

机器读心术之神经网络与深度学习 第5周

DATAGURU专业数据分析社区

【声明】 本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

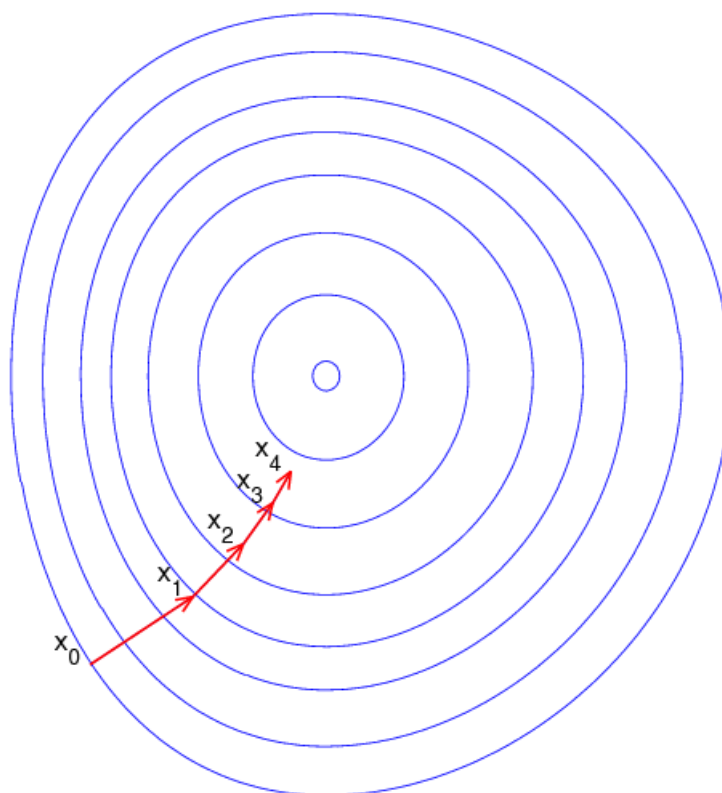
<http://edu.dataguru.cn>

关注炼数成金企业微信



- 提供全面的数据价值资讯，涵盖商业智能与数据分析、大数据、企业信息化、数字化技术等，各种高性价比课程信息，赶紧掏出您的手机关注吧！



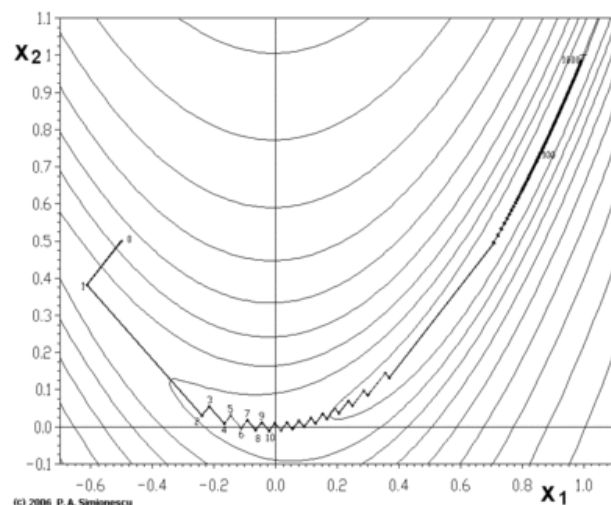


- 目标函数必须可微
- 学习率难以选取，太大会产生“之字形”震荡，太小迭代次数太多，前进很慢
- 容易陷入局部最优

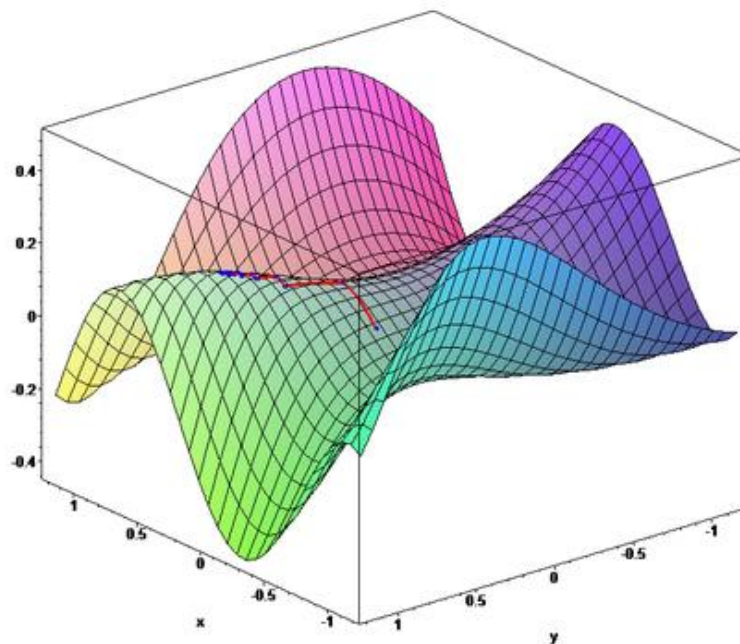
