UNIT 5

1. Outliers: Introduction, Challenges of outlier detection,
2. Outlier detection methods: Introduction, Supervised and semi supervised methods,
3. Unsupervised methods,
4. Statistical and proximity based methods,
5. Statistical approaches,
6. Statistical data mining,
7. Data Mining and recommender systems,
8. data mining for financial data analysis,
9. Data mining for Intrusion detection

Introduction

- Traditional Data Mining Categories
  - Majority of Objects
    - Dependency detection
    - Class identification
    - Class description
  - Exceptions
    - Exception/outlier detection

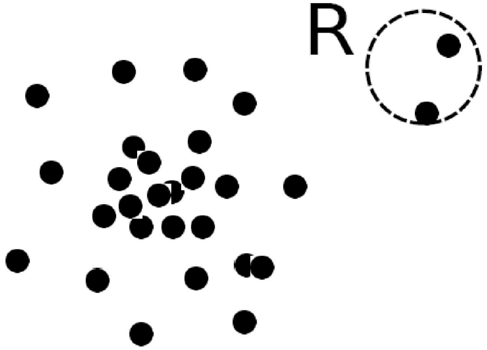
Motivation for Outlier Analysis

- Fraud Detection (Credit card, telecommunications, criminal activity in e-Commerce)
- Customized Marketing (high/low income buying habits)
- Medical Treatments (unusual responses to various drugs)
- Analysis of performance statistics (professional athletes)
- Weather Prediction
- Financial Applications (loan approval, stock tracking)

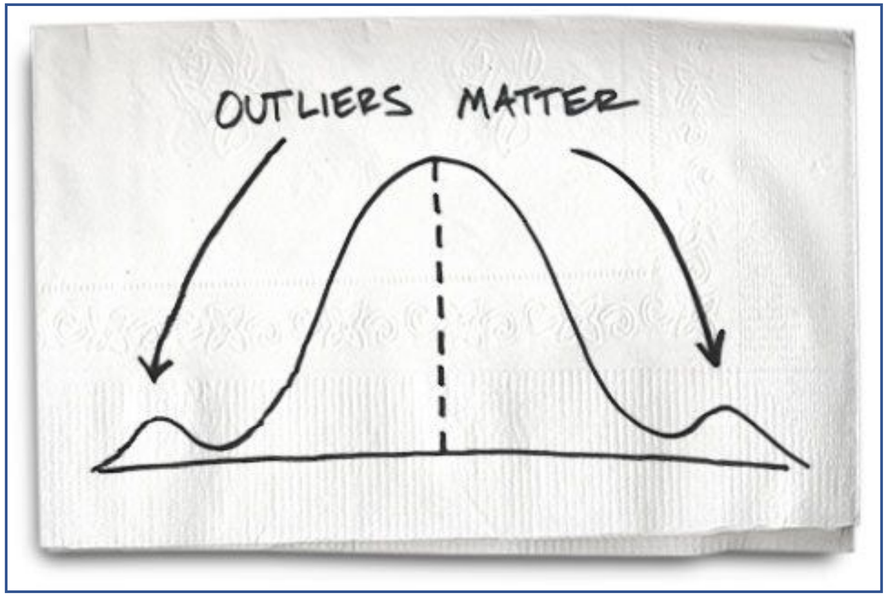
“One persons noise could be another person’s signal.”

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 Jordon, 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: It violates 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



When trying to detect outliers in a dataset it is very important to keep in mind the context and try to answer the question: ***"¿Why do I want to detect outliers?" The meaning of your findings will be dictated by the context.***

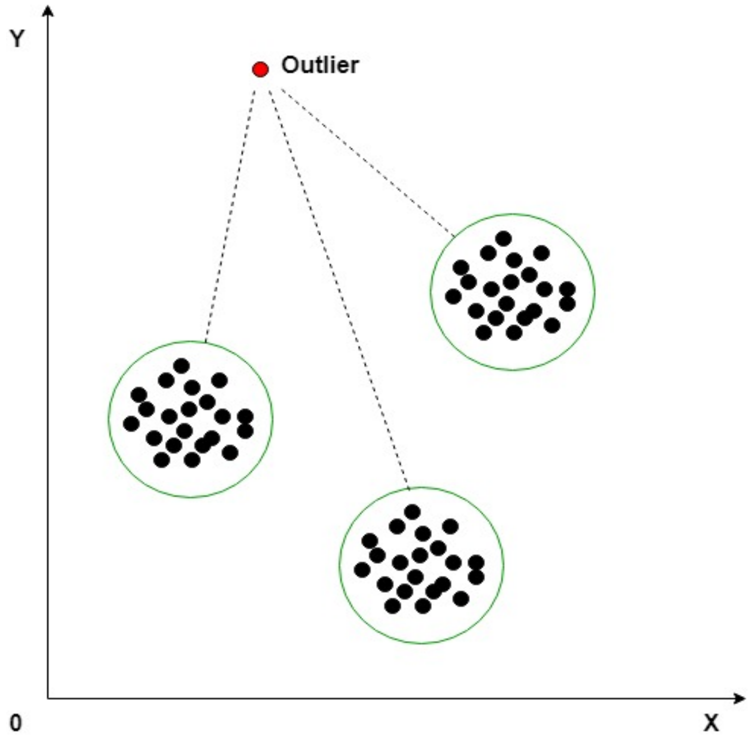


Causes of Outliers

- Poor data quality / contamination
- Low quality measurements, malfunctioning equipment, manual error
- Correct but exceptional data

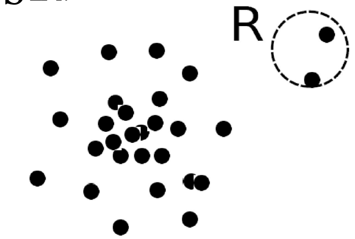
Why outlier analysis?

Most data mining methods discard outliers noise or exceptions, however, in some applications such as fraud detection, the rare events can be more interesting than the more regularly occurring one and hence, the outlier analysis becomes important in such case.



Types of Outliers

- 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
  - Issue: Find an appropriate measurement of deviation

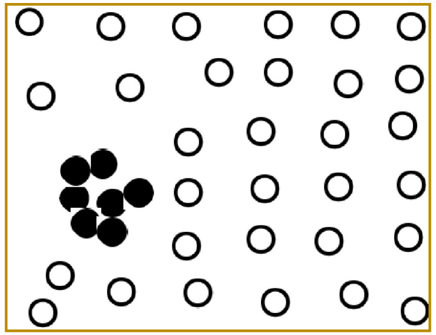


Global Outlier

- **Contextual outlier** (or *conditional outlier*)
  - Object is O_c if it deviates significantly based on a selected context
  - Ex. 80° F in Urbana: 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?

What Are Outliers? (Cont'd)

- **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



Collective Outlier

- 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

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
- 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

Why A Special Technique to Identify Outliers?

- Why not just modify clustering or other algorithms to detect outliers?
  - Performance considerations
  - Subjective to the clustering algorithm and clustering parameters
  - Only certain attributes may have outlier properties, no need to disqualify the entire tuple
  - Contamination may occur by “column”, not by row

OUTLIER DETECTION METHODS

Two ways to categorize outlier detection methods:

1. Based on whether user-*labeled* examples of outliers can be obtained:
   - Supervised
   - semi-supervised
   - unsupervised methods

2. Based on *assumptions about normal data and outliers*:
   - Statistical
   - proximity-based
   - clustering-based methods

Method 1

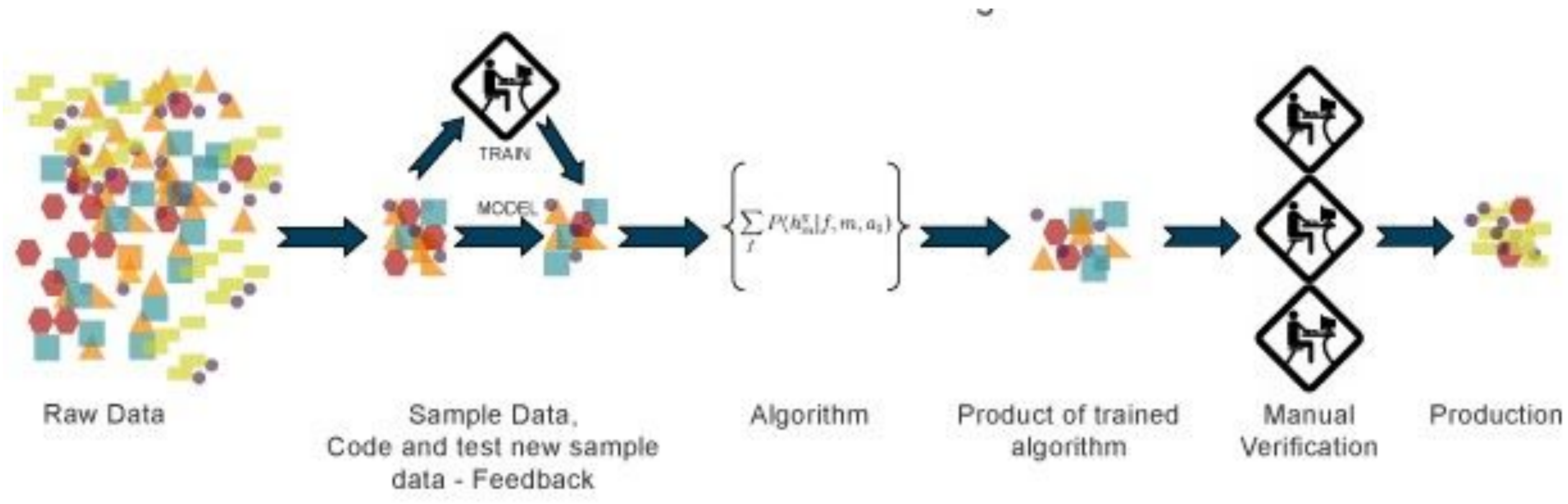
Outlier detection based on whether user-*labeled* examples of outliers can be obtained:

Supervised, semi-supervised & unsupervised methods

Outlier Detection I: 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)

Supervised Methods (Cont’d)



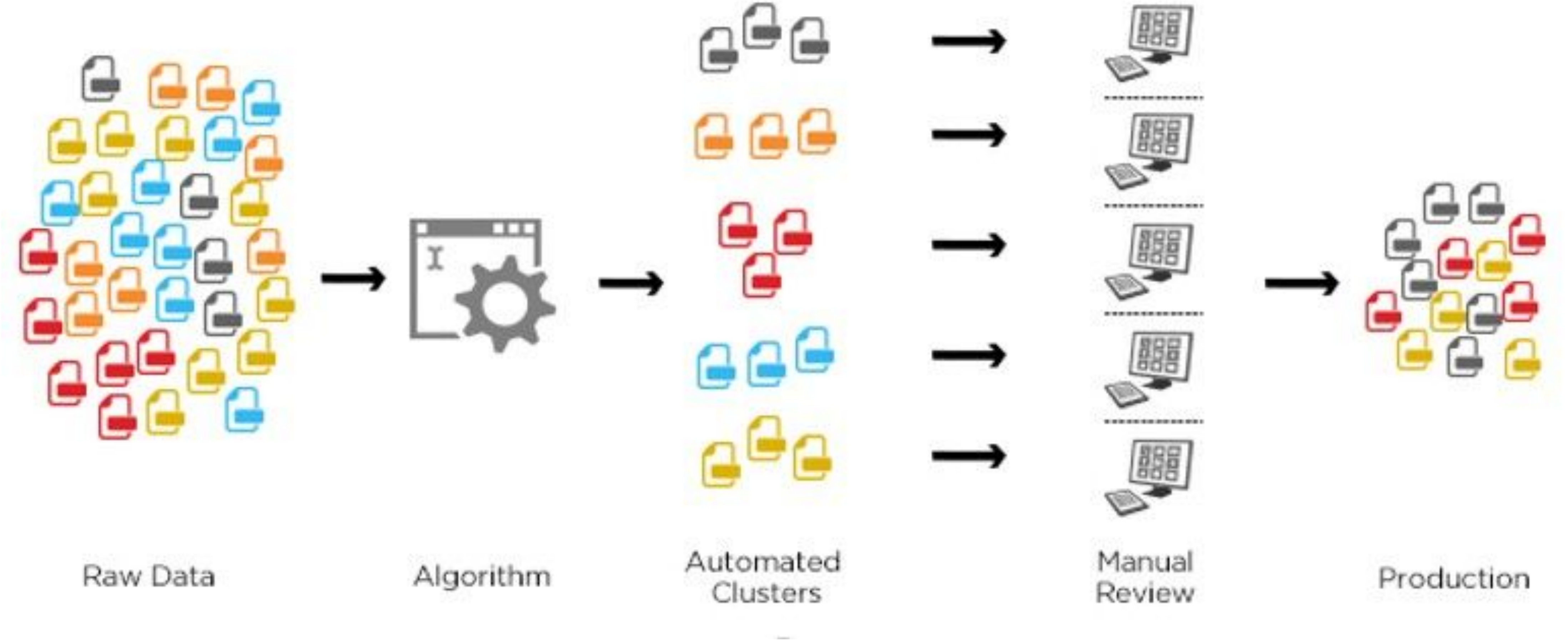
Outlier Detection II: Unsupervised Methods

- 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
- Ex. In some intrusion or virus detection, normal activities are diverse
- Many clustering methods can be adapted for unsupervised methods.

Unsupervised Methods (Cont’d)

- 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

Unsupervised Methods (Cont’d)



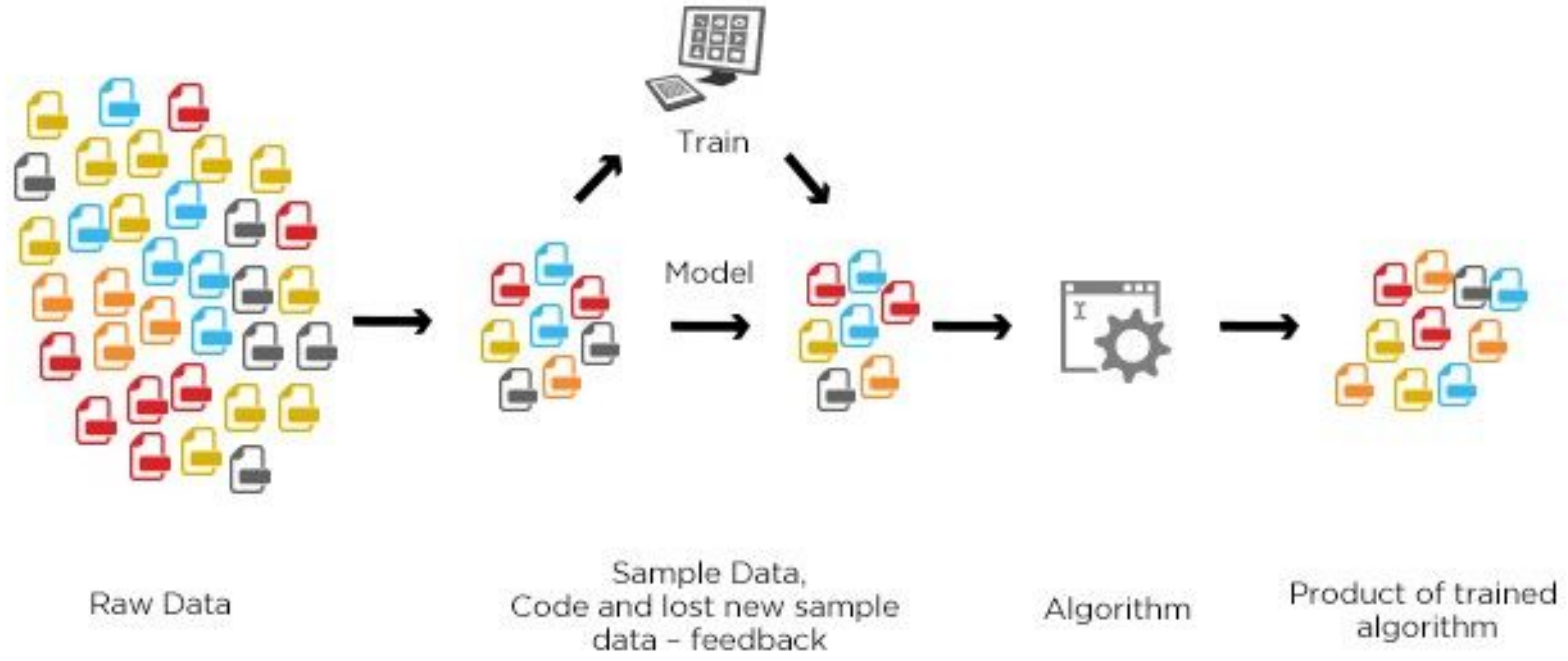
Outlier Detection III: Semi-Supervised Methods

- Labels could be on outliers only, normal objects only, or both
- Semi supervised outlier detection: Regarded as application of semi supervised learning
- This can be done in 2 ways
  - □ If some labeled normal objects are available
  - □ If only some labeled outliers are available

Semi-Supervised Methods (Cont’d)

- 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 many 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

Semi-Supervised Methods (Cont'd)



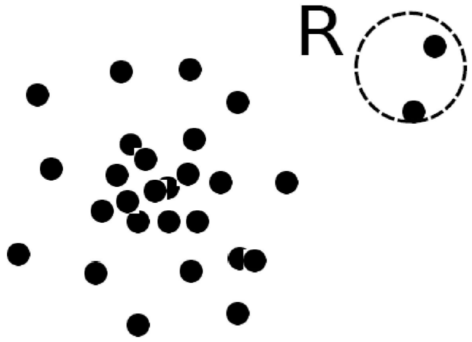
Method 2

Outlier detection based on *assumptions about normal data and outliers*

(Statistical, proximity-based & clustering-based methods)

Outlier Detection (1): Statistical Methods

- Statistical methods (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 side figure):

  Step 1: Use Gausion distribution model

  Step 2: Consider the object y in region R.

  Step 3: Estimate the probability of y fits the Gausion distribution ($g_D(y)$)

  Step 4: If $g_D(y)$ is low, y is an outlier.

  First use Gaussian distribution to model the normal data

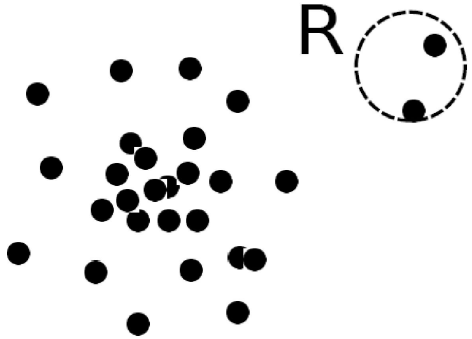
  - For each object y in region R, estimate $g_D(y)$, the probability of y fits the Gaussian distribution
  - If $g_D(y)$ is very low, y is unlikely generated by the Gaussian model, thus an outlier

Statistical Methods (Cont’d)

- 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

Outlier Detection (2): 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 is **significantly deviates** from the proximity of most of the other objects in the same data set

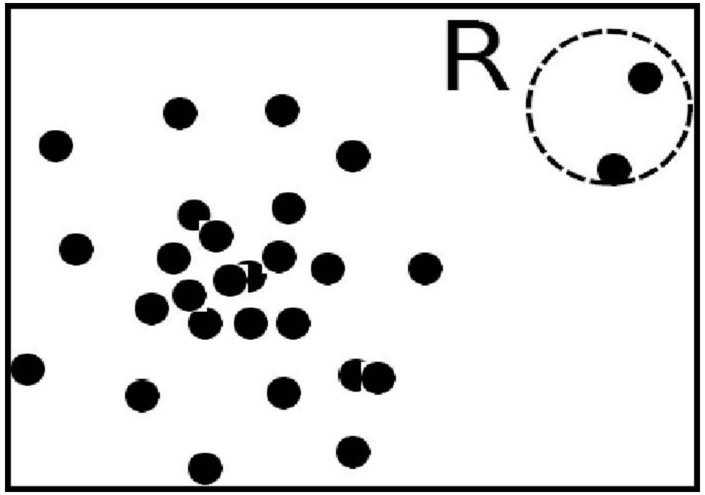


- Example (right figure): Model the proximity of an object using its 3 nearest neighbors
  - Objects in region R are substantially different from other objects in the data set.
  - Thus the objects in R are outliers
- The effectiveness of proximity-based methods highly relies on the proximity measure.
- 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

Outlier Detection (3): Clustering-Based Methods

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

Example: 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 adaption of a clustering method for outlier detection can be costly and does not scale up well for large data sets.

