

## **Outlier Analysis**

*Francesco Marcelloni*

Department of Information Engineering  
University of Pisa  
ITALY

Some slides belong to the collection


Jiawei Han, Micheline Kamber, and Jian Pei  
University of Illinois at Urbana-Champaign  
Simon Fraser University

©2011 Han, Kamber, and Pei. All rights reserved.

1

## **Chapter 12. Outlier Analysis**

---

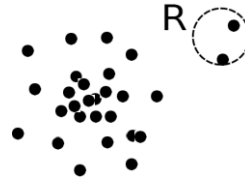
- Outlier and Outlier Analysis 
- Outlier Detection Methods
- Statistical Approaches
- Proximity-Base Approaches
- Clustering-Base Approaches
- Classification Approaches
- Mining Contextual and Collective Outliers
- Outlier Detection in High Dimensional Data
- Summary

2

2

## What Are Outliers?

- **Outlier**: A data object that **deviates significantly** from the normal objects as if it were **generated by a different mechanism**
  - Ex.: Unusual credit card purchase, sports: Michael Jordan, Wayne Gretzky, ...
- **Outliers are different from the noise data**
  - Noise is random error or variance in a measured variable
  - Noise should be removed before outlier detection
- Outliers are interesting: They violate the mechanism that generates the normal data
- Outlier detection vs. *novelty detection*: early stage, outlier; but later merged into the model
- Applications:
  - Credit card fraud detection
  - Telecom fraud detection
  - Customer segmentation
  - Medical analysis

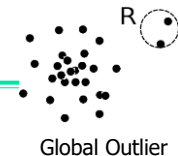


3

3

## Types of Outliers (I)

- Three kinds: *global*, *contextual* and *collective* outliers

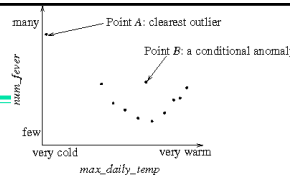


- **Global outlier** (or point anomaly)
  - Object is  $O_g$  if it significantly deviates from the rest of the data set
  - Ex. Intrusion detection in computer networks, fault detection in industry
  - **Issue**: Find an appropriate measurement of deviation

4

4

## Types of Outliers (II)



Contextual Outlier

- **Contextual outlier** (or *conditional outlier*)
  - Object is  $O_c$  if it deviates significantly based on a selected context
  - Ex. 35° C in Pisa: outlier? (depending on summer or winter?)
  - Attributes of data objects should be divided into two groups
    - **Contextual attributes:** defines 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*—whose density significantly deviates from its local area
  - **Issue:** How to define or formulate meaningful context?

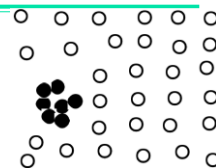
5

5

## Types of Outliers (III)

### ■ **Collective Outliers**

- A subset of data objects *collectively* deviate significantly from the whole data set, even if the individual data objects may not be outliers
- Applications: E.g., *intrusion detection*:
  - When 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.
- A data set may have multiple types of outlier
- One object may belong to more than one type of outlier



Collective Outlier

6

6

## Challenges of Outlier Detection

---

- **Modeling normal objects and outliers properly**
  - Hard to enumerate all possible normal behaviors in an application
  - The border between normal and outlier objects is often a gray area
- **Application-specific outlier detection**
  - Choice of distance measure among objects and the model of relationship among objects are often application-dependent
  - E.g., clinic data: a small deviation could be an outlier; while in marketing analysis, larger fluctuations

7

7

## Challenges of Outlier Detection

---

- **Handling noise in outlier detection**
  - Noise may distort the normal objects and blur the distinction between normal objects and outliers. It may help 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

8

8

## Chapter 12. Outlier Analysis

---

- Outlier and Outlier Analysis
- Outlier Detection Methods 
- Statistical Approaches
- Proximity-Base Approaches
- Clustering-Base Approaches
- Classification Approaches
- Mining Contextual and Collective Outliers
- Outlier Detection in High Dimensional Data
- Summary

9

9

## Outlier Detection

---

- Two ways to categorize outlier detection methods:
  - Based on **whether user-labeled examples of outliers can be obtained**:
    - Supervised, semi-supervised vs. unsupervised methods
  - Based on **assumptions about normal data and outliers**:
    - Statistical, proximity-based, and clustering-based methods

10

10

## Outlier Detection I: Supervised Methods

### 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, or
  - 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, i.e., recall is more important than accuracy (i.e., not mislabeling normal objects as outliers)**

11

11

## Outlier Detection II: Unsupervised Methods

### Unsupervised Methods

- Assume the normal objects are somewhat ``clustered'' into multiple groups, each having some distinct features
- **An outlier is expected to be far away from any groups of normal objects**
- **Weakness: Cannot detect collective outlier effectively**
  - Normal objects may not share any strong patterns, but the collective outliers may share high similarity in a small area
- Ex. In some intrusion or virus detection, normal activities are diverse
  - **Unsupervised methods may have a high false positive rate but still miss many real outliers.**
  - Supervised methods can be more effective, e.g., identify attacking some key resources
- **Many clustering methods can be adapted for unsupervised methods**
  - Find clusters, then outliers: not belonging to any cluster
  - Problem 1: **Hard to distinguish noise from outliers**
  - Problem 2: **Costly since first clustering: but far less outliers than normal objects**
    - Newer methods: **tackle outliers directly**

12

12

## Outlier Detection III: Semi-Supervised Methods

### Semi-supervised methods

- Situation: In many applications, the number of labeled data is often small: Labels could be on outliers only, normal objects only, or both
- **Semi-supervised outlier detection:** Regarded as applications of semi-supervised learning
- 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, a small number of labeled outliers may not cover the possible outliers well
  - To improve the quality of outlier detection, one can get help from models for normal objects learned from unsupervised methods

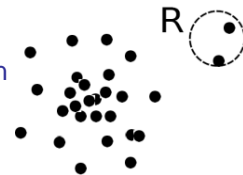
13

13

## Outlier Detection (1): Statistical Techniques

Outlier Detection Techniques: **Statistical techniques** (also known as model-based methods) assume that the normal data follow some statistical model (a stochastic model)

- **The data not following the model are outliers.**
- Example (right figure): First use Gaussian distribution to model the normal data
  - For each object  $y$  in region  $R$ , estimate  $g_D(y)$ , the probability of  $y$  fits the Gaussian distribution
  - If  $g_D(y)$  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 rich alternatives to use various statistical models
  - E.g., parametric vs. non-parametric

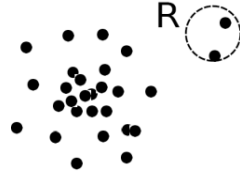


14

14

## Outlier Detection (2): Proximity-Based Tech.

- Outlier Detection Techniques: **proximity-based**. 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



15

15

## Outlier Detection (2): Proximity-Based Methods

- The 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 stay close to each other**
- Two major types of proximity-based outlier detection
  - Distance-based vs. density-based

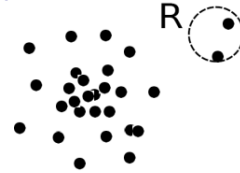
16

16



## Outlier Detection (3): Clustering-Based Methods

- Outlier Detection Techniques: **Clustering-based**. Normal data belong to large and dense clusters, whereas outliers belong to small or sparse clusters, or do not belong to any clusters
- 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
- Since there are many clustering methods, there are many clustering-based outlier detection methods as well
- **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



17

17

## Chapter 12. Outlier Analysis

- Outlier and Outlier Analysis
- Outlier Detection Methods
- Statistical Approaches 
- Proximity-Based Approaches
- Clustering-Based Approaches
- Classification Approaches
- Mining Contextual and Collective Outliers
- Outlier Detection in High Dimensional Data
- Summary