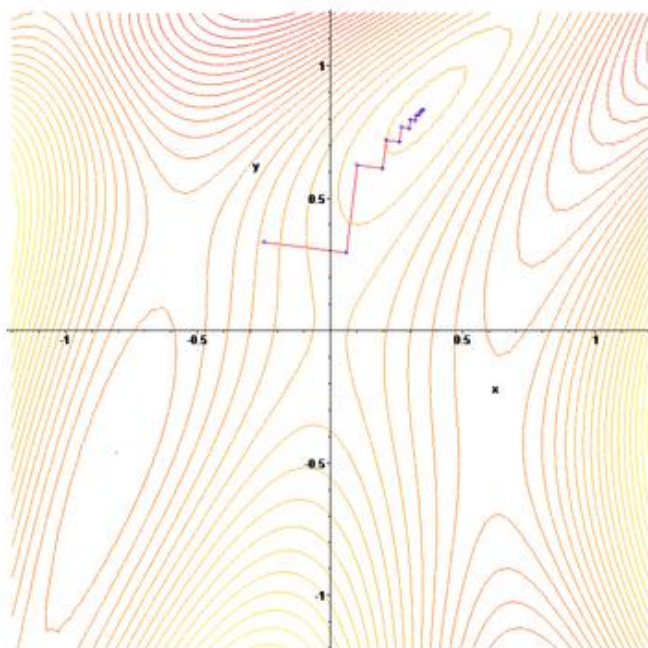
梯度下降法处理一些复杂的非线性函数会出现问题，例如Rosenbrock函数

$$f(x, y) = (1 - x)^2 + 100(y - x^2)^2.$$

其最小值在 $(x, y) = (1, 1)$ 处，数值为 $f(x, y) = 0$ 。但是此函数具有狭窄弯曲的山谷，最小值 $(x, y) = (1, 1)$ 就在这些山谷之中，并且谷底很平。优化过程是之字形的向极小值点靠近，速度非常缓慢。



下面这个例子也鲜明的示例了"之字"的下降，这个例子用梯度下降法求 $F(x, y) = \sin\left(\frac{1}{2}x^2 - \frac{1}{4}y^2 + 3\right) \cos(2x + 1 - e^y)$ 的极小值。







