深度生成式模型与遥感影像压缩

刘志
www.linkedin.com/in/zhiliuln

[1] 智能感知理解与图像理解教育部重点实验室
西安电子科技大学

[2] School of Electric Engineering
Xidian University

初稿：2016 年 05 月 17 日, 修改：2017 年 04 月 15 日

追根溯源-神经网络中的能量函数

磁矩与自旋

Definitions

磁矩 (*magnetic moment*) 是磁铁物质的物理性质, 决定了其处于外磁场时的转矩. 载流回路、电子、分子或行星等都有磁矩.

自旋 (*spin*) 是粒子所具有的内禀性质 (如质量、电量), 为粒子与生俱来的一种角动量, 为量子化的且大小不可变, **自旋可以产生磁矩**. 自旋为 0 的粒子, 从各方向看都一样; 自旋为 1、2 的粒子分别旋转 360 度 (如手)、180 度后看起来一样; 自旋为 $\frac{1}{2}$ 的粒子, 旋转 720 度后看起来一样.

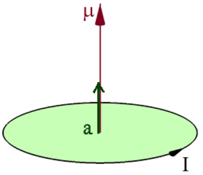


图: 平面载流环产生磁偶极矩

$\mu = Ia)$

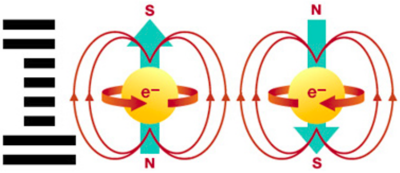
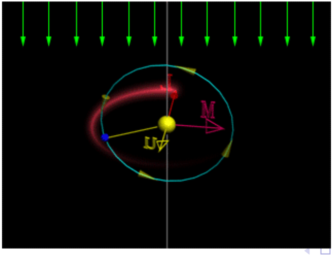


图: 电子自旋 (自旋向上和自旋向下)

追根溯源-神经网络中的能量函数

磁矩、磁力矩与能量

- **磁矩** (μ) 描述载流线圈或微观粒子磁性的物理量，与外磁场无关;
- **磁力矩** ($M = \mu \times B$) 是载流线圈或微观粒子在外磁场中受到的力矩，与外磁场有关.
- 力矩做功: $W = \int_{\theta_0}^{\theta} M \mathrm{d}\theta = -\mu B \cos\theta + \mu B \cos\theta_0 = -\mu \cdot B + C$
- **磁矩在外磁场中具有的能量**: $E = -\mu \cdot B + C$, 如果取磁矩 μ 与外磁场 B 垂直时具有的能量为 0, 则 $E = -\mu \cdot B$. 磁矩在与磁矩方向相反的外加磁场中有了附加能量 $E = \mu \cdot B$, 磁矩在与磁矩方向相同的外加磁场中有了附加能量 $E = -\mu \cdot B$.



追根溯源-神经网络中的能量函数

Ising model

Definitions

伊辛模型 (*Ising model*) 是一个以物理学家恩斯特·易辛为名的数学模型, 用于描述物质的铁磁性. 记 Λ 为所有晶格点 (磁矩通常会按照某种规则排列, 形成晶格) 的集合, 每个晶格点的邻接晶格点 (数学上的图) 形成一个 d 维晶格. 对于每个晶格点 $k \in \Lambda$ 都有一个离散变数 (描述单个原子磁矩的参数) $\sigma_k \in \{+1, -1\}$, 代表一个晶格点的自旋状态 ($+1$ 自旋向上, -1 自旋向下), 所有变数的集合 $\sigma = (\sigma_k)_{k\in\Lambda}$ 则称作*自旋组态*.

对于两个相邻的晶格点 $i, j \in \Lambda$, 引入**交互作用参数** J_{ij}, 并假设每个晶格点 $j \in \Lambda$ 的自旋受外加磁场 h_j 的作用, 则整个系统的**哈密顿量** (即能量) 为:

$$H(\sigma) = -\sum_{\langle i\ j\rangle} J_{ij}\sigma_i\sigma_j - \mu\sum_j h_j\sigma_j \tag{1}$$

其中, $\langle i\ j\rangle$ 代表晶格点 i 和晶格点 j 相邻, 因而哈密顿量的第一项代表所有自旋之间的交互作用能量, 而第二项是外界磁场与自旋交互作用的能量, μ 为晶格点的磁矩值.

追根溯源-神经网络中的能量函数

Ising model

Note

$J_{ij} > 0$ ，系统为铁磁性 $h_j > 0$ ，晶格点 j 倾向于正向

$J_{ij} < 0$ ，系统为反铁磁性 $h_j < 0$ ，晶格点 j 倾向于负向

$J_{ij} = 0$ ，自旋间无交互作用 $h_j = 0$ ，没有外加磁场作用于自旋

当热力学温度 (即绝对温度, 一般工程中指正的绝对温度, 但负的绝对温度是存在的) 的倒数 $\beta \geq 0$ 时, 该系统的**组态概率** (*configuration probability*)$P(\sigma)$ 服从玻尔兹曼分布 (*boltzmann distribution*):

$$P_\beta(\sigma) = \frac{e^{-\beta H(\sigma)}}{Z_\beta} \tag{2}$$

其中, $\beta = \frac{1}{k_B}\left(\frac{\partial S}{\partial E}\right)_{V,N} = \frac{1}{k_B T}$ (k_B 为玻尔兹曼常数, S 为熵, E 为能量, V 为体积, N 粒子数, T 绝对温度), 且 $Z_\beta = \sum_\sigma e^{-\beta H(\sigma)}$ 为归一化常数, 在统计力学中又称配分函数 (*partition function*).

追根溯源-神经网络中的能量函数

Ising model

Definition

在统计力学和数学中, 玻尔兹曼分布 (*boltzmann distribution*) 是一个系统粒子状态 (如位置) 上的概率分布, 或者频率分布 (能量越低概率越大), 系统处于状态 i 时的概率分布表示成: $p_i \propto e^{-\frac{E_i}{kT}}$

因为系统的自旋组态非常多, 伊辛模型一般很难直接进行数值计算, 如对于一个拥有 L 个晶格点的模型, 每个晶格点 σ_j 有两种自旋状态, 因而有 2^L 种的自旋组态, 常采用蒙特卡罗方法.

