第 10 章 降维与度量学习

k 近邻（k-Nearest Neighbor, kNN）

监督学习。

“懒惰学习”的代表，没有显式的训练过程。

工作机制：给定测试样本，基于某种距离度量找出训练集中与其最靠近的 k 个训练样本，然后基于这 k 个“邻居”的信息来进行预测。

分类任务：使用“投票法”，即选择这 k 个样本中出现最多的类别标记作为预测结果。

回归任务：使用“平均法”，即将这 k 个样本的实值输出标记的平均值作为预测结果。

错误率与贝叶斯最优分类器错误率对比：

$$
\begin{aligned}
P(err) &= 1 - \sum_{c \epsilon y} P(c|\boldsymbol{x}) P(c|\boldsymbol{z}) \\
&\approx 1 - \sum_{c \epsilon y} P^2(c|\boldsymbol{x}) \\
&\leq 1 - P^2(c^*|\boldsymbol{x}) \\
&= \big(1 + P(c^*|\boldsymbol{x})\big)\big(1 - P(c^*|\boldsymbol{x})\big) \\
&\leq 2 \times (1 - P(c^*|\boldsymbol{x}))
\end{aligned}
$$

$c^* = argmax_{c \epsilon y} P(c|\boldsymbol{x})$表示贝叶斯最优分类器的结果，所以错误率不超过贝叶斯最优分类器的错误率的两倍。

降维

为什么能降维：数据样本虽是高维的，但与学习任务密切相关的也许仅是某个低维分布，即高维空间中的一个低维“嵌入”。

主成分分析（Principal Component Analysis, PCA）

最近重构性：样本点到这个超平面的距离都足够近。

新坐标系$\{\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \dots, \boldsymbol{\omega}_d\}$，其中$||\boldsymbol{\omega}_i||_2 = 1, \boldsymbol{\omega}_i^T \boldsymbol{\omega}_j = 0 (i \neq j)$。

将维度降到$d' < d$，则样本点$\boldsymbol{x}_i$在低维坐标系中的投影为$\boldsymbol{z}_i = (z_{i1}; z_{i2}; \dots; z_{id'}), z_{ij} = \boldsymbol{\omega}_j^T \boldsymbol{x}_i$。重构$\boldsymbol{x}_i$可得$\hat{\boldsymbol{x}}_i = \sum_{j=1}^{d'} z_{ij} \boldsymbol{\omega}_j$。

考虑整个训练集，原样本点$\boldsymbol{x}_i$与重构样本点$\hat{\boldsymbol{x}}_i$之间的距离为

$$
\begin{aligned}
\sum_{i=1}^{m} \left\| \sum_{j=1}^{d'} z_{ij} \boldsymbol{\omega}_j - \boldsymbol{x}_i \right\|_2^2 &= \sum_{i=1}^{m} \big|| \boldsymbol{W} \boldsymbol{z}_i - \boldsymbol{x}_i \big||_2^2 \\
&= \sum_{i=1}^{m} (\boldsymbol{W} \boldsymbol{z}_i)^T (\boldsymbol{W} \boldsymbol{z}_i) - 2 \sum_{i=1}^{m} (\boldsymbol{W} \boldsymbol{z}_i)^T \boldsymbol{x}_i + \sum_{i=1}^{m} (\boldsymbol{x}_i)^T (\boldsymbol{x}_i) \\
&= \sum_{i=1}^{m} \boldsymbol{z}_i^T \boldsymbol{z}_i - 2 \sum_{i=1}^{m} \boldsymbol{z}_i^T \boldsymbol{W}^T \boldsymbol{x}_i + \sum_{i=1}^{m} (\boldsymbol{x}_i)^T (\boldsymbol{x}_i)
\end{aligned}
$$

$$= -\sum_{i=1}^{m} \boldsymbol{x}_i^T \boldsymbol{W}\boldsymbol{W}^T \boldsymbol{x}_i + const$$

$$\propto -\mathrm{tr}\left(\boldsymbol{W}^T \sum_{i=1}^{m} \boldsymbol{x}_i \boldsymbol{x}_i^T \boldsymbol{W}\right)$$

$$= -tr(\boldsymbol{W}^T \mathbf{X}\boldsymbol{X}^T \mathbf{W})$$

其中$\mathbf{W} = (\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \dots, \boldsymbol{\omega}_d)$。即优化目标为

