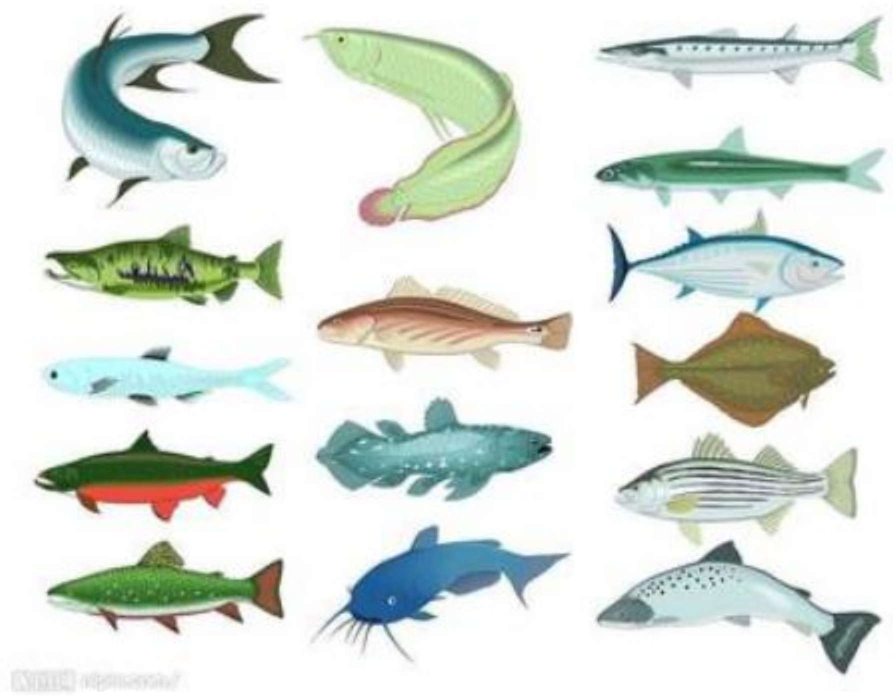
大数据挖掘与统计学习

软件工程系
文化遗产数字化国家地方工程联合中心
可视化技术研究所
张海波
讲师/博士(后)

k-近邻算法

分类问题



分类问题


杰夫·布里吉斯
冥界警局
硬汉杀怪升级之路
克里斯·派恩
特工争风
CIA特工情敌大战
尼古拉斯·凯奇
劫案迷云
影帝凯奇火爆回归
科林·法瑞尔
全面回忆
布莱德利·库珀
永无止境
小药片引发逆天潜能
科洛·莫瑞兹
海扁王
艾尔顿·塞纳
永远的车神
赛车王子传奇一生
全智贤·金允石
夺宝联盟
中韩盗贼大比拼
詹姆斯·弗兰科
猩球崛起
西尔维斯特·史泰龙
敢死队2

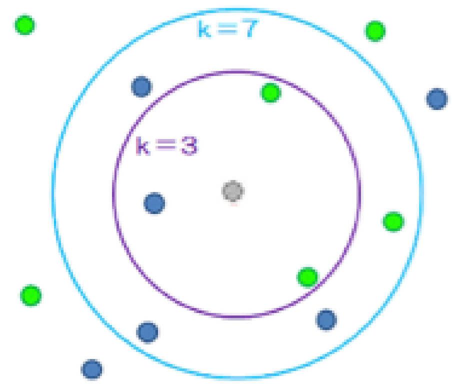
- 爱情片、剧情片、喜剧片、家庭片、伦理片、
文艺片、音乐片、歌舞片、动漫片、
西部片、武侠片、古装片、动作片、
恐怖片、惊悚片、冒险片、犯罪片、悬疑片、
记录片、战争片、历史片、传记片、体育片、
科幻片、魔幻片、奇幻片

Supervised learning

- **K-**近邻算法（**KNN**算法）是一种用于分类和回归的非参数统计方法。
- 最近邻方法在**1970**年代初被用于统计估计和模式识别领域。
- 该方法仍然是十大数据挖掘算法之一。

- 近朱者赤近墨者黑

- 把这种思想用于数据方面



Note:

- kNN算法的核心思想是如果一个样本在特征空间中的k个最相邻的样本中的大多数属于某一个类别，则该样本也属于这个类别，并具有这个类别上样本的特性。
- “K”表示分类考虑的数据集项目的数量。
- K-NN是一种基于实例的学习。
- k-近邻算法是所有的机器学习算法中最简单的方法之一。

- 给定训练数据（或已标记数据）$\{(x_1, y_1), \ldots, (x_N, y_N)\}$ 以及测试点x
  - N 对；x_i 是D维特征所组成的向量, y_i - 标记或类别
- 目标: 输出对未知标记或类别的样本 x的预测y
- 预测准则：寻找训练数据中最近的K个样本

- 输出形式:

  分类问题: 离散值 $y_i \in \{1, \ldots, C\}$

  多数投票(majority voting)

  回归问题：连续的（实值）变量$y_i \in R$

  平均值 average response

- 这个算法需要:

  参数 K：寻找的近邻个数

  距离函数: 计算样本之间的相似度

常见的度量方式

- 欧氏距离 （ Euclidean distance ）最常使用

在二维欧式平面中，两点 $\mathbf{p} = (p_1, p_2)$ 和 $\mathbf{q} = (q_1, q_2)$ 的距离为

$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}.$$

三维空间中的欧氏距离

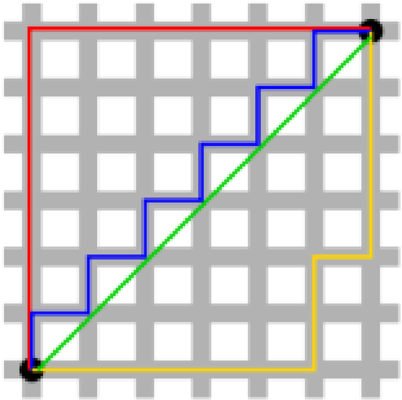
$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + (p_3 - q_3)^2}.$$

一般的，n维空间中的距离

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_i - q_i)^2 + \cdots + (p_n - q_n)^2}.$$

- 曼哈顿距离（Manhattan Distance）

$$d\ (\mathbf{p},\mathbf{q}) = |p_1 - q_1| + |p_2 - q_2|$$



一般的，n维空间中的两点的曼哈顿距离是

$$d\ (\mathbf{p},\mathbf{q}) = \|\mathbf{p} - \mathbf{q}\| = \sum_{i=1}^{n} |p_i - q_i|$$

- 闵可夫斯基距离 (**Minkowski Distance**)

闵氏距离不是一种距离，而是一组距离的定义，是对多个距离度量公式的概括性的表述。

两个n维变量 $p=(p_1, p_2, \ldots, p_n)$和 $q=(q_1, q_2, \ldots, q_n)\in\mathbb{R}^n$

之间的闵式距离定义为:

$$\left(\sum_{i=1}^{n}|p_i-q_i|^m\right)^{1/m}$$

m取1或2时的闵氏距离是最为常用的，m=2即为欧氏距离，而m=1时则为曼哈顿距离。当m取无穷时的极限情况下，可以得到切比雪夫距离。

- 夹角余弦（ **Cosine similarity** ）

几何中，夹角余弦可用来衡量两个向量方向的差异；机器学习中，借用这一概念来衡量样本向量之间的差异。

两个n维样本点的夹角余弦为：

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}$$

夹角余弦取值范围为[-1,1]。余弦越大表示两个向量的夹角越小，余弦越小表示两向量的夹角越大。当两个向量的方向重合时余弦取最大值1，当两个向量的方向完全相反余弦取最小值-1。

- **Hamming distance** (汉明距离)

  两个等长字符串s1与s2的汉明距离为：将其中一个变为另外一个所需要作的最小字符替换次数。

例如: 左右字符串之间的汉明距离分别是:

- **1011101** and **1001001** is 2.
- **2173896** and **2233796** is 3.

汉明距离在包括信息论、编码理论、密码学等领域都有应用。比如在信息编码过程中，为了增强容错性，应使得编码间的最小汉明距离尽可能大。

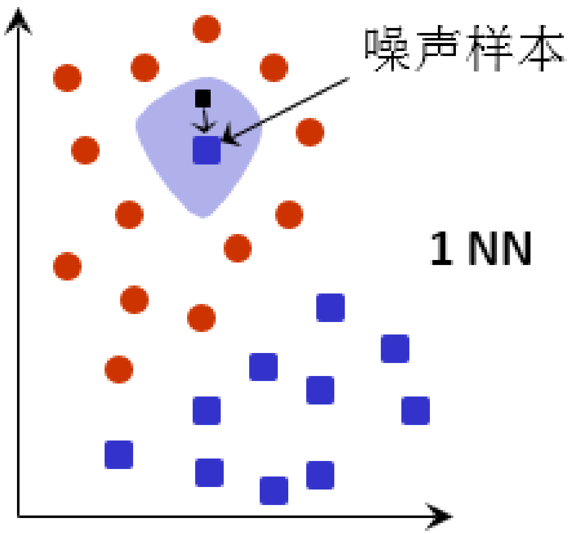
K的选择

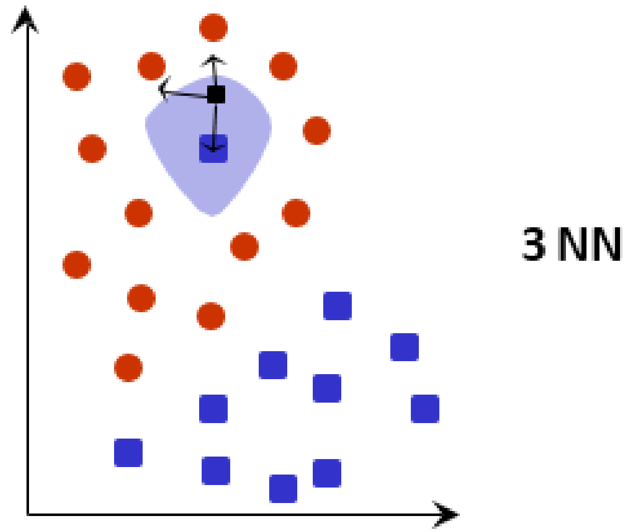
- 理论上，如果有无穷多的样本，k越大，分类效果越好.
- 这是不可能实现的，实际中样本个数总是有限的

