第十一讲 特征选择和稀疏学习

高级机器学习

特征

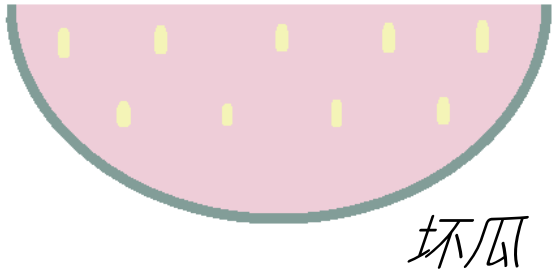
- 特征：描述物体的属性

- 几类不同的特征
  - **相关特征**：对当前学习任务有用的属性
  - **无关特征**：与当前学习任务无关的属性
  - **冗余特征**：其所包含信息能由其他特征推演出来

例子：西瓜的特征

西瓜的特征
- 无关特征
  - 颜色
  - 纹理
  - 触感
- 相关特征
  - 根蒂
  - 声音

当前任务：西瓜是否是好瓜



坏瓜

好瓜

特征选择

- “特征选择”应运而生，其目的主要在于
  - 从给定的特征集合中选出**任务相关**特征子集
  - 不丢失重要特征
- 特征选择的优势
  - 减轻维度灾难：在少量属性上构建模型
  - 降低学习难度：留下关键信息

