现代物理与人工智能-II（第一部分）

韩子慕，王浩朴

1 序言

本复习资料针对《现代物理与人工智能》课程，主要内容忠实于授课 PPT，并对 PPT 上难以理解（或有省略）的部分进行了补充说明。能给出例题和示例的部分，我们准备了一些练习题供大家参考。2025 年考过的题目均以脚注的形式标明。

本复习资料由韩子慕和王浩朴编写。其中热力学与人工智能、统计物理与人工智能、物理仿真内容由韩子慕编写，量子力学与人工智能、混沌与分形部分由王浩朴编写。如有问题请联系对应编者。

2 从热力学熵到信息熵

2.1 热力学熵

熵（Entropy, S）是贯穿热力学与统计力学的核心概念，用于衡量系统的无序程度或信息丢失程度。熵的增大意味着系统从相对有序（低熵）的状态演变为更加混乱、更加无序（高熵）的状态。

在对卡诺定理及热力学第二定律的研究中，克劳修斯推出了克劳修斯不等式。对于可逆循环，总吸热 Q_1、总放热 Q_2 与热源温度 T_1、T_2（$T_1 > T_2$）满足

$$\frac{Q_1}{T_1} = \frac{Q_2}{T_2}.$$

由于 Q_1、T_1 以及 Q_2、T_2 分别对应循环的两个等温过程，而在绝热过程中 $Q = 0$，故对整个循环过程有

$$\oint \frac{\delta Q}{T} = 0.$$

对于不可逆循环，对循环过程进行微分并与相同热源下的可逆循环比较，利用卡诺定理可得

$$\oint \frac{\delta Q}{T} < 0.$$

综合上述结果，对于一切在给定热源下的工作循环，系统流入的热量与热源温度满足**克劳修斯不等式**

$$\oint \frac{\delta Q}{T} \leq 0,$$

其中等号当且仅当工作循环为可逆循环时成立。

对于可逆循环过程，由 $\oint \frac{\delta Q}{T} = 0$ 可知，循环中某一过程 L（始、末状态分别为 a、b）的积分

$$\int_L \frac{\delta Q}{T}$$

之值仅取决于始末状态 a 与 b，而与具体路径无关。这意味着存在一个**状态函数**，其微分等于 $\frac{\delta Q}{T}$。克劳修斯于 1865 年将这一状态函数正式命名为熵，记作 S，即

$$dS = \frac{\delta Q_{\text{rev}}}{T}.$$

此处下标 rev 强调该定义适用于可逆过程。

2.2 信息熵

香农将热力学中的熵概念引入信息论，提出了**信息熵**的概念，因而信息熵又被称为**香农熵**。

设离散随机变量 X 的可能取值为 $x_1, x_2, \ldots, x_n$，对应的概率分别为 $p_1, p_2, \ldots, p_n$，满足 $\sum_{i=1}^{n} p_i = 1$ 且 $p_i \leq 1$。信息熵 $H(X)$ 定义为

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_b p(x_i).$$

其中 b 为对数的底：当 $b = 2$ 时，熵的单位为比特（bit）；当 $b = e$ 时，单位为奈特（nat）；当 $b = 10$ 时，单位为哈特（Hart）。

信息熵用于衡量随机信源平均所包含的信息量，或等价地，衡量该信源的不确定性程度。

设有系统 S 内存在多个事件 $E = \{E_1, \ldots, E_n\}$，每个事件的概率分布为 $P = \{p_1, \ldots, p_n\}$，则每个事件本身的**自信息**为

$$I_i = -\log_2 p_i.$$

例如，英语有 26 个字母，若每个字母出现次数平均，则每个字母的自信息量为 $-\log_2 \frac{1}{26} \approx 4.7$；常用汉字有 2500 个，若均匀分布，则每个汉字的自信息量为 $-\log_2 \frac{1}{2500} \approx 11.3$。

3 相对熵——KL 散度

3.1 KL 散度的定义

相对熵又称 KL 散度。若对同一随机变量 x 存在两个概率分布 $P(x)$ 与 $Q(x)$，可使用 KL 散度衡量这两个分布的差异。通常情况下，P 表示数据的真实分布，Q 表示数据的理论分布或 P 的近似分布。

对于离散随机变量，概率分布 P 与 Q 的 KL 散度定义为[1]

$$D_{\mathrm{KL}}(P\|Q) = \sum_{i=1}^{n} P(i) \ln \frac{P(i)}{Q(i)}.$$

对于连续随机变量，KL 散度按积分方式定义为

$$D_{\mathrm{KL}}(P\|Q) = \int_{-\infty}^{\infty} p(x) \ln \frac{p(x)}{q(x)}\, dx,$$

其中 p 与 q 分别为分布 P 与 Q 的概率密度函数。

KL 散度的直观含义是：**当用分布 Q 来拟合真实分布 P 时，所需要的额外信息的平均量或编码长度**。

KL 散度具有以下两个基本性质：

非负性：

$$D_{\mathrm{KL}}(P\|Q) \geq 0,$$

当且仅当 P 与 Q 完全相同时（即对所有 i，$P(i) = Q(i)$），KL 散度为零。这表明用一个分布近似自身不会产生任何信息损失。

不对称性：

$$D_{\mathrm{KL}}(P\|Q) \neq D_{\mathrm{KL}}(Q\|P).$$

因此 KL 散度不是真正意义上的距离度量。

3.2 KL 散度与变分自编码器（VAE）

变分自编码器是生成模型的重要类型，KL 散度在其中发挥了关键作用。变分自编码器由编码器、隐空间与解码器三个核心部分组成。

[1] 一定要注意 KL 散度对于两个分布并不是对称的。

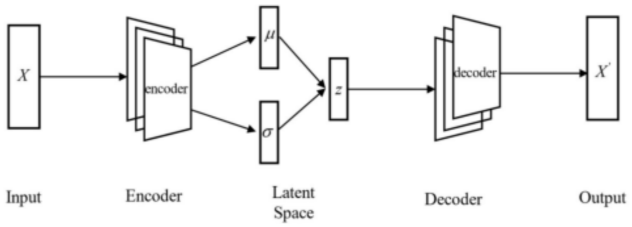

图 1: VAE 模型

编码器的输入为原始数据集合 $X = \{x_i\}_{i=1}^N$，其中每个数据点 x_i 相互独立。解码器的输出为生成的数据集合 $X' = \{x_i'\}_{i=1}^N$。通过编码器的学习，将高维输入数据映射到更为紧凑且具有潜在结构的低维空间，编码得到的隐变量 Z 在生成网络中被解码，从而能够生成具有相似特征的新数据。

变分自编码器假设隐变量 Z 所在空间具有简单的先验分布 $P(z)$（最常见的是标准正态分布 $\mathcal{N}(0, I)$）。编码器负责隐变量 Z 后验分布的近似推断，解码器负责生成变量。由于编码器编码得到的隐变量 Z 无法直接观测，从而无法直接进行变分推断生成新数据；因此变分自编码器引入已知的后验分布 $q(z|x)$，用以替代无法观测到的真实后验分布 $P(z|x)$。这一已知分布 $q(z|x)$ 充当编码器部分，而条件分布 $P(x'|z)$ 则充当解码器部分。

根据**贝叶斯公式**，$P(x|z)$ 与 $P(z|x)$ 之间存在如下约束关系：

$$P(z|x) = \frac{P(x|z)P(z)}{P(x)}.$$

其中各符号含义为：$P(x)$ 为事件 x 的先验概率；$P(z)$ 为事件 z 的先验概率；$P(x|z)$ 为 z 发生条件下 x 的概率；$P(z|x)$ 为 z 的后验概率。

设编码器输出的分布为 $Q(z|x)$，为使 $Q(z|x)$ 与真实后验 $P(z|x)$ 尽可能相似，可使用 KL 散度进行衡量。通过对参数的不断优化，最小化 KL 散度，使两个分布尽可能接近：

$$\arg\min_{\phi} D_{\mathrm{KL}}\big(Q(z|x)\|P(z|x)\big).$$

然而，在高维空间中，尤其是在 z 的维度较高时，数值积分的计算成本呈指数级增长，直接计算和采样 $P(z|x)$ 在实践中几乎不可行。

因此，通过数学变换，将问题转化为优化**证据下界**（ELBO）。从数据

对数似然 $\log P(x)$ 出发，推导证据下界：

$$\begin{aligned}
\log P(x) &= \log \int P(x,z)\,dz \\
&= \log \int Q(z|x)\frac{P(x,z)}{Q(z|x)}\,dz \\
&\geq \int Q(z|x)\log\frac{P(x,z)}{Q(z|x)}\,dz \quad \text{（由 Jensen 不等式）} \\
&= \int Q(z|x)\log P(x|z)\,dz - \int Q(z|x)\log\frac{Q(z|x)}{P(z)}\,dz \\
&= \mathbb{E}_{Q(z|x)}[\log P(x|z)] - D_{\mathrm{KL}}\big(Q(z|x)\|P(z)\big).
\end{aligned}$$

此下界被称为**证据下界**（Evidence Lower Bound, ELBO），记为

$$ELBO = \mathbb{E}_{z\sim Q(z|x)}[\log P(x|z)] - D_{\mathrm{KL}}\big(Q(z|x)\|P(z)\big).$$

一个好的生成模型应能准确捕捉数据的真实分布，给观测到的数据分配高概率（最大化 $\log P(x)$）。因此，变分自编码器的实际优化目标是最大化 ELBO，等价于最小化负 ELBO，最终损失函数为

$$L_{\mathrm{VAE}} = -\mathbb{E}_{z\sim Q(z|x)}[\log P(x|z)] + D_{\mathrm{KL}}\big(Q(z|x)\|P(z)\big).$$

4 交叉熵与交叉熵损失函数

4.1 交叉熵的定义

交叉熵直观地表示用分布 Q 描述真实分布 P 所需的信息量，其值越大表明两者差异越显著。

从 KL 散度的公式出发进行展开：

$$\begin{aligned}
D_{\mathrm{KL}}(P\|Q) &= \sum_i P(i)\log\frac{P(i)}{Q(i)} \\
&= \sum_i P(i)\big(\log P(i) - \log Q(i)\big) \\
&= \sum_i P(i)\log P(i) - \sum_i P(i)\log Q(i) \\
&= -\Big(-\sum_i P(i)\log P(i)\Big) - \sum_i P(i)\log Q(i).
\end{aligned}$$

根据信息熵的定义，$H(P) = -\sum_i P(i)\log P(i)$ 即为真实分布 P 的信息熵。而后面一项 $-\sum_i P(i)\log Q(i)$ 即为**交叉熵**。

对于离散概率分布 P（通常代表真实分布）与 Q（通常代表模型预测的分布），交叉熵 $H(P,Q)$ 定义为

$$H(P,Q) = -\sum_i P(i)\log Q(i).$$

对于连续概率分布，则为

$$H(P,Q) = -\int_{-\infty}^{\infty} p(x)\log q(x)\,dx.$$

交叉熵与 KL 散度、信息熵之间存在如下关系：

$$D_{\mathrm{KL}}(P\|Q) = H(P,Q) - H(P).$$

4.2 二元交叉熵

对于二分类问题（例如判断邮件是否为垃圾邮件、图片中是否有猫），模型的输出通常是一个介于 0 和 1 之间的概率值，表示属于正类的概率，而真实标签通常是 0 或 1。此时损失函数可写为如下的**二元交叉熵**

$$Loss = -\big[y\cdot\log(\hat{y}) + (1-y)\cdot\log(1-\hat{y})\big],$$

其中 y 为真实标签（0 或 1），$\hat{y}$ 为模型预测为正类的概率。

4.3 多类别交叉熵

对于多分类问题且类别之间互斥的情况（例如一张图片只能是猫、狗、鸟中的一种），模型输出为一组概率，所有类别概率之和为 1。此时真实标签通常为 One-Hot 编码向量（例如三个类别，猫 $=[1,0,0]$，狗 $=[0,1,0]$，鸟 $=[0,0,1]$）。损失函数可写为如下的**多类别交叉熵**

$$Loss = -\sum_i y_i\log(\hat{y}_i).$$

由于 y 为 One-Hot 向量，只有一个 y_i 为 1，其余均为 0，求和实际上只剩下对应真实类别 k 的一项：

$$Loss = -\log(\hat{y}_k),$$

其中 k 为真实类别的索引。这意味着损失只关心模型对唯一正确类别所预测的概率，预测概率越高，损失越小。

示例：考虑三分类问题（猫、狗、鸟），真实标签为狗，对应 One-Hot 表示为 $y = [0, 1, 0]$。设模型预测概率为 $\hat{y} = [0.2, 0.7, 0.1]$，则

$$Loss = -\big[0 \cdot \log(0.2) + 1 \cdot \log(0.7) + 0 \cdot \log(0.1)\big] = -\log(0.7) \approx 0.357.$$

若另一模型预测为 $\hat{y} = [0.4, 0.3, 0.3]$，则

$$Loss = -\log(0.3) \approx 1.204.$$

第二个模型对正确类别的预测概率仅为 30%，与真实情况差距更大，因此交叉熵损失显著更高。

5 互信息

5.1 互信息的定义

信息熵可以表征单个概率分布的不确定性，KL 散度和交叉熵可以表征两个概率分布之间的差异。为量化两个随机变量之间的关系，引入**互信息**。

考虑两个随机变量 X 与 Y。若 X 与 Y 完全独立，则知道 X 的值不能提供任何关于 Y 的信息，此时互信息为零，即 $I(X;Y) = 0$。相反，若 X 与 Y 高度相关，则知道 X 的值能显著减少关于 Y 的不确定性，此时互信息大于零，即 $I(X;Y) > 0$。

互信息的基本定义为

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X),$$

其中 $H(X)$、$H(Y)$ 为变量 X 与 Y 的信息熵；$H(X|Y)$、$H(Y|X)$ 为**条件熵**，表示已知一个变量后，另一个变量的剩余不确定性。

条件熵 $H(X|Y)$ 的定义为：对 Y 的所有可能取值 y 按边缘概率 $P(y)$ 加权平均，计算已知 $Y = y$ 后 X 的剩余不确定性，即

$$H(X|Y) = \sum_{y} P(y) H(X|Y = y) = -\sum_{x}\sum_{y} P(x, y) \log P(x|y).$$

此处 $P(x, y)$ 为联合概率质量函数，$P(x|y) = P(x, y)/P(y)$ 为条件概率质量函数；最后一步利用了 $P(y) \cdot P(x|y) = P(x, y)$。