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 θ
  - The probability density function of the parametric distribution $f(x, \vartheta)$ 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

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

- 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 μ and σ

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

  - Taking derivatives with respect to μ and σ^2, we derive the following maximum likelihood estimates

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

  - For the above data with n = 10, we have $\hat{\mu} = 28.61$ $\hat{\sigma} = \sqrt{2.29} = 1.51$
  - Then (24 – 28.61) /1.51 = – 3.04 < –3, 24 is an outlier since $\mu \pm 3\sigma$ region contains 99.7% data

Parametric Methods I: The Grubb’s Test

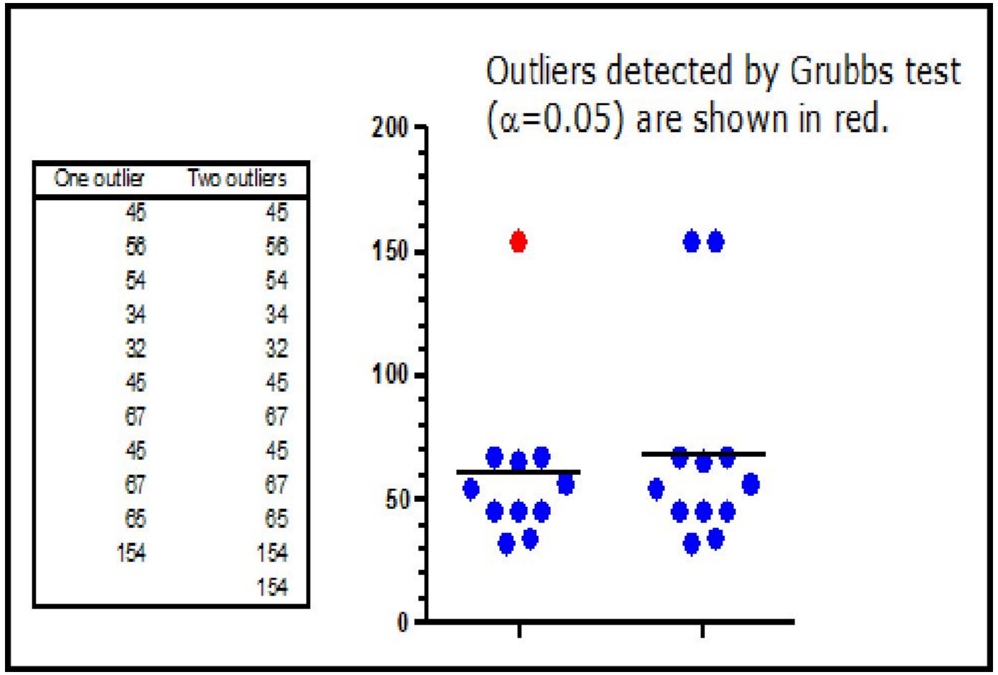
- Univariate outlier detection: The Grubb's test (maximum normed residual test) ─ another statistical method under normal distribution
  - For each object x in a data set, compute its z-score: x is an outlier if

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

where $t^2_{\alpha/(2N),N-2}$ is the value taken by a t-distribution at a significance level of α/(2N), and N is the # of objects in the data set

The Grubb’s Test (Cont’d)

- Detect outliers in univariate data
- Assume data comes from normal distribution
- Detects one outlier at a time, remove the outlier, and repeat
- H0: There is no outlier in data
- HA: There is at least one outlier

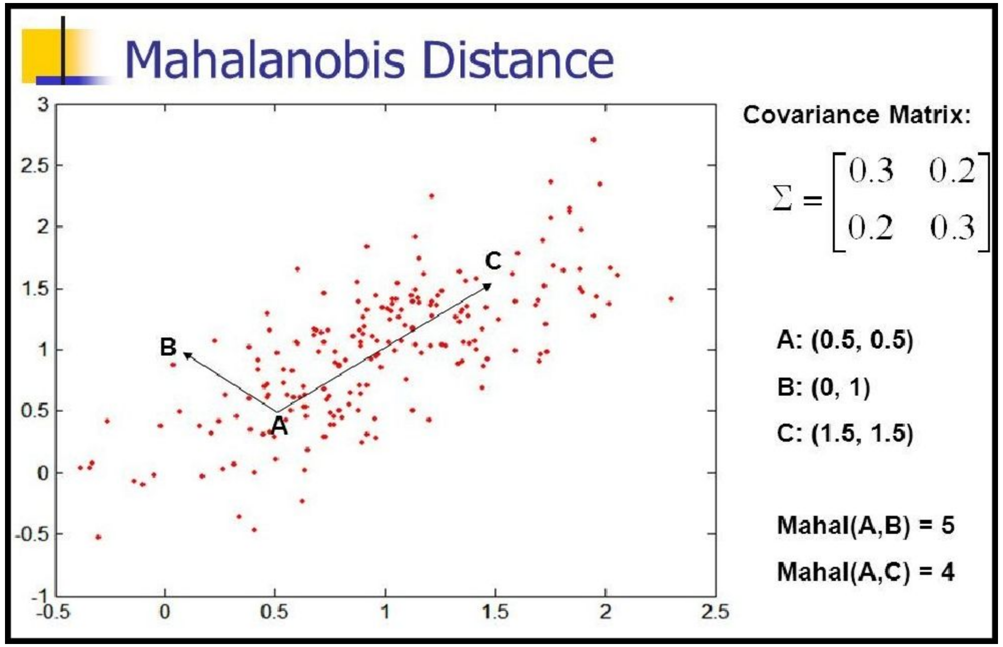


One outlier	Two outliers
45	45
56	56
54	54
34	34
32	32
45	45
67	67
45	45
67	67
65	65
154	154
	154

Outliers detected by Grubbs test (α=0.05) are shown in red.

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. Compute Mahalaobis distance
    - Method 2. Use χ^2–statistic



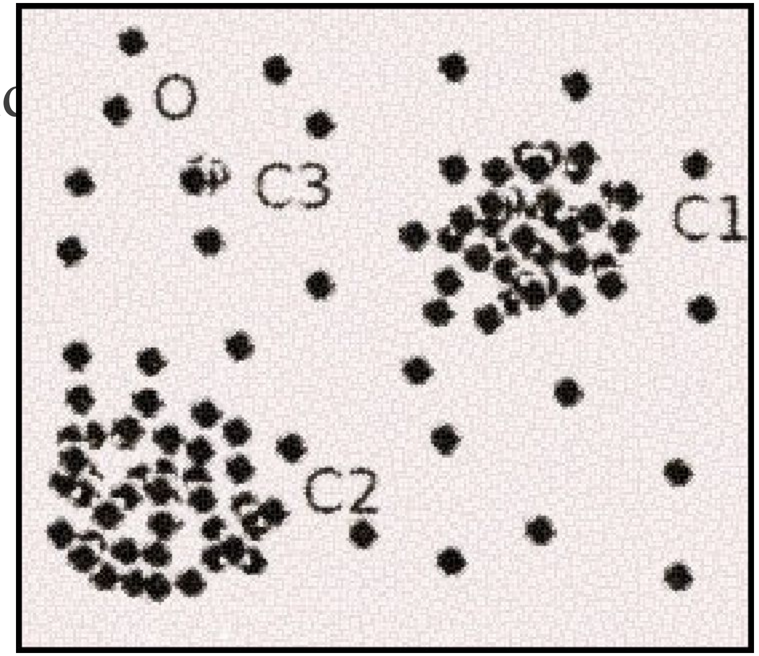
Detection of Multivariate Outliers (Cont’d)

- Method 1. Compute Mahalaobis distance
  - Let ō be the mean vector for a multivariate data set. Mahalaobis distance for an object o to ō is MDist(o, ō) = $(o - \bar{o})^T S^{-1}(o - \bar{o})$ where S is the covariance matrix
  - Use the Grubb's test on this measure to detect outliers
- Method 2. Use χ^2 –statistic:
  - where E_i is the mean of the *i*-dimension among all objects, and n is the dimensionality
  - If χ^2 –statistic is large, then object o_i is an outlier

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

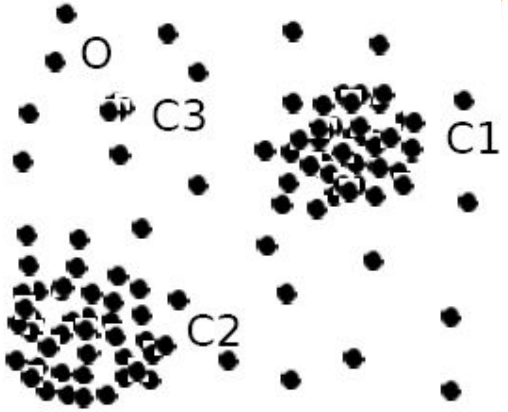
Parametric Methods (III) – Using Mixture of Parametric Distributions

- Assuming data generated by a normal distribution coul be sometimes overly simplified.
- Example: The objects between the two clusters cannot be captured as outliers since they are close to the estimated mean.
- To overcome this problem, assume the normal.
- data is generated by two normal distributions.