例子：判断是否好瓜时的特征选择

西瓜的特征
- 颜色
- 纹理
- 触感

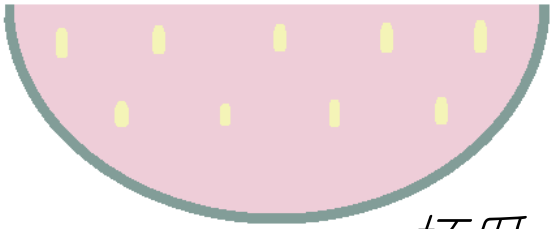
→ 无关特征

- 根蒂
- 声音

→ 相关特征

特征选择：选择当前任务相关特征

当前任务：西瓜是否是好瓜



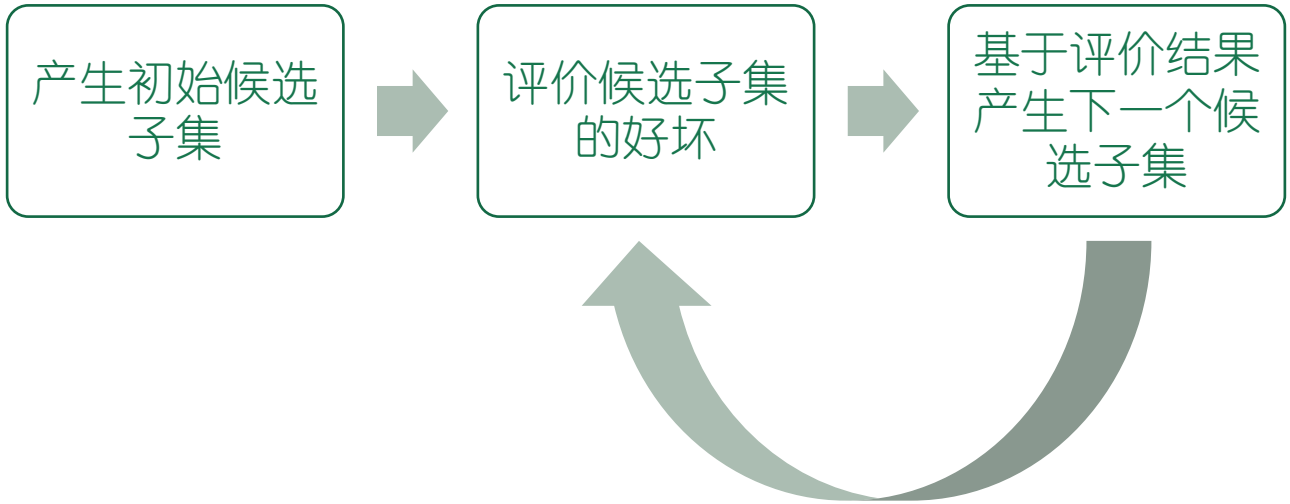
坏瓜

好瓜

特征选择的一般方法

- 遍历所有可能的子集
  - 组合爆炸，不可行

- 可行方法



两个关键环节：子集搜索 和 子集评价

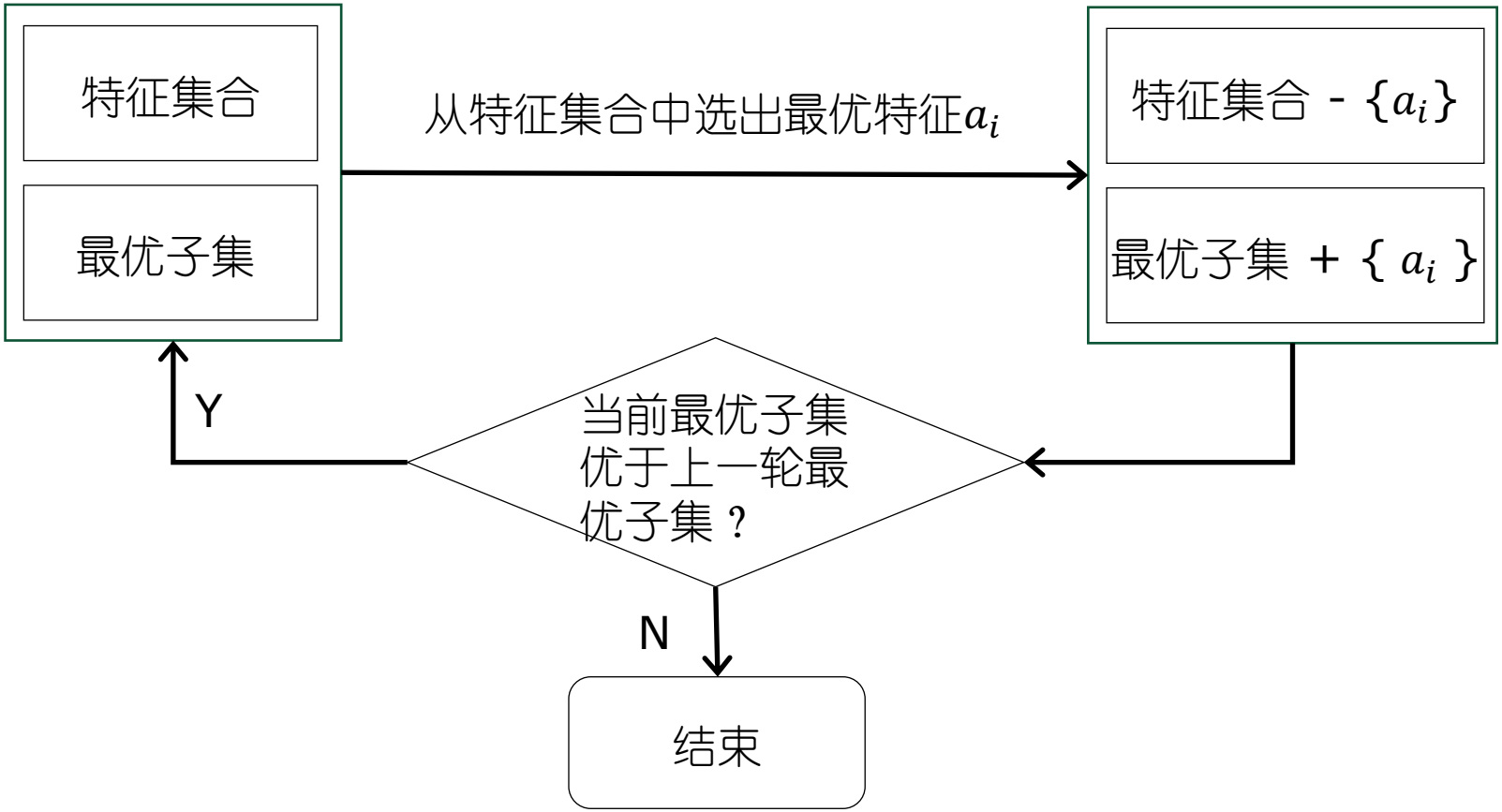
子集搜索

用贪心策略选择包含重要信息的特征子集

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

前向搜索

- 最优子集初始为空集，特征集合初始时包括所有给定特征



子集搜索

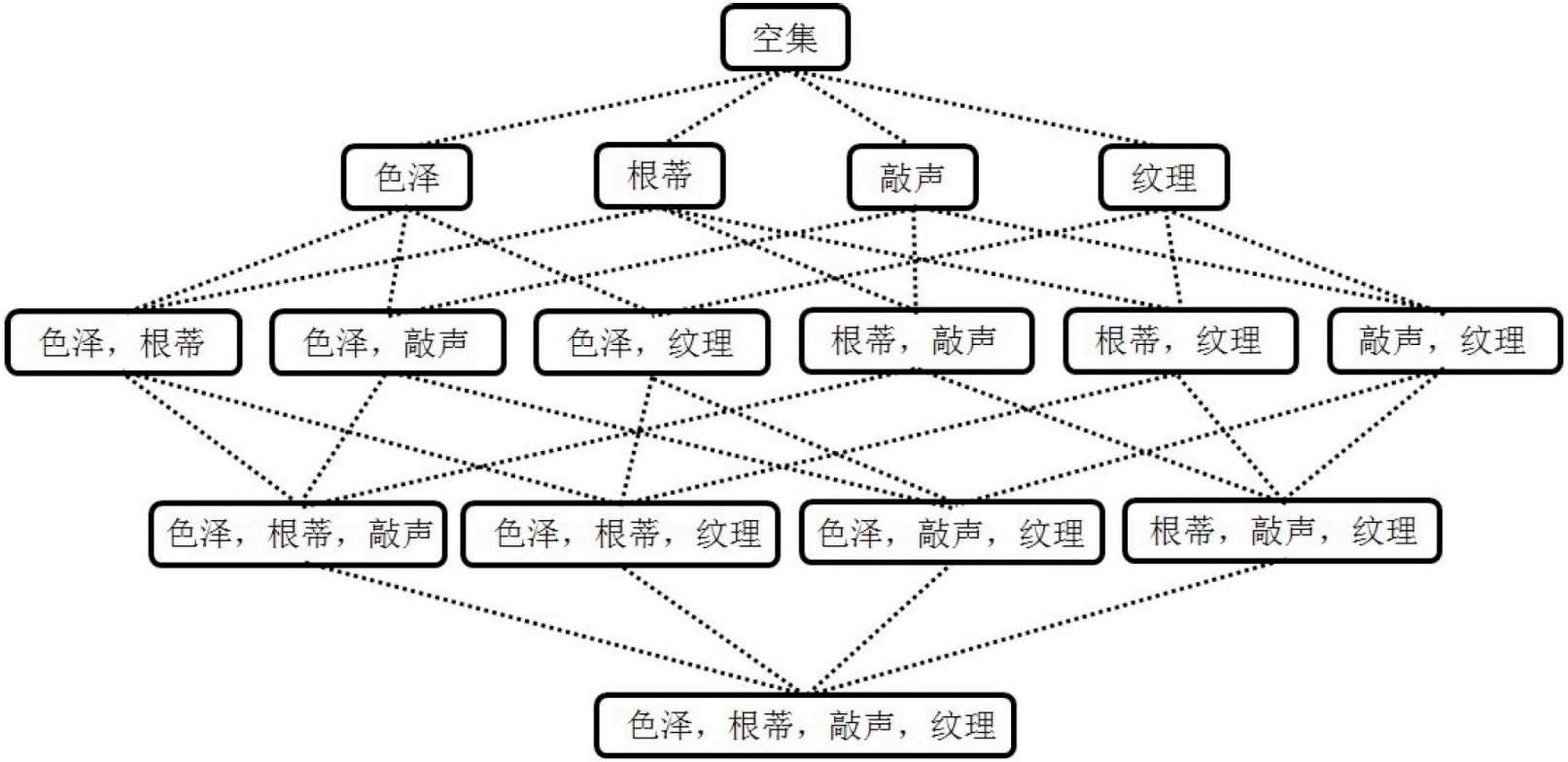


图 1.1　由4个属性及在属性子集上的交并运算定义的格

子集评价

- 特征子集确定了对数据集的一个划分
  - 每个划分区域对应着特征子集的某种取值