两种极端情况：

- k=1　最近样本的类别
- k=N　样本个数最多的类别

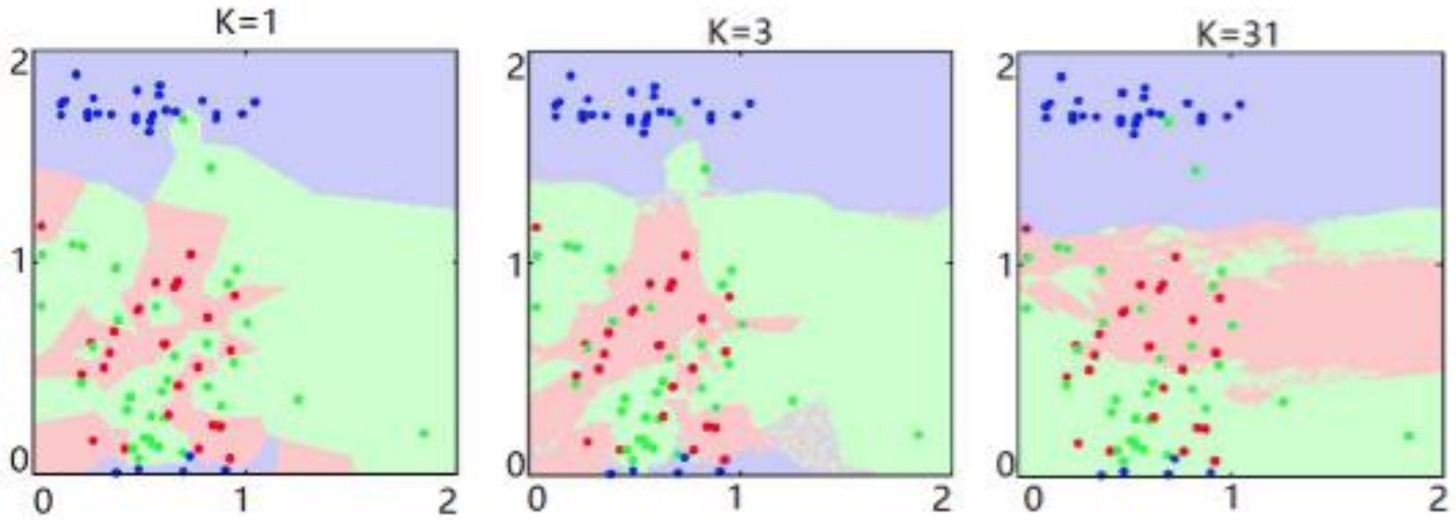
- k = 1 最常用， 效果也较好, 但是却对“噪声” 敏感





任何浅蓝色区域内的样本都会被错分为蓝色类别。

任何浅蓝色区域内的样本都会被正确分类为红色类别。



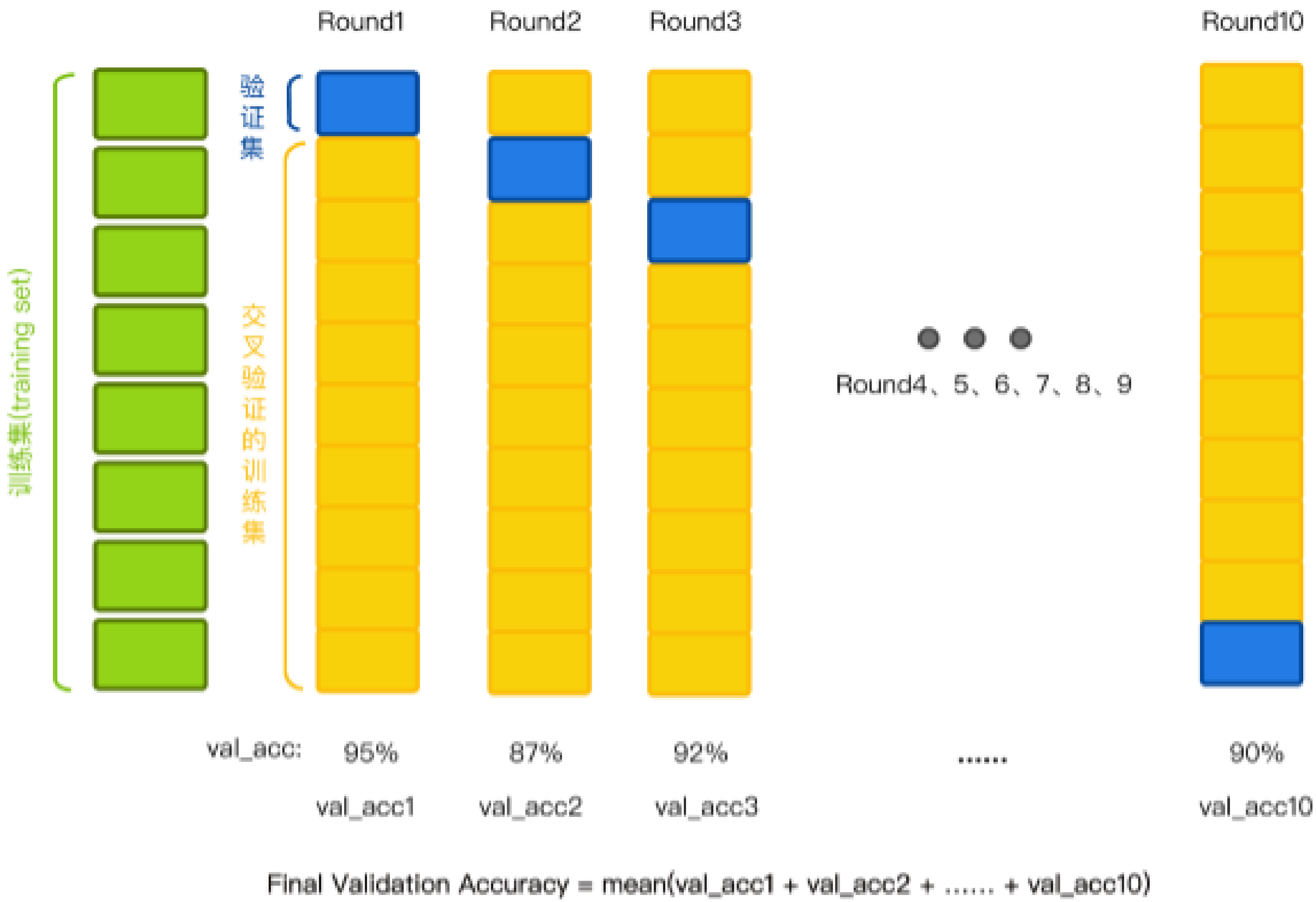
小 K

对每个类别都创建了许多小的分类区域
对“噪声敏感”
非平滑的决策边界，（可能导致过拟合)

大 K

创建了少数大范围的区域,
通常产生更平滑的决策边界
可以降低噪声样本的影响
(注意过于平滑的决策边界可能导致欠拟合)

常用优化K的方法：K折交叉验证



总结：距离度量&K的选择

- 如何选择距离度量方式?

  欧氏距离（Euclidean）最为常用
  具体问题具体分析
  例如：对于一个复杂的问题，不同维度上也可以使用不同的度量方式

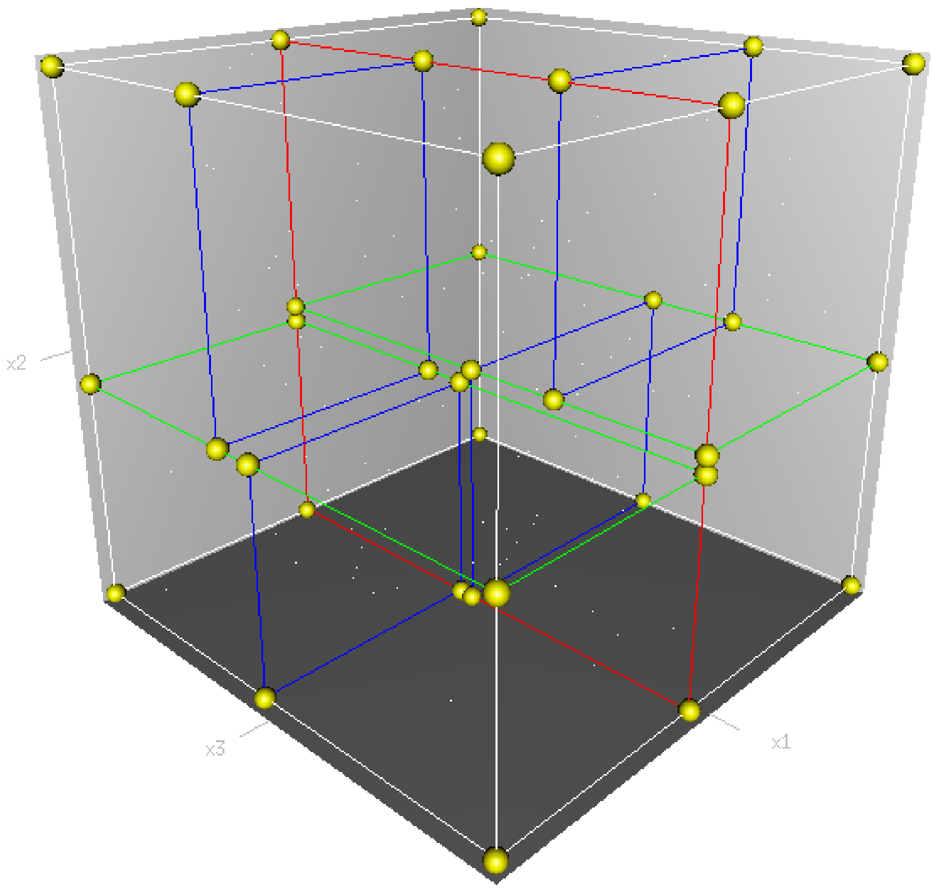
- K的选择

  最好是奇数
  1-NN 在实践中经常表现不错
  一个有趣的理论性质是$k < \mathrm{sqrt}(n)$, n是样本个数
  可以通过交叉验证法（cross-validation）等

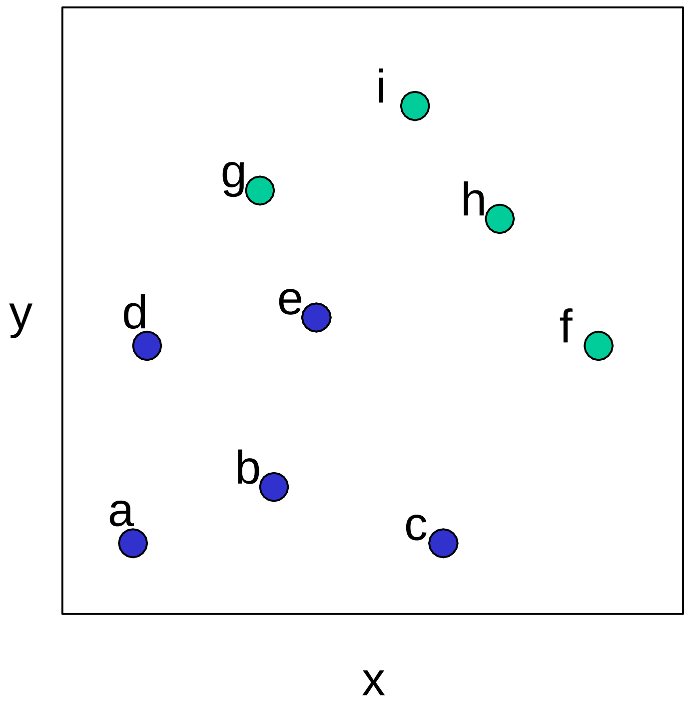
最常用的最近邻搜索算法： k-d树

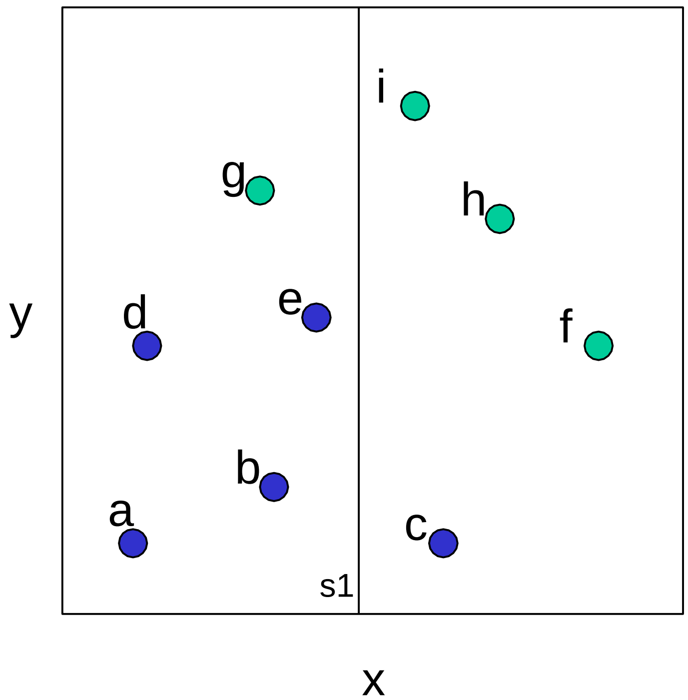
- 20世纪70年代由Jon Bentley发明，k维空间中划分的一种数据结构，主要应用于多维空间范围搜索和最近邻搜索
- Kd-树是K-dimension tree的缩写，名称原来是指“3-D树，4-d树等”，其中k是尺寸的数量
- 思想：树的每个节点划分仅使用1个维比较。
  - 用于存储空间数据。
  - 最邻居搜索。
  - 范围查询。
  - 快速查找！

