



高级机器学习

特征选择与稀疏学习



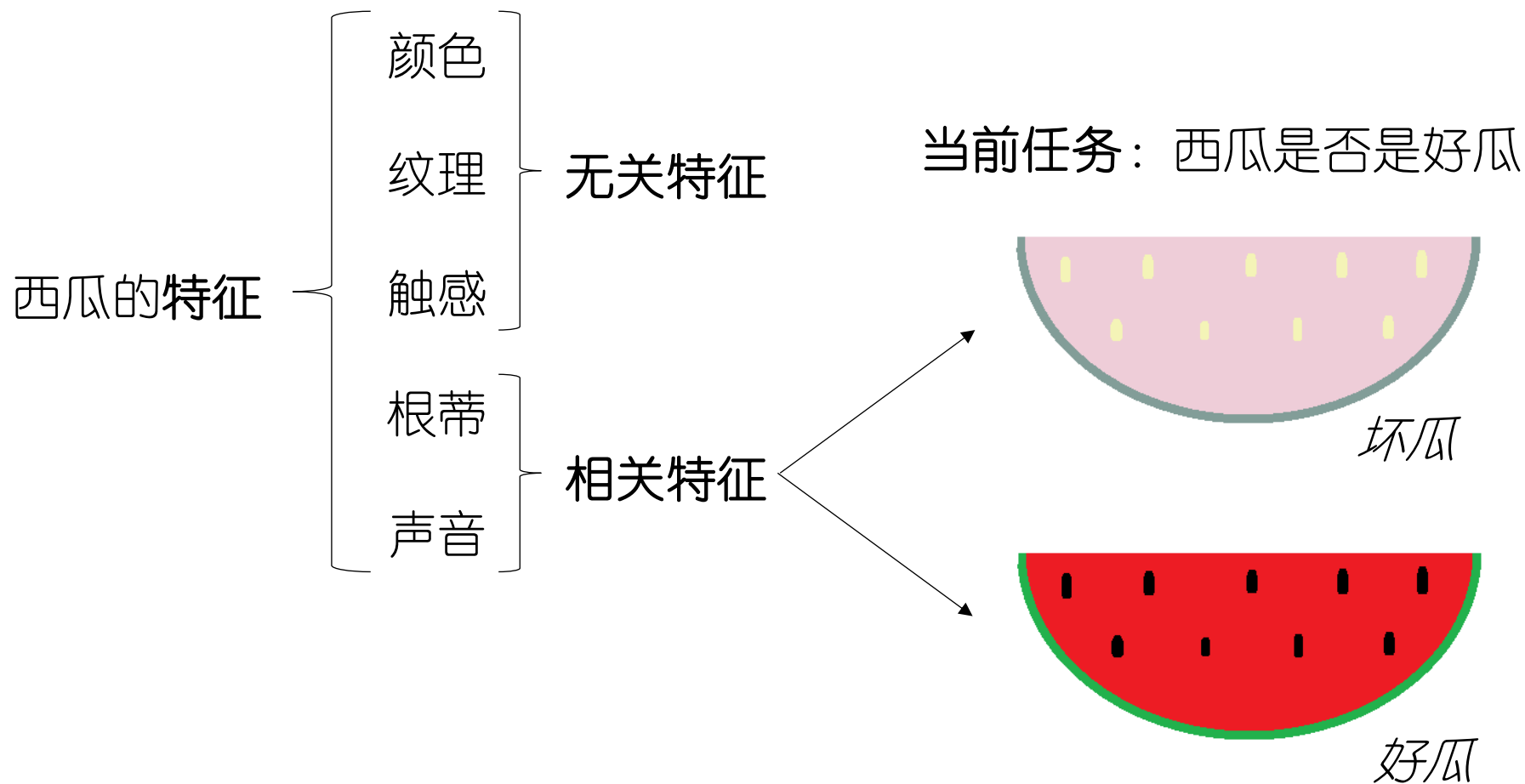
特征

□ 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



特征选择

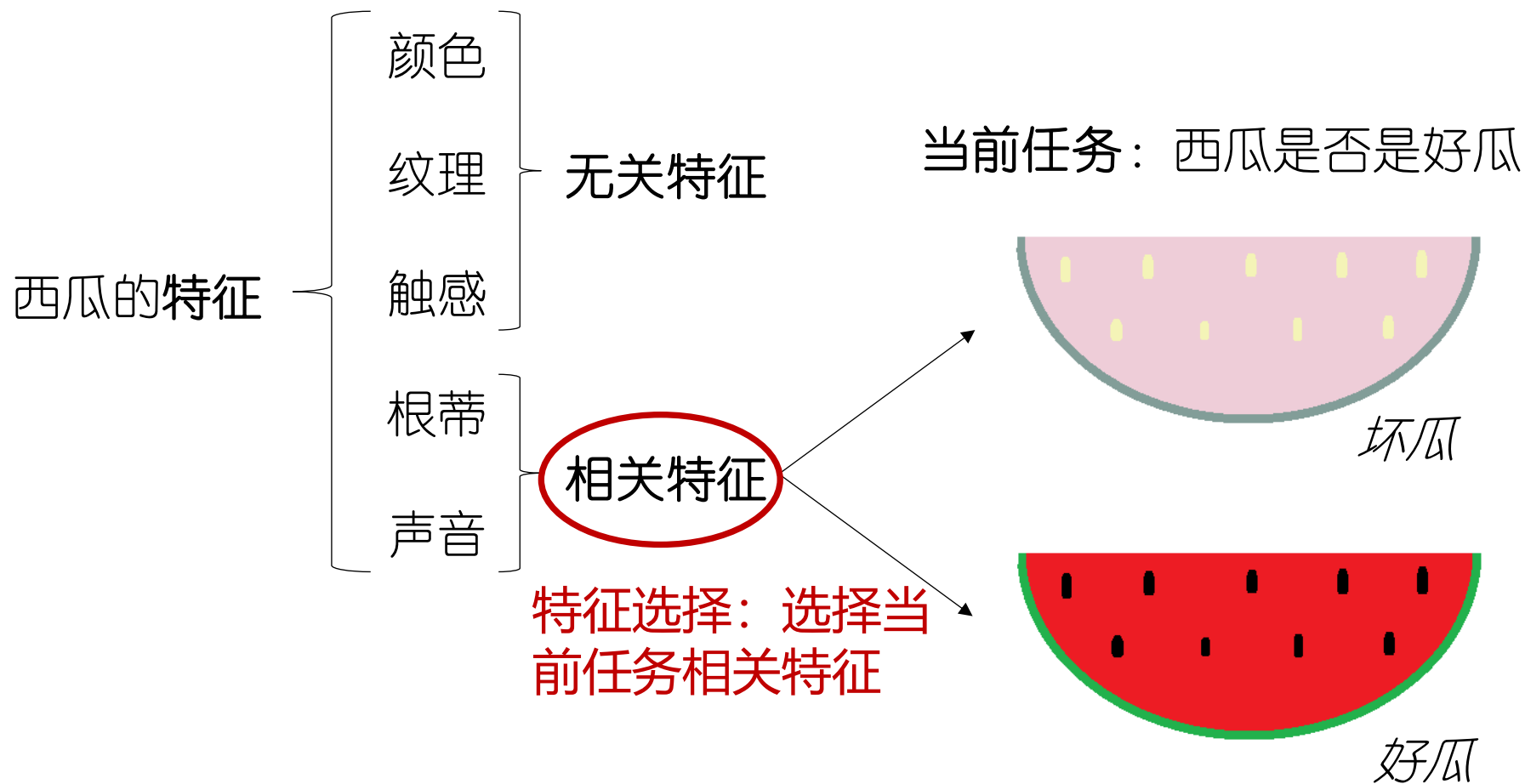
