



## 第六章 异常检测

## 目录 CONTENTS

6.1 异常检测的基本概念

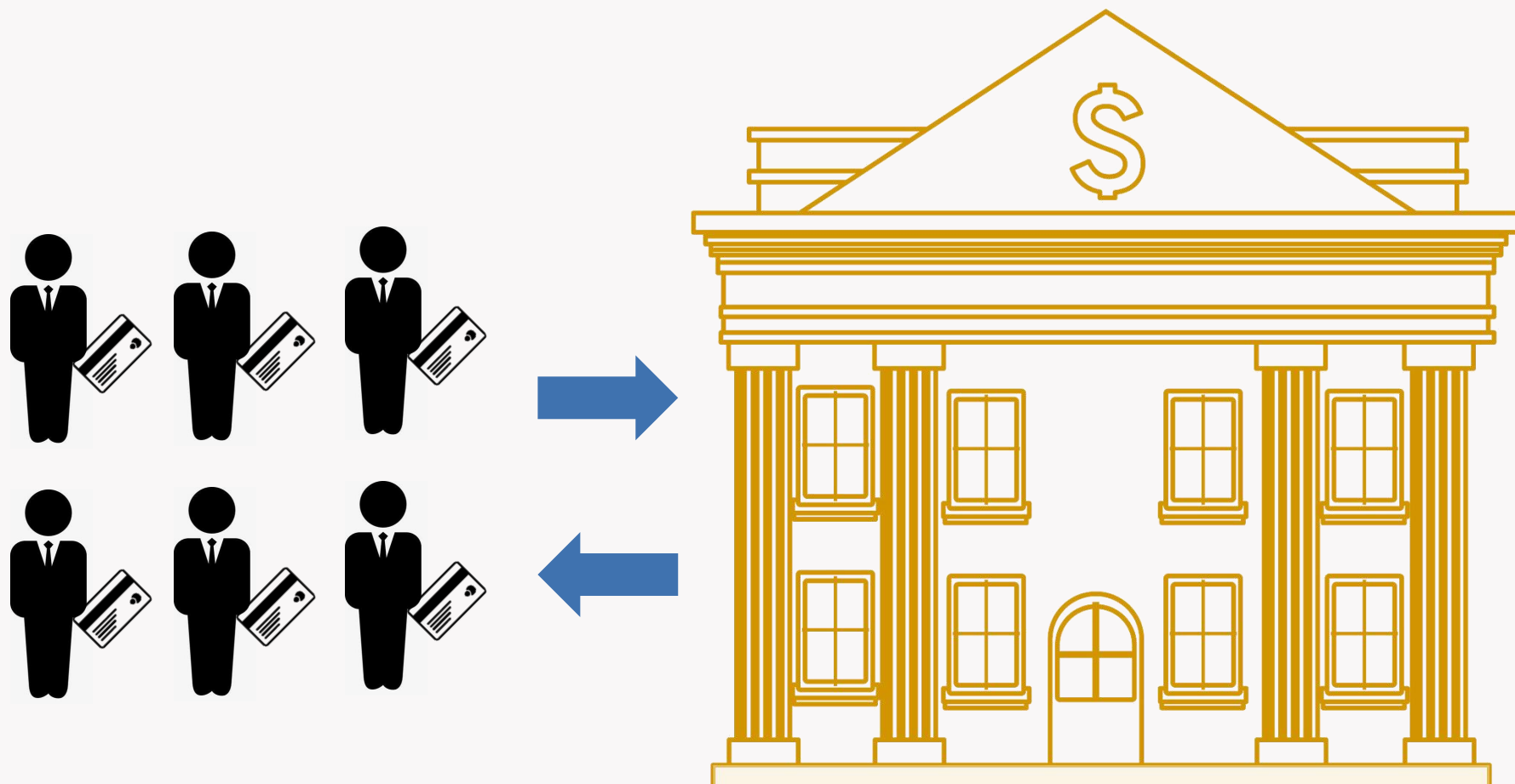
6.2 异常检测方法



# Chapter 6.1

## 异常检测的基本概念

## □ 信用卡欺诈检测



## 异常检测的定义

■ **异常检测** (Anomaly detection) 指的是识别不符合预期行为的对象的过程，或者是找出与其余数据不相似的、不寻常的事件、样本或行为的过程。

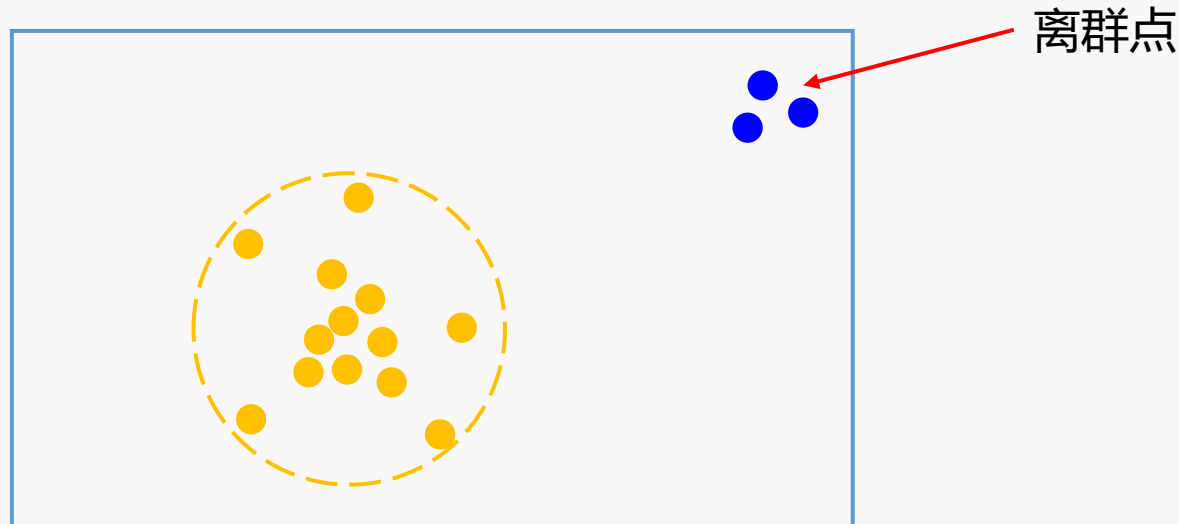
■ 异常检测还有以下名称：

- 离群点检测 (Outlier detection)
- 新颖性检测 (Novelty detection)
- 偏差检测 (Deviation detection)