Note

蒙特卡罗方法 (*Monte Carlo method*) 也称**统计模拟方法**, 于二十世纪四十年代, 由冯·诺依曼和斯塔尼斯拉夫·乌拉姆提出, 并借驰名世界的赌城-摩纳哥的 Monte Carlo 来命名, 为它蒙上一种神秘色彩. **旨在通过随机化方法计算积分**. 比如计算积分 $\int_a^b h(x)\mathrm{d}x$, 若无解析解, 为避免枚举, 将 $h(x)$ 分解为某个函数与一个定义在 (a,b) 上的 PDF 的乘积, 这样积分变为: $\int_a^b f(x)p(x)\mathrm{d}x = \mathbb{E}_{p(x)}[f(x)] \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i)$. 问题就转换为**如何采集服从分布** $p(x)$ **的样本**.

追根溯源-神经网络中的能量函数

Hopfield Net

Definition

霍普菲尔德神经网络 (*Hopfield Neural Network*) 是一种递归神经网络, 由约翰·霍普菲尔德 (John Hopfield) 于 1982 年发明. 是一种结合**存储**[a]系统和二元系统的神经网络.

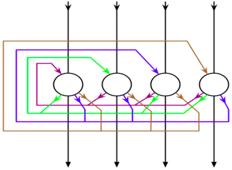
[a]联想存储器 (Content-addressable memory, CAM), 是一种特殊类型的计算机存储.

离散 Hopfield 神经网络特点:

- 是一个单层的二值神经网络;
- 权重对称性: 每个神经元都有到其它神经元的反馈 ($w_{ij} = w_{ji}$), 各神经元节点没有自反馈 ($w_{ij} = 0$);
- 神经单元是二进制阈值单元, 即只有两个状态 ($\{-1, 1\}$ 或 $\{0, 1\}$);
- 两种工作方式: 异步方式和同步方式.

神经元状态更新规则:

$$s_i \leftarrow \begin{cases} +1 & \text{if } \sum_j w_{ij} s_j \geq \theta_i \\ -1 & \text{otherwise.} \end{cases}$$



追根溯源-神经网络中的能量函数

Energy in Hopfield Net

Hopfield 神经网络的标量能量函数 (属于 Ising models 模型 1) 为:

$$E = -\frac{1}{2}\sum_{i,j} w_{i,j}s_i s_j + \sum_i \theta_i s_i \tag{3}$$

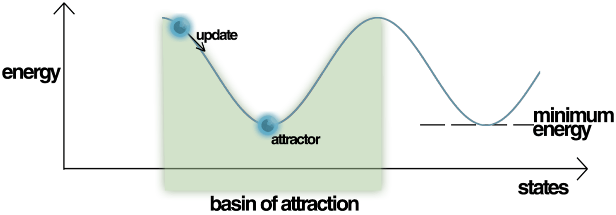


图: Hopfield 网络的能量图 (from WiKi).

Note:

权重对称性的要求是必须的, 它保证了能量方程在满足神经元激活规则时单调递减, 而不对称的权重可能导致周期性的递增或噪声.

基于能量的模型（Energy-based Models）

定义

模型的定义

基于能量的模型为每一个感兴趣的变量分配一个能量函数 E, 模型的学习相当于修改能量函数, 使其具有理想性质, 定义基于能量的概率分布为:

$$p(\boldsymbol{x}) = \frac{e^{-E(\boldsymbol{x})}}{Z} \tag{4}$$

其中, $Z = \sum_{\boldsymbol{x}} e^{-E(\boldsymbol{x})}$ 为归一化常量, 类似于物理系统中的配分函数.

模型学习

我们的目的是**最大化**模型的概率, 可通过执行训练数据 $\mathcal{D}$ 上的经验负对数似然上的梯度下降学习:

$$\mathcal{L}(\boldsymbol{\theta}|\mathcal{D})^{a} = -\frac{1}{N}\sum_{\boldsymbol{x}^{(i)}\in\mathcal{D}} \ln p(\boldsymbol{x}^{(i)}) \tag{5}$$

其中, $\boldsymbol{\theta}$ 为模型的参数, N 为样本数目.

[a]对于模型的参数 $\boldsymbol{\theta}$, 频率统计 (*frequentist statistics*): fixed but unknown, 看作参数, 点估计 $\hat{\boldsymbol{\theta}}$ 是一个随机变量; 贝叶斯估计 (*bayes statistics*): uncertain or unknown, 看作随机变量.

基于能量的模型（Energy-based Models）

参数的更新

参数的更新规则

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} + \underbrace{\eta \frac{\partial}{\partial \boldsymbol{\theta}^{(t)}} \left(\sum_{i=1}^{N} \ln \mathcal{L}(\boldsymbol{\theta}^{(t)} | \boldsymbol{x}^{(i)}) \right) - \lambda \boldsymbol{\theta}^{(t)} + \nu \Delta \boldsymbol{\theta}^{t-1}}_{\Delta \boldsymbol{\theta}^{(t)}} \tag{6}$$

其中, $\eta \in \mathbb{R}^{+}$ 为学习率, $\lambda \in \mathbb{R}^{+}, \nu \in \mathbb{R}^{+}$ 分别为权重衰减惩罚 (*weight decay penalizer*) 项 $-\boldsymbol{\theta}^{(t)}$ 和动量 (*momentum*) 项 $\Delta \boldsymbol{\theta}^{(t-1)}$ 的平衡因子, 权重衰减惩罚是为了防止参数值过大, 通常在目标函数中加入衰减项 $||\boldsymbol{\theta}||^{2}/2$; 动量项的加入, 有助于防止迭代过程中的震荡并能加速前馈神经网络的学习过程.