18

18

## Statistical Approaches

- Statistical approaches assume that the 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 are divided into two categories: *parametric* vs. *non-parametric*
- **Parametric method**
  - 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
  - The smaller this value, the more likely  $x$  is an outlier
- **Non-parametric method**
  - Not assume an a-priori statistical model and determine the model from the input data
  - Not completely parameter free but consider the number and nature of the parameters are flexible and not fixed in advance
  - Examples: histogram and kernel density estimation

19

19

## Parametric Methods I: Detection Univariate Outliers Based on Normal Distribution

- **Univariate data:** A data set involving only one attribute or variable
- Often assume that data are generated from a **normal distribution**, learn the parameters from the input data, and identify the points with low probability as outliers
- How is it possible to determine whether a distribution is normal?
  - **Shapiro-Wilk Test:** test for verifying whether a random sample comes from a normal distribution
  - **Q-Q (Quantile-Quantile) plot:** in general, this plot can be used to estimate whether a random sample comes from a continuous distribution (for instance, a normal distribution) as long as the quantiles can be calculated

20

20

## Parametric Methods I: Detection Univariate Outliers Based on Normal Distribution

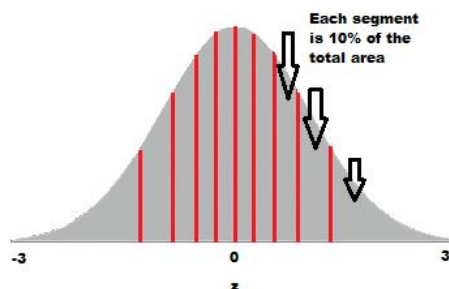
- **Q-Q plot: plots of two quantiles against each other.** Recall: a quantile is a fraction where certain values fall below that quantile
- To verify whether a random sample comes from a normal distribution we **plot the quantiles corresponding to the random samples against the quantile of a normal distribution**
- An example. Let us assume that we have to determine whether the following samples from a normal distribution  
7.19, 6.31, 5.89, 4.5, 3.77, 4.25, 5.19, 5.79, 6.79
- We adopt the following procedure

21

21

## Parametric Methods I: Detection Univariate Outliers Based on Normal Distribution

- **Step 1. Sort the values from the smallest to the largest**  
3.77, 4.25, 4.50, 5.19, 5.79, 5.89, 6.31, 6.79, 7.19
- **Step 2. Draw a normal distribution curve and divide it into  $n+1$  segments, where  $n$  is the number of values. In our example**



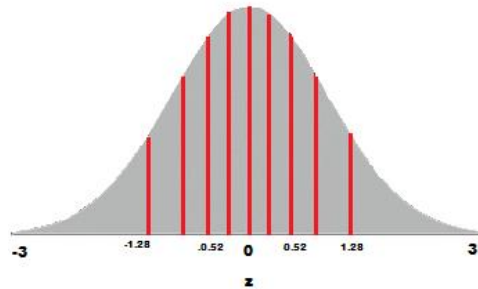
22

22

## Parametric Methods I: Detection Univariate Outliers Based on Normal Distribution

- **Step 3.** Find the z-value (cut-off point for each segment). Recall the z-value is the number of standard deviations from the mean a data point is

10% = -1.28
20% = -0.84
30% = -0.52
40% = -0.25
50% = 0
60% = 0.25
70% = 0.52
80% = 0.84
90% = 1.28
100% = 3.0

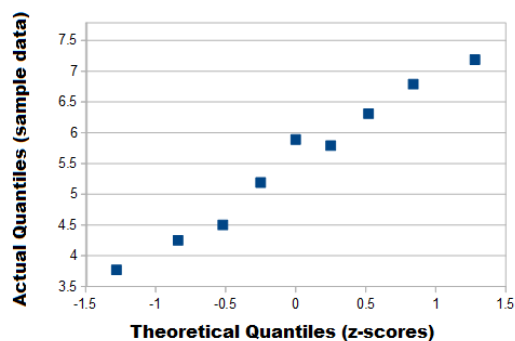


23

23

## Parametric Methods I: Detection Univariate Outliers Based on Normal Distribution

- **Step 4.** Plot the data set values against the normal distribution cut-off points.
- A (almost) straight line on the Q-Q plot indicates that the data distribution is approximately normal.



24

24

## Parametric Methods I: Detection Univariate Outliers Based on Normal Distribution

- Method 1: model the normal distribution and consider the probability of the points to belong to this distribution

Ex: Avg. temp.: {24.0, 28.9, 28.9, 29.0, 29.1, 29.1, 29.2, 29.2, 29.3, 29.4}

- Use the maximum likelihood method to estimate  $\mu$  and  $\sigma$ 
  - The probability that a point  $x_i$  is generated by the model is

$$P(x_i|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

- Consequently, the likelihood that  $X$  is generated by the model is

$$L(\mathcal{N}(\mu, \sigma^2) : X) = P(X|\mu, \sigma^2) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

25

25

## Parametric Methods I: Detection Univariate Outliers Based on Normal Distribution

- The task of learning the generative model is to find the parameters such that the likelihood is maximized, that is, finding

$$\mathcal{N}(\mu_0, \sigma_0^2) = \arg \max \{L(\mathcal{N}(\mu, \sigma^2) : X)\}$$

$$\ln \mathcal{L}(\mu, \sigma^2) = \sum_{i=1}^n \ln f(x_i|\mu, \sigma^2) = -\frac{n}{2} \ln(2\pi) - \frac{n}{2} \ln \sigma^2 - \frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2$$

- Taking derivatives with respect to  $\mu$  and  $\sigma^2$ , we obtain the following maximum likelihood estimates

$$\hat{\mu} = \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad \hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2$$

26

26

## Parametric Methods I: Detection Univariate Outliers Based on Normal Distribution

- For the above data with  $n = 10$ , we have

$$\hat{\mu} = 28.61 \qquad \hat{\sigma} = \sqrt{2.29} = 1.51$$

- Then  $(24 - 28.61) / 1.51 = -3.04 < -3$ , Thus, 24 is an outlier since

$\mu \pm 3\sigma$  region contains 99.7% data

27

27

## Parametric Methods I: The Grubb's Test

- **Method 2: The Grubb's test** (maximum normed residual test) — another statistical method under normal distribution. The test finds whether a minimum value or a maximum value is an outlier.
- The test checks for outliers by looking for **the maximum of the absolute differences between the values and the mean**. Basically, the steps are:
  1. Find the G test statistic.
  2. Find the G Critical Value.
  3. Compare the test statistic to the G critical value.
  4. Reject the point as an outlier if the test statistic is greater than the critical value.

28

28

# Parametric Methods I: The Grubb's Test

- 1. Find the G test statistics:
  - a) Order the data points from smallest to largest.
  - b) Find the mean ( $\bar{Y}$ ) and standard deviation (s) of the data set.
  - c) Calculate the G test statistic using the following equation:

$$G = \frac{\max_{i=1,\dots,N} |Y_i - \bar{Y}|}{s}$$

29

29

# Parametric Methods I: The Grubb's Test

- 2. Find the G critical value:

Several tables exist for finding the critical value for Grubbs' test. The table on the right is a partial table for several G critical values and alpha levels.

