Ludwig Maximilians Universität München
Institut für Informatik
Lehr- und Forschungseinheit für Datenbanksysteme

Skript zur Vorlesung

Knowledge Discovery in Databases

im Wintersemester 2006/2007

Kapitel 6: Outlier Detection

http://www.dbs.ifi.lmu.de/Lehre/KDD

6 Outlier Detection

Übersicht

Was ist ein Outlier?

Was ist ein Outlier?

- Beim Clustering: Rauschen (alle Punkte, die zu keinem Cluster gehören)
- Generell : keine allgemein gültige und akzeptierte Definition
- „One person‘s noise could be another person‘s signal.“

Beispiele:

- Sport: Michael Jordon, Thomas "Icke" Häßler, ...

Anwendungen:

- Kreditkarten-Mißbrauch
- Telefonkunden-Betrug
- Medizinische Analyse

Was ist ein Outlier?

Definitionen

- Nach 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.”

Erkennung von Outliern

- Ziel: Erkennen des anderen Mechanismus
- Wenn Trainingsbeispiele für diesen anderen Mechanismus exisitieren kann das Problem mit Klassifikation gelöst werden
  ACHTUNG: Probleme da Trainingsdatenmenge für “outlier”- Klasse meist sehr viel kleiner als für “normal”
- In den meisten Anwendungen: keine Trainingsdaten für “outlier” vorhanden

  => Outlier Detection ist ein *unsupervised learning* Task

Outlier Detection in der Statistik

Idee

- Modelliere Daten als multivariate Normalverteilung
- Punkte deren Abstand (quadratische Formdistanz) zum Mittelwert μ größer als Grenzwert Θ (z.B. $\Theta = 3 \cdot \sigma$) ist, sind Outlier

Multivariate Normalverteilung

$$N(x) = \frac{1}{(2\pi\sigma^2)^{\frac{1}{2}}} e^{-\frac{1}{2}[(x-\mu)^T \Sigma^{-1} (x-\mu)]}$$

Quadratische Formdistanz (Mahalanobis Distanz) des Punktes x vom Mittelwert μ der Normalverteilung

Quadratische Formdistanz

- Die quadratischen Formdistanzen der Punkte zum Mittelwert der Normalverteilung folgen einer χ^2 (Chi-Square)-Verteilung mit d Freiheitsgraden (d = Dimensionalität des Datenraums)

Algorithms zur Erkennung multivariater Outlier

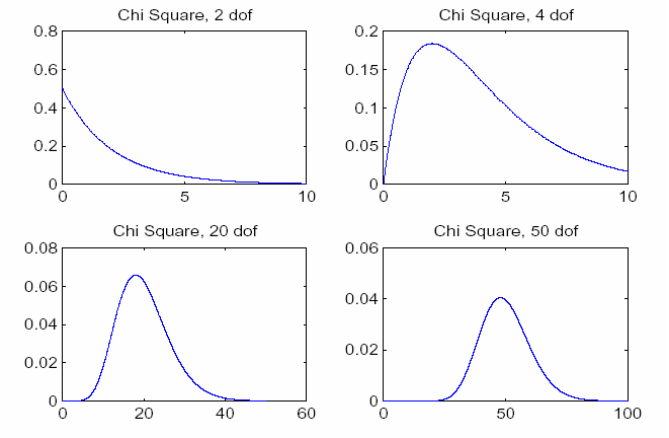
- Input: d-dimensionale Punktmenge DB
  - Berechne den Mittelwert μ_{DB} aller Punkte $\mu_{DB} = \frac{1}{|DB|} \sum_{x \in DB} x$
  - Berechne die (d x d) Kovarianzmatrix Σ_{DB} aller Punkte
  - Berechne für jeden Punkt $x \in DB$ die quadratische Formdistanz von x zum Mittelwert μ_{DB}

    $D(x, \mu_{DB}) = (x - \mu_{DB})^T \Sigma^{-1} (x - \mu_{DB})$
- Output: alle Punkte x, deren Abstand zum Mittelwert größer als $\chi^2(0{,}975)$ ist

  $\text{OutlierSet} = \{x \in DB \mid D(x, \mu_{DB}) > \chi^2(0{,}975)\}$

Probleme

- "Curse of Dimensionality"
  - Distanzen werden in hochdimensionalen Räumen unaussagekräftig
  - Je höher die Dimensionalität des Datenraums (Freiheitsgrade der Verteilung), desto ähnlicher werden die quadratischen Formdistanzen



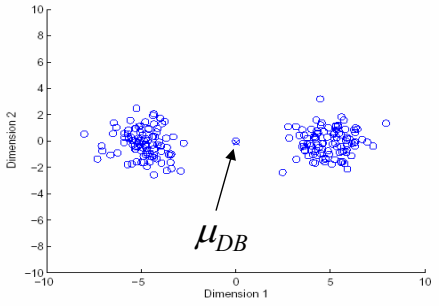
dof = degree of freedom

Probleme (cont.)