- 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；与样本标记对应的划分的差异越小，则说明当前特征子集越好

用信息熵进行子集评价

- 特征子集A确定了对数据集D的一个划分
  - A上的取值将数据集D分为V份，每一份用D^v表示
  - $\mathrm{Ent}(D^v)$表示D^v上的信息熵
- 样本标记Y对应着对数据集D的真实划分
  - $\mathrm{Ent}(D)$表示D上的信息熵

D上的信息熵定义为

$$\mathrm{Ent}(D) = \sum_{i=1}^{|Y|} p_k \log_2 p_k$$

第i类样本所占比例为p_i

特征子集A的信息增益为：

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

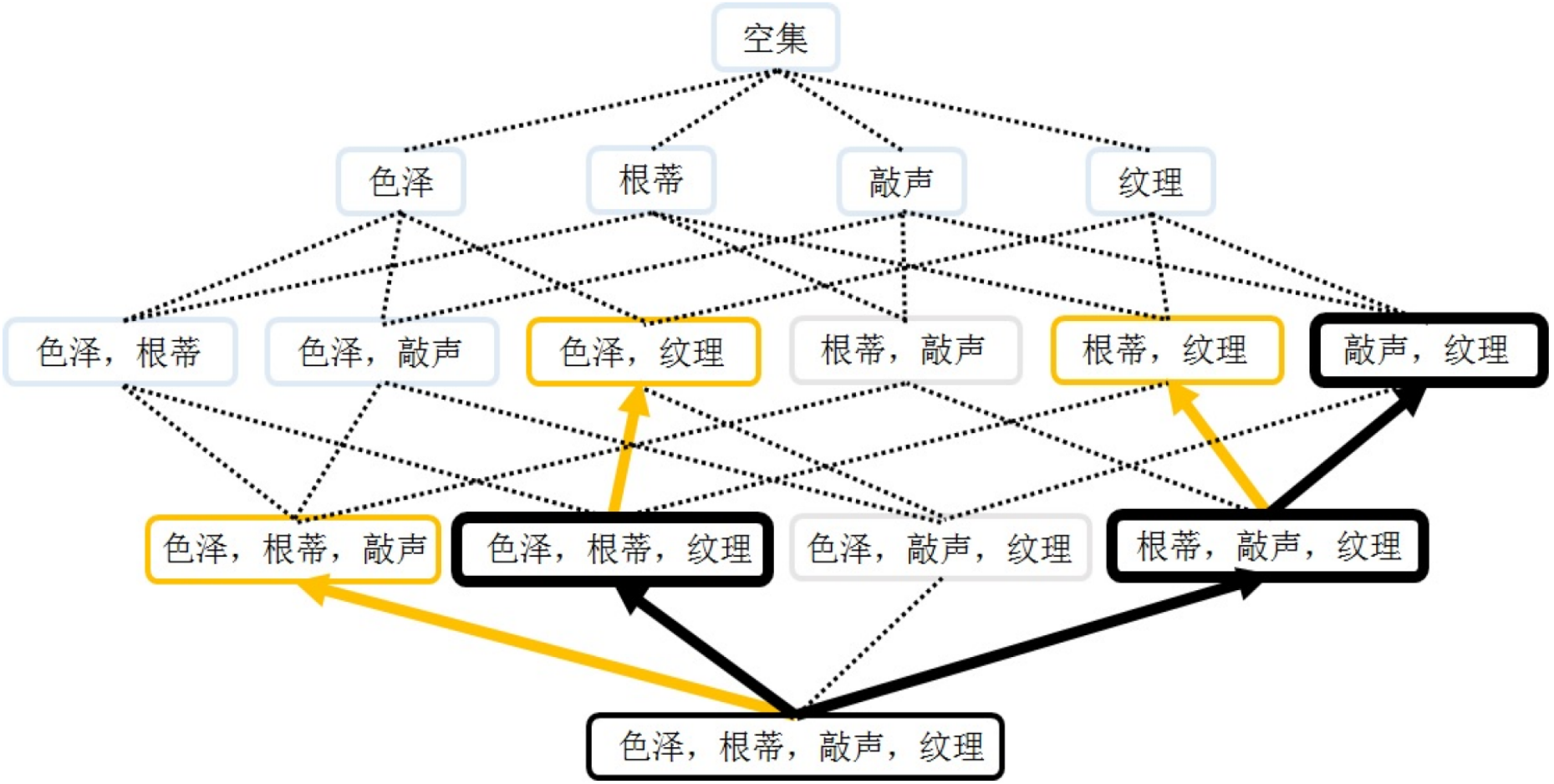


图 1.3　分支限界策略在格上的搜索路径：黑色箭头指向该策略每一步选取的属性子集（粗黑线框标注出每层去掉一个属性后可行的选择结果）；黄色箭头表示每一步尝试但不成功的搜索方向(黄色方框框出了在每次尝试后决定剪枝的属性子集）；蓝色框中的属性子集无需评价（已经被剪枝或一定会被其他分支搜索）