Manually, you can find the G critical value

$$G > \frac{N - 1}{\sqrt{N}} \sqrt{\frac{t^2_{\alpha/(2N), N-2}}{N - 2 + t^2_{\alpha/(2N), N-2}}}$$

where  $t^2_{\alpha/(2N), N-2}$  is the value taken by a t-distribution with (N-2) degrees of freedom at a significance level of  $\alpha/(2N)$

Alpha					
N	0.1	0.075	0.05	0.025	0.01
3	1.15	1.15	1.15	1.15	1.15
4	1.42	1.44	1.46	1.48	1.49
5	1.6	1.64	1.67	1.71	1.75
6	1.73	1.77	1.82	1.89	1.94
7	1.83	1.88	1.94	2.02	2.1
8	1.91	1.96	2.03	2.13	2.22
9	1.98	2.04	2.11	2.21	2.32
10	2.03	2.1	2.18	2.29	2.41
11	2.09	2.14	2.23	2.36	2.48
12	2.13	2.2	2.29	2.41	2.55
13	2.17	2.24	2.33	2.46	2.61
14	2.21	2.28	2.37	2.51	2.66
15	2.25	2.32	2.41	2.55	2.71
16	2.28	2.35	2.44	2.59	2.75
17	2.31	2.38	2.47	2.62	2.79

30

30

# Parametric Methods I: The Grubb's Test

- 3. Compare the test statistic to the G critical value
- Compare the G test statistic to the G critical value
  - a)  $G_{test} < G_{critical}$ : keep the sample in the data set; it is not an outlier.
  - b)  $G_{test} \geq G_{critical}$ : reject the sample as an outlier

31

31

# Parametric Methods I: The Grubb's Test

- An example of application

145
125
190
135
220
130
210
3
165
165
150

min	3
mean	148.9091
stdev	57.81082
G	2.523906
alpha	0.05
N	11

- In the table, we find a  $G_{crit} = 2.23$ . Thus,  $G > G_{crit}$  and therefore G is an outlier

32

32



## Parametric Methods II: Detection of Multivariate Outliers

- **Multivariate data:** A data set involving two or more attributes or variables
- Transform the multivariate outlier detection task into a univariate outlier detection problem
- **Method 1. Use the Mahalanobis distance**
  - Mahalanobis' distance is the distance between a point and a distribution. And not between two distinct points. It is effectively a multivariate equivalent of the Euclidean distance.
  - Actually, the **Euclidean distance works fine as long as the dimensions are equally weighted and are independent of each other**

33

33

## Parametric Methods II: Detection of Multivariate Outliers

- **Equally weighted.** The two following tables show the 'area' and 'price' of the same objects. Only the units of the variables change. Nevertheless, the distances between any two rows are different.

Area (sq.ft)	Price (\$ 1000's)	Area (acre)	Price (\$M)
2400	156000	0.0550944	156
1950	126750	0.0447642	126.75
2100	105000	0.0482076	105
1200	78000	0.0275472	78
2000	130000	0.045912	130
900	54000	0.0206604	54

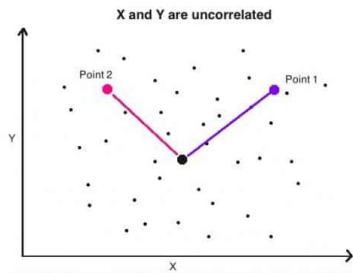
The problem can be overcome by scaling the variables, by computing the z-score  $((x - \text{mean}) / \text{std})$  or making it vary within a specific range (for instance, between 0 and 1)

34

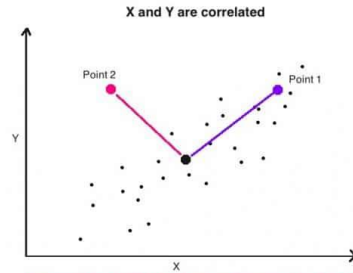
34

## Parametric Methods II: Detection of Multivariate Outliers

- **Independent of each other.** The Euclidean distance between a point and the center of the points (distribution) can give little or misleading information about how close a point really is to the cluster. When the variables X and Y are correlated Point 1 is closer to the cluster than point 2



When X and Y are uncorrelated, the Euclidean distance from the Centroid can be useful to infer if a point is member of the distribution. The further it is, the less likely it is a member.



Both Point 1 and Point 2 have the same Euclidean distance from centroid. But only Point 1 is a member of the distribution. To detect Point 2 as outlier, dist(Point 2, centroid) should be much higher than dist(Point 1, Centroid). Mahalanobis distance can be used here instead.

35

35

## Parametric Methods II: Detection of Multivariate Outliers

- How is the Mahalanobis distance different from the Euclidean distance?
  - Transforms the columns into uncorrelated variables
  - Scale the columns to make their variance equal to 1
  - Finally, it calculates the Euclidean distance
- Let  $\bar{\mathbf{o}}$  be the mean vector for a multivariate data set. The Mahalanobis distance for an object  $\mathbf{o}$  to  $\bar{\mathbf{o}}$  is defined as
 
$$MDist^2(\mathbf{o}, \bar{\mathbf{o}}) = (\mathbf{o} - \bar{\mathbf{o}})^T \mathbf{S}^{-1} (\mathbf{o} - \bar{\mathbf{o}})$$
 where  $\mathbf{S}$  is the covariance matrix.
- To divide by the covariance matrix is essentially a multivariate equivalent of the regular standardization  $z = (x - \text{mean}) / \text{std}$
- If the variables are strongly correlated, then the covariance will be high and the distance will be reduced; otherwise, the covariance will be low and the distance will not be reduced

36

36

## Parametric Methods II: Detection of Multivariate Outliers

- **Method 1. Use the Mahalaobis distance (continued)**
  - The multivariate outlier detection problem is transformed as follows
    1. Calculate the mean vector from the multivariate data set.
    2. For each object  $o$ , calculate  $MDist(o, \bar{o})$ , the Mahalanobis distance from  $o$  to  $\bar{o}$ .
    3. Detect outliers in the transformed univariate data set,  $\{MDist(o, \bar{o}) | o \in D\}$ .
    4. If  $MDist(o, \bar{o})$  is determined to be an outlier, then  $o$  is regarded as an outlier as well.
  - Use the Grubb's test on this measure to detect outliers

Weaknesses:

- Computationally heavy (covariance matrix and its inverse)
- Needs to store the covariance matrix and its inverse

37

37

## Parametric Methods II: Detection of Multivariate Outliers

- **Method 2. Use  $\chi^2$ -statistic:**
  - **Assumption:** the population of  $\mathbf{O}$  follows a multivariate distribution with the mean vector  $\bar{\mathbf{o}}$  and the covariance matrix  $\mathbf{S}$ .
  - The method exploits a distance measure based on the chi-square test statistic

$$\chi^2 = \sum_{i=1}^n \frac{(o_i - E_i)^2}{E_i}$$

where  $o_i$  and  $E_i$  are the observed value and the expected value of the  $i$ th variable and  $n$  is the number of variables. Using the average values as estimates of the expectation, we have

$$\chi^2 = \sum_{i=1}^n \frac{(X_i - \bar{X}_i)^2}{\bar{X}_i}$$

38

38

## Parametric Methods II: Detection of Multivariate Outliers

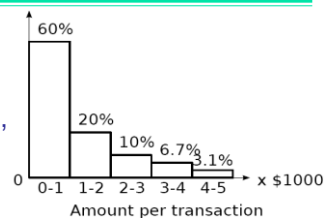
- **Method 2. Use  $\chi^2$ -statistic:**
  - According to the central limit theorem, when the number of variables is large enough (i.e., greater than 30),  $\chi^2$  as the sum of squared differences between the observed and the expected values of those variables has approximately a normal distribution.
  - Since we are interested in detecting significantly large  $\chi^2$  values for intrusion detection, we need to set only the upper control limit  $\overline{\chi^2} + 3S_{\chi^2}$ , that is, if the computed  $\chi^2$  for an observation is greater than  $\overline{\chi^2} + 3S_{\chi^2}$  we signal an anomaly.

39

39

## Non-Parametric Methods: Detection Using Histogram

- The model of normal data is learned from the input data without any *a priori* structure.
- Often makes fewer assumptions about the data, and thus can be applicable in more scenarios
- Outlier detection using histogram:
  - Figure shows the histogram of purchase amounts in transactions
  - A transaction in the amount of \$7,500 is an outlier, since only 0.2% transactions have an amount higher than \$5,000
- **Problem:** Hard to choose an appropriate bin size for histogram
  - Too small bin size → normal objects in empty/rare bins, false positive
  - Too big bin size → outliers in some frequent bins, false negative
- **Solution:** Adopt kernel density estimation to estimate the probability density distribution of the data. If the estimated density function is high, the object is likely normal. Otherwise, it is likely an outlier.