- Robustheit
  - Mittelwert und Varianz/Kovarianz extrem sensitiv gegenüber Outliern
  - Verwendung der quadratische Formdistanz zur Outlier-Entdeckung obwohl diese Distanz selbst durch Outlier beeinflußt ist (da abhängig von der Kovarianzmatrix)

    => Minimum Covariance Determinant [Rousseeuw, Driessen 99] minimiert

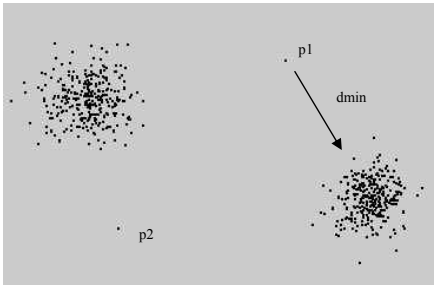
    den Einfluss von Outliern auf die quadratische Formdistanz
- Flexibilität
  - Datenverteilung muß vorher bekannt sein
  - Keine "Mixture of Gaussians"
  - Beispiel: Mittelwert der Daten ist ein Outlier!!!



6.2 Distanzbasierte Outlier

Definition "*(pct,dmin)*-Outlier" [Knorr, Ng 97]

- Ein Objekt *p* in einem Datensatz *DB* ist ein *(pct,dmin)*-Outlier, falls mindestens *pct* - Prozent von Objekten aus *DB* eine größere Distanz als *dmin* zu *p* haben.



Wahl von *pct* und *dmin* wird einem Experten überlassen.

Beispiel: $p_1 \in DB$, *pct=0.95, dmin=8*

p_1 *ist (0.95,8)-Outlier* => *95%* von Objekten aus *DB* haben eine Distanz > *8* zu p_1

6.2 Distanzbasierte Outlier

Alternative Definitionen

- „(k,dmax)"-Outlier [Kollios, Gunopulos, Kiudas, Berchtold 03]

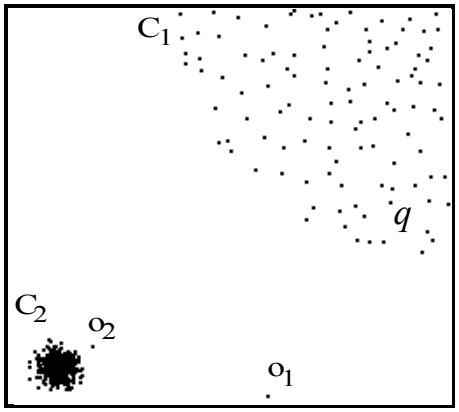
  Ein Objekt *p* in einem Datensatz *DB* ist ein *(k,dmax)*-Outlier, falls höchstens *k* Objekte aus *DB* eine kleinere Distanz als *dmax* zu *p* haben.

- *k*NN-Outlier [Ramaswamy, Rastogi, Shim 03]

  Die *n* Objekte in *DB* mit den höchsten *k*-nächste-Nachbar-Distanzen sind Outlier

Probleme (siehe Beispiel)

- *(pct,dmin)*-Outlier: welche Werte sollen *pct* und *dmin* annehmen, so daß o_2 ein Outlier ist, nicht aber die Objekte des Cluster C_1 (z.B. $q \in C_1$)?
- *(k,dmax)*-Outlier: analog
- *k*NN-Outlier: *k*NN-Distanz der Objekte in C_1 größer als von o_2



Lokale Identifikation von Outlier

- Nicht nur binäre Eigenschaften für Outlier (Outlier? JA oder NEIN)
- Bei Clustern mit unterschiedlicher Dichte, können beim *distance-based* - Ansatz Probleme auftreten

Lösung : *Density-based Local Outlier*

– Weise jedem Objekt einen *Grad* zu, zu dem das Objekt ein Outlier ist

$\Rightarrow$ Local Outlier Factor (LOF)