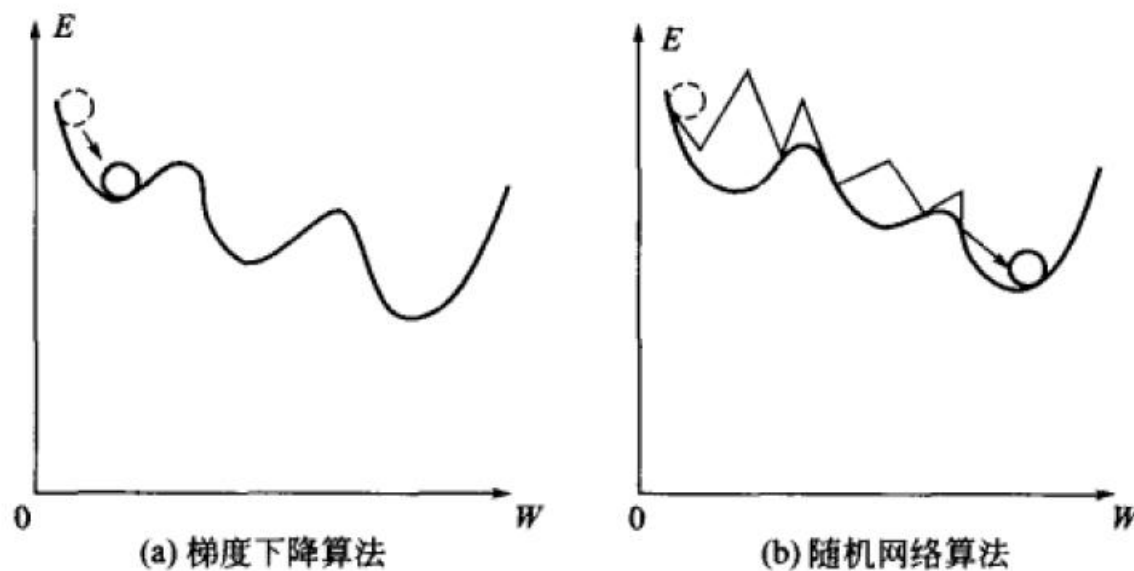
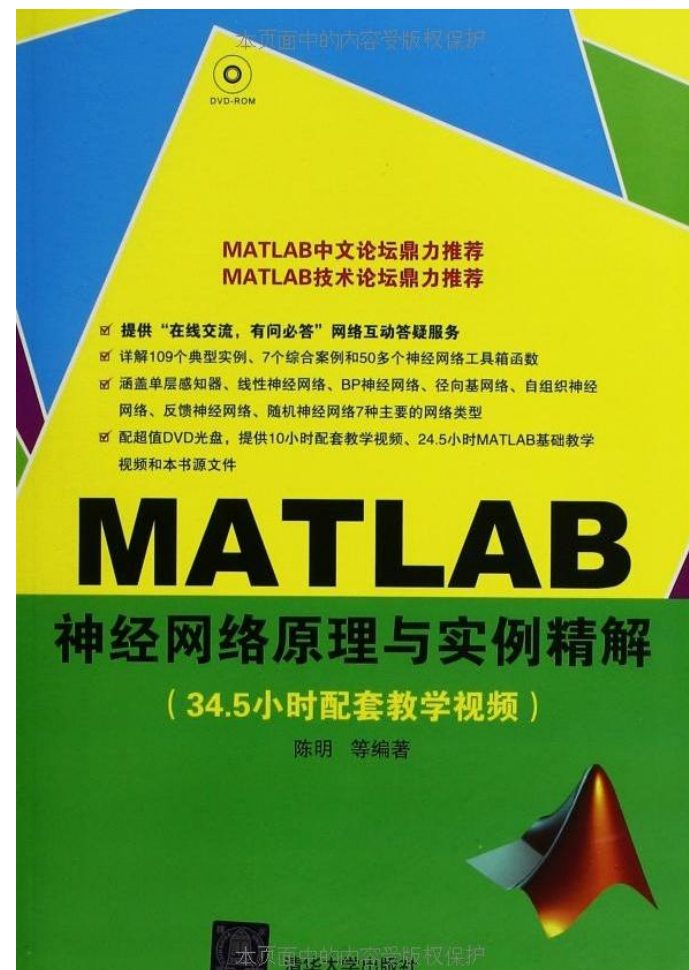
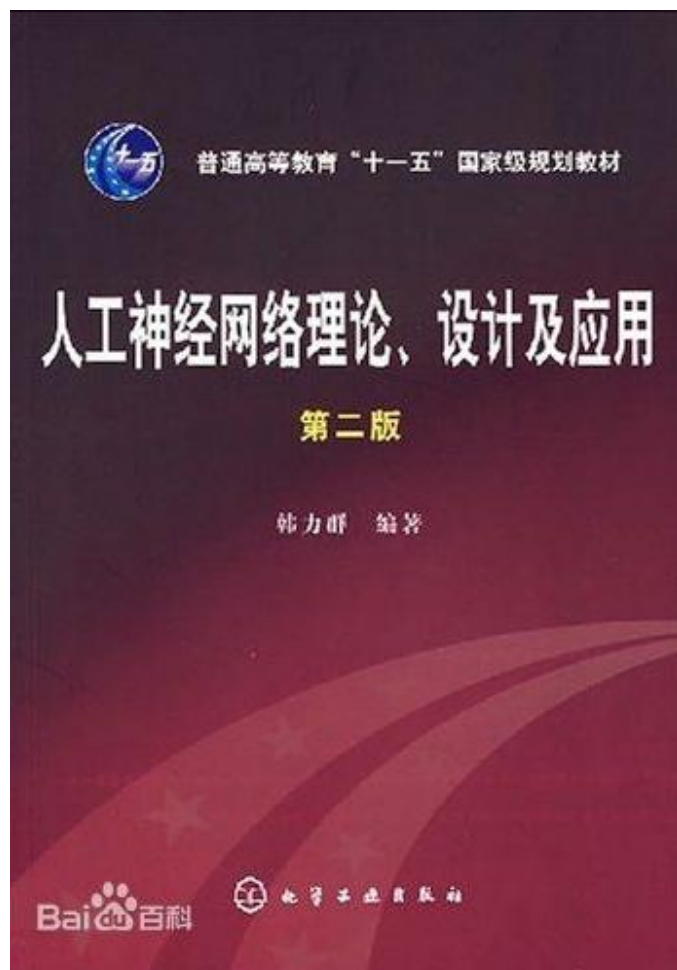


