Advanced Talk to Variational Autoencoder (VAE)

Jingbo Xia

Huazhong Agricultural University

xiajingbo.math@gmail.com

2026 年 4 月 12 日

目录 I

1. 从 Auto-encoder 到 VAE 3
   - 资源与准备 4
   - Autoencoder— representative learning / dimension reduction 5
   - VAE — 基于 Variational inference 7
2. VAE 建模 10
   - Objective 的形成 11
3. VAE 训练 22
   - Objective: KL divergence 23
   - Objective: Expectation of logP(X/z) 29
   - Optimization: E-M 算法框架中的梯度计算 30
   - Optimization: Re-parameterization trick 31
4. VAE Implementation 35
   - VAE 的多种 training strategy 36
   - VAE Implementation 中的痛点：Posterior Collapse 42
5. 太慢了？VAE 中的 Stochastic Variational Inference 43
6. 梯度的估计 55
   - Score function/ log-derivative trick 使用了 Monte Carlo estimate 56
   - Re-parameteriazation trick 使得 back-propagation 成为可能 57

Outline

1. 从 Auto-encoder 到 VAE 3
   - 资源与准备 4
   - Autoencoder— representative learning / dimension reduction 5
   - VAE — 基于 Variational inference 7
2. VAE 建模 10
   - Objective 的形成 11
3. VAE 训练 22
   - Objective: KL divergence 23
   - Objective: Expectation of logP(X/z) 29
   - Optimization: E-M 算法框架中的梯度计算 30
   - Optimization: Re-parameterization trick 31
4. VAE Implementation 35
   - VAE 的多种 training strategy 36
   - VAE Implementation 中的痛点：Posterior Collapse 42
5. 太慢了？VAE 中的 Stochastic Variational Inference 43
6. 梯度的估计 55
   - Score function/ log-derivative trick 使用了 Monte Carlo estimate 56
   - Re-parameteriazation trick 使得 back-propagation 成为可能 57

1 从 Auto-encoder 到 VAE 3
- 资源与准备 4
- Autoencoder— representative learning / dimension reduction 5
- VAE — 基于 Variational inference 7

2 VAE 建模 10

3 VAE 训练 22

4 VAE Implementation 35

5 太慢了? VAE 中的 Stochastic Variational Inference 43

6 梯度的估计 55

资源与文献阅读

我们主要参考的文献:

1. Hinton 在 2006 年发表的 Auto-encoder 论文 [1].
2. Kingma 和 Max 在 2013 年发表的第一篇 VAE 论文 [2].
3. 一篇易于阅读的 VAE 教程 [3].
4. Bohan Li 等人在 2019 年发表的关于 VAE 训练中 “posterior collapse” 的论文 [4].
5. 

[5]

[1] Hinton, Geoffrey E., and Ruslan R. Salakhutdinov. "Reducing the dimensionality of data with neural networks." science 313, no. 5786 (2006): 504-507.
[2] Kingma, Diederik P., and Max Welling. "Auto-encoding variational bayes." arXiv preprint arXiv:1312.6114 (2013).
[3] Doersch, Carl. "Tutorial on variational autoencoders." arXiv preprint arXiv:1606.05908 (2016).
[4] Li, Bohan, Junxian He, Graham Neubig, Taylor Berg-Kirkpatrick, and Yiming Yang. "A Surprisingly Effective Fix for Deep Latent Variable Modeling of Text." arXiv preprint arXiv:1909.00868 (2019).
[5] DeepBayes 2019. https://github.com/bayesgroup/deepbayes-2019.

1 从 Auto-encoder 到 VAE 3
- 资源与准备 4
- Autoencoder— representative learning / dimension reduction 5
- VAE — 基于 Variational inference 7

2 VAE 建模 10

3 VAE 训练 22

4 VAE Implementation 35

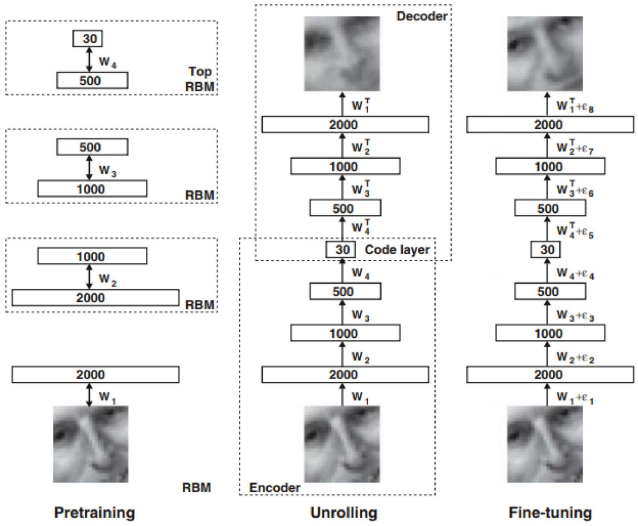
5 太慢了? VAE 中的 Stochastic Variational Inference 43

6 梯度的估计 55

Hinton 的 "Autoencoder"

高维数据可以通过训练一个具有较小中心层的多层神经网络（用于重构高维输入向量）来转换为低维代码。梯度下降可用于微调这种 "autoencoder" 网络中的权重，但这只有在初始权重接近良好解决方案时才有效。我们描述了一种初始化权重的有效方法，该方法允许深度 autoencoder 网络学习低维代码，作为一种 dimension reduction（降维）工具，其效果比主成分分析（principal components analysis）好得多。[a]

[a] http://www-public.imtbs-tsp.eu/~gibson/Teaching/Teaching-ReadingMaterial/HintonSalakhutdinov06.pdf



注意：

1. PCA 的非线性推广 (从维度约简的角度而言)。
2. 使用自适应的多层 "encoder/编码" 网络将高维数据 X 转换为低维代码 Z。
3. 一个类似的 "decoder/解码" 网络从代码中恢复数据。
4. 从两个网络中的随机权重开始，它们可以通过最小化原始数据与其重构之间的差异来一起训练。(利用重构误差)
5. 所需的梯度很容易通过使用链式法则获得，首先通过 decoder 网络，然后通过 encoder 网络反向传播误差导数。

——该模型被称为 "Autoencoder"。

1 从 Auto-encoder 到 VAE 3
- 资源与准备 4
- Autoencoder— representative learning / dimension reduction 5
- VAE — 基于 Variational inference 7

2 VAE 建模 10

3 VAE 训练 22

4 VAE Implementation 35

5 太慢了？VAE 中的 Stochastic Variational Inference 43

6 梯度的估计 55

VAE — 基于 Variational inference I

"...VAE 已经成为无监督学习（unsupervised learning）复杂分布的最受欢迎的方法之一。VAE 具有吸引力，因为它们建立在标准函数逼近器（neural networks）之上，并且可以使用stochastic gradient descent 进行训练。" (Wikipedia)

> 我们以 VAE 算法的建模为例，熟悉隐变量模型、EM 算法等算法概念及其数学符号化阐释。

VAE — 基于 Variational inference II

> 我们以 VAE 算法的建模为例，熟悉隐变量模型、EM 算法等算法概念及其数学符号化阐释。

- 在 VAE 中，引入 latent variable z 遵循了 Bayesian inference 框架中的老技巧。(隐变量)
- Expectation maximization (EM 模型) 用于求解 VAE。
- 像 Autoencoder 一样，VAE 中的 z 扮演着数据 X 的低维代表的角色。
- $q(\cdot)$ 是一个 Variational function，与 Variational inference 中使用的符号相同。

VAE — 基于 Variational inference III

VAE 的基础
VAE 处理分布 $p(X)$ 的模型，该分布定义在某个潜在的高维空间 $\mathcal{X}$ 中的数据点 X 上。

VAE 的基础:

1. 高维空间 $\mathcal{Z}$ 中的 latent variables z。
2. 人们可以根据定义在 $\mathcal{Z}$ 上的某个概率密度函数 (PDF) $p(z)$ 轻松地对 z 进行抽样。
3. 我们有一族确定性函数 $f(z;\theta)$，由某个空间 Θ 中的向量 θ 参数化，其中 $f\colon \mathcal{Z}\times\Theta\to\mathcal{X}$。$f$ 是一个确定性函数，如果 z 是随机的且 θ 是固定的，它就是 $\mathcal{X}$ 中的一个随机变量。
4. 我们希望优化 θ，使得我们可以从 $p(z)$ 中对 z 进行抽样，并且以很高的概率，$f(z;\theta)$ 将会像我们数据集中的 X 一样。