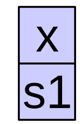
3D K-d 树

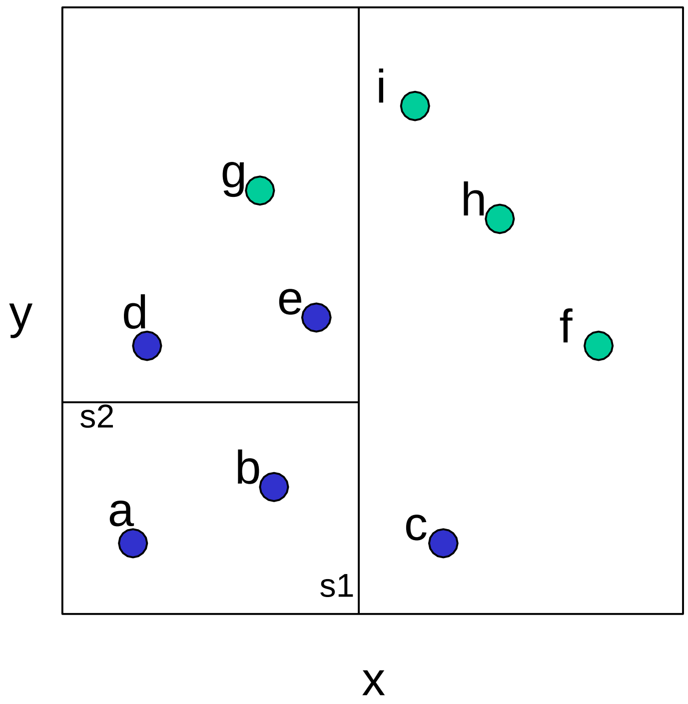


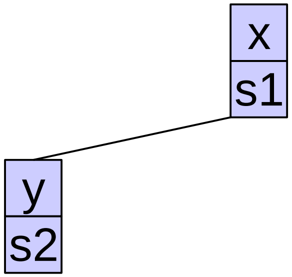
K-d 树构造

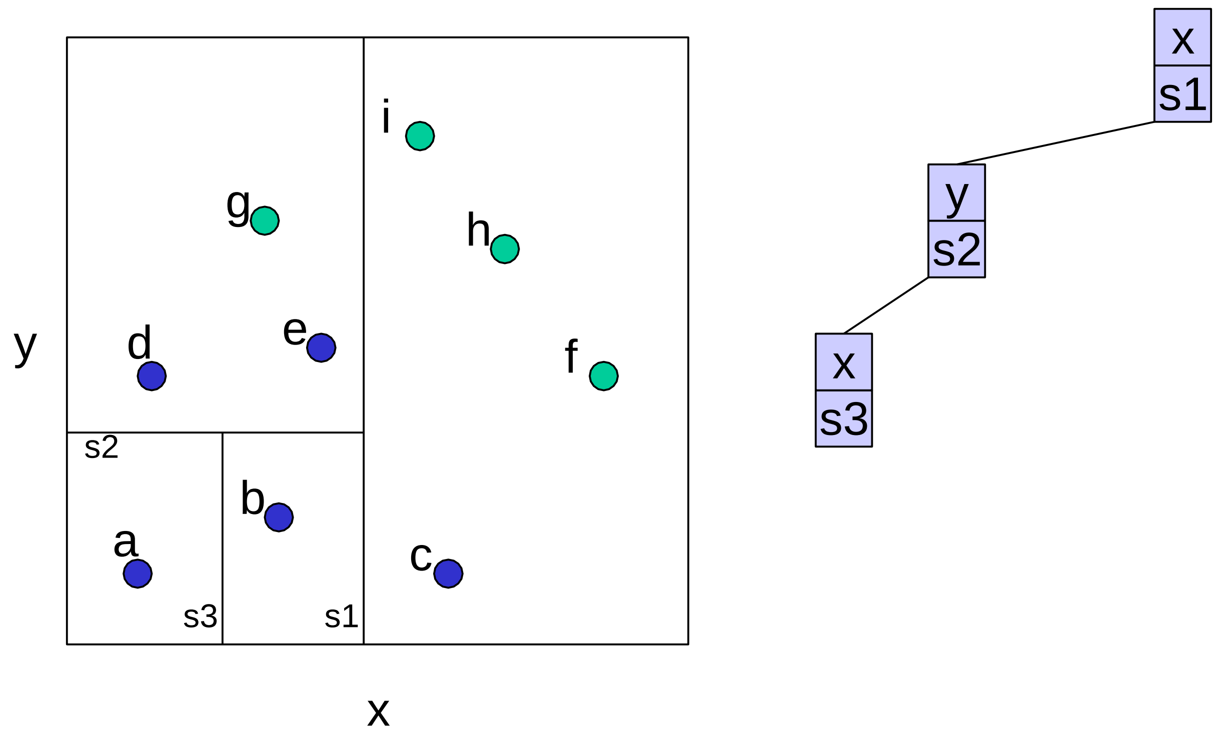


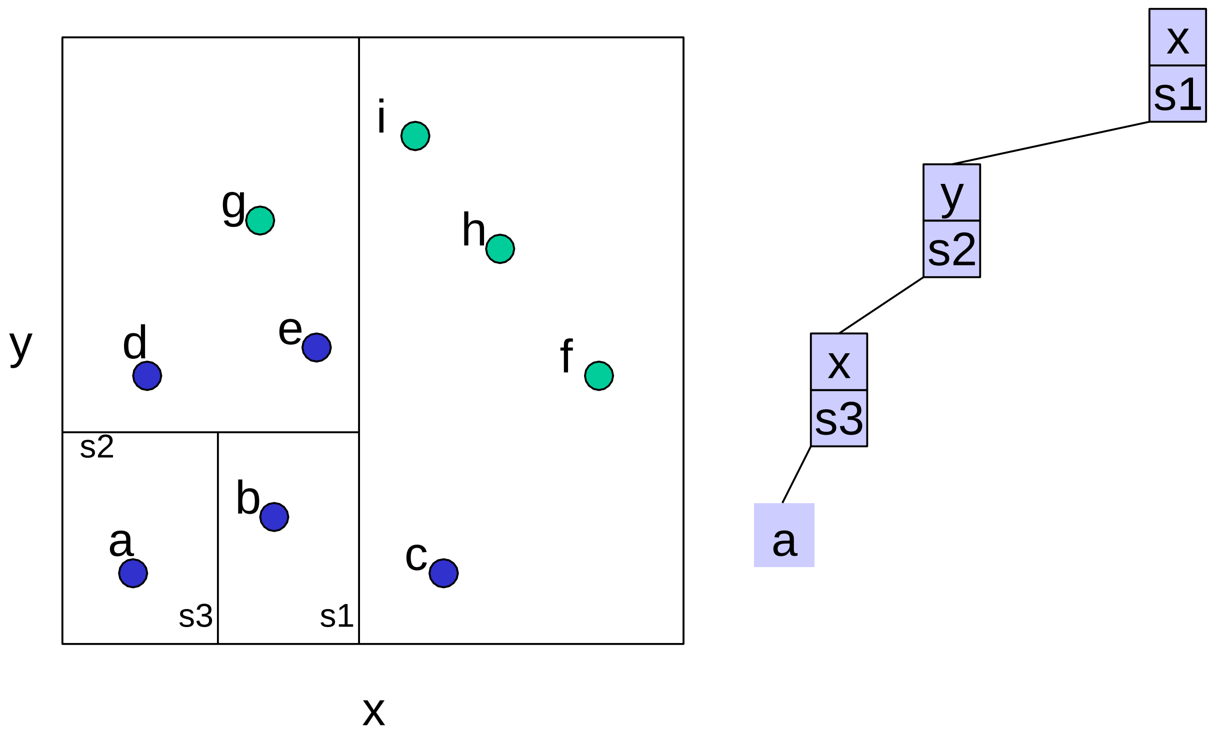

i
g
h
y
d
e
f
b
a
c
s1
x

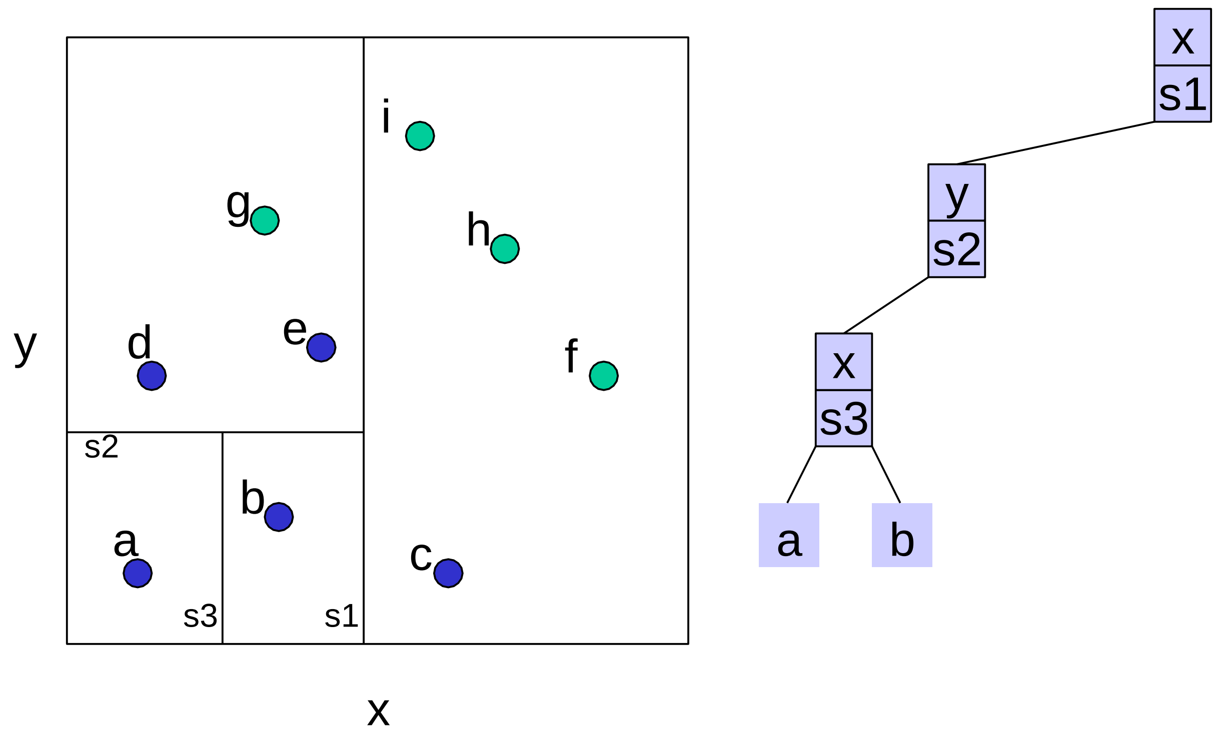

x
s1

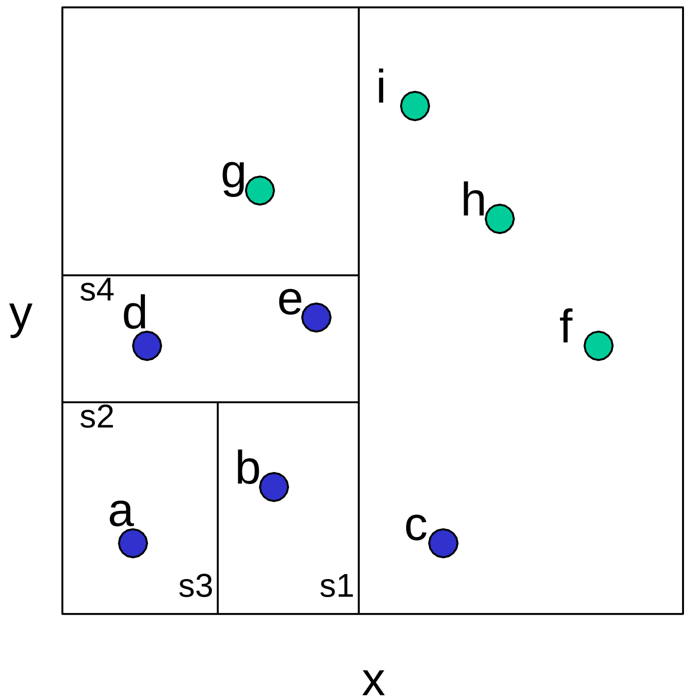

i
g
h
y
d
e
f
s2
b
a
c
s1
x

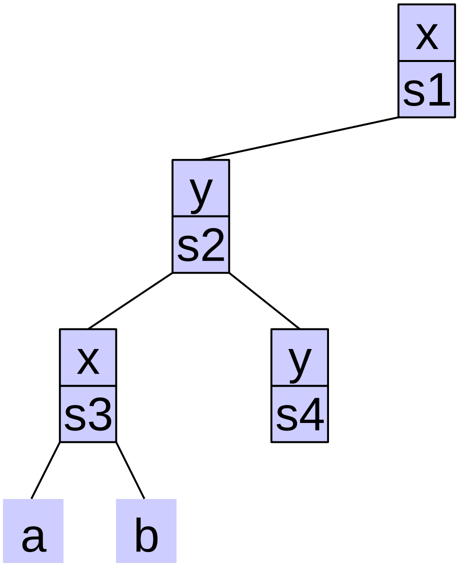

x
s1
y
s2

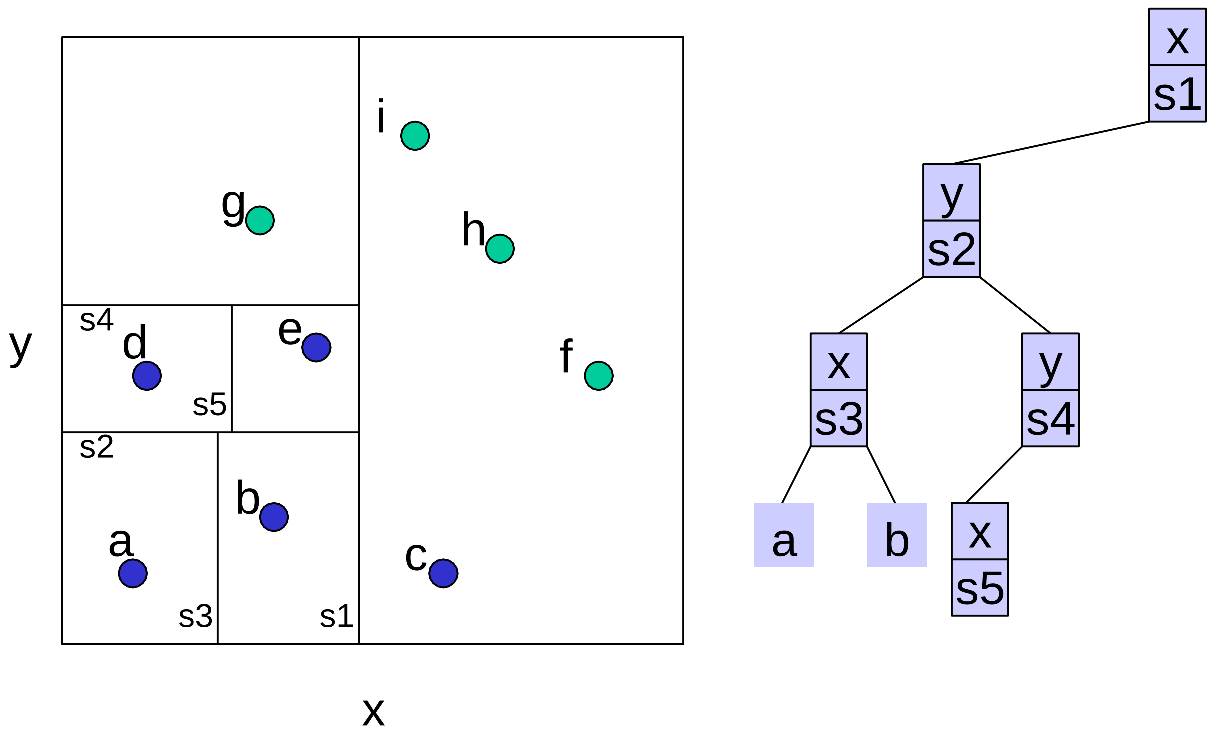

i
g
h
d
e
f
y
s2
b
a
c
s3
s1
x
x
s1
y
s2
x
s3

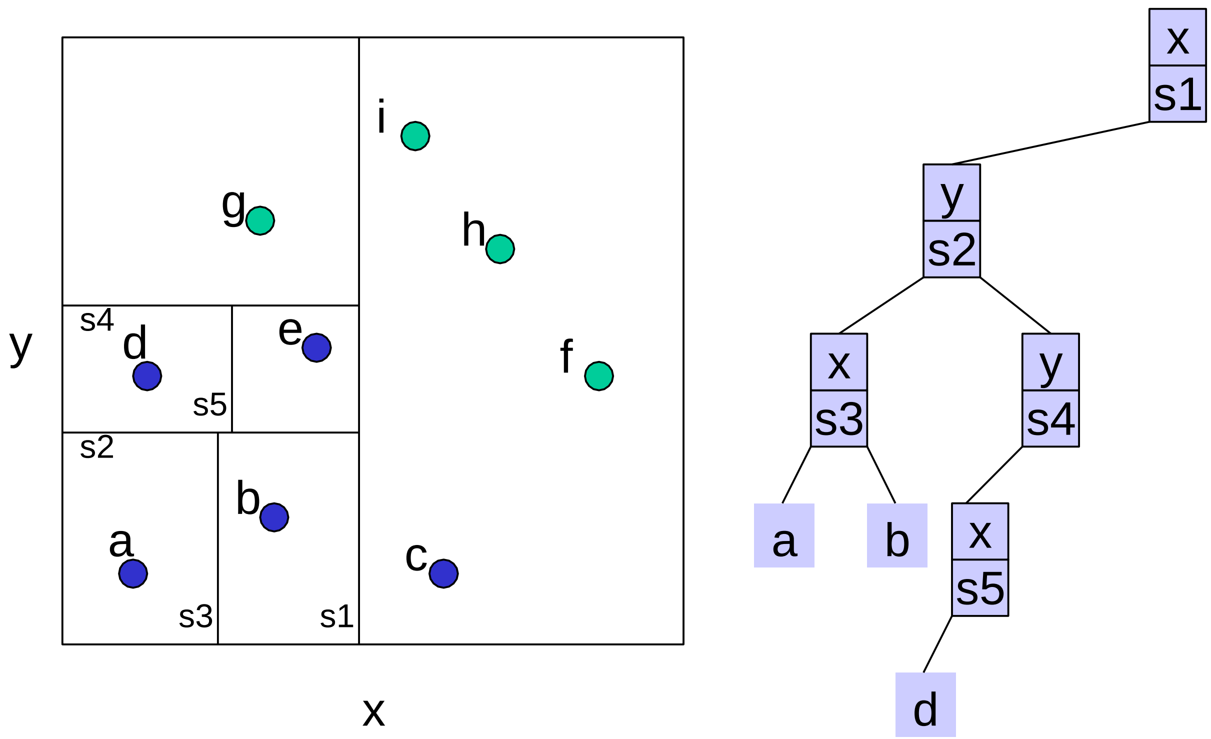

y
x
i
g
h
d
e
f
s2
b
a
c
s3
s1
x
s1
y
s2
x
s3
a