$$\min_{\boldsymbol{W}} -tr(\boldsymbol{W}^T \mathbf{X}\boldsymbol{X}^T \mathbf{W})$$

$$\text{s.t. } \boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{I}$$

最大可分性：样本点在这个超平面上的投影能尽可能分开。

样本点$\boldsymbol{x}_i$的投影为$\boldsymbol{W}^T\boldsymbol{x}_i$，投影尽可能分开，即投影后样本点的方差最大化，投影后样本点的方差为

$$\sum_i \boldsymbol{W}^T \boldsymbol{x}_i \boldsymbol{x}_i^T \boldsymbol{W}$$

即优化目标为

$$\max_{\boldsymbol{W}} tr(\boldsymbol{W}^T \mathbf{X}\boldsymbol{X}^T \mathbf{W})$$

$$\text{s.t. } \boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{I}$$

显然与最近重构性得出的优化目标等价。

求解：对上面两个优化目标中的任意一个使用拉格朗日乘子法可得

$$\mathbf{X}\boldsymbol{X}^T\boldsymbol{\omega}_i = \lambda_i \boldsymbol{\omega}_i$$

只需对协方差矩阵$\mathbf{X}\boldsymbol{X}^T$进行特征值分解，将求得的特征值排序后取前$d'$个特征值对应的特征向量构成主成分分析的解：

$$\boldsymbol{W}^* = (\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \dots, \boldsymbol{\omega}_{d'})$$

d'的设置：

1. 用户指定。
2. 在低维空间中对 k 近邻或其他分类器进行交叉验证。
3. 设置重构阈值，例如 $t = 95\%$，然后选取最小的d'使得$\frac{\sum_{i=1}^{d'} \lambda_i}{\sum_{i=1}^{d} \lambda_i} \geq t$。

度量学习

两个样本x_i、x_j的欧式距离：

$$d(x_i, x_j) = (x_i - x_j)^T (x_i - x_j)$$

欧式距离无法很好地表达样本之间的关系，引入变换矩阵 A，于是得到马氏距离：

$$d(x_i, x_j) = (Ax_i - Ax_j)^T (Ax_i - Ax_j)$$

近邻成分分析（Neighbourhood Components Analysis, NCA）

样本x_i将样本x_j作为近邻并且被标记为与x_j相同标记的概率为：

$$p_{ij} = \frac{\exp(-||Ax_i - Ax_j||^2)}{\sum_{k \neq i} \exp(-||Ax_i - Ax_k||^2)} \quad , \quad p_{ii} = 0$$

因此，样本x_i被正确标记的概率为：

$$p_i = \sum_{j \in C_i} p_{ij}, C_i = \{j \mid c_i = c_j\}$$

由此可以得到目标函数，即所有样本被正确标记的概率为：

$$f(A) = \sum_i \sum_{j \in C_i} p_{ij} = \sum_i p_i$$

即优化目标为：

$$\max_A f(A)$$

第 11 章 特征选择与稀疏学习

特征分类：

1. 相关特征：对当前学习任务有用的属性。
2. 无关特征：与当前学习任务无关的属性。
3. 冗余特征：其所包含的信息能由其他特征推演出来。

特征选择：从给定的特征集合中选出任务相关的特征子集且必须确保不丢失重要特征。

特征选择的原因：减轻维度灾难；降低学习难度。

子集搜索（皆为贪心策略）：

1. 前向搜索：逐渐增加相关特征。
2. 后向搜索：从完整的特征集合开始，逐渐减少特征。
3. 双向搜索：每一轮逐渐增加相关特征，同时减少无关特征。

子集评价：特征子集确定了对数据集的一个划分，样本标记对应着对数据集的真实划分，通过估算这两个划分的差异就能对特征子集进行评价，差异越小说明特征子集越好。

A 为特征子集，D 为数据集，D 中第 i 类样本占比为$p_i(i = 1,2,\dots,|y|)$，A 上的取值将 D 分为 V 份，每份用D^v表示，则 A 的信息增益为：

$$\mathrm{Gain}(A) = \mathrm{Ent}(D) - \sum_{v=1}^{V} \frac{|D^v|}{|D|} \mathrm{Ent}(D^v)$$