Outline

1 从 Auto-encoder 到 VAE 3
- 资源与准备 4
- Autoencoder— representative learning / dimension reduction 5
- VAE — 基于 Variational inference 7

2 VAE 建模 10
- Objective 的形成 11

3 VAE 训练 22
- Objective: KL divergence 23
- Objective: Expectation of logP(X/z) 29
- Optimization: E-M 算法框架中的梯度计算 30
- Optimization: Re-parameterization trick 31

4 VAE Implementation 35
- VAE 的多种 training strategy 36
- VAE Implementation 中的痛点：Posterior Collapse 42

5 太慢了？VAE 中的 Stochastic Variational Inference 43

6 梯度的估计 55
- Score function/ log-derivative trick 使用了 Monte Carlo estimate 56
- Re-parameteriazation trick 使得 back-propagation 成为可能 57

1 从 Auto-encoder 到 VAE 3

2 VAE 建模 10
- Objective 的形成 11

3 VAE 训练 22

4 VAE Implementation 35

5 太慢了？VAE 中的 Stochastic Variational Inference 43

6 梯度的估计 55

VAE 建模—Objective 的形成 I

证据最大化.

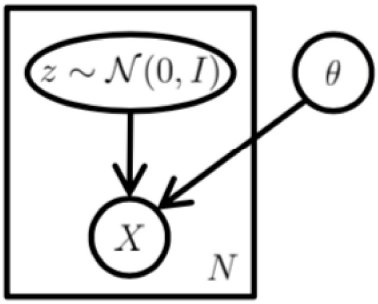
在 Generative model 的一般概念下，我们的目标是根据以下公式最大化整个生成过程中训练集中每个 X 的概率：

$$p(X) = \int p(X|z;\theta)p(z)dz. \tag{1}$$

VAE 建模—Objective 的形成 II

引入隐变量。

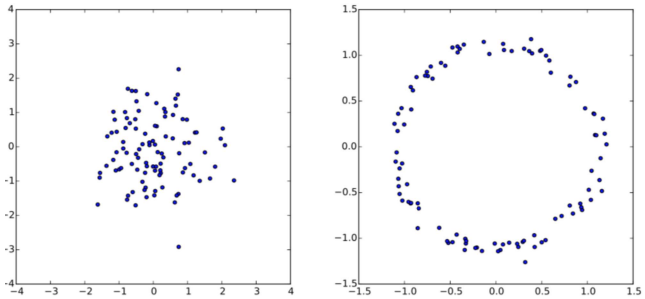
引入 latent variable z 是一个老技巧。图形模型如下。



为了求解方程 (1)，VAE 必须处理两个问题：如何定义 latent variables z (即，决定它们代表什么信息)，以及如何处理关于 z 的积分。VAE 对这两者都给出了明确的答案。

VAE 建模—Objective 的形成 III

理解 f存在的合理性，以及 z可以从一个标准分布中抽样获得.



例如，假设我们想要构建一个 2D 随机变量，其值位于一个圆环上。如果 z 是 2D 且 normally distributed ($z \sim \mathcal{N}(0, 1)$)，则 $f(z; \theta) := g(z) = z/10 + z/||z||$ 大致呈圆环状，如图所示。

VAE 建模—Objective 的形成 IV

理解如何通过神经网络获得 z的抽样，并且刻画 $p(X|z; \theta)$.

因此，如果提供强大的 function approximators，我们可以简单地学习一个函数，该函数将我们独立的、normally-distributed z 值映射到模型可能需要的任何 latent variables，然后将这些 latent variables 映射到 X。

1. 在大多数情况下，我们实际上不知道 $f(z, \theta)$ 是什么。
2. 将 $f(z, \theta)$ 视为一个 NN。
3. 从... 中抽样 X

$$p(X|z; \theta) = \mathcal{N}(X|f(z; \theta), \sigma^2 * I). \tag{2}$$

VAE 建模—形成 objective

如果 $f(z;\theta)$ 是一个多层 neural network，那么我们可以想象该网络使用其前几层将 normally distributed z 以完全正确的统计特征映射到潜在值（如数字标识、笔画粗细、角度等）。然后它可以使用后面的层将这些潜在值映射到一个完全渲染的数字。一般来说，我们不需要担心确保这种潜在结构的存在。如果这种潜在结构有助于模型准确地重构（即最大化似然）训练集，那么网络将在某一层学习到该结构。

现在剩下的就是 [6]

$$\text{最大化方程 (1)}: p(X) = \int p(X|z;\theta)p(z)\,dz$$
$$\text{其中} p(z) = \mathcal{N}(0, I).$$

[6]我们还没使用 variational... 如果我们想使用它，该怎么做?

VAE 建模—形成 objective

追加要求：让 z的抽样生成，更加有利于数据重构.

在实践中，对于大多数 z，$p(X|z)$ 将接近于零，因此对我们估计 $p(X)$ 几乎没有贡献。[7]

Variational autoencoder 背后的核心思想是尝试对可能产生 X 的 z 的值进行抽样，并仅根据这些值计算 $p(X)$。

这意味着我们需要一个新函数 $q(z|X)$ [8] 它可以接受一个 X 的值，并给我们一个关于可能产生 X 的 z 值的分布。[9]

[7]同时，请注意 $p(X|z)$ 通常是未知的。
[8]是的! $q(z|X)$ 是 variational，它是一个 posterior，旨在逼近 $p(z|X)$。与 variational inference 中的想法相同。
[9]并且，$q(z|X)$ 可以从 exponential family 分布中选择，特别是 Gaussians。

VAE 建模—形成 objective

所以:

1. 我们希望 $q(z|X)$ 下的 z 来自一个比 $P(z)$ 下的完整 z 空间小得多的空间。
2. 一如既往，我们的目标是最大化 evidence $p(X|z)$。显然，现在我们将其更新为

$$E_{z\sim q}[p(X|z)]. \tag{3}$$

3. 如何获得逼近 $p(z|X)$，同时最大化方程 (3) 的 $q(z|X)$? 通过定义 ELBO 和 KL-divergence，所有的技巧都与标准 VI 中相同。

VAE 建模—形成 objective

考虑 $q(z|X)$ 和 $p(z|X)$ 之间的 Kullback-Leibler divergence：

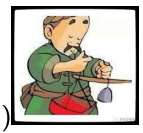
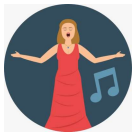
$$\mathbb{KL}[q(z|X)||p(z|X)] = E_{z\sim q}[\log q(z|X) - \log p(z|X)].$$

由于 $p(z|X) = p(X|z) \cdot p(z)/p(X)$，将 $p(X|z)$ 和 $p(X)$ 放入以下等式是一个老技巧：

$$\mathbb{KL}[q(z|X)||p(z|X)] = E_{z\sim q}[\log q(z|X) - \log p(X|z) - \log p(z)] + \log p(X).$$

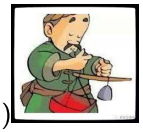
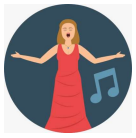
或者

$$\log p(X) - \mathbb{KL}[q(z|X)||p(z|X)] = E_{z\sim q}[\log p(X|z)] - \mathbb{KL}[q(z|X)||p(z)]. \quad (4)$$

