

Skript zur Vorlesung
Knowledge Discovery in Databases
im Sommersemester 2011

Kapitel 6: Outlier Detection

Vorlesung+Übungen:
PD Dr. Matthias Schubert, Dr. Arthur Zimek

Skript © 2011 Peer Kröger, Arthur Zimek
basiert auf Tutorial von Hans-Peter Kriegel, Peer Kröger, Arthur Zimek: Outlier Detection Techniques
(PAKDD-09, Bangkok, Thailand)

[http://www.dbs.ifi.lmu.de/cms/Knowledge_Discovery_in_Databases_I_\(KDD_I\)](http://www.dbs.ifi.lmu.de/cms/Knowledge_Discovery_in_Databases_I_(KDD_I))

Übersicht

- 6.1 Einleitung
- 6.2 Statistical Tests
- 6.3 Depth-based Approaches
- 6.4 Deviation-based Approaches
- 6.5 Distance-based Approaches
- 6.6 Density-based Approaches
- 6.7 High-dimensional Approaches
- 6.8 Summary
- Literatur

Was ist ein Outlier?

Definition nach Hawkins [Hawkins 1980]:

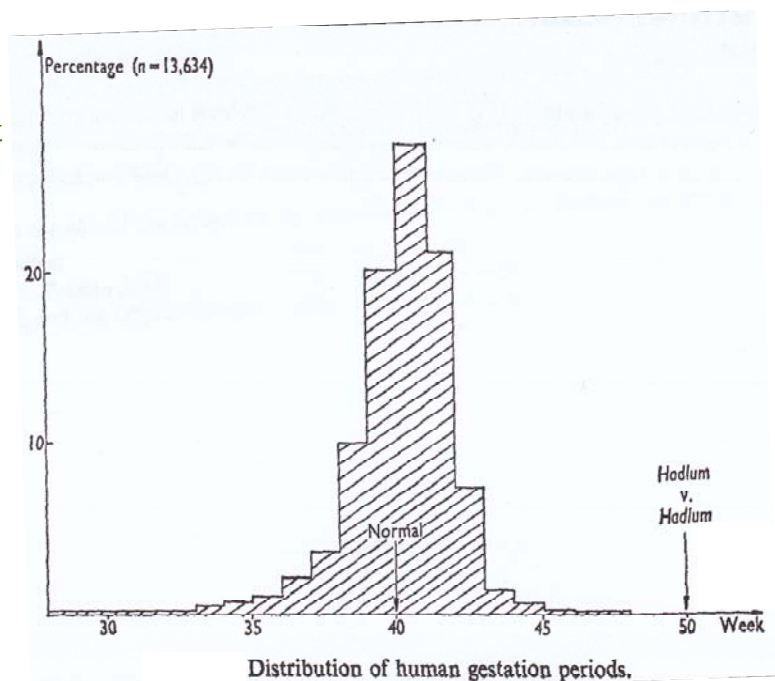
“Ein Outlier ist eine *Beobachtung*, die sich von den anderen *Beobachtungen* so deutlich unterscheidet, daß man denken könnte, sie sei von einem anderen Mechanismus generiert worden.”

Was meint “Mechanismus”?

- Intuition aus der Statistik: “erzeugender Mechanismus” ist ein (statistischer) Prozess.
- Abnormale Daten (outlier) zeigen eine verdächtig geringe Wahrscheinlichkeit, aus diesem Prozess zu stammen.

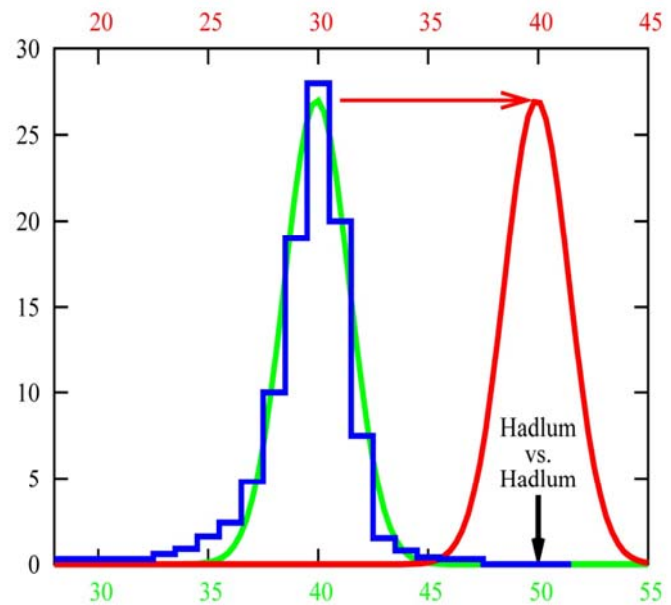
Beispiel: Hadlum vs. Hadlum (1949) [Barnett 1978]

- Geburt eines Kindes von Mrs. Hadlum 349 Tage nachdem Mr. Hadlum zum Militärdienst abwesend war.
- Durchschnittliche Dauer einer menschlichen Schwangerschaft ist 280 Tage (40 Wochen)
- Ist eine Schwangerschaftsdauer von 349 Tagen ein Outlier?



Beispiel: Hadlum vs. Hadlum (1949) [Barnett 1978]

- Blau: statistische Beobachtungsbasis (13634 erhobene Schwangerschaften)
- Grün: angenommener zugrundeliegender Gauss-Prozess
 - sehr geringe Wahrscheinlichkeit, dass die Geburt aus diesem Prozess stammt
- Rot: Annahme von Mr. Hadlum (ein anderer Gauss-Prozess, in dem die Schwangerschaft später beginnt, ist für die Geburt verantwortlich)
 - unter dieser Annahme hat die Schwangerschaftsdauer einen Durchschnittswert und höchst-mögliche Wahrscheinlichkeit



Anwendungsgebiete:

- Betrugsentdeckung
 - Kaufverhalten mit einer Kreditkarte ändert sich, wenn die Karte gestohlen wurde
 - Ungewöhnliche Kauf-Muster können Kreditkarten-Mißbrauch anzeigen
- Medizin
 - Ungewöhnliche Symptome oder Test-Ergebnisse können mögliche gesundheitliche Probleme eines Patienten anzeigen
 - Ob ein bestimmtes Testergebnis ungewöhnlich ist, kann von anderen Eigenschaften des Patienten abhängen (z.B. Geschlecht, Alter, Gewicht, ...)
- Öffentliches Gesundheitswesen
 - Auftauchen einer bestimmten Krankheit (z.B. Tetanus) verstreut über verschiedene Krankenhäuser einer Stadt zeigt Probleme mit dem zugehörigen Impfprogramm an
 - Ob das Auftreten der Krankheit unnormal ist hängt von verschiedenen Aspekten ab, z.B. Häufigkeit, räumliche Korrelation etc.