Using Mixture of Parametric Distributions (Cont'd)

- Assuming data generated by a normal distribution could be sometimes overly simplified
- Example (right figure): The objects between the two clusters cannot be captured as outliers since they are close to the estimated mean



- To overcome this problem, assume the normal data is generated by two normal distributions. For any object o in the data set, the probability that o is generated by the mixture of the two distributions is given by

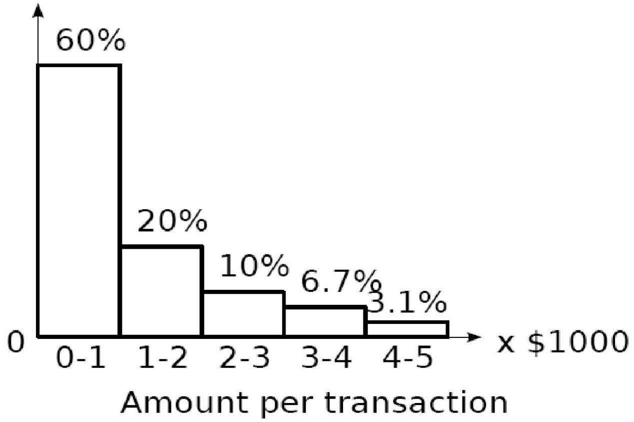
$$Pr(o|\Theta_1, \Theta_2) = f_{\Theta_1}(o) + f_{\Theta_2}(o)$$

where $f_{\theta 1}$ and $f_{\theta 2}$ are the probability density functions of θ_1 and θ_2

- Then use EM algorithm to learn the parameters μ_1, σ_1, μ_2, σ_2 from data
- An object o is an outlier if it does not belong to any cluster

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.

Proximity-Based Approaches

- 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

Proximity-Based Approaches (I) – Distance based Approaches

- General Idea
  - Judge a point based on the distance(s) to its neighbors
  - Several variants proposed

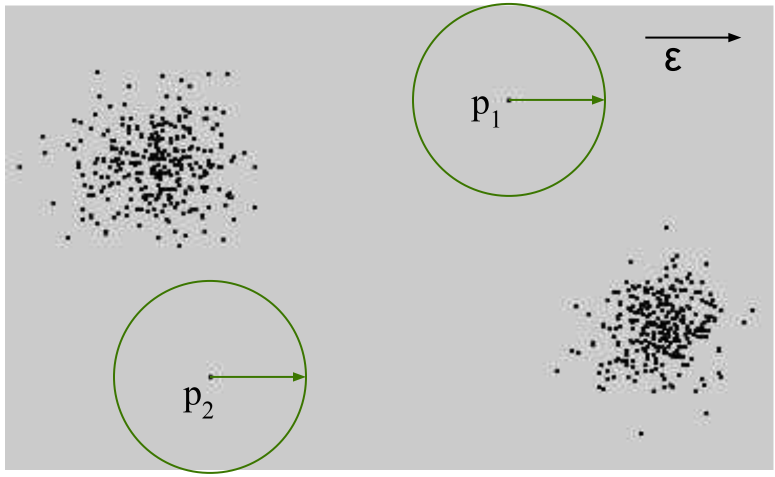
- Basic Assumption
  - Normal data objects have a dense neighborhood
  - Outliers are far apart from their neighbors, i.e., have a less dense neighborhood

Distance based Approaches (Cont’d)

- DB(ε,π)-Outliers
  - Basic model [Knorr and Ng 1997]
    - Given a radius ε and a percentage π
    - A point *p* is considered an outlier if at most π percent of all other points have a distance to *p* less than ε

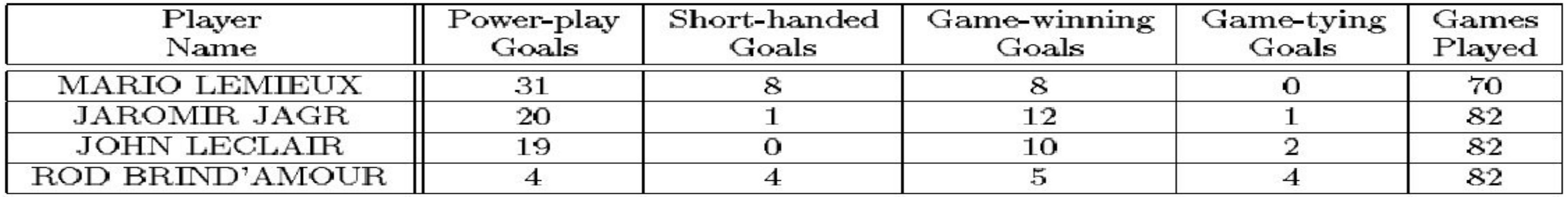
$$OutlierSet(\varepsilon,\pi) = \{p \mid \frac{Card(\boxed{\{q \in DB \mid dist(p,q) < \varepsilon\}})}{Card(DB)} \leq \pi\}$$

range-query with radius ε



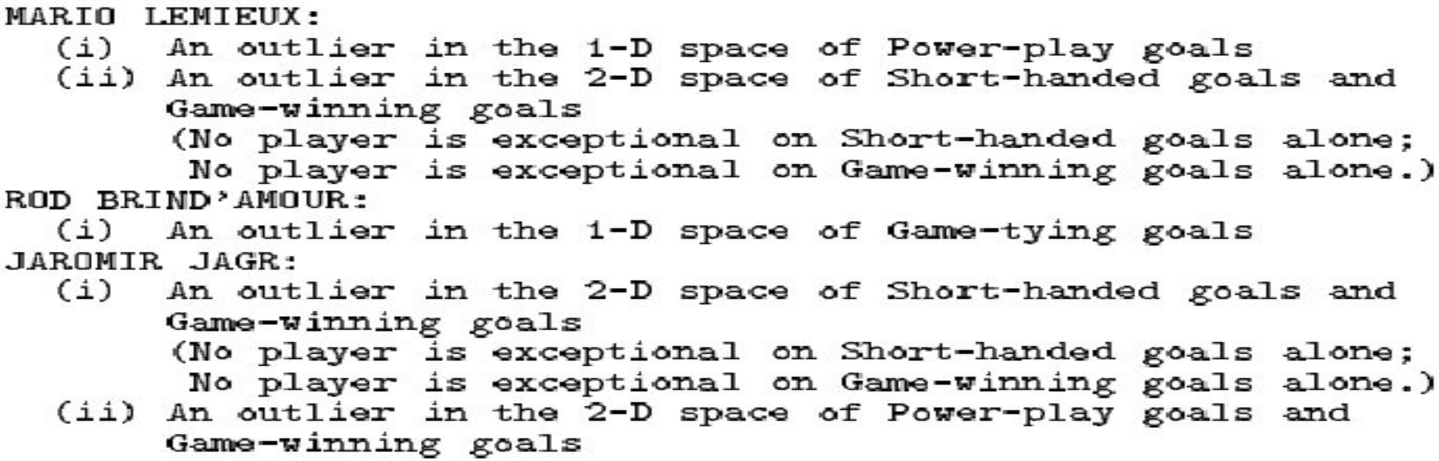
Distance based Approaches (Cont’d)

- Deriving intentional knowledge [Knorr and Ng 1999]
  - Relies on the DB(ε,π)-outlier model
  - Find the minimal subset(s) of attributes that explains the “outlierness” of a point, i.e., in which the point is still an outlier
  - Example
    - Identified outliers



Player Name	Power-play Goals	Short-handed Goals	Game-winning Goals	Game-tying Goals	Games Played
MARIO LEMIEUX	31	8	8	0	70
JAROMIR JAGR	20	1	12	1	82
JOHN LECLAIR	19	0	10	2	82
ROD BRIND’AMOUR	4	4	5	4	82

    - Derived intensional knowledge (sketch)



```
MARIO LEMIEUX:
   (i)   An outlier in the 1-D space of Power-play goals
   (ii)  An outlier in the 2-D space of Short-handed goals and
         Game-winning goals
         (No player is exceptional on Short-handed goals alone;
          No player is exceptional on Game-winning goals alone.)
ROD BRIND'AMOUR:
   (i)   An outlier in the 1-D space of Game-tying goals
JAROMIR JAGR:
   (i)   An outlier in the 2-D space of Short-handed goals and
         Game-winning goals
         (No player is exceptional on Short-handed goals alone;
          No player is exceptional on Game-winning goals alone.)
   (ii)  An outlier in the 2-D space of Power-play goals and
         Game-winning goals
```

Distance based Approaches (Cont’d)

- Nested-loop based [Knorr and Ng 1998]
  - Divide buffer in two parts.
  - Use second part to scan/compare all points with the points from the first part.

- Outlier scoring based on *k*NN distances
  - Take the *k*NN distance of a point as its outlier score [Ramaswamy et al 2000]
  - Aggregate the distances of a point to all its 1NN, 2NN, …, *k*NN as an outlier score [Angiulli and Pizzuti 2002]

Distance based Approaches (Cont’d)

- **Index-based [Knorr and Ng 1998]**

– Compute distance range join using spatial index structure.