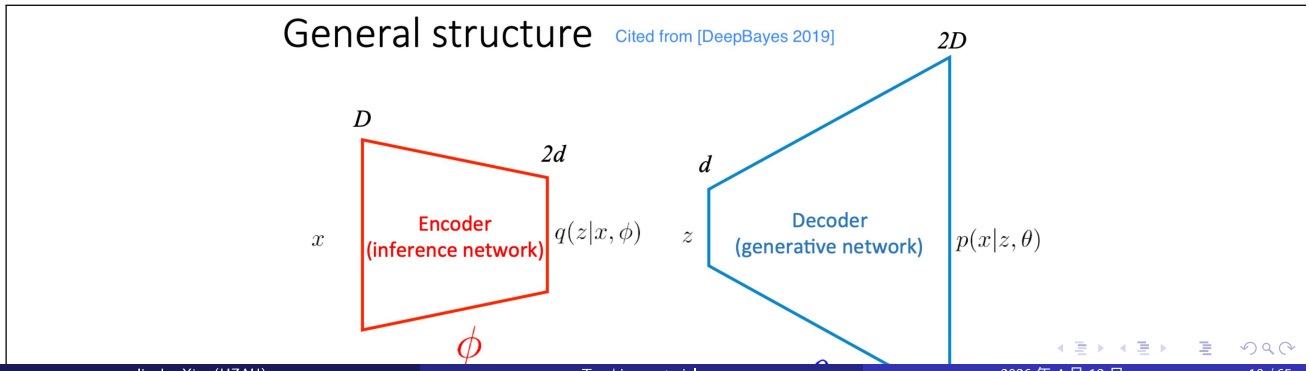
(Tuning $q(\cdot)$)  &  (Encoding:$p(z|X,\phi)$; Decoding: $p(X|z,\theta)$)

VAE 建模—形成 objective

Objective 和 NN 结构

(Tuning $q(\cdot)$)  &  (Encoding:$p(z|X,\phi)$; Decoding: $p(X|z,\theta)$)

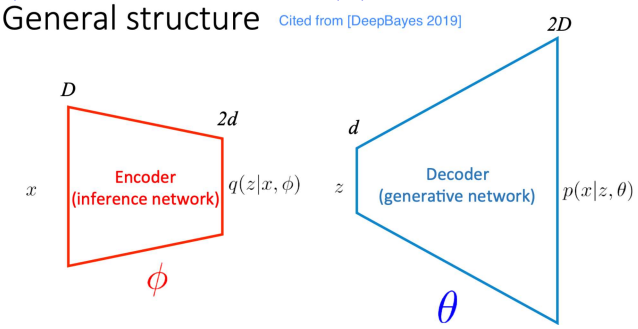
$$\text{Objective: } \mathcal{L}(\theta,\phi,X) = E_{q(z|X,\phi)}[\log p(X|z,\theta)] - \mathbb{KL}[q(z|X,\phi)||p(z)].^{10} \quad (5)$$



VAE 建模—形成 objective

备注：

一石二鸟：在寻找最佳 $q(z|X)$ 时，我们希望最大化 $\log p(X)$，同时，我们希望最小化 $\mathbb{KL}[q(z|X)||p(z|X)]$。



为什么称为 VAE?

1. "Variational"：函数调优 $q(z|X,\phi)$ 并抽取 z。
2. "Encoder"：$q(z|X,\phi)$。
   一个推理网络计算 $q(z|X,\phi)$，并对 2 维均值 $\mu(X)$ 和协方差 $\Sigma(X)$ 进行抽样。z 是根据 $q(z|X) = \mathcal{N}(\mu(X),\Sigma(X))$ 抽取的。
3. "Decoder"：$p(X|z,\theta)$。
   在 NN 中学习 $f(z;\theta)$，使得 $p(X|z;\theta) = \mathcal{N}(X|f(z;\theta),\sigma^2 * I)$ (2)。

$$\text{Objective: } \mathcal{L}(\theta, \phi, X) = E_{q(z|X,\phi)}[\log p(X|z,\theta)] - \mathbb{KL}[q(z|X,\phi)||p(z)]. \quad (5)$$

1. 观测值 $X \in D$，D 是训练数据集。
2. 对 evidence $p(X)$ 是一般且共同的目的。
3. 从 NN 训练的角度来看，它是最大化 $p(X)$ 的期望，即 $E_{X\sim D}[p(X)]$。
4. 引入 z 是一种贝叶斯技巧，并且 $p(X) = \int p(X|z;\theta)p(z)\,dz$ (1)。
5. $p(X|z)$ 被计算为 $p(X|z;\theta) = \mathcal{N}(X|f(z;\theta), \sigma^2 * I)$ (2)。它在 NN 中被视为 “Decoder” 层，并且 $f(z;\theta)$ 是通过 NN 学习的。
6. 为了选择更好的 z，对 $z \sim q(z|X)$ 进行抽样，并希望 $q(z|X)$ 逼近 posterior $p(z|X)$。
7. 因此，$q(z|X)$ 通过 $q(z|X) = \mathcal{N}(\mu(X), \Sigma(X))$ 进行抽样，这个过程被称为 “Encoding”。

Outline

1 从 Auto-encoder 到 VAE 3
- 资源与准备 4
- Autoencoder— representative learning / dimension reduction 5
- VAE — 基于 Variational inference 7

2 VAE 建模 10
- Objective 的形成 11

3 VAE 训练 22
- Objective: KL divergence 23
- Objective: Expectation of logP(X/z) 29
- Optimization: E-M 算法框架中的梯度计算 30
- Optimization: Re-parameterization trick 31

4 VAE Implementation 35
- VAE 的多种 training strategy 36
- VAE Implementation 中的痛点：Posterior Collapse 42

5 太慢了？VAE 中的 Stochastic Variational Inference 43

6 梯度的估计 55
- Score function/ log-derivative trick 使用了 Monte Carlo estimate 56
- Re-parameteriazation trick 使得 back-propagation 成为可能 57

1 从 Auto-encoder 到 VAE 3

2 VAE 建模 10

3 VAE 训练 22
- Objective: KL divergence 23
- Objective: Expectation of logP(X/z) 29
- Optimization: E-M 算法框架中的梯度计算 30
- Optimization: Re-parameterization trick 31

4 VAE Implementation 35

5 太慢了？VAE 中的 Stochastic Variational Inference 43

6 梯度的估计 55

Objective: KL divergence

因此，现在是优化方程 (5) 中 objective 的时候了。

$$\mathcal{L}(\theta, \phi, X) = E_{q(z|X,\phi)}[\log p(X|z,\theta)] - \mathbb{KL}[q(z|X,\phi)||p(z)]. \quad (5)$$

首先，请注意

$$\mathbb{KL}(\mathcal{N}(\mu_0, \Sigma_0)||\mathcal{N}(\mu_1, \Sigma_1)) = \frac{1}{2}(Tr(\Sigma_1^{-1}\Sigma_0) + (\mu_1 - \mu_0)^T\Sigma_1^{-1}(\mu_1 - \mu_0) - k + \log(\frac{\det \Sigma_1}{\det \Sigma_0})), \quad (6)$$

[11]

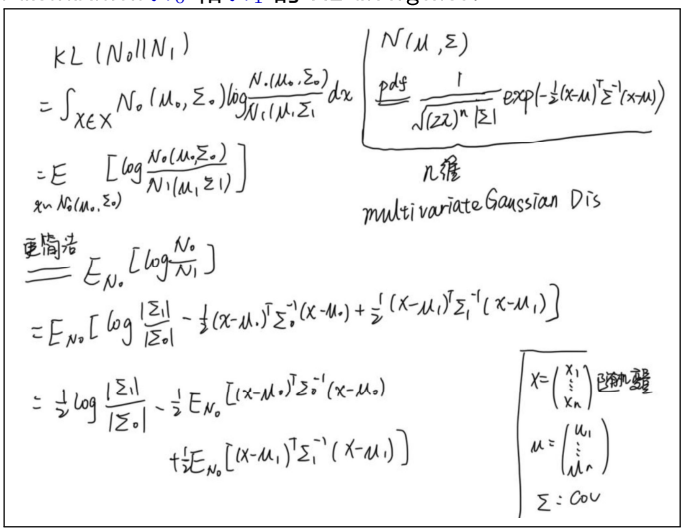
其中 k 是分布的维度。然后我们有

$$\mathbb{KL}(q(z|X)||p(z)) = \mathbb{KL}(\mathcal{N}(\mu(X), \Sigma(X)), \mathcal{N}(0, I)) = \frac{1}{2}(Tr(\Sigma(X) + (\mu(X)^T\mu(X) - k - \log(\det \Sigma(X))). \quad (7)$$

