

## 9. Outlier Analysis

### Knowledge Discovery in Databases

Dominik Probst, [Dominik.probst@fau.de](mailto:Dominik.probst@fau.de)

Chair of Computer Science 6 (Data Management), Friedrich-Alexander-University Erlangen-Nürnberg

Summer semester 2024

---

# Outline

1. **Outlier and Outlier Analysis**
2. **Outlier-Detection Methods**
3. **Statistical Approaches**
4. **Proximity-Based Approaches**
5. **Summary**

---

# Outlier and Outlier Analysis

# What are Outliers?

## Outlier

An *outlier* is a data tuple that deviates significantly from normal data tuples as if generated by a different mechanism.

- **Outliers are different from noise.**
  - Noise is a random error or variance in a measured variable.
  - Noise should be removed before outlier detection.
- **Outliers are interesting.**
  - They violate the mechanism that generates data tuples that are considered normal.
  - Could occur by chance, measurement error, or any other reason.  
→ Justification *why* a tuple is an outlier is important.
- **Outlier detection vs. novelty detection:** Early stage: outlier; but later merged into the model.
- **Applications:** fraud detection, customer segmentation, medical analysis, industry damage detection.

# Types of Outliers I

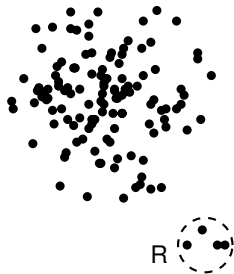
Three kinds: global, contextual, and collective outliers

## 1. **Global** outlier (or **point anomaly**):

- Significantly deviates from the rest of the data set.
- Simplest form of outlier, therefore, most methods focus on finding these.
- Issue: Find an appropriate measurement of deviation.

## 2. **Contextual** outlier (or **conditional outlier**):

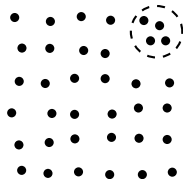
- Deviates significantly based on a selected context.
  - Example: Is 23 °C in Erlangen an outlier? (Depending on summer or winter).
- Attributes of data objects divided into two groups:
  - **Contextual attributes**: define the context, e.g., time & location.
  - **Behavioral attributes**: characteristics of the object, used in outlier evaluation, e.g., temperature.
- Can be viewed as a generalization of local outliers (**density significantly deviates from its local area**).
- Issue: Formulation of a meaningful context.



## Types of Outliers II

### 3. **Collective** outlier:

- A **subset** of data objects that collectively deviates significantly from the whole data set.
- Example: intrusion detection – a number of computers keep sending denial-of-service packages to each other.
- **Detection of collective outliers:**
  - Consider not only behavior of individual objects, but also that of groups of objects.
  - Need to have the background knowledge on the relationship among data objects, such as a distance or similarity measure on objects.



## Outliers in a Data Set

- A data set may have multiple types of outliers.
- One data tuple may belong to more than one type of outlier.

# Challenges of Outlier Detection

- **Modeling normal objects and outliers properly.**
  - Hard to enumerate all possible normal behaviors in an application.
  - No clear line between normal data tuples and outliers.
- **Application-specific outlier detection.**
  - Choice of distance measure among objects and the model of relationship among objects are application-dependent.
  - E.g. clinical data: a small deviation could be an outlier; while in marketing analysis: larger fluctuations.
- **Handling noise in outlier detection.**
  - Noise may distort the normal objects and blur the distinction between normal objects and outliers.
  - It may hide outliers and reduce the effectiveness of outlier detection.
- **Understandability.**
  - Understand why these are outliers: justification of the detection.
  - Specify the degree of an outlier: the unlikelihood of the object being generated by a normal mechanism.

---

# Outlier-Detection Methods



# How can we detect outliers?

## Two ways to categorize outlier-detection methods:

Grouping according to

1. **How many samples are labeled:**  
I.e. supervised, semi-supervised vs. unsupervised methods.
2. **Assumptions** regarding normal and abnormal samples.  
I.e. statistical, proximity-based, and clustering-based methods.

# Outlier Detection: Grouping According to Label Existence I

Domain expert labelled all, some, or no samples. **Supervised Methods:**

- Modeling outlier detection as a **classification problem**:  
Samples examined by domain experts used for training & testing.
- Methods for learning a classifier for outlier detection effectively:
  - Model normal objects & report those not matching the model as outliers.
  - Model outliers and treat those not matching the model as normal.
- **Challenges:**
  - Imbalanced classes, i.e., outliers are rare:  
Boost the outlier class and make up some artificial outliers.
  - Catch as many outliers as possible.  
Therefore: recall is more important than accuracy  
(i.e., not mislabeling normal objects as outliers).

# Outlier Detection: Grouping According to Label Existence II

## Unsupervised Methods:

- No labels available.
- **Implicit assumptions:**
  - Normal objects are somewhat “clustered” into multiple groups, each having some distinct features.
  - Outliers are expected to be far away from any group of normal objects.
- Adapt clustering methods for unsupervised outlier detection:
  1. Find clusters.
  2. Samples not falling in any cluster are outliers.
- **Challenges:**
  - Samples outside of clusters may not be outliers.
  - Costly to find clusters.
  - Hard to distinguish noise from outliers.
  - Can't detect collective outliers effectively.

