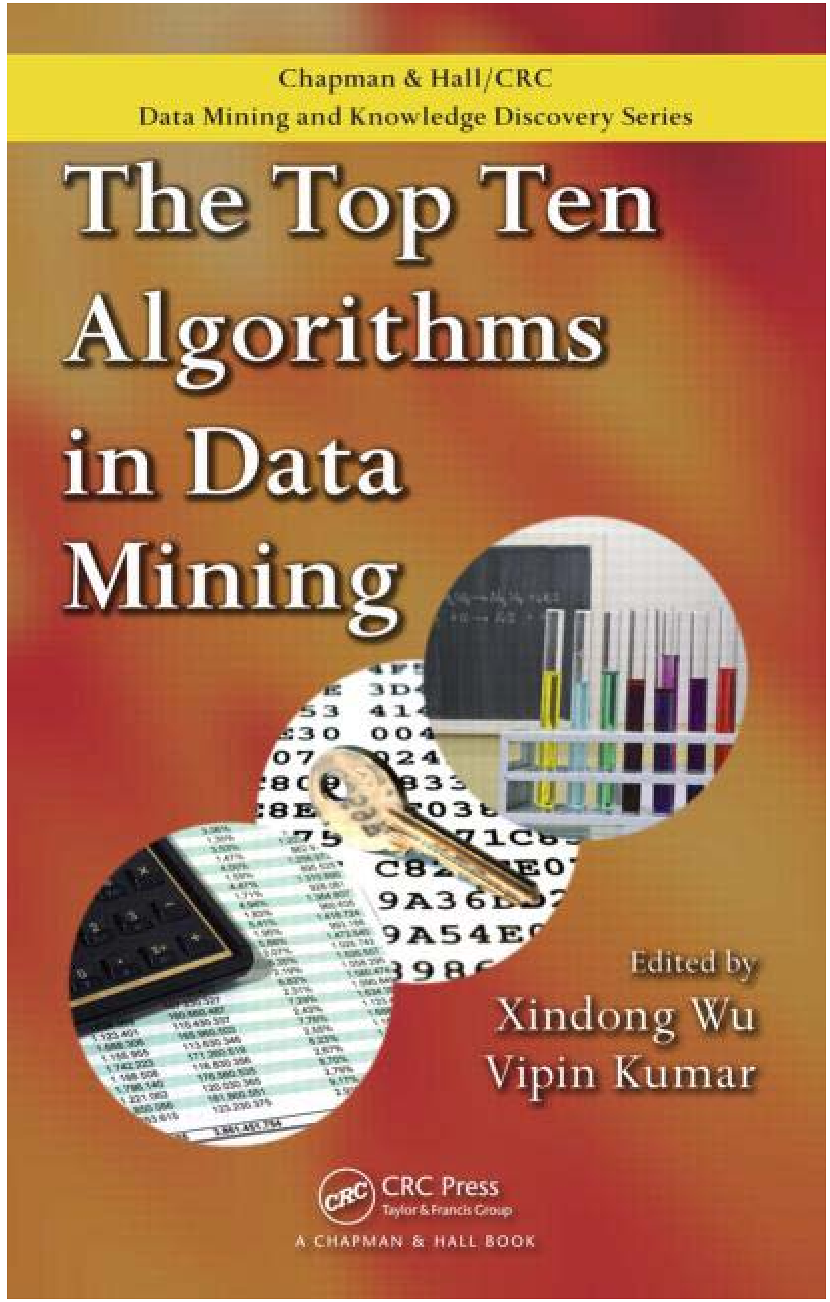
机器学习

2. 决策树


Chapman & Hall/CRC
Data Mining and Knowledge Discovery Series
The Top Ten Algorithms in Data Mining
Edited by
Xindong Wu
Vipin Kumar
CRC Press
Taylor & Francis Group
A CHAPMAN & HALL BOOK