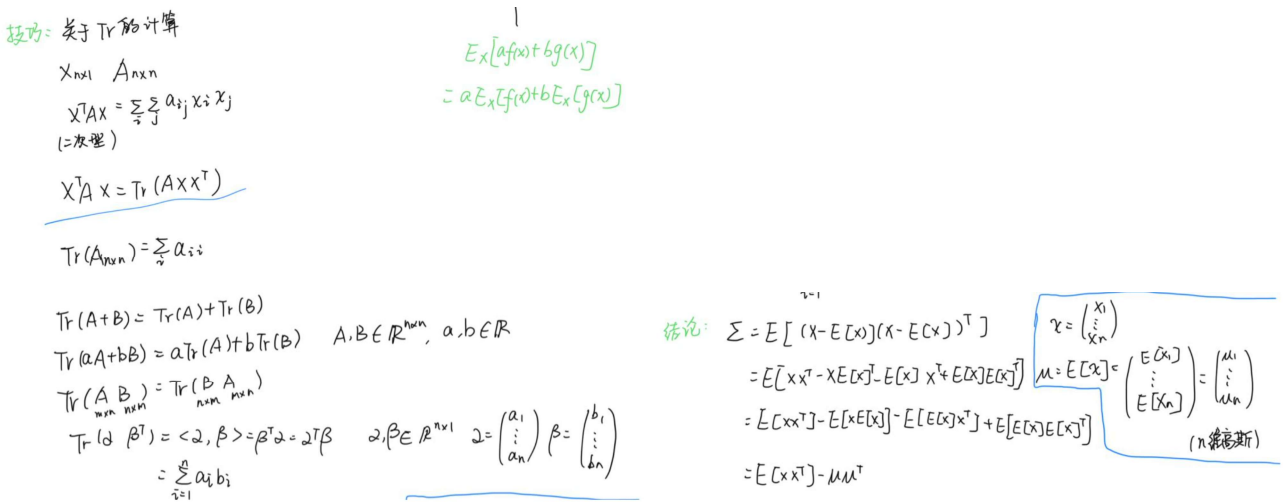
[11] 留作练习。

两个 Gaussians 的 relative entropy (KL divergence) I

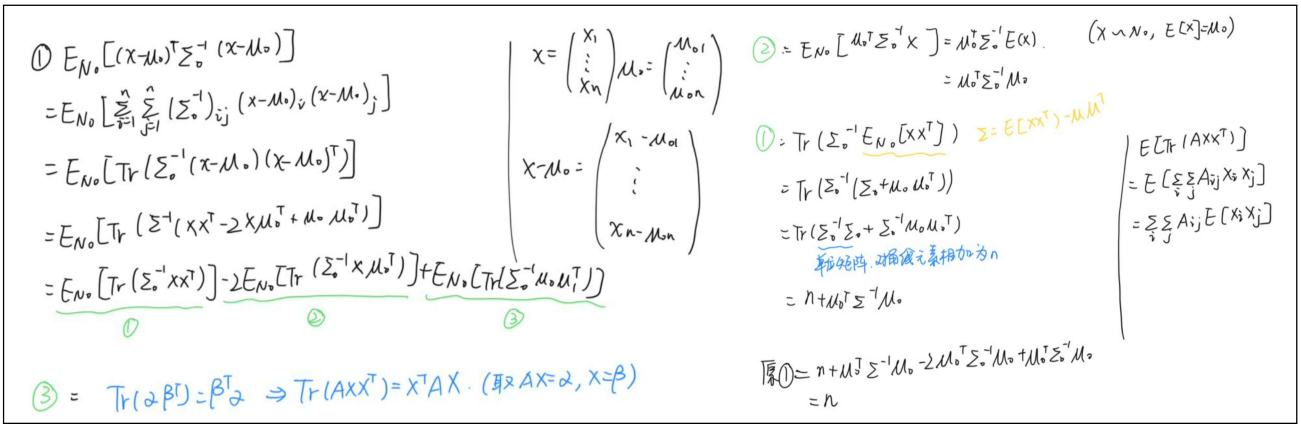
两个 multivariate Gaussian distribution $\mathcal{N}_0$ 和 $\mathcal{N}_1$ 的 KL-divergence：



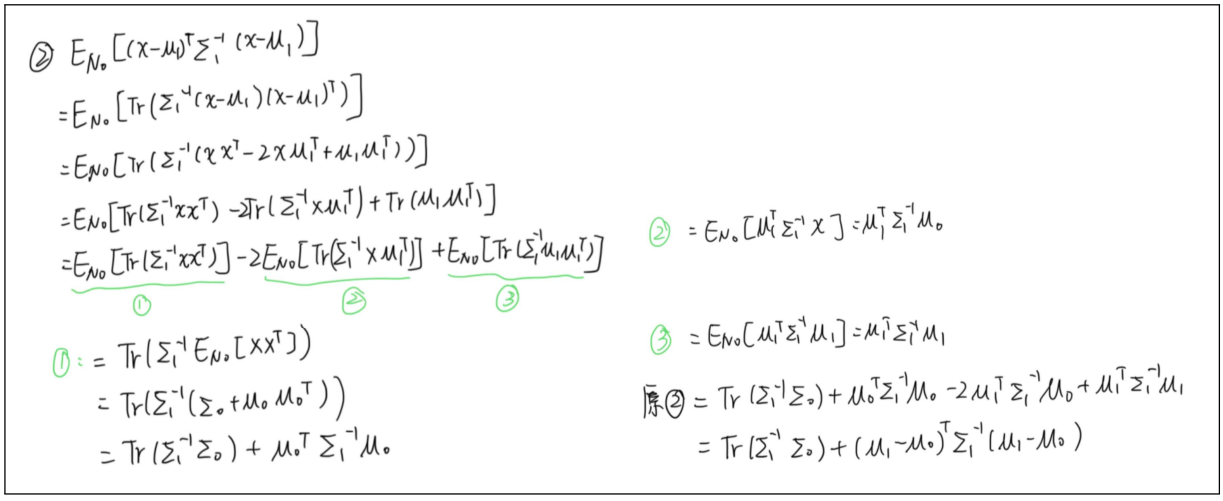
两个 Gaussians 的 relative entropy (KL divergence) II



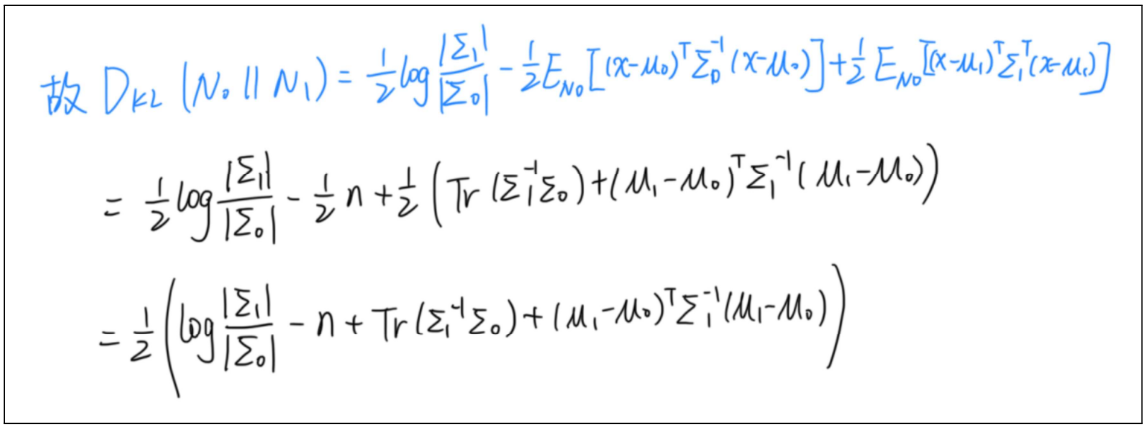
两个 Gaussians 的 relative entropy (KL divergence) III



两个 Gaussians 的 relative entropy (KL divergence) IV



两个 Gaussians 的 relative entropy (KL divergence) V



感谢 Sizhuo Ouyang 提供上述电子笔记的截图。

1 从 Auto-encoder 到 VAE 3
2 VAE 建模 10
3 VAE 训练 22
- Objective: KL divergence 23
- Objective: Expectation of logP(X/z) 29
- Optimization: E-M 算法框架中的梯度计算 30
- Optimization: Re-parameterization trick 31
4 VAE Implementation 35
5 太慢了? VAE 中的 Stochastic Variational Inference 43
6 梯度的估计 55