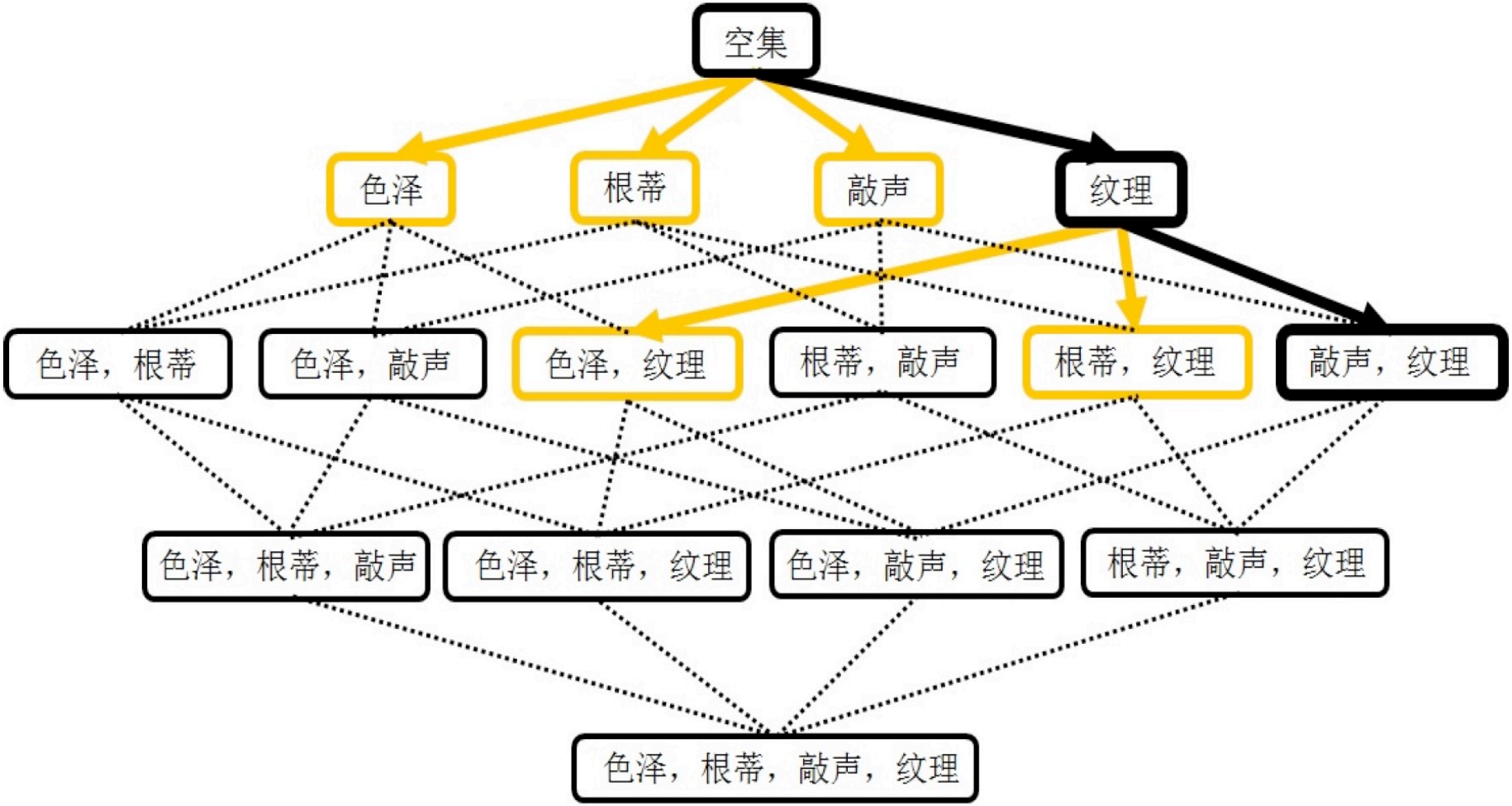


图 1.4　贪心搜索策略在格上的搜索路径：黑色箭头标注出该策略每一步选取的属性子集；黄色箭头表示每一步尝试的搜索方向；粗黑线框标注出最终的选择结果；黄色线框出了策略未选中的属性子集；其他框中的属性子集无需进行评估。

