评论 Nijkamp 等人 2019 年的《剖析基于马尔可夫链蒙特卡洛的能量模型的最大似然学习》

潘旻琦

2020 年 4 月 4 日

1. 虽然传统理论预期了收敛，但 [Nijkamp et al., 2019] 发现其实在实践中很难实现收敛；[Nijkamp et al., 2019] 的实验非常丰富，但是缺乏理论层面的解释，理论的短板具体在哪里呢？另外 v_t 和 d_{s_t} 两个量的定义是作者的核心关注点，这两个量存在神奇的互相关和自相关性完全是通过实验发现的，没有理论解释，理论上可不可以解释这个现象？除了 v_t 和 d_{s_t} 还可以定义哪些量进行观察？定义新的量或许可以浮现新的有趣现象

2. CNN 能量函数除了 MCMC 还有别的方法取样吗？CNN 可以换成其他的神经网络吗？甚至可以换成无参模型吗？我认为训练能量函数本质上实在模拟一个长期记忆体，或许可以看看神经生物学人类的大脑是怎么实现记忆的，我猜不一定是一个 CNN 的形状的神经网络；但是换能量函数可能还是得保留可微的特性，否则怎么优化又是个问题

3. [Nijkamp et al., 2019] 用的 Langevin 只是众多 MH（Metropolis–Hastings）的一种，朱松纯 98 年的 FRAME [Zhu et al., 1998] 用的是另外一种 MH，叫 Gibbs Sampler，他的论文 [Zhu et al., 1998] 确实很严谨，各个量都有数学上的定义，非常值得学习；Gibbs Sampler 和 Langevin 的区别在于，Gibbs Sampler 为每个维度分别选择一个新样本（如 [Zhu et al., 1998] 的 Algorithm 2 ），而不是一次为所有维度选择样本，所以在计算上比较昂贵，而 Langevin 采样可以一次性整体采样；例如 100 张 $3 \times 32 \times 32$ 的彩色图片的总计 307200 个维度可以一次性同时更新

4. 计算机视觉领域里人们用 Langevin 采样的原因，我猜是这样的：把图片的每个像素想象成一粒悬浮在水中的花粉，物理学家 Langevin 认为每粒花粉受到两个力的作用：一个粘性阻力，这个力根据斯托克定律是 $6\pi\eta r\dot{x}$，在机器学习领域可以类比于能量函数 $U(X;\theta)$ 对 MCMC 的排斥力，一阶导 $\dot{x}$ 对应 $\frac{\partial}{\partial x}U(X;\theta)$，促使下一次随机游走向能量函数低的地方走；另一个力是水分子对花粉的波动的持续冲击力，

Langevin 认为是一个零均值的高斯过程，在机器学习领域可以类比于 MCMC 中的蒙特卡洛随机化；这个猜测还需要找更多 Langevin 方面的旧文献来证实

5. 看了源代码后发现 [Nijkamp et al., 2019] 的实验的 Langevin 实现是作者手撕的；作者在论文中也提到这个 Langevin 实现并没有进行动量更新和 MH 更新，这样会不会出问题？完整的 Langevin 实现会不会带来不同的结果？

6. [Nijkamp et al., 2019] 的实验 §4.1 有一些未明说的细节，我看了一下源代码，这里记录一下我看到的一些的细节：该实验假定每张图像的维度是 $2\times 1\times 1$，即只有两个通道、一个像素，且服从真实概率分布 $q(\mathbf{x}):\mathbb{R}^{2\times 1\times 1}\to[0,\infty)$；这个实验构造了两种多峰分布，一个是八汤圆分布，一个是四环分布；四环分布的定义从源代码反推可知

$$q(\mathbf{x})=\frac{1}{8\pi}\cdot\sum_{k=0}^{3}\frac{1}{k+1}\mathcal{N}\left(\|\mathbf{x}\|_2;\mu=k+1,\sigma^2=0.15^2\right) \tag{1}$$

八汤圆分布的定义从源代码反推可知

$$q(\begin{bmatrix}x_0\\x_1\end{bmatrix})=\frac{1}{8}\sum_{k=0}^{7}\mathcal{N}\left(\begin{bmatrix}x_1\\x_0\end{bmatrix};\mu=\begin{bmatrix}\cos(\frac{2\pi k}{8})\\\sin(\frac{2\pi k}{8})\end{bmatrix},\Sigma=\begin{bmatrix}0.15^2&0\\0&0.15^2\end{bmatrix}\right) \tag{2}$$

所训练的一族有参能量函数叫 $\mathrm{ToyNet}:\mathbb{R}^2\to\mathbb{R}$，其定义从源代码反推可知

$$\mathrm{ToyNet}\equiv C_{64,1}\circ R\circ C_{64,64}\circ R\circ C_{64,64}R\circ C_{32,64}\circ R\circ C_{2,32}$$