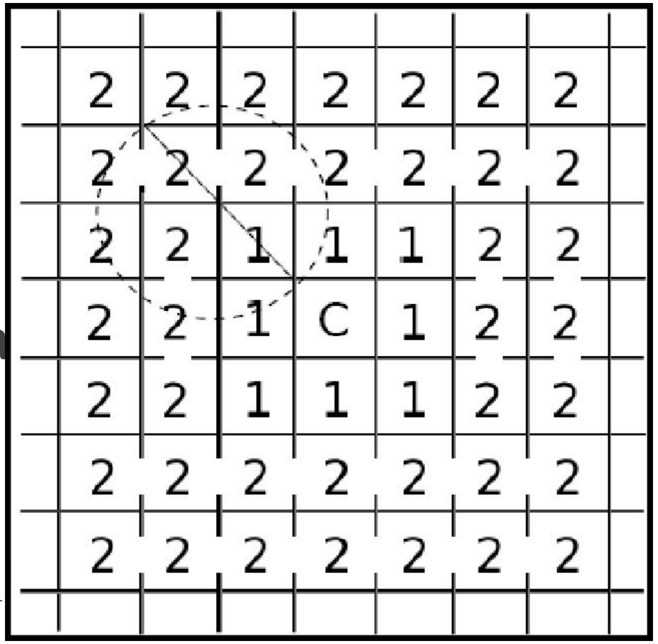
–Exclude point from further consideration if its ε-neighborhood contains more than *Card*(*DB*) **.** π points.

- **Grid-based [Knorr and Ng 1998]**

–Build grid such that any two points from the same grid cell ha

distance of at most ε to each other.

– Points need only compared with points from neighboring cel

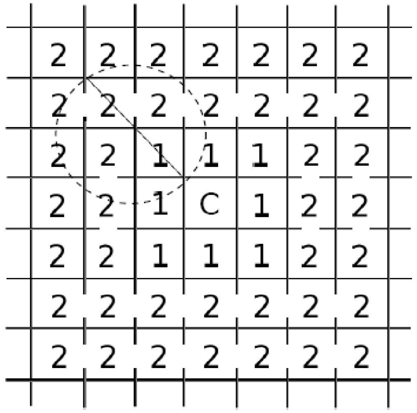


2	2	2	2	2	2	2
2	2	2	2	2	2	2
2	2	1	1	1	2	2
2	2	1	C	1	2	2
2	2	1	1	1	2	2
2	2	2	2	2	2	2
2	2	2	2	2	2	2

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:
  - For any possible point x in cell C and any possible point y in a level-1 cell, dist(x,y) $\leq$ r
  - For any possible point x in cell C and any point y such that dist(x,y) $\geq$ r, y is in a level-2 cell



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

- 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)

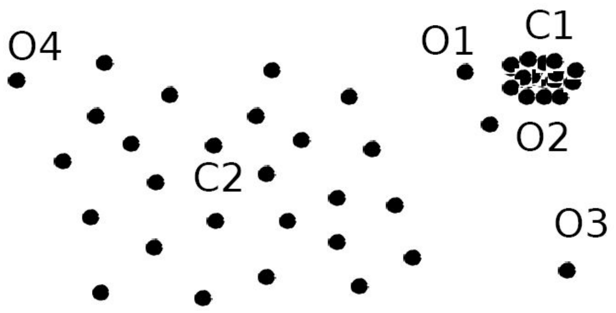
Density-based Approaches

- General idea
  - Compare the density around a point with the density around its local neighbors
  - The relative density of a point compared to its neighbors is computed as an outlier score
  - Approaches essentially differ in how to estimate density

- Basic assumption
  - The density around a normal data object is similar to the density around its neighbors
  - The density around an outlier is considerably different to the density around its neighbors

Density-Based Outlier Detection

- Local outliers: Outliers comparing to their local neighborhoods, instead of the global data distribution
- In Fig., o_1 and o2 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 o_1 and o_2 are outlier (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 outliers
- *k-distance* of an object o, $\text{dist}_k(o)$: distance between o and its k-th NN
- *k-distance neighborhood* of o, $N_k(o) = \{o' | o' \text{ in } D, \text{dist}(o, o') \leq \text{dist}_k(o)\}$
  - $N_k(o)$ could be bigger than k since multiple objects may have identical distance to o

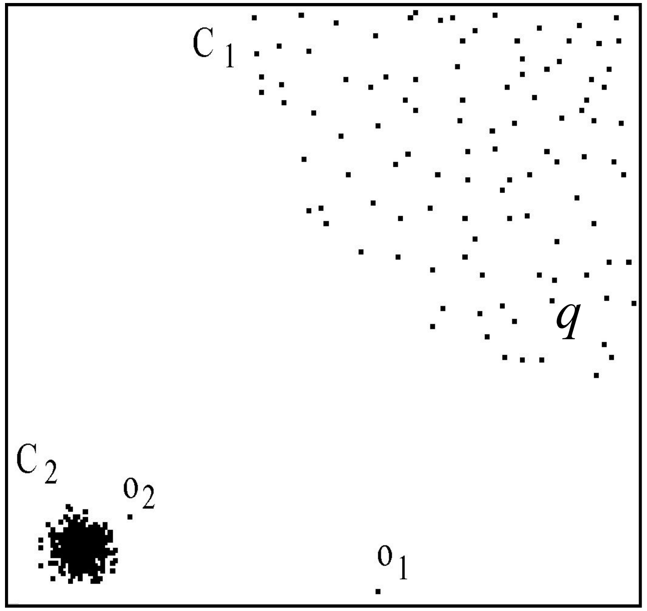
- Distance-based outlier detection models have problems with different densities .
- Compare the neighborhood of points from areas of different densities
- Compare the density around a point with the density around its local neighbors
- The relative density of a point compared to its neighbors is computed as an outlier score
- Approaches also differ in how to estimate density.

Basic assumptions

- The density around a normal data object is similar to the density around its neighbors.
- The density around an outlier is considerably different to the density

Density-based Approaches

- Local Outlier Factor (LOF) [Breunig et al. 1999], [Breunig et al. 2000]
  - Motivation:
    - Distance-based outlier detection models have problems with different densities
    - How to compare the neighborhood of points from areas of different densities?
    - Example
      - DB(ε,π)-outlier model
        - Parameters ε and π cannot be chosen so that o_2 is an outlier but none of the points in cluster C_1 (e.g. *q*) is an outlier
      - Outliers based on kNN-distance
        - kNN-distances of objects in C_1 (e.g. *q*) are larger than the kNN-distance of o_2



  - Solution: consider relative density

Local Outlier Factor: LOF

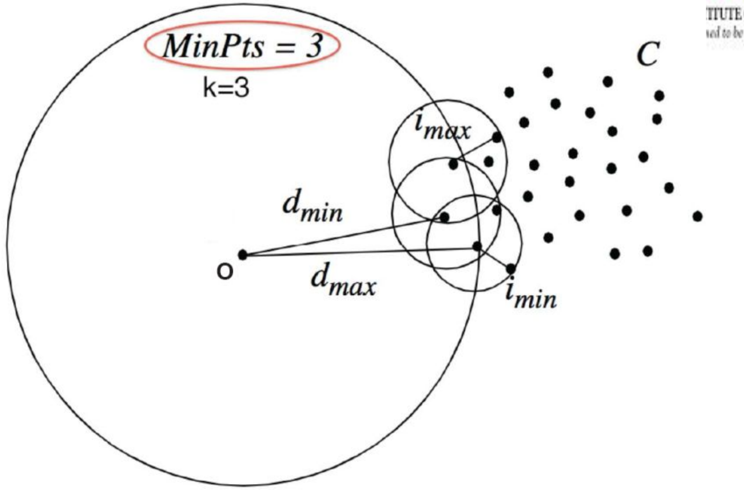
- Reachability distance from *o'* to *o*:

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

  - where k is a user-specified parameter

- Local reachability density of *o*:

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



- 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

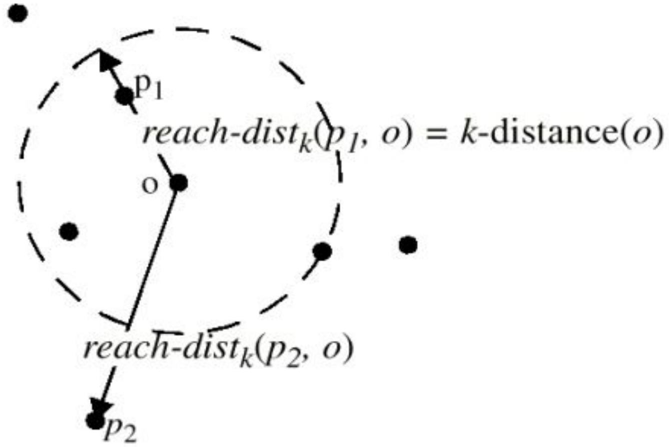
$$LOF_k(o) = \frac{\sum_{o' \in N_k(o)} \frac{lrd_k(o')}{lrd_k(o)}}{\|N_k(o)\|} = \sum_{o' \in N_k(o)} lrd_k(o') \cdot \sum_{o' \in N_k(o)} reachdist_k(o' \leftarrow o)$$

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

Density-based Approaches

- Model
  - Reachability distance $reach-dist_k(p,o) = \max\{k-\text{distance}(o), dist(p,o)\}$
    - Introduces a smoothing factor
  - Local reachability distance (lrd) of point *p*
    - Inverse of the average reach-dists of the *k*NNs of *p*

$$lrd_k(p) = 1 / \left(\frac{\sum_{o \in kNN(p)} reach-dist_k(p,o)}{Card(kNN(p))} \right)$$

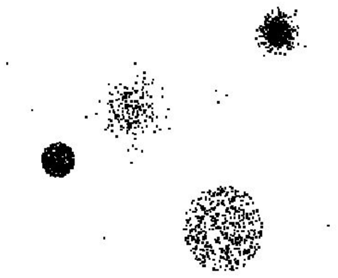


  - Local outlier factor (LOF) of point *p*
    - Average ratio of lrds of neighbors of *p* and lrd of *p*