常见的特征选择方法

子集搜索机制与子集评价机制相结合，形成各种的特征选择方法

常见的特征选择方法大致分为如下三类：

- 过滤式
- 包裹式
- 嵌入式

Filter式

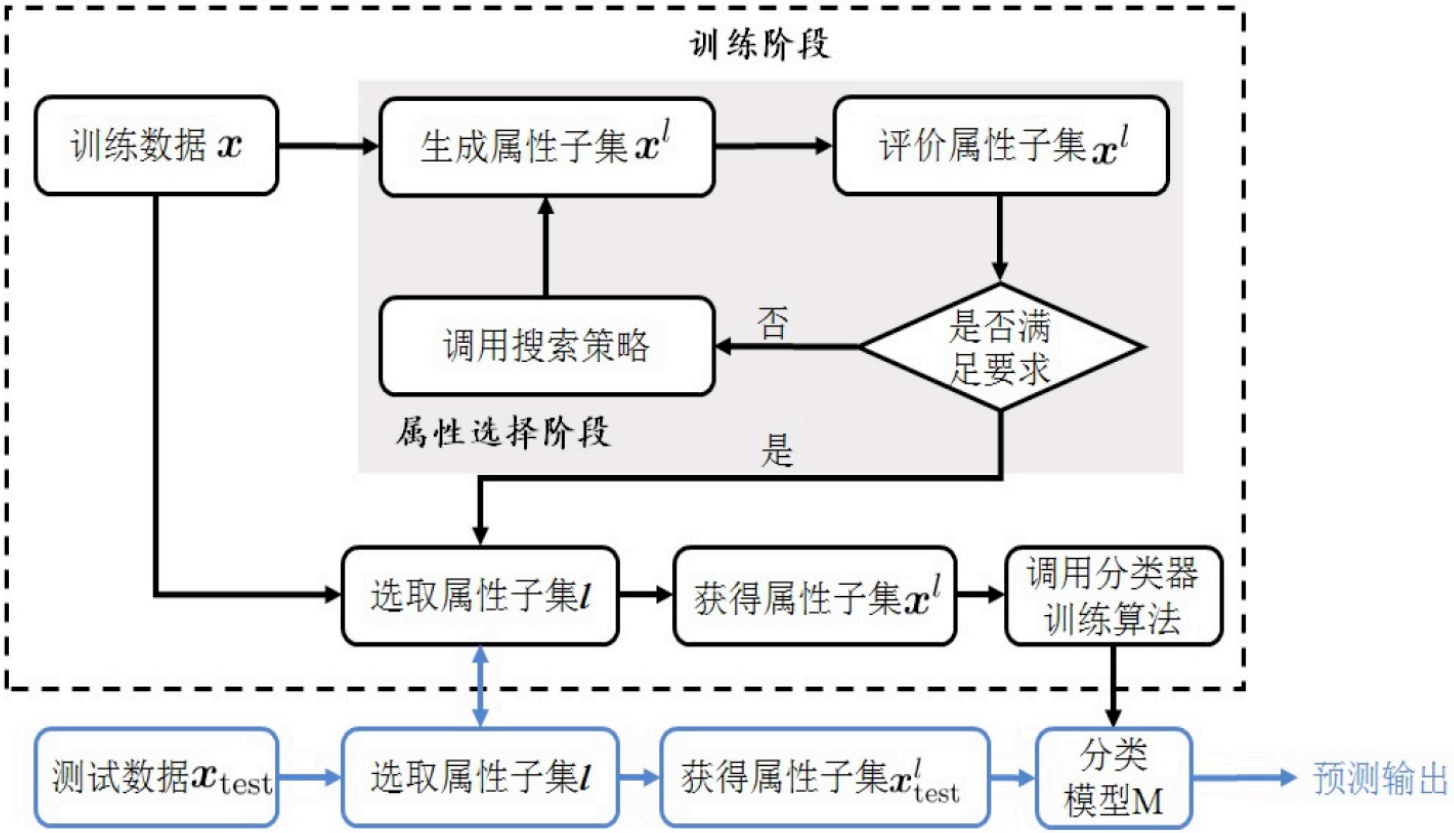


图 1.5　Filter式属性选择的一般流程

Wrapper式

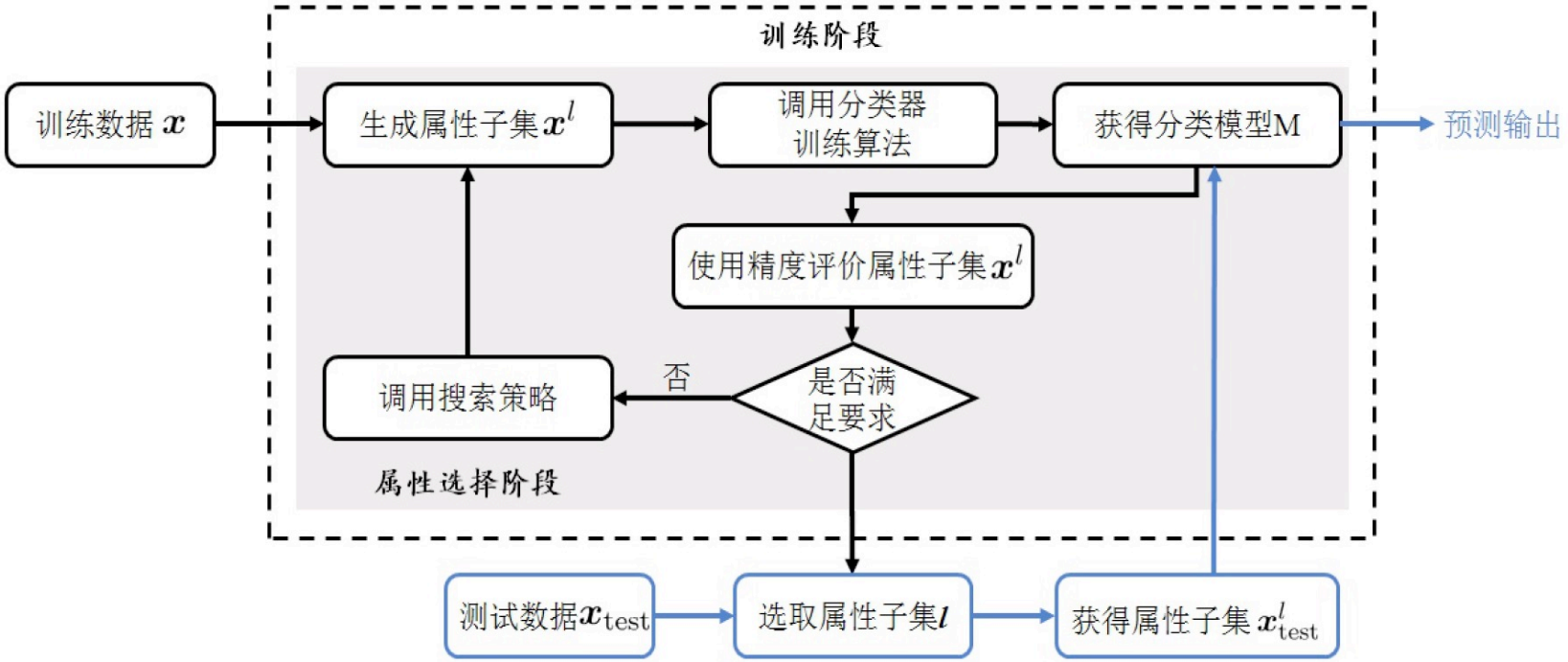


图 1.6　Wrapper式属性选择的一般流程

过滤式选择 - Relief

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) 方法 [Kira and Rendell, 1992]
  - 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
  - 设计一个阈值，然后选择比阈值大的相关统计量分量所对应的特征
  - 或者指定欲选取的特征个数，然后选择相关统计量分量最大的指定个数特征