Objective: Expectation of logP(X/z)

NN 训练实际上是在从训练数据集 D 中抽样的不同 X 上的 gradient descent。要优化的完整方程是 $E_{x\sim D}[\mathcal{L}(\theta,\phi,X)]$，即：

$$E_{X\sim D}[E_{z\sim Q}[\log P(X|z)] - \mathbb{KL}(Q(z|X)||P(z))]. \tag{8}$$

在获得方程 (7) 后，剩下的是计算 $E_{X\sim D}[E_{z\sim Q}[\log P(X|z)]]$，得出

$$E_{X\sim D}[E_{\varepsilon\sim\mathcal{N}(0,I)}[\log P(X|z=\mu(X)+\Sigma^{\frac{1}{2}}(X)*\varepsilon)]]. \tag{9}$$

这个期望的易读形式实际上是

$$E_{X\sim D}[\text{从 VAE NN decoding 层输出的 } X \text{ 的均值}] \tag{10}$$

这是通过 cross entropy 实现的 [12]。

[12]给定集合上分布 p 和 q 的 cross entropy 为 $CE(p,q)=E_p[-\log q]$。离散形式：$CE(p,q)=-\sum_{x\in\mathcal{X}}p(x)\log q(x)$。
https://en.wikipedia.org/wiki/Cross_entropy

1 从 Auto-encoder 到 VAE 3
2 VAE 建模 10
3 VAE 训练 22
- Objective: KL divergence 23
- Objective: Expectation of logP(X/z) 29
- Optimization: E-M 算法框架中的梯度计算 30
- Optimization: Re-parameterization trick 31
4 VAE Implementation 35
5 太慢了? VAE 中的 Stochastic Variational Inference 43
6 梯度的估计 55

Optimization: E-M 算法框架中的梯度计算

EM 迭代在以下步骤之间交替执行 [13]：

1. 一个expectation (E) step：估计 latent variable 上的分布。
2. 和一个maximization (M) step：最大化关于参数的联合对数似然在 latent variables 上的期望。

[E-Step] 通过获得 ϕ 来估计 $q(z|X,\phi)$：

$$\phi^{(t+1)} = \phi^{(t)} + \eta_\phi \frac{\sum_i \partial \mathcal{L}(\theta^{(t)}, \phi, X_i)}{\partial \phi} \tag{11}$$

[M-Step] 通过获得 θ 来最大化 $E_{q(z|X,\phi)}[\log p(X,z|\theta)]$：[14]

$$\theta^{(t+1)} = \theta^{(t)} + \eta_\theta \frac{\sum_i \partial \mathcal{L}(\theta, \phi^{(t+1)}, X_i)}{\partial \theta} \tag{12}$$

[13] DeepBayes 2019, https://github.com/bayesgroup/deepbayes-2019

[14] 给定 ϕ^*，要最大化 $E_{q(z|X,\phi^*)}[\log p(X,z|\theta)]$ 只需最大化 $\mathcal{L}(\theta,\phi^*,X_i)$ 即可。留作家庭作业。

1 从 Auto-encoder 到 VAE 3

2 VAE 建模 10

3 VAE 训练 22
- Objective: KL divergence 23
- Objective: Expectation of logP(X/z) 29
- Optimization: E-M 算法框架中的梯度计算 30
- Optimization: Re-parameterization trick 31

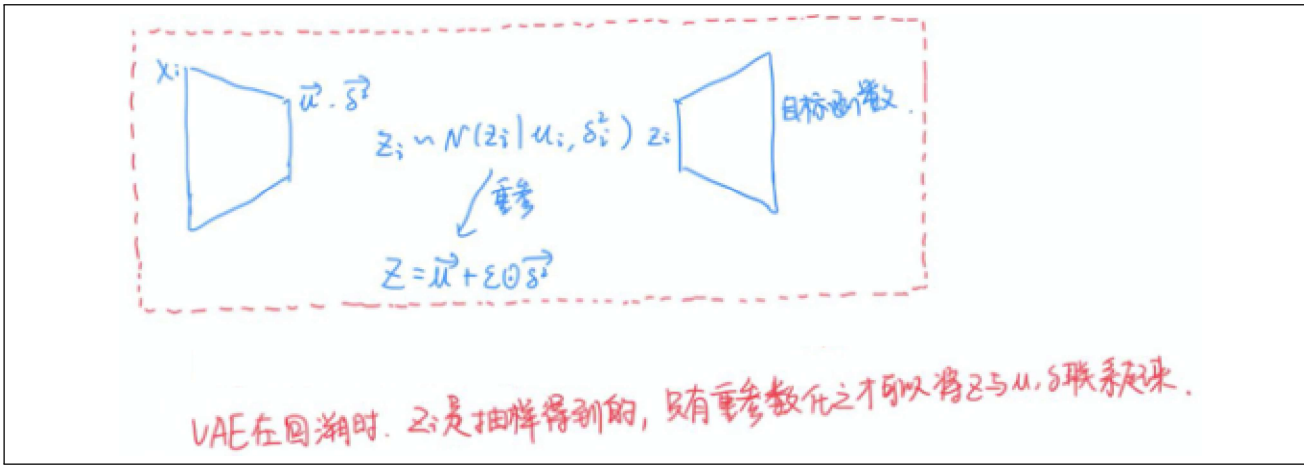
4 VAE Implementation 35

5 太慢了? VAE 中的 Stochastic Variational Inference 43

6 梯度的估计 55

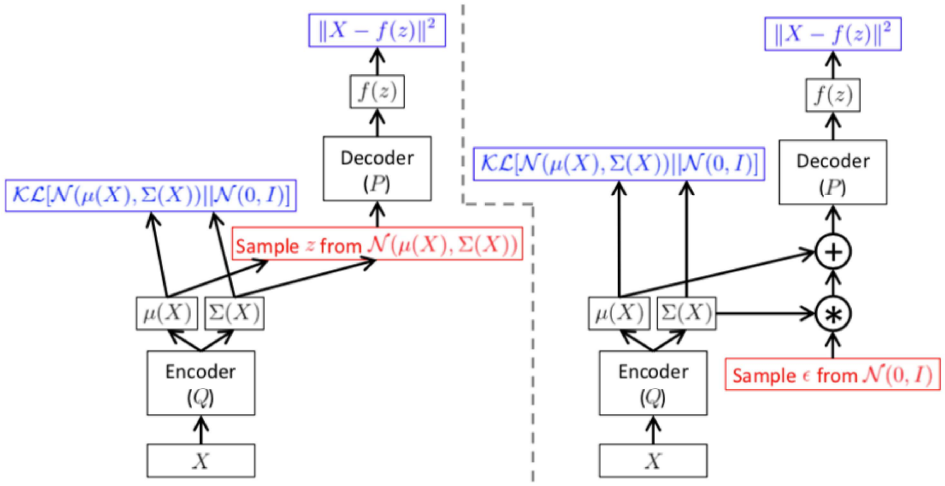
Optimization: Re-parameterization trick I

为什么进行 re-parameterization?



Optimization: Re-parameterization trick II

答：使梯度计算成为可能!