## □ 离群点的定义

- **离群点** (Outlier) 指的是不同于其他数据对象（正常的训练样本）的数据对象，仿佛离群点是由不同的机制（分布）产生的一样。



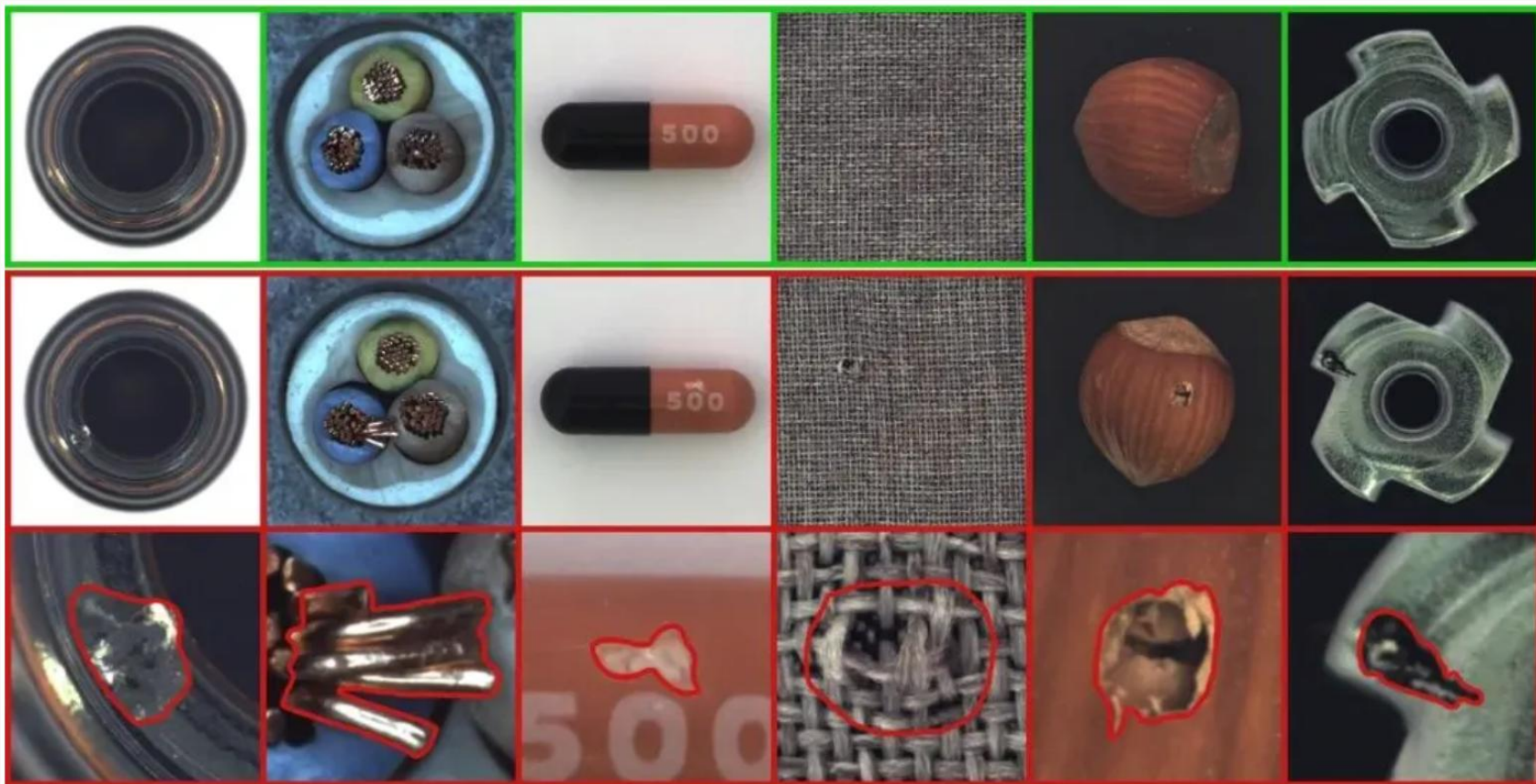


## □ 离群点与噪声的区别

- 在非异常检测任务中，离群点和噪声通常不做区分。但在异常检测中，离群点和噪声是两个**不同**概念：
  - 噪声指的是被观测变量的随机误差或方差，通常不令人感兴趣且需要在数据清理时被清除；
  - 离群点的出现可能暗示着存在不同的数据产生机制，所以离群点是有趣的。

## 异常检测的应用

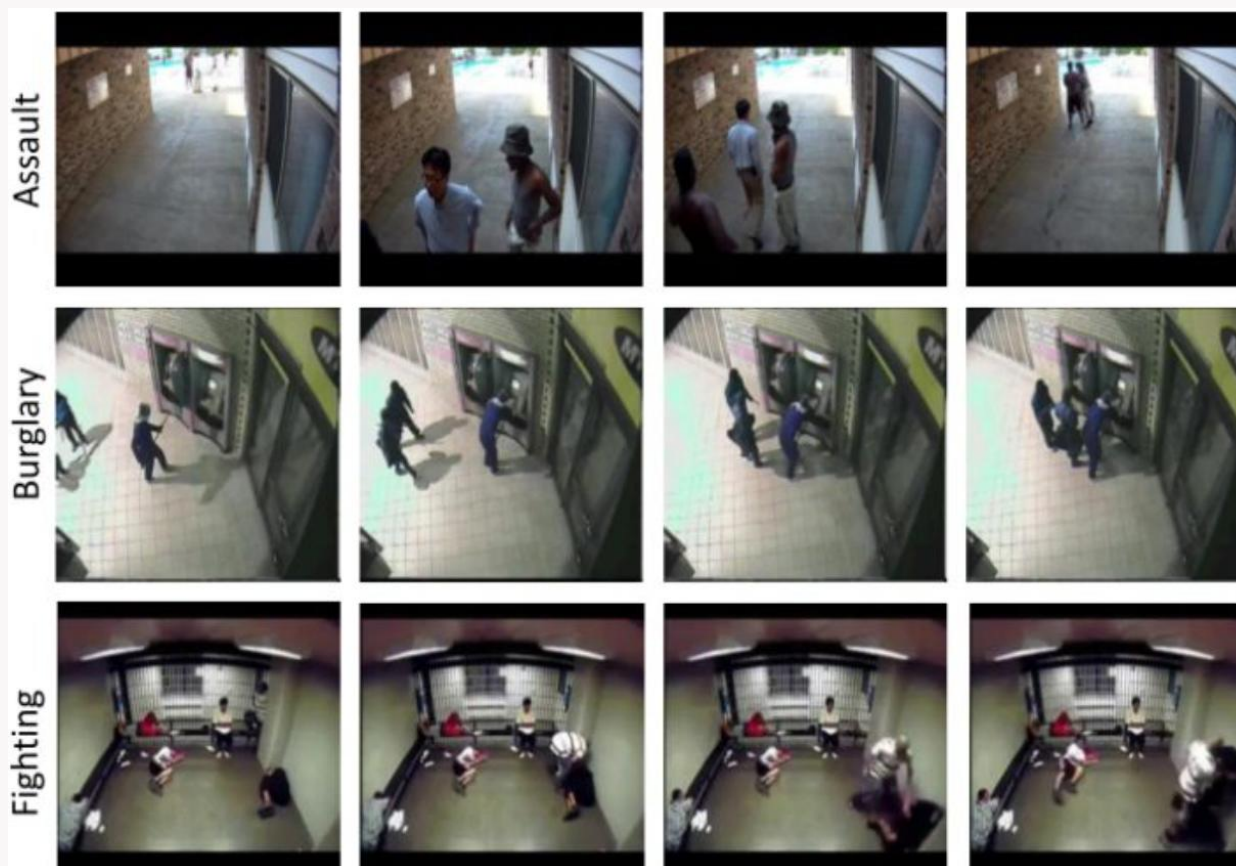
### 工业异常检测





## 异常检测的应用

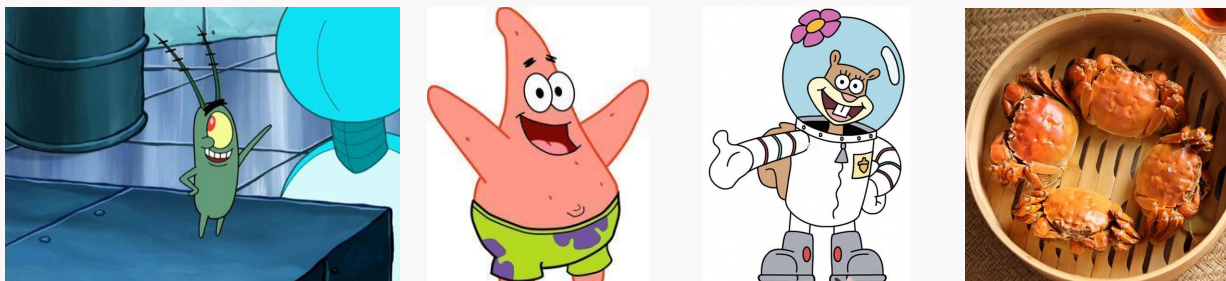
### 监控异常检测



## □ 异常检测的应用

- 网络安全：监测网络流量，检测恶意攻击或入侵行为；
- 城市规划：检测交通流量、城市设施的异常情况；
- 舆情监测：检测社交媒体中的异常信息或情绪，用于舆情分析和预警；
- 市场营销：监测市场营销数据，检测异常的消费者行为、新兴的商品销售趋势。