– Lokale Nachbarschaft von Objekten wird berücksichtigt

Local Outlier Factor (LOF) [Breunig, Kriegel, Ng, Sander 00]

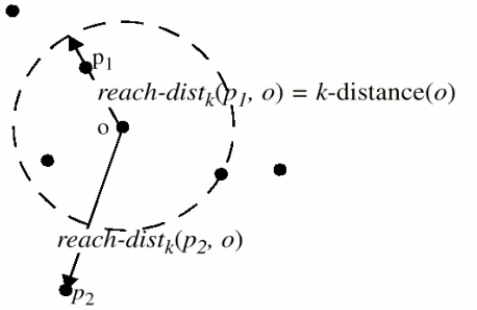
- *k-Distanz* von *p* = *dist(p,o),* für jedes *k,* so dass gilt: (*o* Î *DB*)
  (i) für mindestens *k* Objekte *q* Î *DB* gilt : dist(p,q) ≤ dist(p,o)
  (ii) für höchstens *k-1* Objekte *q* Î *DB* gilt : dist(p,q) < dist(p,o)

- *k-Distanz - Nachbarschaft* von *p*:

  $N_{\text{k-distance(p)}}(p) = \{q \in DB \setminus \{p\} \mid dist(p,q) \leq \text{k-distance}(p)\}$

- *Erreichbarkeits-Distanz* :

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



Local Outlier Factor (LOF)

- Als Parameter nur *MinPts*
- Lokale Erreichbarkeits-Distanz von *p*:

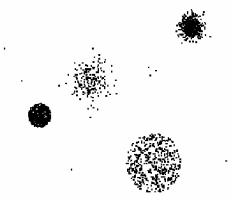
$$lrd_{MinPts}(p) = 1 \Big/ \left(\frac{\sum_{o \in N_{MinPts}(p)} reach\text{-}dist_{MinPts}(p, o)}{|N_{MinPts}(p)|} \right)$$

- Local Outlier Factor von *p* (LOF):

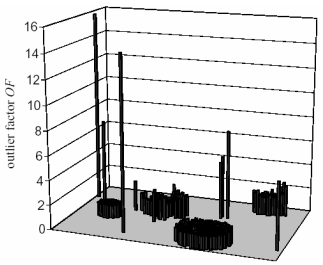
$$LOF_{MinPts}(p) = \frac{\sum_{o \in N_{MinPts}(p)} \frac{lrd_{MinPts}(o)}{lrd_{MinPts}(p)}}{|N_{MinPts}(p)|}$$

Local Outlier Factor (LOF)

- LOF (*p*) » 1:
  Punkt liegt weit innen im Cluster
- LOF (*p*) >> 1:
  Punkt ist ein starker lokaler Outlier

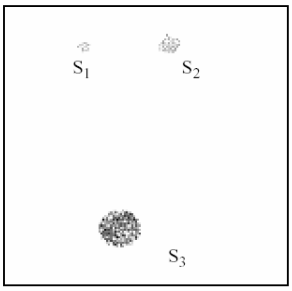


Datensatz

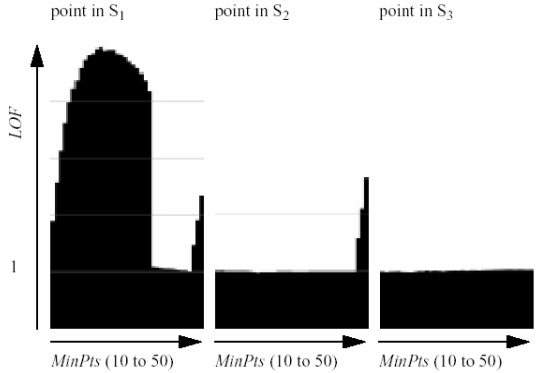


LOFs (*MinPts* = 40)

Sensitivität bzgl. *MinPts*



Example dataset



6.4 Referenzpunktbasierte Outlier

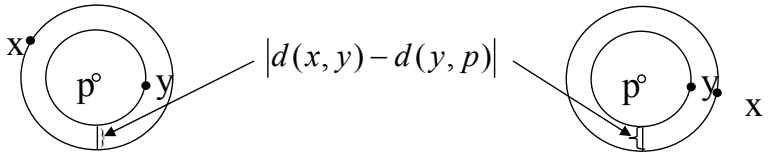
Bisherige Outlier Detection Verfahren haben eine Worst-Case-Komplexität von $O(n^2)$
=> für große Datenmengen schwer anwendbar

Idee: Outlier Detection mit Referenzpunkten
Geg: Datenmenge $DB = \{x_1,..,x_n\}$,
Referenzpunkte $P = \{p_1,..,p_l\}$, Distanzmetrik $d(x,y)$.