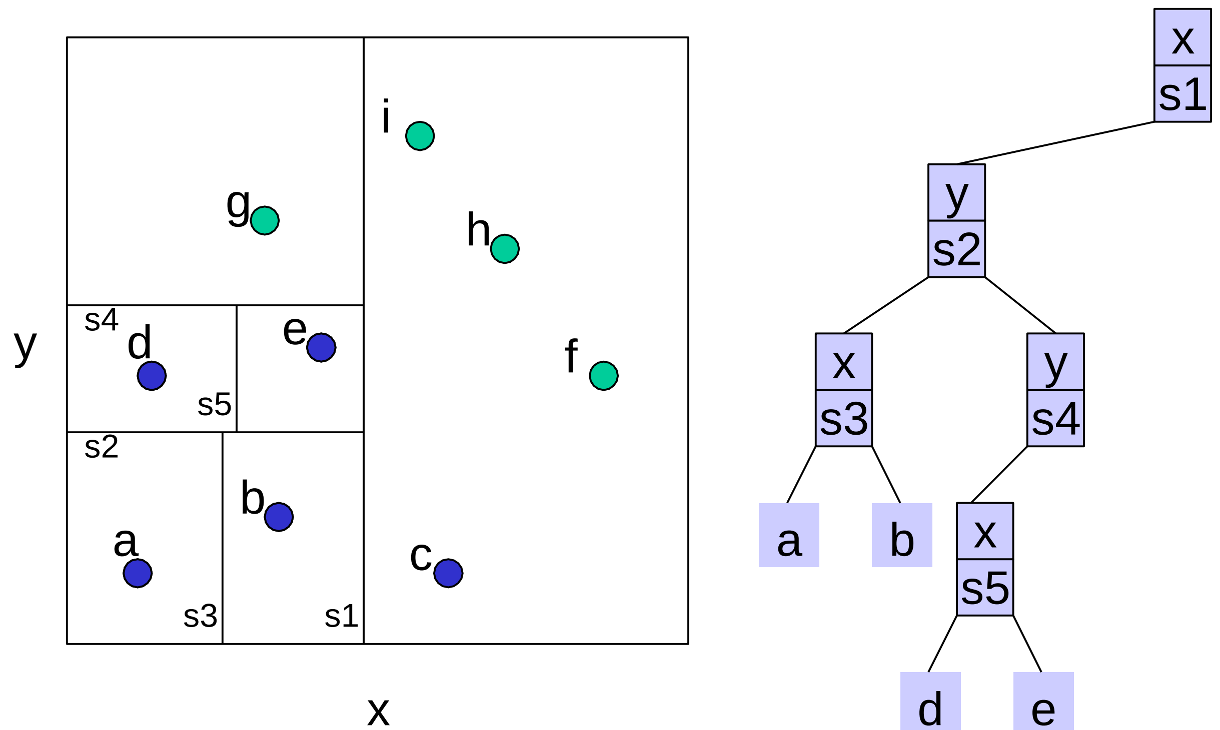

i
g
h
y
d
e
f
s2
b
a
c
s3
s1
x
x
s1
y
s2
x
s3
a
b

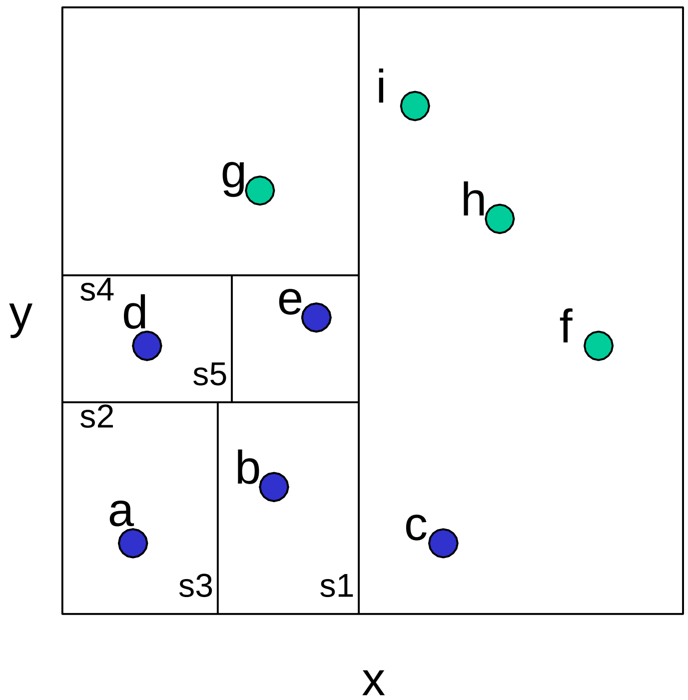

i
g
h
s4
d
e
y
f
s2
b
a
c
s3
s1
x

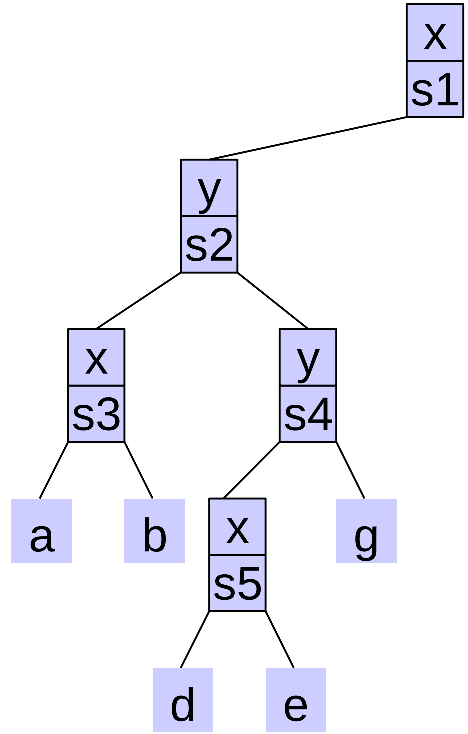

x
s1
y
s2
x
s3
y
s4
a
b

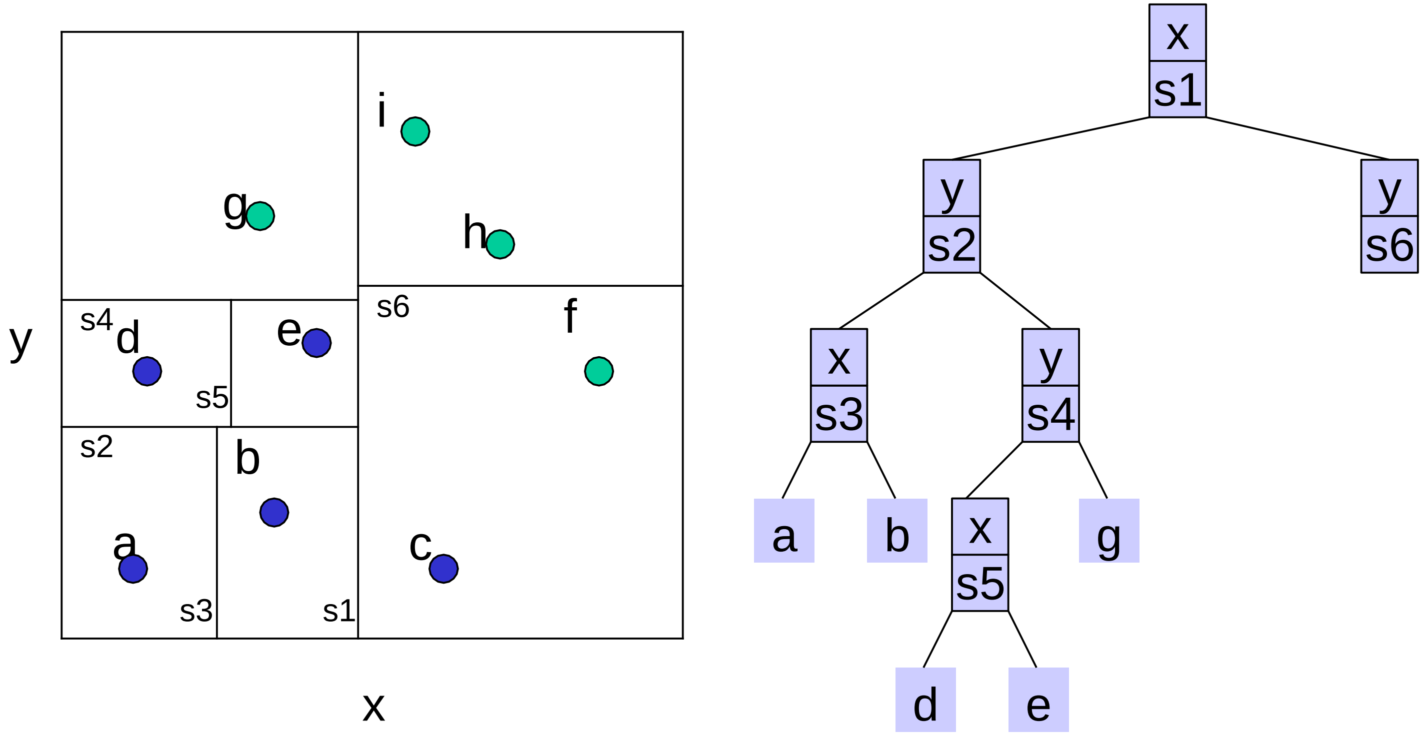

i
g
h
f
e
d
s4
s5
s2
b
a
c
s3
s1
y
x
x
s1
y
s2
x
s3
y
s4
a
b
x
s5

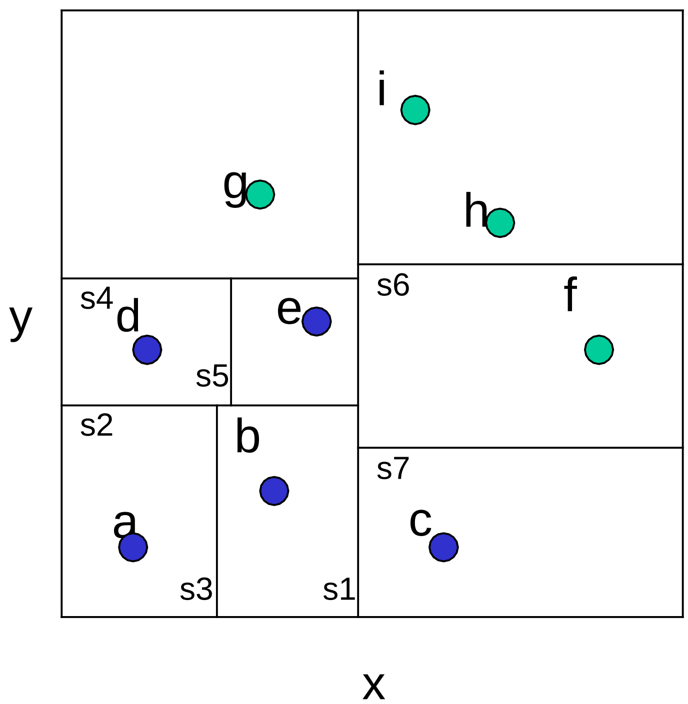

i
g
h
f
e
d
s4
s5
s2
b
a
c
s3
s1
y
x
x
s1
y
s2
x
s3
y
s4
a
b
x
s5
d

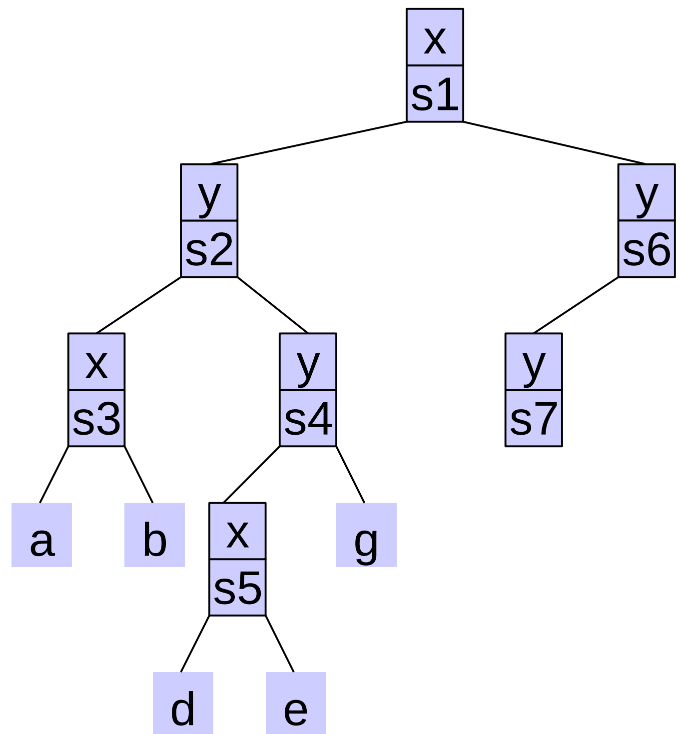

y
x
i
g
h
s4
d
e
f
s5
s2
b
a
c
s3
s1
x
s1
y
s2
x
s3
y
s4
a
b
x
s5
d
e

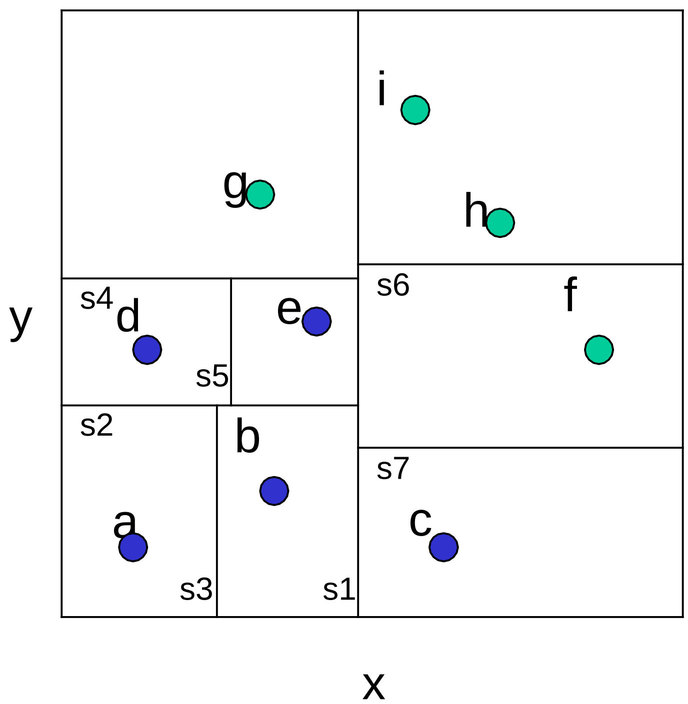