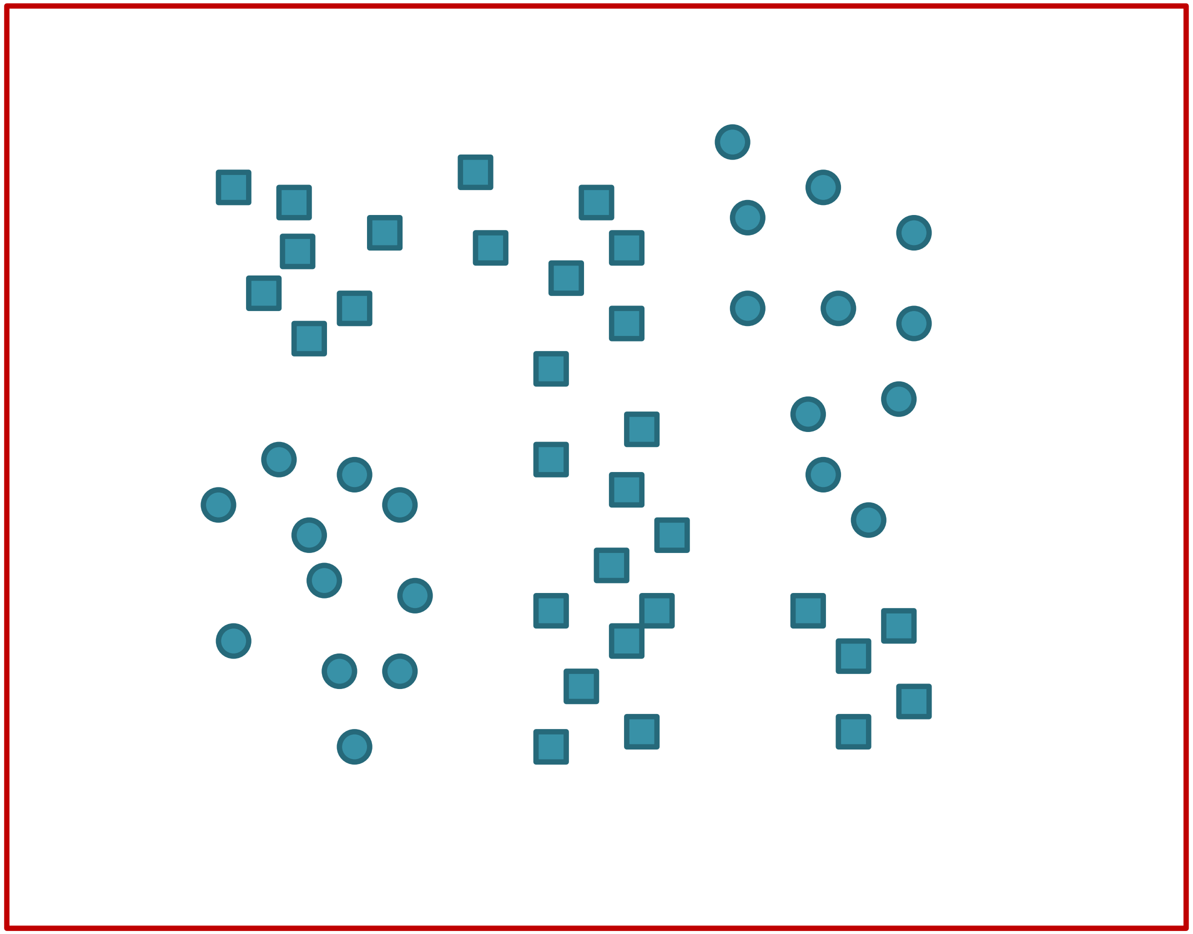
主要内容

- 基本思想
- 预测机理
- CART: Regression Tree
- 过拟合与正则化
- CART: Classification Tree

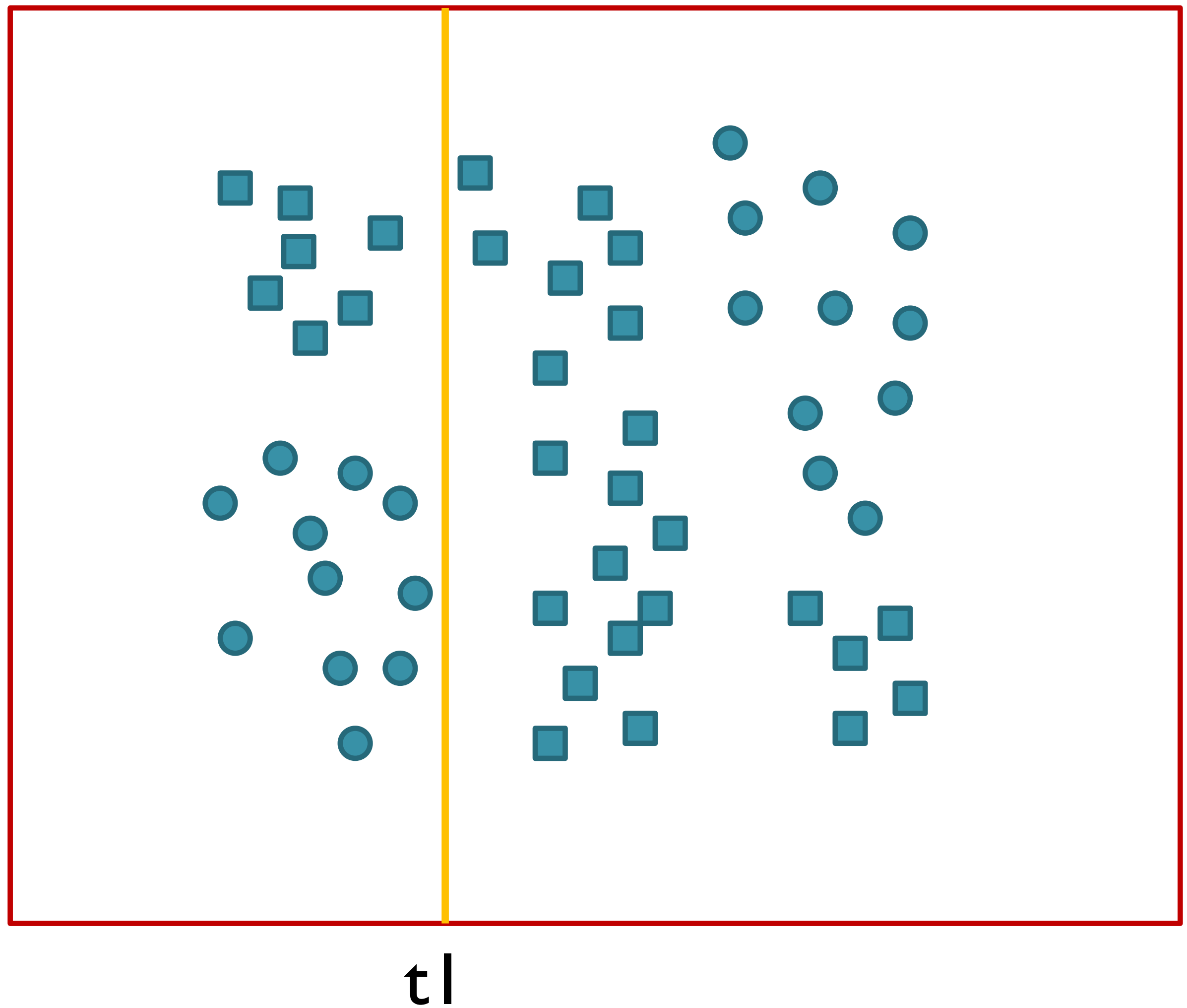
主要内容

- 基本思想
- 预测机理
- CART: Regression Tree
- 过拟合与正则化
- CART: Classification Tree

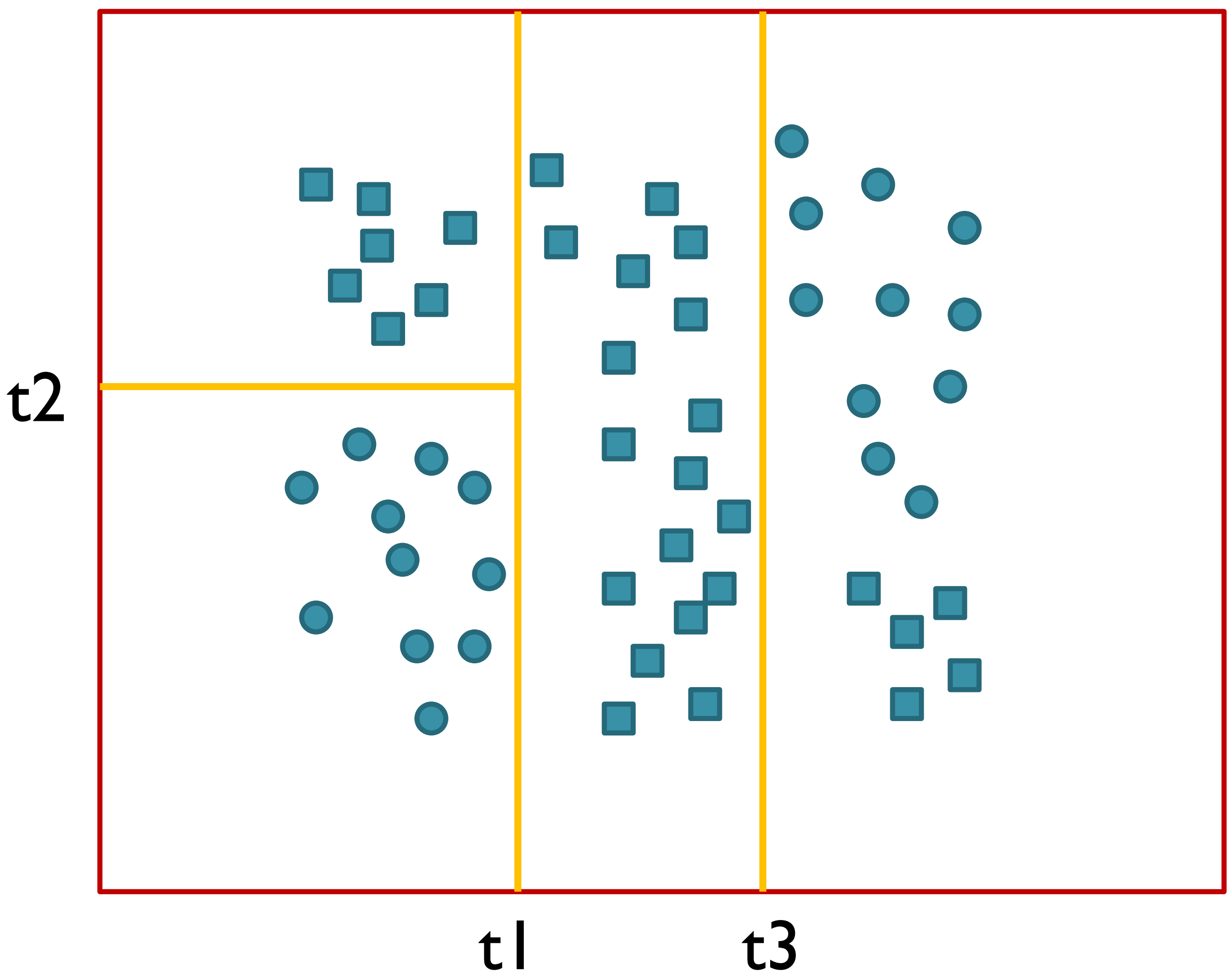
基本思想



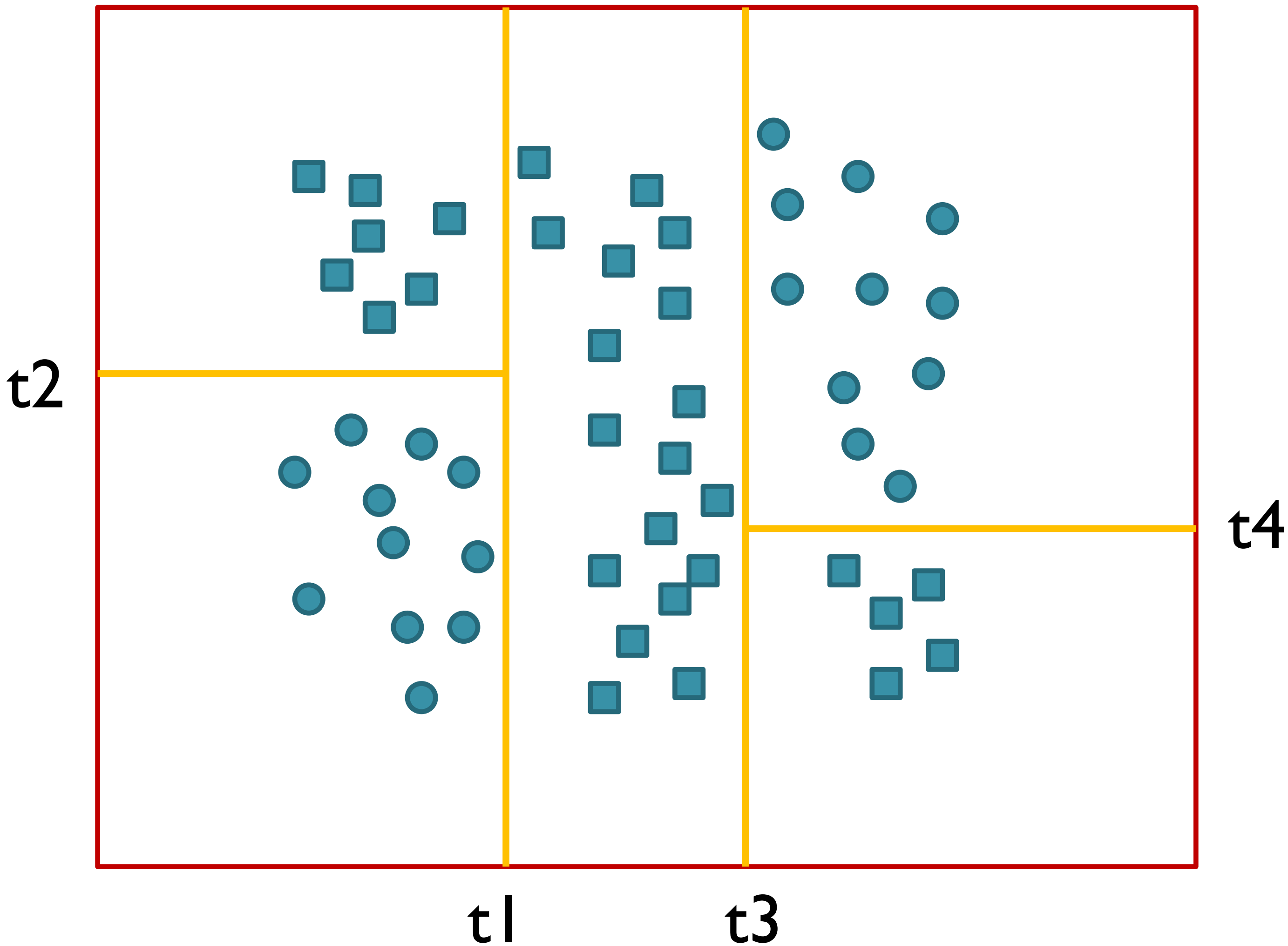
基本思想