Anwendungsgebiete:

- Sport Statistiken
 - In vielen Sportarten werden diverse Parameter aufgezeichnet, um die Leistung eines Spielers zu bewerten
 - Außergewöhnliche (in positivem wie negativem Sinne) Spieler können durch ungewöhnliche Werte bestimmt werden
 - Manchmal ist nur eine Teilmenge der Parameter ungewöhnlich
- Entdecken von Messfehlern
 - Daten aus Sensoren (z.B. in einem wissenschaftlichen Experiment) können Meßfehler enthalten
 - Ungewöhnliche Werte können ein Hinweis auf Meßfehler sein
 - Solche Meßfehler aus den Daten zu entfernen, kann wichtig sein für erfolgreiche Datenanalyse und Data Mining

„One person’s noise could be another person’s signal.“

Diskussion der Intuition von Hawkins

- Daten sind gewöhnlich multivariat (mehr-dimensional)
=> Basis-Modell ist univariat (ein-dimensional)
- Ein Datensatz stammt oft aus mehr als einem erzeugenden Prozess
=> Basis-Modell nimmt nur einen einzelnen genuinen erzeugenden Mechanismus an
- Anomalien können eine andere Klasse von Objekten sein (aus einem anderen Prozess erzeugt), die nicht besonders selten sind
=> Basis-Modell nimmt an, dass Outlier sehr selten sind

Eine große Zahl von Methoden wurde entwickelt, um über die Basis-Annahmen hinauszugelangen. Dabei liegen jedoch stets andere, oft nicht explizite Annahmen zugrunde.

Generelle Szenarien der Anwendung:

- supervised
 - in manchen Anwendungsgebieten gibt es Trainingsdaten mit normalen und ungewöhnlichen Fällen
 - es kann mehrere normale und ungewöhnliche Klassen geben
 - meist ist das Klassifikationsproblem unbalanziert
- semi-supervised
 - in manchen Szenarien gibt es Trainingsdaten nur für die normale oder nur für die ungewöhnliche Klasse
- unsupervised
 - in den meisten Szenarien gibt es keine Trainingsdaten

In dieser Vorlesung konzentrieren wir uns auf das unsupervised Szenario.

Erkennung von Outliern

- Nebenprodukt von Clustering?
- Manche Cluster-Algorithmen ordnen nicht jeden Punkt einem Cluster zu, sondern lassen "Noise" übrig.
- Idee: Wende Cluster-Verfahren an, betrachte Noise als Outlier.

- Problem:
 - Clustering Algorithmen sind daraufhin entwickelt und optimiert, Cluster zu finden.
 - Qualität der Outlier Detection hängt von Qualität der Cluster-Struktur und der Eignung des Clustering Algorithmus für diese Struktur ab.
 - Mehrere Outlier, die einander ähnlich sind, bilden eventuell auch selbst ein (kleines) Cluster, können also nicht entdeckt werden.

Klassifikation von Outlier Detection Algorithmen

- Globaler vs. lokaler Ansatz:
Wird die "Outlierness" bestimmt bezüglich des gesamten Datensatzes (global) oder nur bezüglich einer Auswahl?
- Labeling vs. Scoring
Bestimmt der Algorithmus den Outlier-Grad eines Punktes (Scoring) oder wird für jeden Punkt eine Entscheidung getroffen (Label: Outlier/kein Outlier)
- Eigenschaften des Outlier Modells
Auf welchen Eigenschaften beruht die Modellierung von "Outlierness"

- Global vs. Lokal
 - bezieht sich auf die Auflösung der Referenzmenge bezüglich derer die "Outlierness" bestimmt wird
 - Globale Ansätze:
 - Referenzmenge enthält gesamten Datensatz
 - Basis-Annahme: nur ein einziger (normaler) erzeugender Mechanismus
 - Grundlegendes Problem: Outlier sind auch in Referenzmenge und verfälschen die Ergebnisse
 - Lokale Ansätze:
 - Referenzmenge enthält nur eine (kleine) Teilmenge des Datensatzes
 - Meist keine Annahme über Anzahl der Mechanismen
 - Grundlegendes Problem: wie ist eine geeignete Referenzmenge zu bestimmen?
 - Beachte: Manche Ansätze liegen dazwischen
 - Auflösung der Referenzmenge wird im Verfahren variiert

- Labeling vs. Scoring
 - bezieht sich auf das Ergebnis, das der Algorithmus liefert
 - Labeling Ansätze:
 - binäre Entscheidung
 - Daten-Objekt wird als Outlier markiert oder als normal
 - Scoring Ansätze:
 - kontinuierlicher Output: für jedes Objekt wird ein Score geliefert (z.B. die Wahrscheinlichkeit, ein Outlier zu sein)
 - Objekte können nach ihrem Score geordnet werden
 - Beachte:
 - Viele Scoring-Ansätze bestimmen nur die top-n Outlier (Parameter n wird durch Benutzer angegeben)
 - Scoring-Ansätze können grundsätzlich in Labeling-Ansätze transformiert werden, wenn ein geeigneter Grenzwert angegeben werden kann, dessen Überschreitung zum Label "Outlier" führt

- Klassen von zugrundeliegenden Modellen
 - Statistisches Modell
 - Überlegung:
 - Wende ein Modell an, das die normalen Daten statistisch beschreibt (z.B. Gauss-Verteilung)
 - Outlier sind Punkte, die nicht gut zu diesem Modell passen (eine geringe Erzeugungswahrscheinlichkeit haben)
 - Beispiele:
 - Wahrscheinlichkeitstests basierend auf statistischen Modellen
 - Tiefen-basierte Ansätze
 - Deviation-based Ansätze
 - Manche Subspace Outlier Detection Ansätze

- Modellierung durch räumliche Nähe
 - Überlegung:
 - Untersuche die räumliche Nachbarschaft jedes Punktes im Datenraum
 - Wenn die Nachbarschaft deutlich andere Struktur (z.B. geringere Dichte) aufweist als die Nachbarschaften von anderen Punkten, kann der betreffende Punkt als Outlier angesehen werden.
 - Beispiele:
 - Distanz-basierte Ansätze
 - Dichte-basierte Ansätze
 - Manche Subspace Outlier Detection Ansätze

- Modellierung durch Winkel-Spektrum
 - Überlegung:
 - Bestimme das Spektrum paarweiser Winkel zwischen einem gegebenen Punkt und anderen (alle? Auswahl?) Punkten
 - Outlier sind Punkte, die eine geringe Varianz haben

Im Folgenden:

Orientierung an den verschiedenen Modellierungen

Übersicht

6.1 Einleitung ✓

6.2 Statistical Tests

6.3 Depth-based Approaches

6.4 Deviation-based Approaches

6.5 Distance-based Approaches

6.6 Density-based Approaches

6.7 High-dimensional Approaches

6.8 Summary

Literatur

Statistisches Modell

Modellierung durch
räumliche Nähe

Anpassung verschiedener
Modelle an spezielles Problem

General idea

- Given a certain kind of statistical distribution (e.g., Gaussian)
- Compute the parameters assuming all data points have been generated by such a statistical distribution (e.g., mean and standard deviation)
- Outliers are points that have a low probability to be generated by the overall distribution (e.g., deviate more than 3 times the standard deviation from the mean)

Basic assumption

- Normal data objects follow a (known) distribution and occur in a high probability region of this model
- Outliers deviate strongly from this distribution

A huge number of different tests are available differing in

- Type of data distribution (e.g. Gaussian)
- Number of variables, i.e., dimensions of the data objects (univariate/multivariate)
- Number of distributions (mixture models)
- Parametric versus non-parametric (e.g. histogram-based)