40

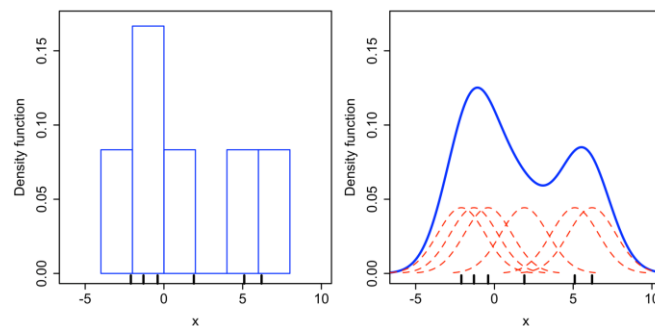
40

## Non-Parametric Methods: Kernel Density Estimation

- Kernel Density Estimation

$$\hat{f}_h(x) = \frac{1}{n} \sum_{i=1}^n K_h(x - x_i) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right),$$

where  $K(\bullet)$  is the kernel — a symmetric but not necessarily positive function that integrates to one — and  $h > 0$  is a smoothing parameter called the bandwidth.



41

41

## Statistical Methods: Computational cost


- The computational cost of statistical methods depends on the models.
- Simple parametric models (e.g., a Gaussian)
  - Linear time.
- More sophisticated models (e.g., mixture models, where the Expectation Maximization (EM) algorithm is used in learning)
  - several iterations. Each iteration, however, is typically linear with respect to the data set's size.
  - For kernel density estimation, the model learning cost can be up to quadratic.
- Once the model is learned, the outlier detection cost is often very small per object.

42

42

## Chapter 12. Outlier Analysis

---

- Outlier and Outlier Analysis
- Outlier Detection Methods
- Statistical Approaches
- Proximity-Based Approaches 
- Clustering-Based Approaches
- Classification Approaches
- Mining Contextual and Collective Outliers
- Outlier Detection in High Dimensional Data
- Summary

43

43

### Proximity-Based Approaches: Distance-Based vs. Density-Based Outlier Detection

---

- **Intuition:** Objects that are far away from the others are outliers
- **Assumption of proximity-based approach:** The proximity of an outlier deviates significantly from that of most of the others in the data set
- Two types of proximity-based outlier detection methods
  - **Distance-based outlier detection:** An object  $o$  is an outlier if its neighborhood does not have enough other points
  - **Density-based outlier detection:** An object  $o$  is an outlier if its density is relatively much lower than that of its neighbors

44

44

## Distance-Based Outlier Detection

- For each object  $o$ , examine the # of other objects in the  $r$ -neighborhood of  $o$ , where  $r$  is a user-specified **distance threshold**
- An object  $o$  is an outlier if most (taking  $\pi$  as a **fraction threshold**) of the objects in  $D$  are far away from  $o$ , i.e., not in the  $r$ -neighborhood of  $o$
- An object  $o$  is a  $DB(r, \pi)$  outlier if 
$$\frac{||\{o' | dist(o, o') \leq r\}||}{||D||} \leq \pi$$
- Equivalently, one can check the distance between  $o$  and its  $k$ -th nearest neighbor  $o_k$ , where  $k = \lceil \pi ||D|| \rceil$ .  $o$  is an outlier if  $dist(o, o_k) > r$
- **Efficient computation:** Nested loop algorithm
  - For any object  $o_i$ , calculate its distance from other objects, and count the # of other objects in the  $r$ -neighborhood.
  - If  $\pi \cdot n$  other objects are within  $r$  distance, terminate the inner loop
  - Otherwise,  $o_i$  is a  $DB(r, \pi)$  outlier
- Efficiency: Actually CPU time is not  $O(n^2)$  but linear to the data set size since for most non-outlier objects, the inner loop terminates early

45

45

## Distance-Based Outlier Detection

**Algorithm:** Distance-based outlier detection.

**Input:**

- a set of objects  $D = \{o_1, \dots, o_n\}$ , threshold  $r$  ( $r > 0$ ) and  $\pi$  ( $0 < \pi \leq 1$ );

**Output:**  $DB(r, \pi)$  outliers in  $D$ .

**Method:**

```

for  $i = 1$  to  $n$  do
     $count \leftarrow 0$ 
    for  $j = 1$  to  $n$  do
        if  $i \neq j$  and  $dist(o_i, o_j) \leq r$  then
             $count \leftarrow count + 1$ 
            if  $count \geq \pi \cdot n$  then
                exit  $\{o_i \text{ cannot be a } DB(r, \pi) \text{ outlier}\}$ 
            endif
        endif
    endfor
    print  $o_i$   $\{o_i \text{ is a } DB(r, \pi) \text{ outlier according to (Eq. 12.10)}\}$ 
endfor;

```

46

46

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

- Why efficiency is still a concern? When the complete set of objects cannot be held into main memory, cost I/O swapping
- The major cost:** (1) each object tests against the whole data set, why not only its close neighbor? (2) check objects one by one, why not group by group?
- Grid-based method (CELL):** Data space is partitioned into a multi-D grid. Each cell is a hyper cube with diagonal length  $r/2$
- Pruning using the level-1 & level 2 cell properties:
  - Level-1:** For any possible point  $x$  in cell  $C$  and any possible point  $y$  in a level-1 cell (one cell away from  $C$ )  $\text{dist}(x,y) \leq r$
  - Level-2:** For any possible point  $x$  in cell  $C$  and any point  $y$  in a level-2 cell (two cells away from  $C$ )  $\text{dist}(x,y) \geq r$

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

47

47

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

- Let  $a$  be the number of objects in cell  $C$ ,  $b_1$  be the total number of objects in the level-1 cells, and  $b_2$  be the total number of objects in the level-2 cells. We can apply the following rules.
  - Level-1 cell pruning rule:** Based on the level-1 cell property, if  $a+b_1 > [\pi n]$ , then every object  $o$  in  $C$  is not a  $DB(r, \pi)$ -outlier because all those objects in  $C$  and the level-1 cells are in the  $r$ -neighborhood of  $o$ , and there are at least  $[\pi n]$  such neighbors.
  - Level-2 cell pruning rule:** Based on the level-2 cell property, if  $a+b_1+b_2 < [\pi n] + 1$ , then all objects in  $C$  are  $DB(r, \pi)$ -outliers because each of their  $r$ -neighborhoods has less than  $[\pi n]$  other objects.
- Thus we only need to check the objects that cannot be pruned, and even for such an object  $o$ , only need to compute the distance between  $o$  and the objects in the level-2 cells (since beyond level-2, the distance from  $o$  is more than  $r$ )

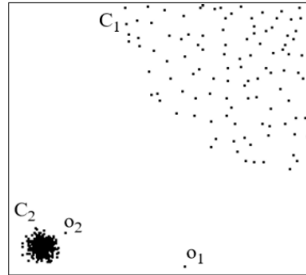
48

48



## Local Distance-Based Outlier Detection

- Problem with different densities



- Outlier  $o_2$  has similar density as elements of cluster  $C_1$ .
- Solution: outlierness
  - Is point relatively far away from its neighbors?