如何确定相关统计量？

Relief方法中相关统计量的确定

- 猜中近邻（near-hit）：$\boldsymbol{x}_i$的同类样本中的最近邻$\boldsymbol{x}_{i,nh}$
- 猜错近邻（near-miss）：$\boldsymbol{x}_i$的异类样本中的最近邻$\boldsymbol{x}_{i,\mathrm{nm}}$
- 相关统计量对应于属性j的分量为

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

若j为离散型，则$x_a^j, = x_b^j$时$\mathrm{diff}(x_a^j, x_b^j) = 0$，否则为$1$；
若j为连续型，则$\mathrm{diff}(x_a^j, x_b^j) = |x_a^j - x_b^j|$，注意$x_a^j$，$x_b^j$已规范化到[0,1]区间

- 相关统计量越大，属性j上，猜对近邻比猜错近邻越近，即属性j对区分对错越有用
- Relief方法的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

Relief方法的多类拓展

Relief方法工作在二分类问题，其扩展版本Relief-F[Kononenko, 1994]能处理多分类问题

- 数据集中的样本来自$|\mathcal{Y}|$个类别，其中$\boldsymbol{x}_i$属于第k类
- 猜中近邻：第k类中$\boldsymbol{x}_i$的最近邻$\boldsymbol{x}_{i,nh}$
- 猜错近邻：第k类之外的每个类中找到一个$\boldsymbol{x}_i$的最近邻作为猜错近邻，记为$\boldsymbol{x}_{i,l,nm}(l = 1,2,\dots,|\mathcal{Y}|; l \neq k)$
- 相关统计量对应于属性的分量为

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

p_l为第l类样本在数据集D中所占的比例

包裹式选择

包裹式选择把最终使用的学习器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
- 包裹式选择方法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
- 包裹式特征选择过程中需多次训练学习器，计算开销通常比过滤式特征选择大得多

LVW包裹式特征选择方法

LVW（Las Vegas Wrapper）[Liu and Setiono, 1996] 在拉斯维加斯方法框架下使用随机策略来进行子集搜索，并以最终分类器的误差作为特征子集评价准则

基本步骤

- 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，则该算法有可能给不出解

输入： 属性全体上的数据$\mathbf{X}^L$，L为全体属性组成的集合
输入： 最大迭代次数K
输入： 数据标记$\mathbf{Y}$
过程：
1: 初始化err $\leftarrow 0$，$k \leftarrow 0$，$C \leftarrow \mathrm{card}(L)$，$T \leftarrow L$
2: **repeat**
3: 随机获取属性子集L^*，$C^* \leftarrow \mathrm{card}(L^*)$
4: err$^* \leftarrow$ Learner$(\mathbf{X}^{L^*}, \mathbf{Y}$，'CrossValidation'$)$
5: **if** err$^* <$ err or (err* = err and $C^* < C$) **then**
6: $k \leftarrow 0$，err $\leftarrow$ err*，$C \leftarrow C^*$，$T = L^*$
7: **else**
8: $k = k + 1$
9: **end if**
10: **until** err连续K次不更新，即$k = K$
11: **return** T

计算机领域以两个著名的赌城命名了两个著名的随机方法：拉斯维加斯（Las Vegas）方法和蒙特卡洛（Monte Carlo）方法，这两者在效用上区别在于，前者在规定时间内或者获得需要的解或者不输出，后者在规定的时间内可能会得到达不到要求的解。但如果无执行时间限制，两者都能输出满足需求的解。蒙特卡洛方法在图模型一章有介绍。

图 1.7　LVW算法伪代码

嵌入式选择

嵌入式特征选择是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，在学习器训练过程中自动地进行特征选择

- 考虑最简单的线性回归模型，以平方误差为损失函数，并引入L_2范数正则化项防止过拟合，则有

$$\min_{\boldsymbol{w}} \sum_{i=1}^{m} (y_i - \boldsymbol{w}^\top \boldsymbol{x}_i)^2 + \lambda \|\boldsymbol{w}\|_2^2$$