# Outlier Detection: Grouping According to Label Existence III

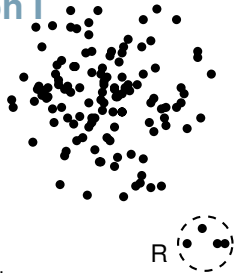
## Semi-Supervised Methods:

- Only a small set of samples are labeled as normal or as outlier.
- **If some labeled normal objects are available:**
  - Use the labeled examples and the proximate unlabeled objects to train a model for normal objects.
  - Those not fitting the model of normal objects are detected as outliers.
- **If only some labeled outliers are available, that small number may not cover all possible outliers well.**
  - To improve the quality of outlier detection: get help from models for normal objects learned from unsupervised methods.

# Outlier Detection: Grouping Based on Assumption I

## Statistical Methods

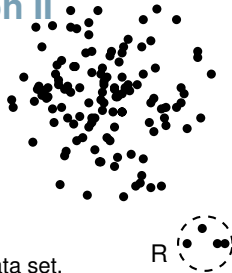
- (Also known as model-based methods)
- Assume that the **normal data follow some statistical model**.
  - The data not following the model are outliers.
- **Example (right figure):**
  - First use Gaussian distribution  $\mathcal{N}_D(x \mid \mu, \sigma)$  to model the normal data.
  - For each object  $y$  in region  $R$ , estimate  $\mathcal{N}_D(y \mid \mu, \sigma)$ , the probability that  $y$  fits the Gaussian distribution.
  - If  $\mathcal{N}_D(y \mid \mu, \sigma)$  is very low,  $y$  is unlikely generated by the Gaussian model, thus an outlier.
- **Effectiveness of statistical methods:**
  - Highly depends on whether the assumption of statistical model holds in the real data.
- **There are many kinds of statistical models.**
  - E.g., parametric vs. non-parametric.



# Outlier Detection: Grouping Based on Assumption II

## Proximity-Based Methods

An object is an outlier if the **nearest neighbors of the object are far away**, i.e., the proximity of the object significantly deviates from the proximity of most of the other objects in the same data set.



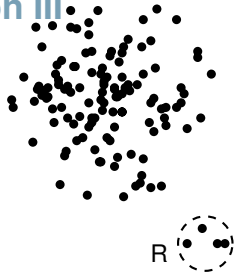
- **Example (right figure):**
  - Model the proximity of an object using its 3 nearest neighbors.
  - Objects in region R are substantially different from other objects in the data set.
  - Thus the objects in R are outliers.
- **Effectiveness of proximity-based methods:**
  - Highly relies on the proximity measure.
  - In some applications, proximity or distance measures cannot be obtained easily.
  - Often have a difficulty in finding a group of outliers which are close to each other.
- **Two major types of proximity-based outlier detection:**
  - Distance-based vs. density-based.

# Outlier Detection: Grouping Based on Assumption III

## Clustering-Based Methods

Normal data belong to large and dense clusters, whereas outliers belong to **small or sparse clusters**, or do not belong to any cluster.

- **Example (right figure): Two clusters.**
  - All points not in R form a large cluster.
  - The two points in R form a tiny cluster, thus are outliers.
- **Many clustering methods:**
  - Thus also many clustering-based outlier detection methods.
- **Clustering is expensive.**
  - Straightforward adaptation of a clustering method for outlier detection can be costly and does not scale up well for large data sets.



---

# Statistical Approaches



## Statistical Approaches

- **Assumption:** Objects in a data set are **generated by a stochastic process** (a generative model).
- **Idea:** Learn a generative model fitting the given data set, and then identify the objects in low-probability regions of the model as outliers.
- **Methods divided into two categories:**

### Parametric Methods

- Assumes that the normal data is generated by a parametric distribution with parameter  $\theta$ .
- The probability density function of the parametric distribution  $f(x, \theta)$  gives the probability that object  $x$  is generated by the distribution.
- Small values indicate potential outlier.

## Statistical Approaches

- **Assumption:** Objects in a data set are **generated by a stochastic process** (a generative model).
- **Idea:** Learn a generative model fitting the given data set, and then identify the objects in low-probability regions of the model as outliers.
- **Methods divided into two categories:**

### Parametric Methods

- Assumes that the normal data is generated by a parametric distribution with parameter  $\theta$ .
- The probability density function of the parametric distribution  $f(x, \theta)$  gives the probability that object  $x$  is generated by the distribution.
- Small values indicate potential outlier.

### Non-Parametric Methods

- Do not assume an a-priori statistical model and determine the model from the input data.
- Not completely parameter-free, but consider number and nature of the parameters to be flexible and not fixed in advance.
- **Examples:** **histogram** and kernel-density estimation.

## Parametric Methods I: Detection Based on Normal Distribution

- **Univariate data:** A data set involving *only one attribute* or variable.
- **Assumption:** Data are generated from a normal distribution.
- **Learn the parameters from the input data, and identify the points with low probability as outliers.**

**Example:** Assume data follows a **normal distribution**.

- Recall: normal distribution  $\mathcal{N}(\mu, \sigma)$ 
  - Characterized by two parameters: mean  $\mu$ , and standard deviation  $\sigma$ .
  - Probability density function:  $f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$ .
- **Idea:** Estimate parameters  $\mu$  and  $\sigma$  so that normal distribution fits data as close as possible.
- **Question:** How to estimate these parameters? → *Maximum likelihood method*.

# Excursus: Maximum Likelihood Estimation I

## Maximum Likelihood Estimation

*Maximum Likelihood Estimation (MLE) estimates parameters of an assumed probability distribution such that the distribution fits the data as closely as possible.*

- **Likelihood function**  $\mathcal{L}(\theta|X)$  with
  - parameter space  $\theta$  and
  - observed data  $X = \{x_1, x_2, \dots, x_n\}$

describes the joint probability of observed data as a function of the parameters of the assumed distribution.

