周志华 著

MACHINE LEARNING

机器学习

清华大学出版社

本章课件致谢：
徐淼

第十一章：特征选择

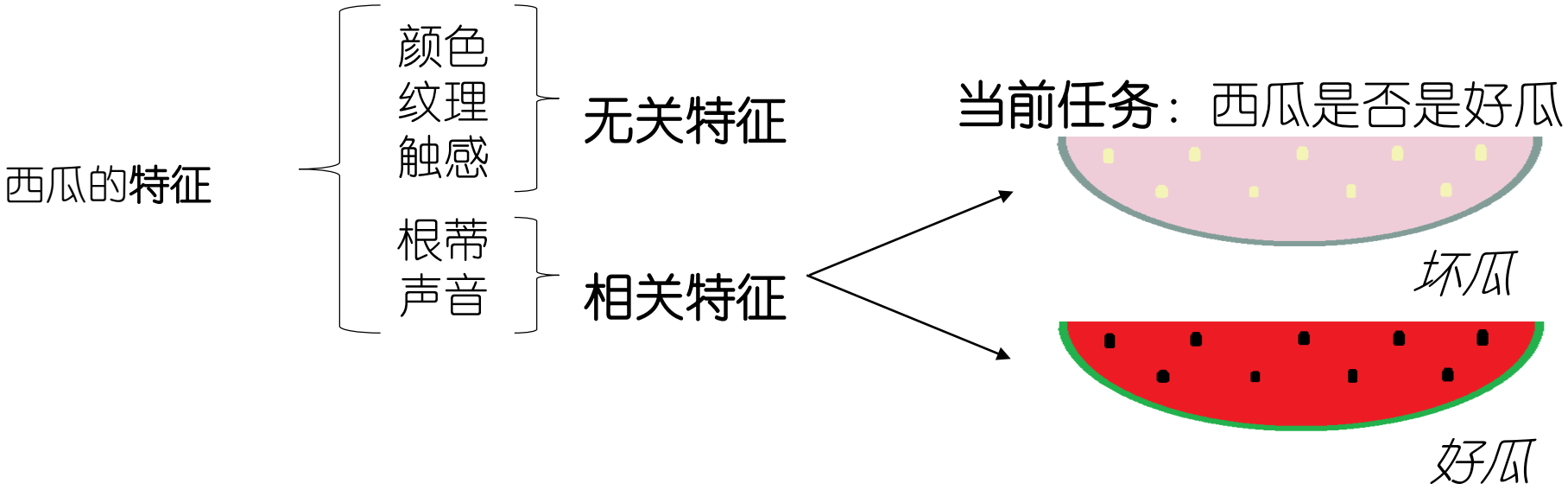
目录

- 特征选择：
  - 子集搜索与评价
  - 过滤式选择
  - 包裹式选择
  - 嵌入式选择

特征

- 特征
  - 描述物体的属性

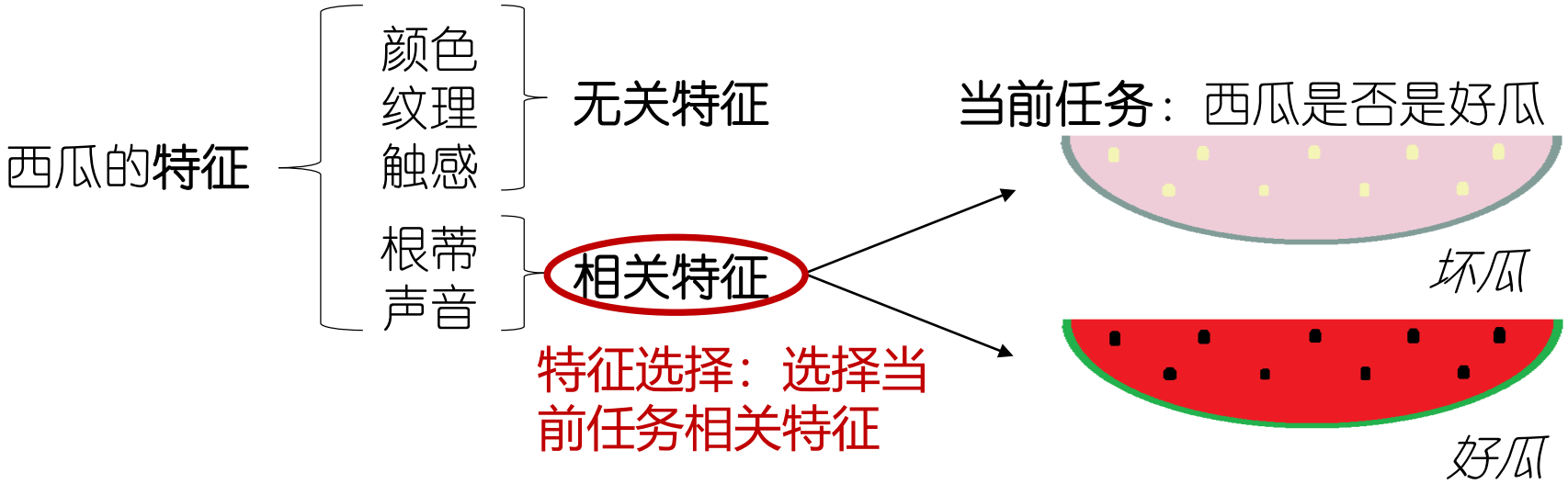
- 特征的分类
  - 相关特征：对当前学习任务有用的属性
  - 无关特征：与当前学习任务无关的属性