□ “特征选择” 应运而生，其目的主要在于

- 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：判断西瓜好坏时的特征选择

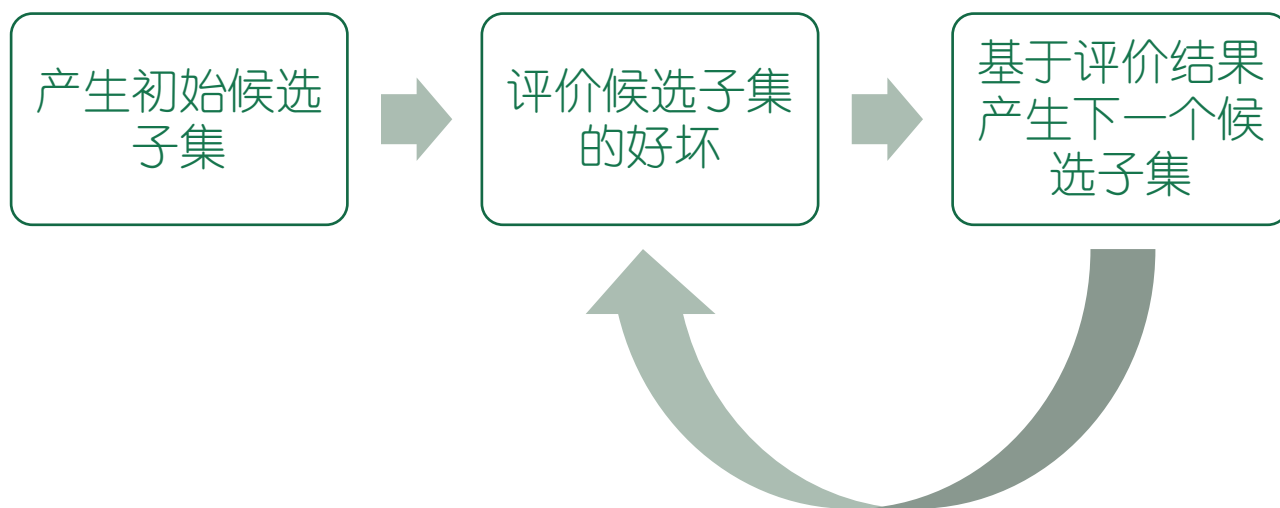


特征选择的一般方法

❑ 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