- Frequently assumed distribution: Gaussian (normal) distribution  $\mathcal{N}(\mu, \sigma)$ .
- Thus, the likelihood function  $\mathcal{L}(\theta|X)$  with parameter space  $\theta = \{\mu, \sigma\}$  is the Gaussian process:

$$\mathcal{L}(\theta|X) = \mathcal{L}(\mu, \sigma|x_1, \dots, x_n) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}}$$

- Find good estimates for  $\theta$  such that  $\hat{\theta} = \arg \max_{\theta} \mathcal{L}(\theta|X)$ .

## Excursus: Maximum Likelihood Estimation II

- In the case of a Gaussian distribution: find estimates for  $\mu$  and  $\sigma$  such that

$$\begin{aligned}\hat{\mu} &= \arg \max_{\mu} \mathcal{L}(\theta|X), \\ \hat{\sigma} &= \arg \max_{\sigma} \mathcal{L}(\theta|X)\end{aligned}$$

- **General procedure:**

1. Generate two derivatives, with respect to  $\mu$  and  $\sigma$ , respectively.
  2. Solve each equation by setting them equal to zero.
- Instead of taking the derivative directly, we take the log of the likelihood function as this makes it easier to take derivatives.

$$\ln \mathcal{L}(\theta|x_1, \dots, x_n) = \ln \mathcal{L}(\mu, \sigma|x_1, \dots, x_n) = \ln \left( \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}} \right)$$

- Logarithm is monotonically increasing, thus satisfying  $\arg \max_{\theta} \ln(\mathcal{L}(\theta|X)) = \operatorname{argmax}_{\theta} \mathcal{L}(\theta|X)$ .
- Logarithm turns multiplication  $\prod$  into addition  $\sum$  making derivatives easier to compute.

## Excursus: Maximum Likelihood Estimation III

0. Transform equation:

$$\ln \mathcal{L}(\mu, \sigma | x_1, \dots, x_n) = \ln \left( \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}} \right) = -\frac{n}{2} \ln 2\pi - \frac{n}{2} \ln \sigma^2 - \frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2$$

1. To estimate parameters, take the partial derivative with respect to  $\mu$  and  $\sigma$ :

$$\frac{\partial}{\partial \mu} \ln (\mathcal{L}(\mu, \sigma | x_1, \dots, x_n)) = \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu)$$

$$\frac{\partial}{\partial \sigma} \ln (\mathcal{L}(\mu, \sigma | x_1, \dots, x_n)) = -\frac{n}{\sigma} + \frac{1}{\sigma^3} \sum_{i=1}^n (x_i - \mu)^2$$

2. Then, by setting these equations equals zero, we derive the following likelihood estimates:

$$\text{mean } \hat{\mu} = \frac{1}{n} \sum_{i=1}^n x_i \quad \text{standard deviation } \hat{\sigma} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \mu)^2}$$

Refer to appendix for proof.

## Parametric Methods I: Detection Based on Normal Distribution (2)

### Example:

- Average yearly temperatures:  $\{24.0, 28.9, 28.9, 29.0, 29.1, 29.1, 29.2, 29.2, 29.3, 29.4\}$ .
- For these data with  $n = 10$ , we have

$$\hat{\mu} = 28.61, \quad \hat{\sigma} = \sqrt{2.29} = 1.51.$$

- Then the most deviating value 24.0 is 4.61 away from the estimated mean.
- **Recall:**  $\mu \pm 3\sigma$  contains 99.7% of the data under the assumption of normal distribution.
- $\mu \pm 3\sigma$

## Parametric Methods I: Detection Based on Normal Distribution (2)

### Example:

- Average yearly temperatures:  $\{24.0, 28.9, 28.9, 29.0, 29.1, 29.1, 29.2, 29.2, 29.3, 29.4\}$ .
- For these data with  $n = 10$ , we have

$$\hat{\mu} = 28.61, \quad \hat{\sigma} = \sqrt{2.29} = 1.51.$$

- Then the most deviating value 24.0 is 4.61 away from the estimated mean.
- **Recall:**  $\mu \pm 3\sigma$  contains 99.7% of the data under the assumption of normal distribution.
- $\mu \pm 3\sigma \Leftrightarrow \mu - 3\sigma \leq x \leq \mu + 3\sigma$



## Parametric Methods I: Detection Based on Normal Distribution (2)

### Example:

- Average yearly temperatures:  $\{24.0, 28.9, 28.9, 29.0, 29.1, 29.1, 29.2, 29.2, 29.3, 29.4\}$ .
- For these data with  $n = 10$ , we have

$$\hat{\mu} = 28.61, \quad \hat{\sigma} = \sqrt{2.29} = 1.51.$$

- Then the most deviating value 24.0 is 4.61 away from the estimated mean.
- **Recall:**  $\mu \pm 3\sigma$  contains 99.7% of the data under the assumption of normal distribution.
- $\mu \pm 3\sigma \Leftrightarrow \mu - 3\sigma \leq x \leq \mu + 3\sigma \Leftrightarrow \frac{\mu - x}{\sigma} \leq 3 \leq \frac{\mu + x}{\sigma}$
- Plugging in the minimum value (24.0) into the equation yields  $\frac{\mu - x}{\sigma} = \frac{28.61 - 24}{1.51} = \frac{4.61}{1.51} = 3.04 > 3$
- Probability that 24.0 is generated by a normal distribution is less than 0.15%.
  - Each *tail* to the left and to the right of the 99.7% has 0.15%.
- Hence, 24.0 identified as an outlier.

# Parametric Methods I: The Grubbs' Test<sup>1</sup>

## Grubbs' Test

The *Grubbs' Test* is a statistical test to detect outlier in a univariate data set under the assumption that this data set follows a normal distribution. Grubbs' Test is also known as *maximum normed residual test*.

