DATABASE SYSTEMS GROUP

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

LMU

Skript zur Vorlesung

Knowledge Discovery in Databases

im Sommersemester 2013

Kapitel 7: Evaluation von unsupervised Verfahren

Vorlesung: Dr. Arthur Zimek
Übungen: Erich Schubert

Skript © 2013 Arthur Zimek

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

7. Evaluation von unsupervised Verfahren

Überblick

7.1 Bewertung von Clustern und Outliern

Unterschied der Aufgaben (7.1 vs. 7.2-3):

- Bewertung von Clustern/Outliern:
  - Anwendung eines Verfahrens auf ein konkretes Problem (einen bestimmten Datensatz, über den man Neues erfahren möchte).

    Zur Erinnerung:

    Knowledge Discovery in Databases (*KDD*) ist der Prozess der (semi-) automatischen Extraktion von Wissen aus Datenbanken, das *gültig, bisher unbekannt* und *potentiell nützlich* ist.
  - Frage: Was kann man mit den Ergebnissen anfangen?

- Bewertung (Evaluation) von Clustering-/Outlier-Detection-*Algorithmen*
  - nicht unbedingt neue Erkenntnisse, aber gut überprüfbare
  - Anwendung auf Daten, die bereits gut bekannt sind
  - Anwendung auf künstliche Daten, deren Struktur *by design* bekannt ist
  - Frage: Werden Eigenschaften/Strukturen gefunden, die der Algorithmus nach seinem Model finden sollte? Besser als andere Algorithmen?
  - Überprüfbarkeit *alleine* ist aber fragwürdig!

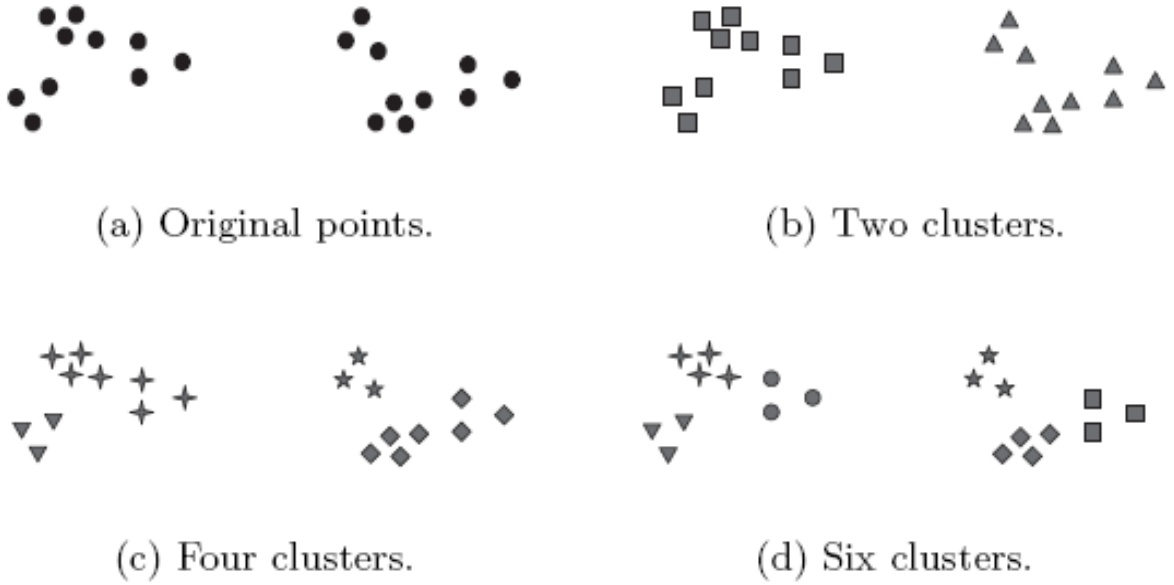
7.1 Bewertung von Clustern und Outliern

grundsätzlich: Clustering ist unsupervised

- ein Clustering ist nicht richtig oder falsch, sondern mehr oder weniger sinnvoll
- ein “sinnvolles” Clustering wird von den verschiedenen Verfahren auf der Grundlage von verschiedenen Annahmen (Heuristiken!) angestrebt
- Überprüfung der Sinnhaftigkeit erfordert Fachwissen über die Datengrundlage

7.1 Bewertung von Clustern und Outliern

verschiedene Möglichkeiten, eine Punktmenge zu clustern



(a) Original points.

(b) Two clusters.

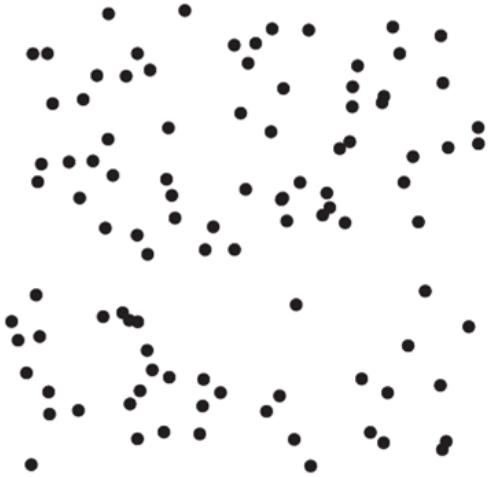
(c) Four clusters.