## □ 解决异常检测



## □ 异常检测的难点

- **异常数据缺失**：离群点可能是未知的，比如说新型电信诈骗、恐怖袭击和网络入侵。
- **数据不平衡**：真实异常通常是数据集中的极少部分，类别不平衡使得模型难以准确捕捉异常模式。
- **噪声与异常重叠**：有时异常样本可能与噪声相似，或者噪声本身可能被误认为异常。
- **离群点的定义**：离群点没有客观的定义，甚至在某些领域正常数据和离群点并没有明确的界限。

## □ 离群点类型

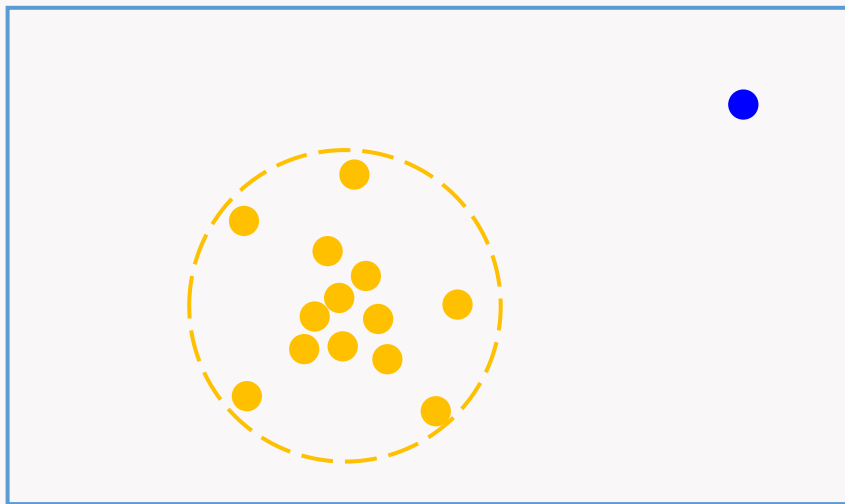
### ■ 离群点可以分为三类：

- 全局离群点
- 情境离群点
- 集体离群点



## □ 全局离群点

- 在给定的数据集中，如果**某个**数据对象显著地偏离了数据集中的其余对象，那么这个数据对象被称作**全局离群点**（Global outlier），有时也被称作**点异常**（Point anomalies）。
- 全局离群点是最简单的、最基础的一类离群点，大部分离群点检测方法的目标都是找到全局离群点。



## □ 情境离群点

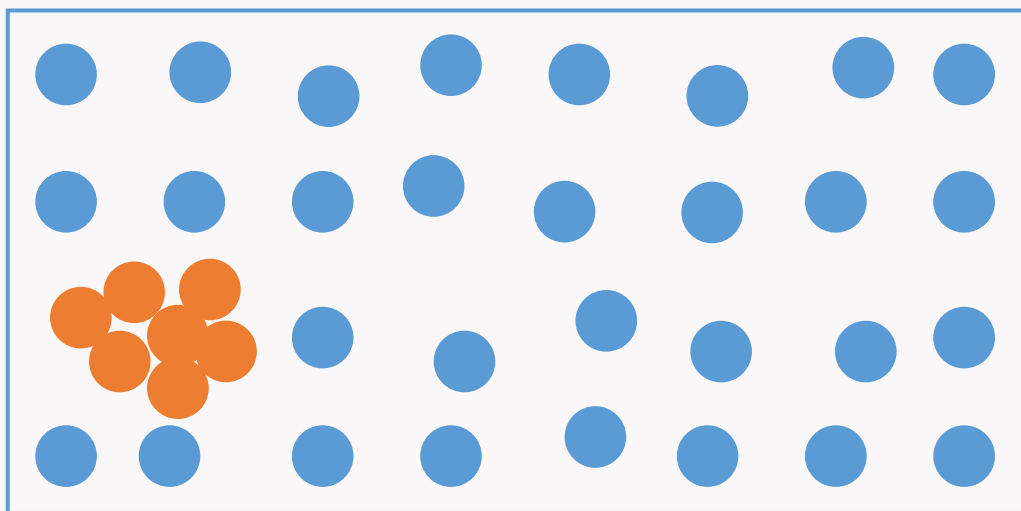
- 在给定的数据集中，**在某个特定情境下**，如果某个数据对象显著地偏离了数据集中的其余对象，那么这个数据对象被称作**情境离群点** (Contextual outlier) 。
- 全局离群点可以看作是情境离群点的特例，此时情境条件为空。

情境属性：情境描述    行为属性：目标的特征

时间	地点	最低温度	最高温度
2022-12-16	南京	-1	5
2022-12-16	哈尔滨	-25	-14
2022-12-16	苏州	0	9

## □ 集体离群点

- 在给定的数据集中，如果**某些**数据对象**作为整体**显著地偏离了数据集中的其余对象，那么这些数据对象被称作**集体离群点**（Global outlier）。
- 集体离群点检测需要同时考虑个体对象的行为和群组对象的行为，所以需要对象关系的背景知识以设置对象之间相似度的度量方法。



## □ 离群点类型总结

- 实际应用中可能包含多个类型的离群点，每个数据对象又可能同时属于多个离群点类型：
  - 对全局离群点的检测最简单，不需要任何背景知识。
  - 检测全局离群点的关键是针对所考虑的应用找到一个合适的偏离度量。

## □ 离群点类型总结

- 实际应用中可能包含多个类型的离群点，每个数据对象又可能同时属于多个离群点类型：
  - 情境离群点分析依赖于情境的设定，一个数据对象在某种情境下可能是离群点，在另一种情况可能不是离群点。
  - 全局离群点可以作为无情境（以整个数据集作为情境）的情境离群点。
  - 可以根据不同的需求设置不同的情境，因此为用户提供了较高的灵活性。



## □ 离群点类型总结

- 实际应用中可能包含多个类型的离群点，每个数据对象又可能同时属于多个离群点类型：
  - 集体离群点强调了集体的偏离，其中的个体数据对象可能不是离群点。
  - 对集体离群点的检测需要对象关系的背景知识，如对象之间的距离或相似度。

## □ 异常检测的挑战

- **正常对象和离群点的有效建模**：离群点检测的质量高度依赖于对正常对象和离群点的建模，但有时正常对象和离群点之间的边界并不清晰，所以如何有效建模是离群点检测中最难的挑战。
- **针对应用的离群点检测**：对相似性、距离度量、数据对象关系的描述至关重要，但是离群点的定义高度依赖于具体应用，所以几乎不存在通用的离群点检测方法。
- **噪声问题**：低质量数据会模糊正常对象和离群点之间的边界，甚至会掩盖离群点，因此噪声的存在给离群点检测带来了巨大的挑战。
- **可理解性**：算法在检测出离群点的同时要给出判断的理由。