特征选择

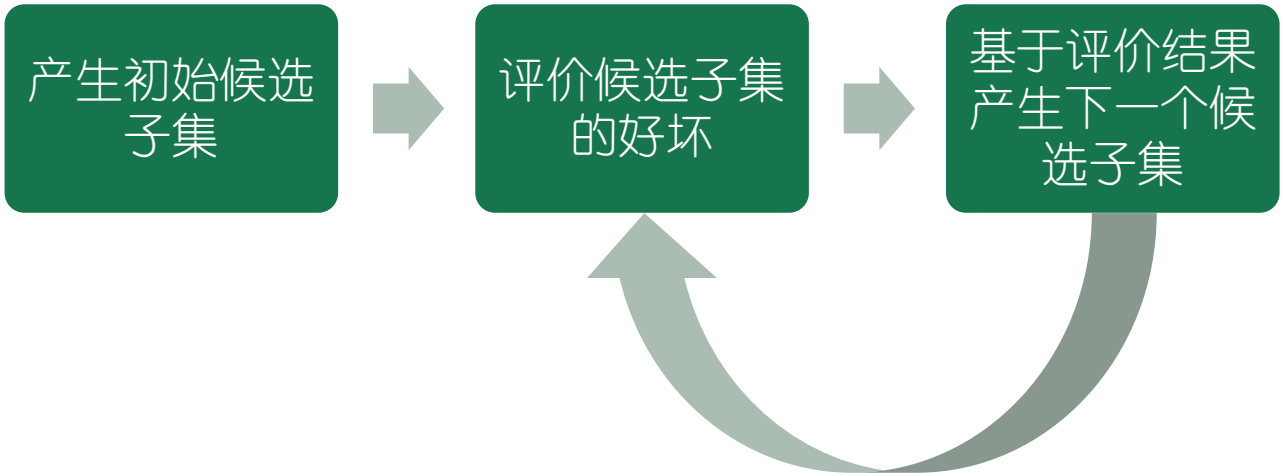
- 特征选择
  - 从给定的特征集合中选出**任务相关**特征子集
  - 必须确保不丢失重要特征