其中信息熵定义为：

$$\mathrm{Ent}(D) = -\sum_{k=1}^{|y|} p_k \log_2 p_k$$

特征选择方法

1. **过滤式：**先用特征选择过程过滤原始数据，再用过滤后的特征来训练模型，特征选择过程与后续学习器无关。

   Relief（Relevant Features）

   - 为每个初始特征赋予一个“相关统计量”，度量特征的重要性。
   - 特征子集的重要性由子集中每个特征所对应的相关统计量之和决定。
   - 设置一个阈值，选择相关统计量大于阈值的分量所对应的特征。
   - 指定欲选取的特征个数 k，选择相关统计量分量最大的 k 个特征。
   - **确定相关统计量**

     猜中近邻：$x_{i,nh}$，表示x_i同类样本中的最近邻。

     猜错近邻：$x_{i,nm}$，表示x_i异类样本中的最近邻。

相关统计量对应于属性 j 的分量为：

$$\delta^j = \sum_i -\text{diff}\left(x_i^j, x_{i,nh}^j\right)^2 + \text{diff}(x_i^j, x_{i,nm}^j)^2$$

若 j 为离散型，则$x_a^j = x_b^j$时，$\text{diff}\left(x_a^j, x_b^j\right) = 0$，否则为 1；若 j 为连续型，则$\text{diff}\left(x_a^j, x_b^j\right) = |x_a^j - x_b^j|$，$x_a^j, x_b^j$已经规范到[0, 1]区间。

- 相关统计量越大，属性 j 上，猜中近邻比猜错近邻越近，即属性 j 对区分对错越有用。
- 时间开销随采样次数以及原始特征数线性增长，运行效率很高。
- 多分类拓展 —— Relief-F

  数据集中的样本来自$|y|$个类别，其中x_i属于第 k 类。

  猜中近邻：$x_{i,nh}$，表示第 k 类中x_i的最近邻。

  猜错近邻：$x_{i,l,nm}(l = 1,2,\ldots,|y|; l \neq k)$，表示第 k 类之外的每一类中找到一个x_i的最近邻。

  相关统计量对应于属性 j 的分量为：

$$\delta^j = \sum_i -\text{diff}\left(x_i^j, x_{i,nh}^j\right)^2 + \sum_{l \neq k}(p_l \times \text{diff}(x_i^j, x_{i,l,nm}^j)^2)$$

  其中p_l为第 l 类样本在数据集 D 中的占比。

2. **包裹式：**直接把最终将要使用的学习器的性能作为特征子集的评价准则。
   - 包裹式特征选择的目的就是为给定学习器选择最有利于其性能、“量身定做”的特征子集。
   - 直接针对给定学习器进行优化，因此从最终学习器性能来看，包裹式特征选择比过滤式特征选择更好。
   - 需多次训练学习器，计算开销通常比过滤式特征选择大得多。

   LVW（Las Vegas Wrapper）
   - 在循环的每一轮随机产生一个特征子集。
   - 在随机产生的特征子集上通过交叉验证推断当前特征子集的误差。
   - 进行多次循环，在多个随机产生的特征子集中选择误差最小的特征子集作为最终解。

   若有运行时间限制，该算法有可能给不出解。
3. **嵌入式：**将特征选择过程与学习器训练过程融为一体，两者在同一个优化过程中完成，在学习器训练中自动进行特征选择。
   - 考虑最简单的线性回归模型，以平方误差为损失函数，引入L_2范数正则化项防止过拟合，则有

$$\min_\omega \sum_{i=1}^m (y_i - \omega^T x_i)^2 + \lambda||\omega||_2^2$$

     其中正则化参数 $\lambda > 0$，上式称为“岭回归”（ridge regression）。
   - 将L_2范数替换为L_1范数，则有

$$\min_\omega \sum_{i=1}^m (y_i - \omega^T x_i)^2 + \lambda||\omega||_1$$

     其中正则化参数$\lambda > 0$，上式称为LASSO（Least Absolute Shrinkage and Selection Operator）。
   - 通过等值线可以看出L_1范数更易于获得稀疏解。