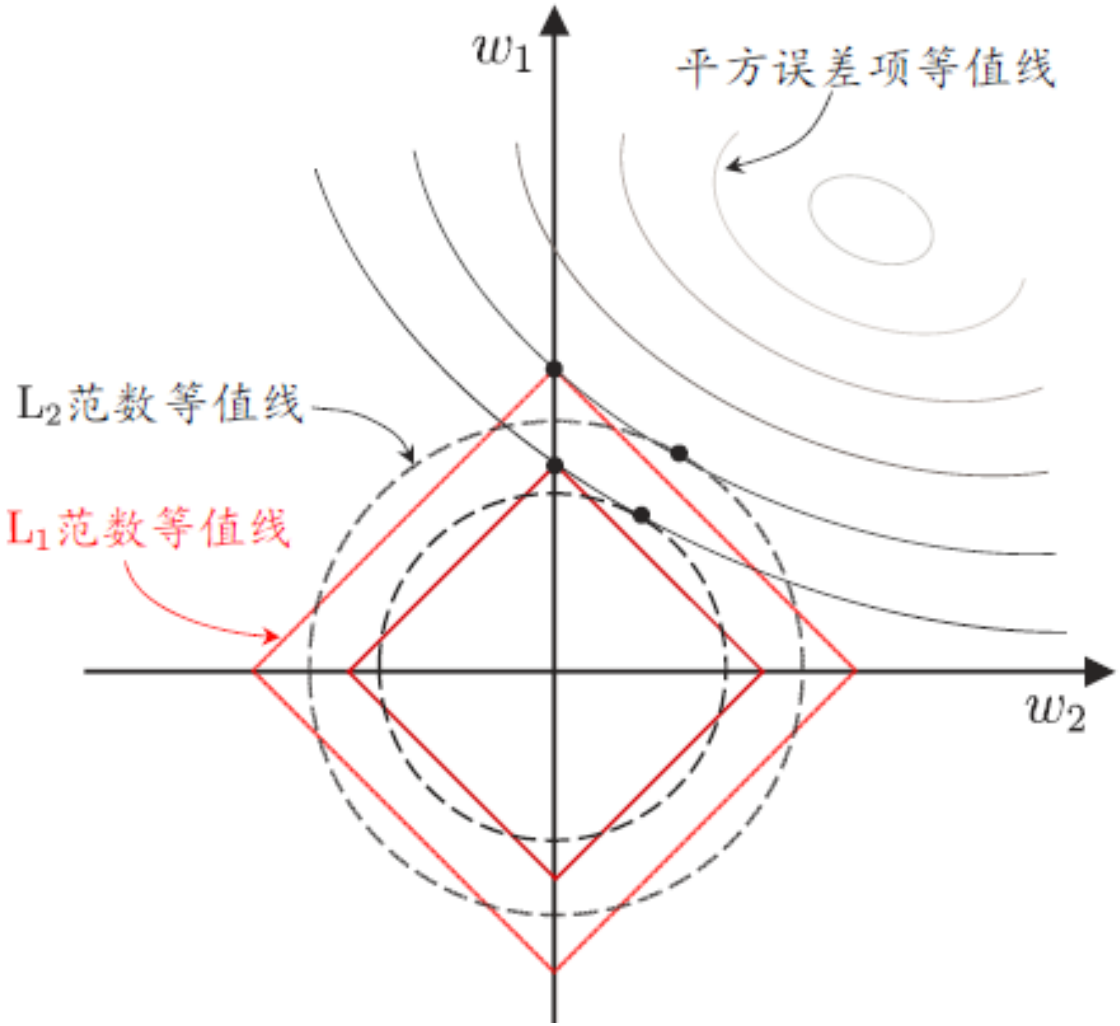
岭回归 (ridge regression)
[Tikhonov and Arsenin, 1977]

- 将L_2范数替换为L_2范数，则有**LASSO** [Tibshirani, 1996]

$$\min_{\boldsymbol{w}} \sum_{i=1}^{m} (y_i - \boldsymbol{w}^\top \boldsymbol{x}_i)^2 + \lambda \|\boldsymbol{w}\|_1$$

易获得稀疏解，是一种嵌入式特征选择方法

使用L_1范数正则化易获得稀疏解



假设$\boldsymbol{x}$仅有两个属性，那么$\boldsymbol{w}$有两个分量w_1和w_2. 那么目标优化的解要在平方误差项与正则化项之间折中，即出现在图中平方误差项等值线与正则化等值线相交处.

从图中看出，采用L_1范数时交点常出现在坐标轴上，即产生w_1或者w_2为0的稀疏解.

等值线即取值相同的点的连线

L_1正则化问题的求解(1)

$$\min_{w\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n}(y_i - \langle w, x_i\rangle)^2 + \lambda\|w\|_1,$$

$$\min_{x} \quad \sum_{i=1}^{m} f(x) + \quad \|x\|_1$$

近端梯度下降（Proximal Gradient Descend，简称PGD）解法

[Boyd and Vandenberghe, 2004]

- 写出$f(\boldsymbol{x})$的二阶泰勒展式

$$f(\boldsymbol{x}) = f(\boldsymbol{x}_k) + \langle \nabla f(\boldsymbol{x}_k), \boldsymbol{x} - x_k\rangle + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_k)^\top \frac{\delta^2 f(\boldsymbol{x}_k)}{\delta \boldsymbol{x}_k^2}(\boldsymbol{x} - \boldsymbol{x}_k)$$

- 假设$f(\boldsymbol{x})$满足L-Lipschitz条件，　即存在常数$L > 0$, 使得

$$\|\nabla f(\boldsymbol{x}') - \nabla f(\boldsymbol{x})\| \leq L\|\boldsymbol{x}' - \boldsymbol{x}\|_2^2$$

L1正则化问题的求解(2)

- L-Lipschitz条件代入泰勒展式，可得

$$f(\boldsymbol{x}) \cong f(\boldsymbol{x}_k) + \langle \nabla f(\boldsymbol{x}_k), \boldsymbol{x} - \boldsymbol{x}_k \rangle + \frac{L}{2}\|\boldsymbol{x} - \boldsymbol{x}_k\|^2$$
$$= \frac{L}{2}\|\boldsymbol{x} - (\boldsymbol{x}_k - \frac{1}{L}\nabla f(\boldsymbol{x}_k))\|^2 + \text{const}$$

- 将上式关于$f(\boldsymbol{x})$的近似代入到原优化问题中，得

$$\min_{\boldsymbol{x}} \sum_{i=1}^{m} \frac{L}{2}\|\boldsymbol{x} - (\boldsymbol{x}_k - \frac{1}{L}\nabla f(\boldsymbol{x}_k))\|^2 + \lambda\|\boldsymbol{x}\|_1$$