定理：互信息满足如下等价关系

$$I(X;Y) = H(X) + H(Y) - H(X,Y).$$

证明：需先建立条件熵与联合熵之间的关系。联合熵 $H(X,Y)$ 定义为

$$H(X,Y) = -\sum_x \sum_y P(x,y) \log P(x,y).$$

考察 $H(X,Y) - H(Y)$：

$$\begin{aligned}
H(X,Y) - H(Y) &= -\sum_x \sum_y P(x,y) \log P(x,y) - \Big[-\sum_y P(y) \log P(y)\Big] \\
&= -\sum_x \sum_y P(x,y) \log P(x,y) + \sum_y \Big[\sum_x P(x,y)\Big] \log P(y) \\
&= -\sum_x \sum_y P(x,y) \log P(x,y) + \sum_x \sum_y P(x,y) \log P(y) \\
&= \sum_x \sum_y P(x,y) \big[\log P(y) - \log P(x,y)\big] \\
&= \sum_x \sum_y P(x,y) \log \frac{P(y)}{P(x,y)} \\
&= -\sum_x \sum_y P(x,y) \log \frac{P(x,y)}{P(y)}.
\end{aligned}$$

由条件概率的定义 $P(x|y) = P(x,y)/P(y)$，代入得

$$\begin{aligned}
H(X,Y) - H(Y) &= -\sum_x \sum_y P(x,y) \log P(x|y) \\
&= H(X|Y).
\end{aligned}$$

于是得到**条件熵的链式关系**

$$H(X|Y) = H(X,Y) - H(Y).$$

将此关系代入互信息的定义 $I(X;Y) = H(X) - H(X|Y)$：

$$\begin{aligned}
I(X;Y) &= H(X) - \big[H(X,Y) - H(Y)\big] \\
&= H(X) + H(Y) - H(X,Y).
\end{aligned}$$

证毕。

对称地，由 $H(Y|X) = H(X,Y) - H(X)$ 可得 $I(X;Y) = H(Y) - H(Y|X)$，与原始定义自洽。

5.2 互信息的 KL 散度表达

互信息可通过 KL 散度表达为如下等价形式：

$$I(X;Y) = D_{\mathrm{KL}}\big(P(X,Y)\|P(X)P(Y)\big).$$

即互信息是联合分布 $P(X,Y)$ 与独立分布 $P(X)P(Y)$ 之间的 KL 散度，用于衡量变量间的独立性偏离程度。

证明：从 KL 散度的定义出发，对离散情形有

$$D_{\mathrm{KL}}\big(P(X,Y)\|P(X)P(Y)\big) = \sum_x \sum_y P(x,y) \log \frac{P(x,y)}{P(x)P(y)}.$$

展开对数项：

$$\begin{aligned} D_{\mathrm{KL}}\big(P(X,Y)\|P(X)P(Y)\big) &= \sum_x \sum_y P(x,y)\big[\log P(x,y) - \log P(x) - \log P(y)\big] \\ &= \sum_x \sum_y P(x,y) \log P(x,y) - \sum_x \sum_y P(x,y) \log P(x) \\ &\quad - \sum_x \sum_y P(x,y) \log P(y). \end{aligned}$$

分别处理三项。

第一项：

$$\sum_x \sum_y P(x,y) \log P(x,y) = -H(X,Y).$$

第二项：先对 y 求和，利用 $\sum_y P(x,y) = P(x)$（边缘化），得

$$\begin{aligned} \sum_x \sum_y P(x,y) \log P(x) &= \sum_x \Big[\sum_y P(x,y)\Big] \log P(x) \\ &= \sum_x P(x) \log P(x) \\ &= -H(X). \end{aligned}$$

第三项：先对 x 求和，利用 $\sum_x P(x,y) = P(y)$（边缘化），得

$$\begin{aligned} \sum_x \sum_y P(x,y) \log P(y) &= \sum_y \Big[\sum_x P(x,y)\Big] \log P(y) \\ &= \sum_y P(y) \log P(y) \\ &= -H(Y). \end{aligned}$$

将三项结果代回：

$$\begin{aligned} D_{\mathrm{KL}}\big(P(X,Y)\|P(X)P(Y)\big) &= -H(X,Y) - (-H(X)) - (-H(Y)) \\ &= H(X) + H(Y) - H(X,Y) \\ &= I(X;Y). \end{aligned}$$

最后一步利用了上一小节已证明的 $I(X;Y) = H(X)+H(Y)-H(X,Y)$。证毕。

5.3 互信息的离散形式与连续形式

由 KL 散度表达 $I(X;Y) = D_{\mathrm{KL}}(P(X,Y)\|P(X)P(Y))$ 直接代入 KL 散度的离散定义，即得离散变量的互信息计算公式：

$$I(X;Y) = \sum_{x\in X}\sum_{y\in Y} P(x,y)\log\frac{P(x,y)}{P(x)P(y)},$$

其中 $P(x,y)$ 为 X 与 Y 的联合概率质量函数，$P(x) = \sum_y P(x,y)$ 与 $P(y) = \sum_x P(x,y)$ 分别为 X 与 Y 的边缘概率质量函数。

对数项中的分子 $P(x,y)$ 为两变量同时取特定值的联合概率，分母 $P(x)P(y)$ 为假设两变量独立时的概率乘积。该比值在独立假设成立时等于 1，对数为 0，求和结果为 0，与互信息为零的结论一致；当两变量存在统计依赖时，该比值偏离 1，对数非零，累积形成正的互信息。

对连续随机变量，将上述求和替换为积分，得到连续变量的互信息计算公式：

$$I(X;Y) = \int_X\int_Y p(x,y)\log\frac{p(x,y)}{p(x)p(y)}\,dx\,dy,$$

其中 $p(x,y)$ 为 X 与 Y 的联合概率密度函数，$p(x) = \int_Y p(x,y)\,dy$ 与 $p(y) = \int_X p(x,y)\,dx$ 分别为 X 与 Y 的边缘概率密度函数。推导逻辑与离散情形完全一致，仅将求和运算替换为积分运算。

5.4 互信息与深度学习

信息瓶颈理论指出，为寻找输入数据的最优表示，应当实现两个目标：**既要求表示尽可能精简（最小化 $I(X;T)$），又需保留足够信息以完成目标任务（最大化 $I(T;Y)$）**。其中 $I(X;T)$ 表示输入 X 与中间特征 T 的互信

息，$I(T;Y)$ 衡量特征 T 对标签 Y 的预测能力，β 为平衡压缩与预测的系数。该目标被形式化为如下优化问题：

$$\min_{\beta}\big[I(X;T)-\beta I(T;Y)\big].$$

通过对训练过程中互信息的研究发现，神经网络的训练过程可分为两个阶段：

拟合阶段：网络通过快速增加 $I(X;T)$ 记忆训练数据细节，包括有效特征与噪声，此时训练损失迅速下降但泛化能力较弱。

压缩阶段：网络开始降低 $I(X;T)$，同时提升 $I(T;Y)$，通过丢弃与任务无关的冗余信息来增强泛化性。这一阶段损失下降趋缓，但模型逐渐从记忆数据过渡到学习本质规律。

6 第一定律与能量函数

6.1 热力学第一定律

热力学第一定律本质上是能量守恒定律在热力学系统中的具体表述。其声明：能量既不能凭空产生，也不能凭空消失，只能从一种形式转化为另一种形式，或者从一个系统传递给另一个系统。在能量的转化和传递过程中，总能量保持不变。数学表示为

$$\Delta U = Q + W,$$

其中 ΔU 为系统内能的变化量（$\Delta U>0$ 表示内能增加，$\Delta U<0$ 表示内能减少）；Q 为系统吸收的热量（$Q>0$ 表示系统吸热，$Q<0$ 表示系统放热）；W 为外界对系统做的功（$W>0$ 表示外界对系统做功，$W<0$ 表示系统对外界做功）。

6.2 能量函数与能量基模型

受能量守恒定律启发，引入**能量基函数**概念，作为人工智能模型内在知识表示与推理计算的核心工具。**能量基模型**为系统的可能状态赋予明确、量化的能量值，从而使模型能够更明确地识别和学习不同状态之间的动态关系。

能量基模型用能量函数 $E(X,Y)$ 衡量观测变量 X 与待预测变量 Y 之间的相容性。给定 X 时，模型产生使能量函数 $E(X,Y)$ 最小化的输出 Y。

给定训练集 S，构建和训练能量基模型需要设计以下四个组件：

- **架构**：能量函数 $E(W, Y, X)$ 的内部结构；
- **推理算法**：给定任意 X，找到最小化 $E(W, Y, X)$ 的 Y 值的方法；
- **损失函数**：$L(W, S)$，使用训练集度量能量函数的质量；
- **学习算法**：给定训练集，在能量函数簇中找到使损失函数最小的参数 W。

6.3 回归问题的能量基模型示例

以回归问题为例，目标是使回归函数 $G_W(X)$ 的输出尽可能接近真实值 Y。最好的推断可表示为 $Y^* = G_W(X)$。定义能量函数为回归函数输出与正确值之间的平方差：

$$E(W, Y, X) = \frac{1}{2}\|G_W(X) - Y\|^2.$$

在训练过程中，对于给定的训练数据 $\{X_i, Y_i\}_{i=1}^P$，训练即优化模型参数 W，使得训练集上的整体匹配程度最好（能量最低），形式化表达为

$$W^* = \arg\min_W L_{\text{energy}}(W, S) = \frac{1}{P}\sum_{i=1}^{P} E(W, Y_i, X_i).$$

在推理过程中，给定新输入 X 但不知真实标签 Y 时，推理阶段的目的即通过回归函数找到最可能的输出 Y 值。根据能量基模型的定义，推理过程相当于选择能量函数最小的输出值：

$$Y^* = \arg\min_Y E(W^*, Y, X).$$

对于此回归问题，这等价于 $Y^* = G_{W^*}(X)$。

7 熵增定律与模型设计

7.1 熵增定律

回顾：熵（Entropy, S）作为关键物理量，用于衡量系统的无序程度或信息丢失程度。熵的增大意味着系统从相对有序（低熵）的状态演变为更加混乱、更加无序（高熵）的状态。

在热力学范畴内，熵的变化与能量转化过程密切相关。**热力学第二定律**表明，在孤立系统中，熵总是趋向于增加，或在理想可逆情况下保持不变，可表述为

$$\Delta S \geq 0.$$

为精确描述和计算熵的变化，克劳修斯提供了基于可逆过程的熵变定义：在经历可逆过程的系统中，熵的变化量等于系统在该过程中吸收或放出的微小热量 δQ_{rev} 与其所处绝对温度 T 之比的积分：

$$\Delta S = \int \frac{\delta Q_{\text{rev}}}{T}.$$

7.2 最大熵正则化

当模型处理类别数量庞大、数据不平衡的任务时，容易出现**过拟合**现象。典型症状为模型对训练数据预测过于自信，泛化能力差，即输出的概率分布熵值很低。

正是基于这种认识，研究者们提出并应用了**熵正则化**技术，旨在缓解过拟合并提升模型泛化能力。其核心思想是在原有损失函数（通常是交叉熵损失）基础上额外加入一个与模型输出概率分布的熵直接相关的正则项。

在典型分类问题中，给定输入样本 z 及其真实标签 y，模型输出对应 C 个类别的原始得分 $\{z_i\}_{i=1}^{C}$，随后通过 Softmax 函数转换为概率分布：

$$p_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}}.$$

该任务中最常用的损失函数为交叉熵损失，定义为

$$L_{\text{CE}} = -\log p_y,$$

其中 p_y 为模型预测输入样本属于真实类别 y 的概率。最小化交叉熵损失旨在最大化正确类别的预测概率。

计算 L_{CE} 相对于模型原始得分 z_i 的梯度，使用链式法则：

$$\frac{\partial L_{\text{CE}}}{\partial z_i} = \sum_{k=1}^{C} \frac{\partial L_{\text{CE}}}{\partial p_k} \frac{\partial p_k}{\partial z_i}.$$

Softmax 的标准导数为

$$\frac{\partial p_y}{\partial z_i} = p_y(\delta_{yi} - p_i),$$

其中 δ_{yi} 为克罗内克函数：

$$\delta_{yi} = \begin{cases} 1, & i = y, \\ 0, & i \neq y. \end{cases}$$

代入 Softmax 导数：

$$\frac{\partial L_{\mathrm{CE}}}{\partial z_i} = -\frac{1}{p_y} \cdot p_y(\delta_{yi} - p_i) = -(\delta_{yi} - p_i) = p_i - \delta_{yi}.$$

整理得

$$\frac{\partial L_{\mathrm{CE}}}{\partial z_i} = \begin{cases} p_i - 1 < 0, & i = y, \\ p_i > 0, & i \neq y. \end{cases}$$

从梯度表达式可以看出，在优化过程中，模型会持续增大 z_y 并减小所有 z_i（$i \neq y$），这将导致模型在训练集上的预测概率分布趋向于尖锐的 One-Hot 向量，从而造成数据过拟合，同时产生非常低的信息熵。

为对抗这种过度自信的趋势，**最大熵正则化**（Maximum Entropy Regularization, MER）策略通过对模型输出概率分布 p 的熵进行正则化来改善模型泛化能力。信息熵 $H(p)$ 的标准定义为

$$H(p) = -\sum_{i=1}^{C} p_i \log p_i,$$

其中 C 为目标类别数，p_i 为第 i 类目标的预测概率。熵 $H(p)$ 度量了概率分布的不确定性程度：当 p 为 One-Hot 向量时熵取得最小值 0；当 p 为均匀分布 $p_i = 1/C$ 时熵最大。

最大熵正则化旨在增大输出分布的熵，因此将负熵项 $-H(p)$ 作为正则惩罚项添加到原始交叉熵损失中：

$$L_{\mathrm{MER}} = -H(p) = \sum_{i=1}^{C} p_i \log p_i.$$

最终的正则化损失函数定义为

$$L_{\mathrm{REG}} = L_{\mathrm{CE}} + \lambda L_{\mathrm{MER}} = -\log p_y + \lambda \sum_{i=1}^{C} p_i \log p_i,$$

其中 $\lambda \geq 0$ 为超参数，用于权衡原始交叉熵损失与熵正则化项的重要性。这两个目标存在一定的冲突：L_{CE} 倾向于使分布尖锐化，而 L_{MER} 倾向于使分布平坦化，超参数 λ 用于控制两者之间的平衡。当 $\lambda = 0$ 时，上式退化为标准的交叉熵损失。