图 6.10 随机网络算法与梯度下降算法的区别



模拟退火算法包含 Metropolis 算法和退火过程两部分。Metropolis 算法又叫 Metropolis 抽样，是模拟退火算法的基础。在早期的科学计算中蒙特卡罗方法（Monte Carlo）是对大量原子在给定温度下的平衡态的随机模拟，但蒙特卡罗方法计算量偏大。

1953 年，Metropolis 提出重要性采样法，即以概率来接受新状态，而不是使用完全确定的规则，称为 Metropolis 准则，可以显著减小计算量。假设前一状态为 $x(n)$ ，系统受到一定扰动，状态变为 $x(n)$ ，相应地，系统能量由 $E(n)$ 变为 $E(n+1)$ 。定义系统由 $x(n)$ 变为 $x(n+1)$ 的接受概率为 P ：

$$p = \begin{cases} 1 & E(n+1) < E(n) \\ e^{\left(\frac{E(n+1)-E(n)}{T}\right)} & E(n+1) \geq E(n) \end{cases}$$

退火温度表包括以下内容：

(1) 温度的初始值 $T(0)$ ，初始温度应选得足够高，使得所有可能的状态转移都能被接受。初始温度越高，获得高质量的解的概率也越大，耗费的时间越长。初始温度可以根据抽样结果计算某些数值参数加以确定，也可以根据经验确定。

(2) 退火速率。最简单的速率下降方式是指数式下降：

$$T(n) = \lambda T(0) \quad n = 1, 2, \dots$$

λ 是一个小于 1 的正数，一般取值在 0.8 和 0.99 之间。使得对每一温度，有足够的转移尝试。指数式下降的收敛速度比较慢，其他下降方式如下所示：

$$T(n) = \frac{T(0)}{\log(1+t)}$$
$$T(n) = \frac{T(0)}{1+t}$$

模拟退火算法的运行步骤如下：

(1) 初始化。第一步是确定问题域，包括变量 x 的个数和维度，以及代价函数 $f(\cdot)$ 的计算方式，这里的代价函数相当于离散 Hopfield 网络中的能量函数，求解的目的是使代价函数最小。随机选择一定的值作为变量的初值 $x(0)$ ，并设置初始温度 $T(0)$ ，终止温度 T_{final} 和温度的下降公式及相应的参数。

(2) 运行 Metropolis 算法。以一定规则在当前状态 $x(n)$ 附近产生新的状态 $x'(n)$ ，计算 $f(x(n))$ 与 $f(x'(n))$ ，得到

$$\Delta f = f(x'(n)) - f(x(n))$$

如果 $\Delta f < 0$ ，说明 $x'(n)$ 优于 $x(n)$ ，就用 $x'(n)$ 作为下一状态的值： $x(n+1) = x'(n)$ 。如果 $\Delta f > 0$ ，说明能量变大，进行概率操作，计算

$$p = e^{-\frac{\Delta f}{T}}$$

再从 0~1 之间产生一个随机数 ξ ，如果 $\xi < p$ ，则接受 $x'(n)$ 为下一个状态值，否则拒绝 $x'(n)$ ，下一状态值保持不变： $x(n+1) = x(n)$ 。