i
g
h
y
s4
d
e
f
s5
s2
b
a
c
s3
s1
x

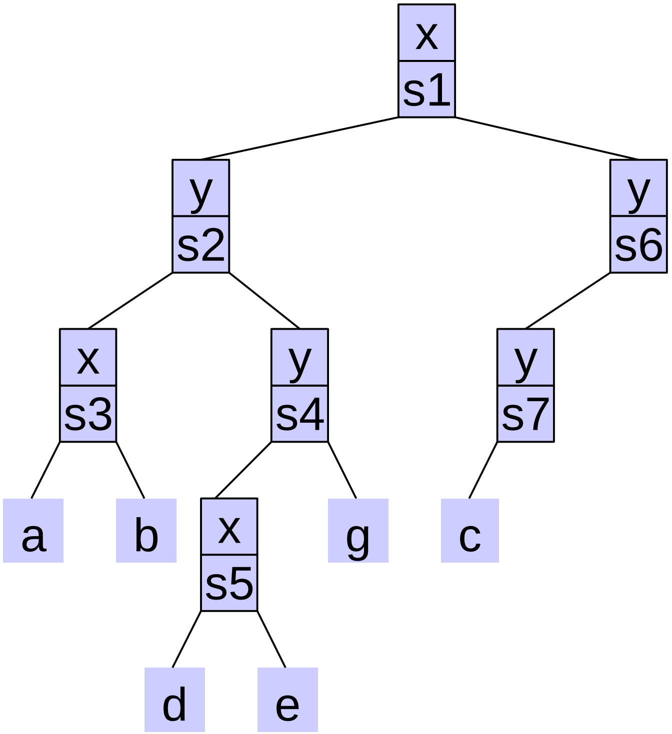

x
s1
y
s2
x
s3
y
s4
a
b
x
s5
g
d
e

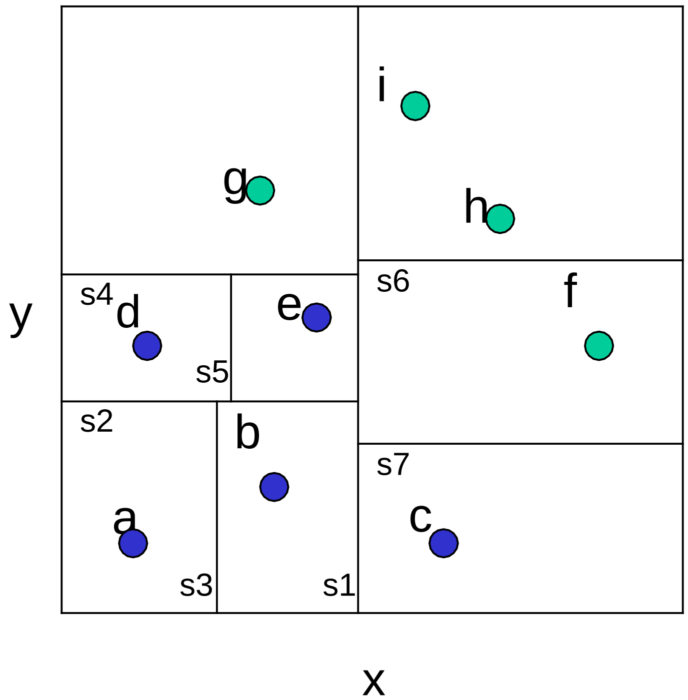

i
g
h
s4
d
e
s6
f
s5
y
s2
b
a
c
s3
s1
x
x
s1
y
s2
y
s6
x
s3
y
s4
a
b
x
s5
g
d
e

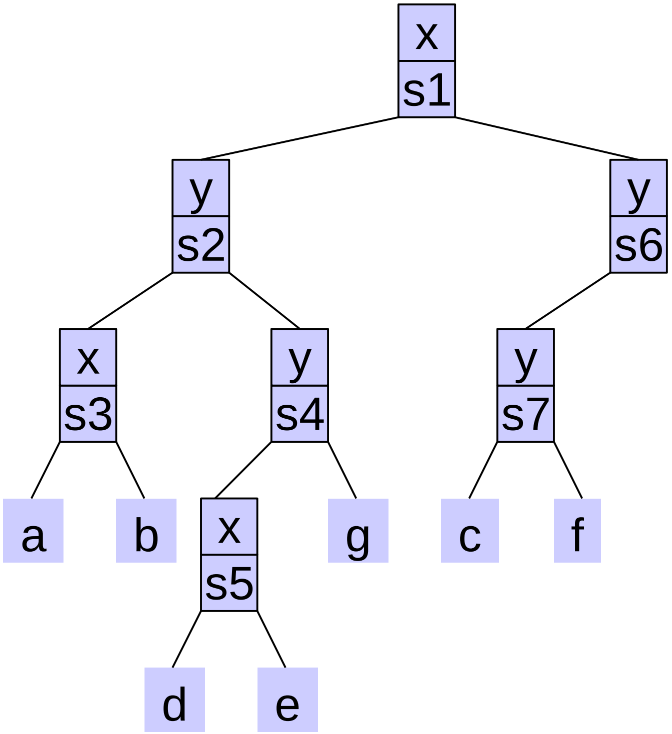

i
g
h
s6
f
s4
d
e
s5
y
s2
b
s7
c
a
s3
s1
x

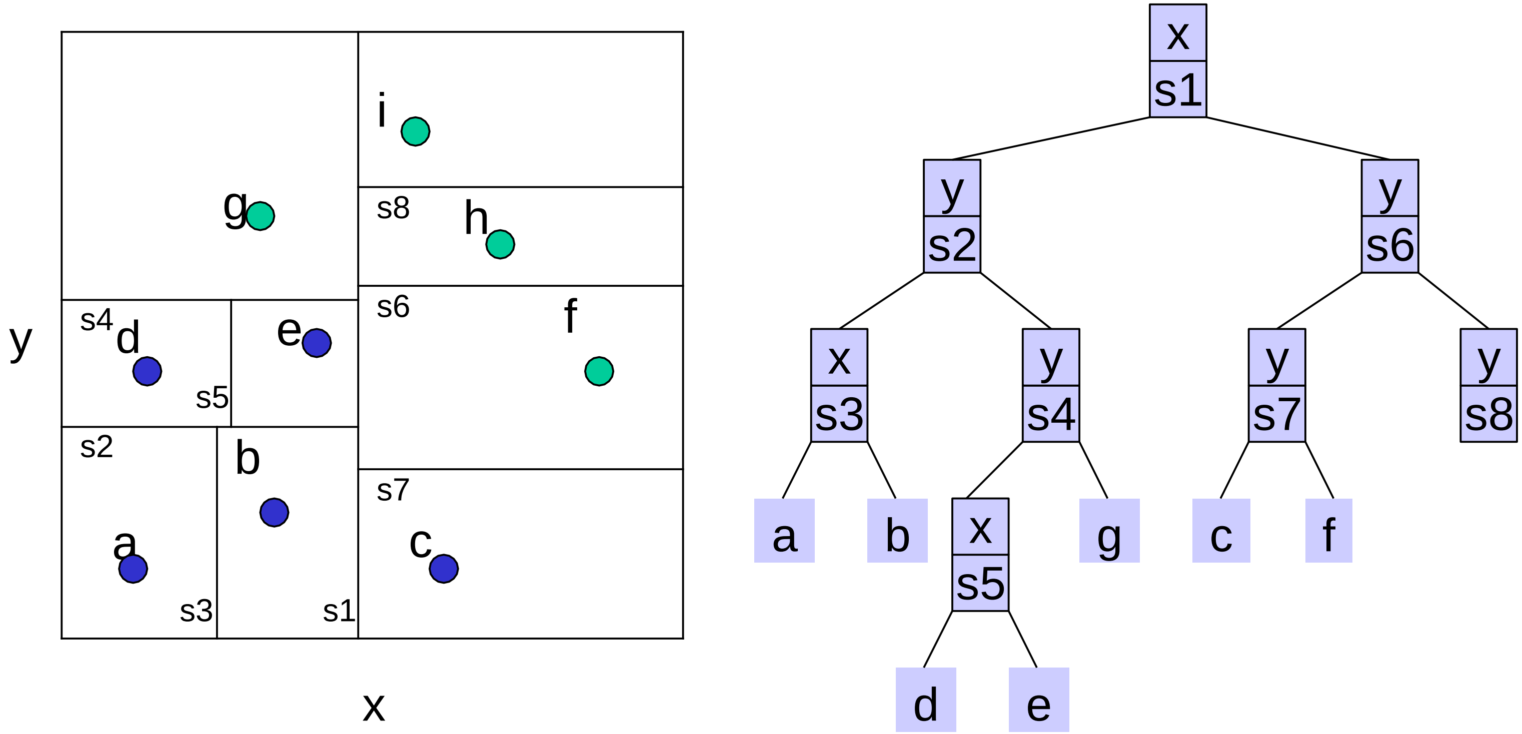

x
s1
y
s2
y
s6
x
s3
y
s4
y
s7
a
b
x
s5
g
d
e

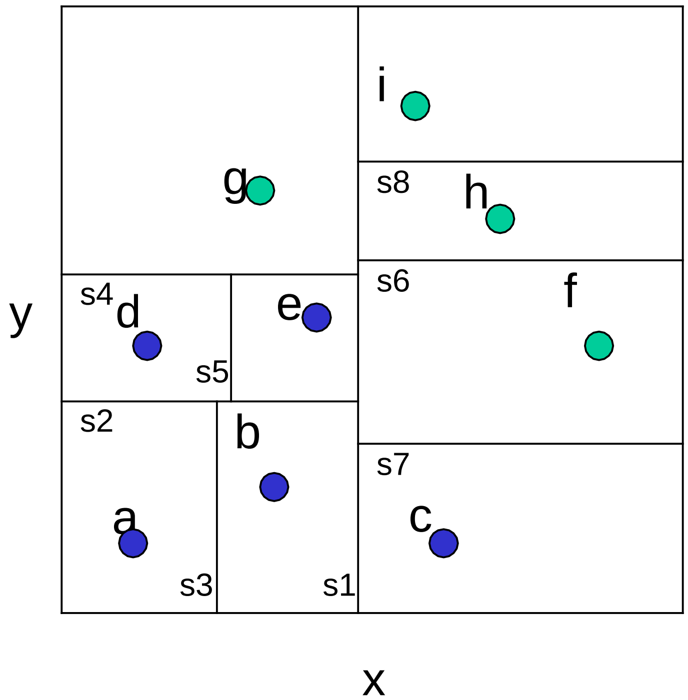

i
g
h
s4
d
e
s6
f
y
s5
s2
b
s7
a
c
s3
s1
x

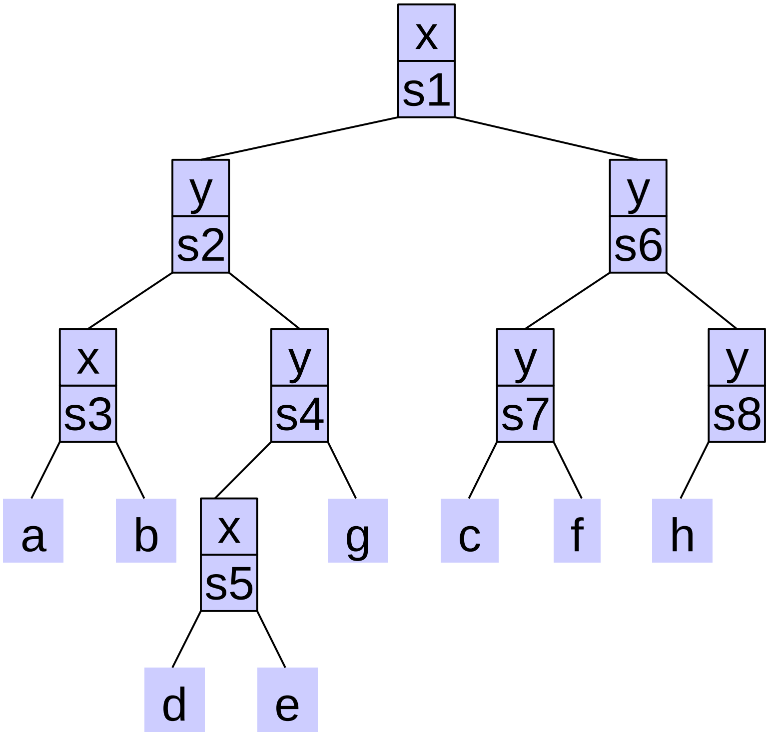

x
s1
y
s2
y
s6
x
s3
y
s4
y
s7
a
b
x
s5
g
c
d
e