7.3 熵增与扩散模型

扩散模型是一类强大的深度生成模型，在图像、音频等多种数据模态的生成任务中取得显著进展。该类模型在理论层面上直接受到非平衡热力学与统计物理中熵演化过程的启发，其核心思想是通过定义逐步破坏数据结构的正向扩散过程，随后训练深度学习模型掌握该过程的逆向操作，从而实现从简单噪声分布到复杂数据分布的有效映射。

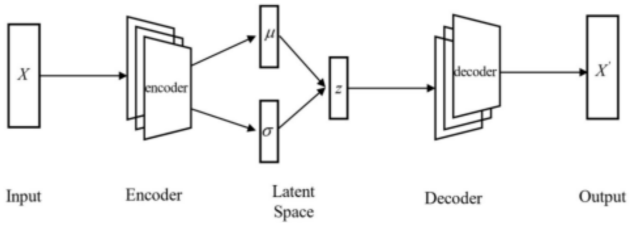

图 2: 扩散模型

扩散模型的核心原理由两个方向相反但紧密关联的马尔可夫过程构成：正向扩散过程用于破坏数据结构（对应数据空间中的熵增过程），反向生成过程用于恢复数据结构（对应受约束的熵减过程）。

7.3.1 扩散模型——正向过程

正向过程 q 是一个预先设定好、无需学习的马尔可夫链。它模拟数据样本 x_0（来自真实数据分布）随着时间 t 从 0 到 T 逐步被噪声污染的过程。在每一步 t，噪声以高斯分布的形式被添加到前一步的样本中，其方差由预设的调度 $\beta_t \in (0,1)$ 控制：

$$q(x_t|x_{t-1}) := \mathcal{N}(x_t; \sqrt{1-\beta_t}\, x_{t-1}, \beta_t I).$$

其中 $\sqrt{1-\beta_t}$ 因子用于控制信号的衰减，$\beta_t I$ 定义了添加的高斯噪声的协方差。

这个过程可看作将数据 x_0 逐渐溶解到噪声中的过程，类似于物理系统在热浴中逐渐达到热平衡、丢失初始状态信息的过程。从信息论角度看，随着噪声不断叠加，样本分布的不确定性持续增加，其信息熵呈单调上升趋势，最终趋向于近似最大熵的分布（通常为标准高斯分布）。

由于每一步添加的噪声都是高斯噪声，这个过程有一个重要特性：任意时刻 t 的带噪样本 x_t 可以直接由初始样本 x_0 和一个标准高斯噪声表示。

令 $\alpha_t = 1 - \beta_t$，则 x_t 的条件分布为

$$q(x_t|x_0) = \mathcal{N}\big(x_t; \sqrt{\bar{\alpha}_t}\, x_0, (1-\bar{\alpha}_t)I\big), \quad \bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s.$$

或等价地写为采样形式：

$$x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\, \epsilon, \quad \epsilon \sim \mathcal{N}(0, I).$$

其中 $\sqrt{\bar{\alpha}_t}$ 代表了原始信号 x_0 在时刻 t 的保留比例，$\sqrt{1-\bar{\alpha}_t}$ 代表了噪声的标准差。当 $t \to T$ 且 β_t 的选择使得 $\bar{\alpha}_T \to 0$ 时，x_T 的分布将趋近于与 x_0 无关的标准高斯分布 $p(x_T) = \mathcal{N}(0, I)$。该正向过程将复杂的数据分布 $q(x_0)$ 转化为简单、易处理的先验分布。

7.3.2 扩散模型——反向过程

反向生成过程 p_θ 学习如何逆转上述正向扩散过程，该过程同样是一个马尔可夫链。反向生成过程从已知的先验分布 $p(x_T) = \mathcal{N}(0, I)$ 开始，逐步从 x_T 迭代到 x_0，尝试在每一步去除噪声、恢复数据结构，定义为

$$p_\theta(x_{0:T}) := p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t).$$

关键在于学习每一步的反向转移概率 $p_\theta(x_{t-1}|x_t)$。一些物理学家证明了，若正向过程的每步噪声方差 β_t 足够小，则对应的反向过程转移概率也具有与正向过程相同的函数形式。既然正向过程是高斯过程，反向转移也可合理地参数化为高斯分布：

$$p_\theta(x_{t-1}|x_t) := \mathcal{N}\big(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)\big).$$

其中均值 $\mu_\theta(x_t, t)$ 与协方差 $\Sigma_\theta(x_t, t)$ 是需要通过神经网络学习的目标。

7.3.3 扩散模型——训练机制

训练扩散模型的主要目标是通过学习反向生成过程 p_θ 中的参数 θ，使得该过程中采样得到的 x_0 的分布 $p_\theta(x_0)$ 尽可能接近真实数据分布。这一目标等价于最大化模型对真实数据的对数似然估计：

$$\mathbb{E}_{x_0 \sim q(x_0)}\big[\log p_\theta(x_0)\big].$$

鉴于直接计算和优化 $\log p_\theta(x_0)$ 是困难的，转而优化其**变分下界**（Variational Lower Bound, VLB），这一做法借鉴了变分推断的思路。

推导过程：[2]

扩散模型的训练目标是最大化模型对真实数据的对数似然：

$$\mathbb{E}_{x_0\sim q(x_0)}[\log p_\theta(x_0)]. \tag{1}$$

引入扩散过程中的隐变量 $x_{1:T} = (x_1, x_2, \ldots, x_T)$。正向扩散为

$$q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}),$$

反向生成过程为

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t),$$

其中 $p(x_T) = \mathcal{N}(0, I)$。

从对数似然出发，对任意 x_0 有

$$\log p_\theta(x_0) = \log \int p_\theta(x_{0:T})\, dx_{1:T}. \tag{2}$$

该积分不可直接求解，引入变分分布（选用正向扩散）：

$$q(x_{1:T}|x_0).$$

插入变分分布：

$$\log p_\theta(x_0) = \log \int q(x_{1:T}|x_0) \frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)}\, dx_{1:T}. \tag{3}$$

利用 Jensen 不等式（$\log \mathbb{E}[\cdot] \geq \mathbb{E}[\log \cdot]$）获得变分下界：

$$\log p_\theta(x_0) \geq \mathbb{E}_{q(x_{1:T}|x_0)} \left[\log \frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)} \right]. \tag{4}$$

定义 VLB（ELBO）：

$$L_{\text{vlb}} = \mathbb{E}_{q(x_{1:T}|x_0)} \left[\log \frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)} \right]. \tag{5}$$

展开联合分布。分子（反向过程）为

$$\log p_\theta(x_{0:T}) = \log p(x_T) + \sum_{t=1}^{T} \log p_\theta(x_{t-1}|x_t). \tag{6}$$

[2]这部分推导内容不考，建议心情好的时候学习一下，是这门课为数不多有用的知识。

分母（正向过程）为

$$\log q(x_{1:T}|x_0) = \sum_{t=1}^{T} \log q(x_t|x_{t-1}). \tag{7}$$

将 (6) 与 (7) 代入 (5) 并整理：

$$L_{\text{vlb}} = \mathbb{E}_q \left[\log p(x_T) + \sum_{t=1}^{T} \log p_\theta(x_{t-1}|x_t) - \sum_{t=1}^{T} \log q(x_t|x_{t-1}) \right]. \tag{8}$$

引入真实后验，构造 KL 项。注意到正向扩散是马尔可夫过程，利用贝叶斯关系可写出

$$q(x_{t-1}|x_t, x_0) = \frac{q(x_t|x_{t-1})q(x_{t-1}|x_0)}{q(x_t|x_0)}.$$

将上式整理为 KL 散度形式，得到三项分解：

（1）终点 KL（先验匹配项）：

$$L_T = D_{\text{KL}}\big(q(x_T|x_0) \| p(x_T)\big).$$

（2）中间步骤 KL（核心训练项）：对 $t = 2, \ldots, T$，

$$L_{t-1} = D_{\text{KL}}\big(q(x_{t-1}|x_t, x_0) \| p_\theta(x_{t-1}|x_t)\big).$$

（3）重建项（数据项）：

$$L_0 = -\log p_\theta(x_0|x_1).$$

最终的 VLB 分解形式为

$$L_{\text{vlb}} = \underbrace{D_{\text{KL}}\big(q(x_T|x_0) \| p(x_T)\big)}_{L_T} + \sum_{t=2}^{T} \underbrace{D_{\text{KL}}\big(q(x_{t-1}|x_t, x_0) \| p_\theta(x_{t-1}|x_t)\big)}_{L_{t-1}} \underbrace{- \log p_\theta(x_0|x_1)}_{L_0}. \tag{9}$$

7.3.4 变分下界各项的含义

L_T：衡量扩散过程最终状态 x_T 的分布 $q(x_T|x_0)$ 与先验分布 $p(x_T)$（通常为 $\mathcal{N}(0, I)$）之间的 KL 散度。由于正向过程 q 与先验 $p(x_T)$ 都是预先固定的，这一项在训练中不依赖于模型参数，可视为常数或忽略。

L_{t-1} $(t = 2, \ldots, T)$：这是训练的核心部分。它衡量的是模型学习到的反向转移分布 $p_\theta(x_{t-1}|x_t)$ 与真实反向后验 $q(x_{t-1}|x_t, x_0)$ 之间的 KL 散度。神经网络通过最小化此项来学习去噪。

L_0：重构项，表示从 x_1（经过一步去噪的状态）生成最终无噪数据 x_0 的负对数似然，要求模型在最后一步能够准确恢复原始数据。

其中 L_{vlb} 是 $\log p_\theta(x_0)$ 的一个下界，最大化 VLB 相当于最大化数据的对数似然。从信息论角度看，该下界的优化过程可以理解为在熵增约束下，最小化模型分布与真实分布之间的相对熵差异。

8 量子物理回顾

8.1 量子态

量子态的数学表征结合了两种已有方法：波函数描述和态矢量表征。

8.1.1 波函数描述

波函数在一维空间中是一个复数函数，定义为 $\psi(x)$。

$\psi(x)$ 表示粒子在位置 x 处的波函数，其模平方 $|\psi(x)|^2$ 表示粒子位置的概率密度（重点）。通过概率密度可以得到粒子在某一区间内出现的概率。该波函数满足归一化条件：

$$\int |\psi(x)|^2 \, dx = 1$$

8.1.2 态矢量表征

量子态还可以用一个复数矢量表示，称为态矢量。态矢量通常用一个有限维或无限维列向量表示，通常用狄拉克（Dirac）符号 $|\psi\rangle$ 表示。

态矢量分为右矢与左矢，从矢量角度看，右矢 $|\psi\rangle$ 是一维列向量，左矢 $\langle\psi|$ 是一维行向量。

态矢量的内积表示两个态之间的重叠程度或者相似度。对于两个量子态 $|\psi_1\rangle$ 与 $|\psi_2\rangle$，它们的内积将被表示为 $\langle\psi_1|\psi_2\rangle$，内积的模平方表示两个态之间的相似性，即 $|\langle\psi_1|\psi_2\rangle|^2$(重点)。

8.1.3 量子态叠加特性

一个量子态可以由多个量子态叠加表示，即量子态的叠加特性。用态势量来描述，如果一个系统有不同的本征态 $|\psi_1\rangle, |\psi_2\rangle, |\psi_3\rangle$ 等，那该系统的任意一个总态可表示为

$$|\psi\rangle = c_1|\psi_1\rangle + c_2|\psi_2\rangle + c_3|\psi_3\rangle + ...$$

其中，c_1, c_2, c_3 表示各个本征态在总态中的权重，c_1^2, c_2^2, c_3^2 表示了对应本征态被测量到的概率。

8.1.4 量子态纠缠特性

当几个粒子在彼此相互作用后，由于各个粒子所拥有的特性已综合成为整体性质，无法单独描述各个粒子的性质，只能描述整体系统的性质，则称这现象为量子纠缠。究其本质，其实是两个量子态之间存在相关性，并不完全独立。对于有两个量子态的系统（例：$|01\rangle$ 表示第一个量子态为 0，第二个量子态为 1）。

$$|\Psi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$$

如果该量子态可以写成

$$|\Psi\rangle = (a_1|0\rangle + b_1|1\rangle)(c_1|0\rangle + d_1|1\rangle)$$

即 $ad = bc$，则说明这两个量子态是完全独立的，不存在纠缠。反之，若无法写成上述格式，即 $ad \neq bc$，则说明这两个量子态是存在纠缠的。在纠缠中有更特殊的情况，即 Bell 态（两个量子态完美相关）

$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

在 Bell 态中，当一个量子态确定，另一个量子态也就确定了。

8.2 经典比特与量子比特

8.2.1 比特（bit）

记录经典信息的二进制存储单元称为经典比特 (bit)，由经典状态的 0 和 1 表示。

对于量子信息而言，记录量子信息的存储单元称为量子比特 (qubit)，一个 qubit 的状态是一个二维复数空间的矢量，它的两个极化状态和 $|0\rangle, |1\rangle$ 对应于经典状态的 0 和 1。

如果把 $|0\rangle, |1\rangle$ 当成列向量来处理，那么其构成二维复数空间的一对正交基底，即长度为 1，内积为 0 的复数向量。

qubit 与 bit 本质上的不同在于，除了 $|0\rangle, |1\rangle$ 状态外，qubit 可以是两个状态的任意叠加状态。则 qubit 的瞬间值可以取

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix}\alpha \\ \beta\end{bmatrix}$$

其中要满足归一化条件 $\alpha^2 + \beta^2 = 1$

8.2.2 量子比特的测定

通过一个被称为是测定或观测的过程，可以把一个 qubit 的状态以概率幅的方式变换成 bit 信息。对于

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix}\alpha \\ \beta\end{bmatrix}$$

我们左乘一个 $|0\rangle$, 或者 $|1\rangle$, 我们有：

$$\langle 0|\psi\rangle = \langle 0|(\alpha|0\rangle + \beta|1\rangle) = \begin{bmatrix}1 \\ 0\end{bmatrix}^T \begin{bmatrix}\alpha \\ \beta\end{bmatrix} = \alpha$$

$$\langle 1|\psi\rangle = \langle 1|(\alpha|0\rangle + \beta|1\rangle) = \begin{bmatrix}0 \\ 1\end{bmatrix}^T \begin{bmatrix}\alpha \\ \beta\end{bmatrix} = \beta$$

其中 α^2, β^2 为量子比特 $|\Psi\rangle$ 变换为 bit0 与 bit1 的对应概率。

也就是说，当我们想从 qubit 中得到我们存储的对应信息时，我们只要对这个 qubit 左乘对应的本征态即可。

8.3 量子比特对

这里我们讨论两个量子比特组成的量子对及它们的性质。两个量子比特组成的不重复的量子比特对为 $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$，和单个量子比特一

