theta(z \mid x)$ 和证据 $p_\theta(x)$ 都难以计算）
  - 引入编码器神经网络 $q_\phi(z \mid x)$ 来近似后验$p_\theta(z \mid x)$，通过最大化ELBO来同时优化编码器和解码器
    - $\mathcal{L}(\theta, \phi) = \mathbb{E}_{z \sim q_\phi(z|x)}[\log p_\theta(x \mid z)] - D_{KL}\big(q_\phi(z \mid x) \| p(z)\big)$
    - 看作：ELBO = (预期收益) - (解释成本)

极大似然 – 基于ELBO的推导

似然函数$\log p_\theta(x)$

- 【目标函数】 $\log p_\theta(x)$
- 引入简单的隐变量分布$q_\phi(z \mid x)$，逼近真实隐变量分布$p_\theta(z \mid x)$【 引入$\frac{q_\phi(z|x)}{p_\theta(z|x)}$ 】
- 第一步：引入期望
  - $\log p_\theta(x)$ 不依赖于 z，所以它关于任何关于 z 的分布 $q_\phi(z \mid x)$ 的期望就是其自身
    - $\log p_\theta(x) = \mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x)]$
  - 这一步的目的是为了把 $q_\phi(z \mid x)$ 引入推导框架中
- 第二步：利用贝叶斯定理【引入$p_\theta(z \mid x)$】
  - 将 $p_\theta(x)$ 用联合概率和条件概率表示：$p_\theta(x) = \frac{p_\theta(x,z)}{p_\theta(z|x)}$。代入
  - $\log p_\theta(x) = \mathbb{E}_{z\sim q_\phi(z|x)}\left[\log \frac{p_\theta(x,z)}{p_\theta(z|x)}\right]$

似然函数$\log p_\theta(x)$

- 【目标函数】 $\log p_\theta(x)$
- 第三步：引入 " 魔法项 " $q_\phi(z \mid x)$
  - 核心技巧。我们在对数函数内部同时乘以和除以我们的近似分布 $q_\phi(z \mid x)$ 。这并不会改变表达式的值：
  - $\log p_\theta(x) = \mathbb{E}_{z\sim q_\phi(z|x)}\left[\log\left(\frac{p_\theta(x,z)}{q_\phi(z|x)} \cdot \frac{q_\phi(z|x)}{p_\theta(z|x)}\right)\right]$
- 第四步：拆分对数项【引入对比项$\frac{q_\phi(z|x)}{p_\theta(z|x)}$】
  - $\log p_\theta(x) = \mathbb{E}_{z\sim q_\phi(z|x)}\left[\log\frac{p_\theta(x,z)}{q_\phi(z|x)}\right] + \mathbb{E}_{z\sim q_\phi(z|x)}\left[\log\frac{q_\phi(z|x)}{p_\theta(z|x)}\right]$
  - $\log p_\theta(x) = \mathcal{L}_{ELBO}(\theta,\phi) + D_{KL}\left(q_\phi(z \mid x)\|p_\theta(z \mid x)\right)$

似然函数$\log p_\theta(x)$

- 【优化项】$\log p_\theta(x) = \mathcal{L}_{ELBO}(\theta,\phi) + D_{KL}\left(q_\phi(z \mid x) \| p_\theta(z \mid x)\right)$
- 第五步：识别最终形式
  - $\mathcal{L}_{ELBO}(\theta,\phi) = \mathbb{E}_{z\sim q_\phi(z|x)}\left[\log \frac{p_\theta(x,z)}{q_\phi(z|x)}\right]$
    - 证据下界（Evidence Lower Bound，ELBO），记作 $\mathcal{L}_{ELBO}(\theta,\phi)$
  - $D_{KL}\left(q_\phi(z \mid x) \| p_\theta(z \mid x)\right) = \mathbb{E}_{z\sim q_\phi(z|x)}\left[\log \frac{q_\phi(z|x)}{p_\theta(z|x)}\right]$
    - KL散度，它衡量了近似分布 $q_\phi(z \mid x)$ 与真实后验分布 $p_\theta(z \mid x)$ 之间的 " 距离 "
- 于是，得到变分推断中最核心的恒等式
  - $\log p_\theta(x) = \mathcal{L}_{ELBO}(\theta,\phi) + D_{KL}\left(q_\phi(z \mid x) \| p_\theta(z \mid x)\right)$