(3) 根据内循环终止准则，检查是否达到热平衡。在内循环中，系统在一定的温度 T 下迭代，直到满足热平衡条件，此时应修改温度，再开始循环。内循环终止的条件有：代价函数 f 的值是否趋于稳定，按一定步数进行抽样等。如果判断当前已经达到热平衡，则转到第(4)步，否则转到第(2)步继续迭代。

(4) 按照公式调整 T ，根据外循环终止准则检查退火算法是否收敛。如果新的温度值小于给定的终止温度，则算法结束，此时的状态 $x(n)$ 即所求的最优点。否则转到第(2)步继续迭代。外循环终止的准则也可以设置为固定的迭代次数，达到该次数以后系统即停止计算。如果系统的熵值已经达到最小，此时可以认为已经达到了最低温度。或者连续降温若干次，代价函数都没有改善，也可以作为达到终止温度的判据。

Learning and Relearning in Boltzmann Machines

G. E. HINTON and T. J. SEJNOWSKI

Many of the chapters in this volume make use of the ability of a parallel network to perform cooperative searches for good solutions to problems. The basic idea is simple: The weights on the connections between processing units encode knowledge about how things normally fit together in some domain and the initial states or external inputs to a subset of the units encode some fragments of a structure within the domain. These fragments constitute a problem: What is the whole structure from which they probably came? The network computes a "good solution" to the problem by repeatedly updating the states of units that represent possible other parts of the structure until the network eventually settles into a stable state of activity that represents the solution.

Terry Sejnowski

- John Hopfield的博士生
- Hinton的长期合作者
- Terrence (Terry) Joseph Sejnowski (born 13 August 1947) is an Investigator at the Howard Hughes Medical Institute and is the Francis Crick Professor at The Salk Institute for Biological Studies where he directs the Computational Neurobiology Laboratory. In 2004 he was named the Francis Crick Professor and the Director of the Crick-Jacobs Center for Theoretical and Computational Biology at the Salk Institute. His research in neural networks and computational neuroscience has been pioneering.
- Sejnowski is also Professor of Biological Sciences and Adjunct Professor in the Departments of Neurosciences, Psychology, Cognitive Science, Computer Science and Engineering at the University of California, San Diego, where he is Co-Director of the Institute for Neural Computation.
- With Barbara Oakley, he co-created and taught Learning How To Learn: Powerful mental tools to help you master tough subjects, the world's most popular online course, It is available on Coursera.
- https://en.wikipedia.org/wiki/Terry_Sejnowski