样，量子比特对也可以取这些基底状态的叠加状态。对一般的量子比特对来说

$$|\Psi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$$

其中 $a^2 + b^2 + c^2 + d^2 = 1$。

8.4 量子逻辑门

量子逻辑门可以大体理解为作用在 qubit 上的变换，多个量子逻辑门又可以构成量子电路，常用量子电路（又称“量子线路”）对量子比特进行操作。下介绍几种常见的量子逻辑门：

8.4.1 量子逻辑非门

可以用以下 2×2 的泡利矩阵 σ_x 表示：

$$\sigma_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

对状态 $\alpha|0\rangle + \beta|1\rangle$ 实施该操作得到如下结果（用矢量形式表示）

$$\sigma_x \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix}$$

本质就是把 qubic 变成 $|0\rangle, |1\rangle$ 状态的概率交换。

8.4.2 位相翻转门

可以用以下 2×2 的泡利矩阵 σ_z 表示：

$$\sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

对状态 $\alpha|0\rangle + \beta|1\rangle$ 实施该操作得到如下结果（用矢量形式表示）

$$\sigma_z \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \alpha \\ -\beta \end{bmatrix}$$

本质就是对 qubic 的状态 $|0\rangle$ 不改变，状态 $|1\rangle$ 发生位相翻转变成 $-|1\rangle$

8.4.3 H 门

可以用矩阵 H 表示：

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

对状态 $\alpha|0\rangle + \beta|1\rangle$ 实施该操作得到如下结果（用矢量形式表示）

$$H \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} \alpha + \beta \\ \alpha - \beta \end{bmatrix}$$

本质相当于对基底进行了变换，从以 $|0\rangle, |1\rangle$ 为基底，变成了以 $\frac{|0\rangle+|1\rangle}{\sqrt{2}}, \frac{|0\rangle-|1\rangle}{\sqrt{2}}$，为基底。即让 $|0\rangle, |1\rangle$ 进行了叠加，为干涉产生了可能。

9 纯量子算法——全量子学习

9.1 QuantumFlow 总体架构

下图展示了 QuantumFlow 总体架构。总的来说，左侧的量子神经网络是一种抽象结构，相当于神经网络中的各种网络架构；右侧的量子电路是一种物理架构，相当于正常计算机 GPU 中的各个算子甚至更低层的实现。同时左右两者是协同设计的，如果硬件不支持某些门那么网络结构要改，如果电路太深那么模型结构要简化。

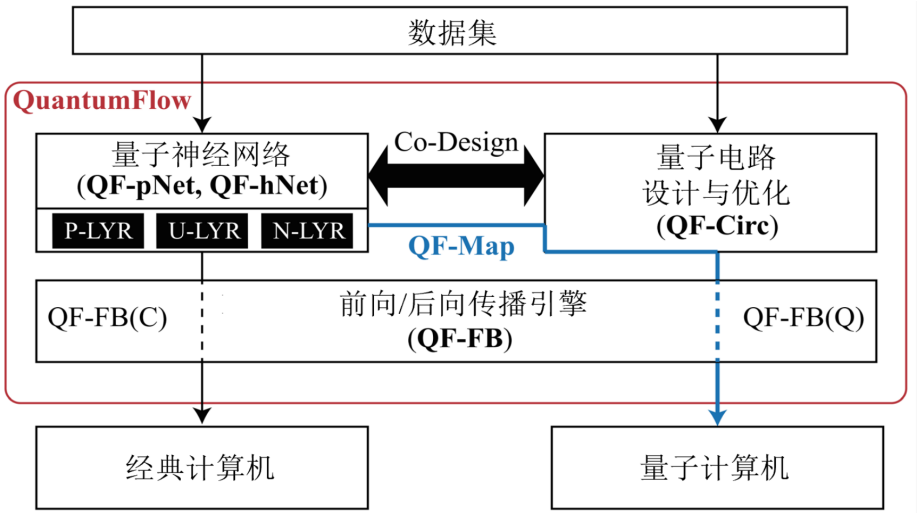

图 3: QuantumFlow 架构

其中的 QF-Map，它起到桥梁的作用，把抽象的神经网络层映射成具体量子电路（从抽象结构到底层电路实现）。

QF-FB 是用来进行正向传播和反向传播的。QF-FB(C) 是在经典计算机中进行正向传播与梯度计算的，QF-FB(Q) 是在量子计算机中进行量子态演化（正向）与参数偏移法求梯度（反向）的。

9.2 QF-pNet

QF-pNet 是概率型量子神经网络，其核心是通过量子逻辑门处理随机变量 (量子比特编码)，利用量子比特的概率特性实现网络最大灵活性。

QF-pNet 由多级 P-LYR（概率神经计算层）构成，结构如下图所示：

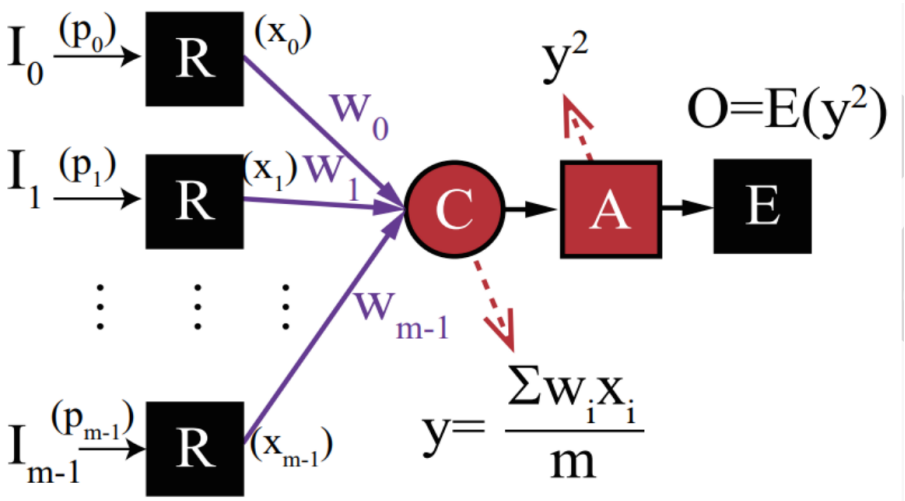

图 4: P—LYR

P-LYR 包含以下四种不同操作：

9.2.1 R

随机变量建模，就是将输入由实数变为服从两点分布的随机变量。

$$X_i \sim \begin{cases} 1, & p_i \\ 0, & 1-p_i \end{cases}$$

其中 $p_i = f(x_i)$。和量子态的对应关系：

$$|\psi\rangle = \sqrt{1-p}|0\rangle + \sqrt{p}|1\rangle$$

（用概率分布模拟了量子叠加）

9.2.2 C

对随机变量进行加权平均和

$$S = \frac{\sum_i \omega_i x_i}{m}$$

9.2.3 A

非线性激活

$$y = g(S)$$

例如，$y = S^2$

9.2.4 E

通过期望得到 [0,1] 内的实数

$$y = E(Y)$$

(从量子态到经典实数的转变)

9.3 QF-hNet

为了降低 QF-pNet 的计算复杂度以便更好地实现量子计算优势，混合型量子神经网络 QF-hNet 应运而生。由 P-LYR 与 U-LYR（酉神经计算层）两类计算结构组成，核心是利用酉矩阵的特性来最小化实现量子优势所需的量子门数量。U-LYR 结构示意图如下

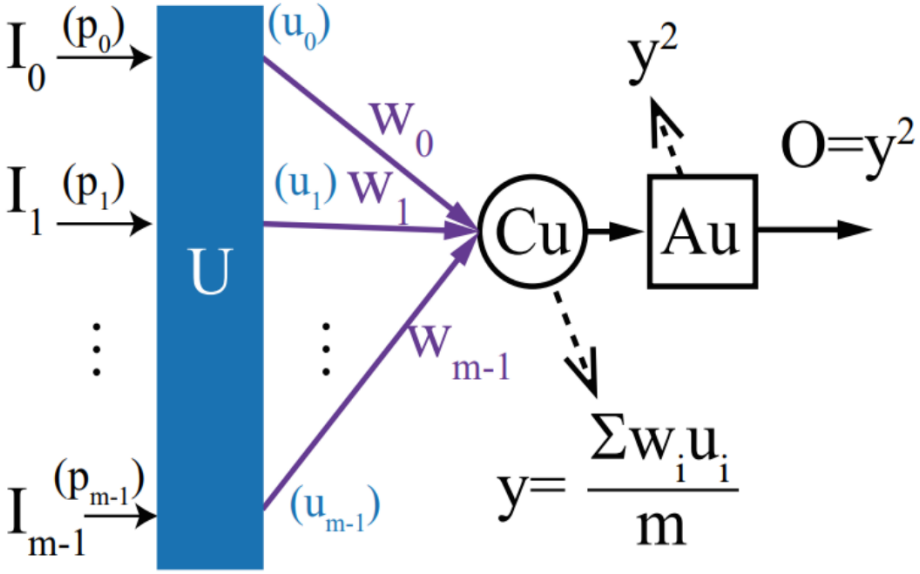

图 5: U—LYR

U-LYR 包含以下三种不同操作：

9.3.1 U

酉矩阵转换，就是把 m 维数据 $x \in R^m$ 映射为一个 m×m 酉矩阵 $U(x)$ 的第一列。本质上就是把一个现实中的输入数据建模成了一个量子态，从而可以在量子计算机中进行后续操作。在数学上

$$x \to |\psi(x)\rangle = U(x)|0\rangle$$

其中 $U(x)$ 为酉矩阵，满足 $U^TU = I$，并且 $U(x)$ 的第一列就是量子态 $|\psi(x)\rangle$。

这里有一个数学上的表示 $U(x)|0\rangle$，这其实等价于 $U(x)$ 矩阵的第 1 列。同理的，对 $U(x)|i\rangle$，等价于 $U(x)$ 的第 $i+1$ 列。

通过酉矩阵变化，我们得到了一个量子态 $|\psi(x)\rangle = \begin{bmatrix} u_1 & u_2 & ... & u_m \end{bmatrix}^T$ 其中 u_0 表示 $|\underbrace{000...0}_{k}\rangle$ 的系数，u_m 表示 $|\underbrace{111...1}_{k}\rangle$ 的系数。那么我们就可以用 $k = log_2(m)$ 个单量子态来描述这 m 维输入。

这也将后续需要的量子门数量从 QF-pNet 的 $O(2^k)$ 个降至 QF-hNet 的 $O(k^2)$ 个，更好地实现了量子优势。（这里的 k^2 是由于我们不仅要对每一个量子态后续进行门变换，还要对每两个量子态间进行交互）

9.3.2 C_u 与 A_u

这里的 C_u 与 A_u 和 P-LYR 中的 C,A 是一样的，不做多余解释。

9.4 QF-FB

9.4.1 P-LYR

先考虑前向传播，就是分别通过各个操作，最后通过 E 操作得到所需的输出实数。

再考虑后向传播，通过经典的随机梯度下降 SGD（这里 PPT 写错了，写成 SVD 了）进行更新即可。

9.4.2 U-LYR

先考虑前向传播，其他操作都与 P-LYR 相同，只有怎么构造一个 U 矩阵需要特殊考虑。对此，我们只需先构造一个方阵 $A \in R^{m*m}$，以输入数据

为第一列。再对其进行 SVD 分解即可 $U_m = SVD(A)$。

后向传播同 P-LYR 一致。

10 量子神经网络

10.1 量子聚类

聚类是一种无监督学习方式，其核心目标是：将数据对象划分为若干组（簇，cluster），使得同一簇内的数据尽可能相似，而不同簇之间的数据尽可能不相似。（机器学习课程中亦有所涉及）

10.1.1 经典 k-means 算法

对数据集 $X = \{x_1, ..., x_n\}$，预设聚类 k 类。首先，初始化聚类中心，常采用初始随机化方法，得到聚类中心 $C = \{c_1, ..., c_k\}$。（在 k-means++ 初始化中，随机选 1 个样本作为第一个中心 c_1 ，然后计算每个样本到该中心的距离，距离越大的样本被选为下一个中心的概率越高，直到选满 k 个中心）

其次，分配样本到最近的聚类中心，即对对每个样本 x_i，计算其到 k 个聚类中心的欧氏距离，将样本分配到最近的簇，完成一轮簇分配。

最后，更新聚类中心直至满足终止条件。对每个簇 c_j，计算其所有样本的均值作为新的聚类中心

$$c_j^{new} = \frac{1}{|c_j|} \sum_{x_i \in c_j} x_i$$

其中 $|c_j|$ 表示簇 c_j 中的样本数量。

对比更新前后的聚类中心，满足以下任一条件则停止迭代：

$$\begin{cases} \underset{j}{max} ||c_j^{new} - c_j^{old}|| < \delta \\ \text{迭代次数达到预设的最大次数} \end{cases}$$

10.1.2 量子 k-means 算法

主要区别在分配样本到最近的聚类中心阶段，其余阶段基本与经典 k-means 算法一致。

量子 k-means 通过对实向量间的距离作比较，通过寻找向量 u 与集合 $\{v\}$ 中的哪个量的距离最近，从而不断分类，进而不断自动获得聚类结果。

其最小化目标如下：

$$\underset{c}{argmin}|u - \frac{1}{n}\sum_{j=1}^{n} v_j^c|$$

其中 n 表示样本总数，c 表示聚类中第 c 个簇。

此时就引出了另一问题，就是我们如何去计算以量子态形式存储的两个向量之间的距离。首先构造了两个向量

$$|\varphi\rangle = \frac{1}{\sqrt{2}}(|u\rangle|0\rangle + \frac{1}{\sqrt{m}}\sum_{j=1}^{m}|v_j^c\rangle|j\rangle)$$

$$|\phi\rangle = \frac{1}{\sqrt{2}}(|u\rangle|0\rangle - \frac{1}{\sqrt{m}}\sum_{j=1}^{m}|v_j^c\rangle|j\rangle)$$