ELBO的进一步推导

- 【优化目标】 $\mathcal{L}_{ELBO} = \mathbb{E}_{z\sim q_\phi(z|x)}\left[\log \frac{p_\theta(x,z)}{q_\phi(z|x)}\right]$
- 利用概率链式法则 $p_\theta(x,z) = p_\theta(x \mid z)p(z)$ ：
  - $\mathcal{L}_{ELBO} = \mathbb{E}_{z\sim q_\phi(z|x)}\left[\log \frac{p_\theta(x|z)p(z)}{q_\phi(z|x)}\right]$
- 再次拆分对数项：
  - $\mathcal{L}_{ELBO} = \mathbb{E}_{z\sim q_\phi(z|x)}\left[\log p_\theta(x \mid z) + \log \frac{p(z)}{q_\phi(z|x)}\right]$
- 利用期望的线性性质，得到ELBO最常见的形式
  - $\mathcal{L}_{ELBO}(\theta,\phi) = \underbrace{\mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x \mid z)]}_{\text{重建项 (Reconstruction Term)}} - \underbrace{D_{KL}\big(q_\phi(z \mid x)||p(z)\big)}_{\text{正则化项 (Regularization Term)}}$

ELBO的物理意义

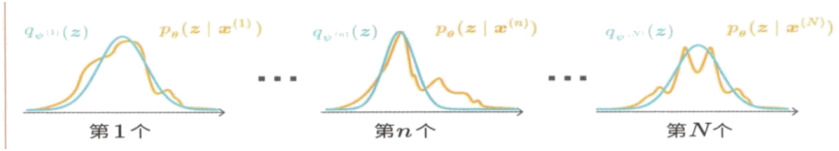
ELBO的进一步推导 - 物理意义解读

- $\mathcal{L}_{ELBO}(\theta,\phi) = \underbrace{\mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x \mid z)]}_{\text{重建项 (Reconstruction Term)}} - \underbrace{D_{KL}\big(q_\phi(z \mid x)||p(z)\big)}_{\text{正则化项 (Regularization Term)}}$
- 重建项 $\mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x \mid z)]$
  - 对于输入图像 x，通过编码器得到z 的分布 $q_\phi(z \mid x)$，从中采样一个具体的 z，然后让解码器根据这个 z 生成图像。该项衡量原始图像 x 在这个生成过程下的对数似然
  - 最大化这一项等价于最小化重建损失。常采用MSE或BCE损失
- 正则化项$-D_{KL}\big(q_\phi(z \mid x)||p(z)\big)$：
  - KL散度：衡量编码器为图像 x 推断出的分布 $q_\phi(z \mid x)$ 与预设标准正态分布$p(z)$之间的差异
    - 看作：用$p(z)$编码$q_\phi(z \mid x)$付出的额外信息成本
  - 起到正则项作用，使得编码器产生的潜变量分布不能离标准正态分布太远
    - 保证了潜空间的连续性和结构性，使得可以从先验 $p(z)$ 中采样来生成全新的、有意义的图像

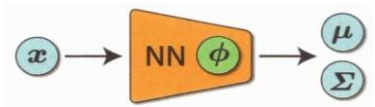
基于神经网络的优化

ELBO正则项优化

- $\mathcal{L}_{ELBO}(\theta,\phi) = \underbrace{\mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x \mid z)]}_{\text{重建项 (Reconstruction Term)}} - \underbrace{D_{KL}\left(q_\phi(z \mid x)||p(z)\right)}_{\text{正则化项 (Regularization Term)}}$
- N 个数据 $\left\{\mathbf{x}^{(1)},\mathbf{x}^{(2)},\cdots,\mathbf{x}^{(N)}\right\}$。$\sum_{n=1}^{N}\text{ELBO}\left(\mathbf{x}^{(n)};\boldsymbol{\theta},\boldsymbol{\psi}^{(n)}\right) = \sum_{n=1}^{N}\int q_{\psi^{(n)}}(\mathbf{z})\log\frac{p_\theta\left(\mathbf{x}^{(n)},\mathbf{z}\right)}{q_{\psi^{(n)}}(\mathbf{z})}\mathrm{d}\mathbf{z}$
  - 每个数据 $\mathbf{x}^{(n)}$：概率分布 $q_{\psi^{(n)}}(z)$，参数为 $\boldsymbol{\psi}^{(n)} = \left\{\boldsymbol{\mu}^{(n)},\sum^{(n)}\right\}$ 的正态分布
  - 最小化 ELBO的正则化项，则 $q_{\psi^{(n)}}(z)$ 应接近后验分布 $p_\theta\left(z \mid \mathbf{x}^{(n)}\right)$



  - 摊销推理 (amortized inference)



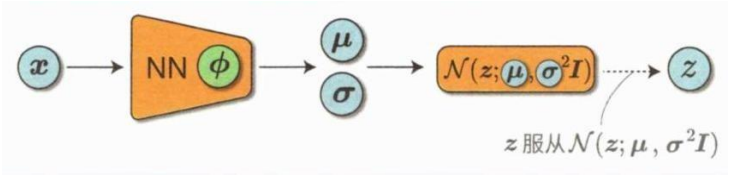
  - 用输入为 $\mathbf{x}^{(n)}$ 、输出为 $\left\{\boldsymbol{\mu}^{(n)},\sum^{(n)}\right\}$ 的 “编码器” (encoder) 神经网络来计算近似后验分布参数