稀疏表示：

- 将数据集考虑成一个矩阵，每行对应一个样本，每列对应一个特征。
- 矩阵中有很多零元素，且非整行整列出现。
- 稀疏表示的优势：文本数据线性可分；存储高效。

字典学习

为普通稠密表达的样本找到合适的字典，将样本转化为稀疏表示，这一过程称为字典学习。

- 给定数据集$\{\boldsymbol{x}_1,\boldsymbol{x}_2,\dots,\boldsymbol{x}_m\},\boldsymbol{x}_i\in \mathrm{R}^{n\times k}$。
- 学习目标是字典矩阵$\mathbf{B}\in \mathrm{R}^{d\times k}$以及样本的稀疏表示$\boldsymbol{\alpha}_i\in \mathrm{R}^k$。
- K 称为字典的词汇量，通常由用户指定。
- 则最简单的字典学习的优化形式为

$$\min_{\boldsymbol{B},\boldsymbol{\alpha}_i}\sum_{i=1}^{m}||\boldsymbol{x}_i-\boldsymbol{B}\boldsymbol{\alpha}_i||_2^2+\lambda\sum_{i=1}^{m}||\boldsymbol{\alpha}_i||_1$$

第 12 章 计算学习理论

第 13 章 半监督学习

第 14 章 概率图模型

概率模型：提供了一种描述框架，将描述任务归结为计算变量的概率分布。在概率模型中，利用已知变量推测未知变量的分布称为“推断（inference）”，其核心是如何基于可观测变量推测出未知变量的条件分布。

概率图模型：是一类用图来表达变量相关关系的概率模型。它以图为表示工具，用结点表示随机变量，边表示变量之间的概率相关关系。分类：使用有向无环图表示变量间的依赖关系，称为有向图模型或贝叶斯网；使用无向图表示变量间的相关关系，称为无向图模型或马尔可夫网。

隐马尔可夫模型（Hidden Markov Model, HMM）

组成：

1. 状态变量，即隐变量，$\boldsymbol{y}=\{y_1,y_2,\dots,y_n\}$，$y_i\in\boldsymbol{y}$表示 i 时刻的系统状态。
2. 观测变量，$\boldsymbol{\chi}=\{x_1,x_2,\dots,x_n\}$，$x_i\in\boldsymbol{\chi}$表示 i 时刻的观测值。

马尔可夫链：t 时刻的状态x_t仅依赖于 t - 1 时刻状态x_{t-1}，与其余 n - 2 个状态无关。即系统下一时刻状态仅由当前状态决定，不依赖于以往的任何状态。

所有变量的联合概率分布：

$$P(x_1,y_1,\dots,x_n,y_n)=P(y_1)P(x_1|y_1)\prod_{i=2}^{n}P(y_i|y_i-1)P(x_i|y_i)$$

确定一个 HMM 需三组参数（$\boldsymbol{\lambda}=[\mathbf{A},\mathbf{B},\boldsymbol{\pi}]$）：

1. 状态转移概率：模型在各个状态间转换的概率。$\mathrm{A}=[a_{ij}]_{N\times N},a_{ij}=P(y_{t+1}=s_j|y_t=s_i),1\le i,j\le N$，即任意时刻 t,若状态为s_i，则在下一刻状态

为s_j的概率。

2. 输出观测概率：模型根据当前状态获得各个观测值的概率。$\mathrm{B}=[b_{ij}]_{N\times M}, b_{ij}=P(x_t=o_j|y_t=s_i), 1\le i\le N, 1\le j\le M$，即任意时刻 t，若状态为s_i，则观测值o_j被获取的概率。
3. 初始状态概率：模型在初始时刻各状态出现的概率。$\boldsymbol{\pi}=(\pi_1,\pi_2,\dots,\pi_n), \pi_i=P(y_1=s_i), 1\le i\le N$，表示模型的初始概率为$s_i$的概率。

生出观测序列的过程：

1. 设置 $t=1$，并根据初始状态概率$\boldsymbol{\pi}$选择初始状态y_1；
2. 根据状态y_t和输出观测概率 **B** 选择观测变量取值x_t；
3. 根据状态y_t和状态转移概率 **A** 转移模型状态，即确定y_{t+1}；
4. 若 $t<n$，设置 $t=t+1$，并转到第 2 步，否则停止。

HMM 的基本问题：

1. 评估模型和观测序列之间的匹配程度：给定模型$\lambda=[\mathbf{A},\mathbf{B},\boldsymbol{\pi}]$，如何有效计算其产生观测序列$\mathbf{x}=\{x_1,x_2,\dots,x_T\}$的概率$P(\mathbf{x}|\lambda)$？