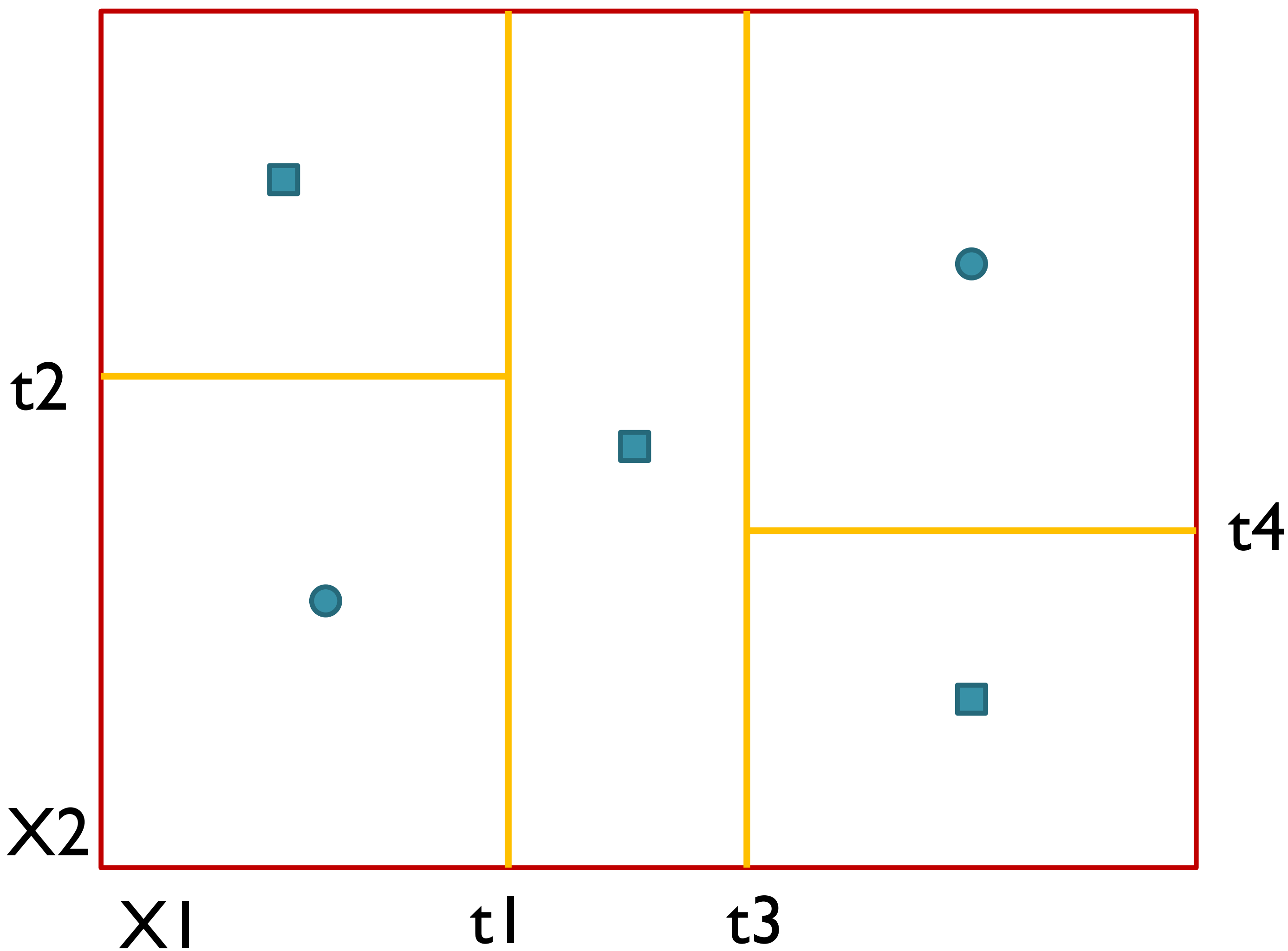
基本思想

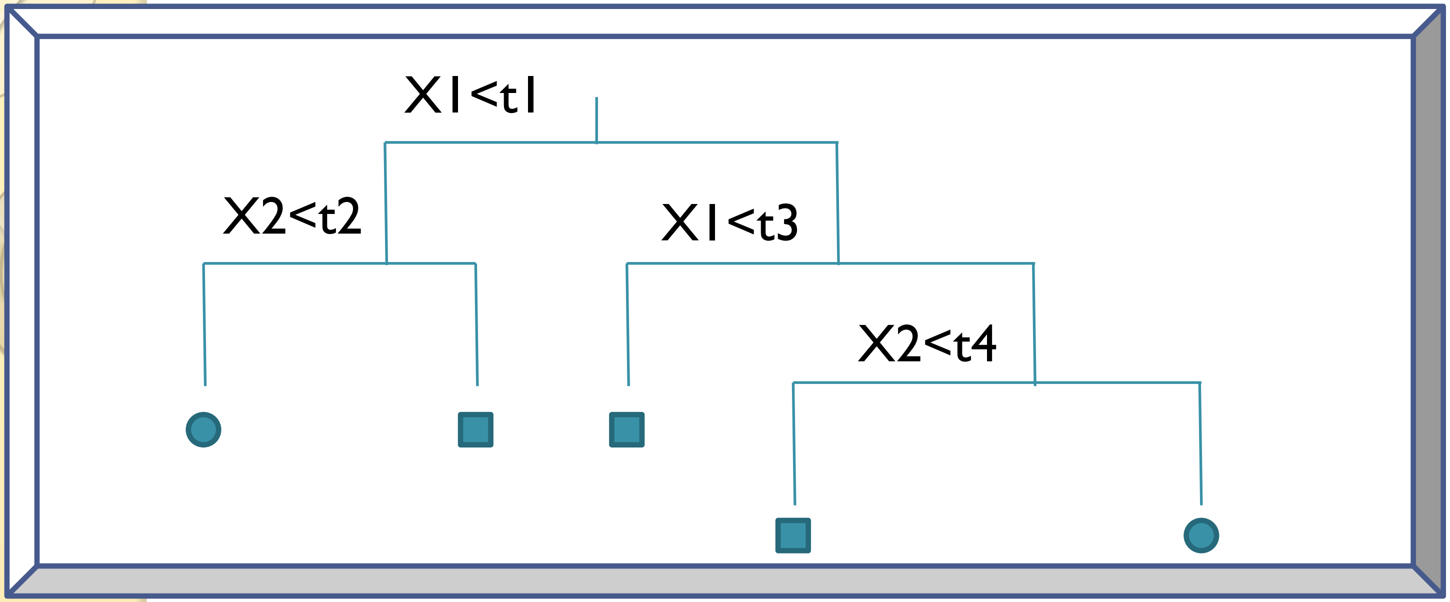


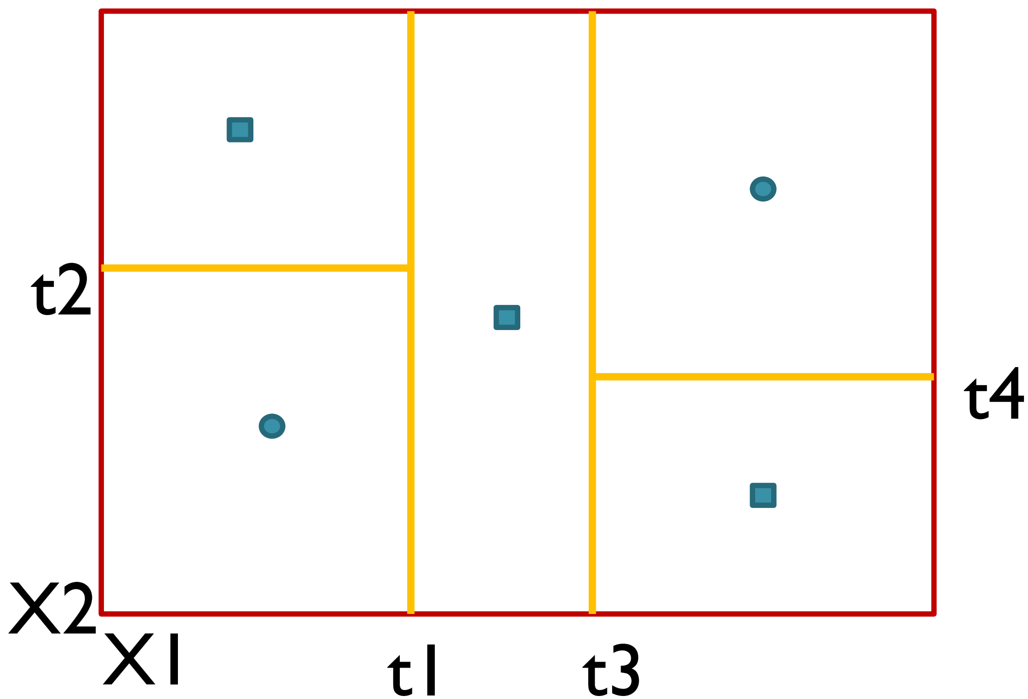
基本思想

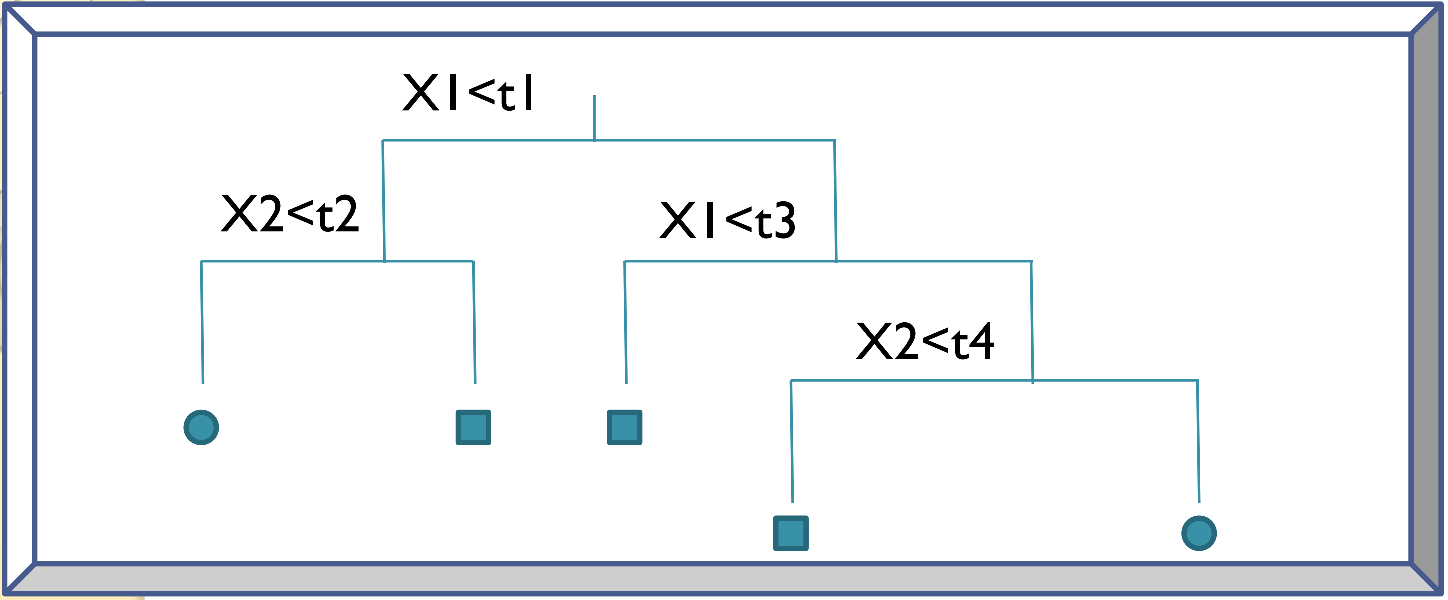


基本思想

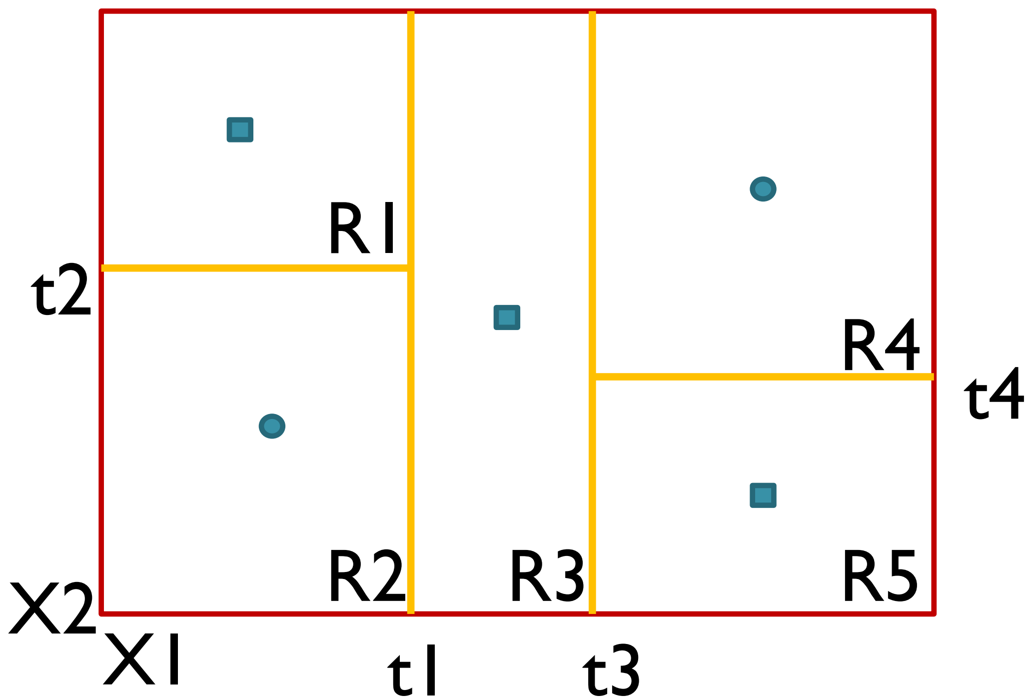


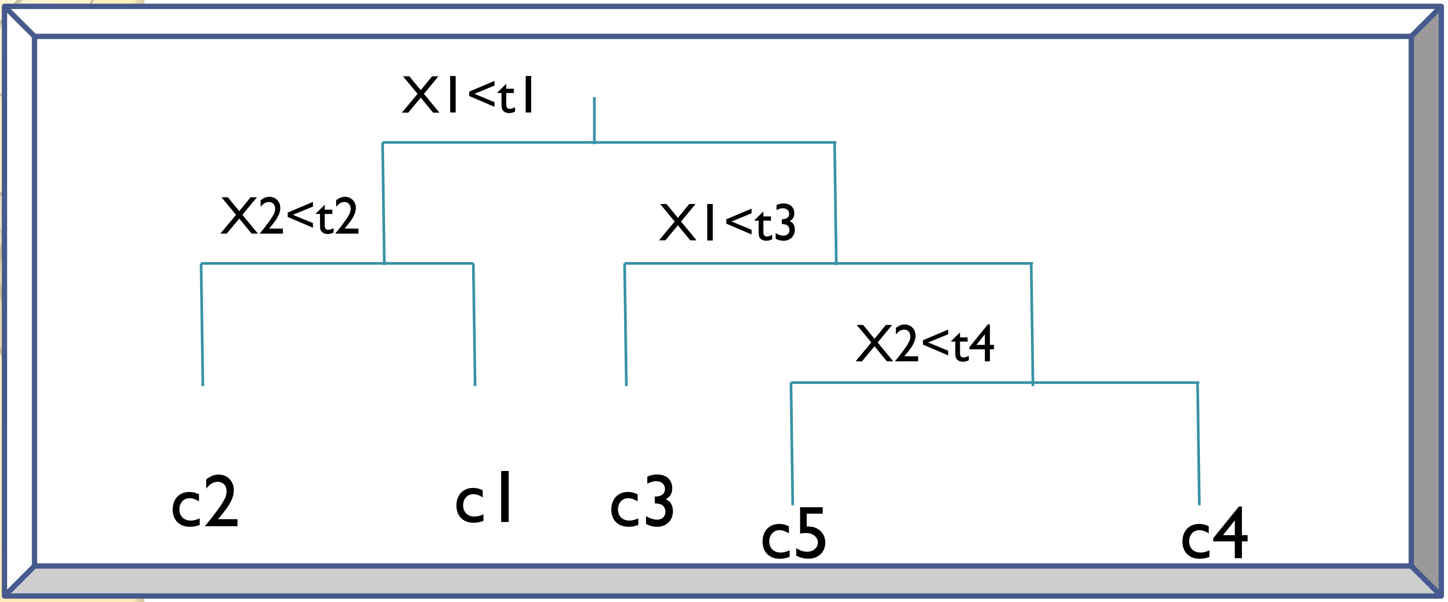

X1<t1
X2<t2
X1<t3
X2<t4


t2
t4
X2
X1
t1
t3

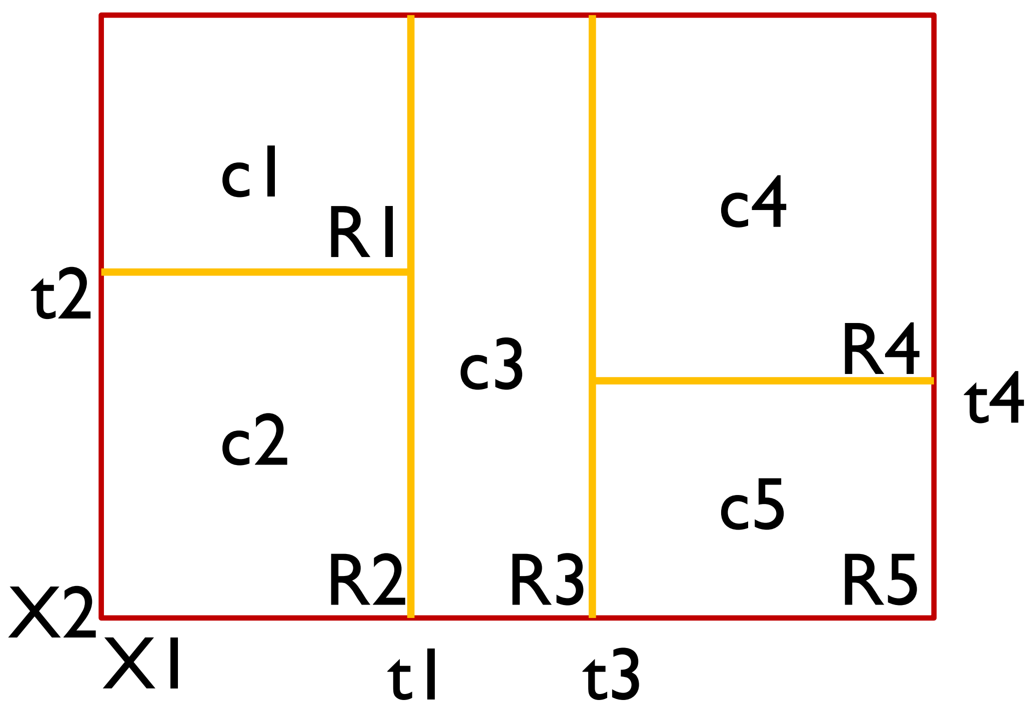