再通过两个向量的内积计算距离

$$D = \sqrt{2\left|\langle\phi|\varphi\rangle\right|^2}\, Z$$

其中 Z 表示归一化系数

$$Z = |u|^2 + (n^{-1}\sum_{j=1}^{n}|v_j^c|^2)$$

那么我们怎么用量子电路实现 $|\langle\phi|\varphi\rangle|^2$ 呢，通常是通过 controlled-SWAP 电路实现。

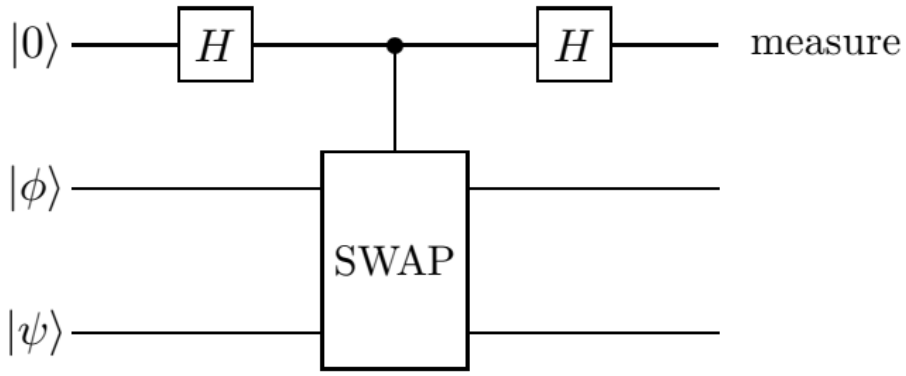

图 6: controlled-SWAP 电路

第一条路是控制比特，下面两条路是受控量子比特

第一条路的 H 门把 $|0\rangle$ 变成了 $\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)$

然后是 SWAP 操作，当控制比特为 $|0\rangle$ 时，不进行交换，当控制比特为 $|1\rangle$ 时，进行交换，得到 $\frac{1}{\sqrt{2}}(|0\rangle|\varphi\rangle|\phi\rangle+|1\rangle|\phi\rangle|\varphi\rangle)$

再通过一个 H 门得到 $state_{final}=\frac{1}{2}(|0\rangle(|\varphi\rangle|\phi\rangle+|\phi\rangle|\varphi\rangle)+|1\rangle(|\varphi\rangle|\phi\rangle-|\phi\rangle|\varphi\rangle))$

此时我们通过测量这个最终态为 0 的概率，有

$$P(0)=||\frac{1}{2}(|\varphi\rangle|\phi\rangle+|\phi\rangle|\varphi\rangle)||^2$$

逐项展开并计算得

$$P(0)=\frac{1+|\langle\phi|\varphi\rangle|^2}{2}$$

所以我们就可以通过测量最终态为 0 的概率来倒推 $|\langle\phi|\varphi\rangle|^2$ 了。

10.2 量子 SVM

SVM 支持向量机是解决分类问题的经典算法。以二分类问题为例：

给定线性可分数据集 $(x_i,y_i)_{i=1}^n$，其中 x_i 是输入特征向量，$y_i\in\{-1,1\}$ 是类别标签。核心目标是找一个最优的超平面 $w^Tx+b=0$ 使得两类数据点到超平面的间隔最大化。

10.2.1 经典 SVM

对点 x_i 到决策平面的距离为 $\frac{|w^Tx_i+b|}{||w||}=\frac{y_i(w^Tx_i+b)}{||w||}$

则最大间隔优化问题为 (也就是要将两类样本间离得最近的点距离最大化)

$$\underset{w,b}{argmax}\{\underset{i}{min}\frac{y_i(w^Tx_i+b)}{||w||}\}=\underset{w,b}{argmax}\{\frac{1}{||w||}\underset{i}{min}(w^Tx_i+b)y_i\}$$

我们可以发现当 $w\to\alpha w,b\to\alpha b$ 时，上式值不变，所以我们可以令 $\underset{i}{min}(w^Tx_i+b)y_i=1$，此时我们对所有点 x_i 有 $(w^Tx_i+b)y_i\geq 1$。

所以我们可以将原问题转换为最优化问题：

$$\underset{w,b}{min}\frac{1}{2}||w||^2,\ s.t.(w^Tx_i+b)y_i\geq 1$$

此时只需应用拉格朗日法，构造拉格朗日函数 $L(w,b,\alpha)=\frac{1}{2}||w||^2-\sum_{i=1}^n\alpha_i[(w^Tx_i+b)y_i-1]$，并对其求导得:

$$\frac{\partial L}{\partial w}=w-\sum_{i=1}^n\alpha_ix_iy_i=0$$

$$\frac{\partial L}{\partial b} = -\sum_{i=1}^{n} \alpha_i y_i = 0$$

后续再用 SMO 算法（笔者也不知道这是什么）求得最优的 $\{\alpha_i\}_{i=1}^n$，从而求得 $w = \sum_{i=1}^n \alpha_i x_i y_i$。

对于偏置项 b，PPT 上写的很清楚：

对于偏置项b的计算，可以通过**支持向量**（既满足$\alpha_i > 0$的样本）来确定。

假设x_s是一个支持向量，则有

$$y_s(w^T x_s + b) = 1,$$

两边同乘y_s从而可以解得

$$b = y_s - w^T x_s.$$

在实际计算中，为了提高稳定性，通常会对所有支持向量计算b后取平均值

$$b = \frac{1}{|S|}\sum_{s\in S}\left(y_s - w^T x_s\right)$$

其中S是支持向量的集合。

图 7: 偏置项 b 的计算

这只是针对可以线性区分的数据，对于非线性数据，我们需要用到核方法。

给出核函数定义:

$$k(x_i, x_j) = \phi(x_i)^T \phi(x_j)$$

其中非线性映射 $\phi : R^d \to H$，H 为高维特征空间。所以核函数 k 表示了高维空间的内积。实际对非线性数据操作时只需把推导式中出现的 $x_i^T x_j$ 内积替换为 $k(x_i, x_j)$ 即可。常见的核函数有：

线性核：$k(x_i, x_j) = x_i^T x_j$

多项式核：$k(x_i, x_j) = (x_i^T x_j + c)^p$

高斯核：$k(x_i, x_j) = \exp\left(-\| x_i - x_j \|^2 / (2\sigma^2)\right)$

图 8: 常见核函数

10.2.2 量子 SVM

一般来说 SVM 都是二分类器，对于多分类问题，可采用：

- 一对一 (One-vs-One, OVO)：为每对类别训练一个二分类器。对于 K 类问题需训练 $K(K-1)/2$ 个模型，预测时通过投票决定类别。
- 一对其余 (One-vs-All, OVA)：为每个类别训练一个二分类器，将该类视为正类，其余视为负类。只需训练 K 个模型，选择决策函数值最大者。

图 9: 多分类问题

量子 SVM 的研究方向有量子特征映射, 量子核函数法, 变分量子 SVM, 量子 SVM 求解器。

1. 对于量子特征映射，其研究的是怎么将经典数据编码到量子态的过程。最简单的编码方式是基态编码，将经典数据直接编码为量子比特的基态。对于 n 位二进制数据 $x=(x_1,x_2,...,x_n)$，其中 $x_i\in\{-1,1\}$，其对应量子态为 $|x\rangle=|x_1x_2..x_n\rangle$。

还有一种编码方式是振幅编码。对于 D 维归一化向量 $||x||=1$，其编码形式为：$|\phi(x)\rangle=\sum_{i=1}^{D}x_i|i\rangle$。这里的 $|0\rangle$ 表示 $|\underbrace{00..0}_{log_2D}\rangle$。说明我们的量子态 $|\phi(x)\rangle$ 只需 log_2D 个 qubit 承载信息，只不过会损失模长特征。

2. 对于量子核函数，其核心思想是将经典数据点编码至高维量子态空间，通过测量量子态之间的重叠度来定义核函数。对于已经映射为量子态的输入数据 $|\phi(x)\rangle=U(x)|0\rangle$, 量子核函数定义为: $K_Q(x,z)=|\langle\phi(x)|\phi(z)\rangle|^2=|\langle0|U^T(z)U(x)|0\rangle|^2$。量化了数据点 x 与 z 在量子特征空间中的相似程度。

量子核函数具有以下性质：正定性（$U^T(z)U(x)$ 是正定的）；对称性 $(K_Q(x,z)=K_Q(z,x))$；有界性 $(1\geq K_Q(x,z)\geq0)$；值得注意的是，量子核函数并不像经典核函数一样能化简出解析形式，需要在量子线路上通过 SWAP Test（就是先前介绍的 controlled-SWAP 电路）进行测量估计，这也是量子核方法潜在优势的来源。其工作流程如下：数据预处理（标准化处理），量子特征映射（映射为量子态），量子核计算（计算核函数值），经典 SVM 训练（放入经典优化器计算），预测。

3. 对于变分量子 SVM，该方法通过量子电路直接构建非线性决策边界

(端到端)，其判别函数：

$$f(x) = \langle 0|U^T(x,\theta)MU(x,\theta)|0\rangle$$

其中 U 是量子电路，M 是测量算符，通过优化参数 θ，系统可以学习到能够有效区分不同类别的最优决策边界。

变分量子 SVM 基本架构包括：数据编码层（编码为量子态），变分层(学习最优边界)，测量层（提取判别信息)。

10.3 量子 CNN

10.3.1 经典 CNN

核心组件有：卷积层（提取局部特征)、池化层（减少特征图大小)、全连接层（输出不同维度结果)、激活函数（增加非线性)。

训练时，在前向传播时常用 MSE 作为误差函数，再进行更新迭代：

$$y = F_\theta(I)$$

$$L(\theta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$\theta_{new} = \theta_{old} - \eta\nabla_\theta L(\theta)$$

10.3.2 量子 CNN(QCNN)

量子 CNN 包括量子态制备、量子卷积层、量子池化层、量子全连接层等几个组件，大致流程如图所示：

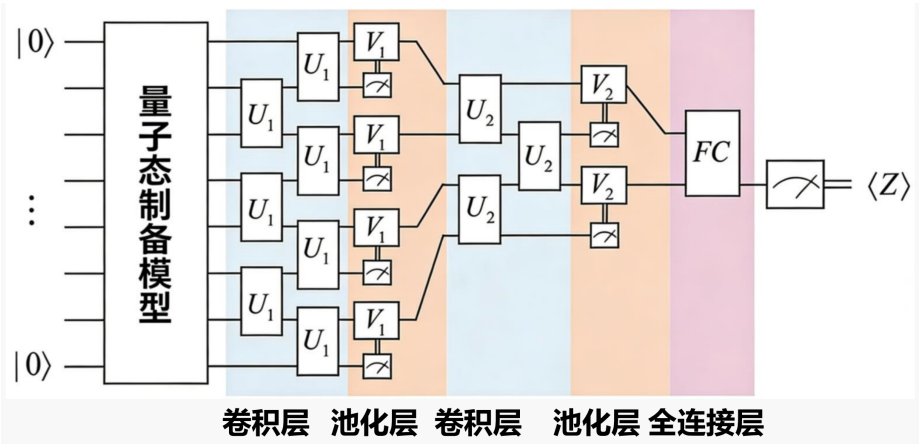

图 10: QCNN 结构

量子态制备常用有基态编码和振幅编码（更常用），先前介绍过。

量子卷积层仅作用于输入量子比特中相邻的两个量子比特，以平移不变的且共享参数的形式对输入量子比特两两进行处理。所以它是一个双量子比特门，可以分解为基础的单量子比特门，一种分解如下：

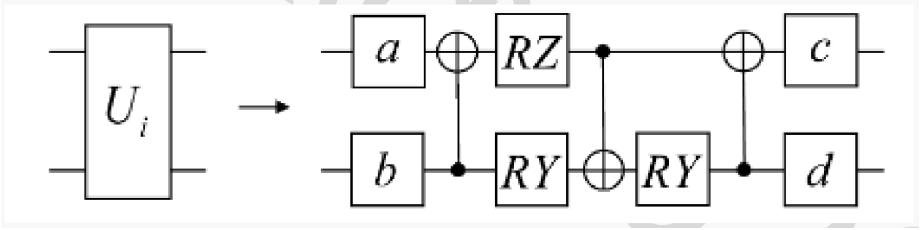

图 11: 双量子比特门分解

该方法仅使用受控非门（CNOT）、Y 轴旋转门（RY）、Z 轴旋转门（RZ）三类基础量子门，量子门 a,b,c,d 为单量子比特门，可进一步分解为 RZ 门、RY 门、RY 门的级联形式。(a,b 先进行了局部操作，中间基础门引入了量子间的纠缠，最后 c,d 再一次进行了局部操作）

量子池化层会测量一部分量子比特，被测量的量子比特（作为控制比特）用来判断是否对剩下的量子比特施加幺正旋转（一种旋转变换），其结构如图所示 (a,b 为单量子比特门):

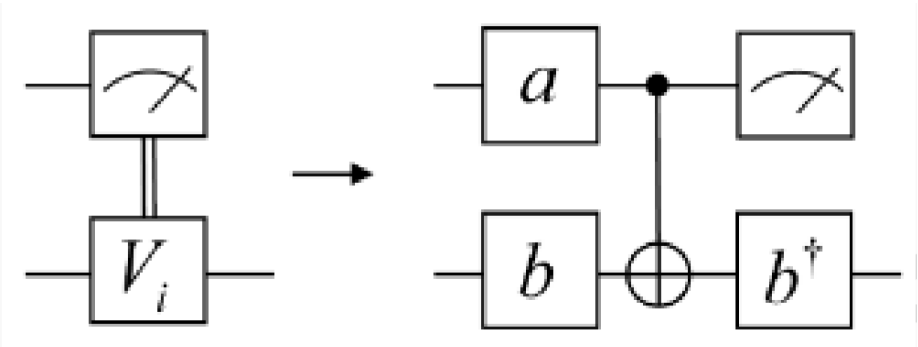

图 12: 量子池化层

量子全连接层对剩余的这 k 个量子比特施加一个幺正变换 $|\psi\rangle = F|\psi^{(L-1)}\rangle$，其中 $|\psi^{(L-1)}\rangle$ 是最后一层池化后剩余比特的量子态。

对二分类问题的分类规则为选取单个量子比特进行测量，并将其测量期望值直接作为模型输出

$$f(\theta, x_{in}) = \langle Z \rangle$$

当测量得到的 $\langle Z \rangle \geq 0$ 归为一类；当测量得到的 $0 \geq \langle Z \rangle$ 归为另一类。

对量子电路梯度的求解，假设损失函数 l_θ 是观测期望值集合 $\{\langle O_k \rangle_\theta\}_{k=1}^K$ 的函数，则只需求 $\frac{\partial \langle O_k \rangle_\theta}{\partial \theta_j}$ 即可。我们有计算公式:

$$\frac{\partial \langle O_k \rangle_\theta}{\partial \theta_j} = \frac{\langle O_k \rangle_{\theta + \frac{\pi}{2} e_j} - \langle O_k \rangle_{\theta - \frac{\pi}{2} e_j}}{2}$$