- 原因
  - 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



特征选择的一般方法

- 遍历所有可能的子集
  - 计算上遭遇组合爆炸，不可行
- 可行方法

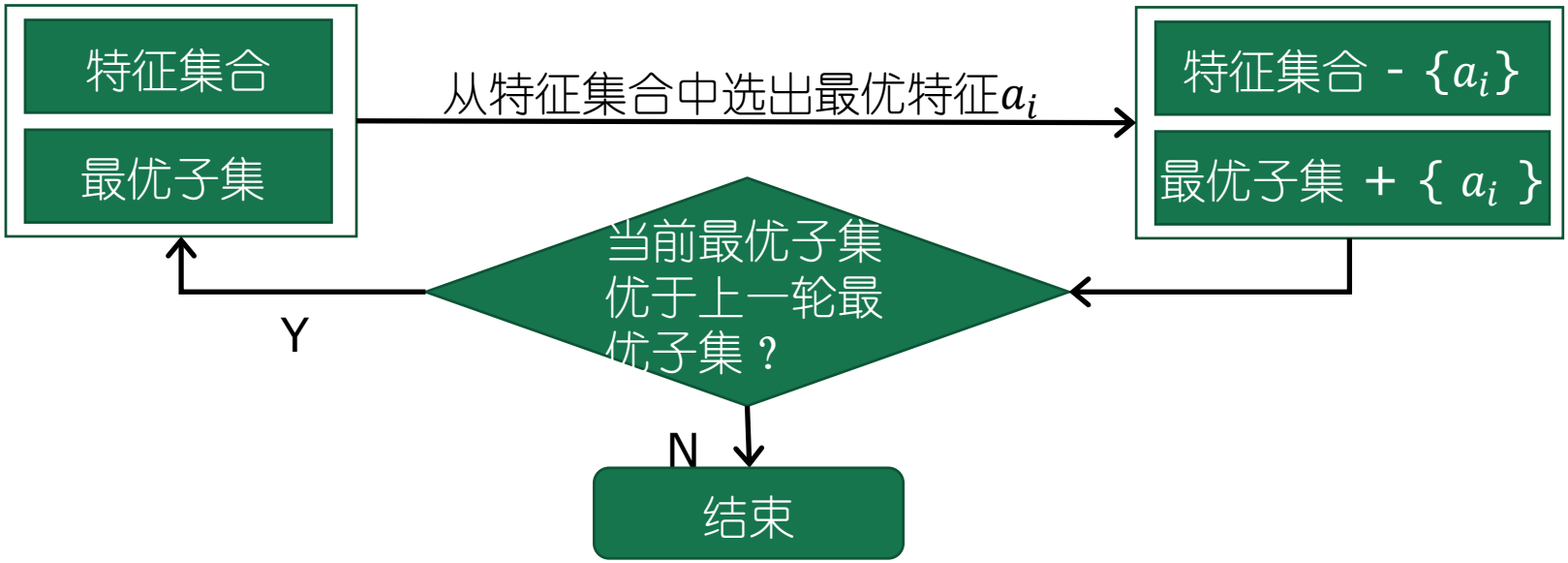


两个关键环节：子集搜索和子集评价

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：最优子集初始为空集，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

- 1、把每个特征看作一个候选子集，特征集{a_1，a_2，a_3 ...a_d}
- 2、对单个特征子集进行评价，选定{a_2}最优。
- 3、将{a_2}作为第一轮选定集，然后在上一轮选定集中加一特征，构成两特征候选子集，在这d-1个候选子集中选出{a_2，a_4 }最优，且优于{a_2}，则将{a_2，a_4}作为本轮选定集。
- 4、重复上述步骤，不断添加特征子集，直到当前最优子集不再优于上一轮的选定集，则停止搜索。

后向搜索

- 1、把每个特征看作一个候选子集，特征集{a_1，a_2，a_3 ...a_d}
- 2、从特征集全集出发。
- 3、第一次拿出一个识别率最低的属性，然后依次拿出属性直到拿出一个属性后该属性集合的性能不升高反而下降，就停止搜索。

双向搜索

- 算法描述：使用序列前向选择(SFS)与序列后向选择(SBS)分别从两端开始搜索，两者搜索到一个相同的特征子集Y才停止搜索。
- 为了确保序列前向选择与序列后向选择会搜索到相同的子集，需要确保：

  (1) 被SFS选中的特征SBS就不能去除

  (2) 被SBS去除的特征SFS就不能选择
- 算法评价： BDS结合了SFS与SBS，其时间复杂度比SFS与SBS小，但是兼有SFS与SBS的缺点。

双向搜索

- 算法流程：

1. Start SFS with the empty set $Y_F=\{\varnothing\}$
2. Start SBS with the full set $Y_B=X$
3. Select the best feature

$$x^+ = \underset{\substack{x\notin Y_{F_k}\\ x\in Y_{B_k}}}{\arg\max}\left[J\left(Y_{F_k}+x\right)\right]$$

$$Y_{F_{k+1}} = Y_{F_k} + x^+$$

3. Remove the worst feature

$$x^- = \underset{\substack{x\in Y_{B_k}\\ x\notin Y_{F_{k+1}}}}{\arg\max}\left[J\left(Y_{B_k}-x\right)\right]$$

$$Y_{B_{k+1}} = Y_{B_k} - x^-; \quad k = k+1$$

4. Go to 2

子集评价

- 特征子集A 确定了对数据集D的一个划分
  - 每个划分区域对应着特征子集A的某种取值
- 样本标记Y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
- 通过计算属性子集的信息增益，评价子集的好坏。
- 信息增益：