基于能量的模型 (Energy-based Models)

引入隐变量

许多时候, 样例 $\boldsymbol{x}$ 不可完全观测, 或者我们想引入一些不可观测变量以增强模型的表达能力. 因而需考虑可观测部分 (这里仍记为 $\boldsymbol{x}$) 和一个隐藏部分 $\boldsymbol{h}$. 模型的联合概率分布从而表示为:

$$p(\mathbf{x}=\boldsymbol{x}, \mathbf{h}=\boldsymbol{h}) = \frac{1}{Z} e^{-E(\boldsymbol{x},\boldsymbol{h})} \tag{7}$$

由于只有 $\mathbf{x}$ 是可观测的, 感兴趣的是模型在 $\mathbf{x}$ 上边缘分布:

$$p(\boldsymbol{x}) = \sum_{\boldsymbol{h}} p(\boldsymbol{x}, \boldsymbol{h}) = \frac{1}{Z} \sum_{\boldsymbol{h}} e^{-E(\boldsymbol{x},\boldsymbol{h})} \tag{8}$$

其中 $Z = \sum_{\boldsymbol{x}} \sum_{\boldsymbol{h}} e^{-E(\boldsymbol{x},\boldsymbol{h})}$, 为得到与4式相同形式, 引入自由能 (*free energy*, 受物理启发):

$$\mathcal{F}(\boldsymbol{x}) = -\ln \sum_{\boldsymbol{h}} e^{-E(\boldsymbol{x},\boldsymbol{h})} \tag{9}$$

则模型的概率分布变为:

$$p(\boldsymbol{x}) = \frac{e^{-\mathcal{F}(\boldsymbol{x})}}{Z} \tag{10}$$

其中, $Z = \sum_{\boldsymbol{x}} e^{-\mathcal{F}(\boldsymbol{x})}$.

基于能量的模型 (Energy-based Models)

正相与负相阶段

目的是最大化边缘概率 p(x), 取其负对数似然函数为:

$$-\ln p(\boldsymbol{x}) = \mathcal{F}(\boldsymbol{x}) + \ln Z \tag{11}$$

最大化边缘概率 $p(\boldsymbol{x})$, 即最小化负对数似然, 负对数似然上的梯度为:

$$\begin{aligned}
-\frac{\partial \ln p(\boldsymbol{x})}{\partial \boldsymbol{\theta}} &= \frac{\partial \mathcal{F}(\boldsymbol{x})}{\partial \boldsymbol{\theta}} + \frac{1}{Z}\frac{\partial Z}{\partial \boldsymbol{\theta}} \\
&= \frac{\partial \mathcal{F}(\boldsymbol{x})}{\partial \boldsymbol{\theta}} + \frac{1}{Z}\frac{\partial \sum_{\boldsymbol{x}} e^{-\mathcal{F}(\boldsymbol{x})}}{\partial \boldsymbol{\theta}} \\
&= \frac{\partial \mathcal{F}(\boldsymbol{x})}{\partial \boldsymbol{\theta}} + \frac{1}{Z}\sum_{\tilde{\boldsymbol{x}}} \frac{-\partial \mathcal{F}(\tilde{\boldsymbol{x}})}{\partial \boldsymbol{\theta}} e^{-\mathcal{F}(\tilde{\boldsymbol{x}})} \\
&= \frac{\partial \mathcal{F}(\boldsymbol{x})}{\partial \boldsymbol{\theta}} - \sum_{\tilde{\boldsymbol{x}}} p(\tilde{\boldsymbol{x}}) \frac{\partial \mathcal{F}(\tilde{\boldsymbol{x}})}{\partial \boldsymbol{\theta}}.
\end{aligned} \tag{12}$$

上述梯度包含两项, 即正相 (*positive phase*) 和负相 (*negative phase*) 阶段. 术语中的正和负不是指等式中每项前面的符号, 而是指它们**对模型概率密度的影响**. 第一项增大训练数据的概率 (通过减少相应的自由能), 而第二项减小模型生成样本的概率.

基于能量的模型 (Energy-based Models)

梯度的计算

上述梯度涉及分布 p 上的期望计算 $\mathbb{E}_p\left[\frac{\partial \mathcal{F}(\boldsymbol{x})}{\partial \boldsymbol{\theta}}\right]$, 通常很难直接计算, 这与计算所有 x 的期望的代价是相当的. 为使计算可行, 第一步使用固定数量的样本来估计期望 (即负相梯度), 这些样本称为负粒子 (*negative particles*), 记为 $\mathcal{N}$, 这样:

$$-\frac{\partial \ln p(\boldsymbol{x})}{\partial \boldsymbol{\theta}} \approx \frac{\partial \mathcal{F}(\boldsymbol{x})}{\partial \boldsymbol{\theta}} - \frac{1}{|\mathcal{N}|}\sum_{\tilde{\boldsymbol{x}} \in \mathcal{N}} \frac{\partial \mathcal{F}(\tilde{\boldsymbol{x}})}{\partial \boldsymbol{\theta}}. \tag{13}$$

其中, $\mathcal{N}$ 中的元素 $\tilde{\boldsymbol{x}}$ 是按照分布 p 采样的样本, 即蒙特卡罗采样. 提取这些负粒子最为有效的方法是马尔可夫链蒙特卡罗方法 (*Markov Chain Monte Carlo*, MCMC), 简单来讲 **MCMC 的基本思想**就是利用马尔可夫链来产生指定分布下的样本.

基于能量的模型（Energy-based Models）

马尔可夫链蒙特卡罗方法——随机过程及马尔可夫性质

什么是随机过程

安德雷·马尔可夫 (Andrey Markov, 也有人译作马尔科夫) 是俄国数学家, 开创了随机过程这个新领域. 随机过程 (*stochastic process*, or *random process*) 产生于 20 世纪初期, 研究随“时间”变化的“动态”的随机现象, 随机过程与概率论的关系就像动力学与静力学的关系.

Definition

马尔可夫性质 (*markov property*) 因俄国数学家安德雷·马尔可夫得名. 当一个随机过程在给定现在状态及所有过去状态情况下, **其未来状态的条件概率分布仅依赖于当前状态**; 换句话说, 在给定现在状态时, 它与过去状态是条件独立的, 那么此随机过程即具有马尔可夫性质.

Definition

马尔可夫过程 (*markov process*) 是指一个具备了马尔可夫性质的随机过程. 通常马尔可夫链是指具备离散状态的马尔可夫过程, 又称离散时间马尔可夫链, 但允许时间取连续的值.