Example on the following slides

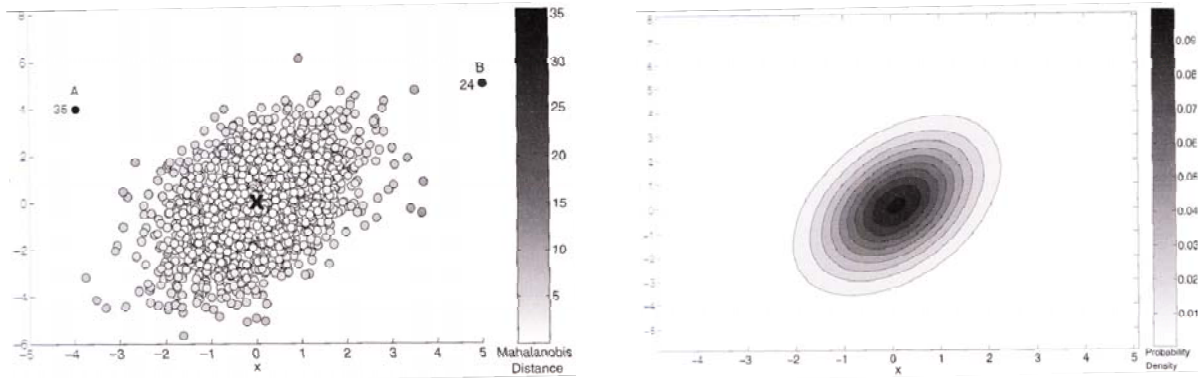
- Gaussian distribution
- Multivariate
- 1 model
- Parametric

Probability density function of a multivariate normal distribution

$$N(x) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} e^{-\frac{(x-\mu)^T \Sigma^{-1} (x-\mu)}{2}}$$

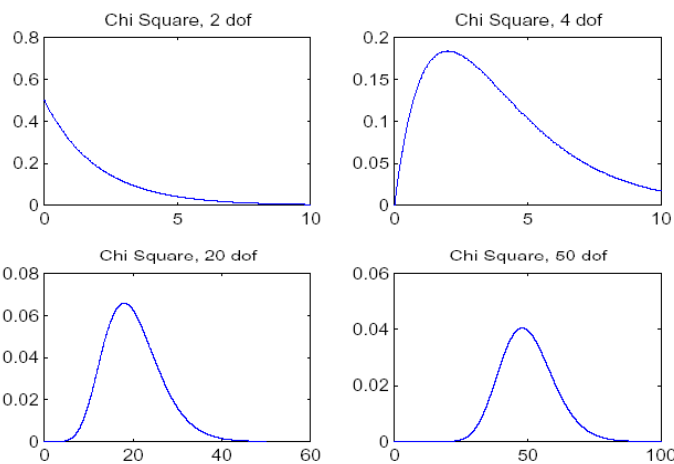
- μ is the mean value of all points (usually data are normalized such that $\mu=0$)
- Σ is the covariance matrix from the mean
- $MDist(x, \mu) = (x - \mu)^T \Sigma^{-1} (x - \mu)$ is the Mahalanobis distance of point x to μ
- MDist follows a χ^2 -distribution with d degrees of freedom ($d =$ data dimensionality)
- All points x , with $MDist(x, \mu) > \chi^2(0,975)$ [$\approx 3 \cdot \sigma$]

Visualization (2D) [Tan et al. 2006]



Problems

- Curse of dimensionality
 - The larger the degree of freedom, the more similar the MDist values for all points



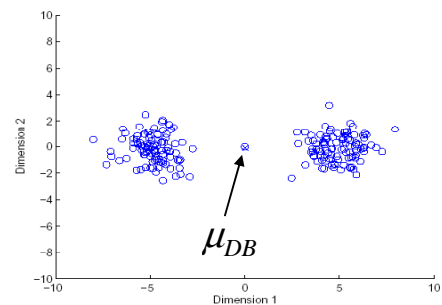
x-axis: observed *MDist* values
y-axis: frequency of observation

Problems (cont.)

- Robustness
 - Mean and standard deviation are very sensitive to outliers
 - These values are computed for the complete data set (including potential outliers)
 - The *MDist* is used to determine outliers although the *MDist* values are influenced by these outliers
- ⇒ Minimum Covariance Determinant [Rousseeuw and Leroy 1987]
minimizes the influence of outliers on the Mahalanobis distance

Discussion

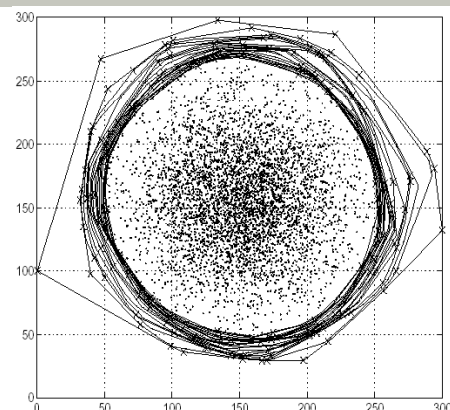
- Data distribution is fixed
- Low flexibility (no mixture model)
- Global method
- Outputs a label but can also output a score



23

General idea

- Search for outliers at the border of the data space but independent of statistical distributions
- Organize data objects in convex hull layers
- Outliers are objects on outer layers



Picture taken from [Johnson et al. 1998]

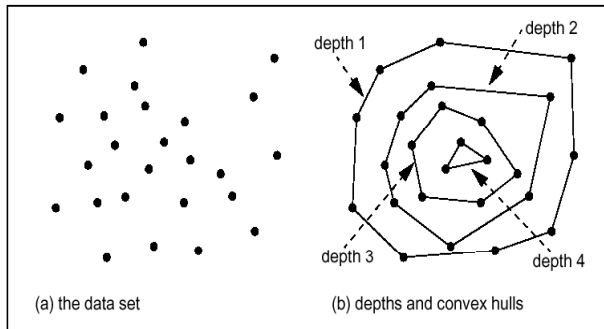
Basic assumption

- Outliers are located at the border of the data space
- Normal objects are in the center of the data space

24

Model [Tukey 1977]

- Points on the convex hull of the full data space have depth = 1
- Points on the convex hull of the data set after removing all points with depth = 1 have depth = 2
- ...
- Points having a depth $\leq k$ are reported as outliers



Picture taken from [Preparata and Shamos 1988]

Sample algorithms

- ISODEPTH [Ruts and Rousseeuw 1996]
- FDC [Johnson et al. 1998]

Discussion

- Similar idea like classical statistical approaches ($k = 1$ distributions) but independent from the chosen kind of distribution
- Convex hull computation is usually only efficient in 2D / 3D spaces
- Originally outputs a label but can be extended for scoring easily (take depth as scoring value)
- Uses a global reference set for outlier detection

General idea

- Given a set of data points (local group or global set)
- Outliers are points that do not fit to the general characteristics of that set, i.e., the variance of the set is minimized when removing the outliers

Basic assumption

- Outliers are the outermost points of the data set

Model [Arning et al. 1996]