编码器 (Encoder)

- VAE 编码器执行的处理可以表示为

$$\boldsymbol{\mu}, \boldsymbol{\sigma} = \text{NeuralNet}(\mathbf{x}; \phi)$$

$$q_\phi(\mathbf{z} \mid \mathbf{x}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\sigma}^2\mathbf{I})$$



- 注意：此处的采样计算是不可微的

ELBO重建误差项计算

- 计算过程
  - 计算开始于编码器 NeuralNet $(\mathbf{x};\phi)$，从输入数据 $\mathbf{x}$ 中采样 z
  - 在此基础上，通过解码器NeuralNet$(\mathbf{z};\theta)$ 重新生成数据 $\hat{\mathbf{x}}$
  - 重新生成的数据 $\hat{\mathbf{x}}$ 越接近原始输入数据 $\mathbf{x}$,J_1 的值就越大。因此，J_1被称为 " 重建误差项 "

$$J_1 \approx \log \mathcal{N}(\mathbf{x};\hat{\mathbf{x}},\mathbf{I})$$

$$= \log\left(\frac{1}{\sqrt{(2\pi)^D|\mathbf{I}|}}\exp\left(-\frac{1}{2}(\mathbf{x}-\hat{\mathbf{x}})^\top\mathbf{I}^{-1}(\mathbf{x}-\hat{\mathbf{x}})\right)\right)$$

$$= -\frac{1}{2}(\mathbf{x}-\hat{\mathbf{x}})^\top(\mathbf{x}-\hat{\mathbf{x}}) + \underbrace{\log\frac{1}{\sqrt{(2\pi)^D}}}_{\text{常数}}$$

$$= -\frac{1}{2}\sum_{d=1}^{D}(x_d-\hat{x}_d)^2 + \text{const}$$

ELBO重建项计算

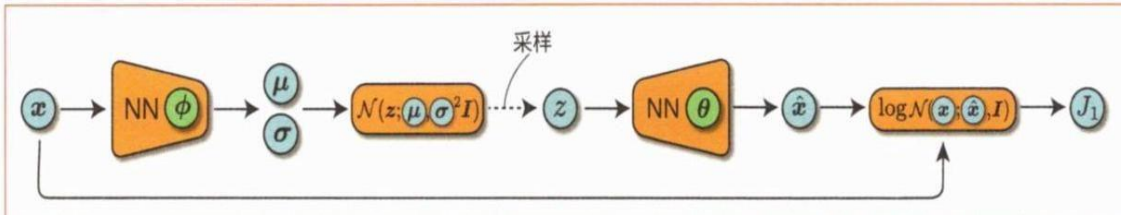
- 重建项$J_1 = \mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(\mathbf{x} \mid \mathbf{z})]$的蒙特卡洛方法来近似计算
  - 基于 $q_\phi(z \mid \mathbf{x})$ 生成一些随机数样本 z，然后计算 $\log p_\theta(\mathbf{x} \mid z)$ 的平均值
  - 即使样本量为 1, VAE 的效果通常也很好。
- 如果样本量为 1, 通过 $z \sim q_\phi(z \mid \mathbf{x})$ 采样一个 z ,并通过计算 $\log p_\theta(\mathbf{x} \mid z)$ 中的 J_1

$$\boldsymbol{\mu}, \boldsymbol{\sigma} = \text{NeuralNet}(\mathbf{x}; \phi)$$

$$z \sim \mathcal{N}(z; \boldsymbol{\mu}, \sigma^2 \mathbf{I})$$

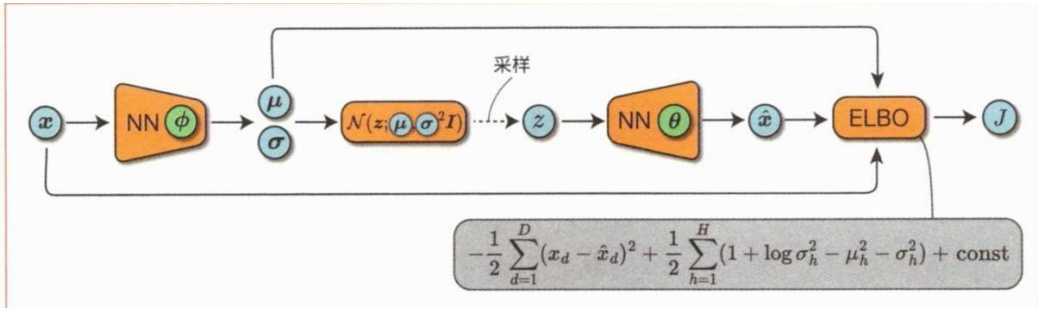
$$\hat{\mathbf{x}} = \text{NeuralNet}(\mathbf{z}; \boldsymbol{\theta})$$

$$J_1 \approx \log \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}, \mathbf{I})$$