- Postulates following hypotheses:
  - $H_0$  : Data set contains no outliers.
  - $H_a$  : Data set contains exactly one outlier.
- For a data set  $\{x_1, x_2, \dots, x_n\}$  compute the **Grubbs' test statistic**:  $G = \frac{\max |x_i - \bar{x}|}{s}$ , with sample mean  $\bar{x}$  and sample standard deviation  $s$ .
- The two-sided test with significance level  $\alpha$  to reject the null hypothesis is as follows:

$$G > \frac{N-1}{\sqrt{N}} \sqrt{\frac{t_{\frac{\alpha}{2N}, N-2}^2}{N-2 + t_{\frac{\alpha}{2N}, N-2}^2}}$$

where  $t_{\frac{\alpha}{2N}, N-2}^2$  is the value taken by a  $t$ -distribution at a significance level of  $\frac{\alpha}{2N}$ , and  $N$  is the number of objects in the data set.

<sup>1</sup>Named after Frank E. Grubbs.

## Parametric Methods II: Detection of Multivariate Outliers

- **Multivariate data:** A data set involving **two or more attributes** or variables.
- Univariate outlier detection methods can be extended to the multivariate case.
- **Central Idea:** Transform the multivariate outlier-detection task into a univariate outlier-detection problem.
- Methods include but not limited to:
  1. Mahalanobis distance.
  2.  $\chi^2$  statistic

### Beware: Term “Multivariate” Defined Differently Throughout Disciplines

Depending on the discipline of the paper you may find yourself reading, the term multivariate data, though, maybe misleading. For instance, in probability and statistics, a *multivariate random variable* is a column vector  $X = \{x_1, x_2, \dots, x_n\}$ , i. e. univariate data. However, a *multivariate time series* contains multiple univariate time series (column vectors).

## Parametric Methods II: Mahalanobis Distance and $\chi^2$ Statistic

### Mahalanobis Distance

- Measures the distance between an object  $\mathbf{x}$  and the data set's distribution.
- Defined as

$$\text{MD}(\mathbf{x}) = \sqrt{(\mathbf{x} - \bar{\mathbf{x}})^T \mathbf{C}^{-1} (\mathbf{x} - \bar{\mathbf{x}})}$$

with mean  $\bar{\mathbf{x}}$ , and covariance matrix  $\mathbf{C}$ , and  $\mathbf{x} = (x_1, x_2, \dots, x_n)$  where  $x_i$  is a column vector at position  $i$ .

- Use the Grubbs' test on this measure to detect outliers.

### $\chi^2$ Statistic

- Captures multivariate outliers under the assumption of a normal distribution.
- For a data set  $\mathbf{x} = (x_1, x_2, \dots, x_n)$ ,  $\chi^2$  statistic is defined as:

$$\chi^2 = \sum_{i=1}^n \frac{(x_i - \bar{x})^2}{\bar{x}}$$

- If  $\chi^2$  statistic is large, then  $\mathbf{x}_i$  is an outlier.

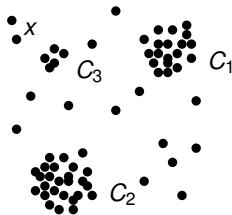
## Parametric Methods III: Using Mixture of Parametric Distributions

- Assuming data follows a normal distribution is too simple for complex data distributions.
- Right figure: The objects between the two clusters cannot be captured as outliers since they are close to the estimated mean.
- Assume data is generated by two normal distributions.**
  - For any object  $\mathbf{x}$  in the data set, the probability that  $\mathbf{x}$  is generated by the mixture of the two distributions  $\mathcal{N}_1(\mu_1, \sigma_1)$  and  $\mathcal{N}_2(\mu_2, \sigma_2)$  is given by

$$P(\mathbf{x} \mid \mathcal{N}_1, \mathcal{N}_2) = f_{\mathcal{N}_1}(\mathbf{x} \mid \mu_1, \sigma_1) + f_{\mathcal{N}_2}(\mathbf{x} \mid \mu_2, \sigma_2),$$

where  $f_{\mathcal{N}_1}$  and  $f_{\mathcal{N}_2}$  are the probability density functions of  $\mathcal{N}_1$  and  $\mathcal{N}_2$ , respectively.

- Use expectation-maximization (EM) algorithm<sup>2</sup> to learn the parameters  $\mu_1, \sigma_1, \mu_2, \sigma_2$  from the data.
- An object  $\mathbf{x}$  is an outlier if it does not belong to any cluster.



<sup>2</sup>Expectation-maximization algorithm is not covered in this lecture. If you are interested, you may take a look at chapter 11 of our reference book titled *Data Mining: Concepts and Techniques*.

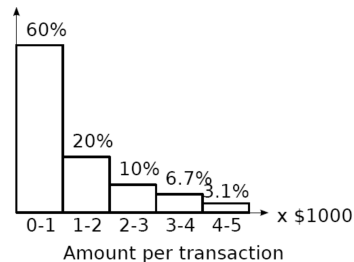
## Non-Parametric Methods: Detection Using Histogram I

*Non-parametric methods* make fewer assumptions about the data, and thus are applicable in more scenarios.

### Outlier detection using histograms:

1. Construct histogram
2. Detect outlier by checking objects against the histogram and determine if is normal or not.

**Example:** A transaction with the amount of \$7,500 is an outlier, since only 0.2% of the transactions have an amount higher than \$5,000.