## □ 异常检测的方法

■ 异常检测任务可以从机器学习和统计学的角度给出解决方案：

➤ 有监督异常检测方法

➤ **无监督异常检测方法**

➤ 半监督异常检测方法

➤ 基于统计学的异常检测方法

➤ **基于邻近性的异常检测方法**



## Chapter 6.2

### 异常检测方法

## □ 有监督异常检测方法

- 对训练集中正常数据和异常数据都添加标签，则异常检测问题可以看作分类问题。其基本思想是训练一个可以区分“正常”数据和异常数据的分类模型。
- 由于**类别不平衡问题**，有监督方法通常构建**一类模型**（One-class model），比如One-class SVM算法，目的是根据带标签数据描述正常数据分布的边界。
- 但是**标注的准确性问题**使得有监督异常方法在实际应用中较少。



## □ 无监督异常检测方法

■ 无监督异常检测方法不需要标签信息，仅依靠样本的特征检测异常。但是要对数据做出如下假设：

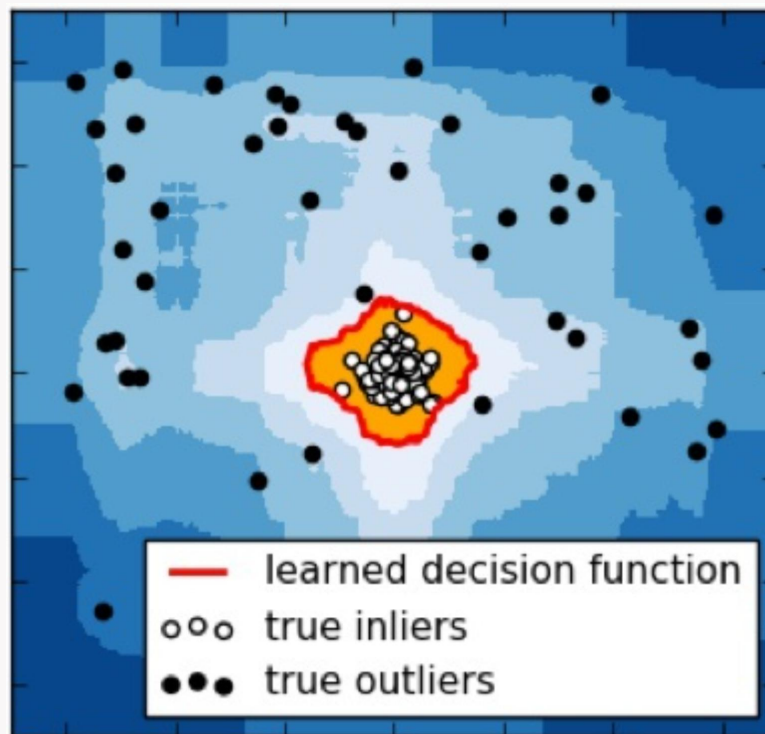
- 训练集中可以包含异常数据，但是正常数据的数量要远远多于异常数据；
- 异常数据和正常数据在特征上存在较大的差异。

■ 常见的算法：

- **孤立森林**
- **自编码器**
- 聚类方法

## □ 孤立森林

- **孤立森林** (Isolation forest) 是一种用于异常检测的基于集成学习的树类算法，将异常点定义为被孤立的样本点。



## □ 孤立森林

- 孤立指的是“把样本点从所有样本中孤立出来”，孤立的过程就是不断切割特征空间的过程，直到每个小空间仅包含一个样本。



分布约稠密的区域，  
需要切割的次数越多

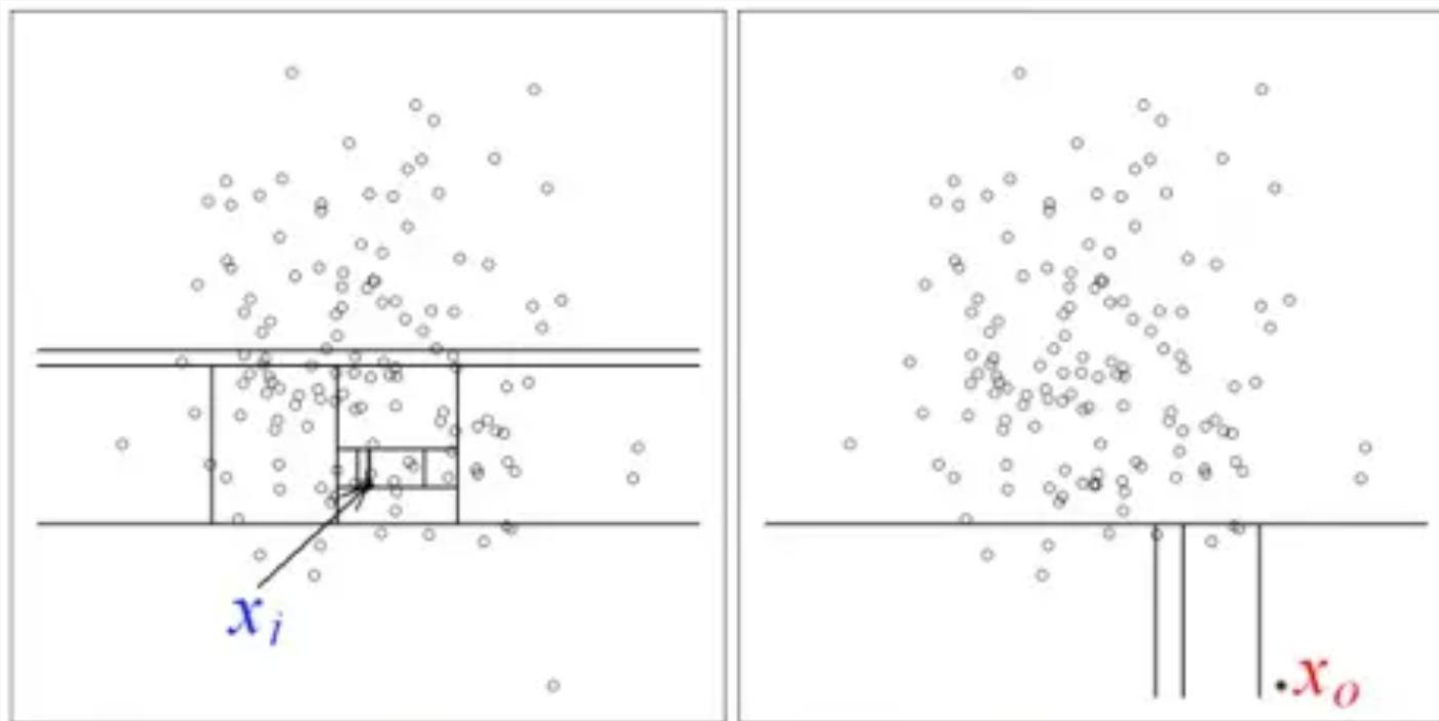
## □ 孤立森林

### ■ 孤立树的建立

- 从训练数据中随机选择  $n$  个样本点作为训练子集，放入孤立树的根节点，作为该节点的样本集合；
- 随机选择一个属性，并在区间内随机产生一个切割点，将区间一分为二；
- 将该节点的样本集合按照区间划分的结果划分为两个不互相交的子集，分别作为两个子节点的样本集合；
- 不断分裂结点，直到节点上只有一个数据，或树已经生长到了所设定的高度。