基于能量的模型（Energy-based Models）

马尔可夫链蒙特卡罗方法——马尔可夫链

Theorem

马尔可夫链 *(Markov chain)* 为状态空间中从一个状态转换到另一个状态的无记忆性的随机过程，即**下一状态的概率分布只由当前状态决定** (网络爬虫原理)，在时间序列中它前面的事件均与之无关.

数学上，设 $\{X_t, t=0,1,\cdots\}$(也可以表示成 $\{X(t), t=0,1,\cdots\}$) 为一随机序列 (即随机变量 X 在离散时间 t 时刻的取值)，若满足:

$$P\{X_{t+1}=s|X_0=s_0,X_1=s_1,\cdots,X_t=s_t\}=P\{X_{t+1}=s|X_t=s_t\} \tag{14}$$

则称该序列为马尔可夫链，称 X_t 的可能取值集合 $\mathbb{S}=\{s_i\}_{i=0}^{n}$ 为其**状态空间**.

Definition

m 阶马尔可夫链是指具有**未来状态仅取决于前 m 个状态**性质的随机序列，即:

$$\begin{aligned} &P\{X_t=s|X_{t-1}=s_{t-1},X_{t-2}=s_{t-2},\cdots,X_0=s_0\} \\ &=P\{X_t=s|X_{t-1}=s_{t-1},X_{t-2}=s_{t-2},\cdots,X_{t-m}=s_{t-m}\} \end{aligned} \tag{15}$$

基于能量的模型（Energy-based Models）

马尔可夫链蒙特卡罗方法——转移概率矩阵

Definition

系统状态 (*states*) 的改变称为转移 (*transition*), 转变的概率称为转移概率 (*transition probabilities*), 如系统从状态 s_i 转移到 s_j 的转移概率为:

$$P_{ij} = P(X_{t+1} = s_j | X_t = s_i)$$

转移矩阵 (*transition matrix*), 也称转移概率矩阵, 表示状态空间 $\mathbb{S}$ 中任意状态间转换的概率, 其第 i 行第 j 列的元素表示随机变量从状态 s_i 转移到状态 s_j 的概率, 设状态数为 $n+1$, 则转移矩阵可表示成:

$$\boldsymbol{P} = \begin{pmatrix} P_{0,0} & P_{0,1} & \cdots & P_{0,n} \\ P_{1,0} & P_{1,1} & \cdots & P_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ P_{n,0} & P_{n,1} & \cdots & P_{n,n} \end{pmatrix} \tag{16}$$

基于能量的模型 (Energy-based Models)

马尔可夫链蒙特卡罗方法——转移概率矩阵

记随机变量 X 在时刻 t 取状态 s_k 的概率为 $\mu_k^{(t)} = P(X_t = s_k)$, 则

$$
\begin{aligned}
\mu_i^{(t+1)} &= P(X_{t+1} = s_i) \\
&= \sum_k P(X_{t+1} = s_i | X_t = s_k) \cdot P(X_t = s_k) \\
&= \sum_k P_{ki} \cdot \mu_k^{(t)}
\end{aligned}
\tag{17}
$$

若记概率矢量 (随机变量 X 在 t 时刻的取值概率) 为:$\boldsymbol{\mu}^{(t)} = (\mu_0^{(t)}, \cdots, \mu_n^{(t)})$, 则有:

$$
\boldsymbol{\mu}^{(t+1)} = \boldsymbol{\mu}^{(t)} \cdot \boldsymbol{P} \quad \text{or} \quad \boldsymbol{\mu}^{(t)} = \boldsymbol{\mu}^{(0)} \cdot \boldsymbol{P}^t \tag{18}
$$

Definition

- 周期性 (*periodic*): 存在某一状态, 从该状态出发, 经过固定次数的转移总能回到自身, 即遍历图有可能陷入死循环.
- 不可约性 (*irreducible*): 任一状态都可来自任意其它状态, 即图是联通的.
- 各态遍历的 (*ergodic*): 即非周期且不可约.

基于能量的模型（Energy-based Models）

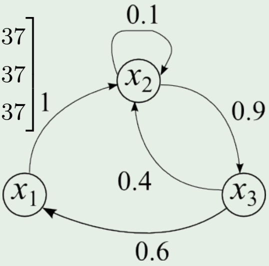
马尔可夫链蒙特卡罗方法——转移概率矩阵

Example

如图，为一马尔可夫链的图表示，可见该马尔可夫链具有各态遍历的性质，状态空间为：$\mathbb{S} = \{x_1, x_2, x_3\}$，设初始概率矢量 $\boldsymbol{\mu}^{(0)} = (0.3, 0.5, 0.2)$，容易求得转移概率矩阵以及进行 k 步转移后的概率矢量 $\boldsymbol{\mu}^{(k)}$：

转移概率矩阵：

$$\boldsymbol{P} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0.1 & 0.9 \\ 0.6 & 0.4 & 0 \end{bmatrix}, \boldsymbol{P}^{(100)} = \begin{bmatrix} 0.22 & 0.41 & 0.37 \\ 0.22 & 0.41 & 0.37 \\ 0.22 & 0.41 & 0.37 \end{bmatrix}$$



$k = 10, 30$ 时分别有概率矢量：

$$\boldsymbol{\mu}^{(10)} = (0.22567119, 0.41033498, 0.36399383)$$

$$\boldsymbol{\mu}^{(30)} = (0.22131563, 0.40982273, 0.36886165)$$

具有各态遍历性的马尔可夫链，会使概率矢量的分布收敛？

基于能量的模型（Energy-based Models）

马尔可夫链蒙特卡罗方法——各态遍历与平稳分布

Theorem

记 $\boldsymbol{P}$ 为一概率转移矩阵，**平稳分布** *(stationary distribution)* 是指满足如下条件的分布 π:

$$\boldsymbol{\pi} = \boldsymbol{\pi} \cdot \boldsymbol{P} \tag{19}$$

可反转马尔可夫链类似于应用贝叶斯定理反转一个条件概率，即存在一个分布 π，满足:

$$\pi_i P_{ij} = \pi_j P_{ji} \tag{20}$$

这个条件被称为**细致平衡** *(detailed balance*, 物理上：从状态 i 转移到状态 j 的概率质量，恰好可以被从状态 j 到 i 转回，即状态 i 上的概率质量是稳定的) 条件，即分布 π 为平稳分布的充分条件[a].