其中 $C_{i,j}$ 是一些从 $\mathbb{R}^{i\times 1\times 1}$ 到 $\mathbb{R}^{j\times 1\times 1}$ 的 1×1 卷积仿射变换，R 是一些负斜率恒为 0.05 的 Leaky ReLU 函数

7. [Nijkamp et al., 2019] 的实验 §4.3 也有一些未明说的细节，我看了一下源代码，这里记录一下我看到的一些的细节：该实验假定每张图像的维度是 $3\times 32\times 32$，即三个彩色通道、每个通道 32×32 个像素，且服从真实概率分布 $q(\mathbf{x}):\mathbb{R}^{3\times 32\times 32}\to[0,\infty)$，其定义从源代码反推可知

$$q(\mathbf{x})=\frac{1}{8189}\sum_{k=1}^{8189}\mathcal{N}\left(\mathbf{x};\mu=\mathrm{Flower}_k,\Sigma=I_{3\times 32\times 32}\right) \tag{3}$$

其中 Flower_k 是牛津 102 类花数据集的 8189 张图片的第 k 张的 $3\times 32\times 32$ 维向量；被训练的能量函数叫 $\mathrm{NonlocalNet}:\mathbb{R}^{3\times 32\times 32}\to\mathbb{R}$，其定义从源代码反推可知

$$\begin{aligned}
\mathrm{NonlocalNet}(\mathbf{x})&\equiv B_4(B_3(B_2(B_1(\mathbf{x}))))\\
B_1(\mathbf{x})&\equiv\mathrm{MaxPool}(\mathrm{ReLU}(\mathrm{Conv}_{3,32}(\mathbf{x})))\\
B_2(\mathbf{x})&\equiv\mathrm{MaxPool}(\mathrm{ReLU}(\mathrm{Conv}_{32,64}(\mathrm{NonLocalBlock}_{32}(\mathbf{x})))\\
B_3(\mathbf{x})&\equiv\mathrm{MaxPool}(\mathrm{ReLU}(\mathrm{Conv}_{64,128}(\mathrm{NonLocalBlock}_{64}(\mathbf{x})))\\
B_4(\mathbf{x})&\equiv\mathrm{FC}_{256,1}(\mathrm{ReLU}(\mathrm{FC}_{2048,256}(\mathbf{x})))
\end{aligned}$$

其中 NonLocalBlock 是两年前 [Wang et al., 2018] 提出的，其想法来源于十五年前的 [Buades et al., 2005]；[Buades et al., 2005] 当时发现只用局部平滑滤波器做图像降噪的效果不如 NL-means 好；NL-means 的方法是根据图像全局的自相似性加权的图像中所有像素的平均；[Wang et al., 2018] 则把 NL-means 包装成一个可插拔的神经网络层

8. [Nijkamp et al., 2019] 的实验 §4.2 用的能量函数叫做 VanillaNet，论文里没写它的构造，这里记录一下我通过源代码反推的构造：

$$C_{256,1} \circ R \circ C_{128,256} \circ R \circ C_{64,128} \circ R \circ C_{32,64} \circ R \circ C_{3,32}$$

其中 R 是一些负斜率恒为 0.05 的 Leaky ReLU 函数，其他 C 的定义如下：

(a) $C_{3,32} : \mathbb{R}^{3\times32\times32} \xrightarrow{3\times3\text{ 卷积核，步伐 }1\text{，填充 }1} \mathbb{R}^{32\times32\times32}$

(b) $C_{32,64} : \mathbb{R}^{32\times32\times32} \xrightarrow{4\times4\text{ 卷积核，步伐 }2\text{，填充 }1} \mathbb{R}^{64\times16\times16}$

(c) $C_{64,128} : \mathbb{R}^{64\times16\times16} \xrightarrow{4\times4\text{ 卷积核，步伐 }2\text{，填充 }1} \mathbb{R}^{128\times8\times8}$

(d) $C_{128,256} : \mathbb{R}^{128\times8\times8} \xrightarrow{4\times4\text{ 卷积核，步伐 }2\text{，填充 }1} \mathbb{R}^{256\times4\times4}$

(e) $C_{256,1} : \mathbb{R}^{256\times4\times4} \xrightarrow{4\times4\text{ 卷积核，步伐 }1\text{，填充 }0} \mathbb{R}^{1\times1\times1}$

9. 复现了 [Nijkamp et al., 2019] 的实验 §4.1，用 Langevin 噪音 $\varepsilon = 0.125$、MCMC 步数 $L = 500$ 训练 ToyNet，让它学习分布 (1)，每批训练 100 个样本，我的实验结果如下：

   (a) 早在 31000 批训练之后，持久初始化五百步 MCMC 短跑取负样本在核密度估计下的图案就可以显现出四环的形状，而此时 CNN 的输出在 $\mathbb{R}^2$ 上的图像还没有完全收敛到四环的形状，这大致符合作者的结论——短跑出样本容易，长跑收敛难