- Given a smoothing factor $SF(I)$ that computes for each $I \subseteq DB$ how much the variance of DB is decreased when I is removed from DB
- With equal decrease in variance, a smaller exception set is better
- The outliers are the elements of the **exception set** $E \subseteq DB$ for which the following holds:

$$SF(E) \geq SF(I) \quad \text{for all } I \subseteq DB$$

Discussion:

- Similar idea like classical statistical approaches ($k = 1$ distributions) but independent from the chosen kind of distribution
- Naïve solution is in $O(2^n)$ for n data objects
- Heuristics like random sampling or best first search are applied
- Applicable to any data type (depends on the definition of SF)
- Originally designed as a global method
- Outputs a labeling

Übersicht

- 6.1 Einleitung ✓
 - 6.2 Statistical Tests ✓
 - 6.3 Depth-based Approaches ✓
 - 6.4 Deviation-based Approaches ✓
 - 6.5 Distance-based Approaches
 - 6.6 Density-based Approaches
 - 6.7 High-dimensional Approaches
 - 6.8 Summary
 - Literatur
- } Statistisches Modell
- } Modellierung durch räumliche Nähe
- } Anpassung verschiedener Modelle an spezielles Problem

General Idea

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

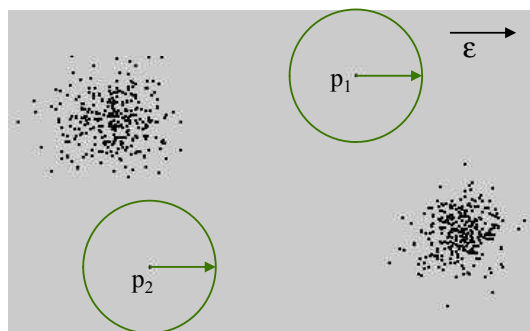
Basic Assumption

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

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

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



range-query with radius ϵ

- Algorithms
 - 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 $Card(DB) \cdot \pi$ points
 - 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
 - Grid-based [Knorr and Ng 1998]
 - Build grid such that any two points from the same grid cell have a distance of at most ϵ to each other
 - Points need only compared with points from neighboring cells

- Deriving intensional knowledge [Knorr and Ng 1999]
 - Relies on the $DB(\epsilon, \pi)$ -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

Outlier scoring based on k NN distances

- General models
 - 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]
- Algorithms
 - General approaches
 - Nested-Loop
 - » Naïve approach:
For each object: compute k NNs with a sequential scan
 - » Enhancement: use index structures for k NN queries
 - Partition-based
 - » Partition data into micro clusters
 - » Aggregate information for each partition (e.g. minimum bounding rectangles)
 - » Allows to prune micro clusters that cannot qualify when searching for the k NNs of a particular point

- Sample Algorithms (computing top- n outliers)
 - Nested-Loop [Ramaswamy et al 2000]
 - Simple NL algorithm with index support for k NN queries
 - Partition-based algorithm (based on a clustering algorithm that has linear time complexity)
 - Algorithm for the simple k NN-distance model
 - Linearization [Angiulli and Pizzuti 2002]
 - Linearization of a multi-dimensional data set using space-fill curves
 - 1D representation is partitioned into micro clusters
 - Algorithm for the average k NN-distance model
 - ORCA [Bay and Schwabacher 2003]
 - NL algorithm with randomization and simple pruning
 - Pruning: if a point has a score greater than the top- n outlier so far (cut-off), remove this point from further consideration
 - => non-outliers are pruned
 - => works good on randomized data (can be done in linear time)
 - => worst-case: naïve NL algorithm
 - Algorithm for both k NN-distance models and the $DB(\epsilon, \pi)$ -outlier model

- Sample Algorithms (cont.)
 - RBRP [Ghoting et al. 2006],
 - Idea: try to increase the cut-off as quick as possible => increase the pruning power
 - Compute approximate k NNs for each point to get a better cut-off
 - For approximate k NN search, the data points are partitioned into micro clusters and k NNs are only searched within each micro cluster
 - Algorithm for both k NN-distance models
 - Further approaches
 - Also apply partitioning-based algorithms using micro clusters [McCallum et al 2000], [Tao et al. 2006]
 - Approximate solution based on reference points [Pei et al. 2006]
- Discussion
 - Output can be a scoring (k NN-distance models) or a labeling (k NN-distance models and the $DB(\epsilon, \pi)$ -outlier model)
 - Approaches are local (resolution can be adjusted by the user via ϵ or k)

Variant

- Outlier Detection using In-degree Number [Hautamaki et al. 2004]
 - Idea
 - Construct the k NN graph for a data set
 - » Vertices: data points
 - » Edge: if $q \in k\text{NN}(p)$ then there is a directed edge from p to q
 - A vertex that has an indegree less or equal T (user defined threshold) is an outlier
 - Discussion
 - The indegree of a vertex in the k NN graph equals to the number of reverse k NNs (Rk NN) of the corresponding point
 - The Rk NNs of a point p are those data objects having p among their k NNs
 - Intuition of the model: outliers are
 - » points that are among the k NNs of less than T other points
 - » have less than T Rk NNs
 - Outputs an outlier label
 - Is a local approach (depending on user defined parameter k)

Resolution-based outlier factor (ROF) [Fan et al. 2006]

- Model
 - Depending on the resolution of applied distance thresholds, points are outliers or within a cluster
 - With the maximal resolution R_{max} (minimal distance threshold) all points are outliers
 - With the minimal resolution R_{min} (maximal distance threshold) all points are within a cluster
 - Change resolution from R_{max} to R_{min} in certain steps: points change from being outlier to being a member of a cluster
 - Cluster is defined similar as in DBSCAN as a transitive closure of r -neighborhoods (where r is the current resolution)
 - ROF value

$$ROF(p) = \sum_{R_{min} \leq r \leq R_{max}} \frac{clusterSize_{r-1}(p) - 1}{clusterSize_r(p)}$$
- Discussion
 - Outputs a score (the ROF value)
 - Resolution is varied automatically from local to global

General idea

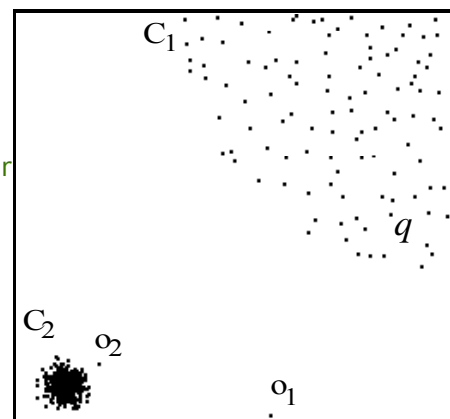
- Compare the density around a point with the density around its local neighbors
- The relative density of a point compared to its neighbors is computed as an outlier score
- Approaches also differ in how to estimate density