## Non-Parametric Methods: Detection Using Histogram II

### Problem:

- Hard to **choose an appropriate bin size** for histogram.
- Too small bin size  $\rightarrow$  normal objects in empty/rare bins, false positive.
- Too big bin size  $\rightarrow$  outliers in some frequent bins, false negative.

### Alternatively: Estimate probability density function with **Kernel Density Estimation** (KDE)

- Given: Univariate data set  $X = \{x_1, x_2, \dots, x_n\}$  that is drawn i. i. d (independent and identically distributed)
- Its kernel density estimate is  $\hat{f}_h(x) = \frac{1}{n} \sum_{i=1}^n K_h(x - x_i) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x-x_i}{h}\right)$ , where
  - $\hat{f}$  is the density function to be estimated,
  - $h > 0$  is a smoothing parameter, also called *bandwidth*, corresponds to the bin width of the histogram. If too small, the curve gets too rough, if too large, shape of  $\hat{f}$  is too washed out.
  - $K$  is the kernel, a non-negative function, typically standard Gaussian function with  $\mu = 0$  and  $\sigma = 1$ .
  - $K_h$  is the scaled kernel  $K_h(x) = \frac{1}{h} K\left(\frac{x}{h}\right)$

---

# Proximity-Based Approaches



## Proximity-Based Approaches

- **Assumption:** Outliers are far away
- Quantifies how far away objects are from each other by employing a distance measure
- Two types of proximity-based outlier detection methods:

### Distance-Based Methods

- Consults the neighborhood.
- Neighborhood defined by a given radius.
- Objects are considered as outlier if its neighborhood does not contain enough objects.
- Different methods: distance-based with nested loop, grid-based.

### Density-Based Methods

- Investigates the density of an object and that of its neighbors.
- Object is considered an outlier if density is lower than that of its neighbors.

## Distance-Based Outlier Detection: Method with a Nested Loop I

- Suppose we have:
  - a set of data objects  $D$  with  $n$  objects, and
  - a user-specified threshold  $r$  with  $r > 0$
- For each object  $\mathbf{o} \in D$  examine the number of other objects in the  $r$ -neighborhood of  $\mathbf{o}$ .
- If most objects in  $D$  are far away from  $\mathbf{o}$ , then  $\mathbf{o}$  is an outlier.
- Formally: **An object  $\mathbf{o}$  is a DB( $r, \pi$ )-outlier, iff**

$$\frac{||\{\mathbf{o}' \mid d(\mathbf{o}, \mathbf{o}') \leq r\}||}{||D||} \leq \pi.$$

where

- $d(\mathbf{o}, \mathbf{o}')$  is a distance function of two objects  $\mathbf{o}$  and  $\mathbf{o}'$
- $\pi$  with  $0 \leq \pi \leq 1$  is a fraction threshold.
- Determine if an object  $\mathbf{o}$  is an outlier by taking a look at the  $k$ -nearest neighbor  $\mathbf{o}_k$  where  $k = \lceil \pi n \rceil$
- $\mathbf{o}$  is an outlier, if  $d(\mathbf{o}, \mathbf{o}_k) > r$ .

## Distance-Based Outlier Detection: Method with a Nested Loop II

### Efficient computation: Nested-loop algorithm:

- For every object  $\mathbf{o}_i \in D$  calculate distance with every other object  $\mathbf{o}_j \in D$  where  $i \neq j$ .
- Also count the number of objects in the  $r$ -neighborhood of  $\mathbf{o}_i$ .
- Terminate inner loop if  $\pi n$  objects found in distance  $r$ :  $\mathbf{o}_i$  is no  $\mathbf{DB}(r, \pi)$ -outlier.
- Otherwise do not terminate:  $\mathbf{o}_i$  is an outlier.

### Efficiency:

- $\mathcal{O}(n^2)$ , but linear CPU-time w.r.t. data set size.
- Early termination: small dataset with few outliers.
- Costly for large dataset not fitting into RAM.

**Data:** a set of objects  $D = \{\mathbf{o}_1, \dots, \mathbf{o}_n\}$ , threshold  $r$ , fraction threshold  $\pi$

**Result:**  $\mathbf{DB}(r, \pi)$ -outlier in  $D$

```

1  foreach  $\mathbf{o}_i \in D$  do
2      count  $\leftarrow 0$ ;
3      foreach  $\mathbf{o}_j \in D$  do
4          if  $\mathbf{o}_i \neq \mathbf{o}_j$  and  $d(\mathbf{o}_i, \mathbf{o}_j) \leq r$  then
5              count  $\leftarrow$  count + 1;
6              if count  $\geq \pi n$  then
7                  /*  $\mathbf{o}_i$  cannot be a  $\mathbf{DB}(r, \pi)$ -outlier */
7                  exit inner loop;
8  return set of all  $\mathbf{DB}(r, \pi)$ -outlier in  $D$ 
  
```

## Distance-Based Outlier Detection: Method with a Nested Loop III

### Why is efficiency still a concern?

- If the complete set of objects cannot be held in main memory, there is significant cost for I/O swapping.

### The major cost:

1. **All-Pair-Similarity:**

Each object is tested against the whole data set,  
why not only against its close neighbors?

2. **Iterative Approach:**

Objects are checked one by one, why not group by group?

Improvements can be handled by using a grid-based method.

## Distance-Based Outlier Detection: A Grid-Based Method I

### • CELL:

- Data space is partitioned into a multi-dimensional grid.
- Each cell is a hyper cube with diagonal length  $\frac{r}{2}$ .
  - $r$ -distance threshold parameter.
  - $l$ -dimensions: edge of each cell  $r/(2\sqrt{l})$  long.

