

● 模块一：特征选择基础 (Basics)

1. 定义与分类：

- **相关特征 (Relevant features)**: 对当前学习任务有用的属性。
- **无关特征 (Irrelevant features)**: 对任务无贡献的属性。
- **冗余特征 (Redundant features)**: 可由其他特征推导出来的属性。
- **特征选择**: 从给定的特征集合中选出一个相关特征子集的过程。

2. 核心意义：

- **缓解维数灾难**: 减少特征数量，降低计算复杂度。
- **降低学习难度**: 去除噪声和无关信息，使模型更易泛化。
- **提高可解释性**: 简化模型，便于理解哪些因素对预测起决定性作用。

● 模块二：子集搜索与评价 (Search & Evaluation)

这是特征选择最传统的流程：**搜索特征组合** → **评价好坏** → **决定是否停止**。

1. 子集搜索策略 (Subset Search):

- **前向搜索 (SFS)**: 从空集开始，每次添加一个使效果提升最大的特征（贪心）。
- **后向搜索 (SBS)**: 从全集开始，每次剔除一个影响最小的特征。
- **双向搜索**: 同时进行增删。
- **注意**: 这些方法多为**贪心策略**，无法保证找到全局最优子集，但计算效率较高。

2. 子集评价准则 (Subset Evaluation):

- **信息增益 (Information Gain)**: 基于信息熵 (Entropy)。通过特征子集划分后，如果数据集的纯度提升越大（信息熵减小越多），则该特征子集越好。

● 模块三：三大特征选择范式 (Three Paradigms)

这是考试中最常考的区别点：

范式	核心特点	典型算法	优缺点
过滤式 (Filter)	先进行特征选择，再训练学习器。两者完全独立。	Relief	优点：计算极快，通用性强。 缺点：不考虑特定学习器的特性。
包裹式 (Wrapper)	直接将最终学习器的性能作为特征子集的评价标准。	LVW (Las Vegas Wrapper)	优点：准确率通常最高（量身定做）。 缺点：计算开销巨大（需反复训练）。
嵌入式 (Embedded)	特征选择与学习器训练过程融为一体，自动完成。	LASSO	优点：效率与效果的折中，全自动化。

- 重点算法解析：

- Relief (过滤式)：通过计算特征对“猜中近邻 (Near-hit)”和“猜错近邻 (Near-miss)”的区分能力来打分。
- LVW (包裹式)：使用随机策略在拉斯维加斯框架下搜索子集，并用交叉验证评估。

● 模块四：稀疏表示与高级方法 (Sparse & Dictionary Learning)

1. ℓ_1 正则化 (LASSO)：

- 原理：在损失函数后加上 ℓ_1 范数惩罚项： $\min \|y - Xw\|^2 + \lambda \|w\|_1$ 。
- 稀疏性原理 (必考几何理解)： ℓ_1 范数的外形是带尖角的（在二维是菱形），与损失函数等高线的切点容易落在轴上，使得某些特征权重 w 严格为 0，从而实现特征自动选择。
- 求解：由于 ℓ_1 在 0 点不可导，通常用近端梯度下降 (PGD) 求解。

2. 字典学习 (Dictionary Learning)：

- 目标：寻找一个“字典”矩阵 B 和对应的稀疏系数 α ，使得原始数据 $x \approx B\alpha$ 。
- 优化：采取交替迭代法：
 1. 固定字典 B ，通过类似 LASSO 的方法求稀疏编码 α 。

2. 固定 α , 更新字典 B (常用 **K-SVD** 算法逐列更新) 。
- **应用场景:** 当原始特征不够简洁时, 通过字典学习找到更本质、更稀疏的表达方式。
-

复习小贴士

- **区分特征选择与降维:** 特征选择是从原特征中“选子集”, 特征本身含义不变; 降维 (如 PCA) 是“空间变换”, 产生的新特征是原特征的组合。
- **计算题重点:** 可能会要求计算简单的**信息熵/信息增益**, 或者描述 **Relief** 算法中正负反馈的逻辑 (Near-hit 距离越小越好, Near-miss 距离越大越好) 。
- **图形理解:** 牢记 PPT 第 28 页关于 ℓ_p 球的交点示意图, 理解为什么 $p \leq 1$ 才能产生稀疏解。