   前向算法：引入前向概率$\alpha_t(i)$，即基于观测序列 **x**，t 时刻观测值为x_t且状态为 i 的概率。
   当 $t=1$ 时，易知前向概率为初始概率与观测概率乘积，即$\alpha_1(i)=\pi_i\cdot b_{ix_1}$；
   当 $t>1$ 时，易知前向概率为 $t-1$ 时刻到 t 时刻所有路径的概率之和乘以 t 时刻的观测概率，即$\alpha_t(i)=b_{ix_t}\cdot\sum_{j=1}^{n}\alpha_{t-1}(j)\cdot a_{ji}$；
   要计算产生观测序列 **x** 的概率，只需由上述前向概率的递归表达式依次从 $t=1$ 计算至 $t=T$，易得$P(\mathbf{x}|\lambda)=\sum_{i=1}^{n}\alpha_T(i)$。

2. 根据观测序列推断出隐藏的模型状态：给定模型$\lambda=[\mathbf{A},\mathbf{B},\boldsymbol{\pi}]$和观测序列$\mathbf{x}=\{x_1,x_2,\dots,x_n\}$，如何找到与此观测序列最匹配的状态序列$\mathbf{y}=\{y_1,y_2,\dots,y_n\}$？

   维特比算法：引入局部概率$\delta_t(i)$，即基于观测序列 **x**，t 时刻观测值为x_t且状态为 i 的最大概率；引入前驱指针$\varphi_t(i)$，即基于观测序列 **x**，t 时刻观测值为x_t且最可能的状态为 i 时的前一个状态。
   当 $t=1$ 时，易知局部概率为初始概率与观测概率乘积，即$\delta_1(i)=\pi_i\cdot b_{ix_1}$，前驱指针$\varphi_1(i)=0$；
   当 $t>1$ 时，易知局部概率为 $t-1$ 时刻到 t 时刻最可能的路径的概率乘以 t 时刻的观测概率，即$\delta_t(i)=b_{ix_t}\cdot\max_j(\delta_{t-1}(j)\cdot a_{ji})$，求得 j 之后，前驱指针$\varphi_t(i)=j$；
   要找出与观测序列最匹配的状态序列 **y**，只需由上述局部概率的递归表达式依次从 $t=1$ 计算至 $t=T$，然后由前驱指针从 $t=T$ 到 $t=1$ 构造出最可能的状态序列 **y**。

3. 参数学习，即训练模型使其能最好地描述观测数据：给定观测序列$\mathbf{x}=\{x_1,x_2,\dots,x_n\}$，如何调整模型参数$\lambda=[\mathbf{A},\mathbf{B},\boldsymbol{\pi}]$使得该序列出现的概率$P(\mathbf{x}|\lambda)$最大？

马尔可夫随机场（Markov Random Field, MRF）

典型的马尔可夫网，著名的无向图模型。

团：对于图中结点的一个子集，其中任意两结点间都有边连接。

极大团：一个团中加入另外任何一个结点都不再形成团，即极大团就是不能被其他团所包含的团。每个结点至少出现在一个极大团中。

基于团的势函数

多个变量之间的联合概率分布能基于团分解为多个因子的乘积，每个因子仅与一个团有关。对于 n 个变量$\mathbf{x} = \{x_1, x_2, \ldots, x_n\}$，所有团构成的集合 C，与团$Q \in C$对应的变量集合为$\mathbf{x}_Q$，则联合概率 $P(\mathbf{x})$定位为

$$P(\mathbf{x}) = \frac{1}{Z}\prod_{Q\in C}\psi_Q(\mathbf{x}_Q)$$

其中ψ_Q为与团 Q 对应的势函数，Z 为概率的规范化因子。

基于极大团的势函数

若变量个数较多，团的数目将会很多，上式将会有很多乘积项，计算复杂。若团 Q 不是极大团，则它必被一个极大团Q^*所包含，即$\mathbf{x}_Q \subseteq \mathbf{x}_{Q^*}$。$C^*$为所有极大团构成的集合，基于极大团定义 $P(\mathbf{x})$

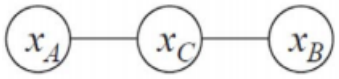
$$P(\mathbf{x}) = \frac{1}{Z^*}\prod_{Q\in C^*}\psi_Q(\mathbf{x}_Q)$$

条件独立性

分离机：若从结点集 A 中的结点到 B 中的结点都必须经过结点集 C 中的结点，则称结点集 A 和 B 被结点集 C 分离，C 称为“分离集”（separating set）。