49

49

## Local Distance-Based Outlier Detection

- Let  $N_k(x_i)$  be the  $k$ -nearest neighbors of  $x_i$
- Let  $D_k(x_i)$  be the average distance to  $k$ -nearest neighbors

$$D_k(x_i) = \frac{1}{k} \sum_{j \in N_k(x_i)} \|x_i - x_j\|$$

- Outlierness is the ratio of  $D_k(x_i)$  to average  $D_k(x_j)$  for its neighbors  $j$

$$O_k(x_i) = \frac{D_k(x_i)}{\frac{1}{k} \sum_{j \in N_k(x_i)} D_k(x_j)}$$

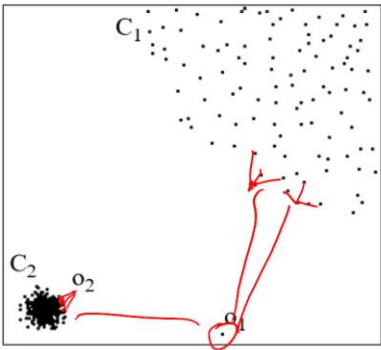
- If outlierness  $> 1$ ,  $x_i$  is further away from neighbors than expected

50

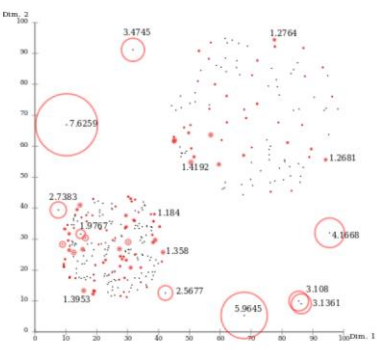
50

# Local Distance-Based Outlier Detection

- Outlierness finds  $o_1$  and  $o_2$ :



- More complicated data:

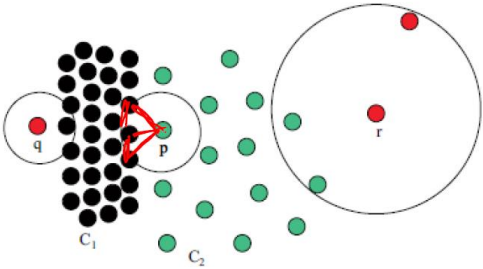


51

51

# Local Distance-Based Outlier Detection

- If clusters are close, outlierness gives unintuitive results



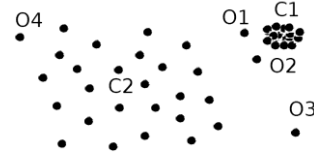
- In this example, p has higher outlierness than q and r:
  - The green points are not part of the KNN list of p for small k

52

52

## Density-Based Outlier Detection

- **Local outliers:** Outliers comparing to their local neighborhoods, instead of the global data distribution
- In Fig.,  $o_1$  and  $o_2$  are local outliers to  $C_1$ ,  $o_3$  is a global outlier, but  $o_4$  is not an outlier. However, proximity-based clustering cannot find that  $o_1$  and  $o_2$  are outliers (e.g., comparing with  $O_4$ ).
- **Intuition (density-based outlier detection):** The density around an outlier object is significantly different from the density around its neighbors
- **Method:** Use the relative density of an object against its neighbors as the indicator of the degree of the object being outlier
- **k-distance of an object o**,  $\text{dist}_k(o)$ : distance between o and its k-th NN
- **k-distance neighborhood of o**,  $N_k(o) = \{o' \mid o' \text{ in } D, \text{dist}(o, o') \leq \text{dist}_k(o)\}$ 
  - $N_k(o)$  could be bigger than k since multiple objects may have identical distance to o



53

53

## Local Outlier Factor: LOF

- **Reachability distance from  $o'$  to  $o$ :**

$$\text{reachdist}_k(o \leftarrow o') = \max\{\text{dist}_k(o), \text{dist}(o, o')\}$$

where k is a user-specified parameter

- **Local reachability density of o:**

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

54

54

## Local Outlier Factor: LOF

- **LOF (Local outlier factor)** of an object  $o$  is the average of the ratio of local reachability of  $o$  and those of  $o$ 's  $k$ -nearest neighbors

$$LOF_k(o) = \frac{\sum_{o' \in N_k(o)} \frac{lrd_k(o')}{lrd_k(o)}}{\|N_k(o)\|} = \sum_{o' \in N_k(o)} lrd_k(o') \cdot \sum_{o' \in N_k(o)} reachdist_k(o' \leftarrow o)$$

- The lower the local reachability density of  $o$ , and the higher the local reachability density of the  $k$ NN of  $o$ , the higher LOF
- This captures a local outlier whose local density is relatively low comparing to the local densities of its  $k$ NN

55

55

## LOF: an example

- Consider the following 4 data points  
 $a(0,0)$ ,  $b(0,1)$ ,  $c(1,1)$ ,  $d(3,0)$
- Compute the distance between the four points (Manhattan distance)
  - $dist(a,b) = 1$
  - $dist(a,c) = 2$
  - $dist(a,d) = 3$
  - $dist(b,c) = 1$
  - $dist(b,d) = 4$
  - $dist(c,d) = 3$
- Let us consider  $k=2$ . Then
 

$dist_2(a) = dist(a,c) = 2$	$N_2(a) = \{b,c\}$
$dist_2(b) = dist(b,a) = 1$	$N_2(b) = \{a,c\}$
$dist_2(c) = dist(c,a) = 2$	$N_2(c) = \{b,a\}$
$dist_2(d) = dist(d,a) = 3$	$N_2(d) = \{a,c\}$

57

57

### LOF: an example

$$lrd_2(a) = \frac{\|N_2(a)\|}{reachdist_2(b \leftarrow a) + reachdist_2(c \leftarrow a)}$$

$$reachdist_2(b \leftarrow a) = \max\{dist_2(b), dist(b, a)\} = \max\{1, 1\} = 1$$

$$reachdist_2(c \leftarrow a) = \max\{dist_2(c), dist(c, a)\} = \max\{2, 2\} = 2$$

Thus

$$lrd_2(a) = \frac{\|N_2(a)\|}{reachdist_2(b \leftarrow a) + reachdist_2(c \leftarrow a)} = \frac{2}{1 + 2} = 0.667$$

$$lrd_2(b) = \frac{\|N_2(b)\|}{reachdist_2(a \leftarrow b) + reachdist_2(c \leftarrow b)} = \frac{2}{2 + 2} = 0.5$$

$$lrd_2(c) = \frac{\|N_2(c)\|}{reachdist_2(b \leftarrow c) + reachdist_2(a \leftarrow c)} = \frac{2}{1 + 2} = 0.667$$

$$lrd_2(d) = \frac{\|N_2(d)\|}{reachdist_2(a \leftarrow d) + reachdist_2(c \leftarrow d)} = \frac{2}{3 + 3} = 0.33$$

58

58

### LOF: an example

$$LOF_2(a) = (lrd_2(b) + lrd_2(c))(reachdist_2(b \leftarrow a) + reachdist_2(c \leftarrow a)) \\ = (0.5 + 0.667)(1 + 2) = 3.501$$

$$LOF_2(b) = (lrd_2(a) + lrd_2(c))(reachdist_2(a \leftarrow b) + reachdist_2(c \leftarrow b)) \\ = (0.667 + 0.667)(2 + 2) = 5.336$$

$$LOF_2(c) = (lrd_2(b) + lrd_2(a))(reachdist_2(b \leftarrow c) + reachdist_2(a \leftarrow c)) \\ = (0.5 + 0.667)(1 + 2) = 3.501$$

$$LOF_2(d) = (lrd_2(a) + lrd_2(c))(reachdist_2(a \leftarrow d) + reachdist_2(c \leftarrow d)) \\ = (0.667 + 0.667)(3 + 3) = 8.004$$

59

59

## LOF: an example

---

The sorted order is

$$LOF_2(d) = 8.004$$

$$LOF_2(b) = 5.336$$

$$LOF_2(a) = 3.501$$

$$LOF_2(c) = 3.501$$

Obviously, top 1 outlier is point d

60

60

## Chapter 12. Outlier Analysis

---

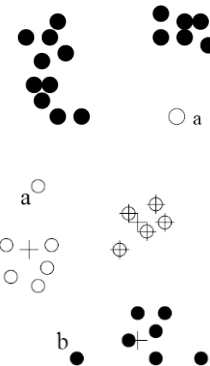
- Outlier and Outlier Analysis
- Outlier Detection Methods
- Statistical Approaches
- Proximity-Based Approaches
- Clustering-Based Approaches 
- Classification Approaches
- Mining Contextual and Collective Outliers
- Outlier Detection in High Dimensional Data
- Summary