■ 韩力群书第143页

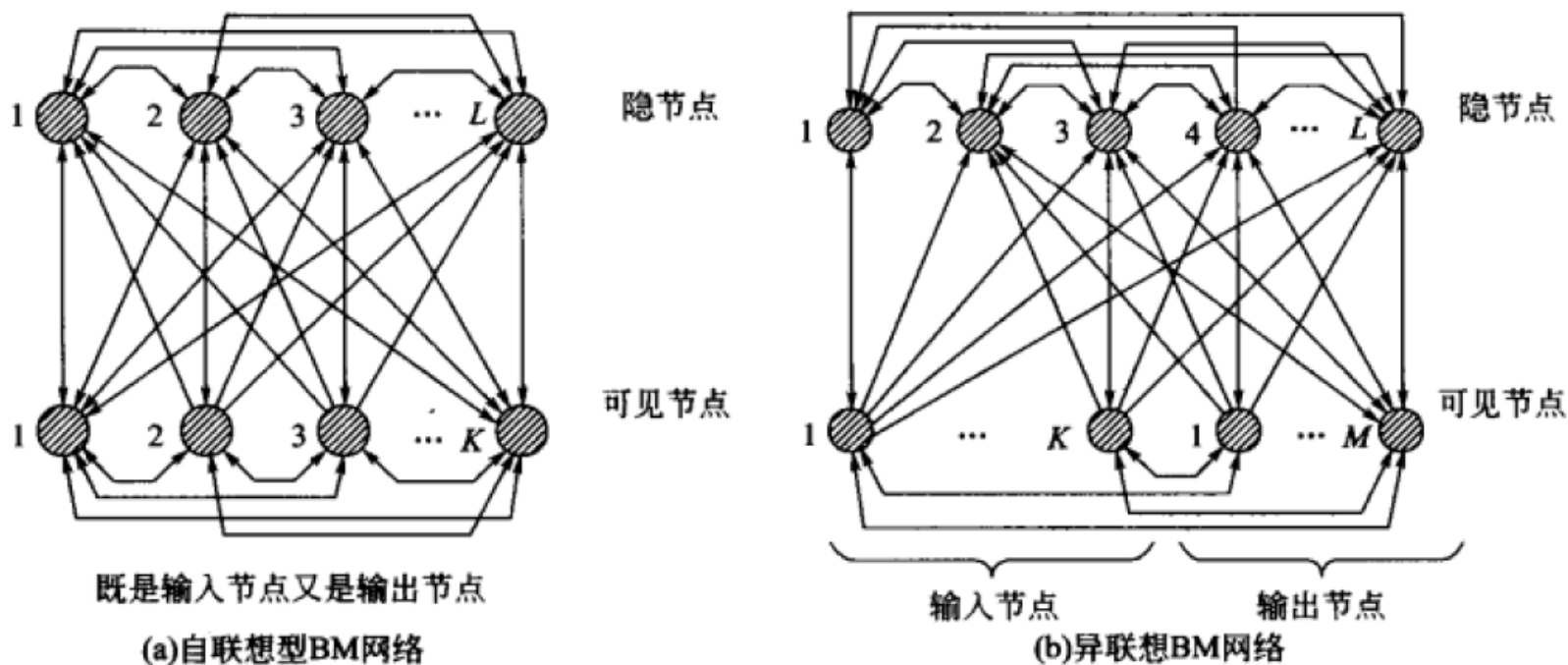


图 6.14 BM 网络学习网络结构

- 可以理解为随机版的Hopfield网络（也有别的理解方式，比如基于马尔科夫链等）
- 试图通过模拟退火算法克服Hopfield网络容易陷入局部极小的问题
- 模拟退火算法机制被糅合到训练和运行过程中

- 权值对称，无自反馈
- 输入节点和输出节点（可见层节点）可能被“钳制”，隐层节点通常自由变化
- 转移概率
- 能量函数，在概率意义下能量函数往减小的方向变化
- 异步方式工作

$$\frac{P_j(0)}{P_j(1)} = e^{\Delta E/T} = e^{(E_1 - E_0)/T} = \frac{e^{-E_0/T}}{e^{-E_1/T}}$$

$$\frac{P(\alpha)}{P(\beta)} = \frac{e^{-E_\alpha/T}}{e^{-E_\beta/T}}$$



■ 训练的目的，韩力群书第143页

提供了 P 对模式，一般有 $P < n$ 。训练集用一组概率分布表示各模式对出现的概率：

$$P(\mathbf{X}^1), P(\mathbf{X}^2), \dots, P(\mathbf{X}^P)$$

以上也是在正向学习阶段期望的网络状态概率分布。当网络自由运行时，相应模式出现的概率为：

$$P'(\mathbf{X}^1), P'(\mathbf{X}^2), \dots, P'(\mathbf{X}^P)$$

训练的目的是使以上两组概率分布相同。

① 在正向学习阶段，用一对训练模式 \mathbf{X}^p 钳住网络的可见节点；在反向学习阶段，用训练模式中的输入部分钳住可见节点中的输入节点。

② 随机选择自由活动节点 j ，使其更新状态：

$$s_j(t+1) = \begin{cases} 1, & s_j(t) = 0 \\ 0, & s_j(t) = 1 \end{cases}$$

③ 计算节点 j 状态更新而引起的网络能量变化 $\Delta E_j = -\Delta s_j(t) \text{net}_j(t)$ 。

④ 若 $\Delta E_j < 0$ ，则接受状态更新；若 $\Delta E_j > 0$ ，当 $P[s_j(t+1)] > \rho$ 时接受新状态，否则维持原状态。 $\rho \in (0, 1)$ 是预先设置的数值，在模拟退火过程中，温度 T 随时间逐渐降低，从式(6.5)可以看出，对于常数 ρ ，为使 $P[s_j(t+1)] > \rho$ ，必须使 ΔE_j 也在训练中不断减小，因此网络的爬山能力是不断减小的。