对于具有**各态遍历性**的马尔可夫链或转移矩阵 $\boldsymbol{P}$，无论初始概率矢量 $\boldsymbol{\mu}^{(0)}$ 服从何种分布，随着转移次数的增加，最终都会**收敛到一平稳分布**π:

$$\lim_{t\to+\infty} \boldsymbol{\mu}^{(0)}\boldsymbol{P}^t = \boldsymbol{\pi} \quad \text{or} \quad \lim_{t\to+\infty} \boldsymbol{P}^t = [\boldsymbol{\pi}^\top, \cdots, \boldsymbol{\pi}^\top]^\top \tag{21}$$

基于能量的模型（Energy-based Models）

马尔可夫链蒙特卡罗方法——Metropolis-Hastings 采样

重要结论

可见 MCMC 方法的关键就是设计合理的状态转移过程，即**转移矩阵的设计**，使得最终收敛的平稳分布正是我们想要的分布!

算法起源

1953 年，Metropolis 在研究粒子系统的平稳性时，首次提出了基于 MCMC 方法的玻尔兹曼分布近似算法 *Metropolis*，并启发了一系列 MCMC 方法，*Metropolis-Hastings* 算法就是 Metropolis 算法的一个变种.

算法的关键是转移矩阵的设计，假设有一个马尔可夫链的转移矩阵为 $\boldsymbol{Q}$，要逼近的分布是 π，一般来讲细致平衡条件并不满足:

$$\pi_i Q_{ij} \neq \pi_j Q_{ji} \tag{22}$$

按照对称性引入矩阵 $\boldsymbol{A}$，使得 $A_{ij} = \pi_j Q_{ji}, A_{ji} = \pi_i Q_{ij}$，则有细致平衡条件成立

$$\pi_i Q_{ij} A_{ij} = \pi_j Q_{ji} A_{ji} \tag{23}$$

基于能量的模型 (Energy-based Models)

马尔可夫链蒙特卡罗方法——Metropolis-Hastings 采样

其中, Q 为提议分布 (*proposal distribution*), Q_{ij}(或 $Q(s_j|s_i)$) 表示在状态 s_i 时提议状态 s_j 的条件概率, A 为接受分布 (*acceptance distribution*), A_{ij}(或 $A(s_i, s_j)$) 表示从状态 s_i 转换到状态 s_j 的提议的接受概率. 为避免出现接受率过小从而收敛太慢的情况, 可以同比例放大 A_{ij} 和 A_{ji}, 最大至 1 即取接受率:

$$A_{ij} = \min\left\{1, \frac{\pi_j Q_{ji}}{\pi_i Q_{ij}}\right\}, \quad \text{or} \quad A(s_i, s_j) = \min\left\{1, \frac{\pi(s_j)Q(s_i|s_j)}{\pi(s_i)Q(s_j|s_i)}\right\} \tag{24}$$

即接受接受率大于 1 的提议, 拒绝接受率小于 1 的提议.

记 $\boldsymbol{T} = \boldsymbol{Q} \odot \boldsymbol{A}$, 其中 $\odot$ 表示操作矩阵对应元素相乘, 则有细致平衡条件成立:

$$\pi_i T_{ij} = \pi_j T_{ji} \quad \text{or} \quad \pi(s_i)T(s_j|s_i) = \pi(s_j)T(s_i|s_j), \tag{25}$$

矩阵 $\boldsymbol{T}$ 就是所设计的转移矩阵.

一般来讲, 提议分布 Q 选择比较简单的概率分布, 如高斯分布等等.

基于能量的模型（Energy-based Models）

马尔可夫链蒙特卡罗方法——Metropolis-Hastings 采样（Algorithm）

Algorithm 1 Metropolis-Hastings 采样算法

Input: 初始化马尔可夫链（Q）初始状态 $X_0 = s_0$.
Output: 感兴趣分布 π 的采样

for $t = 0$ to $N-1$ **do**
  根据马尔可夫链 $Q(s_y|s_t)$，生成提议 s_y
  从均匀分布采样 $u \sim \mathcal{U}(0,1)$
  if $A(s_t, s_y) = \min\left\{\frac{Q(s_t|s_y)\pi(s_y)}{Q(s_y|s_t)\pi(s_t)}, 1\right\} \geq u$ **then**
    接受提议，即 $X_{t+1} = s_y$
  else
    拒绝提议，即 $X_{t+1} = s_t$
  end if
end for

Example

1. 假设**目标分布**为高斯分布：$x \sim \mathcal{N}(\mu, \sigma^2)$，即概率密度函数为 $p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$，其中 $\mu = -2, \sigma = 1$,
2. 取**提议分布** $x \sim \mathcal{N}(0.5, 0.4^2)$.

基于能量的模型（Energy-based Models）

马尔可夫链蒙特卡罗方法——Metropolis-Hastings 采样（例子）

Example

MATLAB 代码如下:

```
n = 250000;
x = zeros(n, 1);
x(1) = 0.5;
mu = -2; sigma = 1;
for i = 1: n
    % proposal from normal distribution
    x_c = normrnd(x(i), 0.4);
    a = min(1, ...
        normpdf(x_c, mu, sigma)/normpdf(x(i), mu, sigma))
    if rand < a
        x(i+1) = x_c;
    else
        x(i+1) = x(i);
    end
end
```

基于能量的模型（Energy-based Models）

马尔可夫链蒙特卡罗方法——Metropolis-Hastings 采样（例子）

Example

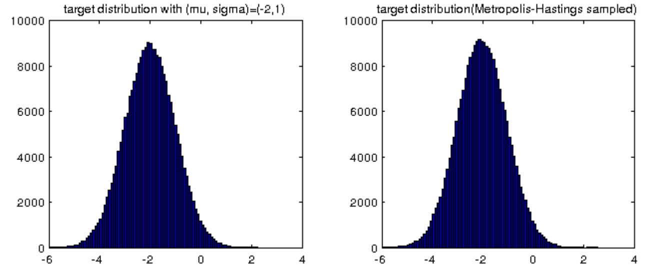


图: (左) 感兴趣目标分布, (右)Metropolis-Hastings 采样逼近的分布

Metropolis Hastings 方法的优劣

- **优点:** 统一的通用框架, 可发展出一系列的 MCMC 方法
- **缺点:** (1) 过于灵活, 状态转移提议分布选择的不好, 容易造成收敛速度过慢; (2) 存在接受率

基于能量的模型（Energy-based Models）
Gibbs 采样

简介

吉布斯采样 (*Gibbs Sampling*) 是一种 MCMC 采样方法, 因物理学家 Josiah Willard Gibbs 而得名, 于 1984 年由 Stuart Geman 和 Donald Geman 两兄弟提出. 其初级版本可以看作是 Metropolis-Hastings 方法的一个特例, 扩展版本可看作对高维总体的抽样框架.