61

61

## Clustering-Based Outlier Detection (1 & 2): Not belong to any cluster, or far from the closest one

- An object is an outlier if (1) it does not belong to any cluster, (2) there is a large distance between the object and its closest cluster, or (3) it belongs to a small or sparse cluster
- **Case 1:** Not belong to any cluster
  - Identify animals not part of a flock: Using a density-based clustering method such as DBSCAN
- **Case 2:** Far from its closest cluster
  - Using k-means, partition data points into clusters
  - For each object  $o$ , assign an outlier score based on its distance from its closest center
    - If  $\text{dist}(o, c_o) / \text{avg\_dist}(c_o)$  is large, likely an outlier
- Ex. Intrusion detection: Consider the similarity between data points and the clusters in a training data set
  - Use a training set to find patterns of “normal” data, e.g., frequent itemsets in each segment, and cluster similar connections into groups
  - Compare new data points with the clusters mined—Outliers are possible attacks (TCP connection data example)

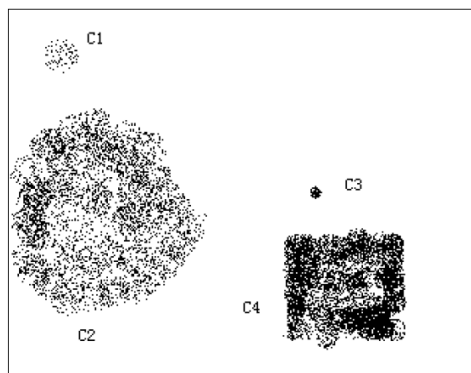


62

62

## Clustering-Based Outlier Detection (3): Detecting Outliers in Small Clusters

- Each of the approaches seen so far detects only individual objects as outliers.
- In a large data set, **some outliers may form a small cluster**. In intrusion detection, for example, hackers who use similar tactics to attack a system. For instance, in the following figure C1 and C3 should be regarded as outliers



63

63

### Clustering-Based Outlier Detection (3): Detecting Outliers in Small Clusters

- To identify the physical significance of the definition of an outlier, an outlier factor, namely **CBLOF (Cluster-Based Local Outlier Factor)** is associated with each object.
- **CBLOF measures both the size of the cluster the object belongs to and the distance between the object and its closest cluster**
- Given two parameters  $\alpha$  and  $\beta$ , we define  $b$  as the boundary of large and small clusters if one of the following formulas holds

$$\begin{cases} (|C_1| + |C_2| + \dots + |C_b|) \geq |D|\alpha & (1) \\ |C_b|/|Cb + 1| \geq \beta & (2) \end{cases}$$

- (1) most data points in the dataset are not outliers (for instance  $\alpha=90\%$ )
- (2) large and small clusters should have significant differences in size (for instance  $\beta = 5$ )
- The set of large clusters is defined as  $LC = \{C_i | i \leq b\}$  and the set of small clusters is defined as  $SC = \{C_j | j > b\}$

64

64

### Clustering-Based Outlier Detection (3): Detecting Outliers in Small Clusters

- For each object  $o$ , **CBLOF (Cluster-Based Local Outlier Factor)** is defined as:

$$CBLOF(o) = \begin{cases} |C_i| \cdot \min(\text{distance}(o, C_j)) & o \in C_i, C_i \in SC \text{ and } C_j \in LC \\ |C_i| \cdot \text{distance}(o, C_i) & o \in C_i \text{ and } C_i \in LC \end{cases}$$

- For the computation of the distance between the object and the cluster, it is sufficient to adopt the similarity measure used in the clustering algorithm
- FindCBLOF algorithm:
  - Cluster the dataset
  - Compute the value of CBLOF for each object

65

65




## Clustering-Based Method: Strength and Weakness

- **Strength**
  - Detect outliers without requiring any labeled data
  - Work for many types of data
  - Clusters can be regarded as summaries of the data
  - Once the clusters are obtained, need only compare any object against the clusters to determine whether it is an outlier (fast)
- **Weakness**
  - Effectiveness depends highly on the clustering method used—they may not be optimized for outlier detection
  - High computational cost: Need to first find clusters
  - A method to reduce the cost: **Fixed-width clustering**
    - A point is assigned to a cluster if the center of the cluster is within a pre-defined distance threshold from the point
    - If a point cannot be assigned to any existing cluster, a new cluster is created and the distance threshold may be learned from the training data under certain conditions

66

## Chapter 12. Outlier Analysis

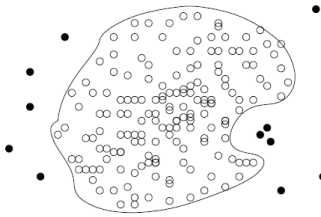
- Outlier and Outlier Analysis
- Outlier Detection Methods
- Statistical Approaches
- Proximity-Base Approaches
- Clustering-Base Approaches
- Classification Approaches 
- Mining Contextual and Collective Outliers
- Outlier Detection in High Dimensional Data
- Summary

67

67

## Classification-Based Method I: One-Class Model

- **Idea:** Train a classification model that can distinguish “normal” data from outliers
- **A brute-force approach:** Consider a training set that contains samples labeled as “normal” and others labeled as “outlier”
  - But, the training set is typically heavily biased: number of “normal” samples likely far exceeds number of outlier samples
  - Necessity to re-balance by using undersampling or oversampling
- Cannot detect unseen anomaly

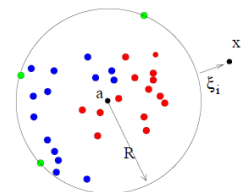


68

68

## Classification-Based Method I: One-Class Model

- **One-class model:** A classifier is built to describe only the normal class.
  - Learn the decision boundary of the normal class
  - Any samples that do not belong to the normal class (not within the decision boundary) are declared as outliers
  - **The most popular approach Support Vector Data Description (SVDD)**
    - Constructs a hyper-sphere around the positive class data that encompassed almost all points in the data set with the minimum radius.
    - The SVDD classifier classifies a given test point as outlier if it falls outside the hyper- sphere. However, SVDD can reject some fraction of positively labelled data when the volume of the hyper- sphere decreases.



69

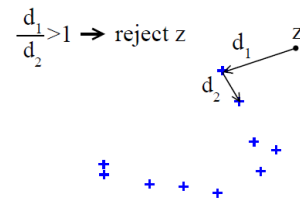
69

## Classification-Based Method I: One-Class Model

- Another approach Nearest Neighbour Description (NN-d), a variant of the Nearest Neighbor method
  - A test object  $z$  is accepted as a member of target class provided that its local density is greater than or equal to the local density of its nearest neighbor in the training set. The following acceptance function is used:

$$f_{NN^{tr}}(z) = I\left(\frac{\|z - NN^{tr}(z)\|}{\|NN^{tr}(z) - NN^{tr}(NN^{tr}(z))\|}\right)$$

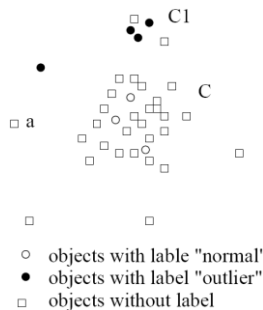
which presents that the distance from object  $z$  to its nearest neighbor in the training set  $NN^{tr}(z)$  is compared to the distance from its nearest neighbor  $NN^{tr}(z)$  to its nearest neighbor.