决策树
（分类）

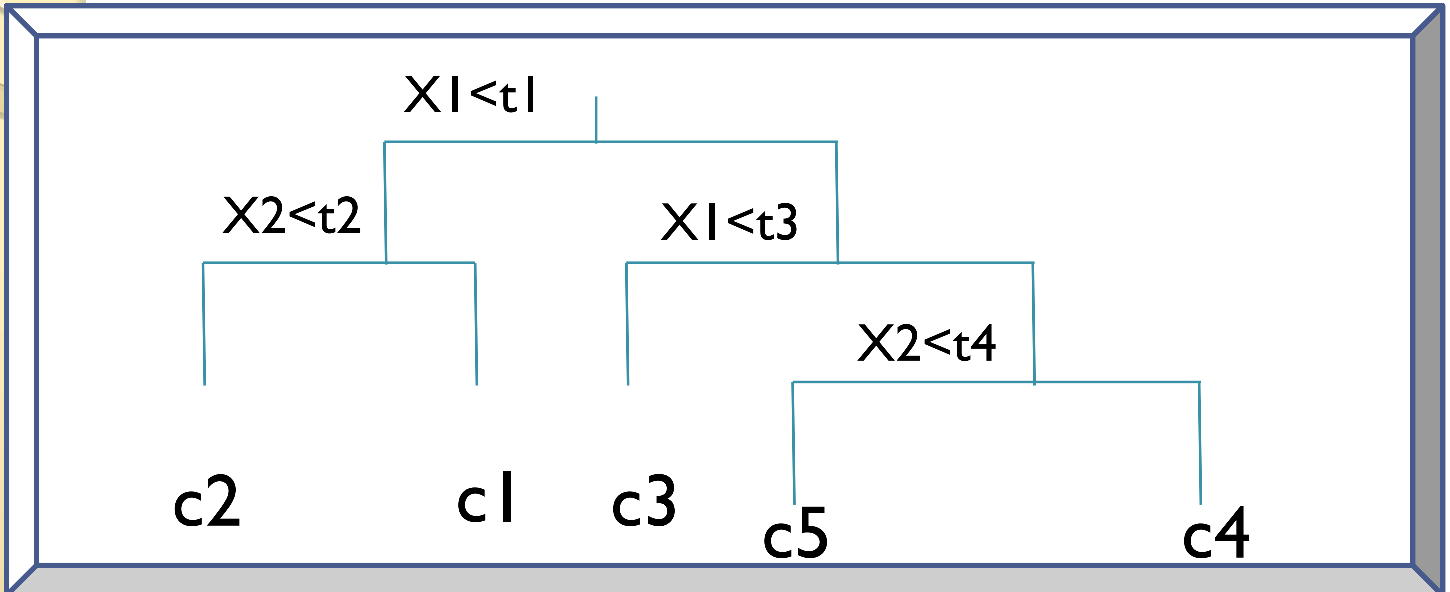



X1<t1
X2<t2
X1<t3
X2<t4
c2
c1
c3
c5
c4

决策树（回归）


c1
R1
t2
c2
R2
c3
R3
c4
R4
t4
c5
R5
X2
X1
t1
t3

基本思想



决策树
（回归）

- 基本原理：将空间分割成块状部分，然后用简单函数拟合每一部分（如常数）。

主要内容

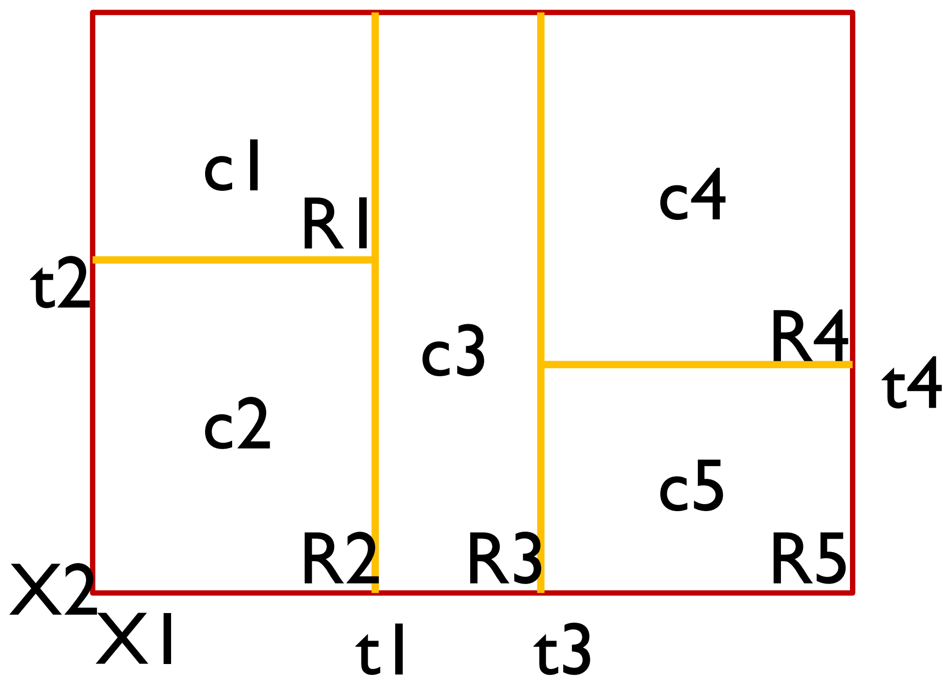
- 基本思想
- 预测机理
- CART: Regression Tree
- 过拟合与正则化
- CART: Classification Tree