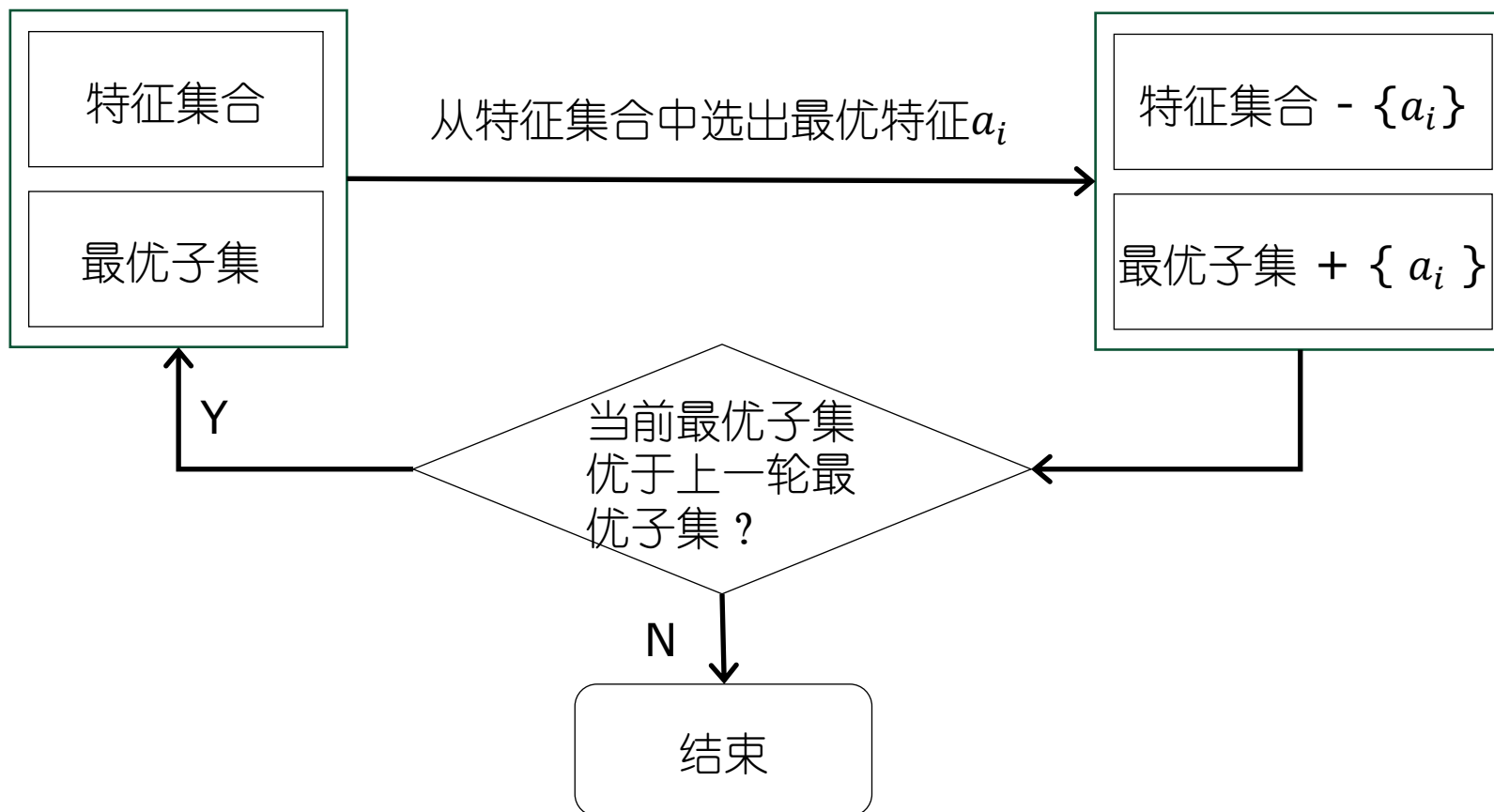
子集搜索

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

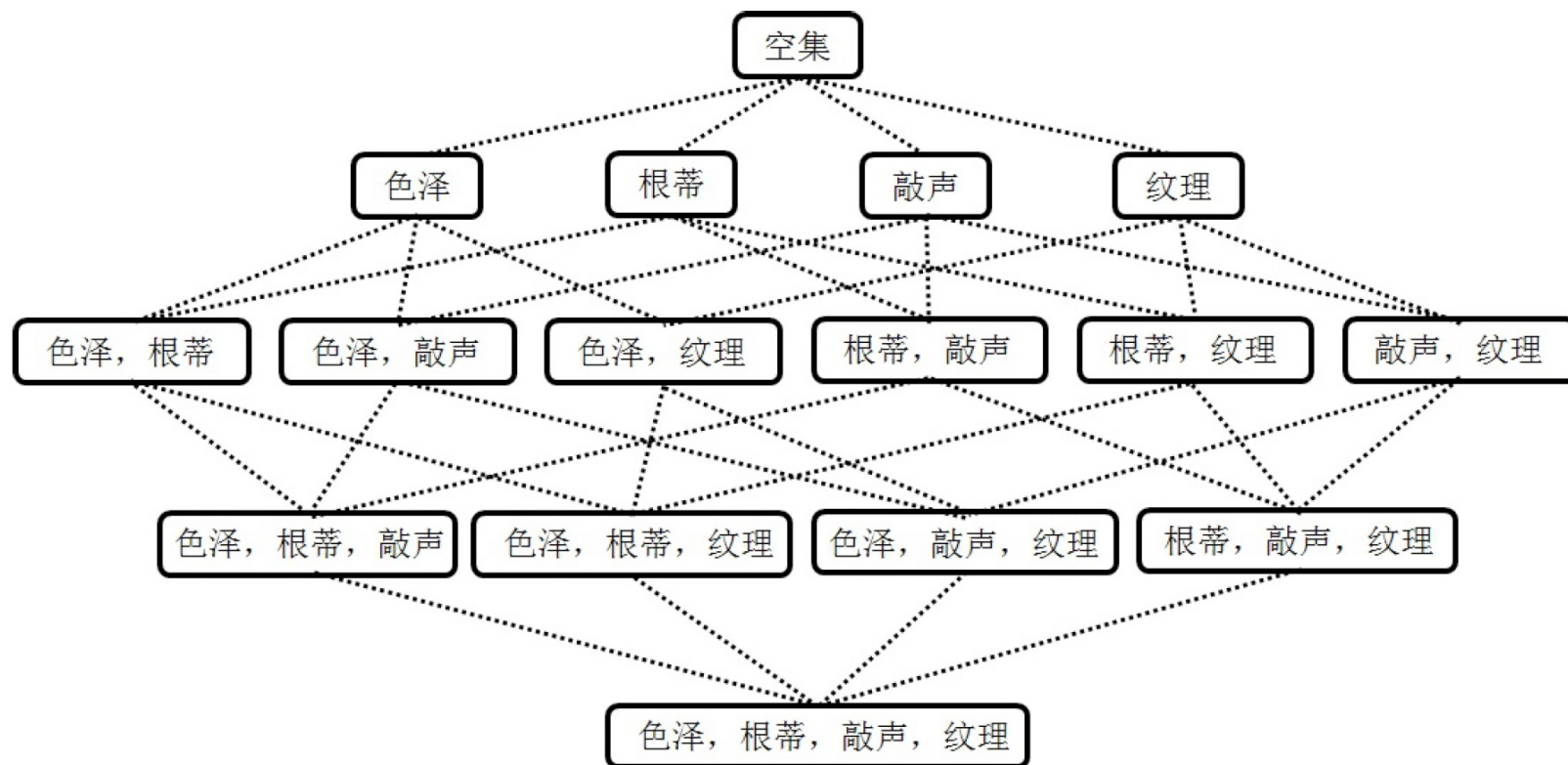


图 1.1 由4个属性及在属性子集上的交并运算定义的格

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
 - $\text{Ent}(D^v)$ 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
 - $\text{Ent}(D)$ 表示 D 上的信息熵