70

## Classification-Based Method II: Semi-Supervised Learning


- **Semi-supervised learning:** Combining classification-based and clustering-based methods
- Method
  - Using a clustering-based approach, find a large cluster,  $C$ , and a small cluster,  $C_1$
  - Since some objects in  $C$  carry the label "normal", treat all objects in  $C$  as normal
  - Use the one-class model of this cluster to identify normal objects in outlier detection
  - Since some objects in cluster  $C_1$  carry the label "outlier", declare all objects in  $C_1$  as outliers
  - Any object that does not fall into the model for  $C$  (such as  $a$ ) is considered an outlier as well
- Comments on classification-based outlier detection methods
  - Strength: Outlier detection is fast
  - Bottleneck: Quality heavily depends on the availability and quality of the training set, but often difficult to obtain representative and high-quality training data



71

71

## Chapter 12. Outlier Analysis

- Outlier and Outlier Analysis
- Outlier Detection Methods
- Statistical Approaches
- Proximity-Base Approaches
- Clustering-Base Approaches
- Classification Approaches
- Mining Contextual and Collective Outliers 
- Outlier Detection in High Dimensional Data
- Summary

72

72

### Mining Contextual Outliers I: Transform into Conventional Outlier Detection

- If **the contexts can be clearly identified**, transform it to conventional outlier detection
  1. Identify the context of the object using the contextual attributes
  2. Calculate the outlier score for the object in the context using a conventional outlier detection method
- Ex. Detect outlier customers in the context of customer groups
  - Contextual attributes: *age group, postal code*
  - Behavioral attributes: *# of trans/yr, annual total trans. amount*
- Steps: (1) locate c's context, (2) compare c with the other customers in the same group, and (3) use a conventional outlier detection method

73

73

## Mining Contextual Outliers I: Transform into Conventional Outlier Detection

- If the context contains very few customers, generalize contexts
  - For instance, for a customer  $c$ , if the corresponding context contains very few or even no other customers, the evaluation of whether  $c$  is an outlier using the exact context is unreliable or even impossible.
  - To overcome this challenge, we can assume that customers of similar age and who live within the same area should have similar normal behavior.
    - Learn a mixture model  $U$  on the contextual attributes, and another mixture model  $V$  of the data on the behavior attributes
    - Learn a mapping  $p(V_i|U_j)$ : the probability that a data object  $o$  belonging to cluster  $U_j$  on the contextual attributes is generated by cluster  $V_i$  on the behavior attributes
- Outlier score:

$$S(o) = \sum_{U_j} p(o \in U_j) \sum_{V_i} p(o \in V_i) p(V_i|U_j)$$

74

74

## Mining Contextual Outliers II: Modeling Normal Behavior with Respect to Contexts

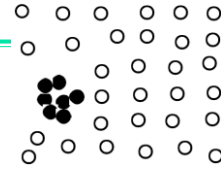
- In some applications, one cannot clearly partition the data into contexts
  - Ex. if a customer suddenly purchased a product that is unrelated to those she recently browsed, it is unclear how many products browsed earlier should be considered as the context
- Model the “normal” behavior with respect to contexts
  - Using a training data set, train a model that predicts the expected behavior attribute values with respect to the contextual attribute values
  - An object is a contextual outlier if its behavior attribute values significantly deviate from the values predicted by the model
- Using a prediction model that links the contexts and behavior, these methods avoid the explicit identification of specific contexts
- **Methods:** A number of classification and prediction techniques can be used to build such models, such as regression, Markov Models, and Finite State Automaton

75

75

## Mining Collective Outliers I: On the Set of “Structured Objects”

- Collective outlier if objects as a group deviate significantly from the entire data
- Need to examine the **structure of the data set**, i.e., the relationships between multiple data objects
- Each of these structures is inherent to its respective type of data
  - **For temporal data** (such as time series and sequences), we explore the structures formed by time, which occur in segments of the time series or subsequences
  - **For spatial data**, explore local areas
  - **For graph and network data**, we explore subgraphs
- Difference from the contextual outlier detection: the structures are often not explicitly defined, and have to be discovered as part of the outlier detection process.
- Collective outlier detection methods: two categories
  - Reduce the problem to conventional outlier detection
    - **Identify structure units**, treat each structure unit (e.g., subsequence, time series segment, local area, or subgraph) as a data object, and extract features
    - Then outlier detection on the set of “structured objects” constructed as such using the extracted features



76

76

## Mining Collective Outliers II: Direct Modeling of the Expected Behavior of Structure Units

- **Models the expected behavior of structure units directly**
- Ex. 1. Detect collective outliers in online social network of customers
  - Treat each possible subgraph of the network as a structure unit
  - Collective outlier: An **outlier subgraph** in the social network
    - Small subgraphs that are of very low frequency
    - Large subgraphs that are surprisingly frequent
- Ex. 2. Detect collective outliers in temporal sequences
  - Learn a Markov model from the sequences
  - **A subsequence can then be declared as a collective outlier if it significantly deviates from the model**
- Collective outlier detection is subtle due to the challenge of exploring the structures in data
  - The exploration typically uses heuristics, and thus may be application dependent
  - The computational cost is often high due to the sophisticated mining process

77

77

## Chapter 12. Outlier Analysis

---

- Outlier and Outlier Analysis
- Outlier Detection Methods
- Statistical Approaches
- Proximity-Based Approaches
- Clustering-Based Approaches
- Classification Approaches
- Mining Contextual and Collective Outliers
- Outlier Detection in High Dimensional Data 
- Summary

78

78

## Challenges for Outlier Detection in High-Dimensional Data

---

- **Interpretation of outliers**
  - Detecting outliers without saying why they are outliers is not very useful in high-D due to many features (or dimensions) are involved in a high-dimensional data set
  - E.g., which subspaces that manifest the outliers or an assessment regarding the “outlier-ness” of the objects
- **Data sparsity**
  - Data in high-D spaces are often sparse
  - The distance between objects becomes heavily dominated by noise as the dimensionality increases
- **Data subspaces**
  - Adaptive to the subspaces signifying the outliers
  - Capturing the local behavior of data
- **Scalable with respect to dimensionality**
  - # of subspaces increases exponentially

79

79

## Approach I: Extending Conventional Outlier Detection

- Method 1: **Detect outliers in the full space, e.g., HilOut Algorithm**
  - Find distance-based outliers, but use the ranks of distance instead of the absolute distance in outlier detection
  - For each object  $o$ , find its  $k$ -nearest neighbors:  $nn_1(o), \dots, nn_k(o)$
  - The weight of object  $o$ : 
$$w(o) = \sum_{i=1}^k dist(o, nn_i(o))$$
  - All objects are ranked in weight-descending order
  - Top- $l$  objects in weight are output as outliers ( $l$ : user-specified parm)
- Method 2: **Dimensionality reduction**
  - Works only when in lower-dimensionality, normal instances can still be distinguished from outliers
  - PCA: Heuristically, the principal components with low variance are preferred because, on such dimensions, normal objects are likely close to each other and outliers often deviate from the majority

80

80

## Approach II: Finding Outliers in Subspaces

- **Extending conventional outlier detection**: Hard for outlier interpretation
- Find outliers in much lower dimensional subspaces: easy to interpret *why* and *to what extent* the object is an outlier
  - E.g., find outlier customers in certain subspace: *average transaction amount* >> avg. and *purchase frequency* << avg.
- Ex. A grid-based subspace outlier detection method
  - **Project data onto various subspaces** to find an area whose density is much lower than average
  - **Discretize the data into a grid** with  $\phi$  equi-depth (why?) regions

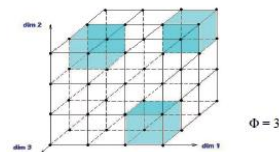
81

81



## Approach II: Finding Outliers in Subspaces

- Search for regions that are significantly sparse
  - Consider a k-d cube: k ranges on k dimensions, with n objects
  - If objects are independently distributed, the expected number of objects falling into a k-dimensional region is  $(1/\phi)^k n = f^k n$ , the standard deviation is  $\sqrt{f^k(1-f^k)n}$
  - The sparsity coefficient of cube C:
 
$$S(C) = \frac{n(C) - f^k n}{\sqrt{f^k(1-f^k)n}}$$
  - If  $S(C) < 0$ , C contains less objects than expected
  - The more negative, the sparser C is and the more likely the objects in C are outliers in the subspace



82

82

## Approach II: Finding Outliers in Subspaces

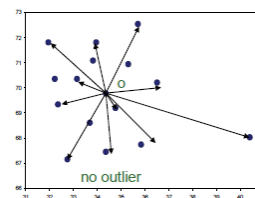
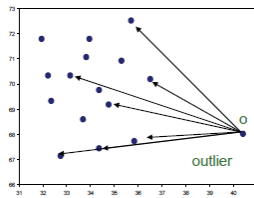
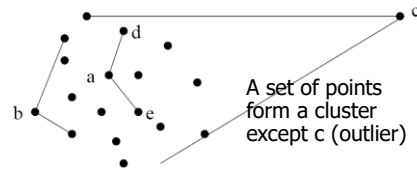
- Algorithm
  - **Find the m grid cells** (projections) with the lowest sparsity coefficients
  - Brute-force algorithm is in  $O(\phi^k)$
  - Evolutionary algorithm (input: m and the dimensionality of the cells)
- Discussion
  - **Very coarse model** (all objects that are in cell with less points than to be expected)
  - **Quality depends on grid resolution and grid position**
  - Implements a **global approach** (key criterion: globally expected number of points within a cell)