## □ 孤立森林

### ■ 分割空间的示意图



在孤立树中，如果样本点所处的节点越浅则越可能是异常点，反之则越不可能是异常点。所以通过限制树的高度以减少计算负担。



## □ 孤立森林

### ■ 集成所有孤立树的结果

- 由于属性和分割点的随机选择带来较大的不确定性，所以利用集成学习的思想，综合多棵孤立树的结果给出最终的判断。对于某个样本点  $x$ ，其异常分数为

$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}}$$

其中  $E(h(x))$  是样本点  $x$  在孤立森林中路径长度的期望， $c(n)$  表示训练子集大小为  $N$  时训练子集上的平均路径长度。

- **异常分数约趋向于1，则说明  $x$  越可能是异常点。**
- **异常分数约趋向于0，则说明  $x$  越可能是正常点。**
- **如果大部分的训练样本的异常分数都接近于0.5，则说明数据集没有明显异常点。**

## □ 孤立森林

### ■ 优点

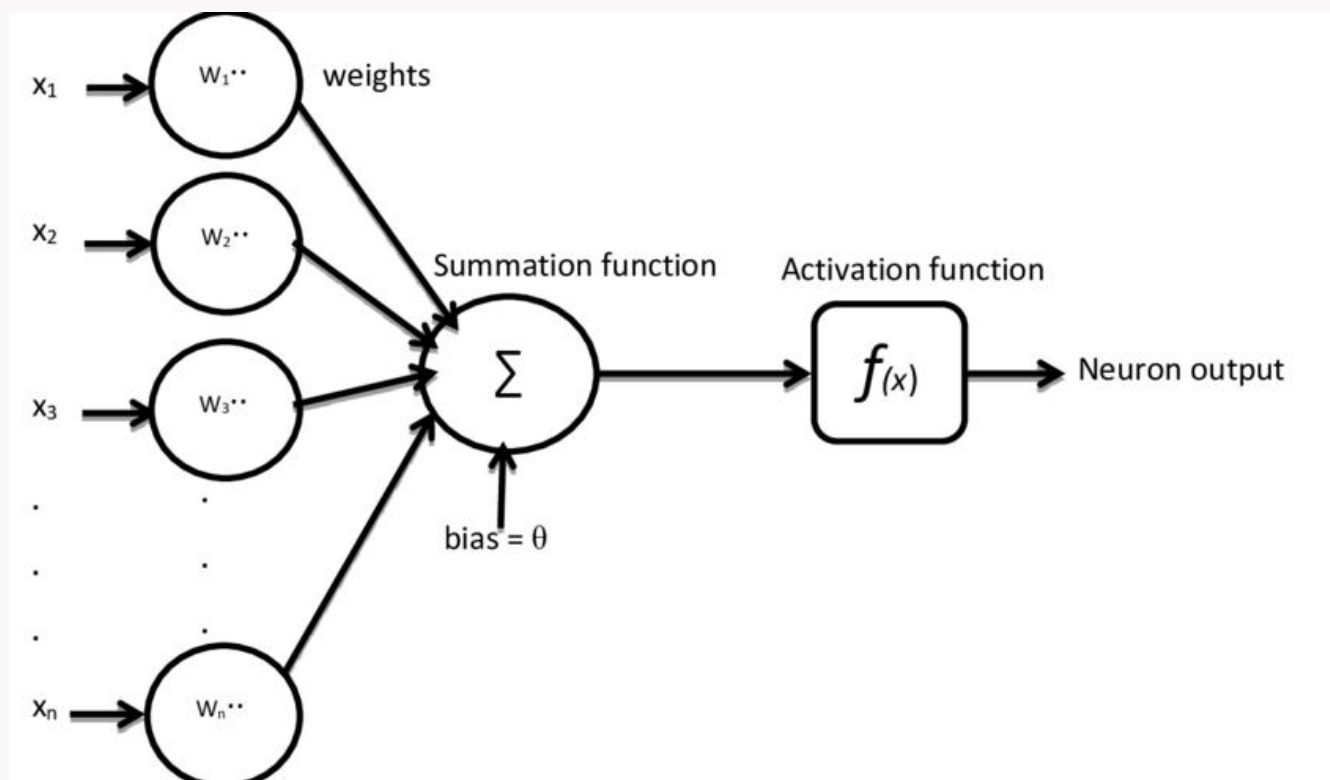
- 具有良好的可伸缩性；
- 训练速度快。

### ■ 缺点

- 在高维稀疏数据上的效果不佳，因为每次划分都随机选择一个属性，可能导致大量的属性没有被利用；
- 难以发现集体离群点。

## □ Vanilla AutoEncoder

- 自编码器 (AutoEncoder, AE) 是一种用于学习数据的低维表示的无监督深度学习算法。



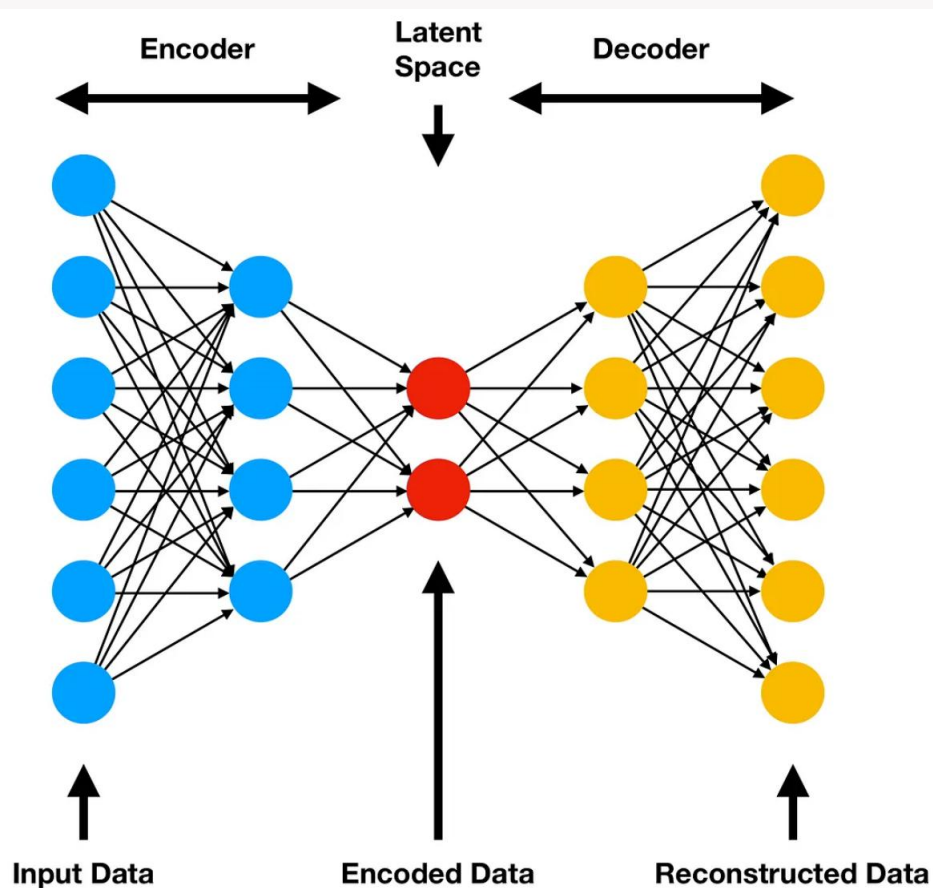
## □ Vanilla AutoEncoder

■ AE 包含编码器 (Encoder) 和解码器 (Decoder) 两部分：

- Encoder的作用是压缩数据；
- Decoder的作用是根据压缩后的表示重建原始输入。

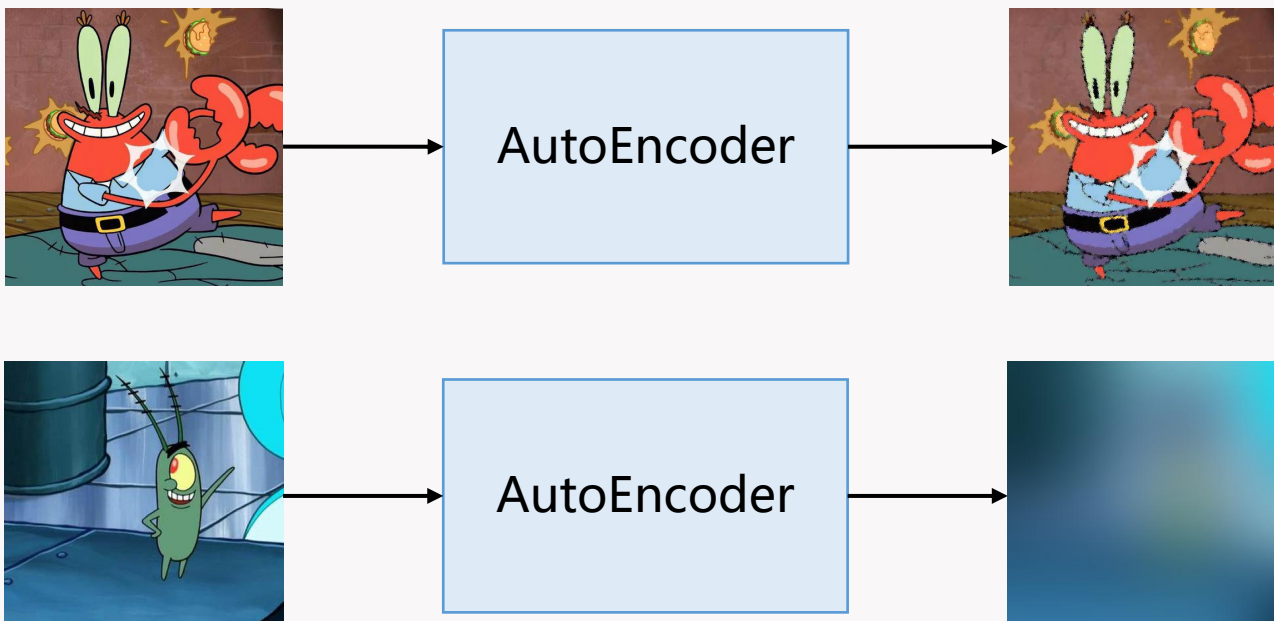
■ 使用均方差损失函数训练AE：

$$loss = \frac{1}{N} \sum_{i=1}^N \|x_i - x'_i\|_2^2$$



## □ Vanilla AutoEncoder

- 当AutoEncoder被训练在仅包含正常样本的数据集上时，它会尽可能学习正常数据的特征、结构或分布，以便有效地重建这些数据。
- 如果测试数据与训练数据差异较大，那么重建的误差就可能会增加。