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

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

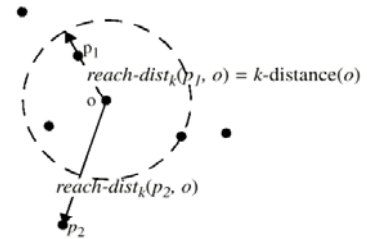


– Model

- Reachability distance

- Introduces a smoothing factor

$$reach-dist_k(p, o) = \max \{k\text{-distance}(o), dist(p, o)\}$$



- Local reachability distance (lrd) of point p

- Inverse of the average reach-dists of the k NNs of p

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

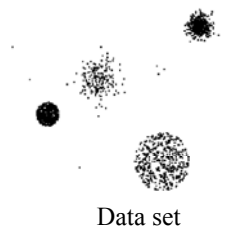
- Local outlier factor (LOF) of point p

- Average ratio of lrd's of neighbors of p and lrd of p

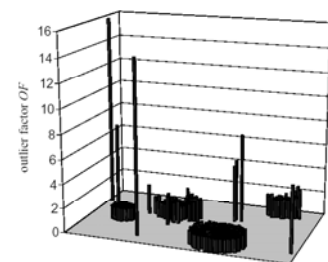
$$LOF_k(p) = \frac{\sum_{o \in kNN(p)} \frac{lrd_k(o)}{lrd_k(p)}}{Card(kNN(p))}$$

– Properties

- $LOF \approx 1$: point is in a cluster (region with homogeneous density around the point and its neighbors)



- $LOF \gg 1$: point is an outlier



LOFs ($MinPts = 40$)

– Discussion

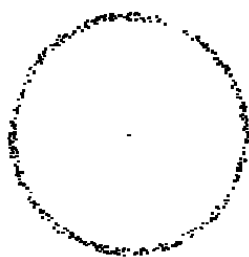
- Choice of k ($MinPts$ in the original paper) specifies the reference set
- Originally implements a local approach (resolution depends on the user's choice for k)
- Outputs a scoring (assigns an LOF value to each point)

Variants of LOF

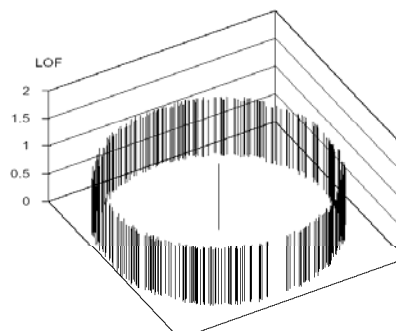
- Mining top- n local outliers [Jin et al. 2001]
 - Idea:
 - Usually, a user is only interested in the top- n outliers
 - Do not compute the LOF for all data objects => save runtime
 - Method
 - Compress data points into micro clusters using the CFs of BIRCH [Zhang et al. 1996]
 - Derive upper and lower 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

Variants of LOF (cont.)

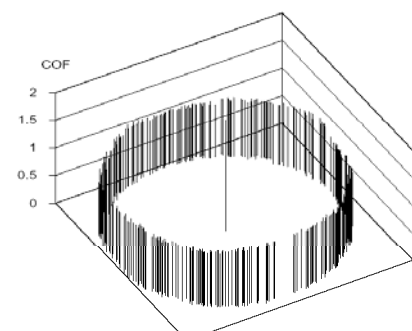
- Connectivity-based outlier factor (COF) [Tang et al. 2002]
 - Motivation
 - In regions of low density, it may be hard to detect outliers
 - Choose a low value for k is often not appropriate
 - Solution
 - Treat “low density” and “isolation” differently
 - Example



Data set



LOF

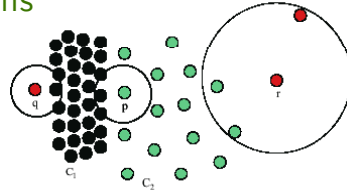


COF

Influenced Outlierness (INFLO) [Jin et al. 2006]

– Motivation

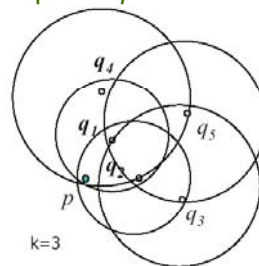
- If clusters of different densities are not clearly separated, LOF will have problems



Point p will have a higher LOF than points q or r which is counter intuitive

– Idea

- Take symmetric neighborhood relationship into account
- Influence space ($kIS(p)$) of a point p includes its $kNN(p)$ and its reverse $kNNs$ ($RkNN(p)$)



$$kIS(p) = kNN(p) \cup RkNN(p) = \{q_1, q_2, q_4\}$$

– Model

- Density is simply measured by the inverse of the kNN distance, i.e., $den(p) = 1/k\text{-distance}(p)$

- Influenced outlierness of a point p

$$INFLO_k(p) = \frac{\sum_{o \in kIS(p)} den(o)}{den(p) \cdot Card(kIS(p))}$$

- INFLO takes the ratio of the average density of objects in the neighborhood of a point p (i.e., in $kNN(p) \cup RkNN(p)$) to p 's density

– Proposed algorithms for mining top- n outliers

- Index-based
- Two-way approach
- Micro cluster based approach

- Properties
 - Similar to LOF
 - $INFLO \approx 1$: point is in a cluster
 - $INFLO \gg 1$: point is an outlier
- Discussion
 - Outputs an outlier score
 - Originally proposed as a local approach (resolution of the reference set kIS can be adjusted by the user setting parameter k)

Local outlier correlation integral (LOCI) [Papadimitriou et al. 2003]

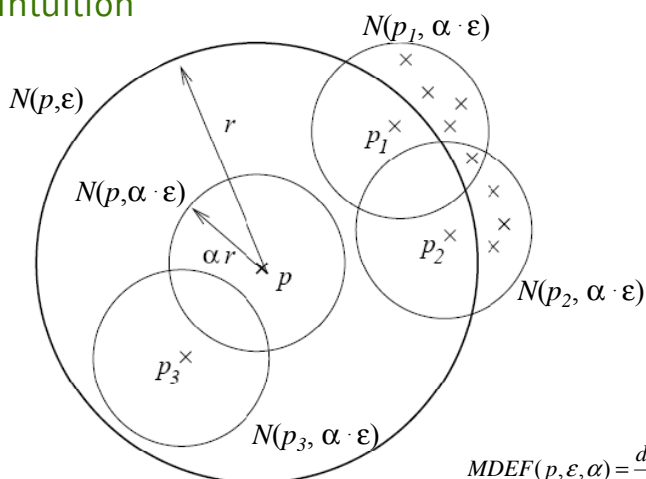
- Idea is similar to LOF and variants
- Differences to LOF
 - Take the ε -neighborhood instead of $kNNs$ as reference set
 - Test multiple resolutions (here called "granularities") of the reference set to get rid of any input parameter
- Model
 - ε -neighborhood of a point p : $N(p, \varepsilon) = \{q \mid dist(p, q) \leq \varepsilon\}$
 - Local density of an object p : number of objects in $N(p, \varepsilon)$
 - Average density of the neighborhood

$$den(p, \varepsilon, \alpha) = \frac{\sum_{q \in N(p, \varepsilon)} Card(N(q, \alpha \cdot \varepsilon))}{Card(N(p, \varepsilon))}$$

- Multi-granularity Deviation Factor (MDEF)

$$MDEF(p, \varepsilon, \alpha) = \frac{den(p, \varepsilon, \alpha) - Card(N(p, \alpha \cdot \varepsilon))}{den(p, \varepsilon, \alpha)} = 1 - \frac{Card(N(p, \alpha \cdot \varepsilon))}{den(p, \varepsilon, \alpha)}$$