83

83

## Approach III: Modeling High-Dimensional Outliers

- Develop new models for high-dimensional outliers directly
- Avoid proximity measures and adopt new heuristics that do not deteriorate in high-dimensional data
- Ex. Angle-based outliers: Kriegel, Schubert, and Zimek [KSZ08]
- Angles are more stable than distances in high dimensional spaces (cf. e.g. the popularity of cosine-based similarity measures for text data)
- Object  $o$  is an outlier if most other objects are located in similar directions
- Object  $o$  is no outlier if many other objects are located in varying directions

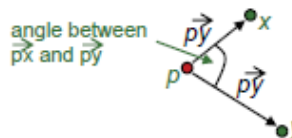


84

84

## Approach III: Modeling High-Dimensional Outliers

- Use the variance of angles for a point to determine outlier
- Combine angles and distance to model outliers
  - Use the distance-weighted angle variance as the outlier score
  - Outliers are at the border of the data distribution
  - Normal points are in the center of the data distribution
- Model
  - Consider for a given point  $p$  the angle between  $px$  and  $py$  for any two  $x, y$  from the database
  - Consider the spectrum of all these angles
  - The broadness of this spectrum is a score for the outlierness of a point



85

85

## Approach III: Modeling High-Dimensional Outliers

- Model (continued)
  - Measure the variance of the angle spectrum
  - Weighted by the corresponding distances (for lower dimensional data sets where angles are less reliable)
  - **Angle-based outlier factor (ABOF):**

$$ABOF(o) = VAR_{x,y \in D, x \neq o, y \neq o} \frac{\langle \overrightarrow{ox}, \overrightarrow{oy} \rangle}{dist(o, x)^2 dist(o, y)^2}$$

- Properties
  - **Small ABOF => outlier**
  - **High ABOF => no outlier**
- Complexity  $O(n^3)$  - Efficient approximation computation methods
- It can be generalized to handle arbitrary types of data

86

86

## Chapter 12. Outlier Analysis

- Outlier and Outlier Analysis
- Outlier Detection Methods
- Statistical Approaches
- Proximity-Base Approaches
- Clustering-Base Approaches
- Classification Approaches
- Mining Contextual and Collective Outliers
- Outlier Detection in High Dimensional Data
- Summary 

87

87

## Summary

- Types of outliers
  - global, contextual & collective outliers
- Outlier detection
  - supervised, semi-supervised, or unsupervised
- Statistical (or model-based) approaches
- Proximity-base approaches
- Clustering-base approaches
- Classification approaches
- Mining contextual and collective outliers
- Outlier detection in high dimensional data

88

88

## References (I)

- B. Abraham and G.E.P. Box. Bayesian analysis of some outlier problems in time series. *Biometrika*, 66:229–248, 1979.
- M. Agyemang, K. Barker, and R. Alhaji. A comprehensive survey of numeric and symbolic outlier mining techniques. *Intell. Data Anal.*, 10:521–538, 2006.
- F. J. Anscombe and I. Guttman. Rejection of outliers. *Technometrics*, 2:123–147, 1960.
- D. Agarwal. Detecting anomalies in cross-classified streams: a bayesian approach. *Knowl. Inf. Syst.*, 11:29–44, 2006.
- F. Angiulli and C. Pizzuti. Outlier mining in large high-dimensional data sets. *TKDE*, 2005.
- C. C. Aggarwal and P. S. Yu. Outlier detection for high dimensional data. *SIGMOD’01*
- R.J. Beckman and R.D. Cook. Outlier...s. *Technometrics*, 25:119–149, 1983.
- I. Ben-Gal. Outlier detection. In *Maimon O. and Rockach L. (eds.) Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers*, Kluwer Academic, 2005.
- M. M. Breunig, H.-P. Kriegel, R. Ng, and J. Sander. LOF: Identifying density-based local outliers. *SIGMOD’00*
- D. Barbar’a, Y. Li, J. Couto, J.-L. Lin, and S. Jajodia. Bootstrapping a data mining intrusion detection system. *SAC’03*
- Z. A. Bakar, R. Mohamad, A. Ahmad, and M. M. Deris. A comparative study for outlier detection techniques in data mining. *IEEE Conf. on Cybernetics and Intelligent Systems*, 2006.
- S. D. Bay and M. Schwabacher. Mining distance-based outliers in near linear time with randomization and a simple pruning rule. *KDD’03*
- D. Barbara, N. Wu, and S. Jajodia. Detecting novel network intrusion using bayesian estimators. *SDM’01*
- V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41:1–58, 2009.
- D. Dasgupta and N.S. Majumdar. Anomaly detection in multidimensional data using negative selection algorithm. In *CEC’02*

89

## References (2)

- E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo. A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data. In *Proc. 2002 Int. Conf. of Data Mining for Security Applications*, 2002.
- E. Eskin. Anomaly detection over noisy data using learned probability distributions. *ICML'00*
- T. Fawcett and F. Provost. Adaptive fraud detection. *Data Mining and Knowledge Discovery*, 1:291–316, 1997.
- V. J. Hodge and J. Austin. A survey of outlier detection methodologies. *Artif. Intell. Rev.*, 22:85–126, 2004.
- D. M. Hawkins. *Identification of Outliers*. Chapman and Hall, London, 1980.
- Z. He, X. Xu, and S. Deng. Discovering cluster-based local outliers. *Pattern Recogn. Lett.*, 24, June, 2003.
- W. Jin, K. H. Tung, and J. Han. Mining top-n local outliers in large databases. *KDD'01*
- W. Jin, A. K. H. Tung, J. Han, and W. Wang. Ranking outliers using symmetric neighborhood relationship. *PAKDD'06*
- E. Knorr and R. Ng. A unified notion of outliers: Properties and computation. *KDD'97*
- E. Knorr and R. Ng. Algorithms for mining distance-based outliers in large datasets. *VLDB'98*
- E. M. Knorr, R. T. Ng, and V. Tucakov. Distance-based outliers: Algorithms and applications. *VLDB J.*, 8:237–253, 2000.
- H.-P. Kriegel, M. Schubert, and A. Zimek. Angle-based outlier detection in high-dimensional data. *KDD'08*
- M. Markou and S. Singh. Novelty detection: A review—part 1: Statistical approaches. *Signal Process.*, 83:2481–2497, 2003.
- M. Markou and S. Singh. Novelty detection: A review—part 2: Neural network based approaches. *Signal Process.*, 83:2499–2521, 2003.
- C. C. Noble and D. J. Cook. Graph-based anomaly detection. *KDD'03*

90

## References (3)

- S. Papadimitriou, H. Kitagawa, P. B. Gibbons, and C. Faloutsos. Loci: Fast outlier detection using the local correlation integral. *ICDE'03*
- A. Patcha and J.-M. Park. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Comput. Netw.*, 51, 2007.
- X. Song, M. Wu, C. Jermaine, and S. Ranka. Conditional anomaly detection. *IEEE Trans. on Knowl. and Data Eng.*, 19, 2007.
- Y. Tao, X. Xiao, and S. Zhou. Mining distance-based outliers from large databases in any metric space. *KDD'06*
- N. Ye and Q. Chen. An anomaly detection technique based on a chi-square statistic for detecting intrusions into information systems. *Quality and Reliability Engineering International*, 17:105–112, 2001.
- B.-K. Yi, N. Sidiropoulos, T. Johnson, H. V. Jagadish, C. Faloutsos, and A. Biliris. Online data mining for co-evolving time sequences. *ICDE'00*

91