其中 $\langle O_k \rangle_{\theta \pm \frac{\pi}{2} e_j}$，表示将原量子电路的第 j 个参数偏移 $\pm \frac{\pi}{2}$ 后，测得的对应观测期望值。该梯度求解方法计算结果精准，且易于在含噪中等规模量子处理器（NISQ）上实现。

11 量子隧穿效应

量子隧穿效应指的是量子力学中粒子能够穿过经典力学认为无法逾越的势垒的现象，其出现的本质原因是波粒二象性。典型例子有 α 衰变、扫描隧道显微镜（STM）成像。

隧穿概率公式

隧穿概率依赖于势垒的宽度L，高度U，和粒子的能量E，通常随势垒变宽或变高指数下降：

$$T \approx \mathrm{e}^{-2\int_a^b \sqrt{\frac{2m}{\hbar^2}(V(x)-E)}\mathrm{d}x}$$

其中，T：透射系数（隧穿概率），m：粒子质量，$\hbar$：约化普朗克常数，$V(x)$：势垒位能。

对于宽高势垒，近似公式为：

$$T\left(L,E\right) \approx 16\frac{E}{U_0}\left(1-\frac{E}{U_0}\right)\mathrm{e}^{-2\beta L}$$

其中，$\beta = \sqrt{2m(U_0 - E)/\hbar}$为衰减常数，反映势垒相关的衰减特性，$T(L,E)$为与势垒宽度$L$和粒子能量$E$相关的透射系数（隧穿概率）

图 13: 量子隧穿效应公式

12 伊辛模型与霍普菲尔德网络

12.1 伊辛模型

伊辛模型是统计物理学中描述铁磁物质相变的经典模型，由威廉·楞次和恩斯特·伊辛提出。该模型通过假设物质由规则排列的自旋（小磁针）构成，每个自旋只有向上或向下两种状态，且仅与最近邻自旋发生相互作用，来模拟铁磁材料在温度变化下的磁性转变，成为研究相变、临界现象和多领域合作行为的典范工具。

伊辛模型的能量函数（哈密顿量）定义为

$$E_{\{s_i\}} = -J\sum_{\langle i,j\rangle} s_i s_j - H\sum_i s_i,$$

其中 $s_i \in \{+1,-1\}$ 表示第 i 个格点处的自旋状态，对应小磁针的两种方向；J 为能量耦合常数，描述相邻小磁针自旋之间的能量关联；H 为外磁场强度，当自旋方向与外磁场方向一致时，系统能量降低。符号 $\langle i,j\rangle$ 表示对最近邻自旋对求和。

在温度变化下的磁性转变表现为两种行为：

（1）当系统温度处于低温时，系统总倾向于使能量函数取最低值的状态演化，即系统中每个小磁针的自旋方向倾向于一致，并与外磁场方向一致。

（2）当系统温度处于高温时，粒子的热运动主导，热运动引起小磁针的状态随机反转，自旋呈现紊乱状态。

能量函数计算示例：考虑 3 个小磁针组成的一维伊辛模型，设 $s_1 = +1$，$s_2 = -1$，$s_3 = -1$，外磁场为 H。则该组态的能量为

$$E_{\{+1,-1,-1\}} = -J\big[1 \times (-1) + (-1) \times (-1)\big] - H(1 - 1 - 1) = J + H.$$

全部 $2^3 = 8$ 种组态的能量分别为：

$$\begin{aligned}
&E_{\{+1,+1,+1\}} = -3J + 3H, && E_{\{-1,-1,-1\}} = -3J - 3H,\\
&E_{\{+1,+1,-1\}} = J - H, && E_{\{+1,-1,+1\}} = J - H,\\
&E_{\{+1,-1,-1\}} = J + H, && E_{\{-1,+1,+1\}} = J + H,\\
&E_{\{-1,+1,-1\}} = J + H, && E_{\{-1,-1,+1\}} = J - H.
\end{aligned}$$

由上述计算可知，当小磁针方向一致且方向与磁场方向一致时，能量函数取到最小值。

12.2 霍普菲尔德网络

伊辛模型为霍普菲尔德网络提供理论启发，借自旋相互作用构建神经网络。霍普菲尔德网络是约翰 · 霍普菲尔德于 1982 年提出的递归神经网络，由全连接的二值神经元（状态为 ± 1）构成，通过能量函数实现联想记忆。其结构直接借鉴伊辛模型，神经元对应自旋，权重对应相互作用。

霍普菲尔德网络的核心功能是实现**联想记忆**——通过不完整或含噪输入恢复完整的存储模式。联想记忆模仿人类记忆的联想特性，当输入部分信息时，网络通过动力学演化恢复完整模式。

联想记忆的数学基础是霍普菲尔德能量函数，其形式直接借鉴统计物理中的伊辛模型：

$$E = -\frac{1}{2} \sum_{i,j \,\&\, i \neq j} w_{ij} x_i x_j - \sum_i \theta_i x_i,$$

其中 w_{ij} 为对称连接权重，θ_i 为神经元阈值（偏置），$x_i \in \{-1, 1\}$ 表示神经元状态。类比伊辛模型，神经元状态对应磁针自旋方向（向上或向下），权重对应自旋间耦合作用：$w_{ij} > 0$ 时 x_i 与 x_j 倾向同号，反之倾向反号。输

入残缺或含噪模式时，网络通过迭代更新神经元状态，演化至能量极小的稳定态（吸引子），输出完整模式。

霍普菲尔德网络训练（以黑白图像为例）：

（1）图像预处理：将待存储的 d 个像素，每个像素视为一个神经元，第 i 像素值为黑色则对应神经元状态 $x_i = -1$，白色对应 $x_i = +1$。

（2）权重计算：假设要存储 p 张图像，且神经元偏置 $\theta_i = 0$，则权重

$$w_{ij} = \frac{1}{d} \sum_{k=1}^{p} x_i^{(k)} x_j^{(k)} \quad (i \neq j),$$

此即为赫布学习规则。

（3）存储容量限制：为保证可靠恢复，存储的模式数量 p 需满足 $p < 0.138d$。若超出此限制，网络会出现大量干扰记忆的虚假稳定态。

赫布学习规则：定义输入 x_i 为第 i 个神经元节点的值，w_{ij} 为第 i 个和第 j 个节点之间的权值，则每个样本作为节点初始化的权值定义为

$$w_{ij} = x_i x_j,$$

p 个样本的权值经过 p 次更新为

$$w_{ij} = \frac{1}{d} \sum_{k=1}^{p} x_i^{(k)} x_j^{(k)} \quad (i \neq j).$$

训练阶段将所有样本的信息以权值求和的形式存储起来，最终的权值存储的是每个样本的记忆，测试阶段则利用这些权值恢复记忆。

测试阶段：先用测试样本初始化节点，利用训练阶段存储的权值，循环随机选择一个节点 x_i，将节点值按下式更新：

$$x_i = \operatorname{sgn}\Big(\sum_{k=1}^{N} w_{ij} x_j \Big).$$

经过若干次迭代，则所有节点会收敛到一个合适的值。

霍普菲尔德网络恢复（以黑白图像为例）：

（1）初始化：将含噪或部分缺失的图像数据 x^0 作为网络的初始状态，输入到神经元中。

（2）迭代更新：通过网络动力学演化，逐步更新神经元状态，使整个系统的能量函数降低。每个神经元按如下规则更新：

$$x_i = +1, \quad \text{if } \sum_j w_{ij} x_j + \theta_i > 0,$$

$$x_i = -1, \quad \text{if } \sum_j w_{ij}x_j + \theta_i < 0.$$

霍普菲尔德网络生成新图像：若需生成新图像，可从随机初始状态出发，通过逐次反转神经元状态使能量函数减小，最终收敛到稳定状态。每次选择一个神经元 i，若反转其状态（$x_i \to -x_i$）能降低能量 E，则接受该变化，否则保持原状。这一过程本质上是求解能量函数最小值的问题，等价于寻找系统的平衡态：

$$\min_{\{x\}}\Big\{E = -\frac{1}{2}\sum_{i,j\ \&\ i\neq j} w_{ij}x_i x_j - \sum_i \theta_i x_i\Big\}.$$

伊辛模型为分析和理解霍普菲尔德网络的性质提供了重要的理论工具。霍普菲尔德网络通过神经元之间的连接权重和能量函数实现了联想记忆功能。两者的结合使得人们能够从统计物理的角度深入研究神经网络的行为，为神经网络的发展和应用奠定了坚实的理论基础。

13 玻尔兹曼分布与玻尔兹曼机

13.1 玻尔兹曼分布

玻尔兹曼分布也称为玻尔兹曼-吉布斯分布，以奥地利物理学家路德维希 · 玻尔兹曼的名字命名，是统计力学和概率论中一个重要的概念，用于描述系统中粒子或状态在不同能量水平上的分布情况，揭示了统计力学中描述系统在热平衡状态下粒子在不同能级上概率分布的核心规律。

在统计力学中，玻尔兹曼分布描述粒子处于特定状态下的概率，是关于状态能量与系统温度的函数。假设系统中某个粒子处于状态 α 时的能量为 E_α，则粒子处于该状态的概率满足

$$p_\alpha = \frac{1}{Z}\exp\Big(-\frac{E_\alpha}{k_B T}\Big),$$

其中 E_α 表示状态 α 的能量；k_B 为玻尔兹曼常数，在机器学习领域中常用 $k_B = 1$ 来简化公式；T 表示系统的绝对温度，控制分布的平滑程度。

$\exp(-E_\alpha/k_B T)$ 称为**玻尔兹曼因子**，是未经过归一化的概率；Z 为归一化因子，称为**配分函数**，对系统所有状态求和以确保概率总和为 1：

$$Z = \sum_\alpha \exp\Big(-\frac{E_\alpha}{k_B T}\Big) \quad \text{(离散情况)},$$

或

$$Z = \int \exp\Big(-\frac{E_\alpha}{k_B T}\Big)\, d\alpha \quad \text{(连续情况)}.$$

玻尔兹曼分布的一个重要性质是两个状态的概率比（即玻尔兹曼因子）仅依赖于两个状态能量的差值：

$$\frac{p_i}{p_j} = \exp\Big(\frac{E_j - E_i}{k_B T}\Big).$$

13.2 玻尔兹曼机神经网络

玻尔兹曼机是一种基于玻尔兹曼分布的随机神经网络，由二值随机神经元构成的两层对称连接网络，其权值通过优化玻尔兹曼能量函数获得。玻尔兹曼机由杰弗里 · 辛顿等人于 1985 年提出，属于无监督学习模型，主要用于特征学习、概率建模和优化问题。

玻尔兹曼机具有以下结构特征：所有变量之间相互连接，是无向图模型，每个变量的状态都以一定概率受到其他变量的影响；每个随机变量都是二值的，取 1 代表被激活，取 0 代表未激活；包含可观测变量 V 和隐变量 H。

玻尔兹曼机分为**自联想型**和**异联想型**：自联想型玻尔兹曼机的可见节点既是输入节点又是输出节点，数据在可见节点和隐节点之间循环流动，以实现对输入模式的记忆和恢复；异联想型玻尔兹曼机用于实现异联想记忆，可见节点分为输入节点和输出节点两部分，输入节点接收输入模式，输出节点输出相关的不同模式。

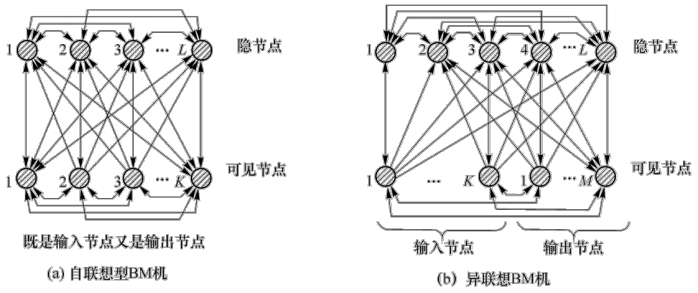

图 14: 两种玻尔兹曼机

玻尔兹曼机的能量函数定义了模型中各个节点状态组合的能量值，是

理解和分析模型的基础。对于具有 n 个节点的玻尔兹曼机，能量函数为

$$E(X) = -\sum_{i<j} \omega_{ij} x_i x_j - \sum_i b_i x_i,$$

其中 $X = (x_1, x_2, \ldots, x_n)$ 为节点状态向量，x_i 表示第 i 个节点的状态（取值通常为 0 或 1，也可以是 -1 和 1）；ω_{ij} 是节点 i 和节点 j 之间的连接权重；b_i 是节点 i 的偏置。第一项 $-\sum_{i<j} \omega_{ij} x_i x_j$ 表示节点之间相互作用的能量，第二项 $-\sum_i b_i x_i$ 表示节点的自身能量。连接权重决定节点之间相互作用的强度和方向，偏置影响单个节点的激活程度。

基于能量函数，玻尔兹曼机中节点状态的概率分布服从玻尔兹曼分布（也称为吉布斯分布）。在温度 T 下（玻尔兹曼常数 $k_B = 1$），状态 X 出现的概率为

$$P(X) = \frac{1}{Z} \exp\Big(-\frac{E(X)}{T}\Big).$$

从该公式可以看出，能量越低的状态，其出现的概率越高。温度控制概率分布的形状：当 T 较高时，不同状态的概率差异较小，系统更加随机；当 T 较低时，系统更倾向于处于低能量状态。

玻尔兹曼机的学习过程是调整连接权重和偏置，使模型能够最好地拟合训练数据。常用的学习算法是**对比散度**算法，权重更新公式为

$$\Delta\omega_{ij} = \varepsilon\big(\langle x_i x_j \rangle^{+} - \langle x_i x_j \rangle^{-}\big),$$

$$\Delta b_i = \varepsilon\big(\langle x_i \rangle^{+} - \langle x_i \rangle^{-}\big),$$

其中 ε 是学习率；$\langle x_i x_j \rangle^{+}$ 和 $\langle x_i \rangle^{+}$ 为正相阶段计算的期望；$\langle x_i x_j \rangle^{-}$ 和 $\langle x_i \rangle^{-}$ 为负相阶段计算的期望。

13.3 受限玻尔兹曼机神经网络

玻尔兹曼机具有强大的无监督学习能力，但训练时间非常长，且难以确切计算其所表示的分布，得到服从该分布的随机样本也很困难。为克服这一问题，杰弗里 · 辛顿和斯莫伦斯基等人引入了**受限玻尔兹曼机**。受限玻尔兹曼机具有一个可见层和一个隐层，层内无连接，不同层的节点之间全连接，而同层节点之间不连接。