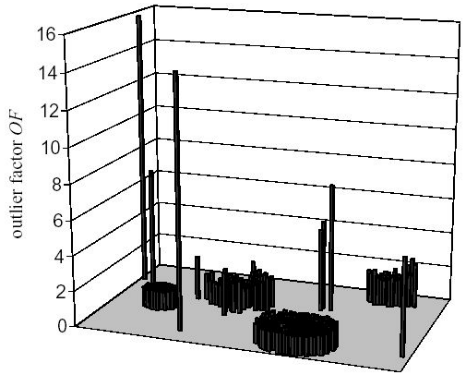
$$LOF_k(p) = \frac{\sum_{o \in kNN(p)} \frac{lrd_k(o)}{lrd_k(p)}}{Card(kNN(p))}$$

Density-based Approaches

- Properties
  - LOF ≈ 1: point is in a cluster (region with homogeneous density around the point and its neighbors)
  - LOF >> 1: point is an outlier



Data set



LOFs (*MinPts* = 40)

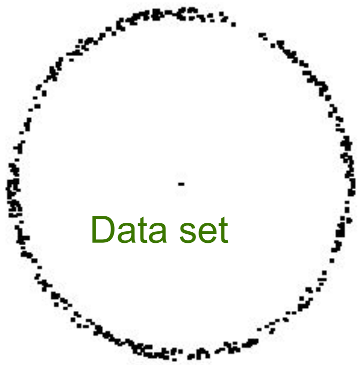
- Discussion
  - Choice of *k* (*MinPts*) specifies the reference set
  - Originally implements a local approach (resolution depends on the user's choice for *k*)
  - Outputs a scoring (assigns an LOF value to each point)

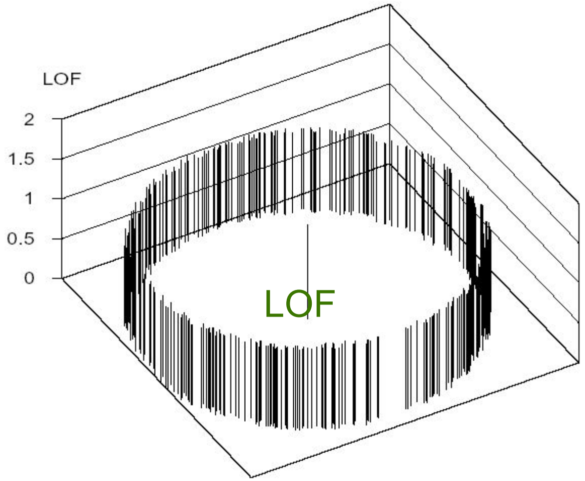
Density-based Approaches

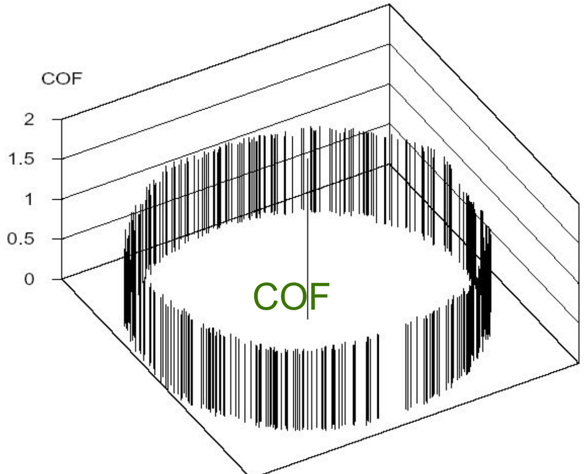
- Variants of LOF
  - Mining top-*n* local outliers [Jin et al. 2001]
    - Idea:
      - Usually, a user is only interested in the top-*n* outliers
      - Do not compute the LOF for all data objects => save runtime
    - Method
      - Compress data points into micro clusters using the CFs of BIRCH [Zhang et al. 1996]
      - Derive upper and lower bounds of the reachability distances, lrd-values, and LOF-values for points within a micro clusters
      - Compute upper and lower bounds of LOF values for micro clusters and sort results w.r.t. ascending lower bound
      - Prune micro clusters that cannot accommodate points among the top-*n* outliers (*n* highest LOF values)
      - Iteratively refine remaining micro clusters and prune points accordingly

Density-based Approaches

- Variants of LOF (cont.)
  - Connectivity-based outlier factor (COF) [Tang et al. 2002]
    - Motivation
      - In regions of low density, it may be hard to detect outliers
      - Choose a low value for *k* is often not appropriate
    - Solution
      - Treat “low density” and “isolation” differently
    - Example

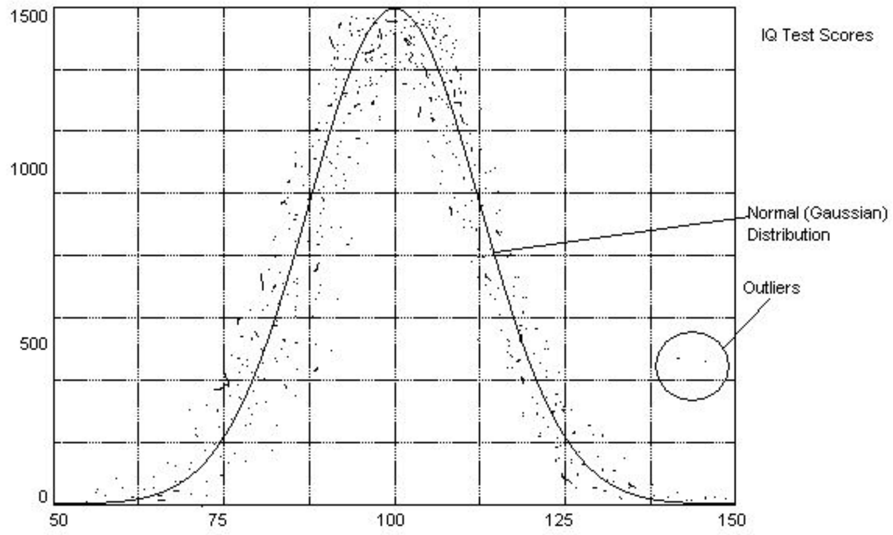






Statistical-Based Outlier Detection (Distribution-based)

- Assumptions:
  - Knowledge of data (distribution, mean, variance)
- Statistical discordancy test
  - Data is assumed to be part of a working hypothesis (working hypothesis)
  - Each data object in the dataset is compared to the working hypothesis and is either accepted in the working hypothesis or rejected as discordant into an alternative hypothesis (outliers)



Working Hypothesis: $H : o_i \in F$, where $i = 1,2,\ldots,n.$

Discordancy Test: is o_i in F within standard deviation $= 15$

Alternative Hypothesis:

-Inherent Distribution: $\overline{H} : o_i \in G$, where $i = 1,2,\ldots,n.$

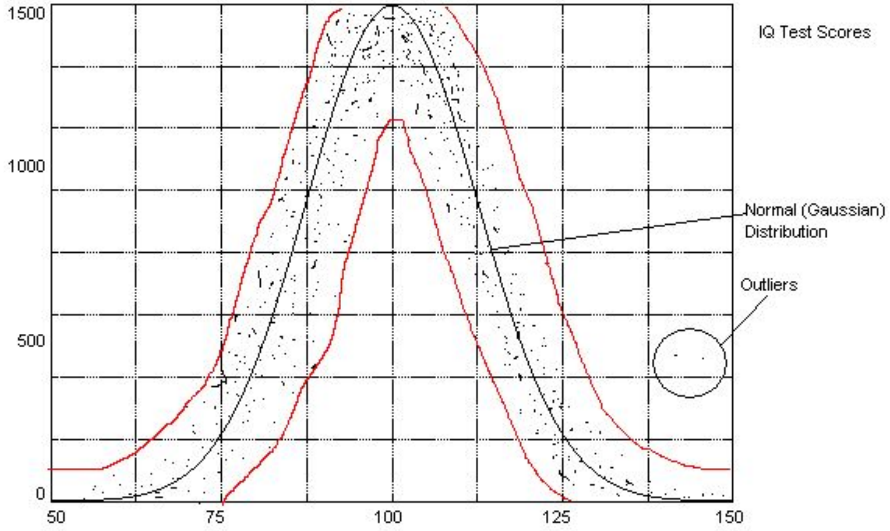
-Mixture Distribution: $\overline{H} : o_i \in (1-\lambda)F + \lambda G$, where $i = 1,2,\ldots,n.$

-Slippage Distibution: $\overline{H} : o_i \in (1-\lambda)F + \lambda F'$, where $i = 1,2,\ldots,n.$

Statistical-Based Outlier Detection (Distribution-based)

- Assumptions:
  - Knowledge of data (distribution, mean, variance)

- Statistical discordancy test
  - Data is assumed to be part of a working hypothesis (working hypothesis)
  - Each data object in the dataset is compared to the working hypothesis and is either accepted in the working hypothesis or rejected as discordant into an alternative hypothesis (outliers)



Working Hypothesis: $H: o_i \in F$, where $i = 1,2,\ldots,n.$

Discordancy Test: is o_i in F within standard deviation $= 15$

Alternative Hypothesis:

-Inherent Distribution: $\overline{H}: o_i \in G$, where $i = 1,2,\ldots,n.$

-Mixture Distribution: $\overline{H}: o_i \in (1-\lambda)F + \lambda G$, where $i = 1,2,\ldots,n.$

-Slippage Distibution: $\overline{H}: o_i \in (1-\lambda)F + \lambda F'$, where $i = 1,2,\ldots,n.$

Statistical-Based Outlier detection (Depth-based)