- *Featurereduktion mit Referenzpunkten*
  (jede Dimension entspricht dem Abstand zu einem Referenzpunkt)
- *Wenn die durchschnittliche kNN Distanz eines Punktes x bereits in einer Dimension hoch ist kann ein Punkt nicht in einem Cluster liegen.*
- *Anstatt zu entscheiden ob ein Punkt ein Outlier ist generiere Ranking (ähnlich LOF)*

6.4 Referenzpunktbasierte Outlier

Definition: Objekt x ist ***Referenzpunkt-nächster Nachbar*** von Objekt y wenn gilt $\left|d(x,y)-d(y,p)\right| = \min_{1\le i\le n}\left|d(x,p)-d(x_i,p)\right|$
bzgl. Referenzpunkt $p \in P$.



k-Referenzpunkt-nächste Nachbarn analog.

Definition: Relativer Outlier-Grad (relative outlier degree)

$$D(x,k,p) = \frac{1}{k}\sum_{j=1}^{k}\left|d(x_j,p)-d(x,p)\right|$$

mit $\{x_1,..,x_k\}$ die k-Referenzpunkt-Nächste Nachbarn

Definition: Nachbarschaftsdichte

$$D^p(x,k) = \min_{1 \le j \le r} \frac{1}{D(x,k,p_j)}$$

mit den Referenzpunkten $P = \{p_1,..,p_r\}$.

Definition: ROS (Reference Outlier Score)

$$ROS(x) = 1 - \frac{D^P(x,k)}{\max_{1 \le i \le n} D^p(x_i,k)}$$

ein hoher ROS deutet auf einen Outlier hin.

Algorithmus:

```
FOR  EACH x ∈ DB DO
  x.D_P = MAXVALUE
FOR  EACH  p ∈ P DO
  FOR EACH  x ∈ DB DO
          Bestimme k-Referenzpunkt-Nächste
  Nachbarn für x bzgl. P
          x.D_P = min(x.D_P, D(x,k,p)  )
FOR EACH x ∈ DB DO
  berechne x.ROS
Sortiere DB nach ROS
```

6.4 Referenzpunktbasierte Outlier

Komplexität:

Für jede Referenzpunkt $p \in P$ und jedes Datenobjekt $x \in DB$ wird $D(x,k,p)$ bestimmt: $O(n \log n)$ (sortiere aller Elemente)

$\Rightarrow$ der Algorithmus hat eine worst-case Zeitkomplexität von $O(|P| \, |DB| \log |DB|)$

Allerdings: Linear bzgl. Speicherkapazität (komplettes Ranking)

6.5. Winkelbasierte Outlier Detection (ABOD)

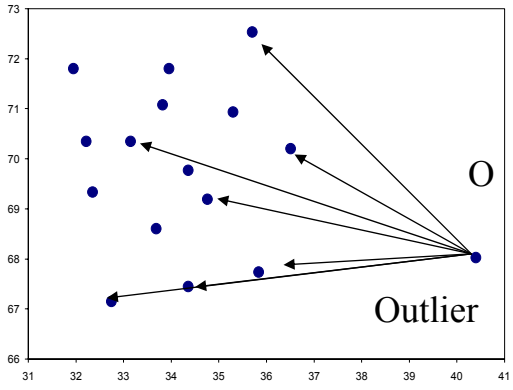
Winkel sind stabiler in hochdimensionalen Daten als euklidische Distanzen.

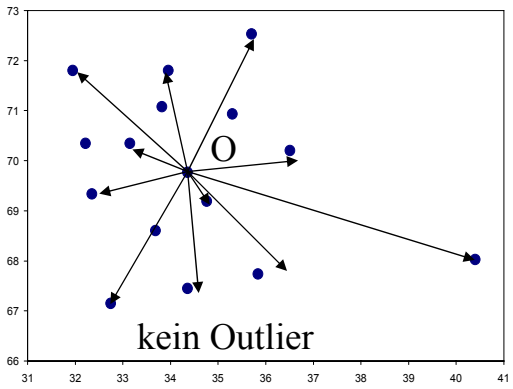
Object O ist ein Outlier

$\Rightarrow$ andere Objekte liegen alle in derselben/wenigen Richtung(en)