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

特征子集 A 的信息增益为：

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

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：

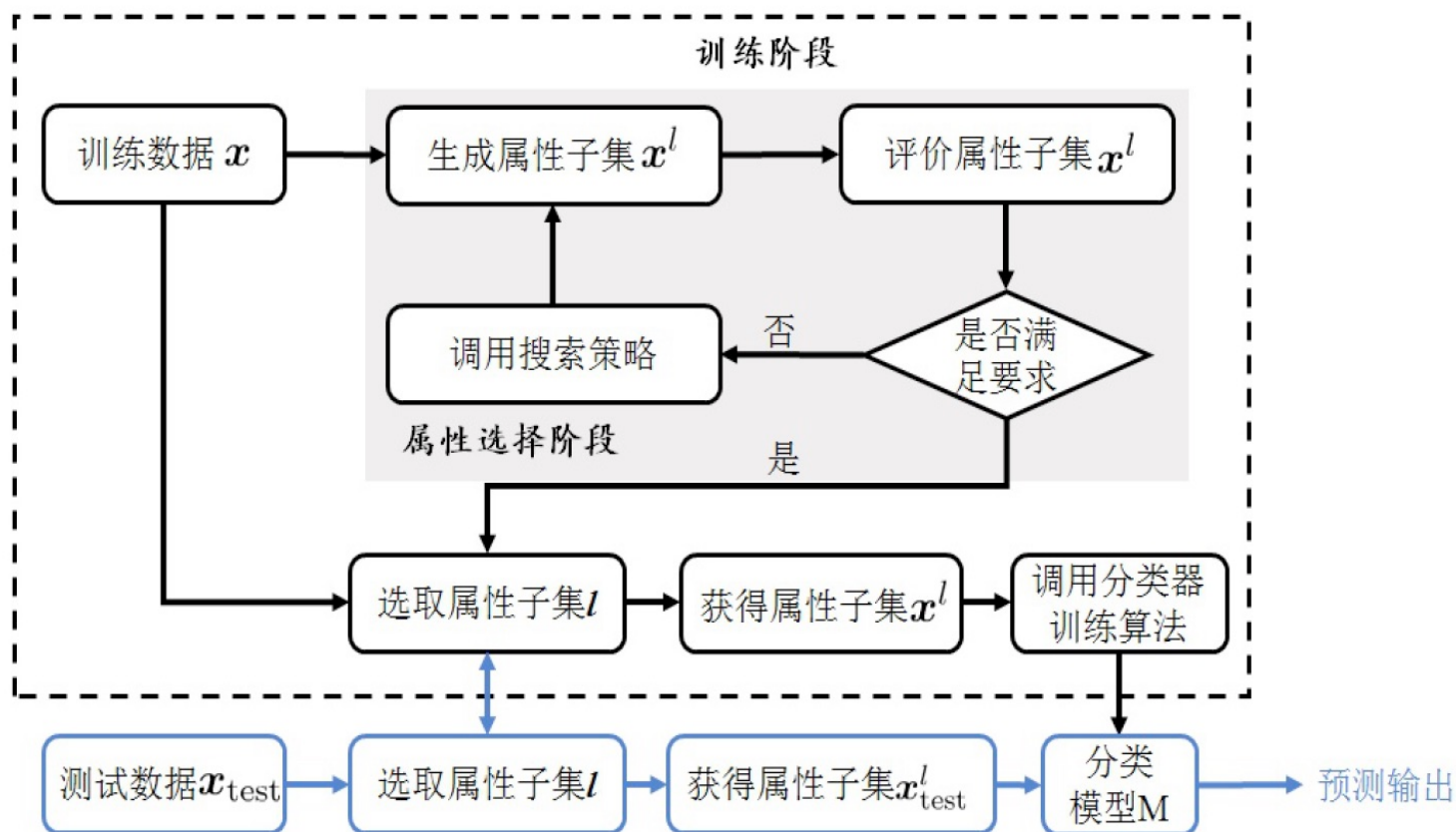
- 过滤式

- 包裹式

- 嵌入式

过滤(Filter)式选择

先对数据集进行特征选择，再训练学习器



过滤(Filter)式选择

□ 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) : $\mathbf{x}_i$ 的同类样本中的最近邻 $\mathbf{x}_{i,nh}$