L1正则化问题的求解(3)

- 每次在$\boldsymbol{x}_k$的附近寻找最优点，不断迭代，即寻找

$$\boldsymbol{x}_{k+1} = \min_{\boldsymbol{x}} \sum_{i=1}^{m} \frac{L}{2} \|\boldsymbol{x} - (\boldsymbol{x}_k - \frac{1}{L}\nabla f(\boldsymbol{x}_k))\|^2 + \lambda \|\boldsymbol{x}\|_1$$

- 假设$\boldsymbol{z} = \boldsymbol{x}_k - 1/L\nabla f(\boldsymbol{x}_k)$，上式有闭式解

$$x_{k+1;i} = \begin{cases} z_i & =L \quad z_i > \quad =L \\ 0 & \quad jz_ij \quad =L \\ z_i + & =L \quad z_i < \quad =L \end{cases}$$

$$\left(\text{prox}_{\gamma R}(x)\right)_i = \begin{cases} x_i - \gamma, & x_i > \gamma \\ 0, & |x_i| \le \gamma \\ x_i + \gamma, & x_i < -\gamma, \end{cases}$$

嵌入式选择

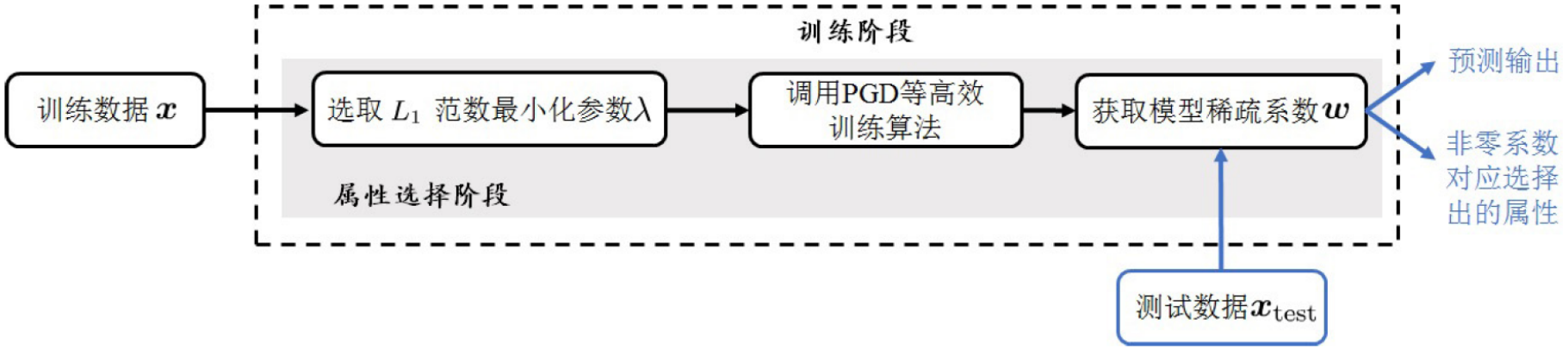


图 1.9　嵌入式属性选择的一般流程

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
- 稀疏表达的优势：
  - 文本数据线性可分
  - 存储高效

能否将稠密表示的数据集转化为“稀疏表示”，使其享受稀疏表达的优势？

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$\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$，$\boldsymbol{x}_i \in \mathbb{R}^{n\times k}$
- 学习目标是字典矩阵$\mathbf{B} \in \mathbb{R}^{d\times k}$以及样本的稀疏表示$\boldsymbol{\alpha}_i \in \mathbb{R}^k$
- 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

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

矩阵补全

客户对书籍的喜好程度的评分

	《笑傲江湖》	《万历十五年》	《人间词话》	《云海玉弓缘》	《人类的故事》
赵大	5	?	?	3	2
钱二	?	5	3	?	5
孙三	5	3	?	?	?
李四	3	?	5	4	?

能否将表中已经通过读者评价得到的数据当作**部分信号**，基于压缩感知的思想**恢复**出**完整信号**从而进行书籍推荐呢？从题材、作者、装帧等角度看（相似题材的书籍有相似的读者），表中反映的信号是**稀疏**的，能通过类似压缩感知的思想加以处理。

“矩阵补全”技术解决此类问题

矩阵补全的优化问题和解法

- 矩阵补全（matrix completion）技术的优化形式为

$$\min_{\mathbf{X}} \operatorname{rank}(\mathbf{X})$$
$$\text{s.t. } \mathbf{X}_{ij} = \mathbf{A}_{ij}, (i,j) \in \Omega$$

约束表明，恢复出的矩阵中$\mathbf{X}_{ij}$应当与已观测到的对应元素相同

- $\mathbf{X}$: 需要恢复的稀疏信号
- rank($\mathbf{X}$): $\mathbf{X}$的秩
- $\mathbf{A}$: 已观测信号
- Ω: $\mathbf{A}$中已观测到的元素的位置下标的集合

- NP难问题. 将rank($\mathbf{X}$)转化为其凸包“核范数”（nuclear norm）

$$\min_{X} \|X\|_*$$
$$\text{s.t. } X_{ij} = A_{ij}; (i;j)\ 2$$

$\|X\|_* = \sum_{j=1}^{\min(m;n)} {}_j(X)$，${}_j(X)$表示$\mathbf{X}$的奇异值，即矩阵的核范数为矩阵的奇异值之和.

- 凸优化问题，通过半正定规划求解（SDP，Semi-Definite Programming）

满足一定条件时，只需观察到$O(mr\log^2 m)$个元素就能完美恢复$\mathbf{A}$
[Recht, 2011]

小结

- 特征选择
  - 定义，动机
  - 常见的方法和原理
- 稀疏表示
  - 动机，基本方法的原理