## □ Vanilla AutoEncoder

### ■ 优点

- 适用范围广，泛化能力强；
- 灵活性强，效果好。

### ■ 缺点

- 数据较少时易发生过拟合问题；
- 需要精心设计网络架构。

## □ 异常检测的方法

■ 异常检测任务可以从机器学习和统计学的角度给出解决方案：

➤ 有监督异常检测方法

➤ **无监督异常检测方法**

➤ 半监督异常检测方法

➤ 基于统计学的异常检测方法

➤ **基于邻近性的异常检测方法**

## □ 基于邻近度的检测方法

- 如果某个数据对象距离其他数据对象都很远，那么这个数据对象大概率是离群点。  
基本假设是，**离群点与它最近邻的邻近度显著偏离其他数据对象与它们最近邻的邻近度。**
- 常用的基于邻近度的检测方法有以下两种：
  - 基于距离的离群点检测：如果某个数据对象的邻域内没有足够多的对象，则被认为是离群点；
  - 基于密度的离群点检测：如果某个数据对象的密度明显低于它的近邻，则被认为是离群点。



## □ 基于距离的离群点检测

- 对于数据集  $D$ ，通过设置距离阈值  $r$  来定义对象的邻域范围。对于数据对象  $p$  计算其邻域范围内的对象个数，如果  $D$  中大多数对象都不处于  $p$  的邻域中，则数据对象  $p$  被视为离群点。
- 令  $r$  表示距离阈值， $\pi$  表示分数阈值，对于数据对象  $p$ ，如果

$$\frac{\|\{x | \text{dist}(p, x) \leq r, x \in D, x \neq p\}\|}{\|D\|} \leq \pi$$

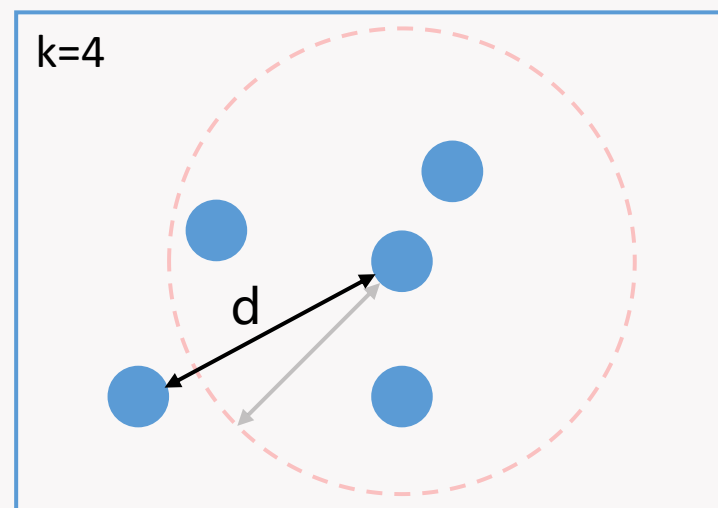
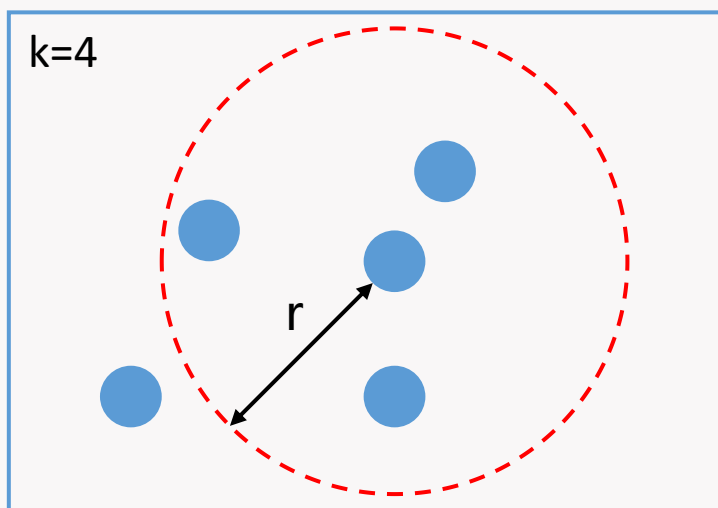
则称  $p$  是一个  $DB(r, \pi)$  离群点。

## 基于距离的离群点检测

令  $k = \lceil \pi \|D\| \rceil$ , 则上式变为

$$|\{x | \text{dist}(p, x) \leq r, x \in D, x \neq p\}| \leq k$$

此式表示当数据对象  $p$  的  **$r$  邻域**内包含的其他数据对象不超过  $k$  个时, 被视为离群点。或者说, 当数据对象  $p$  的**第  $k$  个近邻**的距离超过  $r$  时,  $p$  为离群点。



## □ 算法实现

NestedLoop ( $D, r, \pi$ )