□ 猜错近邻 (near-miss) : $\mathbf{x}_i$ 的异类样本中的最近邻 $\mathbf{x}_{i,nm}$

□ 相关统计量对应于属性 j 的分量为

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

若 j 为离散型, 则 $x_a^j = x_b^j$ 时 $\text{diff}(x_a^j, x_b^j) = 0$, 否则为 1 ;
若 j 为连续型, 则 $\text{diff}(x_a^j, x_b^j) = |x_a^j - x_b^j|$, 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方法的多分类扩展

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 $\mathbf{x}_i$ 属于第 k 类
- 猜中近邻：第 k 类中 $\mathbf{x}_i$ 的最近邻 $\mathbf{x}_{i,nh}$
- 猜错近邻：第 k 类之外的每个类中找到一个 $\mathbf{x}_i$ 的最近邻作为猜错近邻，记为 $\mathbf{x}_{i,l,nm} (l = 1, 2, \dots, |\mathcal{Y}|; l \neq k)$

- 相关统计量对应于属性的分量为

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

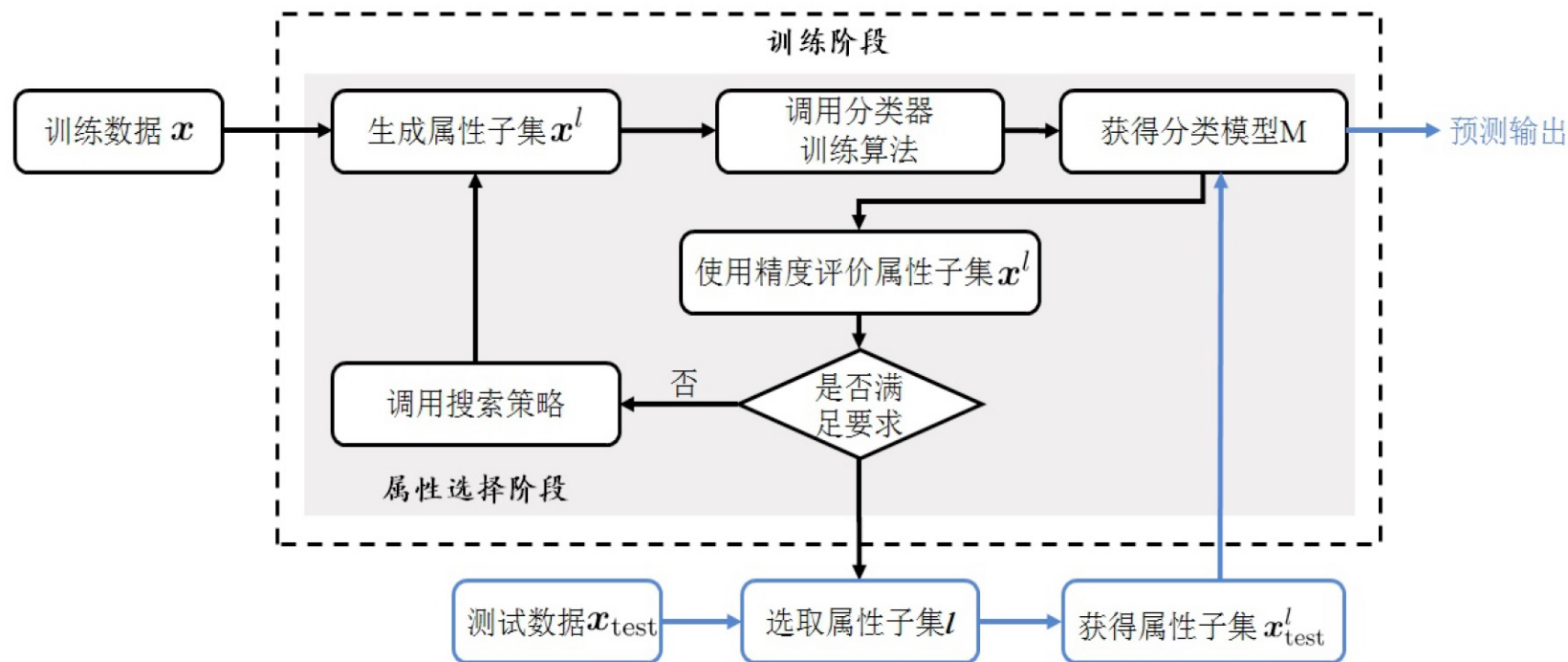
p_l 为第 l 类样本在数据集 D 中所占的比例

包裹式(Wrapper)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
-

包裹式(Wrapper)选择



包裹式(Wrapper)选择

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

包裹式(Wrapper)选择