⑤ 返回步骤②~④直到自由节点被全部选择一遍。

⑥ 按事先选定的降温方程降温，退火算法的降温规律没有统一规定，一般要求初始温度 T_0 足够高，降温速度充分慢，以保证网络收敛到全局最小。下面给出两种降温方程：

$$T(t) = \frac{T_0}{1 + \ln t} \quad (6.46)$$

$$T(t) = \frac{T_0}{1 + t} \quad (6.47)$$

⑦ 返回步骤②～⑥直到对所有自由节点均有 $\Delta E_j = 0$ ，此时认为网络已达到热平衡状态。此状态可供学习算法中统计任意两个节点同时为 1 的概率时使用。

(3) 权值调整算法与步骤 BM 网络的学习算法步骤如下：

① 随机设定网络的初始权值 $w_{ij}(0)$ 。

② 正向学习阶段按已知概率 $P(\mathbf{X}^p)$ 向网络输入学习模式 \mathbf{X}^p , $p=1,2,\dots,P$ 。在 \mathbf{X}^p 的约束下按上述模拟退火算法运行网络到热平衡状态，统计该状态下网络中任意两节点 i 与 j 同时为 1 的概率 p_{ij} 。

③ 反向学习阶段在无约束条件下或在仅输入节点有约束条件下运行网络到热平衡状态，统计该状态下网络中任意两节点 i 与 j 同时为 1 的概率 p'_{ij} 。

④ 权值调整算法为：

$$\Delta w_{ij} = \eta(p_{ij} - p'_{ij}) \quad \eta > 0 \quad (6.48)$$

⑤ 重复以上步骤直到 p_{ij} 与 p'_{ij} 充分接近。

■ 《matlab神经网络原理与实例精解》第316页

(1) 初始化。Boltzmann 机神经元个数为 N ，第 i 个神经元与第 j 个神经元的连接权值为 ω_{ij} ，初始温度规定为 $T(0)$ ，终止温度为 T_{final} ，初始化神经元状态。

(2) 在温度 $T(n)$ 下，选取某个神经元 i ，根据下式计算其输入

$$x_i = \sum_{\substack{j=1 \\ j \neq i}}^N \omega_{ij} y_j + b_i$$

其中 y_j 为神经元状态。如果 $x_i > 0$ ，则能量有减小的趋势，取 1 为神经元 i 的下一状态值。如果 $x_i < 0$ ，则进行概率操作：计算 $p = \frac{1}{1 + e^{-\frac{x_i}{T(n)}}}$ ，然后在区间 $[0,1]$ 随机产生一个随机数 ξ ，若 $\xi < p$ ，则接受 1 为神经元的下一状态，否则状态保持不变。

(3) 检查小循环的终止条件。在小循环中，使用同一个温度值 $T(n)$ ，如果判断当前温度下系统已经达到热平衡，则转到第(4)步进行降温，否则转到第(2)步，继续随机选择一个神经元进行迭代。

(4) 按指定规律降温，并检查大循环的终止条件：判断温度是否到达终止温度，若达到终止温度，则算法结束，否则转到第二步继续计算。

- 计算时间漫长，特别是无约束自由迭代的负向阶段
- 对抽样噪音敏感
- 很难找到合适的应用案例
- 流行软件不支持

- Dataguru（炼数成金）是专业数据分析网站，提供教育，媒体，内容，社区，出版，数据分析业务等服务。我们的课程采用新兴的互联网教育形式，独创地发展了逆向收费式网络培训课程模式。既继承传统教育重学习氛围，重竞争压力的特点，同时又发挥互联网的威力打破时空限制，把天南地北志同道合的朋友组织在一起交流学习，使到原先孤立的学习个体组合成有组织的探索力量。并且把原先动辄成千上万的学习成本，直线下降至百元范围，造福大众。我们的目标是：低成本传播高价值知识，构架中国第一的网上知识流转阵地。
- 关于逆向收费式网络的详情，请看我们的培训网站 <http://edu.dataguru.cn>



Thanks

FAQ时间