受限玻尔兹曼机最早由保罗 · 斯莫伦斯基在 1986 年的和谐模型中提出，后由辛顿在 2000 年代推广和应用。它可以用于降维、分类、回归、协同过滤、特征学习以及主题建模。

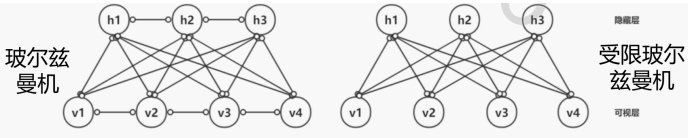

图 15: 玻尔兹曼机与受限玻尔兹曼机

设一个受限玻尔兹曼机有 n 个可见单元和 m 个隐单元，用向量 V 和 H 分别表示可见单元和隐单元的状态。其中 v_i 表示第 i 个可见单元的状态，h_j 表示第 j 个隐单元的状态。对于一组给定的状态 (V, H)，受限玻尔兹曼机的能量函数定义为

$$E(V, H|\theta) = -\sum_{i=1}^{n} a_i v_i - \sum_{j=1}^{m} b_j h_j - \sum_{i=1}^{n}\sum_{j=1}^{m} v_i \omega_{ij} h_j,$$

其中 $\theta = \{\omega, a, b\}$ 表示受限玻尔兹曼机的参数，ω_{ij} 是节点 i 和节点 j 之间的连接权重，a_i 表示可见单元神经元的偏置，b_j 表示隐层神经元的偏置。

基于能量函数，状态向量组 (V, H) 的联合概率密度分布为

$$P(V, H|\theta) = \frac{1}{Z_\theta} \exp\big(-E(V, H|\theta)\big),$$

其中 $Z_\theta = \sum_{V,H} \exp(-E(V, H|\theta))$ 为归一化因子（配分函数）。

由于受限玻尔兹曼机的特殊结构，当给定可见单元的状态时，各隐单元的激活状态之间条件独立；当给定隐单元的状态时，各可见单元的激活状态之间也条件独立。具体地：

$$P(h_j = 1|V, \theta) = \sigma\Big(b_j + \sum_i v_i \omega_{ij}\Big),$$

$$P(v_i = 1|H, \theta) = \sigma\Big(a_i + \sum_j \omega_{ij} h_j\Big),$$

其中 $\sigma(\cdot)$ 为逻辑 sigmoid 函数。

受限玻尔兹曼机的学习任务为求出参数 θ 的值。使用训练数据训练 θ，参数 θ 可以通过最大化受限玻尔兹曼机在训练集（假设包含 T 个样本）上的对数似然函数学习得到：

$$\theta^* = \arg\max_\theta L(\theta) = \arg\max_\theta \sum_{t=1}^{T} \log P(V^{(t)}|\theta).$$

根据极大似然方法构建损失函数 $L(\theta)$，计算可得

$$L(\theta) = \sum_{t=1}^{T} \log \sum_{H} P(V^{(t)}, H|\theta) - \log \sum_{V,H} \exp\bigl(-E(V, H|\theta)\bigr).$$

为获得最优参数 θ^*，使用梯度下降法不断更新求 $L(\theta)$ 的最大值，使得模型逐渐拟合训练数据的分布。

14 固体退火与经典模拟退火算法

14.1 固体退火

固体退火的核心概念是模拟固体从高温冷却到低温的过程，消除材料内部残余应力，减少晶格缺陷（位错、空位），形成均匀晶粒。

退火典型工艺包含三个阶段：

（1）**加热阶段**：升温至再结晶温度以上（熔点的 30% 至 50%），原子动能 $E_k = \frac{3}{2}k_BT$ 克服势垒，晶格解体，进入高能无序态。

（2）**恒温阶段**：高温保持足够时间，原子扩散重排，消除局部应力，形成新晶核。系统处于热力学平衡态，微观状态随机热涨落。

（3）**冷却阶段**：缓慢降温（每小时数摄氏度），原子调整位置形成稳定晶体结构。关键之处在于缓慢冷却使系统逼近热力学基态；过快冷却（淬火）会导致非晶态或亚稳态。

物理基础——统计物理与自由能：

宏观系统以一定概率分布在微观状态中（正则系综）。能量为 E_i 的状态出现概率为

$$P(E_i) = \frac{1}{Z} \exp\Bigl(-\frac{E_i}{k_BT}\Bigr).$$

温度影响表现为：高温时状态概率相近，系统高熵、无序；低温时低能状态概率高，系统趋于有序基态。

系统追求**亥姆霍兹自由能**最小化：

$$F = E - TS,$$

其中 E 为内能，S 为熵，T 为温度。高温时 $-TS$ 熵项占主导，系统探索更多构型；低温时能量项 E 占主导，系统聚焦于低能区域。缓慢冷却使系统逼近基态，过快冷却会陷入局部极小（亚稳态）。

物理基础——梅特罗波利斯准则与微观动力学：

1953 年梅特罗波利斯提出的蒙特卡洛方法用于模拟热平衡。若新状态能量差 $\Delta E = E_j - E_i < 0$（能量降低），则无条件接受；若 $\Delta E \geq 0$（能量升高），则以概率

$$P = \exp\Big(-\frac{\Delta E}{k_B T}\Big)$$

接受。该准则的意义在于允许系统暂时接受高能量状态，从而跳出局部能量极小值，增加找到全局极小值的可能性。

14.2 经典模拟退火算法

算法核心逻辑：

A. 高温出发：计算初始解。

B. 邻域扰动：产生新解（模拟粒子无序运动）。

C. 能量比较：比较新旧解的目标函数值。

D. 概率转化：能量降低则直接转；能量升高则按 Metropolis 概率转化。

E. 降温迭代：达到热平衡后降温，在下一温度继续迭代。

F. 低温收敛：最终达到低温稳定状态，输出结果。

算法流程：

(1)定义决策变量：系统状态为 N 维空间中的向量 $X = (x_1, x_2, \ldots, x_i, \ldots, x_N)$，初始时随机生成初始状态 x_0。

（2）生成新状态：在当前解邻域内生成候选解，常用随机行走方法：

$$x_i' = x_i + \mathrm{Rnd}[-\varepsilon, \varepsilon], \quad i = 1, 2, \ldots, N,$$

其中 $\mathrm{Rnd}[a, b]$ 表示 $[a, b]$ 范围内的均匀分布随机值。新解为 $\mathbf{x}_{\mathrm{new}} = (x_1', x_2', \ldots, x_i', \ldots, x_N')$。

（3）接受函数（基于 Metropolis 准则）：计算能量差 $\Delta F = F(X_{\mathrm{new}}) - F(X)$，接受概率为

$$P(X, X_{\mathrm{new}}) = \begin{cases} 1, & \text{if } F(X_{\mathrm{new}}) < F(X), \\ \exp(-\Delta F/\lambda), & \text{otherwise.} \end{cases}$$

决策规则：若 $P(X, X_{\mathrm{new}}) \geq \mathrm{Rnd}[0, 1]$，则接受新解；否则保持当前解。参数 λ（温度）较大时选择压力小（易接受劣解），应随算法进行逐渐减小。

（4）热平衡：在每一温度下，生成并测试足够多的新状态（通常为 M 个），目标是使解的分布在该温度下接近稳态（玻尔兹曼分布）。

（5）温度下降：目标是 $\lim_{x\to\infty}\lambda_x = 0$（$x > 0$）。常见方法包括：

线性法：$\lambda_n = \lambda_0 - \alpha_x$（$\alpha = \lambda_0/T$，需指定最大迭代次数），其中 λ_0 为初始温度，λ_k 为第 k 次迭代时的温度，T 为总迭代次数，α 为冷却因子。

几何法（推荐）：$\lambda_x = \lambda_0 \cdot \alpha^x$（无需指定最大迭代次数）。

（6）终止条件：达到预设的最大迭代次数；达到最低温度阈值；运行时间达到限制；目标函数的增量改进连续多次低于阈值。

14.3 模拟退火算法的应用

神经网络训练是高维、非线性、非凸的优化问题。传统梯度下降法（如随机梯度下降、Adam）依赖局部梯度，容易陷入局部极小点或鞍点，对初始权重敏感。

SA-BP 算法结合模拟退火的全局搜索能力和反向传播的局部收敛速度：

SA 阶段（全局探索）：利用 Metropolis 准则广泛搜索解空间，定位潜在的全局最优区域，不依赖梯度，避开局部极小。

BP 阶段（局部开发）：基于梯度信息，在 SA 找到的最优区域内进行精细化搜索，快速收敛。

效果：既保证解的质量（避免局部最优），又提高收敛效率。

15 费米-狄拉克分布与 FiDEL

15.1 费米-狄拉克分布

费米-狄拉克分布是量子统计学中描述全同且独立的费米子在热平衡状态下占据量子态概率的统计规律，由恩里科 · 费米和保罗 · 狄拉克于 1926 年分别独立推导而出。

分布函数描述了全同费米子在热平衡状态下占据量子态的概率，其数学表达式为

$$f(E) = \frac{1}{\exp\big((E-\mu)/k_BT\big) + 1},$$

其中 E 为量子态能量；μ 为化学势（在低温下近似等于费米能 E_F）；k_B 为玻尔兹曼常数；T 为温度。

费米-狄拉克分布的核心源于**泡利不相容原理**：每个量子态最多只能被一个费米子（自旋为半整数的粒子）占据。这一限制决定了分布函数 $f(E)$

的取值范围为 $[0,1]$，表示能量为 E 的量子态被占据的概率。$f(E)=1$ 表示该态被完全占据；$f(E)=0$ 表示该态未被占据。

在绝对零度（$T=0\,\mathrm{K}$）时，系统处于能量最低状态，所有能量低于费米能 μ 的量子态都被电子填满，所有高于 μ 的态则完全空着，此时分布函数呈现为理想的阶跃函数：

$$f(E)=\begin{cases}1, & E<\mu,\\ 0, & E>\mu.\end{cases}$$

这种被电子填满的海洋称为**费米海**。

当温度升高到 $T>0\,\mathrm{K}$ 时，热能使部分电子获得足够能量，从略低于 E_F 的态跃迁到略高于 E_F 的空态上。在费米能附近（约 $\sim k_BT$ 的范围内），分布不再陡峭，而是形成平滑过渡的曲线，这种现象称为**热激发拖尾**。

在电子总数固定的系统中，μ 由总电子数守恒确定，满足

$$N=\int g(E)f(E)\,dE,$$

其中 $g(E)$ 为态密度函数。

15.2 FiDEL 算法原理

FiDEL（基于费米-狄拉克的集成学习）是一种将费米-狄拉克分布应用于二分类器函数的集成学习算法。其核心思想是利用费米-狄拉克分布的阈值敏感性和概率饱和性，动态调整分类器权重，以平衡分类边界样本的判别力与噪声鲁棒性。

核心映射关系：

（1）**能量项** $E_i(x)$：基分类器 h_i 对样本 x 的预测置信度 $s_i(x)$ 被映射为能量

$$E_i(x)=-s_i(x),$$

使得高置信度（低能量）的基分类器获得更高权重。

（2）**化学势** $\mu(x)$：反映集成模型对样本 x 的整体置信水平。初始值为基分类器置信度的均值 $\mu_0(x)=(1/M)\sum_i s_i(x)$，后续通过优化损失函数动态调整。

（3）**温度参数** T：控制权重分布的陡峭程度。高温度（$T\gg 1$）下权重分布平缓，适用于噪声较多的数据集；低温度（$T\ll 1$）下权重分布尖锐，适用于置信度明确的清洁数据。

权重分配函数：基分类器 h_i 对样本 x 的权重由费米-狄拉克分布定义

$$w_i(x) = \frac{1}{\exp\big((E_i - \mu)/k_B T\big) + 1}.$$

自适应阈值优化：化学势 $\mu(x)$ 通过最小化交叉熵损失函数在线调整。损失函数为

$$L(\mu) = \sum_{j=1}^{N} \log\Big(1 + \exp\big(-y_j \sum_{i=1}^{M} w_i(x) h_i(x_j)\big)\Big),$$

采用随机梯度下降更新规则

$$\mu_j \leftarrow \mu_j - \eta \frac{\partial L}{\partial \mu_j(x_i)},$$

其中 η 为学习率。

集成预测：对于新测试样本 x，通过加权聚合基分类器输出得到最终预测

$$H(x) = \operatorname{sign}\Big(\sum_{i=1}^{M} w_i(x) h_i(x)\Big).$$

16 玻色-爱因斯坦分布及人工智能应用

16.1 玻色-爱因斯坦分布

玻色-爱因斯坦分布是量子统计力学中描述玻色子热平衡态占据行为的核心理论工具。该分布由玻色与爱因斯坦于 1924 年提出，其物理基础源于全同玻色子的波函数对称性要求及巨正则系综的统计理论。作为量子统计力学的核心分布之一，其独特性在于描述全同玻色子的统计占据规律。

推导基础：系综理论。系综理论是统计物理的基石，由吉布斯于 20 世纪初系统提出，旨在通过统计系综（即大量虚拟系统的集合）描述宏观系统的热力学性质。其核心逻辑是：单个系统的微观状态随时间演化，但宏观可观测量可通过系综平均获得。系综主要分为三类：

（1）**微正则系综**（NVE 系综）：固定粒子数 N、体积 V、能量 E 的孤立平衡体系，与外界无能量和粒子交换。

（2）**正则系综**（NVT 系综）：固定粒子数 N、体积 V、温度 T，体系与恒温大热库接触。

(3) **巨正则系综**（μVT 系综）：固定化学势 μ、温度 T、体积 V，为与大热库大粒子源接触的开放系统。

玻色-爱因斯坦分布必须从巨正则系综推导。核心原因在于：玻色子作为全同粒子可大量占据同一量子态，固定总粒子数的正则系综会导致配分函数难以化简；而巨正则系综允许粒子数涨落、只固定化学势，天然适配玻色体系的粒子流动特性。其最大优势在于可将多能级配分函数分解为单能级的乘积，使单能级占据数的统计可独立求解。

基于巨正则系综的推导过程：

假设系统有 N 个粒子，L 个单粒子能级 $\varepsilon_1 \le \varepsilon_2 \le \varepsilon_3 \le \ldots \le \varepsilon_L$。一个微观状态可表示为 $(n_1, n_2, \ldots, n_L)$，满足约束 $\sum_i n_i = N$，总能量 $E = \sum_i n_i \varepsilon_i$。