   (b) 约 93000 批之后 CNN 在 $\mathbb{R}^2$ 上的输出所作的图像才基本接近真实分布的四环的形状，说明 CNN 成功学习到了真实分布；最终跑到二十万批，中间学习到的分布略有扰动，但整体没有太多的偏离，基本可以认为实现了收敛的最大似然学习

10. 复现了 [Nijkamp et al., 2019] 的实验 §4.2，用 Langevin 噪音 $\varepsilon = 0.01$、MCMC 步数 $L = 150$ 训练 VanillaNet，让它学习分布 (3)，每批训练 100 张图，跑完了十万批，我的实验结果如下：

    (a) 用数据初始化 MCMC 进行十万步长跑，在训练完一万批、两万批……九万批、十万批之后进行 MCMC 长跑，均得到非常糟糕的结果，基本都只有一整块、一整块的颜色块，什么都看不出来，说明学习到的分布没有收敛到真实分布

(b) 用随机噪声初始化 MCMC 一百五十步短跑，100 批之后短跑就能得到隐约的花的形状，8800 批之后短跑就能得到非常好看的图了，之后的九万多批训练完了之后基本没有改善；短跑出来的图始终带有一些椒盐颗粒感，这是可以理解的，因为毕竟是从噪音初始化来的而且只是短跑，跑到山腰上就停下来了；而且用的能量函数还是这么简单的 VanillaNet，说明只出好图不收敛的 EBM 最大似然学习真的非常容易

(c) 进一步观察 d_{s_t} 和 r_t，d_{s_t} 确实在 0 附近震荡；d_{s_t}, r_t 都表现出短时滞上的强烈的负自相关性，d_{s_t}, r_t 也确实呈现出扩张、伸缩相关性，说明该实验在文中第一个轴上的表现是良好的，而在第二个轴上的表现不佳

11. 把 [Nijkamp et al., 2019] 的实验 §4.3 （Langevin 噪音 $\varepsilon = 0.0075$、MCMC 步数 $L = 500$ 训练 NonlocalNet，让它学习分布 (3)，每批训练 100 张图）训练到了二十万批，我的实验结果如下：

(a) 若用五百步 MCMC （用持久初始化）在学习到的分布上取样，那么在仅仅训练了 100 批之后，五百步 MCMC 跑出来的负样本就已经有模糊的花的样子出现了，且带有一些雪花；经过 3300 批之后基本上五百步 MCMC 跑出来的负样本都有视觉上可识别的花出现；一万批之后不再有雪花；五百步这么短的 MCMC 跑出来的负样本都是视觉效果良好的花，于是复现了作者的结论——MCMC 短跑很容易跑出好看的图

(b) 若用十万步 MCMC （用数据初始化而不是持久初始化）在学习到的分布上取样，那么一万批至四万批训练之后长跑得到的都是过饱和的图，直到五万批训练之后 MCMC 长跑才得到一个视觉上略佳的结果，于是复现了作者的结论——MCMC 长跑很难跑出好看的图，最大似然学习要收敛是很难的

(c) 持久初始化后的 MCMC 会随着批次的增加渐渐向真实分布 q 的山峰方向行走（视觉上体现为雪花越来越少），且在三千批之后就基本稳定（视觉上体现为只有微弱的噪音级别的变化）；长期跑的持久初始化的 MCMC 初值出现了跨越山峰的行为，例如十几万批之后一些 MCMC 初值的花的样子变了，而且目测似乎变到了离之前的山峰不太远的另外一个山峰

(d) 十万批、十一万批……二十万批后用数据初始化的十万步 MCMC 跑出来的效果都很好，没有出现任何退化，说明学习到的分布应该是真的收敛到真实分布了

(e) 若用五百步 MCMC （用持久初始化）在学习到的分布上取样还是一如既往的好，说明收敛的 §4.3 至少达到了不收敛的 §4.2 同样好的效果，而且对比实验 §4.2 用噪声初始化 §4.3 在视觉上的椒盐颗粒感稍微弱一些

(f) d_{s_t} 确实在 0 附近震荡；r_t 最终收敛到约 0.06，五万批至二十万批之间基本没有偏离这个均值，看上去 Langevin 确实梯度成功平衡住了 Langevin 噪音

参考文献

[Buades et al., 2005] Buades, A., Coll, B., and Morel, J.-M. (2005). A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 60–65. IEEE.

[Nijkamp et al., 2019] Nijkamp, E., Hill, M., Han, T., Zhu, S.-C., and Wu, Y. N. (2019). On the anatomy of mcmc-based maximum likelihood learning of energy-based models. *arXiv preprint arXiv:1903.12370*.

[Wang et al., 2018] Wang, X., Girshick, R., Gupta, A., and He, K. (2018). Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803.

[Zhu et al., 1998] Zhu, S. C., Wu, Y., and Mumford, D. (1998). Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27(2):107–126.