全局马尔可夫性（global Markov property）：给定两个变量子集$\mathbf{x}_A, \mathbf{x}_B$的分离集$\mathbf{x}_C$，则这两个变量子集条件独立，记为$\mathbf{x}_A \perp \mathbf{x}_B|\mathbf{x}_C$。

全局马尔可夫性验证



上图联合概率：

$$P(\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_C) = \frac{1}{z}\psi_{AC}(\mathbf{x}_A,\mathbf{x}_C)\psi_{BC}(\mathbf{x}_B,\mathbf{x}_C)$$

由条件概率可得：

$$\begin{aligned}
P(\mathbf{x}_A,\mathbf{x}_B|\mathbf{x}_C) &= \frac{P(\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_C)}{P(\mathbf{x}_C)} = \frac{P(\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_C)}{\sum_{\mathbf{x'}_A}\sum_{\mathbf{x'}_B}P(\mathbf{x'}_A,\mathbf{x'}_B,\mathbf{x}_C)} \\
&= \frac{\frac{1}{z}\psi_{AC}(\mathbf{x}_A,\mathbf{x}_C)\psi_{BC}(\mathbf{x}_B,\mathbf{x}_C)}{\sum_{\mathbf{x'}_A}\sum_{\mathbf{x'}_B}\frac{1}{Z}\psi_{AC}(\mathbf{x'}_A,\mathbf{x}_C)\psi_{BC}(\mathbf{x'}_B,\mathbf{x}_C)} \\
&= \frac{\psi_{AC}(\mathbf{x}_A,\mathbf{x}_C)}{\sum_{\mathbf{x'}_A}\psi_{AC}(\mathbf{x'}_A,\mathbf{x}_C)}\cdot\frac{\psi_{BC}(\mathbf{x}_B,\mathbf{x}_C)}{\sum_{\mathbf{x'}_B}\psi_{BC}(\mathbf{x'}_B,\mathbf{x}_C)} \\
P(\mathbf{x}_A|\mathbf{x}_C) &= \frac{P(\mathbf{x}_A,\mathbf{x}_C)}{P(\mathbf{x}_C)} = \frac{\sum_{\mathbf{x'}_B}P(\mathbf{x}_A,\mathbf{x'}_B,\mathbf{x}_C)}{\sum_{\mathbf{x'}_A}\sum_{\mathbf{x'}_B}P(\mathbf{x'}_A,\mathbf{x'}_B,\mathbf{x}_C)} \\
&= \frac{\sum_{\mathbf{x'}_B}\frac{1}{Z}\psi_{AC}(\mathbf{x}_A,\mathbf{x}_C)\psi_{BC}(\mathbf{x'}_B,\mathbf{x}_C)}{\sum_{\mathbf{x'}_A}\sum_{\mathbf{x'}_B}\frac{1}{Z}\psi_{AC}(\mathbf{x'}_A,\mathbf{x}_C)\psi_{BC}(\mathbf{x'}_B,\mathbf{x}_C)} \\
&= \frac{\psi_{AC}(\mathbf{x}_A,\mathbf{x}_C)}{\sum_{\mathbf{x'}_A}\psi_{AC}(\mathbf{x'}_A,\mathbf{x}_C)}
\end{aligned}$$

由此可得：

$$P(\mathbf{x}_A, \mathbf{x}_B|\mathbf{x}_C) = P(\mathbf{x}_A|\mathbf{x}_C)P(\mathbf{x}_B|\mathbf{x}_C)$$

局部马尔可夫性（local Markov property）：给定某变量的邻接变量，则该变量条件独立于其他变量。即，令 V 为图的结点集，$n(v)$为结点 v 在图上的邻接结点，$n^*(v) = n(v) \cup \{v\}$，有$\mathbf{x}_v \perp \mathbf{x}_{V\setminus n^*(v)}|\mathbf{x}_{n(v)}$。

成对马尔可夫性（pairwise Markov property）：给定所有其他变量，两个非邻接变量条件独立。即，令图的结点集和边集分别为 V 和 E，对图中的两个结点 u 和 v，若 $\langle u,v\rangle \notin E$，则$\mathbf{x}_u \perp \mathbf{x}_v|\mathbf{x}_{V\setminus\langle u,v\rangle}$。