预测机理

- 机器学习方法的根本要求在于***预测***

预测机理(回归)

- 机器学习方法的根本要求在于***预测***

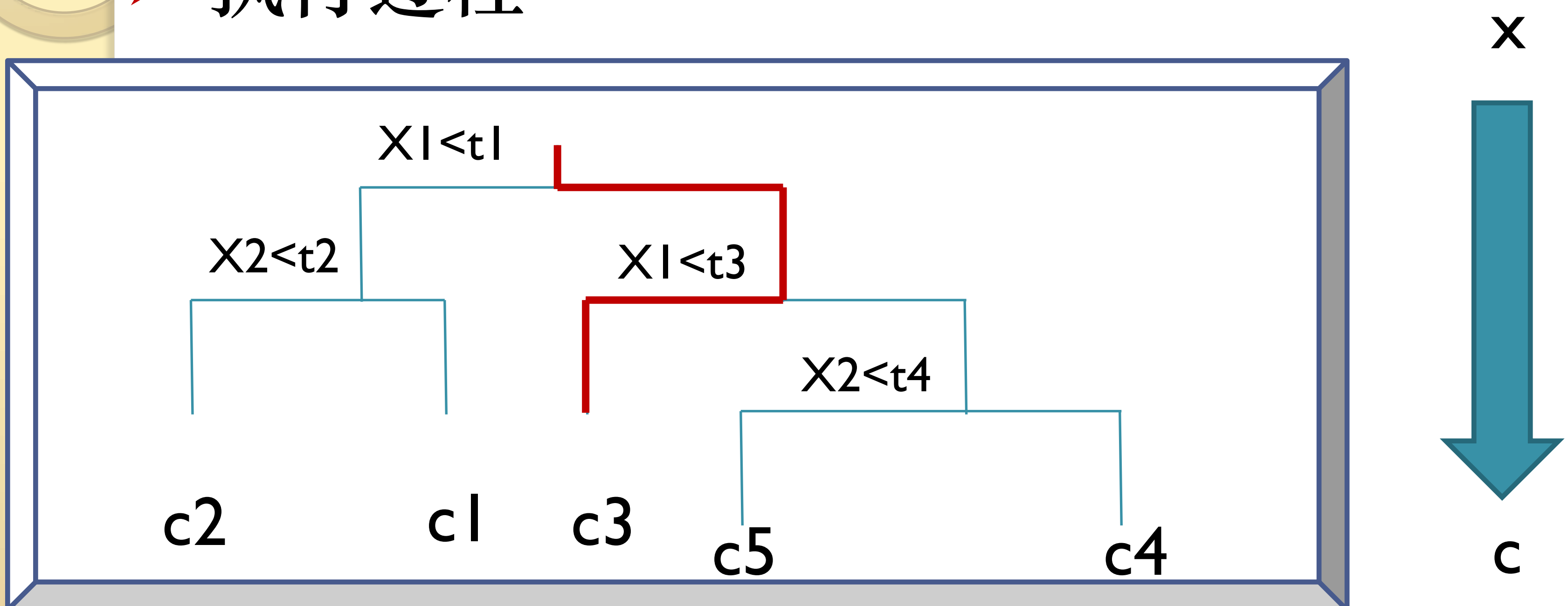


- 预测函数：

$$\hat{f}(\mathbf{x}) = \sum_{\mathrm{m}=1}^{5} c_{\mathrm{m}} I\{(x_1, x_2) \in R_{\mathrm{m}}\}$$

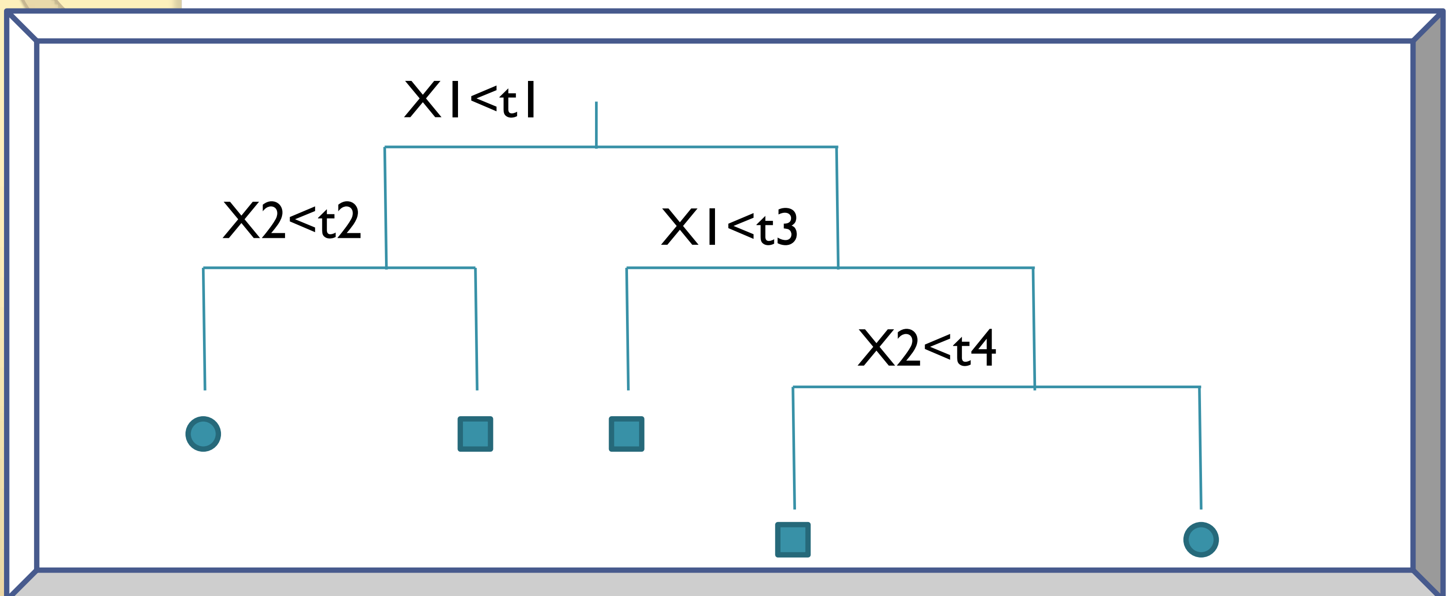
预测机理(回归)

- 执行过程



- 预测函数：$\hat{f}(\mathbf{x}) = \sum_{m=1}^{5} c_m I\{(x_1, x_2) \in R_m\}$

预测机理

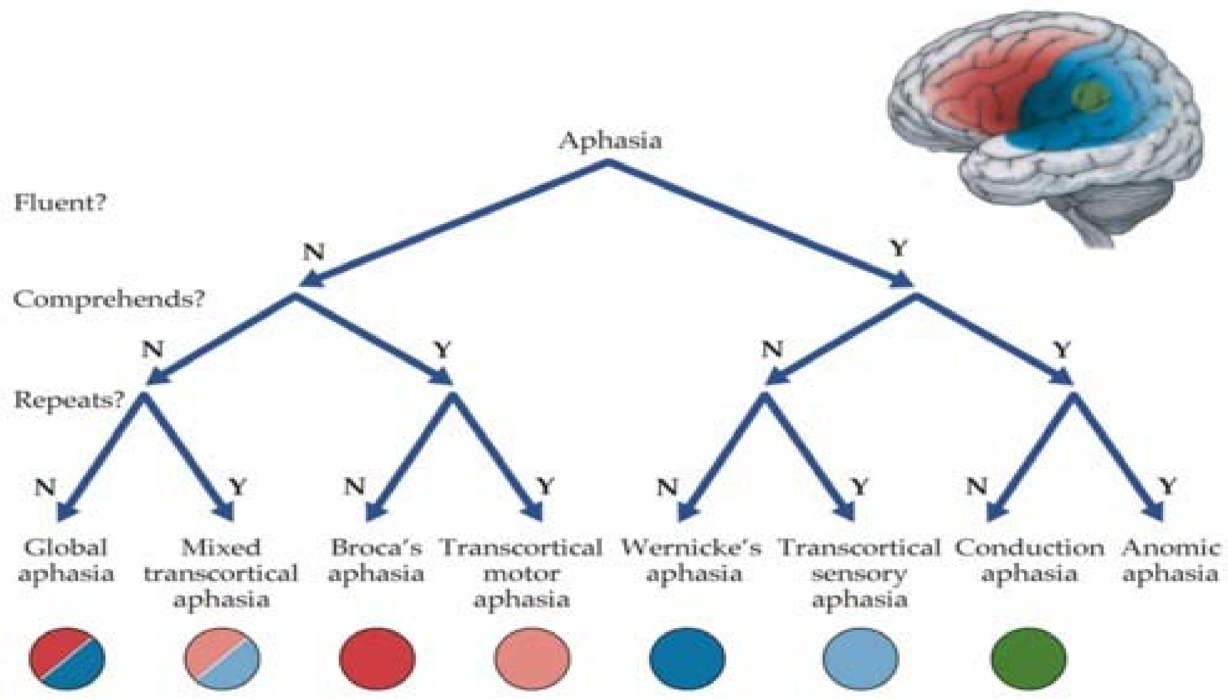


决策树
（分类）

- 预测函数：$$\hat{f}(\mathbf{x}) = \sum_{\mathrm{m}=1}^{5} y_{\mathrm{m}} I\{(x_1, x_2) \in R_{\mathrm{m}}\}$$

预测机理

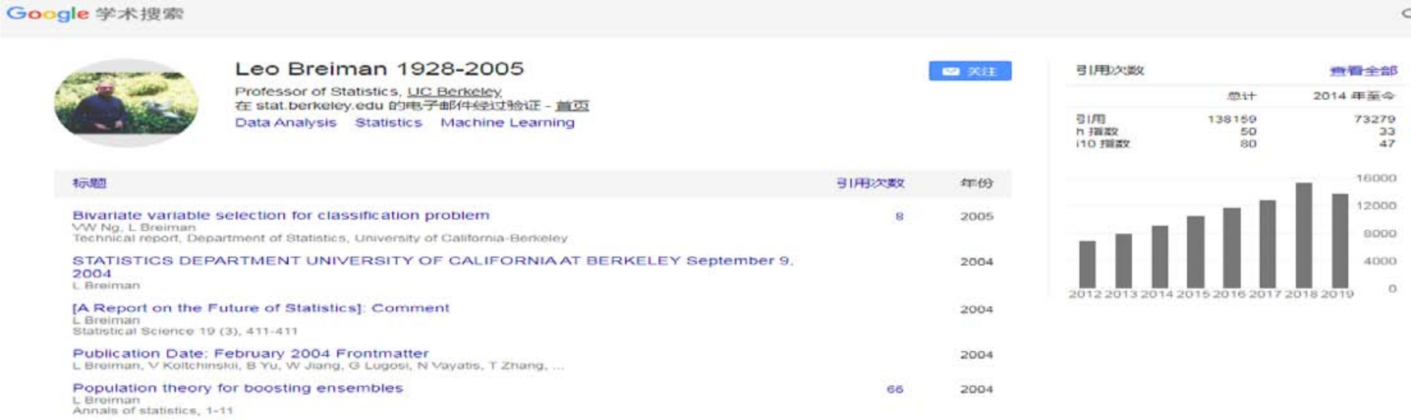
- 独特优势：可解释性
  - 分类/回归判定过程完全对应于对特征的分割过程
  - 在一些应用中，与人类判别过程完全一致，如医学诊断



- 作为一个机器学习方法，还缺乏什么?