Gibbs 采样适用于联合分布未知或难以直接采样, 而每个变量的条件分布已知并容易抽样的情况. 采样序列构成一个马尔可夫链, 该链的平稳分布就是感兴趣的平稳分布.

对于多元分布, 在条件分布上采样要比通过对联合分布积分边缘化概率容易, 假如我们想获得联合分布 $\pi(x_1, \cdots, x_d)$ 的 N 次采样 $\boldsymbol{X}$, 第 t 个采样记为 $\boldsymbol{x}^{(t)} = [x_1^{(t)}, \cdots, x_d^{(t)}]$, 则 Gibbs 采样方法如下:

基于能量的模型（Energy-based Models）

Gibbs 采样

Algorithm 2 吉布斯采样算法

1: 随机初始化系统状态为 $\boldsymbol{x}^{(0)}$
2: 初始化时刻 $t=0$
3: 对下一个采样向量 $\boldsymbol{x}^{(t+1)}$ 的每个成分 $x_i^{(t+1)}, i=1,\cdots,d$, 依次按如下条件概率采样

$$\pi(x_i^{(t+1)}|x_1^{(t+1)},\cdots,x_{i-1}^{(t+1)},x_{i+1}^{(t)},\cdots,x_d^{(t)})$$

4: $t=t+1$
5: 循环 $3-4$ 过程 N 次，即得联合分布 $\pi(\boldsymbol{x})$ 的 N 次采样.

Fact

执行如上采样，有:

- 由于不存在接受率，即接受率为 1，所以收敛速度快;
- 采样近似服从所有变量的联合分布;
- 任意变量子集的边缘分布，可通过仅考虑该变量子集的 *Gibbs* 采样获得;
- 任意变量的期望值可由所有样本的平均值近似.

基于能量的模型 (Energy-based Models)

Gibbs 采样

Proof.

记 d 维随机矢量 $\mathbf{x} = (\mathrm{x}_1, \cdots, \mathrm{x}_d)^\top$ 在 t 时刻的状态为 $\boldsymbol{x}^{(t)} = (x_1^{(t)}, \cdots, x_d^{(t)})^\top$, 并定义 $\mathbf{x}_{-i} = (\mathrm{x}_1, \cdots, \mathrm{x}_{i-1}, \mathrm{x}_{i+1}, \cdots, \mathrm{x}_d)^\top$, $i = 1, \cdots, d$, 为除第 i 个随机变量外的随机矢量, 在 t 时刻的状态 $\boldsymbol{x}_{-i}^{(t)} = (x_1^{(t)}, \cdots, x_{i-1}^{(t)}, x_{i+1}^{(t)}, \cdots, x_d^{(t)})^\top$.

设某一 $t+1$ 时刻 $\boldsymbol{x} = (x_1^{(t+1)}, \cdots, x_{i-1}^{(t+1)}, x_i^{(t)}, \cdots, x_d^{(t)})^\top$, 我们按次序更新随机变量 x_i, 更新后的状态记作 $\boldsymbol{y} = (x_1^{(t+1)}, \cdots, x_{i-1}^{(t+1)}, x_i^{(t+1)}, x_{i+1}^{(t)}, \cdots, x_d^{(t)})^\top$, 取产生从 $\boldsymbol{x}$ 到 $\boldsymbol{y}$ 提议的提议分布为 $q(\boldsymbol{y}|\boldsymbol{x}) = \pi(y_i|\boldsymbol{x}_{-i})$, 则接受率为:

$$A(\boldsymbol{x}, \boldsymbol{y}) = \min\left\{1, \frac{\pi(\boldsymbol{y})q(\boldsymbol{x}|\boldsymbol{y})}{\pi(\boldsymbol{x})q(\boldsymbol{y}|\boldsymbol{x})}\right\} = \min\left\{1, \frac{\pi(\boldsymbol{y})\pi(x_i|\boldsymbol{y}_{-i})}{\pi(\boldsymbol{x})\pi(y_i|\boldsymbol{x}_{-i})}\right\} = 1$$

且有细致平衡条件成立:

$$\pi(\boldsymbol{x})\boldsymbol{T}_{\boldsymbol{xy}} = \pi(\boldsymbol{y})\boldsymbol{T}_{\boldsymbol{yx}}$$

其中, $\boldsymbol{T}_{\boldsymbol{xy}} = q(\boldsymbol{y}|\boldsymbol{x}) = \pi(y_i|\boldsymbol{x}_{-i})$. □

基于能量的模型 (Energy-based Models)

Gibbs 采样

Note

- 系统变量初始值可随机产生或由诸如期望最大化的算法产生;
- 通常忽略起始阶段 (也称 *burn-in period*) 的一些采样, 这是因为: (1) 成功的采样间互相是不独立的; (2)Gibbs 采样过程需要一段时间才能达到平稳分布.
- 在采样初期, 通常采用模拟退火 (*simulated annealing*) 过程来减少" 随机游走"(*random walk*) 的行为.

二值玻尔兹曼机和受限玻尔兹曼机

二分图

Definition

二分图 (*bipartite graph*): 称二部图. 设 $G = (V, E)$ 是一个**无向图**, 如果其顶点 V 可以划分成两个互不相交的子集, 且任一子集内无边连接.