Object O ist kein Outlier

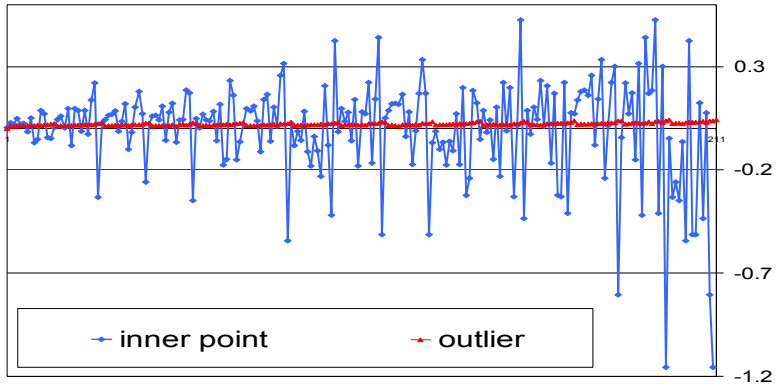
$\Rightarrow$ O ist in der Mitte aller seiner Nachbarn





Winkelbasierter Outlier-Faktor (ABOF)

Idee des ABOF (Angle Based Outlier Factor)
Betrachte für jeden Punkte die Winkel für alle Paare B,C in DB.
Untersuche die Unterschiedlichkeit der auftretenden Winkel.



Fazit:
Ranke Outliers nach der Breite ihres Winkelspectrums.

Angle-Based Outlier Factor (ABOF)

Definition: ABOF

Der ABOF des Punkt P ist die Varianz der Winkel aller Objekt B,C aus der Datenbank DB bzgl. P.

Zusätzlich wird der ABOF noch nach den Distanzen von B,C nach A gewichtet.

$$ABOF(A) = \underset{B,C \in DB}{VAR}\left(\frac{\langle \overline{AB}, \overline{AC} \rangle}{\|\overline{AB}\|^2 \cdot \|\overline{AC}\|^2}\right)$$

kleiner ABOF => Outlier

großer ABOF => kein Outlier

Berechnung des ABOF

Zeitkomplexität von ABOD mit naiven Vorgehen : $O(n^3)$

Idee: Nicht alle Paare anderer Objekte sind notwendig um ein breites spectrum zu erkennen.

Bereits der ABOF einer Stichprobe erlaubt eine Abschätzung die eine untere Schranke des exakten ABOFs darstellt

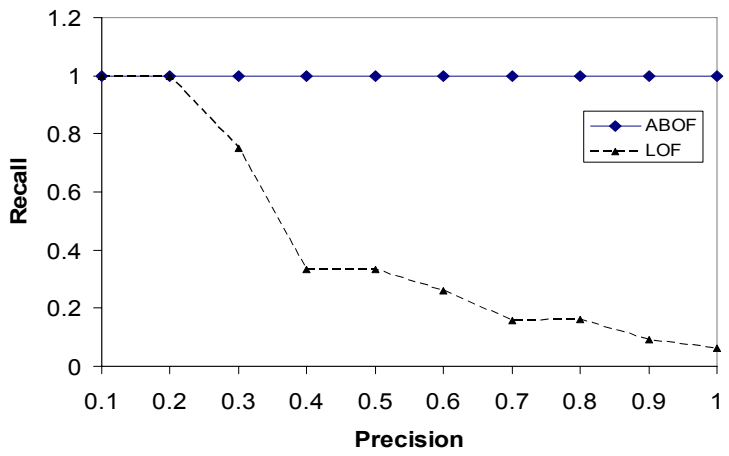
Diese wird für ein Filter-Verfeinerungsverfahren verwendet, die exakten Outlier-Faktoren nur für wenige Punkte berechnene muß.

$\Rightarrow$ ABOF für DB kann mit einer durchschnittlichen Laufzeit von $O(n^2 \cdot s)$ berechnet werden

Experiments

Beispielexperiment auf künstlichen Daten:
100 Dimensionen
5000 Datenobjekte durch Gaussian Mixture Model generiert.
10 gleichverteilte Outlier

Vergleich der Rankings von LOF und ABOF.



Zusammenfassung

- hier: Outlier Detection als unsupervized Task
- Distanzbasierte Ansätze: Basisansatz
- Dichtebasierte Ansatz: Für Datensätze mit unterschiedlich dichten Regionen.
- Referenzpunktansatz: für große Datenmengen
- Winkelbasierter Ansatz: Für hochdimensionale Daten