主要内容

- 基本思想
- 预测机理
- CART: Regression Tree
- 过拟合与正则化
- CART: Classification Tree



CART-1: Regression Tree

- CART: Classification and Regression Tree

已知：$\{\mathbf{x}_i, y_i\}_{i=1}^{n}$， $\mathbf{x}_i = (x_{i1}, x_{i2}, \cdots, x_{id})^{\mathrm{T}}$

求：序列分割变量与分割点，从而将数据空间分割为 M 个区域$R_1, R_2, \cdots, R_{\mathrm{M}}$与对应预测常数$c_1, c_2, \cdots, c_{\mathrm{M}}$，对应预测函数为

$$\hat{f}(\mathbf{x}) = \sum_{\mathrm{m}=1}^{\mathrm{M}} c_{\mathrm{m}} I\{\mathbf{x} \in R_{\mathrm{m}}\}$$

CART-1: Regression Tree

已知：$\{\mathbf{x}_i, y_i\}_{i=1}^{n}$，$\mathbf{x}_i = (x_{i1}, x_{i2}, \cdots, x_{id})^{\mathrm{T}}$

求：序列分割变量与分割点，从而将数据空间分割为 M 个区域$R_1, R_2, \cdots, R_{\mathrm{M}}$与对应预测常数$c_1, c_2, \cdots, c_{\mathrm{M}}$，对应预测函数为

$$\hat{f}(\mathbf{x}) = \sum_{\mathrm{m}=1}^{\mathrm{M}} c_{\mathrm{m}} I\{\mathbf{x} \in R_{\mathrm{m}}\}$$

- **表现度量：**

$$\min_{\hat{f}} \sum_{\mathrm{i}=1}^{\mathrm{n}} \left(\hat{f}(\mathbf{x}_i) - y_i\right)^2$$

CART-1: Regression Tree

已知：$\{\mathbf{x}_i, y_i\}_{i=1}^n$，$\mathbf{x}_i = (x_{i1}, x_{i2}, \cdots, x_{id})^\mathrm{T}$

求：序列分割变量与分割点，从而将数据空间分割为 M 个区域$R_1, R_2, \cdots, R_\mathrm{M}$与对应预测常数$c_1, c_2, \cdots, c_\mathrm{M}$，对应预测函数为

$$\hat{f}(\mathbf{x}) = \sum_{\mathrm{m}=1}^{\mathrm{M}} c_\mathrm{m} I\{\mathbf{x} \in R_\mathrm{m}\}$$

- 表现度量：

$$\min_{\hat{f}} \sum_{\mathrm{i}=1}^{\mathrm{n}} \left(\hat{f}(\mathbf{x}_i) - y_i\right)^2$$

Least Square Error

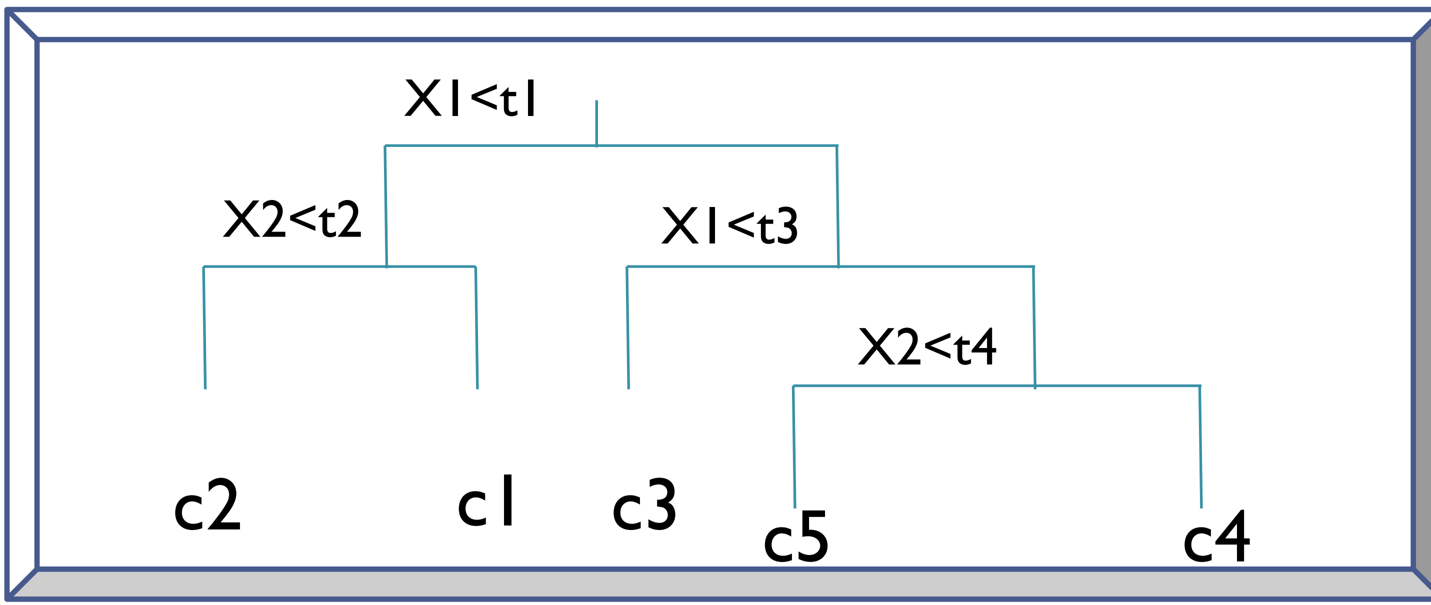
最小二乘误差

最小二乘估计

最小二乘原理

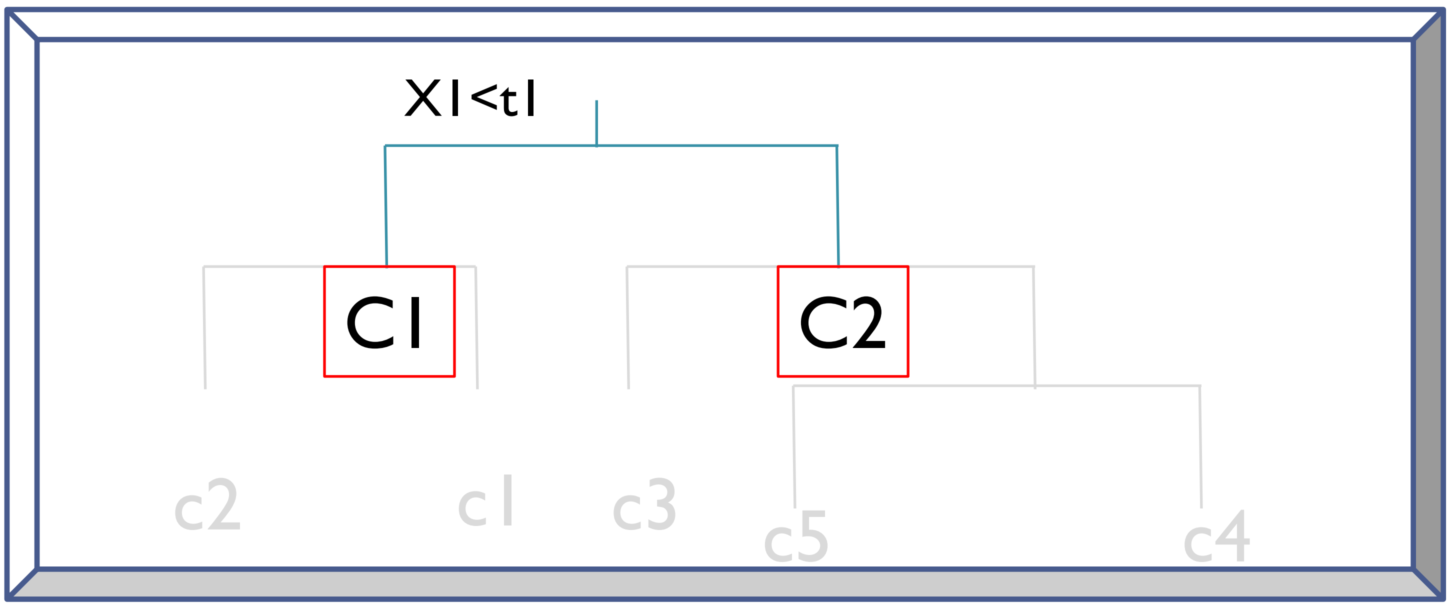
CART-1: Regression Tree

- 基本原理：贪婪算法



CART-1: Regression Tree

- 基本原理：贪婪算法



CART-1: Regression Tree

- **基本原理：贪婪算法**

已知：$\{\mathbf{x}_i, y_i\}_{i=1}^{\mathrm{m}}$，$\mathbf{x}_i = (x_{i1}, x_{i2}, \cdots, x_{id})^{\mathrm{T}}$
求：最佳分割变量x_{ij},最佳分割点s与对应预测常数c_1, c_2，使得以下优化问题达到最优：

$$\min_{j,s,c_1,c_2} \sum_{\mathrm{i}=1}^{\mathrm{m}} \left(\hat{f}(\mathbf{x}_i) - y_i\right)^2 \quad \Leftrightarrow$$

$$\min_{j,s}\left[\min_{c_1} \sum_{\mathbf{x}_i \in R_1(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{\mathbf{x}_i \in R_2(j,s)} (y_i - c_2)^2\right]$$

其中$R_1(j,s) = \{\mathbf{x} = (x_1, x_2, \cdots, x_d)^{\mathrm{T}} | x_j < s\}$，$R_2(j,s) = \{\mathbf{x} | x_j \geq s\}$.