$D$ : 数据集,  $r$ : 距离阈值,  $\pi$ : 分数阈值

```
1  O = D
2  For every x in D do
3    count = 0
4    For every y in D do
5      If  $x \neq y$  and  $\text{dist}(x,y) \leq r$  then
6        count += 1
7      If  $\text{count} > \lceil \pi \|D\| \rceil$  then
8        remove(O, x)
9      Break
10   End If
11 End For
12 End For
13 End For
14 Return O
```

当找到最够的近邻时, 提前停止内循环

从O中去除该数据对象

虽然嵌套循环的时间复杂度为 $O(n^2)$ , 但是实际使用时时间复杂度趋向于线性的。因为一般情况下离群点很少, 对于大部分非离群点内循环会提前结束。

## □ 基于距离的离群点检测

### ■ 嵌套循环的开销主要来自于以下两个方面：

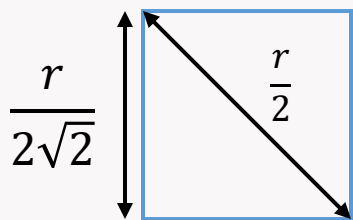
- 外层循环：逐个检查数据集中的每个数据对象；
- 内层循环：遍历整个数据集以确定某个数据对象是否是离群点。

### ■ 改进方法：

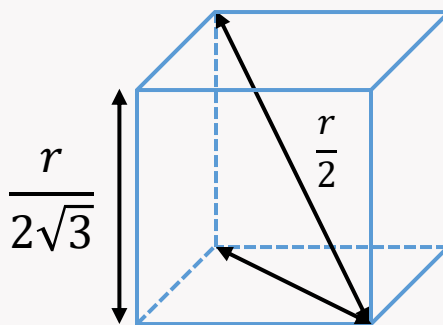
- 外层循环：根据数据对象的邻近性对它们分组，逐组判断离群性；
- 内层循环：利用该数据对象的近邻来确定其是否是离群点。

## □ 基于网格的离群点检测

- 将特征空间划分为多维网格，因此每个数据对象都属于某个网格。通过网格之间的距离成组地判断离群性。
- 假设特征空间的维数为  $m$ ，对于给定的距离阈值  $r$ ，划分后的每个单元（网格）是一个对角线长度固定为  $\frac{r}{2}$  的  $m$  维超立方体，则该单元的边长为  $\frac{r}{2\sqrt{m}}$ 。



$m=2$



$m=3$

## 基于网格的离群点检测

- 以图中的  $C$  作为分析单元，单元  $C$  的近邻可以分为两类，与  $C$  直接相邻的单元构成第一层单元，在任意方向离单元  $C$  一个或两个单元距离的单元构成第二层单元。这两层单元有如下性质：

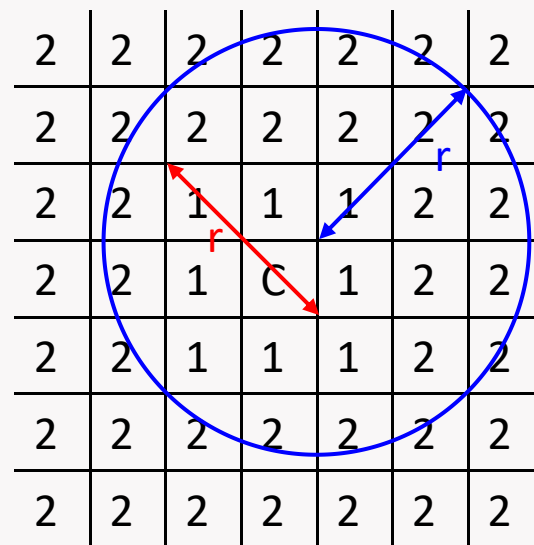
- 第一层单元的性质：** 给定  $C$  中的任意点  $x$  和第一层单元的任意点  $y$ ，有

$$\text{dist}(x, y) \leq r$$

- 第二层单元的性质：** 给定  $C$  中的任意点  $x$  和任意点  $z$ ，有

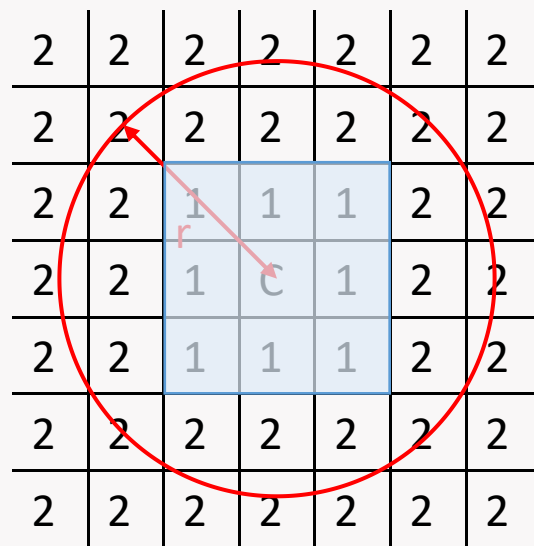
$$\text{dist}(x, z) > r$$

则  $z$  在第二层单元。



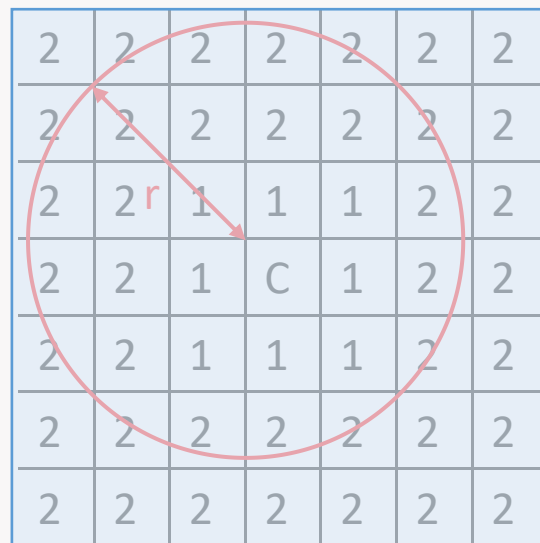
## 基于网格的离群点检测

- 假设  $a$  表示单元  $C$  中的对象的数量,  $b$  表示  $C$  的第一层对象的数量,  $c$  表示  $C$  的第二层对象的数量,  $k = \lceil \pi \|D\| \rceil$ , 则有:
  - 第一层单元剪枝规则**: 如果  $a + b > k$ , 则  $C$  中的数据对象都不是  $DB(r, \pi)$  离群点, 因为第一层单元都在  $C$  中任意对象的  $r$  邻域中, 而且至少有  $k$  个近邻。



## 基于网格的离群点检测

- 假设  $a$  表示单元  $C$  中的对象的数量,  $b$  表示  $C$  的第一层对象的数量,  $c$  表示  $C$  的第二层对象的数量,  $k = \lceil \pi \|D\| \rceil$ , 则有:
- **第二层单元剪枝规则**: 如果  $a + b + c < k + 1$ , 则  $C$  中的所有数据对象都是  $DB(r, \pi)$  离群点, 因为它们的  $r$  邻域中包含的对象数量都少于  $k$  个。