– Intuition



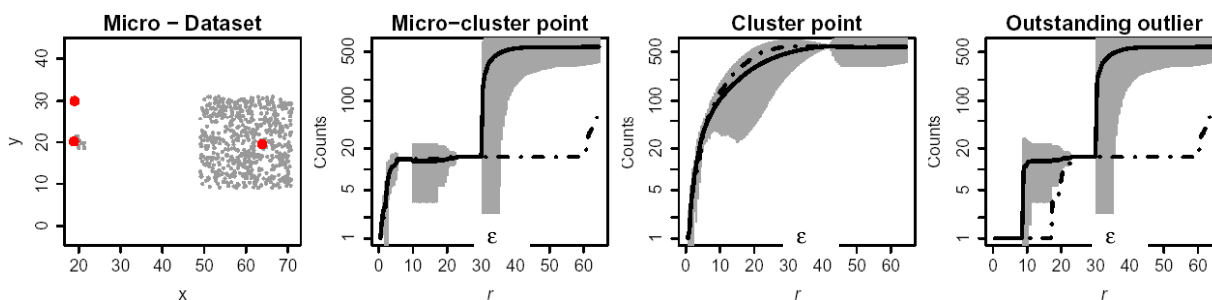
$$den(p, \epsilon, \alpha) = \frac{\sum_{q \in N(p, \epsilon)} Card(N(q, \alpha \cdot \epsilon))}{Card(N(p, \epsilon))}$$

$$MDEF(p, \epsilon, \alpha) = \frac{den(p, \epsilon, \alpha) - Card(N(p, \alpha \cdot \epsilon))}{den(p, \epsilon, \alpha)} = 1 - \frac{Card(N(p, \alpha \cdot \epsilon))}{den(p, \epsilon, \alpha)}$$

- $\sigma MDEF(p, \epsilon, \alpha)$ is the normalized standard deviation of the densities of all points from $N(p, \epsilon)$
- Properties
 - $MDEF = 0$ for points within a cluster
 - $MDEF > 0$ for outliers or $MDEF > 3 \cdot \sigma MDEF \Rightarrow$ outlier

– Features

- Parameters ϵ and α are automatically determined
- In fact, all possible values for ϵ are tested
- LOCI plot displays for a given point p the following values w.r.t. ϵ
 - $Card(N(p, \alpha \cdot \epsilon))$
 - $den(p, \epsilon, \alpha)$ with a border of $\pm 3 \cdot \sigma den(p, \epsilon, \alpha)$



– Algorithms

- Exact solution is rather expensive (compute MDEF values for all possible ϵ values)

- aLOCI: fast, approximate solution

- Discretize data space using a grid with side length $2\alpha\epsilon$

- Approximate range queries through grid cells

- ϵ - neighborhood of point p : $\zeta(p, \epsilon)$

- all cells that are completely covered by ϵ -sphere around p

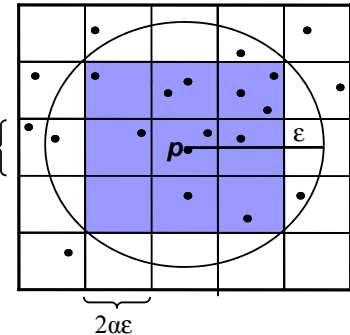
- Then,

$$\text{Card}(N(q, \alpha \cdot \epsilon)) = \frac{\sum_{c_j \in \zeta(p, \epsilon)} c_j^2}{\sum_{c_j \in \zeta(p, \epsilon)} c_j}$$

where c_j is the object count the corresponding cell

- Since different ϵ values are needed, different grids are constructed with varying resolution

- These different grids can be managed efficiently using a Quad-tree



– Discussion

- Exponential runtime w.r.t. data dimensionality

- Output:

- Label: if MDEF of a point $> 3 \cdot \sigma \text{MDEF}$ then this point is marked as outlier

- LOCI plot

- » At which resolution is a point an outlier (if any)

- » Additional information such as diameter of clusters, distances to clusters, etc.

- All interesting resolutions, i.e., possible values for ϵ , (from local to global) are tested

Übersicht

- 6.1 Einleitung ✓
 - 6.2 Statistical Tests ✓
 - 6.3 Depth-based Approaches ✓
 - 6.4 Deviation-based Approaches ✓
 - 6.5 Distance-based Approaches ✓
 - 6.6 Density-based Approaches ✓
 - 6.7 High-dimensional Approaches ✓
 - 6.8 Summary
 - Literatur
- } Statistisches Modell
 } Modellierung durch räumliche Nähe
 } Anpassung verschiedener Modelle an spezielles Problem

- One sample class of adaptations of existing models to a specific problem (high dimensional data)
- Why is that problem important?
 - Some (ten) years ago:
 - Data recording was expensive
 - Variables (attributes) were carefully evaluated whether or not they are relevant for the analysis task
 - Data sets usually contain only a few number of relevant dimensions
 - Nowadays:
 - Data recording is easy and cheap
 - "Everyone measures everything", attributes are not evaluated just measured
 - Data sets usually contain a large number of features
 - » Molecular biology: gene expression data with >1,000 of genes per patient
 - » Customer recommendation: ratings of 10-100 of products per person
 - » ...

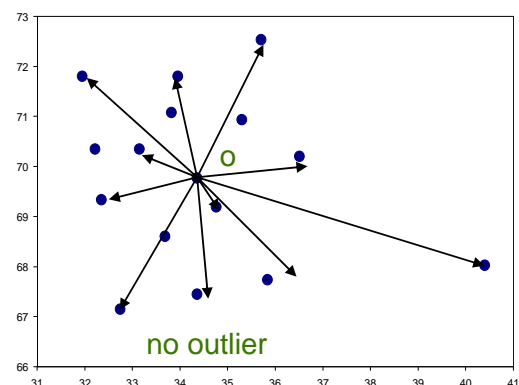
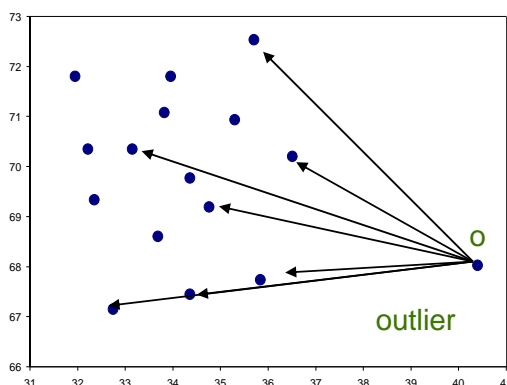
Challenges

- Curse of dimensionality
 - Relative contrast between distances decreases with increasing dimensionality
 - Data are very sparse, almost all points are outliers
 - Concept of neighborhood becomes meaningless

- Solutions
 - Use more robust distance functions and find full-dimensional outliers
 - Find outliers in projections (subspaces) of the original feature space

ABOD – angle-based outlier degree [Kriegel et al. 2008]