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

- 信息熵：

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

基于评价准则划分特征选择方法

将特征子集搜索机制与子集评价机制相结合，即可得到特征选择方法

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

基于评价准则划分特征选择方法

将特征子集搜索机制与子集评价机制相结合，即可得到特征选择方法

常见的特征选择方法大致分为如下三类：

- 过滤式

  先对数据集进行特征选择，然后再训练学习器，特征选择过程与后续学习器无关。

  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。

- 包裹式

- 嵌入式

过滤式选择-- Relief算法

- Relief (Relevant Features) 方法是一种著名的过滤式特征选择方法。

  - Relief算法最早由Kira提出，最初局限于两类数据的分类问题。
  - Relief算法是一种特征权重算法(Feature weighting algorithms)，根据各个特征和类别的相关性赋予特征不同的权重（相关统计量），权重小于某个阈值的特征将被移除。
  - Relief算法中特征和类别的相关性是基于特征对近距离样本的区分能力。
  - Relief的关键是如何确定权重（相关统计量）？

过滤式选择-- Relief算法

- Relief (Relevant Features) 方法是一种著名的过滤式特征选择方法。

  - Relief算法算法从训练集D中随机选择一个样本$\boldsymbol{x}_i$，然后
    - 从和$\boldsymbol{x}_i$同类的样本中寻找最近邻样本，称为**猜中近邻**（near-hit）
    - 从和 $\boldsymbol{x}_i$ 不同类的样本中寻找最近邻样本，称为**猜错近邻**（near-miss）
  - 然后根据以下规则更新每个特征的权重：
    - 如果$\boldsymbol{x}_i$和猜中近邻在某个特征上的距离小于$\boldsymbol{x}_i$和猜错近邻上的距离，则说明该特征对区分同类和不同类的最近邻是有益的，则增加该特征的权重；
    - 反之，如果$\boldsymbol{x}_i$和猜中近邻在某个特征的距离大于$\boldsymbol{x}_i$和猜错近邻上的距离，说明该特征对区分同类和不同类的最近邻起负面作用，则降低该特征的权重。
  - 以上过程重复m次，最后得到各特征的平均权重。

  - 特征的权重越大，表示该特征的分类能力越强，反之，表示该特征分类能力越弱。
  - 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。

过滤式选择-- Relief算法的多类拓展

- Relief算法比较简单，但运行效率高，并且结果也比较令人满意，因此得到广泛应用，但是其局限性在于只能处理两类别数据

- 1994年Kononeill进行了扩展，得到了ReliefF作算法，可以处理多类别问题，用于处理目标属性为连续值的回归问题。

  - ReliefF算法在处理多类问题时，每次从训练样本集中随机取出一个样本$\boldsymbol{x}_i$
    - 从和$\boldsymbol{x}_i$同类的样本集中找出$\boldsymbol{x}_i$的 个**猜中近邻样本**
    - 从每个$\boldsymbol{x}_i$的不同类的样本集中均找出 个**猜错近邻样本**

  - 然后，更新每个特征的权重

过滤式选择-- Relief算法的多类拓展

Relief 是为二分类问题设计的, 其扩展变体 Relief-F [Kononenko, 1994] 能处理多分类问题. 假定数据集 D 中的样本来自 $|\mathcal{Y}|$ 个类别. 对示例 $\boldsymbol{x}_i$, 若它属于第 k 类 ($k \in \{1, 2, \ldots, |\mathcal{Y}|\}$), 则 Relief-F 先在第 k 类的样本中寻找 x_i 的最近邻示例 $\boldsymbol{x}_{i,\mathrm{nh}}$ 并将其作为猜中近邻, 然后在第 k 类之外的每个类中找到一个 $\boldsymbol{x}_i$ 的最近邻示例作为猜错近邻, 记为 $\boldsymbol{x}_{i,l,\mathrm{nm}}$ ($l = 1, 2, \ldots, |\mathcal{Y}|;\ l \neq k$). 于是, 相关统计量对应于属性 j 的分量为

$$\delta^j = \sum_i -\mathrm{diff}(x_i^j, x_{i,\mathrm{nh}}^j)^2 + \sum_{l \neq k} \left(p_l \times \mathrm{diff}(x_i^j, x_{i,l,\mathrm{nm}}^j)^2 \right), \tag{11.4}$$

其中 p_l 为第 l 类样本在数据集 D 中所占的比例.