Optimization: Re-parameterization trick III

虚线右侧显示了 "re-parameterization trick"，这确保了可以进行梯度计算并应用 back-propagation，即：

$$z = \mu(X) + \Sigma^{\frac{1}{2}}(X) * \varepsilon,$$

其中 $\varepsilon \sim \mathcal{N}(0, I)$，并且 $\Sigma^{\frac{1}{2}}(X)$ 是 $\Sigma(X)$ 矩阵的平方根 [15]。

[15] https://en.wikipedia.org/wiki/Square_root_of_a_matrix

Optimization: Re-parameterization trick

编码示例 [16]。

```
def total_loss(_x_reconstruction, _x, mu, log_var):

        reconstruction_loss = F.binary_cross_entropy(_x_reconstruction, _x, reduction='sum')
        kl_div = - 0.5 * torch.sum(1 + log_var - mu.pow(2) - log_var.exp())
        total_loss = reconstruction_loss + kl_div
        # return reconstruction_loss
        # return kl_div
        return total_loss
```

[16] https://stats.stackexchange.com/questions/485488/should-reconstruction-loss-be-computed-as-sum-or-average-over-input-for-variatio

Outline

1. 从 Auto-encoder 到 VAE 3
   - 资源与准备 4
   - Autoencoder— representative learning / dimension reduction 5
   - VAE — 基于 Variational inference 7
2. VAE 建模 10
   - Objective 的形成 11
3. VAE 训练 22
   - Objective: KL divergence 23
   - Objective: Expectation of logP(X/z) 29
   - Optimization: E-M 算法框架中的梯度计算 30
   - Optimization: Re-parameterization trick 31
4. VAE Implementation 35
   - VAE 的多种 training strategy 36
   - VAE Implementation 中的痛点：Posterior Collapse 42
5. 太慢了？VAE 中的 Stochastic Variational Inference 43
6. 梯度的估计 55
   - Score function/ log-derivative trick 使用了 Monte Carlo estimate 56
   - Re-parameteriazation trick 使得 back-propagation 成为可能 57

1. 从 Auto-encoder 到 VAE 3
2. VAE 建模 10
3. VAE 训练 22
4. VAE Implementation 35
   - VAE 的多种 training strategy 36
   - VAE Implementation 中的痛点：Posterior Collapse 42
5. 太慢了？VAE 中的 Stochastic Variational Inference 43
6. 梯度的估计 55

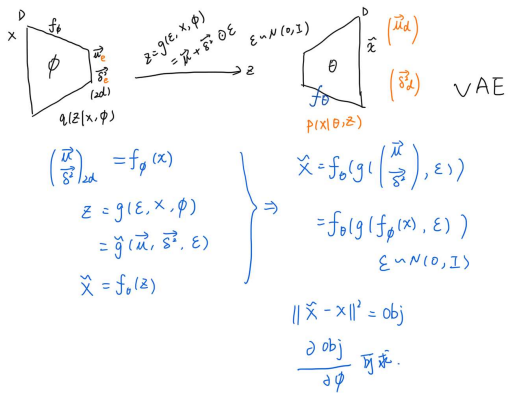
VAE 的多种 training strategy I

实际上，我们可以通过考虑 reconstruction loss：$||\tilde{X} - X||^2$，来训练一个非常简单的网络。

注意 1：
观察结果表明 re-parameterization 有助于计算损失函数的梯度。

注意 2：
而这实际上是一个 Auto-encoder 的想法。

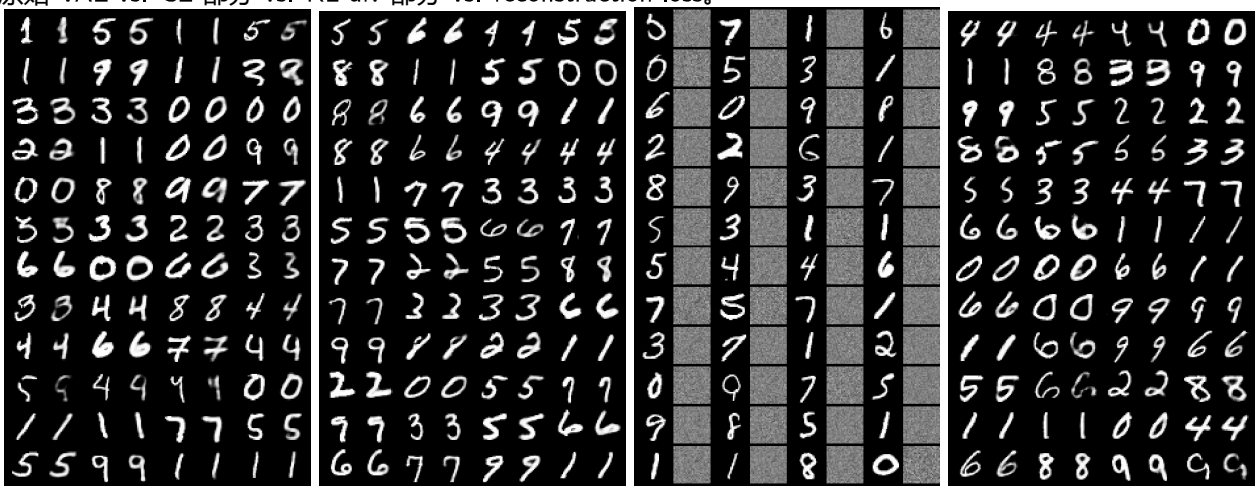
所以，我们使用不同的损失函数并比较其性能。



- 原始 VAE 中的损失函数 (5)：$\mathcal{L}(\theta, \phi, X) = E_{q(z|X,\phi)}[\log p(X|z,\theta)] - \mathbb{KL}[q(z|X,\phi)||p(z)]$，
- (5) 中的 Cross entropy 部分：$E_{q(z|X,\phi)}[\log p(X|z,\theta)]$，
- (5) 中的 KL-divergence 部分：$-\mathbb{KL}[q(z|X,\phi)||p(z)]$，
- reconstruction loss：$||\tilde{X} - X||^2$。

VAE 的多种 training strategy II

原始 VAE vs. CE 部分 vs. KL-div 部分 vs. reconstruction loss。

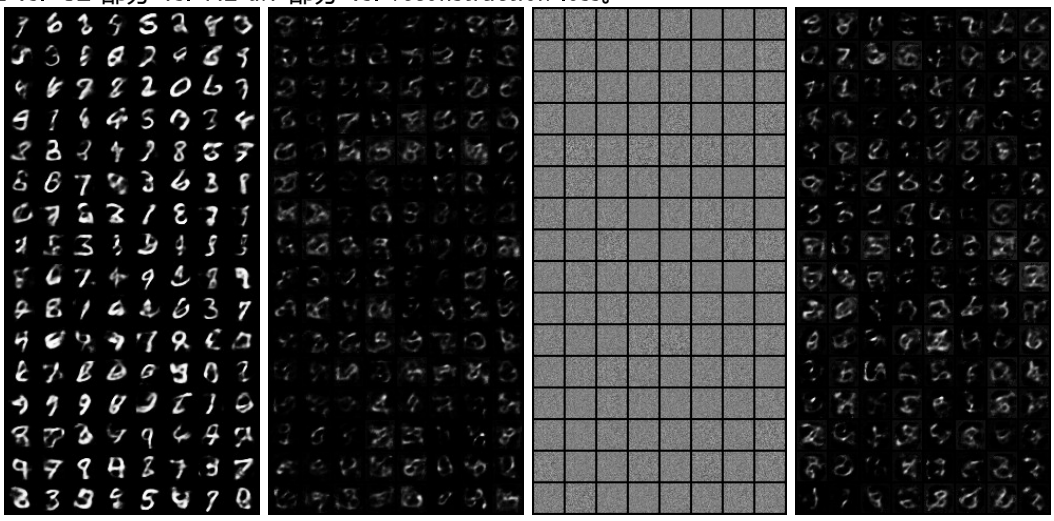


观察重构的图形。
第一列是输入，右边的是生成的内容。

VAE 的多种 training strategy III

VAE 的多种 training strategy IV

原始 VAE vs. CE 部分 vs. KL-div 部分 vs. reconstruction loss。



观察 z 如何生成新图形。

1 从 Auto-encoder 到 VAE 3
2 VAE 建模 10
3 VAE 训练 22
4 VAE Implementation 35
- VAE 的多种 training strategy 36
- VAE Implementation 中的痛点：Posterior Collapse 42
5 太慢了？VAE 中的 Stochastic Variational Inference 43
6 梯度的估计 55

VAE Implementation 中的痛点：Posterior Collapse

Bohan Li 等人在 2019 年发表的关于 VAE 训练中的“posterior collapse”的论文，

Li, Bohan, Junxian He, Graham Neubig, Taylor Berg-Kirkpatrick, and Yiming Yang. "A Surprisingly Effective Fix for Deep Latent Variable Modeling of Text." arXiv preprint arXiv:1909.00868 (2019).