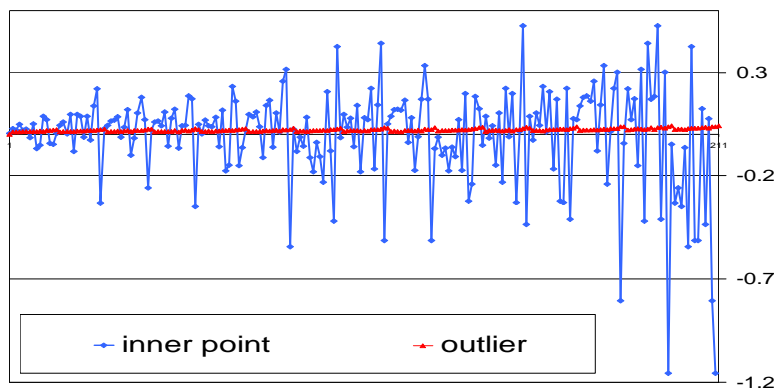
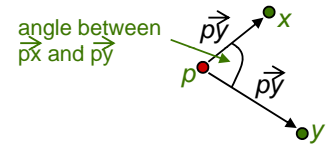
- Rational
 - Angles are more stable than distances in high dimensional spaces (cf. e.g. the popularity of cosine-based similarity measures for text data)
 - Object *o* is an outlier if most other objects are located in similar directions
 - Object *o* is no outlier if many other objects are located in varying directions



- Basic assumption
 - 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 \vec{px} and \vec{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



- Model (cont.)

- Measure the variance of the angle spectrum
- Weighted by the corresponding distances (for lower dimensional data sets where angles are less reliable)

$$ABOD(p) = VAR_{x,y \in DB} \left(\frac{\langle \vec{xp}, \vec{yp} \rangle}{\|\vec{xp}\|^2 \cdot \|\vec{yp}\|^2} \right)$$

• Properties

- Small ABOD => outlier
- High ABOD => no outlier

- Algorithms
 - Naïve algorithm is in $O(n^3)$
 - Approximate algorithm based on random sampling for mining top- n outliers
 - Do not consider all pairs of other points x,y in the database to compute the angles
 - Compute ABOD based on samples \Rightarrow lower bound of the real ABOD
 - Filter out points that have a high lower bound
 - Refine (compute the exact ABOD value) only for a small number of points
- Discussion
 - Global approach to outlier detection
 - Outputs an outlier score (inversely scaled: high ABOD \Rightarrow inlier, low ABOD \Rightarrow outlier)

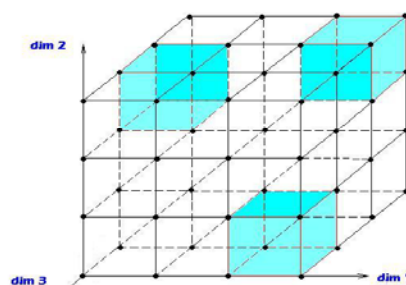
Grid-based subspace outlier detection [Aggarwal and Yu 2000]

- Model
 - Partition data space by an equi-depth grid (Φ = number of cells in each dimension)
 - Sparsity coefficient $S(C)$ for a k -dimensional grid cell C

$$S(C) = \frac{\text{count}(C) - n \cdot (\frac{1}{\Phi})^k}{\sqrt{n \cdot (\frac{1}{\Phi})^k \cdot (1 - (\frac{1}{\Phi})^k)}}$$

where $\text{count}(C)$ is the number of data objects in C

- $S(C) < 0 \Rightarrow \text{count}(C)$ is lower than expected
- Outliers are those objects that are located in lower-dimensional cells with negative sparsity coefficient



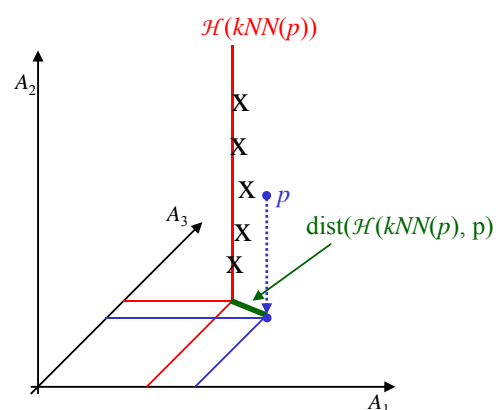
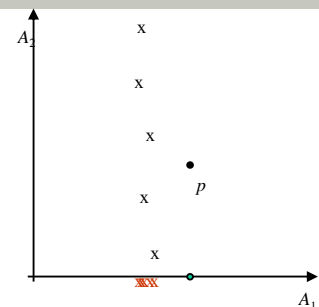
$\Phi = 3$

- Algorithm
 - Find the m grid cells (projections) with the lowest sparsity coefficients
 - Brute-force algorithm is in $O(\Phi^d)$
 - Evolutionary algorithm (input: m and the dimensionality of the cells)

- Discussion
 - Results need not be the points from the optimal cells
 - Very coarse model (all objects that are in cell with less points than to be expected)
 - Quality depends on grid resolution and grid position
 - Outputs a labeling
 - Implements a global approach (key criterion: globally expected number of points within a cell)

SOD – subspace outlier degree [Kriegel et al. 2009]

- Motivation
 - Outliers may be visible only in subspaces of the original data
- Model
 - Compute the subspace in which the k NNs of a point p minimize the variance
 - Compute the hyperplane $\mathcal{H}(kNN(p))$ that is orthogonal to that subspace
 - Take the distance of p to the hyperplane as measure for its "outlierness"



- Discussion
 - Assumes that k NNs of outliers have a lower-dimensional projection with small variance
 - Resolution is local (can be adjusted by the user via the parameter k)
 - Output is a scoring (SOD value)

1. Introduction ✓
2. Statistical Tests ✓
3. Depth-based Approaches ✓
4. Deviation-based Approaches ✓
5. Distance-based Approaches ✓
6. Density-based Approaches ✓
7. High-dimensional Approaches ✓
8. Summary

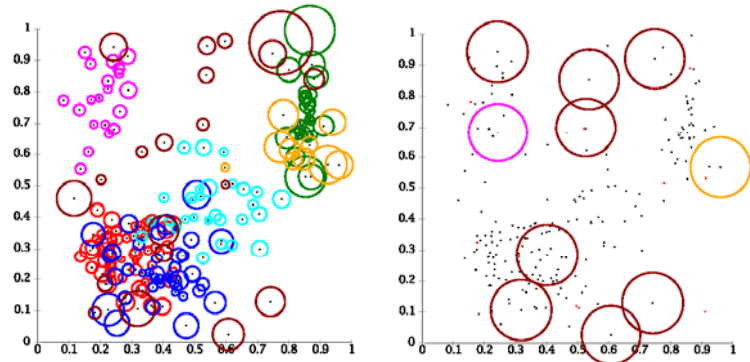
Summary

- Historical evolution of outlier detection methods
 - Statistical tests
 - Limited (univariate, no mixture model, outliers are rare, only one kind of distribution)
 - No emphasis on computational time
 - Extensions to these tests
 - Multivariate, mixture models, ...
 - Still no emphasis on computational time
 - Database-driven approaches
 - First, still statistically driven intuition of outliers
 - Emphasis on computational complexity
 - Database and data mining approaches
 - Spatial intuition of outliers
 - Even stronger focus on computational complexity (e.g. invention of top- n problem to propose new efficient algorithms)

- Consequence
 - Different models are based on different assumptions to model outliers
 - These assumptions are often not explicit but only implicit and not well understood
 - Different models provide different types of output (labeling/scoring)
 - Different models consider outlier at different resolutions (global/local)
 - Thus, different models will produce different results
 - A thorough and comprehensive comparison between different models and approaches is still missing

Outlook

- Experimental evaluation of different approaches to understand and compare differences and common properties
- A first step towards unification of the diverse approaches: providing density-based outlier scores as probability values [Kriegel et al. 2009a]: judging the deviation of the outlier score from the expected value
- Visualization
- New models
- Performance issues
- Complex data types
- High-dimensional data
- ...