过滤式选择-- 医学数据分析实例

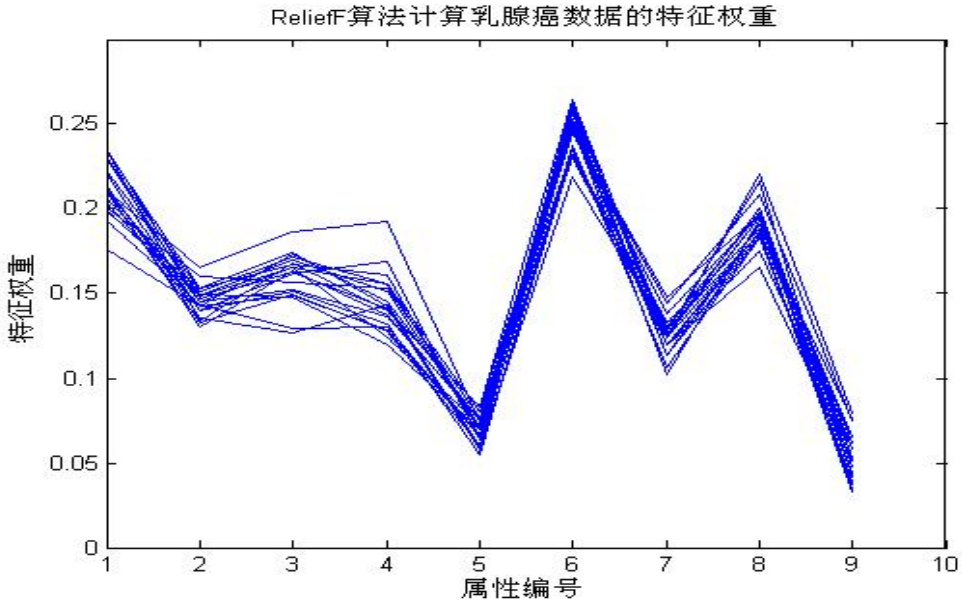
- 选用的数据：威斯康星州乳腺癌数据集，数据来源美国威斯康星大学医院的临床病例报告，每条数据具有9个属性。

属性名称	说明	特征编号
块厚度	范围 1 - 10	1
细胞大小均匀性	范围 1 - 10	2
细胞形态均匀性	范围 1 - 10	3
边缘粘附力	范围 1 - 10	4
单上皮细胞尺寸	范围 1 - 10	5
裸核	范围 1 - 10	6
Bland 染色质	范围 1 - 10	7
正常核仁	范围 1 - 10	8
核分裂	范围 1 - 10	9

- 数据处理思路：先采用ReliefF特征提取算法计算各个属性的权重，剔除相关性最小的属性，然后采用K-means聚类算法对剩下的属性进行聚类分析。

过滤式选择-- 医学数据分析实例

- 乳腺癌数据集特征提取
  - 采用ReliefF算法来计算各个特征的权重，权重小于某个阈值的特征将被移除，针对乳腺癌的实际情况，将对权重最小的2-3种剔除。
  - 将ReliefF算法运行20次，得到了各个特征属性的权重趋势图



特征属性权重的均值

属性 1	0.2237	属性 2	0.1494
属性 3	0.1588	属性 4	0.1408
属性 5	0.0732	属性 6	0.2408
属性 7	0.1243	属性 8	0.1979
属性 9	0.0503		