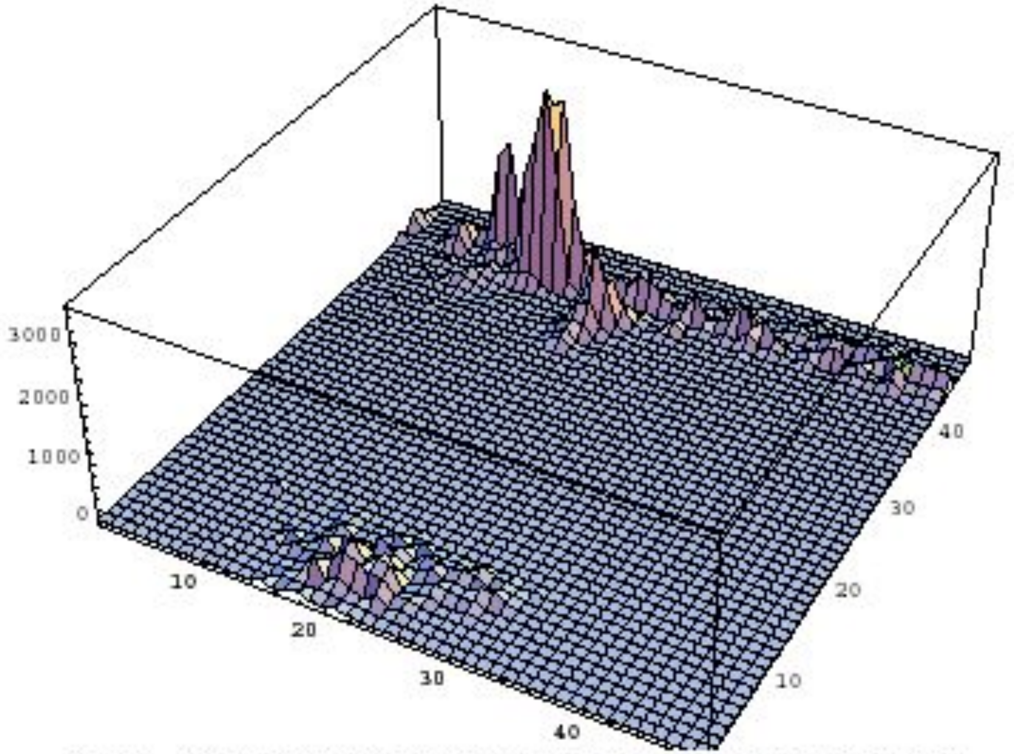
- Data is organized into layers according to some definition of depth
- Shallow layers are more

  likely to contain

  outliers than deep

  layers
- Can efficiently handle

  computation for $k < 4$



YU99 - FindOut: Finding Outliers in Very Large Datasets

Statistical-Based Outlier Detection

- Strengths
  - Most outlier research has been done in this area, many data distributions are known
- Weakness
  - Almost all of the statistical models are univariate (only handle one attribute) and those that are multivariate only efficiently handle k<4
  - All models assume the distribution is known –this is not always the case
  - Outlier detection is completely subjective to the distribution used

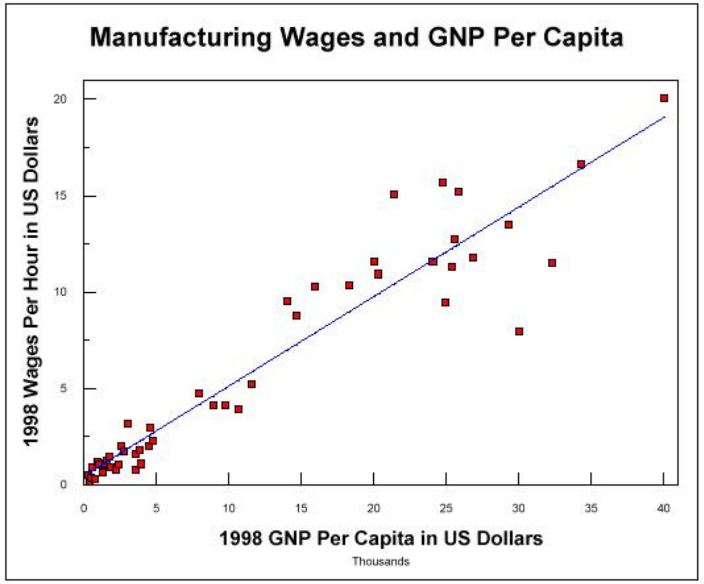
Major Statistical Data Mining Methods

- Regression
- Generalized Linear Model
- Analysis of Variance
- Mixed-Effect Models
- Factor Analysis
- Discriminant Analysis
- Survival Analysis

Statistical Data Mining (1)

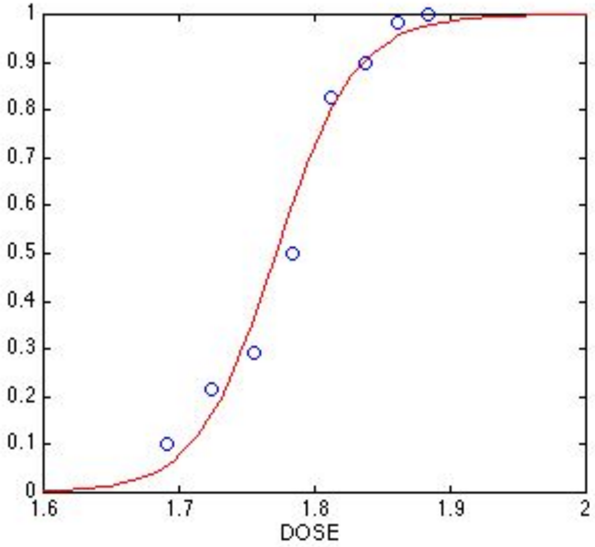
- There are many well-established statistical techniques for data analysis, particularly for numeric data
  - applied extensively to data from scientific experiments and data from economics and the social sciences

- **Regression**
  - predict the value of a response (dependent) variable from one or more predictor (independent) variables where the variables are numeric
  - forms of regression: linear, multiple, weighted, polynomial, nonparametric, and robust



Scientific and Statistical Data Mining (2)

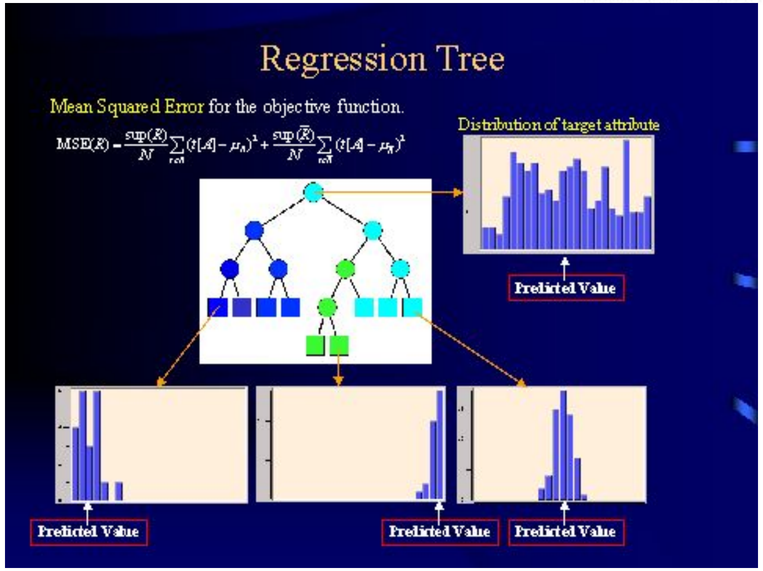
- **Generalized linear models**
  - allow a categorical response variable (or some transformation of it) to be related to a set of predictor variables
  - similar to the modeling of a numeric response variable using linear regression
  - include logistic regression and Poisson regression



- **Mixed-effect models**
  - For analyzing grouped data, i.e. data that can be classified according to one or more grouping variables
  - Typically describe relationships between a response variable and some covariates in data grouped according to one or more factors

Scientific and Statistical Data Mining (3)

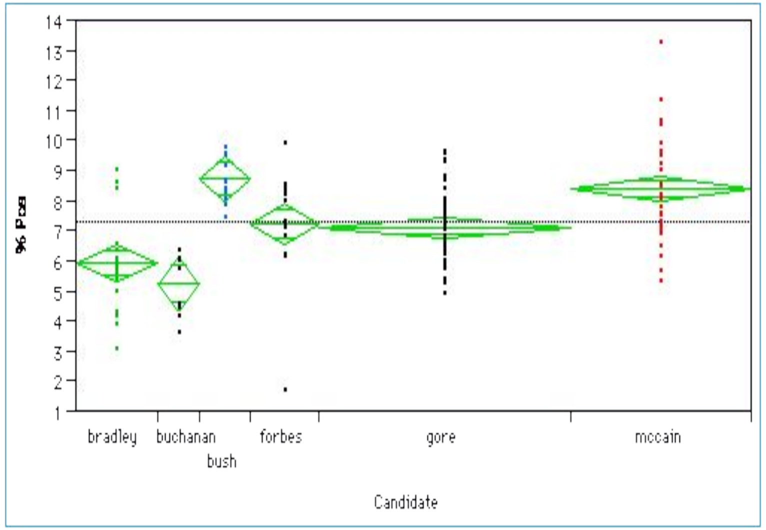
- **Regression trees**
  - Binary trees used for classification and prediction
  - Similar to decision trees:Tests are performed at the internal nodes
  - In a regression tree the mean of the objective attribute is computed and used as the predicted value
- **Analysis of variance**
  - Analyze experimental data for two or more populations described by a numeric response variable and one or more categorical variables (factors)



Regression Tree

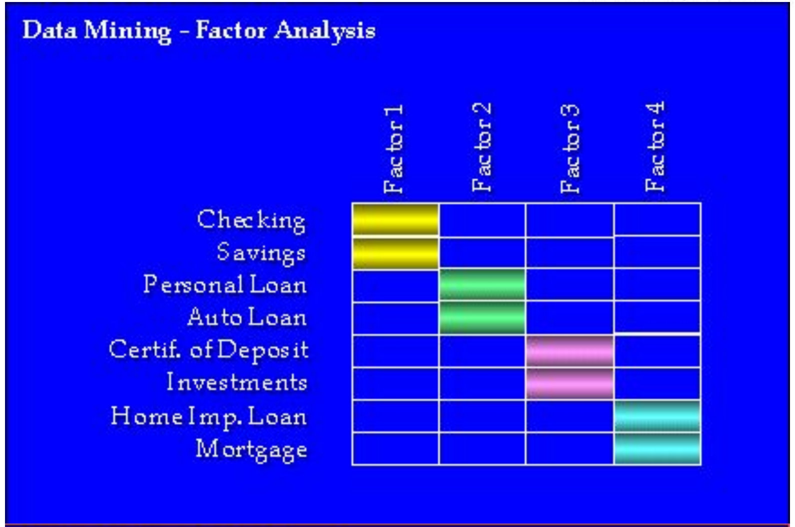
Mean Squared Error for the objective function.