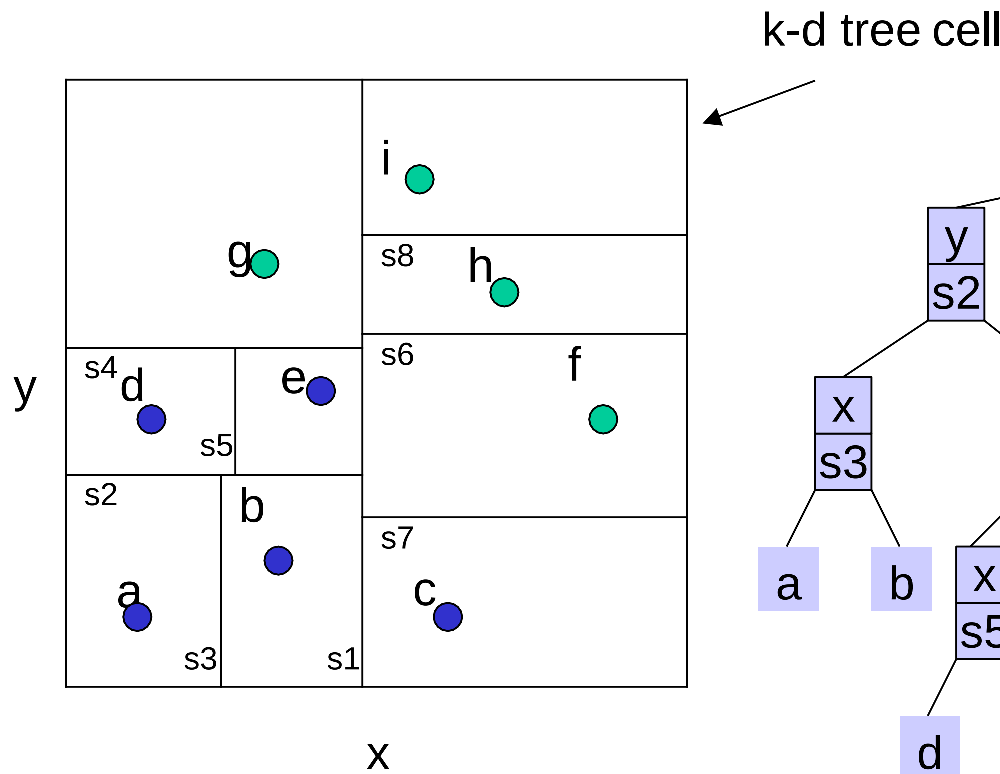

i
g
h
s4
d
e
s6
f
y
s5
s2
b
s7
a
c
s3
s1
x

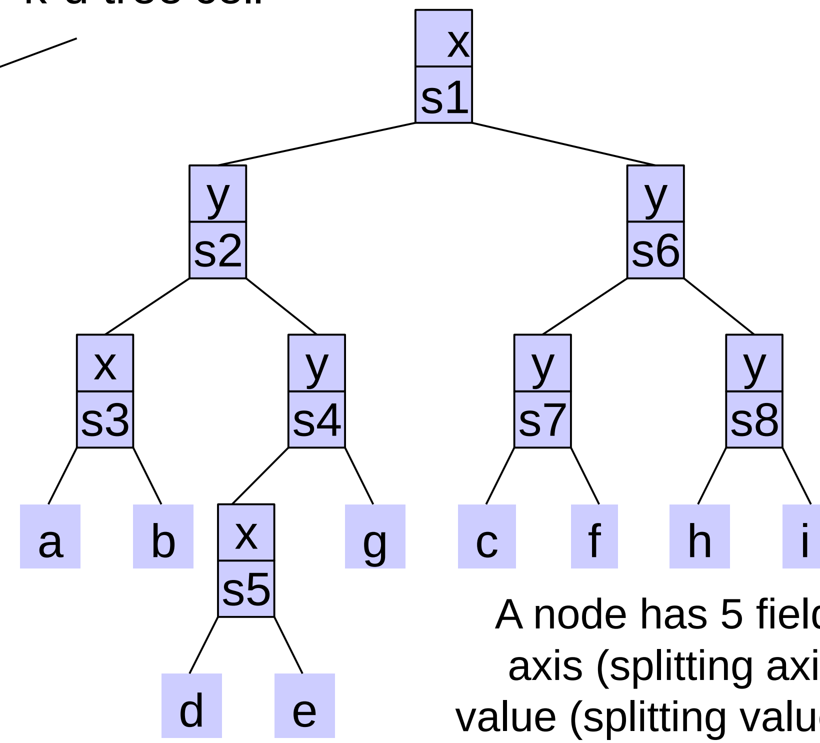

x
s1
y
s2
y
s6
x
s3
y
s4
y
s7
a
b
x
s5
g
c
f
d
e


y
x
i
g
s8
h
s6
f
s4
d
e
s5
s2
b
s7
c
a
s3
s1
x
s1
y
s2
y
s6
x
s3
y
s4
y
s7
y
s8
a
b
x
s5
g
c
f
d
e


i
g
s8
h
s4
d
e
s6
f
y
s5
s2
b
s7
a
c
s3
s1
x


x
s1
y
s2
y
s6
x
s3
y
s4
y
s7
y
s8
a
b
x
s5
g
c
f
h
d
e





A node has 5 fields
axis (splitting axis)
value (splitting value)
left (left subtree)
right (right subtree)
point (holds a point if left and right children are null)

构造策略

- K-D 树的构造策略与二维的情况类似
- 在根节点，根据各个维度的分布情况，选择与x_1-坐标轴垂直的超平面将样本分成大小近似相等的两个子集