2	2	2	2	2	2	2
2	2	2	2	2	2	2
2	2	1	1	1	2	2
2	2	1	C	1	2	2
2	2	1	1	1	2	2
2	2	2	2	2	2	2
2	2	2	2	2	2	2



## □ 基于网格的离群点检测

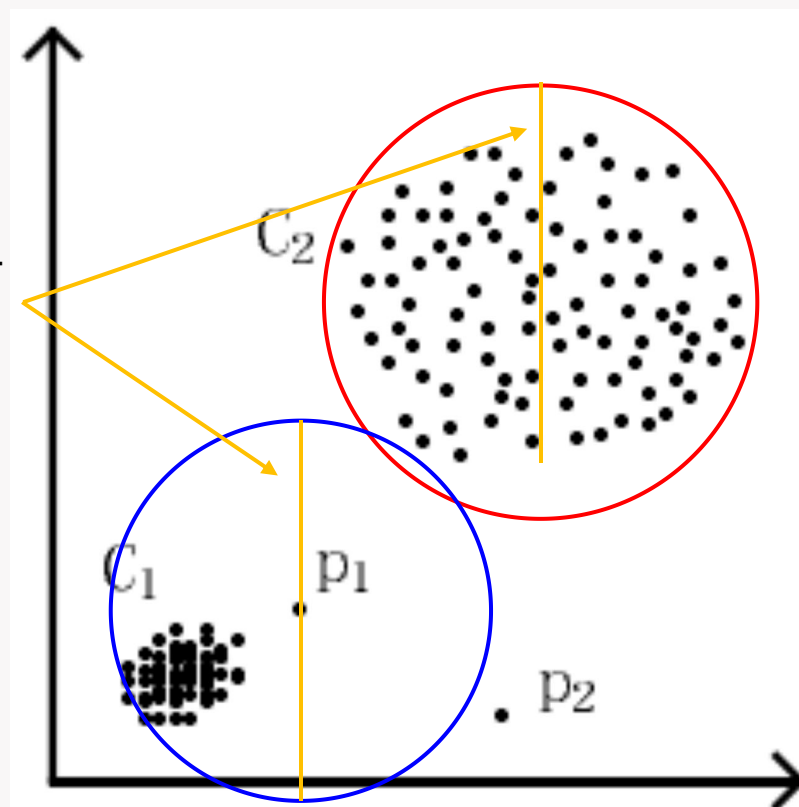
- 每次可以确定一个单元内的数据对象是否都是离群点，且只需要检查有限多个邻近单元。
- 两个剪枝规则并不能覆盖所有的情况，因此存在数据对象无法被两个剪枝规则判断。此时只需要统计这个数据对象在第一层单元和第二层单元中  $r$  邻域内包含的对象数量。

## □ 基于密度的离群点检测

- 基于距离的离群点检测从整个数据集出发判断离群点。但是现实世界中许多数据集都呈现更复杂的结构，我们可能更关心其局部邻域，而非整个数据的分布。

p1到 C1的距离小于C2中对象到其最近邻的平均距离

**p1不是基于距离的离群点，或者C2内的对象都是基于距离的离群点。**



## □ 基于密度的离群点检测

- 基于密度的离群点检测提出了局部离群点的概念，局部离群点可以看作是情境离群点的特例：
  - 如果数据集中某个数据对象的密度显著地偏离它所在局部区域的密度，则该数据对象被认为是**局部离群点** (Local outlier) 。
- 该方法利用相对密度判断离群点。其基本假设是，非离群点对象周围的密度与其近邻周围的密度相似，而离群点对象周围的密度显著不同于其近邻周围的密度。

## □ 基于密度的离群点检测

- 对于数据集  $D$ , 数据对象  $x$  的  $k$ -距离记为  $dist_k(x)$ , 是  $x$  与另一个对象  $y$  之间的距离  $dist(x, y)$ , 使得:
  - 至少有  $k$  个对象  $x' \in D - \{x\}$ , 使得  $dist(x, x') \leq dist(x, y)$ ;
  - 至少有  $k-1$  个对象  $x'' \in D - \{x\}$ , 使得  $dist(x, x'') < dist(x, y)$ 。

## □ 基于密度的离群点检测

- 数据对象  $x$  的  $k$ -距离邻域包含到  $x$  的距离不大于  $dist_k(x)$  的所有对象，记为

$$N_k(x) = \{x' | x' \in D, dist(x, x') \leq dist_k(x)\}$$

$N_k(x)$  中包含的对象数量可能超过  $k$  个，因为可能有多个数据对象到  $x$  的距离是相同的。

- 可以使用  $N_k(x)$  中对象到  $x$  的平均距离作为  $x$  的局部密度度量。

## □ 基于密度的离群点检测

- 如果  $x$  有一个非常近邻的对象  $y$  使得  $dist(x, y)$  很小, 则距离度量的统计波动可能会很大。可以借助于可达距离的概念对距离度量进行平滑:

- 对于两个对象  $x$  和  $x'$ , 它们的可达 $k$ -距离表示为

$$reachdist_k(x, x') = \max\{dist_k(x), dist(x, x')\}$$

- 则  $x$  的**局部可达密度**定义为

$$lrd_k(x) = \frac{|N_k(x)|}{\sum_{x' \in N_k(x)} reachdist_k(x, x')}$$

- 基于密度的聚类方法依赖两个参数计算密度, 而基于密度的离群点检测方法近使用参数  $k$  确定局部区域, 对比数据对象的局部可达密度判断离群性。

## □ 基于密度的离群点检测

- 基于局部可达密度，定义数据对象  $x$  的**局部离群点因子** (Local outlier factor) :

$$LOF_k(x) = \frac{\sum_{x' \in N_k(x)} \frac{lrd_k(x')}{lrd_k(x)}}{|N_k(x)|}$$

- $LOF_k(x)$  描述了  $x$  的局部可达密度与其  $k$ -最近邻的局部可达密度之比的平均值，可进行如下判断：
  - $LOF_k(x)$  远远大于1:  $x$  相对于其近邻点来说密度很低，很可能是离群点；
  - $LOF_k(x)$  接近1:  $x$  相对于其近邻点来说密度相似，很可能是正常点；
  - $LOF_k(x)$  小于1:  $x$  相对于其近邻点来说密度更高，可能处于簇的中心。



# Chapter 6

## 本章小结



## □ 异常检测

- 异常检测的定义、应用、难点和挑战
- 和离群点的定义、类型、与噪声的区别
- 有监督异常检测方法：单类模型
- 无监督异常检测方法：孤立森林，自编码器
- 基于邻近性的异常检测方法：基于距离的检测、基于网格的检测和基于密度的检测



## 第六章 完结



## □要求

- 选择一个数据集，分析该数据集并提出**两个**有意义的问题，使用数据挖掘技术给出问题的**解决方案**以及**最终结论**。
- 截止时间：12月31日

## □ 要求

## ■ 示例

- 对学生的压力来源比较感兴趣，选择Student Stress Factors数据集（1100个样本，20 个与学生压力相关的属性）：
- 心理因素 => '焦虑水平', '自尊', '心理健康历史', '抑郁症'
- 生理因素 => '头痛', '血压', '睡眠质量', '呼吸问题'
- 环境因素 => '噪音水平', '居住条件', '安全', '基本需求'
- 学术因素 => '学业表现', '学习负担', '师生关系', '未来职业忧虑'
- 社会因素 => '社会支持', '同伴压力', '课外活动', '欺凌'
- 压力等级

## □ 要求

## ■ 示例

➤ 提出了以下问题：

1. 心理、生理、社会、环境和学术因素，哪些更容易给学生带来压力？
2. 学生睡眠质量很差的主要因素是什么？
3. 哪些因素之间具有很高的关联性？

## □评分点和相关要求

评分点	说明	分数
问题的明确性	提出的问题具体、有意义，阐述清楚问题的类型（分类/回归，聚类，关联分析，异常检测）	2.5+2.5
流程的完整性	包含数据挖掘的完整流程，包括数据初步探索、数据预处理，数据挖掘，结果分析。	25+25
结论的可理解性	数据挖掘的结果能够以直观、清晰、易于理解的方式呈现给非专业人士或相关利益相关者。	7.5+7.5
报告的可阅读性	按章节设置标题，标题应简短并与相关内容高度对应。无明显格式问题，文本表达清晰有逻辑。	30