Outline

1 从 Auto-encoder 到 VAE 3
- 资源与准备 4
- Autoencoder— representative learning / dimension reduction 5
- VAE — 基于 Variational inference 7

2 VAE 建模 10
- Objective 的形成 11

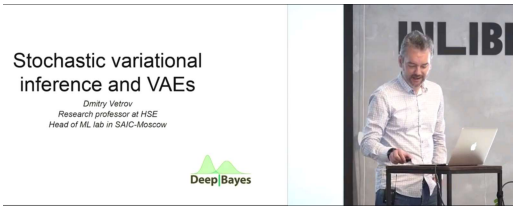
3 VAE 训练 22
- Objective: KL divergence 23
- Objective: Expectation of logP(X/z) 29
- Optimization: E-M 算法框架中的梯度计算 30
- Optimization: Re-parameterization trick 31

4 VAE Implementation 35
- VAE 的多种 training strategy 36
- VAE Implementation 中的痛点：Posterior Collapse 42

5 太慢了？VAE 中的 Stochastic Variational Inference 43

6 梯度的估计 55
- Score function/ log-derivative trick 使用了 Monte Carlo estimate 56
- Re-parameteriazation trick 使得 back-propagation 成为可能 57

VAE 中的 Stochastic Variational Inference



https://github.com/bayesgroup/deepbayes-2019

VAE 中的 Stochastic variational inference

整体思路

Stochastic variational inference and VAE

[DeepBayes 2019]

原因：① 传统的VI算不动，尤其参数多，SVI比VI快

② "随机"并非新 idea (GD ⇝ SGD)

③ 更高的角度看EM，VI如何在EM框架下实现
VI和SVI如何做VAE求解

④ 上述过程中，gradient计算，SVI中 gradient计算的细节

⑤ Gradient 迭代清楚，则设计NN，加入embedding，training OK

VAE 中的 Stochastic variational inference

变量设置

- 连续的隐变量 z (Bayesian inference)

  $z: \{z_1, z_2, \cdots\}$ 隐变量

  x：观测量

  $p(x) = \int p(x, z)\, dz$ → 无中生有

  Evidence

- 离散的隐变量

  x：观测量

  z：Catent variables

  $p(x) = \sum_i p(x, z_i)$

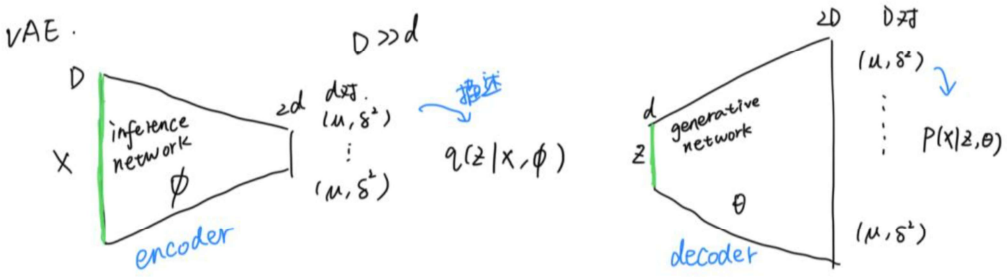
$$p(x) = \int p(x,z)dz = \int p(x|z)p(z)dz$$

E-M 算法中：E-step："估计一个与隐变量有关的分布 $q(z|x;\phi)$"

(Estimate) z 无中生有.

在VI中，这个分布往往用来逼近 $p(z|x;\theta)$ 很难存在

$$q(z_i) \approx p(z_i|x;\theta) = \frac{p(z_i,x_i)}{p(x_i)} = \frac{p(x_i|z_i;\theta)\cdot p(z_i|\theta)}{\int p(x_i|z_i,\theta)\cdot p(z_i|\theta)dz_i}$$



$$\prod_{i=1}^{n} p(x_i|z_i,\theta)p(z_i) = p(x,z|\theta)$$

$$= \prod_{i=1}^{n}\left(\prod_{j=1}^{D} N(x_{ij}|\mu_j(z_i), \delta_j^2(z_i))\right)\cdot N(z_i|0,I)$$

[E-step]："估计一个与隐变量有关的分布"

$$q(z) = \prod_{i=1}^{n} q(z_i) = \prod_{i=1}^{n} p(z_i|x_i,\theta)$$

original

$$= \prod_{i=1}^{n} \frac{p(x_i|z_i,\theta)p(z_i)}{\int p(x_i|z_i,\theta)p(z_i)dz_i}$$

decoder

[VI] $q(z_i|x_i,\phi) \approx p(z_i|x_i,\theta)$

$$q(z_i|x_i,\phi) = \prod_{j=1}^{d} N(z_{ij}|\mu_j(x_i,\phi), \delta_j^2(x_i,\phi))$$

[ELBO for VAE]

在做 VI 时, $q(z_i|x_i,\phi) = \arg\min_q KL(q(z|x,\phi)|p(z|x,\theta))$

等同于最大化 ELBO

$$L(\phi,\theta) = E_{q(z|x,\phi)}[\log \frac{p(x,z|\theta)}{q(z|x,\phi)}]$$

$= E_{q(z|x,\phi)}[\log p(x,z|\theta)] - E_{q(z|x,\phi)}[\log q(z|x,\phi)]$

Optimization w.r.t. θ

$$\mathcal{L}(\phi,\theta) = \int q(Z|X,\phi)\log\frac{p(X|Z,\theta)p(Z)}{q(Z|X,\phi)}dZ$$

- Stochastic gradients w.r.t. θ can be obtained quite straightforwardly
- Mini-batching:

$$\nabla_\theta\mathcal{L}(\phi,\theta) = \nabla_\theta\sum_{i=1}^{n}\int q(z_i|x_i,\phi)\log\frac{p(x_i|z_i,\theta)p(z_i)}{q(z_i|x_i,\phi)}dz_i =$$

$$\sum_{i=1}^{n}\int q(z_i|x_i,\phi)\nabla_\theta\log p(x_i|z_i,\theta)dz_i \approx n\int q(z_i|x_i,\phi)\nabla_\theta\log p(x_i|z_i,\theta)dz_i, \quad i\sim\mathcal{U}\{1,\dots$$

- Monte-Carlo estimation

$$n\int q(z_i|x_i,\phi)\nabla_\theta\log p(x_i|z_i,\theta)dz_i \approx n\nabla_\theta\log p(x_i|z_i^*,\theta), \quad z_i^*\sim q(z_i|x_i,\phi)$$

对 ϕ 求偏导：

$$\int q(z|x,\phi)\log p(x|z,\theta)dz$$

$$\Rightarrow \frac{\partial}{\partial\phi}\int q(z|x,\phi)\log p(x|z,\theta)dz$$

$$= \sum_{i=1}^{n}\frac{\partial}{\partial\phi}\int q(z_i|x_i,\phi)\log p(x_i|z_i,\theta)dz_i$$

(mini batching) $\approx n\frac{\partial}{\partial\phi}\int q(z_i|x_i,\phi)\log p(x_i|z_i,\theta)dz_i$

再回到计算 ϕ 偏导：

即：$n\frac{\partial}{\partial\phi}\int q(z_i|x_i,\phi)\log p(x_i|z_i,\theta)dz_i$

$z_i = g(\varepsilon, x_i, \phi) = \delta(x_i,\phi)\varepsilon + \mu(x_i,\phi)$

$= n\frac{\partial}{\partial\phi}\int r(\varepsilon)\log p(x_i|g(\varepsilon,x_i,\phi),\theta)d\varepsilon$

$\approx n\frac{\partial}{\partial\phi}\log p(x_i|g(\hat{\varepsilon},x_i,\phi),\theta)$ $\quad \hat{\varepsilon}\sim r(\varepsilon)$

VAE: final algorithm

- Input: Training data X, dimension of latent space d
- Pick random $i \sim \mathcal{U}\{1,\ldots,n\}$ and compute stochastic gradients of ELBO w.r.t. θ and ϕ
  - Differentiate w.r.t. θ

$$\text{stoch.grad}_\theta\mathcal{L}(\phi,\theta) = n\frac{\partial}{\partial\theta}\log p(x_i|z_i^*,\theta),$$

    where $z_i^* \sim q(z_i|x_i,\phi)$
  - Differentiate w.r.t. ϕ

$$\text{stoch.grad}_\phi\mathcal{L}(\phi,\theta) = n\frac{\partial}{\partial\phi}\log p(x_i|g(\hat{\epsilon},x_i,\phi),\theta) - \frac{\partial}{\partial\phi}KL(q(z_i|x_i,\phi)||p(z_i)),$$

    where $\hat{\epsilon}\sim r(\epsilon)$
- Update θ and ϕ according to selected stochastic optimization method