CART-1: Regression Tree

- 预测常数c_1, c_2

  $\hat{c}_1 = mean(y_i | \mathbf{x}_i \in R_1(j,s)), \hat{c}_2 = mean(y_i | \mathbf{x}_i \in R_2(j,s))$
- 最佳分割点s: 单变量优化
- 最佳分割变量x_{ij}：共 d 个变量，遍历

CART-1: Regression Tree

- 预测常数c_1, c_2

  $\hat{c}_1 = mean\left(y_i \middle| \mathbf{x}_i \in R_1(j,s)\right), \hat{c}_2 = mean\left(y_i \middle| \mathbf{x}_i \in R_2(j,s)\right)$
- 最佳分割点s：单变量优化
- 最佳分割变量x_{ij}：共 d 个变量，遍历

- **找到最佳分割区域后，将数据集按其所在区域分割为两个部分，继续以上的方法，使"树生长"**

$\sum_{\mathrm{i=1}}^{\mathrm{m}} (y_i - c)^2 \quad \Rightarrow$

$$\min_{j,s}\left[\min_{c_1} \sum_{\mathbf{x}_i \in R_1(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{\mathbf{x}_i \in R_2(j,s)} (y_i - c_2)^2\right]$$

CART-1: Regression Tree

- 预测常数c_1, c_2

  $\hat{c}_1 = mean\left(y_i \middle| \mathbf{x}_i \in R_1(j,s)\right), \hat{c}_2 = mean\left(y_i \middle| \mathbf{x}_i \in R_2(j,s)\right)$
- 最佳分割点s：单变量优化
- 最佳分割变量x_{ij}：共 d 个变量，遍历

- **找到最佳分割区域后，将数据集按其所在区域分割为两个部分，继续以上的方法，使“树生长”**

- **目标函数可保证单调下降！**

$\sum_{\mathrm{i}=1}^{\mathrm{m}}(y_i - c)^2 \quad \Rightarrow$

$$\min_{j,s}\left[\min_{c_1}\sum_{\mathbf{x}_i \in R_1(j,s)}(y_i - c_1)^2 + \min_{c_2}\sum_{\mathbf{x}_i \in R_2(j,s)}(y_i - c_2)^2\right]$$

CART-1: Regression Tree

- Regression tree中，机器学习三个基本元素如何体现?

主要内容

- 基本思想
- 预测机理
- CART: Regression Tree
- 过拟合与正则化
- CART: Classification Tree

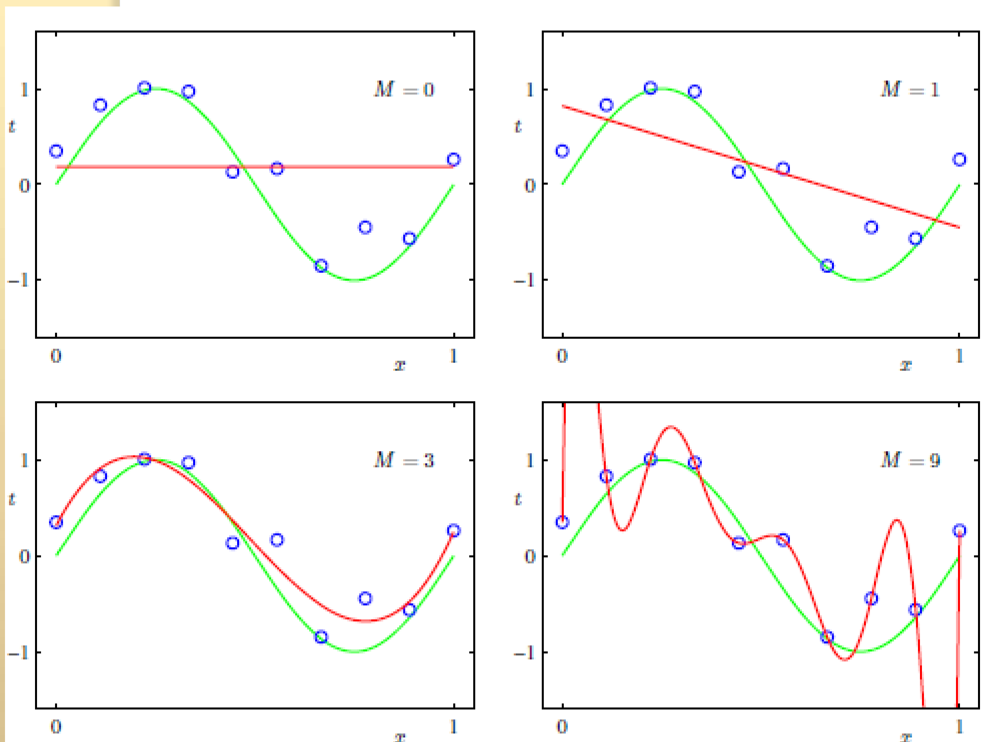
过拟合与正则化

- *若不断进行此迭代步骤，会发生什么情况？*

过拟合与正则化

➢ *若不断进行此迭代步骤，会发生什么情况?*

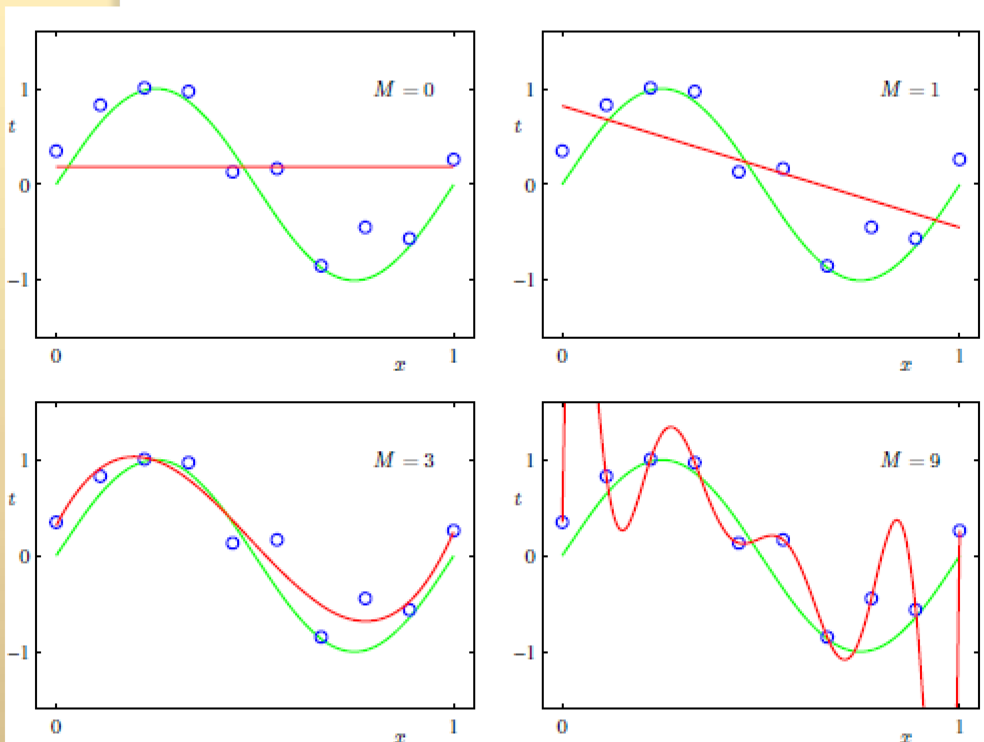
➢一个非常大的树，每个叶结点对应于一个数据；即每个分割区域中，只余一个点



Overfitting

过拟合与正则化

- ➢*若不断进行此迭代步骤，会发生什么情况?*
- ➢一个非常大的树，每个叶结点对应于一个数据；即每个分割区域中，只余一个点



过拟合与正则化

- 一个益智游戏
  - ***31 28 31 30 31 ？***

过拟合与正则化

- 一个益智游戏
  - ***31 28 31 30 31 ?***
    - ***31 ?***
    - ***56 ?***
    - ***32 ?***
    - ***其它?***

过拟合与正则化

- 一个益智游戏
  - *31 28 31 30 31 ?*
    - *31 ?*
    - *56 ?*
    - *32 ?*
    - *其它?*

Occam's Razor

- “如无必要，勿增实体”
- 简单有效原理

过拟合与正则化

- 避免Regression Tree的过拟合策略?

过拟合与正则化

- 避免Regression Tree的过拟合策略
  - 检测目标函数值下降小于一定阈值时终止
  - 当树达到一定预设高度上界时终止
  - 当每个区域包含点个数少于某个预设阈值时终止

过拟合与正则化

- 避免Regression Tree的过拟合策略
  - 检测目标函数值下降小于一定阈值时终止
  - 当树达到一定预设高度上界时终止
  - 当每个区域包含点个数少于某个预设阈值时终止
  - 控制策略嵌入优化模型：正则化

Regularization

过拟合与正则化

- 控制策略嵌入优化模型：正则化

$$\min_{\hat{f}} \sum_{\mathrm{i}=1}^{\mathrm{n}} \left(\hat{f}(\mathbf{x}_i) - y_i\right)^2 + \alpha|\hat{f}|$$

- 在树生长时，前一项（原目标函数）不断减小，后一项（正则项）不断增加

过拟合与正则化

- 避免Regression Tree的过拟合策略
  - 检测目标函数值下降小于一定阈值时终止
  - 当树达到一定预设高度上界时终止
  - 当每个区域包含点个数少于某个预设阈值时终止
  - 控制策略嵌入优化模型
  - 构造验证集
    - 在验证集效果提升的前提下生长树
    - 再生成过拟合完全数后，在验证集效果提升的前提下切割树

Validation Set

主要内容

- 基本思想
- 预测机理
- CART: Regression Tree
- 过拟合与正则化
- CART: Classification Tree

CART-2: Classification Tree

- 完全类似于Regression Tree,不同仅在于表现度量的设定

已知：$\{\mathbf{x}_i, y_i\}_{i=1}^{n}$， $\mathbf{x}_i = (x_{i1}, x_{i2}, \cdots, x_{id})^{\mathrm{T}}$

求：序列分割变量与分割点，从而将数据空间分割为 M 个区域$R_1, R_2, \cdots, R_{\mathrm{M}}$与对应预测标号$l_1, l_2, \cdots, l_{\mathrm{M}}$，对应预测函数为

$$\hat{f}(\mathbf{x}) = \sum_{\mathrm{m}=1}^{\mathrm{M}} l_{\mathrm{m}} I\{\mathbf{x} \in R_{\mathrm{m}}\}$$

表现度量

类别概率

$$p_{\mathrm{mk}} = 1/\mathrm{N_m} \sum_{\mathbf{x}_i \in R_{\mathrm{m}}} I(y_i = k)$$

分类准则

$$k(m) = \mathrm{argmax_k} \; p_{\mathrm{mk}}$$

- 构造原则：体现分类不确定性，分类信息越不确定，表现度量越大。

CART-2: Classification Tree

类别概率

$$p_{\mathrm{mk}} = 1/\mathrm{N_m} \sum_{\mathbf{x}_i \in R_{\mathrm{m}}} I(y_i = k)$$

分类准则

$$k(m) = \mathrm{argmax_k}\ \ p_{\mathrm{mk}}$$

表现度量

错分误差

$$\sum_{\mathrm{m=1}}^{\mathrm{M}} 1 - p_{\mathrm{mk(m)}}$$

基尼系数

$$\sum_{\mathrm{m=1}}^{\mathrm{M}} \sum_{\mathrm{k=1}}^{\mathrm{K}} p_{\mathrm{mk}} (1 - p_{\mathrm{mk}})$$