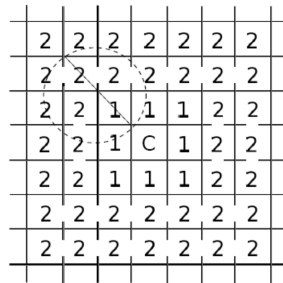
### • Level-1 cells:

- Immediately next to cell **C**.
- For any possible point **x** in **C** and any possible point **y** in a level-1 cell:  $d(x, y) \leq r$ .

### • Level-2 cells:

- One or two cells away from **C**.
- For any possible point **x** in cell **C** and any point **y** such that  $d(x, y) \geq r$ , **y** is in a level-2 cell.

- Example: Given a 2-dimensional data set, the length of each cell edge is  $\frac{r}{2\sqrt{2}}$ .



# Distance-Based Outlier Detection: A Grid-Based Method II

## Cell Pruning Rules

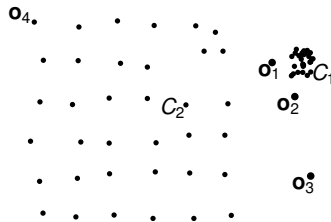
- Total number of objects in cell **C**:  $a$ .
- Total number of objects in level-1 cells:  $b_1$ .
- Total number of objects in level-2 cells:  $b_2$ .
- **Level-1 cell pruning rule:**
  - If  $a + b_1 > \lceil \pi n \rceil$ , then every object **o** in **C** is not a  $\text{DB}(r, \pi)$ -outlier, because all objects in **C** and the level-1 cells are in the  $r$ -neighborhood of **o**, and there are at least  $\lceil \pi n \rceil$  such objects.
- **Level-2 cell pruning rule:**
  - If  $a + b_1 + b_2 < \lceil \pi n \rceil + 1$ , then all objects in **C** are  $\text{DB}(r, \pi)$ -outliers, because all of their  $r$ -neighborhoods have less than  $\lceil \pi n \rceil$  other objects.
- **Only need to check the objects that cannot be pruned.**
  - Even for such an object **o**, only need to compute the distance between **o** and the objects in level-2 cells.
    - Since beyond level-2, distance from **o** is more than  $r$ .

## Density-Based Outlier Detection

- Density around **outlier** object **significantly different**
- Methods use a *relative density* of an object against its neighbor.
- This indicates to which degree an object is considered an outlier.
- **Local outliers:** Outliers compared to their local neighborhoods, not to global data distribution.

### Figure on the right:

- Objects  $\mathbf{o}_1$  and  $\mathbf{o}_2$  are local outliers to  $C_1$ ,  $\mathbf{o}_3$  is a global outlier, but  $\mathbf{o}_4$  is not an outlier.
- However, distance of  $\mathbf{o}_1$  and  $\mathbf{o}_2$  to objects in dense cluster  $C_1$  is smaller than average distance in sparse cluster  $C_2$ .
- Hence,  $\mathbf{o}_1$  and  $\mathbf{o}_2$  are not distance-based outliers.



## Density-Based Outlier Detection: Measure Relative Density

- Use the **relative density** of an object against its neighbors as the indicator of the degree of the object being an outlier.
- **$k$ -distance of an object  $\mathbf{o}$ :**  $d_k(\mathbf{o})$ .  
Distance between  $\mathbf{o}$  and its  $k$ -nearest neighbors.
- Distance  $d(\mathbf{o}, \mathbf{p})$  between  $\mathbf{o}$  and its  $k$ -nearest neighbor  $\mathbf{p}$ .
  - At least  $k$ -objects  $\mathbf{o}' \in \mathbf{D} - \{\mathbf{o}\}$  such that  $d(\mathbf{o}, \mathbf{o}') \leq d(\mathbf{o}, \mathbf{p})$ .
  - At most  $k - 1$  objects  $\mathbf{o}'' \in \mathbf{D} - \{\mathbf{o}\}$  such that  $d(\mathbf{o}, \mathbf{o}'') > d(\mathbf{o}, \mathbf{p})$ .
- $k$ -distance neighborhood of  $\mathbf{o}$ :
  - $N_k(\mathbf{o}) = \{\mathbf{o}' \mid \mathbf{o}' \in \mathbf{D}, d(\mathbf{o}, \mathbf{o}') \leq d_k(\mathbf{o})\}$ .
  - $N_k(\mathbf{o})$  could be bigger than  $k$   
since multiple objects may have identical distance to  $\mathbf{o}$ .
- Measure local distance by using the *average distance* from objects in  $N_k(\mathbf{o})$ .
- **Problem:** If  $\mathbf{o}$  has very close neighbors  $\mathbf{o}'$ , statistical fluctuations of the distance measure can be undesirable high. Overcome this problem with a reachability distance.



## Density-Based Outlier Detection: Reachability Distance

- **Reachability distance from  $\mathbf{o}'$  to  $\mathbf{o}$ :**

$$\text{reachdist}_k(\mathbf{o}' \leftarrow \mathbf{o}) = \max\{d_k(\mathbf{o}), d(\mathbf{o}, \mathbf{o}')\},$$

where  $k$  is a user-specified parameter that adds a smoothing effect.