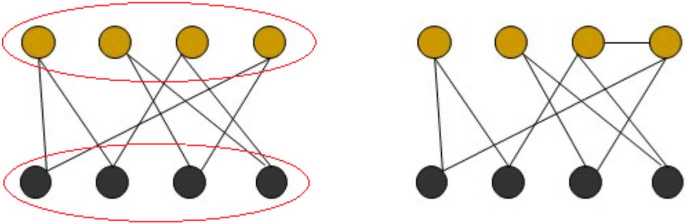


图: 二分图（左），非二分图（右）

二值玻尔兹曼机和受限玻尔兹曼机

引言

人物缩影

路德维希·玻尔兹曼 (德语:*Ludwig Eduard Boltzmann*,1844-1906) 是奥地利的物理学家和哲学家. 1906 年 9 月 5 日，在杜伊诺度假村，路德维希·爱德华·玻尔兹曼再一次情绪失控, 并试图自杀，希望以此结束自己在动理方程和 H 定理上所遭遇的激烈诘难. 1908 年，实验结果最终判定了奥斯特瓦尔德“唯能论”的失败，然而，他的对手玻尔兹曼已经无法见证自己的胜利.

玻尔兹曼机 (*Boltzmann machine, BM*) 是**随机神经网络**和**递归神经网络**的一种，由杰弗里·辛顿 (Geoffrey Hinton) 和特里·谢泽诺斯基 (Terry Sejnowski) 在 1985 年发明, 因**玻尔兹曼分布**而得名。

受限玻尔兹曼机 (*Restricted Boltzmann machine, RBM*) 最初由保罗·斯模棱斯基 (paul Smolensky) 于 1986 年提出, 并命名为簧风琴 (Harmonium). 直到 2000 年代中叶 Hinton 等人发明快速学习算法后才变得知名, 在降维, 分类, 协同过滤, 特征学习, 主题建模等领域得到了应用.

图: 路德维希·玻尔兹曼

二值玻尔兹曼机和受限玻尔兹曼机

玻尔兹曼机

玻尔兹曼机 (*boltzmann machine*) 与 Hopfield 网络一样, 是一个能量模型. BM 的神经元为二值和随机的, 可被视作随机过程的, 其能量函数与 Hopfield 网络具有相同的形式:

$$E(\boldsymbol{x}) = -\boldsymbol{x}\boldsymbol{U}\boldsymbol{x} - \boldsymbol{b}^\top \boldsymbol{x} \tag{26}$$

二进制随机变量 $\boldsymbol{x}$ 上的分布服从玻尔兹曼分布:

$$p(\boldsymbol{x}) = \frac{e^{-E(\boldsymbol{x})}}{Z} \tag{27}$$

其中 $Z = \sum_{\boldsymbol{x}} e^{-E(\boldsymbol{x})}$ 为配分函数, 以保证 $\sum_{\boldsymbol{x}} p(\boldsymbol{x}) = 1$.

- 网络节点为二值的 ($\boldsymbol{x} \in \{0,1\}^d$), 在 Ising 模型中表示晶格点;
- $\boldsymbol{U}$ 是玻尔兹曼机的权重矩阵参数, 在 Ising 模型中表示交互作用参数;
- $\boldsymbol{b}$ 是玻尔兹曼机的偏置向量参数, 在 Ising 模型中表示外加磁场.

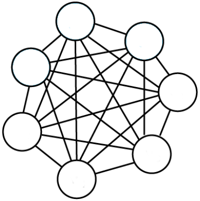


图: 玻尔兹曼机的图表示 (无向图)

二值玻尔兹曼机和受限玻尔兹曼机

引入隐变量

许多时候，样例 $\boldsymbol{x}$ 不可完全观测，或者我们想引入一些不可观测变量以增强模型的表达能力．因而需考虑可观测部分 (这里记为 $\boldsymbol{v}$) 和一个隐藏部分 $\boldsymbol{h}$.

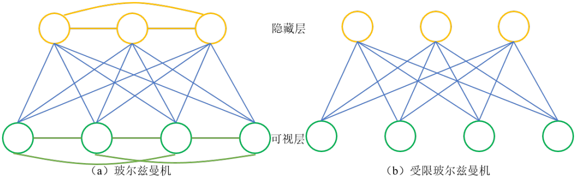


图: 玻尔兹曼机与受限玻尔兹曼机．图中顶层表示隐藏层 (hidden-layer)，底层表示可视层 (visible-layer)，所有的节点都是随机二值变量节点 (即只能取 0 或 1)．假设上述无向图的全概率分布满足 Boltzmann 分布，我们称图 (a) 所示的全连接图为玻尔兹曼机 (BMs)；图 (b) 所示层内节点无连接的模型为受限玻尔兹曼机 (RBMs).

玻尔兹曼机与受限玻尔兹曼机的能量函数分别为 (注意偏置与外加磁场关系1):

$$E(\boldsymbol{v},\boldsymbol{h}) = -\boldsymbol{v}^\top \boldsymbol{R}\boldsymbol{v} - \boldsymbol{h}^\top \boldsymbol{S}\boldsymbol{h} - \boldsymbol{h}^\top \boldsymbol{W}\boldsymbol{v} - \boldsymbol{b}^\top \boldsymbol{v} - \boldsymbol{c}^\top \boldsymbol{h} \tag{28}$$

$$E(\boldsymbol{v},\boldsymbol{h}) = -\boldsymbol{h}^\top \boldsymbol{W}\boldsymbol{v} - \boldsymbol{b}^\top \boldsymbol{v} - \boldsymbol{c}^\top \boldsymbol{h} \tag{29}$$

二值玻尔兹曼机和受限玻尔兹曼机

引入隐变量

模型的概率分布从而表示为:

$$p(\mathbf{v}=\boldsymbol{v},\mathbf{h}=\boldsymbol{h})=\frac{1}{Z}e^{-E(\boldsymbol{v},\boldsymbol{h})} \tag{30}$$

其中 $Z=\sum_{\boldsymbol{v}}\sum_{\boldsymbol{h}}e^{-E(\boldsymbol{v},\boldsymbol{h})}$, 为得到与27式相同形式, 引入自由能 (*free energy*, 受物理启发):

$$\mathcal{F}(\boldsymbol{v})=-\ln\sum_{\boldsymbol{h}}e^{-E(\boldsymbol{v},\boldsymbol{h})} \tag{31}$$