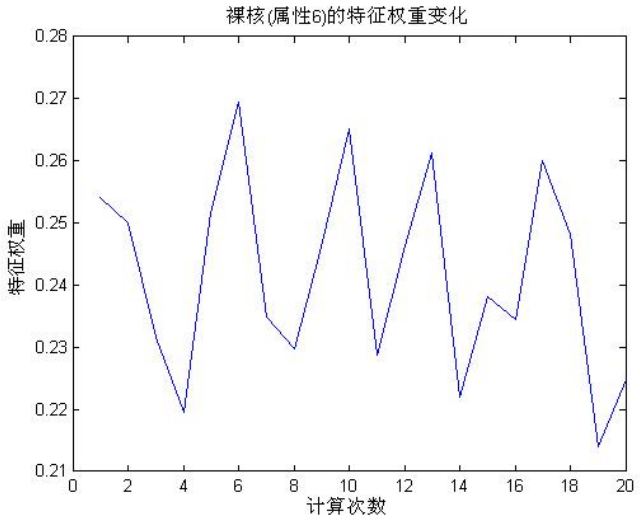
- 按照从小到大顺序排列，可知，各个属性的权重关系如下：

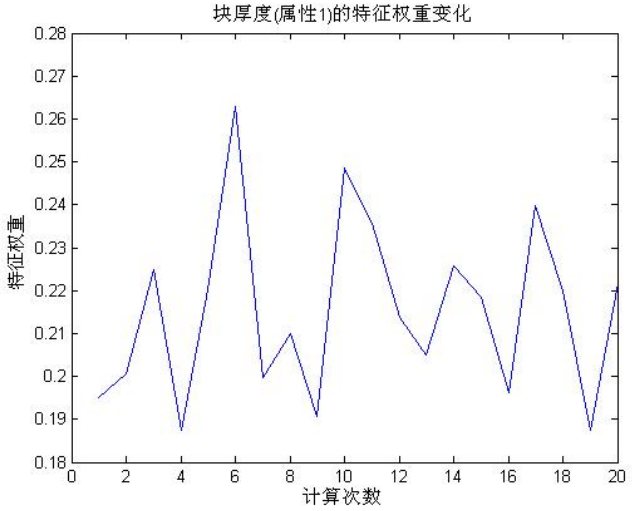
  属性9<属性5<属性7<属性4<属性2<属性3<属性8<属性1<属性6

  我们选定权重阀值为0.02，则属性9、属性4和属性5剔除。

过滤式选择-- 医学数据分析实例

- 乳腺癌数据特征分析
  - 从上面的特征权重可以看出，属性6裸核大小是最主要的影响因素，说明乳腺癌患者的症状最先表现了裸核大小上，将直接导致裸核大小的变化，其次是属性1和属性8等，后几个属性权重大小接近。
  - 几个重要的属性进行分析：





块厚度属性的特征权重在0.19-0.25左右变动，也是权重极高的一个，说明该特征属性在乳腺癌患者检测指标中是相当重要的一个判断依据。进一步分析显示，在单独对属性6，和属性1进行聚类分析，分类的成功率就可以达到91.8%。

包裹式选择

将特征子集搜索机制与子集评价机制相结合，即可得到特征选择方法

常见的特征选择方法大致分为如下三类：

- 过滤式
- 包裹式

  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
- 嵌入式

包裹式选择

- 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（Las Vegas Wrapper）是一个典型的包裹式特征选择方法，LVW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

包裹式选择-- LVW

- LVW基本步骤
  - 在循环的每一轮随
  - 在随机产生的特征
  - 进行多次循环，在征子集作为最终解
- 采用随机策略搜索特学习器，开销很大。
- 若初始特征数很多，止条件

输入： 数据集 D;
特征集 A;
学习算法 $\mathfrak{L}$;
停止条件控制参数 T.

过程：

```
1: E = ∞;
2: d = |A|;
3: A* = A;
4: t = 0;
5: while t < T do
6:    随机产生特征子集 A';
7:    d' = |A'|;
8:    E' = CrossValidation(𝔏(D^{A'}));
9:    if (E' < E) ∨ ((E' = E) ∧ (d' < d)) then
10:      t = 0;
11:      E = E';
12:      d = d';
13:      A* = A'
14:   else
15:      t = t + 1
16:   end if
17: end while
```

输出： 特征子集 A^*

图 11.1 LVW 算法描述

嵌入式选择

将特征子集搜索机制与子集评价机制相结合，即可得到特征选择方法

常见的特征选择方法大致分为如下三类：

- 过滤式
- 包裹式

特征选择过程与学习器训练过程有明显的分别

- 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，在学习器训练过程中自动地进行特征选择

嵌入式选择

- 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_1范数，则有**LASSO** [Tibshirani, 1996]

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

- L_2范数和L_1范数均有助于降低过拟合风险，但是L_1范数易获得稀疏解，即$\boldsymbol{w}$会有更少的非零分量，是一种嵌入式特征选择方法

- L_1正则化问题的求解可使用近端梯度下降算法

本章小结

- 特征选择：
  - 子集搜索与评价
  - 过滤式选择
  - 包裹式选择
  - 嵌入式选择