- 在其他子节点中根据当前子集的分布情况，选择x_2-坐标轴进行划分
- 循环这个过程，直到无法划分，存储数据为叶子结点。

问题1：每次对子空间的划分时，怎样确定在哪个维度上进行划分？

当前最大区间长度的维度, 最大方差，交替选择

问题2：在某个维度上进行划分时，怎样确保在这一维度上的划分得到的两个子集合的数量尽量相等，即左子树和右子树中的结点个数尽量相等？

中位数，区间中点

KD树

- 构造kd树：
- 对深度为j的节点，选择x^l为切分的坐标轴 $l = j(\bmod k)+1$
- 例： $T=\{(2,3)^T,(5,4)^T,(9,6)^T,(4,7)^T,(8,1)^T,(7,2)^T\}$

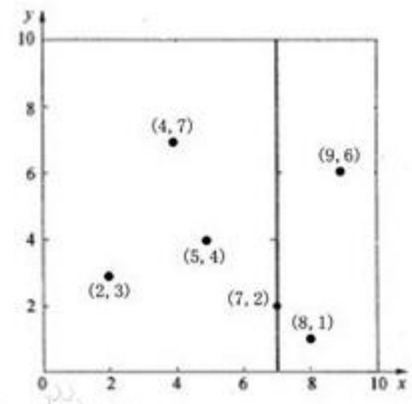


图2 x=7将整个空间分为两部分

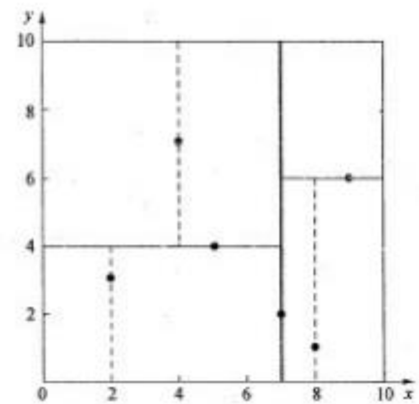
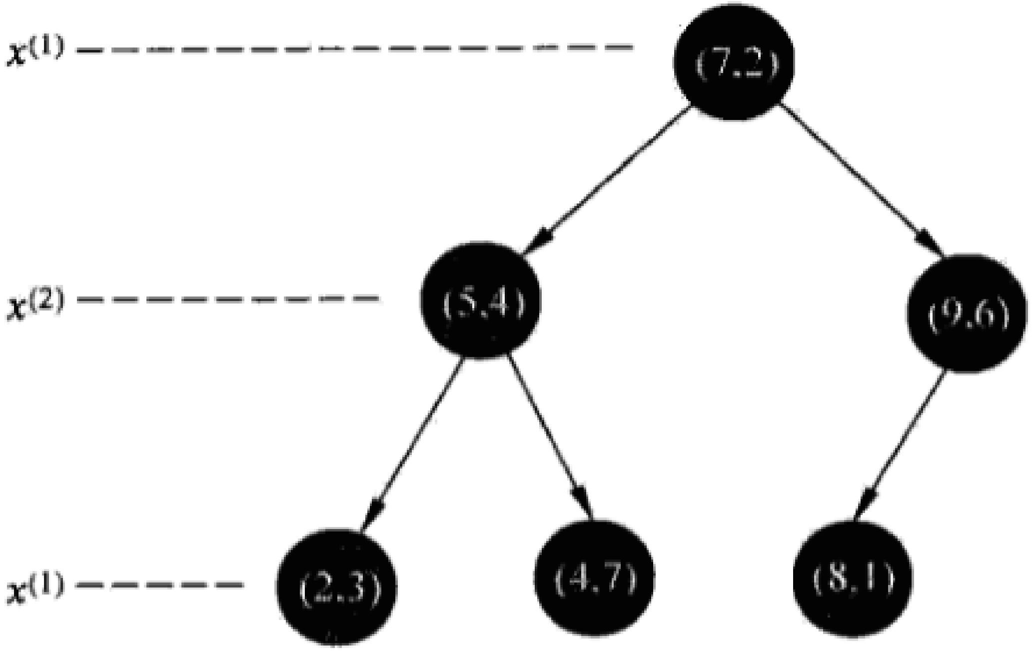


图1 二维数据k-d树空间划分示意图

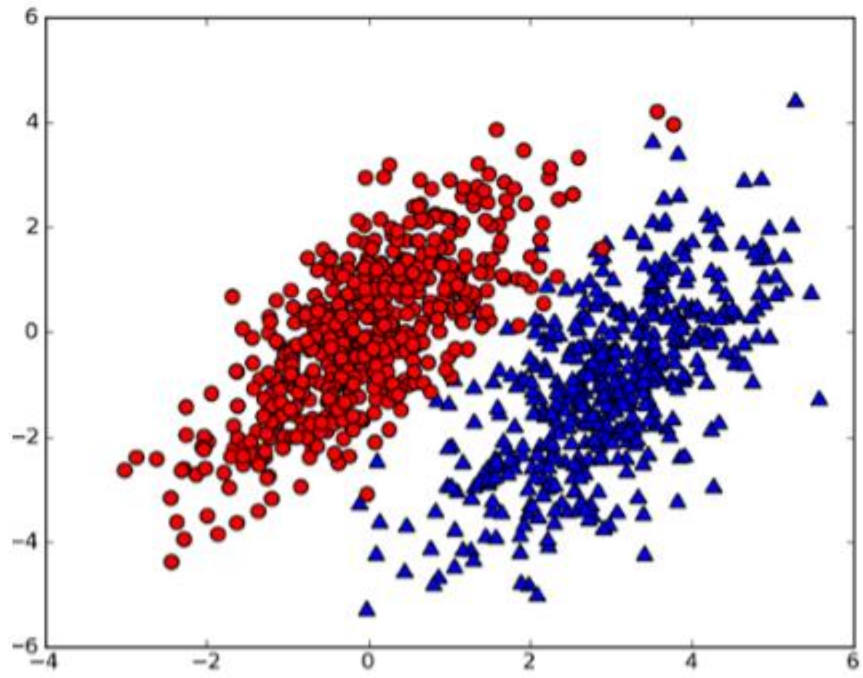
KD 树

- {(2,3),(5,4),(9,6),(4,7),(8,1),(7,2)},
- 建立索引



贝叶斯分类器

问题的提出



KNN ?