- $k$  specifies the minimum neighborhood to be examined to determine the local density of an object.
- **Reachability distance is not symmetric!**

$$\text{reachdist}_k(\mathbf{o}' \leftarrow \mathbf{o}) \neq \text{reachdist}_k(\mathbf{o} \leftarrow \mathbf{o}')$$

- **Local reachability density of  $\mathbf{o}$ :**

$$\text{lrd}_k(\mathbf{o}) = \frac{||N_k(\mathbf{o})||}{\sum_{\mathbf{o}' \in N_k(\mathbf{o})} \text{reachdist}_k(\mathbf{o}' \leftarrow \mathbf{o})}.$$

## Density-Based Outlier Detection: Local Outlier Factor (LOF)

- LOF is the average of the ratio of the local reachability density of  $\mathbf{o}$  and those of  $\mathbf{o}$ 's  $k$ -nearest neighbors.
- The lower lrd and the higher lrd of the  $k$ -nearest neighbors of  $\mathbf{o}$ , then the higher the LOF value.
- LOF of  $\mathbf{o}$  is defined as:

$$\text{LOF}_k(\mathbf{o}) = \frac{\sum_{\mathbf{o}' \in N_k(\mathbf{o})} \frac{\text{lrd}_k(\mathbf{o}')}{\text{lrd}_k(\mathbf{o})}}{||N_k(\mathbf{o})||} = \sum_{\mathbf{o}' \in N_k(\mathbf{o})} \text{lrd}_k(\mathbf{o}') \cdot \sum_{\mathbf{o}' \in N_k(\mathbf{o})} \text{reachdist}_k(\mathbf{o}' \leftarrow \mathbf{o}).$$

- This captures a local outlier whose local density is relatively low comparing to the local densities of its  $k$ -NN.

---

# Summary


## Summary

- **Types of outliers:**
  - Global, contextual & collective outliers.
- **Outlier detection:**
  - Supervised, semi-supervised, or unsupervised.
- **Statistical (or model-based) approaches.**
- **Proximity-based approaches.**
- **Not covered here:**
  - Clustering-based approaches.
  - Classification approaches.
  - Mining contextual and collective outliers.
  - Outlier detection in high dimensional data.

## Any questions about this chapter?

Ask them now or ask them later in our forum:

StudOn Forum

 <https://www.studon.fau.de/frm5699567.html>

---

# Appendix

## Maximum Likelihood Estimation

- **Example:** Assume a normal distribution  $\mathcal{N}(\mu, \sigma)$  with probability density function

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- Likelihood function of the normal distribution for a dataset  $X = \{x_1, \dots, x_n\}$ , therefore, is as follows:

$$\mathcal{L}(\mu, \sigma | x_1, \dots, x_n) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

- General procedure:
  1. Generate two derivatives, with respect to  $\mu$  and  $\sigma$ , respectively.
  2. Solve each equation by setting them equal to zero.
- Instead of taking the derivative directly, we take the log of the likelihood function as this makes it easier to take derivatives.

## Maximum Likelihood Estimation: Log-Transform Equation

$$\begin{aligned}\ln \mathcal{L}(\mu, \sigma | x_1, \dots, x_n) &= \ln \left( \underbrace{\prod_{i=1}^n}_{1.} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}} \right) \\&= \sum_{i=1}^n \ln \left( \underbrace{\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}}}_{2.} \right) \\&= \sum_{i=1}^n \left( \ln \left( \underbrace{\frac{1}{\sqrt{2\pi\sigma^2}}}_{3.a} \right) + \ln \left( \underbrace{e^{-\frac{(x_i - \mu)^2}{2\sigma^2}}}_{3.b} \right) \right) \\&= \sum_{i=1}^n \left( \ln \left( (2\pi\sigma^2)^{-\frac{1}{2}} \right) - \frac{(x_i - \mu)^2}{2\sigma^2} \ln e \right)\end{aligned}$$

1. Log transforms multiplication into addition.
2. Transform each element in log, that is convert multiplication to addition.
3. Convert one (a) over square root and (b) exponent of euler.  
Recall:

$$x^{-v} = \frac{1}{x^v}$$

$$\sqrt[v]{x} = x^{\frac{1}{v}}$$

$$\ln x^v = v \ln x$$



## Maximum Likelihood Estimation: Log-Transform Equation

$$= \sum_{i=1}^n \left( \ln \left( \underbrace{(2\pi\sigma^2)^{-\frac{1}{2}}}_{4.a} \right) - \underbrace{\frac{(x_i - \mu)^2}{2\sigma^2} \ln e}_{4.b} \right)$$

$$= \sum_{i=1}^n \left( -\frac{1}{2} \ln \underbrace{2\pi\sigma^2}_{5.a} - \underbrace{\frac{(x_i - \mu)^2}{2\sigma^2}}_{5.b} \right)$$

$$= \sum_{i=1}^n \left( -\frac{1}{2} \ln 2\pi - \frac{1}{2} \underbrace{\ln \sigma^2}_{6.} - \frac{(x_i - \mu)^2}{2\sigma^2} \right)$$

$$= \sum_{i=1}^n \left( -\frac{1}{2} \ln 2\pi - \underbrace{\frac{1}{2} 2}_{7.} \ln \sigma - \frac{(x_i - \mu)^2}{2\sigma^2} \right)$$

$$= \sum_{i=1}^n \left( -\frac{1}{2} \ln 2\pi - \ln \sigma - \frac{(x_i - \mu)^2}{2\sigma^2} \right) \xrightarrow{7.} = -\frac{n}{2} \ln 2\pi - n \ln \sigma - \sum_{i=1}^n \frac{(x_i - \mu)^2}{2\sigma^2}$$

4. Convert (a) exponent into multiplication and (b) remove  $\ln e$ .

Recall:

$$\ln e = 1$$

5. (a) Transform multiplication to addition. (b) Nothing to do to last term.