则模型的概率分布变为:

$$p(\boldsymbol{v})=\frac{e^{-\mathcal{F}(\boldsymbol{v})}}{Z}\ \text{with Z}=\sum_{\boldsymbol{v}}\mathrm{e}^{-\mathcal{F}(\boldsymbol{v})}. \tag{32}$$

受限玻尔兹曼机

对于受限玻尔兹曼机有能量函数：

$$E(\boldsymbol{v},\boldsymbol{h}) = -\boldsymbol{h}^\top \boldsymbol{W}\boldsymbol{v} - \boldsymbol{b}^\top \boldsymbol{v} - \boldsymbol{c}^\top \boldsymbol{h} \tag{33}$$

从而自由能函数为:

$$\mathcal{F}(\boldsymbol{v}) = -\ln \sum_{\boldsymbol{h}} e^{-E(\boldsymbol{v},\boldsymbol{h})} = -\boldsymbol{b}^\top \boldsymbol{v} - \sum_i \ln \sum_{h_i} e^{h_i(c_i + W_i \boldsymbol{v})} \tag{34}$$

这里， W_i, W_j 分别表示矩阵 $\boldsymbol{W}$ 的第 i 行和第 j 列.

由于 RBMs 的特殊结构, 可视单元和隐藏单元互相条件独立, 利用这个特性, 有:

$$p(\boldsymbol{h}|\boldsymbol{v}) = \prod_i (h_i|\boldsymbol{v}) \tag{35}$$

$$p(\boldsymbol{v}|\boldsymbol{h}) = \prod_j (v_j|\boldsymbol{h}). \tag{36}$$

受限玻尔兹曼机

二值单元

通常研究的二值单元 (即 $v_i, h_i \in \{0, 1\}$) 情况，获得概率版本的常用神经元激活函数:

$$P(h_i = 1|\boldsymbol{v}) = \operatorname{sigm}(c_i + W_i\boldsymbol{v}) \tag{37}$$

$$P(v_j = 1|\boldsymbol{h}) = \operatorname{sigm}(b_j + W_j^\top \boldsymbol{h}) \tag{38}$$

带有二值单元的 RBM 的自由能可以简化为:

$$\mathcal{F}(\boldsymbol{v}) = -\boldsymbol{b}^\top \boldsymbol{v} - \sum_i \log\left(1 + e^{(c_i + W_i\boldsymbol{v})}\right) \tag{39}$$

结合能量模型梯度计算表达式13，和二值 RBM 自由能计算表达式39，得到二值单元 RBM 的对数似然梯度:

$$-\frac{\partial \log p(\boldsymbol{v})}{\partial W_{ij}} = E_{\boldsymbol{v}}[p(h_i|\boldsymbol{v}) \cdot v_j] - v_j^{(i)} \cdot \operatorname{sigm}(W_i \cdot \boldsymbol{v}^{(i)} + c_i) \tag{40}$$

$$-\frac{\partial \log p(\boldsymbol{v})}{\partial c_i} = E_{\boldsymbol{v}}[p(h_i|\boldsymbol{v})] - \operatorname{sigm}(W_i \cdot \boldsymbol{v}^{(i)}) \tag{41}$$

$$-\frac{\partial \log p(\boldsymbol{v})}{\partial b_j} = E_{\boldsymbol{v}}[p(v_j|\boldsymbol{h})] - v_j^{(i)} \tag{42}$$

受限玻尔兹曼机

RBM 中的采样

待续 !!!!!!!!!!!!!!!

受限玻尔兹曼机

对比散度

待续 !!!!!!!!!!!!!!!

待续！！！！！！！！！！！！

基于二值深度玻尔兹曼机的遥感影像压缩

原理

将实数值数据进行二值编码，将二值编码送入深度玻尔兹曼机网络，实现压缩.

1. 对原始图像数据分块（如 8×8）并拉成列向量，将数据进行二值编码（如直接取其二进制形式）从而转换成二值矢量;
2.

基于二值深度玻尔兹曼机的遥感影像压缩

实验结果

512-1000-500-250-16 , PSNR: 17.76 dB, MSE: 1090

基于二值深度玻尔兹曼机的遥感影像压缩

实验结果

网络结构：64-512-1000-500-250-256，压缩比：2

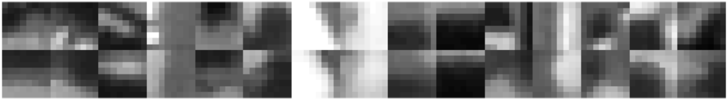


图: 压缩比为 2

网络结构：64-512-1000-500-250-32，压缩比：16

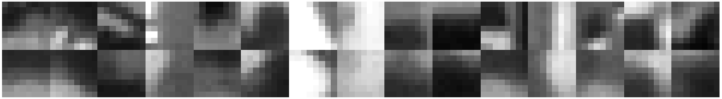


图: 压缩比为 16

基于实数值深度玻尔兹曼机的遥感影像压缩

原理

基于实数值深度玻尔兹曼机的遥感影像压缩

实验结果

基于深层网络的遥感影像压缩技术

方法概览

1. 基于深度自编码网络的遥感影像压缩
2. 基于深度限态自编码网络的遥感影像压缩
3. 基于深度二值深度玻尔兹曼机的遥感影像压缩
4. 基于深度实值深度玻尔兹曼机的遥感影像压缩
5. 基于张量扩展深度玻尔兹曼机的遥感影像压缩

Tip

- 把优化算法的设计看作学习问题，通过神经网络来解决[a]. 用自动学习的更新规则代替人工设计的更新规则，就像深度神经网络在特征提取中扮演的角色：用自动学习的特征替代设计人工的特征那样.
- 改进基于能量模型的学习算法，以加速网络的学习[b].
- 针对于图像压缩任务设计深度压缩解压缩网络.

[a]参考文献：Learning to learn by gradient descent by gradient descent, 2016

[b]参考文献：Deep Directed Generative Models with Energy-Based Probability Estimation, 2016

Thanks !