决策树?

概率方法?

贝叶斯

- 贝叶斯(约1701-1761) Thomas Bayes，英国数学家。约1701年出生于伦敦，做过神甫。1742年成为英国皇家学会会员。1761年4月7日逝世。贝叶斯在数学方面主要研究概率论。他首先将归纳推理法用于概率论基础理论，并创立了贝叶斯统计理论，对于统计决策函数、统计推断、统计的估算等做出了贡献。他死后，理查德·普莱斯(Richard Price)于1763年将他的著作《机会问题的解法》(An essay towards solving a problem in the doctrine of chances)寄给了英国皇家学会，对于现代概率论和数理统计产生了重要的影响

贝叶斯

- 贝叶斯决策就是在不完全情报下，对部分未知的状态用主观概率估计，然后用贝叶斯公式对发生概率进行修正，最后再利用期望值和修正概率做出最优决策。
- 贝叶斯决策理论方法是统计模型决策中的一个基本方法，其基本思想是：
- 1、已知类条件概率密度参数表达式和先验概率。
- 2、利用贝叶斯公式转换成后验概率。
- 3、根据后验概率大小进行决策分类。

贝叶斯网络的应用

- 最早的PathFinder系统，该系统是淋巴疾病诊断的医学系统，它可以诊断60多种疾病，涉及100多种症状;后来发展起来的Internist－I系统，也是一种医学诊断系统，但它可以诊断多达600多种常见的疾病。

- 1995年，微软推出了第一个基于贝叶斯网的专家系统，一个用于幼儿保健的网站OnParent (www.onparenting.msn.com)，使父母们可以自行诊断。

贝叶斯网络的应用

- (1)故障诊断(diagnose)
- (2)专家系统(expert system)
- (3)规划(planning)
- (4)学习(learning)
- (5)分类(classifying)

概率(回顾)

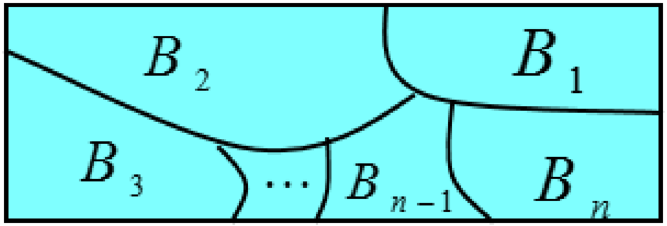
样本空间的划分

定义　设Ω为试验E的样本空间, $B_1, B_2, \cdots, B_n$ 为E的一组事件, 若

1^0　$B_iB_j = \varnothing, i, j = 1, 2, \cdots, n;$

2^0　$B_1 \cup B_2 \cup \cdots \cup B_n = \Omega,$

则称 $B_1, B_2, \cdots, B_n$ 为样本空间Ω的一个划分.



概率(回顾)

全概率公式

定义 设Ω为试验E的样本空间, A为E的事件,

$B_1, B_2, \cdots, B_n$为Ω的一个划分, 且$P(B_i) > 0$

$(i = 1, 2, \cdots, n)$, 则

$$\boxed{\begin{aligned} P(A) &= P(A \mid B_1)P(B_1) + P(A \mid B_2)P(B_2) \\ &\quad + \cdots + P(A \mid B_n)P(B_n) \\ &= \sum_{i=1}^{n} P(B)P(A \mid B_i) \end{aligned}}$$

基本方法

- 训练数据集： $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$

- 由X和Y的联合概率分布P(X,Y)独立同分布产生

- 朴素贝叶斯通过训练数据集学习联合概率分布P(X,Y)，
  - 即先验概率分布： $P(Y=c_k)$， $k=1,2,\cdots,K$
  - 及条件概率分布：
    $$P(X=x\mid Y=c_k)=P(X^{(1)}=x^{(1)},\cdots,X^{(n)}=x^{(n)}\mid Y=c_k),\quad k=1,2,\cdots,K$$
  - 注意：条件概率为指数级别的参数： $K\prod_{j=1}^{n}S_j$

基本方法

- 条件独立性假设：

$$P(X=x\mid Y=c_k)=P(X^{(1)}=x^{(1)},\cdots,X^{(n)}=x^{(n)}\mid Y=c_k)$$
$$=\prod_{j=1}^{n}P(X^{(j)}=x^{(j)}\mid Y=c_k)$$

- “朴素”贝叶斯名字由来，牺牲分类准确性。

- 贝叶斯定理：

$$P(Y=c_k\mid X=x)=\frac{P(X=x\mid Y=c_k)P(Y=c_k)}{\sum_k P(X=x\mid Y=c_k)P(Y=c_k)}$$

- 代入上式：

$$P(Y=c_k\mid X=x)=\frac{P(Y=c_k)\prod_j P(X^{(j)}=x^{(j)}\mid Y=c_k)}{\sum_k P(Y=c_k)\prod_j P(X^{(j)}=x^{(j)}\mid Y=c_k)}$$

基本方法

- 贝叶斯分类器：

$$y=f(x)=\arg\max_{c_k}\frac{P(Y=c_k)\prod_j P(X^{(j)}=x^{(j)}\mid Y=c_k)}{\sum_k P(Y=c_k)\prod_j P(X^{(j)}=x^{(j)}\mid Y=c_k)}$$

- 分母对所有c_k都相同：

$$y=\arg\max_{c_k}P(Y=c_k)\prod_j P(X^{(j)}=x^{(j)}\mid Y=c_k)$$

后验概率最大化的含义：

- 朴素贝叶斯法将实例分到后验概率最大的类中，等价于期望风险最小化，
- 假设选择0-1损失函数：f(X)为决策函数

$$L(Y, f(X)) = \begin{cases} 1, & Y \neq f(X) \\ 0, & Y = f(X) \end{cases}$$

- 期望风险函数： $R_{\exp}(f) = E[L(Y, f(X))]$

- 取条件期望： $R_{\exp}(f) = E_X \sum_{k=1}^{K} [L(c_k, f(X))] P(c_k \mid X)$

后验概率最大化的含义：

- 只需对X=x逐个极小化，得：

$$
\begin{aligned}
f(x) &= \arg\min_{y\in\mathcal{Y}} \sum_{k=1}^{K} L(c_k, y) P(c_k \mid X = x) \\
&= \arg\min_{y\in\mathcal{Y}} \sum_{k=1}^{K} P(y \neq c_k \mid X = x) \\
&= \arg\min_{y\in\mathcal{Y}} (1 - P(y = c_k \mid X = x)) \\
&= \arg\max_{y\in\mathcal{Y}} P(y = c_k \mid X = x)
\end{aligned}
$$

- 推导出后验概率最大化准则： $f(x) = \arg\max_{c_k} P(c_k \mid X = x)$

朴素贝叶斯法的参数估

- 应用极大似然估计法估计相应的概率：
- 先验概率P(Y=c$_k$)的极大似然估计是： $$P(Y=c_k)=\frac{\sum_{i=1}^{N} I(y_i=c_k)}{N},\quad k=1,2,\cdots,K$$
- 设第j个特征x$^{(j)}$可能取值的集合为： $\{a_{j1},a_{j2},\cdots,a_{jS_j}\}$
- 条件概率的极大似然估计：

$$P(X^{(j)}=a_{jl}\mid Y=c_k)=\frac{\sum_{i=1}^{N} I(x_i^{(j)}=a_{jl},y_i=c_k)}{\sum_{i=1}^{N} I(y_i=c_k)}$$

$$j=1,2,\cdots,n;\quad l=1,2,\cdots,S_j;\quad k=1,2,\cdots,K$$

朴素贝叶斯法的参数估计

- 学习与分类算法Naïve Bayes Algorithm：
- 输入：
  - 训练数据集 $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$

$x_i^{(j)}$
  - 第i个样本的第j个特征 $x_i=(x_i^{(1)},x_i^{(2)},\cdots,x_i^{(n)})^{\mathrm{T}}$

a_{jl}
  - 第j个特征可能取的第l个值 $x_i^{(j)}\in\{a_{j1},a_{j2},\cdots,a_{jS_j}\}$

- 输出： $y_i\in\{c_1,c_2,\cdots,c_K\}$
  - x的分类

朴素贝叶斯法的参数估

- 步骤
  - 1、计算先验概率和条件概率

$$P(Y=c_k)=\frac{\sum_{i=1}^{N} I(y_i=c_k)}{N}, \quad k=1,2,\cdots,K$$

$$P(X^{(j)}=a_{jl} \mid Y=c_k)=\frac{\sum_{i=1}^{N} I(x_i^{(j)}=a_{jl}, y_i=c_k)}{\sum_{i=1}^{N} I(y_i=c_k)}$$

$$j=1,2,\cdots,n; \quad l=1,2,\cdots,S_j; \quad k=1,2,\cdots,K$$

朴素贝叶斯法的参数估

- 步骤
  - 2、对于给定的实例 $x=(x^{(1)},x^{(2)},\cdots,x^{(n)})^{\mathrm{T}}$
  - 计算

$$P(Y=c_k)\prod_{j=1}^{n}P(X^{(j)}=x^{(j)}\mid Y=c_k),\quad k=1,2,\cdots,K$$

  - 3、确定x的类别

$$y=\arg\max_{c_k}P(Y=c_k)\prod_{j=1}^{n}P(X^{(j)}=x^{(j)}\mid Y=c_k)$$

朴素贝叶斯网络的缺陷

- 考虑几个问题：
- 1、如果属性之间不相互独立？
- 2、如果属性A和属性B都很重要，但是相关？
- 3、如果属性A，属性B之间独立，但是在属性C下有关？
- 4、属性之间的条件概率究竟有多少个？
- 5、条件概率谬论？

第五次作业

1、什么是模型的泛化能力?

2、简述梯度下降法的思想和内容。

3、简述感知机的原理。

4、简述K近邻的原理。

5、简述朴素贝叶斯的原理。

6、至少实现3-5中的一种方法的实例。