Outline

1 从 Auto-encoder 到 VAE 3
- 资源与准备 4
- Autoencoder— representative learning / dimension reduction 5
- VAE — 基于 Variational inference 7

2 VAE 建模 10
- Objective 的形成 11

3 VAE 训练 22
- Objective: KL divergence 23
- Objective: Expectation of logP(X/z) 29
- Optimization: E-M 算法框架中的梯度计算 30
- Optimization: Re-parameterization trick 31

4 VAE Implementation 35
- VAE 的多种 training strategy 36
- VAE Implementation 中的痛点：Posterior Collapse 42

5 太慢了？VAE 中的 Stochastic Variational Inference 43

6 梯度的估计 55
- Score function/ log-derivative trick 使用了 Monte Carlo estimate 56
- Re-parameteriazation trick 使得 back-propagation 成为可能 57

1 从 Auto-encoder 到 VAE 3
2 VAE 建模 10
3 VAE 训练 22
4 VAE Implementation 35
5 太慢了? VAE 中的 Stochastic Variational Inference 43
6 梯度的估计 55
- Score function/ log-derivative trick 使用了 Monte Carlo estimate 56
- Re-parameteriazation trick 使得 back-propagation 成为可能 57

Score function/ log-derivative trick

Score function/ log-derivative trick 使用了 Monte Carlo estimate!

假设 latent variables $Z \sim q(z|\phi)$，并且假设我们想要优化某个连续可微函数 $f(z)$ 关于分布参数 ϕ 的期望 $E_{q(z|\phi)}[f(z)]$。

我们现在的目标是计算 $E_{q(z|\phi)}[f(z)]$ 的导数，

$$\begin{aligned} \nabla_\phi E_{q(z|\phi)}[f(z)] &= \nabla_\phi \int_z q(z|\phi) f(z)\, dz = \int_z \nabla_\phi q(z|\phi) f(z)\, dz \\ &= \int_z q(z|\phi) f(z) \nabla_\phi \log q(z|\phi)\, dz \text{ (使用了 log-derivative trick。)} \\ &= E_{q(z|\phi)}[f(z) \log q(z|\phi)]. \end{aligned} \tag{13}$$

(13) 中的最后一步表明梯度可以通过 Monte Carlo estimate 进行计算。

缺点：巨大的 variance。

1 从 Auto-encoder 到 VAE 3
2 VAE 建模 10
3 VAE 训练 22
4 VAE Implementation 35
5 太慢了? VAE 中的 Stochastic Variational Inference 43
6 梯度的估计 55
- Score function/ log-derivative trick 使用了 Monte Carlo estimate 56
- Re-parameteriazation trick 使得 back-propagation 成为可能 57

Re-parameteriazation Trick 使得 Back-propagation 成为可能!

"reparameterization trick 很容易使用，...，但仍然相当具有限制性，因为它们排除了许多标准分布，例如 Gamma, Beta, Dirichlet..."。

以下公式参考了"Michael Figurnov, Shakir Mohamed, Andriy Mnih. Implicit Reparameterization Gradients, NIPS 2018."

假设我们想要优化某个连续可微函数 $f(z)$ 关于分布参数 ϕ 的期望 $E_{q(z|\phi)}[f(z)]$。

我们假设可以找到一个standardization function $S_\phi(z)$，当将其应用于来自 $q(z|\phi)$ 的样本时，可以消除其对分布参数的依赖。

$$S_\phi(z) = \varepsilon \sim q(\varepsilon), \quad z \sim S_\phi^{-1}(\varepsilon). \tag{14}$$

例如，单变量 $Z \sim N(z|\mu,\sigma)$，$S_\phi(z) = \varepsilon = \frac{z-\mu}{\sigma} \sim N(\varepsilon|0,1)$。
例如，在 VAE 中，Z 是多元 latent variable，$Z \sim N(z|\mu,\sigma)$，$S_{\mu,\sigma}^{-1}(\varepsilon) = \mu + \varepsilon \cdot \sigma$，并且 $\varepsilon \sim N(\varepsilon|0,I)$。

为了计算避免 standardization function 求逆的 re-parameterization 梯度，我们有

$$\nabla_\phi E_{q(z|\phi)}[f(z)] = E_{q(z|\phi)}[\nabla_z f(z) \nabla_\phi z]. \tag{15}$$

由于假设 $\nabla_z f(z)$ 是可计算的，例如，通过 VAE 中的解码网络，计算 $\nabla_\phi z$ 是一个关键问题。

如果将全梯度应用于等式 $S_\phi(z) = \varepsilon$，我们得到 $\nabla_z S_\phi(z) \nabla_\phi z + \nabla_\phi S_\phi(z) = 0$，并且得出：

$$\nabla_\phi z = -\frac{\nabla_\phi S_\phi(z)}{\nabla_z S_\phi(z)} \tag{16}$$

对于单变量分布，standardization function 由cumulative distribution function (CDF) 给出：
$S_\phi(z) = F(z|\phi) \sim Uniform(0,1)$。并且由 (16) 我们有

$$\nabla_\phi z = -\frac{\nabla_\phi F(z|\phi)}{q(z|\phi)}. \tag{17}$$

因此，计算隐式梯度只需要对 CDF 求导。

基于 Michael 等人的工作，使用 series expansion 来计算 Gamma 分布 $\Gamma(\alpha,1)$ 的 CDF：$\gamma(z,\alpha)$。

$$\gamma(z,\alpha)=\frac{\exp(-z)z^{\alpha}}{\Gamma(\alpha+1)}(1+\sum_{k=1}^{\infty}\frac{z^k}{(\alpha+1)(\alpha+2)\cdots(\alpha+r)})$$

$\gamma(z,\alpha)$ 的导数可以通过 hypergeometric function ${}_2F_2$ 获得：

$$\frac{\partial\gamma(z,\alpha)}{\partial\alpha}=\gamma(z,\alpha)(\log z-\psi(\alpha))+{}_2F_2(\alpha,\alpha;\alpha+1,\alpha+1;-z)\frac{z^{\alpha}}{\alpha\Gamma(\alpha+1)}, \tag{18}$$

其中 $\psi(\alpha)=(\log\Gamma(\alpha))'$ 是 digamma function，并且 ${}_2F_2(\alpha,\alpha;\alpha+1,\alpha+1;-z)=\sum_{k=0}^{\infty}\frac{\alpha^2}{(\alpha+k)^2}\frac{(-z)^k}{k!}$。

假设 $Z\sim Beta(\alpha,\beta)$，则假设 $z=z(z_1,z_2)=\frac{z_1}{z_1+z_2}$，并且 $z_1\sim\Gamma(\alpha,1)$，$z_2\sim\Gamma(\beta,1)$。此外，$\gamma_1(z_1,\alpha)$ 和 $\gamma_2(z_2,\beta)$ 分别是 $\Gamma(\alpha,1)$ 和 $\Gamma(\beta,1)$ 的 CDF。然后我们有

$$\begin{aligned}\frac{\partial z}{\partial\alpha}&=\frac{\partial z}{\partial z_1}\frac{\partial z_1}{\partial\alpha}+\frac{\partial z}{\partial z_2}\frac{\partial z_2}{\partial\alpha}\\ \frac{\partial z}{\partial\beta}&=\frac{\partial z}{\partial z_1}\frac{\partial z_1}{\partial\beta}+\frac{\partial z}{\partial z_2}\frac{\partial z_2}{\partial\beta},\end{aligned} \tag{19}$$

对于 $i=1$ 或 2，$\frac{\partial z_i}{\partial\alpha}$ 和 $\frac{\partial z_i}{\partial\beta}$ 是通过 (18) 计算的。

致谢参与讨论的同学们 I

09/20/2019

致谢参与讨论的同学们 II

07/21/2021

致谢参与讨论的同学们 III

09/12/2021

致谢参与讨论的同学们 IV

感谢参加研讨会的同学们的相关讨论。
感谢 Xinzhi Yao 提供 VAE 训练代码和图形示例。

2026 年 4 月 12 日