ELBO完整计算

- 如果 $q(z)=\mathcal{N}(z;\mu_1,\sigma_1^2 I)$ 、 $p(z)=\mathcal{N}(z;\mu_2,\sigma_2^2 I)$,那么 (式子中的 H 是 z 的维度)。
  - $D_{\mathrm{KL}}(q \parallel p)=-\frac{1}{2}\sum_{h=1}^{H}\left(1+\log\frac{\sigma_{1,h}^2}{\sigma_{2,h}^2}-\frac{(\mu_{1,h}-\mu_{2,h})^2}{\sigma_{2,h}^2}-\frac{\sigma_{1,h}^2}{\sigma_{2,h}^2}\right)$
- 因此
  - $J_2=D_{\mathrm{KL}}\left(q_\phi(z \mid x) \parallel p(z)\right)=-\frac{1}{2}\sum_{h=1}^{H}\left(1+\log\sigma_h^2-\mu_h^2-\sigma_h^2\right)$
- 于是
  - $\mathrm{ELBO}(\mathbf{x};\boldsymbol{\theta},\phi)\approx-\frac{1}{2}\sum_{d=1}^{D}(x_d-\hat{x}_d)^2+\frac{1}{2}\sum_{h=1}^{H}\left(1+\log\sigma_h^2-\mu_h^2-\sigma_h^2\right)+$ const



实现

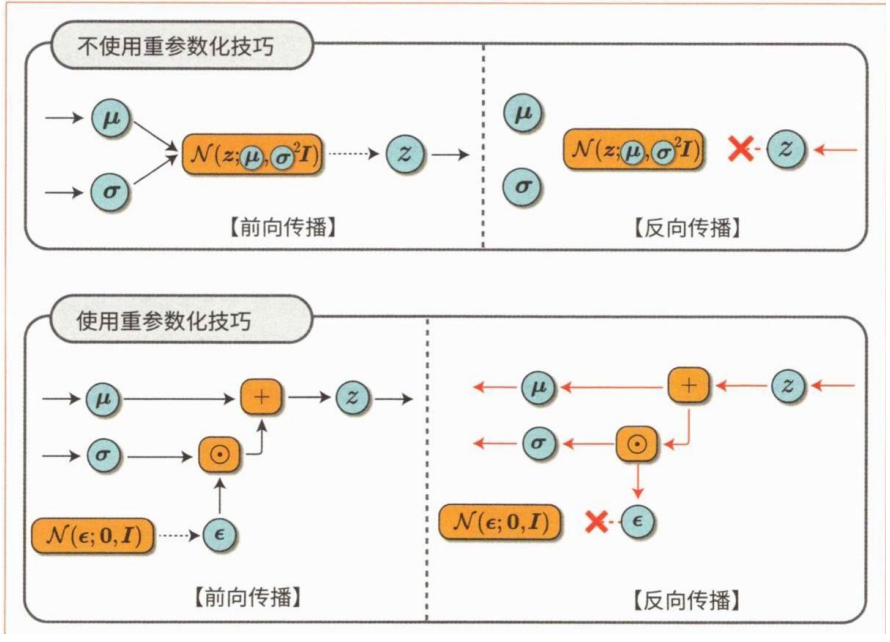
采样的可微分转换：重参数化技巧

➢ $z \sim \mathcal{N}(z; \boldsymbol{\mu}, \sigma^2 \mathbf{I})$ 的采样

$$\varepsilon \sim \mathcal{N}(\varepsilon; \mathbf{0}, \mathbf{I})$$

$$z = \mu + \sigma \odot \varepsilon$$

➢ ε 是基于均值向量为 0 、协方差矩阵为 I (单位矩阵)的标准正态分布采样的



VAE完整实现

- VAE 的 ELBO
  - $\mathrm{ELBO}(\mathbf{x};\boldsymbol{\theta},\phi) \approx -\frac{1}{2}\sum_{d=1}^{D}(x_d-\hat{x}_d)^2+\frac{1}{2}\sum_{h=1}^{H}\left(1+\log\sigma_h^2-\mu_h^2-\sigma_h^2\right)+\ \text{const}$
- VAE 的目标是最大化 ELBO。而在神经网络的训练过程中,通常会设置最小化的函数作为损失函数。因此, 我们将上式乘以 -1 后的式子作为损失函数
  - $\mathrm{Loss}(\mathbf{x};\boldsymbol{\theta},\phi) \approx \sum_{d=1}^{D}(x_d-\hat{x}_d)^2-\sum_{h=1}^{H}\left(1+\log\sigma_h^2-\mu_h^2-\sigma_h^2\right)$