势函数定义：为了满足非负性，指数函数常被用于定义势函数，即：

$$\psi_Q(\mathbf{x}_Q) = e^{-H_Q(\mathbf{x}_Q)}$$

$H_Q(\mathbf{x}_Q)$是一个定义在变量$\mathbf{x}_Q$上的实值函数，常见形式为：

$$H_Q(\mathbf{x}_Q) = \sum_{u,v\in Q, u\neq v} \alpha_{uv} x_u x_v + \sum_{v\in Q} \beta_v x_v$$

其中α_{uv}和β_v是参数。上式中第二项仅考虑单结点，第一项则考虑每一对结点的关系。

条件随机场（Conditional Random Field, CRF）

略。

MRF 与 CRF 对比

	MRF	CRF
概率定义	使用团上的势函数定义概率	使用团上的势函数定义概率
建模	对联合概率建模	有观测变量，对条件概率建模

概率图模型的推断方法：

1. **精确推断：**计算出目标变量的边际分布或条件分布的精确值。计算复杂度随极大团规模增长呈指数增长，适用范围有限。
   - 变量消去
   - 信念传播
2. **近似推断：**精确推断计算开销很大，现实应用中常用近似推断。
   - 采样法（sampling）：通过使用随机化方法完成近似，如 MCMC 采样。
   - 变分推断（variational inference）：使用确定性近似完成推断。

第 15 章 规则学习

序贯覆盖：

- 自顶向下策略：一般到特殊（特化）。
- 自底向上策略：特殊到一般（泛化）。
- 规则评判：增加/删除哪一个候选文字？可通过准确率、信息熵增益等来评估。
- 规避局部最优：可使用集束搜索，即每次保留最优的多个候选规则。

剪枝优化：

- 预剪枝。
- 后剪枝

  减错剪枝（Reduced Error Pruning, REP）
  - 穷举所有可能的剪枝操作，包括删除文字、删除规则等。
  - 用验证集反复剪枝直到准确率无法提高。

  IREP（Incremental REP）

- 每生成一条新规则即对其进行 REP 剪枝。

第 16 章 强化学习

强化学习：通常用马尔可夫决策过程（Markov Decision Process, MDP）来描述。对应于四元组 $E = <X, A, P, R>$，其中 X 为状态空间，A 为动作空间，P 指定了状态转移概率，R 指定了奖赏。

K 摇臂赌博机

增量式计算均值：初始时$Q_0(k) = 0$，对于任意$n \geq 1$，若第 $n-1$ 次尝试后的平均奖赏为$Q_{n-1}(k)$，则在经过第 n 次尝试获得奖赏v_n后，平均奖赏应更新为：

$$Q_n(k) = \frac{1}{n}\left((n-1) \times Q_{n-1}(k) + v_n\right)$$

$$= Q_{n-1}(k) + \frac{1}{n}\left(v_n - Q_{n-1}(k)\right)$$

策略评估

状态值函数：

$$\begin{cases} V_T^{\pi}(x) = E_{\pi}\left[\frac{1}{T}\sum_{t=1}^{T} r_t \,\middle|\, x_0 = x\right], T\text{步累积奖赏} \\ V_{\gamma}^{\pi}(x) = E_{\pi}\left[\frac{1}{T}\sum_{t=0}^{+\infty} \gamma^t r_{t+1} \,\middle|\, x_0 = x\right], \gamma\text{折扣累积奖赏} \end{cases}$$

状态-动作值函数：

$$\begin{cases} Q_T^{\pi}(x,a) = E_{\pi}\left[\frac{1}{T}\sum_{t=1}^{T} r_t \,\middle|\, x_0 = x, a_0 = a\right] \\ Q_{\gamma}^{\pi}(x,a) = E_{\pi}\left[\frac{1}{T}\sum_{t=0}^{+\infty} \gamma^t r_{t+1} \,\middle|\, x_0 = x, a_0 = a\right] \end{cases}$$

Bellman 等式：

由于 MDP 具有马尔可夫性质。即系统下一时刻的状态仅由当前时刻的状态决定，不依赖于以往任何状态，于是值函数有很简单的递归形式。

T 步累积奖赏有

$$V_T^{\pi}(x) = E_{\pi}\left[\frac{1}{T}\sum_{t=1}^{T} r_t \,\middle|\, x_0 = x\right]$$

$$= E_{\pi}\left[\frac{1}{T} r_1 + \frac{T-1}{T}\frac{1}{T-1}\sum_{t=2}^{T} r_t \,\middle|\, x_0 = x\right]$$

$$= \sum_{a \epsilon A} \pi(x,a) \sum_{x' \in X} P_{x \to x'}^{a}\left(\frac{1}{T} R_{x \to x'}^{a} + \frac{T-1}{T} E_{\pi}\left[\frac{1}{T-1}\sum_{t=1}^{T-1} r_t \,\middle|\, x_0 = x'\right]\right), (\text{动作} - \text{状态全概率展开})$$

$$= \sum_{a \epsilon A} \pi(x,a) \sum_{x' \in X} P_{x \to x'}^{a}\left(\frac{1}{T} R_{x \to x'}^{a} + \frac{T-1}{T} V_{T-1}^{\pi}(x')\right)$$