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 $\text{err} \leftarrow 0$, $k \leftarrow 0$, $C \leftarrow \text{card}(L)$, $T \leftarrow L$
 - 2: **repeat**
 - 3: 随机获取属性子集 L^* , $C^* \leftarrow \text{card}(L^*)$
 - 4: $\text{err}^* \leftarrow \text{Learner}(\mathbf{X}^{L^*}, \mathbf{Y}, \text{'CrossValidation'})$
 - 5: **if** $\text{err}^* < \text{err}$ or $(\text{err}^* = \text{err} \text{ and } C^* < C)$ **then**
 - 6: $k \leftarrow 0$, $\text{err} \leftarrow \text{err}^*$, $C \leftarrow C^*$, $T = L^*$
 - 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
-

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

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

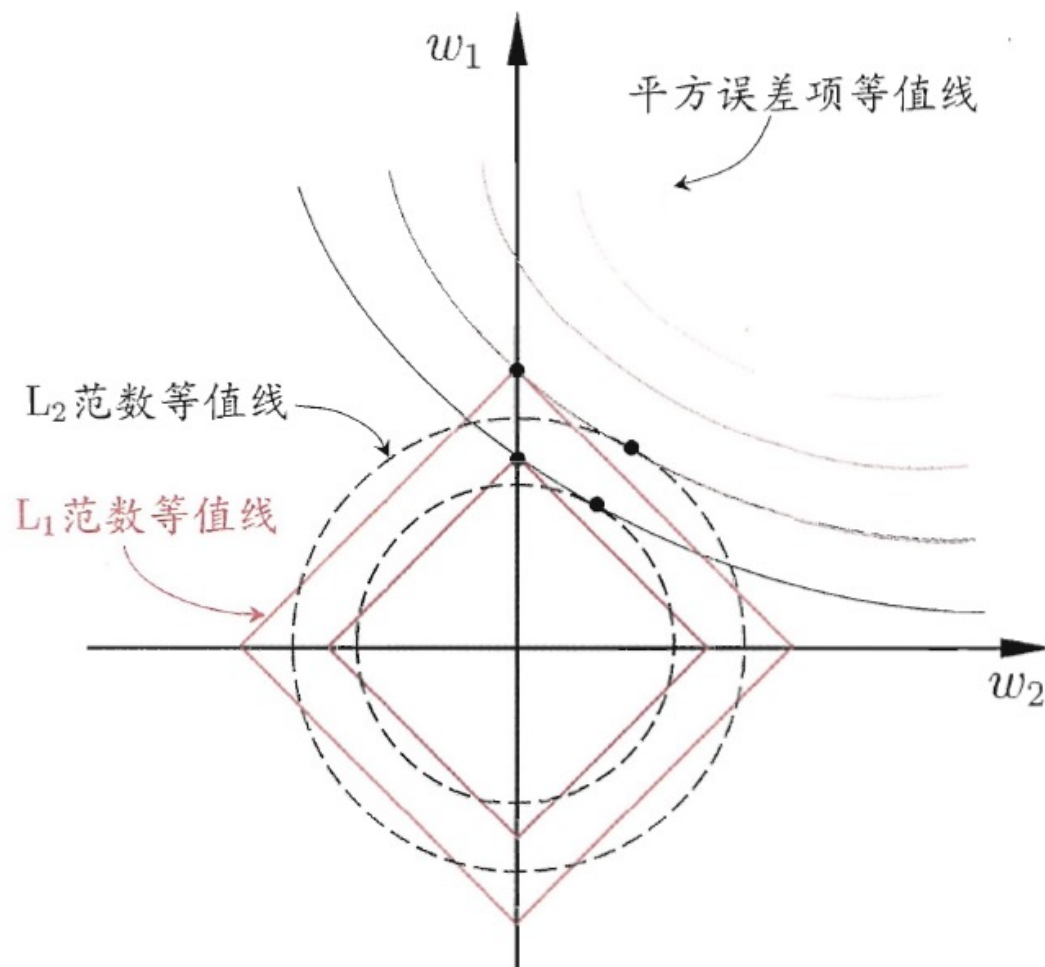
岭回归 (ridge regression)
[Tikhonov and Arsenin, 1977]

- 将 L_2 范数替换为 L_1 范数，则有**LASSO** [Tibshirani, 1996]

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

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 $\mathbf{x}$ 仅有两个属性, 那么 $\mathbf{w}$ 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