在巨正则系综中，出现状态 $(n_1, n_2, \ldots, n_L)$ 的概率为

$$\varphi(n_1, n_2, \ldots, n_L) = \frac{\exp\bigl(-\beta \sum_i n_i(\varepsilon_i - \mu)\bigr)}{Z},$$

其中 $\beta = 1/(k_B T)$，Z 为配分函数：

$$Z = \sum_{n_1, n_2, \ldots, n_L} \exp\bigl(-\beta \sum_i n_i(\varepsilon_i - \mu)\bigr) = \prod_i \frac{1}{1 - e^{-\beta(\varepsilon_i - \mu)}}.$$

第 i 个能级粒子数的期望为

$$\bar{n}_i = \sum_{n_1, n_2, \ldots, n_L} \varphi(n_1, n_2, \ldots, n_L)\, n_i.$$

将概率表达式代入，利用

$$\bar{n}_i = \frac{1}{\beta} \frac{\partial \ln Z}{\partial \varepsilon_i},$$

以及 $\ln Z = -\sum_i \ln(1 - e^{-\beta(\varepsilon_i - \mu)})$，求导可得

$$\bar{n}_i = \frac{1}{e^{\beta(\varepsilon_i - \mu)} - 1}.$$

此即为**玻色-爱因斯坦分布**。

16.2 量子行走

玻色-爱因斯坦分布主要应用于量子算法领域，为设计高效量子算法、突破经典计算瓶颈提供核心工具。本节着重介绍量子行走。

随机行走。随机行走是描述对象在空间中随机移动的数学模型。一维离散模型中，粒子从原点出发，每步以 50% 概率向左或向右移动，步长为 1。n 步之后粒子所处位置 x 的概率分布遵循二项分布。核心结论是位移平方的期望方根与时间方根成正比。

量子化马尔可夫过程。经典马尔可夫过程的状态转移由概率矩阵描述。将其量子化，状态用狄拉克符号（量子比特）表示。对于一个量子比特可表示为

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{pmatrix}\alpha\\ \beta\end{pmatrix},$$

其中 $|0\rangle = [1,0]^\top$，$|1\rangle = [0,1]^\top$。

当转移概率 $p_1 = p_2 = 0.5$ 时，对应的幺正矩阵为阿达马门

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}.$$

其对基态的作用为

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \quad H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle).$$

为使量子演化符合经典马尔可夫过程的行为，引入一组特别的基

$$|H_\uparrow\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle), \quad |H_\downarrow\rangle = \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle).$$

以它们作为初态，无论演化多少次，都能给出符合经典马尔可夫过程的结果。

量子行走具体构建流程：

（1）定义系统构成与初态：粒子位置由波函数 $|\psi_p\rangle$ 描述，存在于所有可能位置的叠加态

$$|\psi_p\rangle = \sum_x c_x|x\rangle,$$

其中 $|x\rangle$ 表示粒子在位置 x 的基态，c_x 是对应的复振幅，满足归一化关系。初始状态用 ψ_Q 描述马尔可夫状态变量的量子比特，初态表示为 $\psi_0 = \psi_{Q_0} \otimes \psi_{P_0}$，其中 $\otimes$ 表示张量积。

（2）定义核心演化算符：

翻转算符 C 对量子比特进行翻转，模拟马尔可夫过程的状态转移：$C = H \otimes I_x$，其中 I_x 表示对坐标波函数的单位矩阵。

粒子移动算符 S 将量子比特的 $|0\rangle$ 分量对应的坐标分量右移，将 $|1\rangle$ 分量对应的坐标分量左移：

$$S = |0\rangle\langle 0| \otimes \sum_x |x+1\rangle\langle x+1| + |1\rangle\langle 1| \otimes \sum_x |x-1\rangle\langle x|.$$

（3）形成一次完整演化：分别用 C 和 S 对初态进行作用，一次完整的量子行走步为

$$|\psi_{n+1}\rangle = S \cdot C|\psi_n\rangle.$$

16.3 量子行走神经网络

基于图的量子行走建模：给定一个无向图 $G = (V, E)$，定义由基向量 $\{e_v^p, v \in V\}$ 张成的向量空间 H_p 描述量子游走者在图上的位置，以及由基向量 $\{e_i^c, i \in \{1, \ldots, d\}\}$ 张成的向量空间 H_c，其中 i 对应于入射到某个顶点的边，d 是图的最大度。二者的张量积 $H_w = H_p \otimes H_c$，其基态形式为 $e_v^p \otimes e_i^c$，表示游走者位于节点 v 且沿第 i 条入射边的自旋状态。

行走者的位置为位置基态的线性叠加 $\sum_{v\in V} \alpha_v e_v^p$，其中 $\sum_v |\alpha_v|^2 = 1$，$|\alpha_v|^2$ 为测量到游走者在节点 v 的概率。

初始叠加态为 $\Phi^{(0)}$，经过 t 步演化后的状态为 $\Phi^{(t)} = U^t \Phi^{(0)}$，其中 U 为单步演化算符。量子行走的演化行为由初始叠加态 $\Phi^{(0)}$ 和硬币算子 C 共同决定，这也是量子行走神经网络可学习参数的来源。

量子行走神经网络架构：量子行走神经网络通过学习行走者的硬币算子 C 和初始叠加态来学习图上的量子随机行走，架构分为量子行走层和量子扩散层。论文中设置了两种硬币算子：空间硬币算子（每个节点一个独立硬币算子）$C_s \in \mathbb{R}^{N\times d\times d}$ 和时间硬币算子（每个时间步一个共享硬币算子）$C_t \in \mathbb{R}^{T\times d\times d}$。给定初始叠加态 $\Phi^{(0)} \in \mathbb{R}^{W\times N\times d}$ 以及移位算子 $S \in \mathbb{Z}_2^{N\times d\times N\times d}$，量子行走神经网络接收节点特征矩阵 $X \in \mathbb{R}^{N\times F}$ 并输出扩散后的特征 $Y \in \mathbb{R}^{N\times F}$。

量子行走层实现量子游走的 T 步演化，硬币算子的迭代公式分别为

$$\Phi_{\cdot i\cdot}^{(t)} \leftarrow \Phi_{\cdot i\cdot}^{(t-1)} \cdot C_{s_i} \quad \text{（空间硬币，} i \text{ 为节点索引）},$$

$$\Phi_{\cdot i\cdot}^{(t)} \leftarrow \Phi_{\cdot i\cdot}^{(t-1)} \cdot C_t \quad \text{（时间硬币，} t \text{ 为步数）}.$$

移位算子的迭代公式为 $\Phi^{(t)} \leftarrow \Phi_{\text{coin}}^{(t)} : S$，其中 $:$ 表示在两个维度上做内积。

量子扩散层将量子行走的最终叠加态 $\Phi^{(T)}$ 转化为经典扩散矩阵 $P \in \mathbb{R}^{N\times N}$，对最终叠加态的自旋状态取模平方求和：

$$P \leftarrow \sum_{k} \Phi^{(T)}_{\cdot\cdot k} \odot \Phi^{(T)}_{\cdot\cdot k},$$

其中 $\odot$ 为按元素乘法，P_{uv} 表示游走者从节点 u 到 v 的转移概率。

最后进行特征扩散与输出：

$$Y \leftarrow h(PX + b),$$

其中 b 为偏置项，h 为激活函数。该公式与经典图神经网络的扩散形式一致，但扩散矩阵 P 由量子游走学习得到。

16.4 量子蒙特卡洛方法

玻色-爱因斯坦分布精准刻画了全同玻色子在热平衡下的能态占据规律，其核心的玻色凝聚特性（大量粒子在临界温度下集中占据基态）为量子系统的最优态定位提供了关键的统计基准。但面对高维多玻色子系统，直接解析求解的复杂度呈指数级增长，经典数值方法难以胜任。量子蒙特卡洛方法借助高效高维采样能力，优化梯度计算、损失求解与参数优化，并可抑制量子测量噪声、提升训练稳定性。

前置知识——变分蒙特卡洛。变分蒙特卡洛基于量子力学的变分原理：对于任何试探波函数 $\psi_T(r)$，哈密顿算符的期望值

$$E_T = \frac{\int \psi_T^*(r)\hat{H}\psi_T(r)\,dr}{\int \psi_T^*(r)\psi_T(r)\,dr} \geq E_0$$

是真实基态能量 E_0 的上限。在变分蒙特卡洛中，目标是通过改变试探波函数 $\psi_T(r)$ 中的参数使 E_T 最小化。

将期望值 E_T 重写为蒙特卡洛友好的形式：

$$E_T = \int P(r)H_L(r)\,dr,$$

其中 $P(r) = |\psi_T(r)|^2$，$H_L(r) = \psi_T^{-1}(r)\hat{H}\psi_T(r)$ 为局部能量。根据分布 $P(r)$ 随机采样配置空间来近似积分

$$E_T \approx \frac{1}{N}\sum_{i=1}^{N} H_L(r_i).$$

量子变分蒙特卡洛：核心用于能量最小化问题，通过参数化量子态（量子神经网络输出态）近似目标量子态，最小化能量期望值。设目标哈密顿量为 $\hat{H}$，量子神经网络输出态为 $|\phi_\theta\rangle$，则能量期望值为

$$\langle H_\theta \rangle = \langle \phi_\theta | \hat{H} | \phi_\theta \rangle.$$

量子变分蒙特卡洛通过采样估计 $\langle H(\theta) \rangle$，并迭代优化量子神经网络参数 θ 以最小化该期望值。量子优势在于利用量子干涉效应抑制能量估计的方差，使优化更稳定。

17 朗之万方程与朗之万扩散模型

17.1 朗之万动力学方程

朗之万方程由法国物理学家保罗 · 朗之万于 1908 年提出，描述随机动力学系统，将确定性阻尼和随机涨落结合，连接微观随机过程与宏观统计规律。

朗之万方程的形式为

$$m\frac{dv}{dt} = -\gamma v + F(t),$$

其中 m 为粒子质量；v 为速度；$-\gamma v$ 为与速度成正比的阻尼力；$F(t)$ 为随机涨落力（模拟粒子与周围分子的无规则碰撞）。

通常假设 $F(t)$ 为白噪声，满足均值 $\langle F(t) \rangle = 0$ 与关联函数

$$\langle F(t)F(t') \rangle = 2k_B T\gamma\delta(t-t').$$

朗之万动力学采样是扩散模型中的核心采样过程。给定概率分布 $p(x)$，通过计算概率密度函数的梯度并加入高斯噪声，迭代采样出高概率样本：

$$x_{t+1} = x_t + \tau\nabla_x \log p(x_t) + \sqrt{2\tau}z, \quad z \sim \mathcal{N}(0, I),$$

其中 τ 为迭代步长；x_0 为初始白噪声。该过程类似于随机梯度下降，但目标是寻找概率密度的极大值区域而非最小化损失函数。

推导朗之万动力学采样：基于牛顿第二定律和力与能量的关系 $F = -\nabla U(x)$，代入布朗运动方程

$$m\frac{dv(t)}{dt} = -\gamma v(t) + \eta, \quad \eta \sim \mathcal{N}(0, \sigma^2 I),$$

其中 $-\gamma v(t)$ 为摩擦力，η 为随机力。在时间离散化（设 Δt 为常数）后

$$\frac{x_{t+1}-x_t}{\Delta t}=\frac{1}{\gamma}\nabla U(x)+\frac{\sigma}{\gamma}z.$$

令 $\tau=\Delta t/\gamma$，得到

$$x_{t+1}=x_t+\tau\nabla U(x_t)+\tau\sigma z_t.$$

为采样出概率密度最大的样本（即势能 $U(x)$ 最低），需修正梯度方向。结合玻尔兹曼分布 $\nabla\log p(x)=-\nabla U(x)$，并令 $\sigma=\sqrt{2/\tau}$，得到最终采样公式

$$x_{t+1}=x_t+\tau\nabla_x\log p(x_t)+\sqrt{2\tau}z.$$

该式与随机梯度下降形式高度相似，但目标是寻找概率密度极大值。

朗之万方程的核心价值与理论意义：确定性与随机性的统一建模；涨落-耗散定理揭示了开放系统中能量耗散与随机涨落的内在联系；推动了朗之万蒙特卡洛算法的发展，广泛应用于贝叶斯推断和扩散模型（如去噪扩散概率模型）的潜空间探索。

17.2 朗之万方程在扩散模型中的应用

扩散模型的本质是通过两个马尔可夫过程实现数据生成：正向过程从真实数据 x_0 出发逐步添加高斯噪声，最终得到纯噪声；反向过程从纯噪声出发逐步去除噪声恢复真实数据。

朗之万方程是连接扩散模型、非平衡热力学与随机过程的关键桥梁：正向加噪过程是漂移项为零的朗之万扩散特例；反向去噪过程以朗之万采样的逆过程为建模依据；采样过程本质上是朗之万采样的数值实现。

扩散模型与朗之万方程的结合：分数匹配生成模型

生成式模型的核心任务是学到具有期望特征的数据分布 $p(x)$。传统神经网络直接用最大似然估计拟合 $p(x)$ 面临困境：分布函数应满足 $\forall x, p(x)>0$ 且 $\int p(x)\,dx=1$，但很难对神经网络进行有效的建模和训练。

新的分布函数构造方法：假设神经网络拟合出普通实函数 $f_\theta(x)$，则可用如下方法构造合法分布函数

$$p_\theta(x)=\frac{e^{-f_\theta(x)}}{Z_\theta},\quad Z_\theta=\int e^{-f_\theta(x)}\,dx,$$

此即能量模型对数据分布的建模。

为绕过配分函数 Z_θ 的计算困难，引入**分数函数**（Score function）

$$s_\theta(x) = \nabla_x \log p_\theta(x).$$

$s_\theta(x)$ 是对数概率密度函数关于随机变量 x 的梯度。当对 x 求梯度时，$\log Z_\theta$ 视为常数变为零，绕开了繁琐的高维积分。

损失函数定义为

$$\theta^* = \arg\min_\theta \mathbb{E}_{p(x)}\left[\left\|s_\theta(x) - s(x)\right\|^2\right],$$

然后使用分数匹配方法，通过数学推导得出不含 $s(x)$ 的优化目标。

扩散过程的朗之万动力学采样：

$$x_{t+1} = x_t + \tau\nabla_x \log p(x_t) + \sqrt{2\tau}z,$$

式中 $\nabla_x \log p(x_t)$ 即分数函数。将 x_t 和 x_{t+1} 分别比作图像在 t 时刻和 $t+1$ 时刻的状态，则此方程可以理解为让图像从某个状态根据分数函数运动到概率较高的真实图像的过程。