类似的，对于 γ 折扣累积奖赏有

$$V_{\gamma}^{\pi}(x) = \sum_{a \epsilon A} \pi(x,a) \sum_{x' \in X} P_{x \to x'}^{a}\left(R_{x \to x'}^{a} + \gamma V_{\gamma}^{\pi}(x')\right)$$

由此可直接计算出状态-动作值函数

$$\begin{cases} Q_T^{\pi}(x,a)=\sum_{x'\in X}P_{x\to x'}^{a}\left(\frac{1}{T}R_{x\to x'}^{a}+\frac{T-1}{T}V_{T-1}^{\pi}(x')\right) \\ Q_\gamma^{\pi}(x,a)=\sum_{x'\in X}P_{x\to x'}^{a}\left(R_{x\to x'}^{a}+\gamma V_\gamma^{\pi}(x')\right) \end{cases}$$

策略改进

理想策略应能最大化累积奖赏

$$\pi^{*}=\mathrm{argmax}_{\pi}\sum_{x\in X}V^{\pi}(x)$$

最优策略对应的值函数称为最优值函数

$$\forall x\in \mathrm{X}:V^{*}(x)=V^{\pi^{*}}(x)$$

最优 Bellman 等式：

改进 Bellman 等式，将对动作求和改为取最优：

$$\begin{cases} V_T^{*}(x)=\max_{a\in A}\sum_{x'\in X}P_{x\to x'}^{a}\left(\frac{1}{T}R_{x\to x'}^{a}+\frac{T-1}{T}V_{T-1}^{*}(x')\right) \\ V_\gamma^{*}(x)=\max_{a\in A}\sum_{x'\in X}P_{x\to x'}^{a}\left(R_{x\to x'}^{a}+\gamma V_\gamma^{*}(x')\right) \end{cases}$$

即：

$$V^{*}(x)=\max_{a\in A}Q^{\pi^{*}}(x,a)$$

代入状态-动作值函数可得最优状态-动作值函数：

$$\begin{cases} Q_T^{*}(x,a)=\sum_{x'\in X}P_{x\to x'}^{a}\left(\frac{1}{T}R_{x\to x'}^{a}+\frac{T-1}{T}\max_{a'\in A}Q_{T-1}^{*}(x',a')\right) \\ Q_\gamma^{*}(x,a)=\sum_{x'\in X}P_{x\to x'}^{a}\left(R_{x\to x'}^{a}+\gamma\max_{a'\in A}Q_\gamma^{*}(x',a')\right) \end{cases}$$

时序差分（Temporal Difference, TD）学习

基于 t 个采样估计出值函数$Q_t^{\pi}(x,a)=\frac{1}{t}\sum_{i=1}^{t}r_i$，则在得到第 $t+1$ 个采样r_{t+1}时，根据上述 K 摇臂赌博机中的公式可得

$$Q_{t+1}^{\pi}(x,a)=Q_t^{\pi}(x,a)+\frac{1}{t+1}(r_{t+1}-Q_t^{\pi}(x,a))$$

令$\alpha=\frac{1}{t+1}$，由状态-动作值函数：

$$\begin{aligned} Q^{\pi}(x,a)&=\sum_{x'\in X}P_{x\to x'}^{a}\left(R_{x\to x'}^{a}+\gamma V^{\pi}(x')\right) \\ &=\sum_{x'\in X}P_{x\to x'}^{a}\left(R_{x\to x'}^{a}+\gamma\sum_{a'\in A}\pi(x',a')Q^{\pi}(x',a')\right) \end{aligned}$$

通过增量求和可得：

$$Q_{t+1}^{\pi}(x,a)=Q_t^{\pi}(x,a)+\alpha(R_{x\to x'}^{a}+\gamma Q_t^{\pi}(x',a')-Q_t^{\pi}(x,a))$$