嵌入式选择

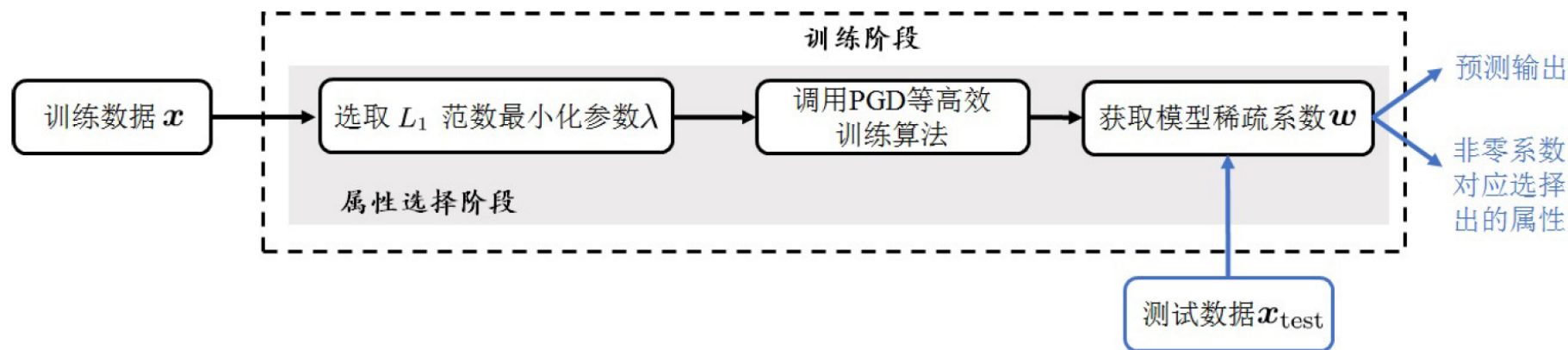


图 1.9 嵌入式属性选择的一般流程

稀疏表示

- 特征选择考虑的问题是特征具有“稀疏性”，即矩阵中的许多列与当前学习任务无关，通过特征选择去除这些列，模型训练过程仅需在较小的矩阵上进行
 - 考虑另一种稀疏性：矩阵中存在许多零元素，但这些零元素并非以整行、整列形式存在的
-

稀疏表示

- 例如，文档分类任务，每个文档看做一个样本，每个字（词）看做一个特征，字（词）在文档中出现的次数看做特征的取值
 - 数据集对应的矩阵为：每行是一个文档，每列是一个字（词），行、列交汇处就是某字（词）在某文档中出现的次数
 - 矩阵由多少列呢？《康熙字典》47035个汉字，《现代汉语常用字表》3500个汉字。
 - 给定一个文档，相当多的字并没有在文档中出现，矩阵的每一行都有大量的零元素，对于不同的文档，零元素出现的列往往很不相同
-

稀疏表示

□ 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（稀疏编码）

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 $\{x_1, x_2, \dots, x_m\}$, $x_i \in \mathbb{R}^{d \times k}$
- 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 $\alpha_i \in \mathbb{R}^k$
- 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

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

字典学习（稀疏编码）

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

□ 固定 $\mathbf{B}$ 优化 $\boldsymbol{\alpha}$

$$\min_{\boldsymbol{\alpha}_i} \|\mathbf{x}_i - \mathbf{B}\boldsymbol{\alpha}_i\|_2^2 + \lambda \|\boldsymbol{\alpha}_i\|_1$$

□ 固定 $\boldsymbol{\alpha}$ 优化 $\mathbf{B}$

$$\min_{\mathbf{B}} \|\mathbf{X} - \mathbf{B}\mathbf{A}\|_F^2$$

压缩感知

如何根据稀疏性从少量观测中复原信号，根据部分信息恢复全部信息

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？

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

$$\min_{\mathbf{X}} \text{rank}(\mathbf{X})$$

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

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
- $\text{rank}(\mathbf{X})$: $\mathbf{X}$ 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 $\text{rank}(\mathbf{X})$ 转化为其凸包“核范数” (nuclear norm)

$$\min_{\mathbf{X}} \|\mathbf{X}\|_*$$

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

$$\|\mathbf{X}\|_* = \sum_{j=1}^{\min\{m,n\}} \sigma_j(\mathbf{X})$$

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
 - 过滤式
 - 包裹式
 - 嵌入式（L1正则化易获得稀疏解）
 - 稀疏表示
 - 字典学习（交替迭代优化）
 - 压缩感知（矩阵补全）
-