- Achtert, E., Kriegel, H.-P., Reichert, L., Schubert, E., Wojdanowski, R., Zimek, A. 2010. Visual Evaluation of Outlier Detection Models. In Proc. International Conference on Database Systems for Advanced Applications (DASFAA), Tsukuba, Japan.
- Aggarwal, C.C. and Yu, P.S. 2000. Outlier detection for high dimensional data. In Proc. ACM SIGMOD Int. Conf. on Management of Data (SIGMOD), Dallas, TX.
- Angiulli, F. and Pizzuti, C. 2002. Fast outlier detection in high dimensional spaces. In Proc. European Conf. on Principles of Knowledge Discovery and Data Mining, Helsinki, Finland.
- Arning, A., Agrawal, R., and Raghavan, P. 1996. A linear method for deviation detection in large databases. In Proc. Int. Conf. on Knowledge Discovery and Data Mining (KDD), Portland, OR.
- Barnett, V. 1978. The study of outliers: purpose and model. Applied Statistics, 27(3), 242-250.
- Bay, S.D. and Schwabacher, M. 2003. Mining distance-based outliers in near linear time with randomization and a simple pruning rule. In Proc. Int. Conf. on Knowledge Discovery and Data Mining (KDD), Washington, DC.
- Breunig, M.M., Kriegel, H.-P., Ng, R.T., and Sander, J. 1999. OPTICS-OF: identifying local outliers. In Proc. European Conf. on Principles of Data Mining and Knowledge Discovery (PKDD), Prague, Czech Republic.
- Breunig, M.M., Kriegel, H.-P., Ng, R.T., and Sander, J. 2000. LOF: identifying density-based local outliers. In Proc. ACM SIGMOD Int. Conf. on Management of Data (SIGMOD), Dallas, TX.

- Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In Proc. Int. Conf. on Knowledge Discovery and Data Mining (KDD), Portland, OR.
- Fan, H., Zaïane, O., Foss, A., and Wu, J. 2006. A nonparametric outlier detection for efficiently discovering top- n outliers from engineering data. In Proc. Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD), Singapore.
- Ghoting, A., Parthasarathy, S., and Otey, M. 2006. Fast mining of distance-based outliers in high dimensional spaces. In Proc. SIAM Int. Conf. on Data Mining (SDM), Bethesda, ML.
- Hautamaki, V., Karkkainen, I., and Franti, P. 2004. Outlier detection using k -nearest neighbour graph. In Proc. IEEE Int. Conf. on Pattern Recognition (ICPR), Cambridge, UK.
- Hawkins, D. 1980. Identification of Outliers. Chapman and Hall.
- Jin, W., Tung, A., and Han, J. 2001. Mining top- n local outliers in large databases. In Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD), San Francisco, CA.
- Jin, W., Tung, A., Han, J., and Wang, W. 2006. Ranking outliers using symmetric neighborhood relationship. In Proc. Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD), Singapore.
- Johnson, T., Kwok, I., and Ng, R.T. 1998. Fast computation of 2-dimensional depth contours. In Proc. Int. Conf. on Knowledge Discovery and Data Mining (KDD), New York, NY.
- Knorr, E.M. and Ng, R.T. 1997. A unified approach for mining outliers. In Proc. Conf. of the Centre for Advanced Studies on Collaborative Research (CASCON), Toronto, Canada.

- Knorr, E.M. and NG, R.T. 1998. Algorithms for mining distance-based outliers in large datasets. In Proc. Int. Conf. on Very Large Data Bases (VLDB), New York, NY.
- Knorr, E.M. and Ng, R.T. 1999. Finding intensional knowledge of distance-based outliers. In Proc. Int. Conf. on Very Large Data Bases (VLDB), Edinburgh, Scotland.
- Kriegel, H.-P., Kröger, P., Schubert, E., and Zimek, A. 2009. Outlier detection in axis-parallel subspaces of high dimensional data. In Proc. Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD), Bangkok, Thailand.
- Kriegel, H.-P., Kröger, P., Schubert, E., and Zimek, A. 2009a. LoOP: Local Outlier Probabilities. In Proc. ACM Conference on Information and Knowledge Management (CIKM), Hong Kong, China.
- Kriegel, H.-P., Schubert, M., and Zimek, A. 2008. Angle-based outlier detection, In Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD), Las Vegas, NV.
- McCallum, A., Nigam, K., and Ungar, L.H. 2000. Efficient clustering of high-dimensional data sets with application to reference matching. In Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD), Boston, MA.
- Papadimitriou, S., Kitagawa, H., Gibbons, P., and Faloutsos, C. 2003. LOCI: Fast outlier detection using the local correlation integral. In Proc. IEEE Int. Conf. on Data Engineering (ICDE), Hong Kong, China.
- Pei, Y., Zaiane, O., and Gao, Y. 2006. An efficient reference-based approach to outlier detection in large datasets. In Proc. 6th Int. Conf. on Data Mining (ICDM), Hong Kong, China.
- Preparata, F. and Shamos, M. 1988. Computational Geometry: an Introduction. Springer Verlag.

- Ramaswamy, S. Rastogi, R. and Shim, K. 2000. Efficient algorithms for mining outliers from large data sets. In Proc. ACM SIGMOD Int. Conf. on Management of Data (SIGMOD), Dallas, TX.
- Rousseeuw, P.J. and Leroy, A.M. 1987. Robust Regression and Outlier Detection. John Wiley.
- Ruts, I. and Rousseeuw, P.J. 1996. Computing depth contours of bivariate point clouds. Computational Statistics and Data Analysis, 23, 153–168.
- Tao Y., Xiao, X. and Zhou, S. 2006. Mining distance-based outliers from large databases in any metric space. In Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD), New York, NY.
- Tan, P.-N., Steinbach, M., and Kumar, V. 2006. Introduction to Data Mining. Addison Wesley.
- Tang, J., Chen, Z., Fu, A.W.-C., and Cheung, D.W. 2002. Enhancing effectiveness of outlier detections for low density patterns. In Proc. Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD), Taipei, Taiwan.
- Tukey, J. 1977. Exploratory Data Analysis. Addison-Wesley.
- Zhang, T., Ramakrishnan, R., Livny, M. 1996. BIRCH: an efficient data clustering method for very large databases. In Proc. ACM SIGMOD Int. Conf. on Management of Data (SIGMOD), Montreal, Canada.