6. Convert exponent.

7. Simplify equation.

**We can now take the derivative w. r. t.  $\mu$  and  $\sigma$  of:**

$$\ln (\mathcal{L}(\mu, \sigma | x_1, \dots, x_n))$$

## Maximum Likelihood Estimation: Take Derivative w. r. t. $\mu$

$$\begin{aligned}
 & \frac{\partial}{\partial \mu} \ln(\mathcal{L}(\mu, \sigma | x_1, \dots, x_n)) \\
 &= \underbrace{\frac{\partial}{\partial \mu} \left(-\frac{n}{2} \ln 2\pi\right)}_{1.} - \underbrace{\frac{\partial}{\partial \mu} (n \ln \sigma)}_{1.} - \sum_{i=1}^n \underbrace{\frac{\partial}{\partial \mu} \frac{(x_i - \mu)^2}{2\sigma^2}}_{2.} \\
 &= \underbrace{\sum_{i=1}^n \frac{-2(x_i - \mu)(-1)}{2\sigma^2}}_{3.} \\
 &= \sum_{i=1}^n \frac{2(x_i - \mu)}{2\sigma^2} \\
 &= \sum_{i=1}^n \frac{x_i - \mu}{\sigma^2} \\
 &= \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu)
 \end{aligned}$$

1. Derivative of this component can be treated as a constant as it does not contain  $\mu$ , therefore it equals to zero.
2. Apply chain rule.
3. Simplify equation.

Recall:

$$\text{Linearity: } (f + g)' = f' + g'$$

$$\text{Product Rule: } (fg)' = f'g + fg'$$

$$\text{Quotient: } \left(\frac{f}{g}\right)' = \frac{f'g - fg'}{g^2}$$

$$\text{Chain Rule: } (f(g(x)))' = f'(g(x))g'(x)$$

$$\text{also denoted as: } \frac{\partial f}{\partial x} = \frac{\partial f}{\partial g} \frac{\partial g}{\partial x}$$

## Maximum Likelihood Estimation: Take Derivative w. r. t. $\sigma$

$$\begin{aligned}
 & \frac{\partial}{\partial \sigma} \ln (\mathcal{L}(\mu, \sigma | x_1, \dots, x_n)) \\
 &= \underbrace{\frac{\partial}{\partial \sigma} \left( -\frac{n}{2} \ln 2\pi \right)}_{1.} - \underbrace{\frac{\partial}{\partial \sigma} (n \ln \sigma)}_{2.} - \sum_{i=1}^n \underbrace{\frac{\partial}{\partial \sigma} \frac{(x_i - \mu)^2}{2\sigma^2}}_{3.} \\
 &= -\frac{n}{\sigma} - \underbrace{\sum_{i=1}^n \frac{(x_i - \mu)^2}{2} (-2) \sigma^{-3}}_{4.} \\
 &= -\frac{n}{\sigma} + \sum_{i=1}^n (x_i - \mu)^2 \underbrace{\sigma^{-3}}_{5.} \\
 &= \underbrace{-\frac{n}{\sigma} + \sum_{i=1}^n \frac{(x_i - \mu)^2}{\sigma^3}}_{6.} \\
 &= -\frac{n}{\sigma} + \frac{1}{\sigma^3} \sum_{i=1}^n (x_i - \mu)^2
 \end{aligned}$$

1. Derivative of this component can be treated as a constant as it does not contain  $\sigma$ , therefore it equals to zero.
2. Take derivative.
3. Easier when expressed as  $\frac{(x_i - \mu)^2}{2} \sigma^{-2}$ . Then take derivative of  $\sigma^{-2}$ . Recall:  $(x^a)' = ax^{a-1}$
4. Two minuses cancel out (minus before sum and minus of  $(-2)$ ). Additionally, simplify by cancel out  $\frac{2}{2}$ .
5. Put back as denominator.
6. Simplify equation.

## Maximum Likelihood Estimation: Two Derivatives w. r. t. $\mu$ and $\sigma$

Derivatives are:

$$\frac{\partial}{\partial \mu} \ln (\mathcal{L}(\mu, \sigma | x_1, \dots, x_n)) = \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu)$$

$$\frac{\partial}{\partial \sigma} \ln (\mathcal{L}(\mu, \sigma | x_1, \dots, x_n)) = -\frac{n}{\sigma} + \frac{1}{\sigma^3} \sum_{i=1}^n (x_i - \mu)^2$$

To find the estimates of  $\mu$  and  $\sigma$ , solve these equations by equal them to zero<sup>3</sup>:

- For  $\mu$ :  $0 = \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu) \xrightarrow{\times \sigma^2} 0 = \sum_{i=1}^n x_i - n\mu \xrightarrow{+n\mu} n\mu = \sum_{i=1}^n x_i \xrightarrow{\times \frac{1}{n}} \underline{\underline{\mu = \frac{1}{n} \sum_{i=1}^n x_i}}$

This equals to mean.

- For  $\sigma$ :  $0 = -\frac{n}{\sigma} + \frac{1}{\sigma^3} \sum_{i=1}^n (x_i - \mu)^2 \xrightarrow{\times \sigma^3} 0 = -n\sigma + \frac{1}{\sigma} \sum_{i=1}^n (x_i - \mu)^2 \xrightarrow{+n\sigma} n\sigma = \frac{1}{\sigma} \sum_{i=1}^n (x_i - \mu)^2$   
 $\xrightarrow{\times \sigma^2} n\sigma^2 = \sum_{i=1}^n (x_i - \mu)^2 \xrightarrow{\times \frac{1}{n}} \sigma^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \mu)^2 \xrightarrow{\sqrt{\phantom{x}}} \underline{\underline{\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \mu)^2}}}$

This equals to standard deviation.

<sup>3</sup>We want to find the value for which the log functions reach their maximum. At this point, the slope of these functions equals zero. Therefore, we equal these functions to zero.