交叉熵

$$\sum_{\mathrm{m=1}}^{\mathrm{M}} \sum_{\mathrm{k=1}}^{\mathrm{K}} -p_{\mathrm{mk}} \log p_{\mathrm{mk}}$$

CART-2: Classification Tree

类别概率

$$p_{\mathrm{mk}} = 1/\mathrm{N_m} \sum_{\mathbf{x}_i \in R_{\mathrm{m}}} I(y_i = k)$$

分类准则

$$k(m) = \mathrm{argmax_k}\ p_{\mathrm{mk}}$$

表现度量

- 错分误差

$$\sum_{\mathrm{m}=1}^{\mathrm{M}} 1 - p_{\mathrm{mk(m)}}$$

- 基尼系数

$$\sum_{\mathrm{m}=1}^{\mathrm{M}} \sum_{\mathrm{k}=1}^{\mathrm{K}} p_{\mathrm{mk}}(1 - p_{\mathrm{mk}})$$

- 交叉熵

$$\sum_{\mathrm{m}=1}^{\mathrm{M}} \sum_{\mathrm{k}=1}^{\mathrm{K}} -p_{\mathrm{mk}} \log p_{\mathrm{mk}}$$

Entropy

CART-2: Classification Tree

- 熵的意义
  - 事件信息量h(x)：惊喜程度 （与事件概率相关）
  - h(x)与p(x)呈单调递减关系
  - 若x,y不相关，$h(x,y) = h(x) + h(y)$
  - $h(x) = ?$

CART-2: Classification Tree

- 熵的意义
  - 事件信息量h(x)：惊喜程度 （与事件概率相关）
  - h(x)与p(x)呈单调递减关系
  - 若x,y不相关，h(x,y) = h(x) + h(y)
  - h(x) = -log p(x) (-log_2 p(x))

CART-2: Classification Tree

- 熵的意义
  - 事件信息量h(x)：惊喜程度 （与事件概率相关）
  - h(x)与p(x)呈单调递减关系
  - 若x,y不相关，h(x,y) = h(x) + h(y)
  - 自然的推测 h(x) = -log p(x) (-log_2 p(x))
- 若要把随机变量x传输给一个接收器，则平均传输信息量为：
  - 熵：H(x) = - sum_x p(x)log p(x)
  - 分布不均匀，熵越小

- KL散度：

D(p//q)=sum_x (p(x)*log(p(x)/q(x)))

CART-2: Classification Tree

类别概率

$$p_{\mathrm{mk}} = 1/\mathrm{N_m} \sum_{\mathbf{x}_i \in R_{\mathrm{m}}} I(y_i = k)$$

分类准则

$$k(m) = \mathrm{argmax_k}\ p_{\mathrm{mk}}$$

表现度量

错分误差

$$\sum_{\mathrm{m}=1}^{\mathrm{M}} 1 - p_{\mathrm{mk(m)}}$$

基尼系数

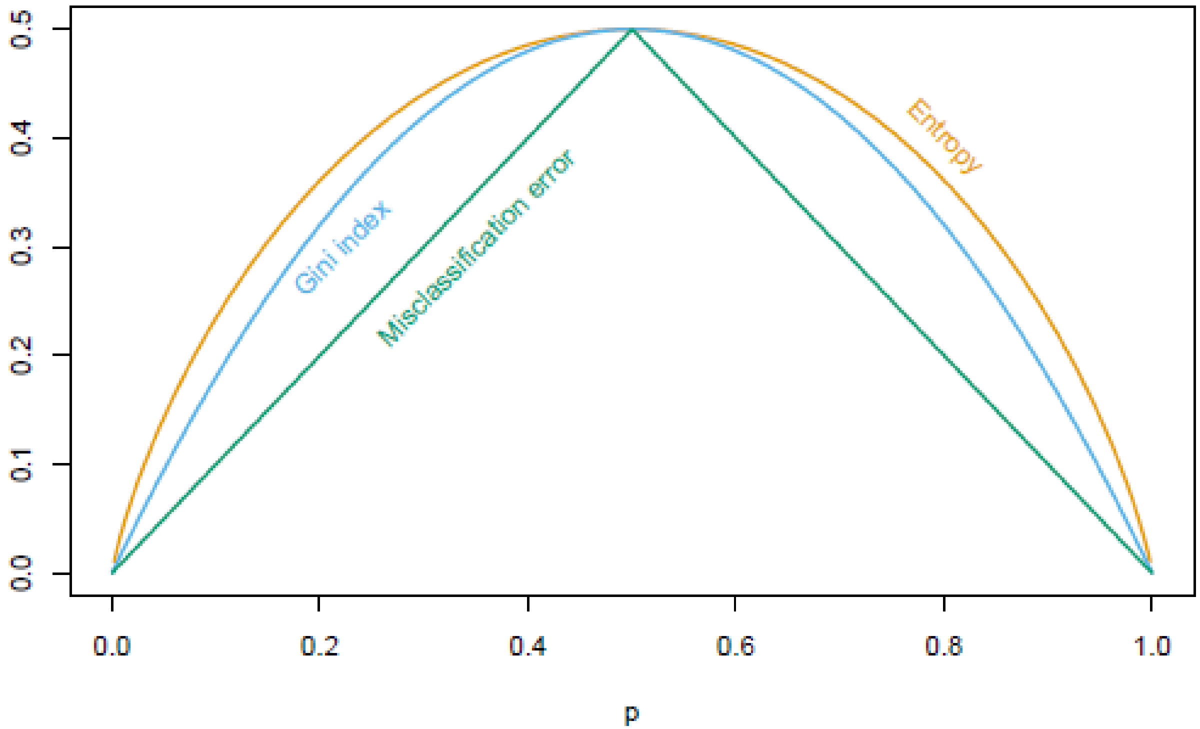
$$\sum_{\mathrm{m}=1}^{\mathrm{M}} \sum_{\mathrm{k}=1}^{\mathrm{K}} p_{\mathrm{mk}}(1 - p_{\mathrm{mk}})$$

交叉熵

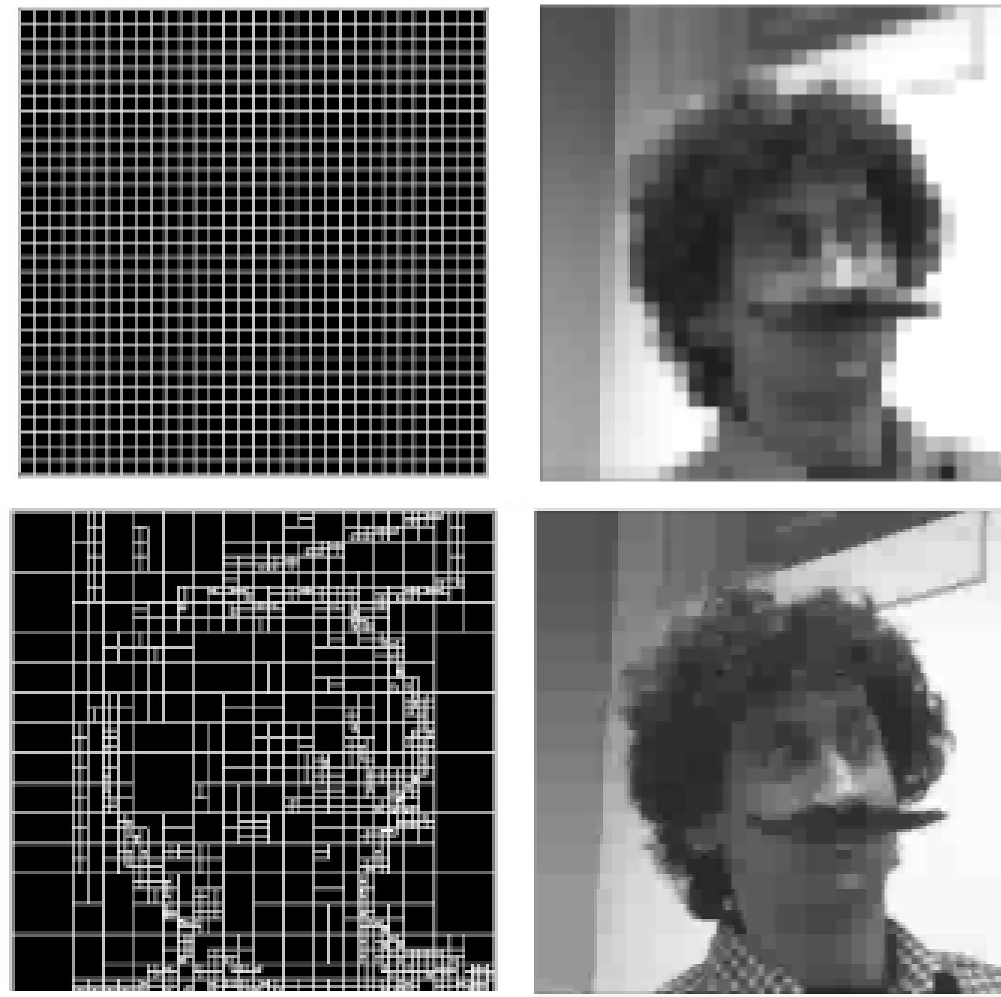
$$\sum_{\mathrm{m}=1}^{\mathrm{M}} \sum_{\mathrm{k}=1}^{\mathrm{K}} -p_{\mathrm{mk}} \log p_{\mathrm{mk}}$$

CART-2: Classification Tree

- 基尼系数与交叉熵 可微，错分误差不可微
- 基尼系数与交叉熵 对类别概率变化更敏感



图像编码



1024 cells in each partition

From Aarti Singh's slides

图像编码

JPEG 0.125 bpp
non-adaptive partitioning

JPEG 2000 0.125 bpp
adaptive partitioning

From Aarti Singh's slides

进一步说明

- 为何用2分不用多分：一方面为了好计算，另一方面多分可能会过于快速将数据空间分割成小的碎片，更易导致过拟合问题
- 其它决策树方法：ID3 与其更近期的版本C4.5,C5
- 扩展：用线性函数而非常数逼近一个区域；用线性而非简单大小判定在制定分区域规则
- 缺点：不稳定性。输入数据的一个小的变化可能导致输出的极大不同
- 决策面非光滑

要求

1. 搞清楚CART的基本操作原理
2. 构造人工数据或下载UCI数据，进行一个计算实例
3. 搞清楚本课涉及的机器学习基本概念

阅读:

[1] The Elements of Statistical Learning: Data Mining, Inference and Prediction. Hastie, Tibshirani, Friedman. Springer, 2008. 9.2 Tree-based Methods
[2] Do we Need Hundreds of Classifiers to Solve Real World Classification Problems? Fernández-Delgado, Cernadas, Barro, Amorim; JMLR:3133–3181, 2014.
[3] UCI 数据集： http://archive.ics.uci.edu/ml/