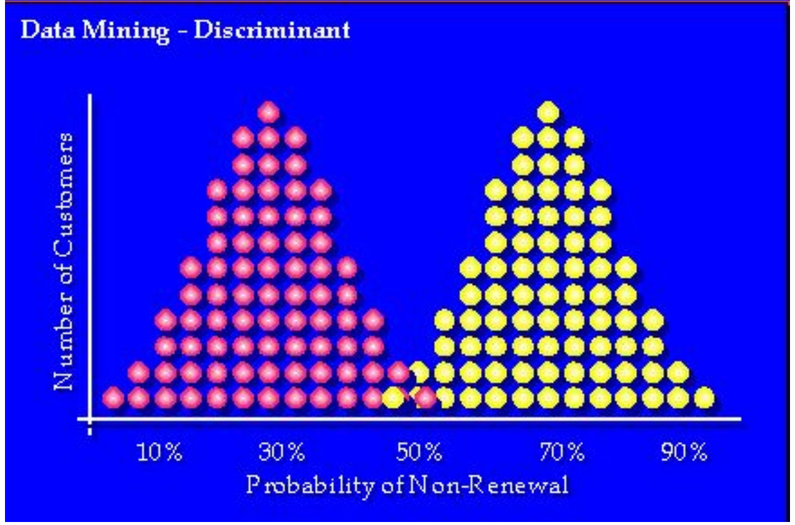
$$MSE(R) = \frac{sup(R)}{N}\sum_{t\in R}(t[A]-\mu_R)^2 + \frac{sup(\bar{R})}{N}\sum_{t\in \bar{R}}(t[A]-\mu_{\bar{R}})^2$$



Statistical Data Mining (4)

- **Factor analysis**
  - determine which variables are combined to generate a given factor
  - e.g., for many psychiatric data, one can indirectly measure other quantities (such as test scores) that reflect the factor of interest
- **Discriminant analysis**
  - predict a categorical response variable, commonly used in social science
  - Attempts to determine several discriminant functions (linear combinations of the independent variables) that discriminate among the groups defined by the response variable



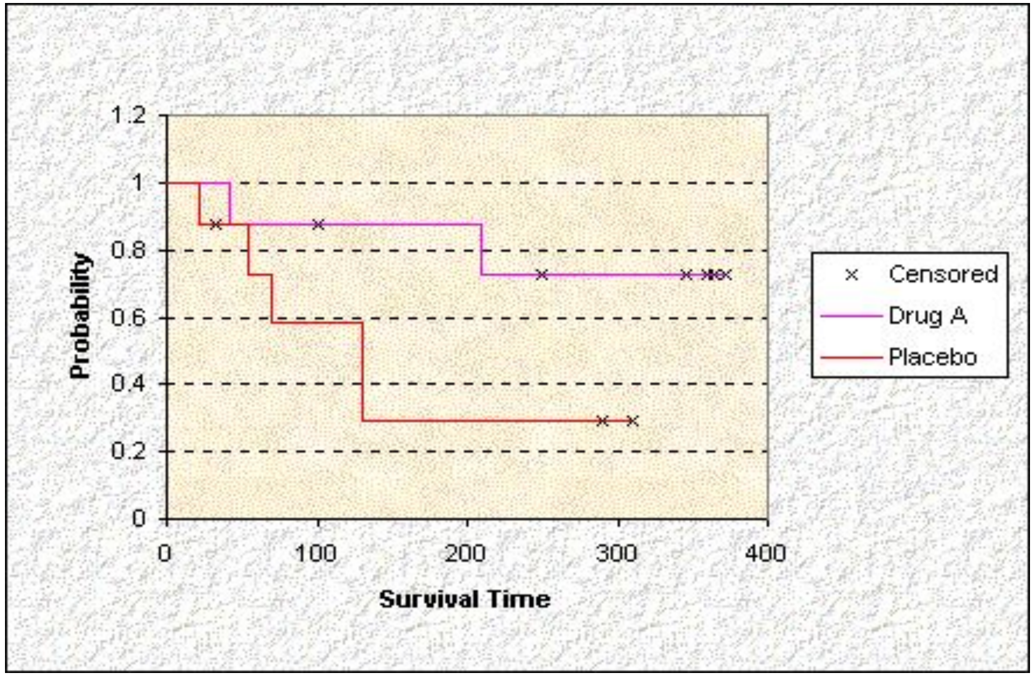


www.spss.com/datamine/factor.htm

Statistical Data Mining (5)

- **Time series**: many methods such as autoregression, ARIMA (Autoregressive integrated moving-average modeling), long memory time-series modeling
- **Quality control:** displays group summary charts

- **Survival analysis**
  - ❑ Predicts the probability that a patient undergoing a medical treatment would survive at least to time *t* (life span prediction)



Data Mining Applications

- Data mining: A young discipline with broad and diverse applications
  - There still exists a nontrivial gap between generic data mining methods and effective and scalable data mining tools for domain-specific applications
- Some application domains (briefly discussed here)
  - Data Mining for Financial data analysis
  - Data Mining for Retail and Telecommunication Industries
  - Data Mining in Science and Engineering
  - Data Mining for Intrusion Detection and Prevention
  - Data Mining and Recommender Systems

Data Mining and Recommender Systems

- Recommender systems: Personalization, making product recommendations that are likely to be of interest to a user
- Approaches: Content-based, collaborative, or their hybrid
  - Content-based: Recommends items that are similar to items the user preferred or queried in the past
  - Collaborative filtering: Consider a user's social environment, opinions of other customers who have similar tastes or preferences
- Data mining and recommender systems
  - Users C × items S: extract from known to unknown ratings to predict user-item combinations
  - Memory-based method often uses k-nearest neighbor approach
  - Model-based method uses a collection of ratings to learn a model (e.g., probabilistic models, clustering, Bayesian networks, etc.)
  - Hybrid approaches integrate both to improve performance (e.g., using ensemble)

Data Mining for Financial Data Analysis (I)

- Financial data collected in banks and financial institutions are often relatively complete, reliable, and of high quality
- Design and construction of data warehouses for multidimensional data analysis and data mining
    - View the debt and revenue changes by month, by region, by sector, and by other factors
    - Access statistical information such as max, min, total, average, trend, etc.
- Loan payment prediction/consumer credit policy analysis
    - feature selection and attribute relevance ranking
    - Loan payment performance
    - Consumer credit rating

Data Mining for Financial Data Analysis (II)

- Classification and clustering of customers for targeted marketing
    - multidimensional segmentation by nearest-neighbor, classification, decision trees, etc. to identify customer groups or associate a new customer to an appropriate customer group
- Detection of money laundering and other financial crimes
    - integration of from multiple DBs (e.g., bank transactions, federal/state crime history DBs)
    - Tools: data visualization, linkage analysis, classification, clustering tools, outlier analysis, and sequential pattern analysis tools (find unusual access sequences)

Data Mining for Retail & Telcomm. Industries (I)

- Retail industry: huge amounts of data on sales, customer shopping history, e-commerce, etc.
- Applications of retail data mining
  - Identify customer buying behaviors
  - Discover customer shopping patterns and trends
  - Improve the quality of customer service
  - Achieve better customer retention and satisfaction
  - Enhance goods consumption ratios
  - Design more effective goods transportation and distribution policies
- Telcomm. and many other industries: Share many similar goals and expectations of retail data mining

Data Mining Practice for Retail Industry

- Design and construction of data warehouses
- Multidimensional analysis of sales, customers, products, time, and region
- Analysis of the effectiveness of sales campaigns
- Customer retention: Analysis of customer loyalty
  - Use customer loyalty card information to register sequences of purchases of particular customers
  - Use sequential pattern mining to investigate changes in customer consumption or loyalty
  - Suggest adjustments on the pricing and variety of goods
- Product recommendation and cross-reference of items
- Fraudulent analysis and the identification of usual patterns
- Use of visualization tools in data analysis

Data Mining in Science and Engineering

- Data warehouses and data preprocessing
  - Resolving inconsistencies or incompatible data collected in diverse environments and different periods (e.g. eco-system studies)
- Mining complex data types
  - Spatiotemporal, biological, diverse semantics and relationships
- Graph-based and network-based mining
  - Links, relationships, data flow, etc.
- Visualization tools and domain-specific knowledge
- Other issues
  - Data mining in social sciences and social studies: text and social media
  - Data mining in computer science: monitoring systems, software bugs, network intrusion

Data Mining for Intrusion Detection and Prevention

- Majority of intrusion detection and prevention systems use
    - Signature-based detection: use signatures, attack patterns that are preconfigured and predetermined by domain experts
    - Anomaly-based detection: build profiles (models of normal behavior) and detect those that are substantially deviate from the profiles
- What data mining can help
    - New data mining algorithms for intrusion detection
    - Association, correlation, and discriminative pattern analysis help select and build discriminative classifiers
    - Analysis of stream data: outlier detection, clustering, model shifting
    - Distributed data mining
    - Visualization and querying tools

Thank you!