(d) Six clusters.

aus: Tan, Steinbach, Kumar: Introduction to Data Mining (Pearson, 2006)

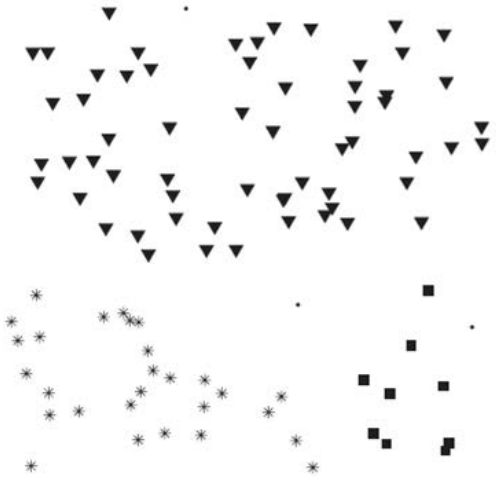
7.1 Bewertung von Clustern und Outliern

Cluster-Ergebnisse in Zufallsdaten (Gleichverteilung)

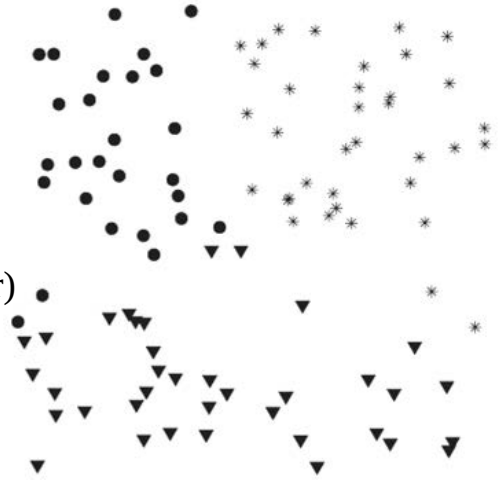


Datensatz
(100 gleichverteilte 2D Punkte)

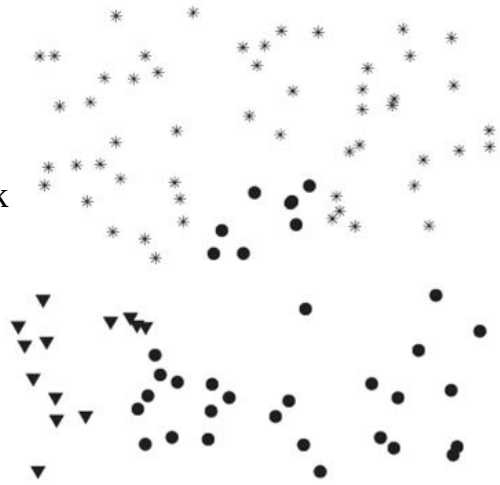
DBSCAN (3 Cluster)



k-means
(3 Cluster)



complete link
(3 Cluster)



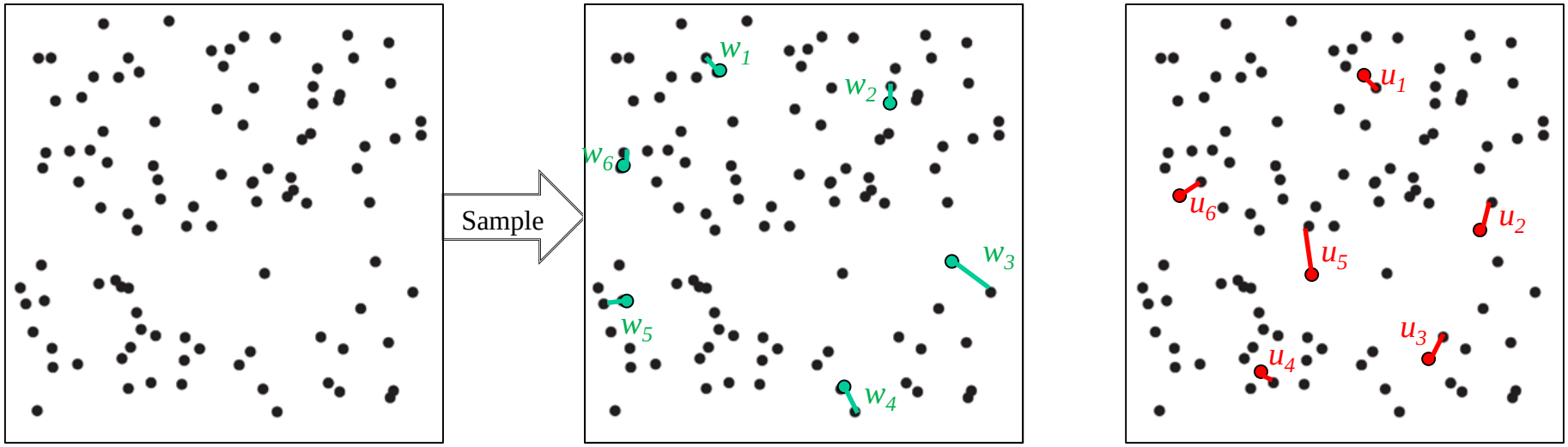
nach: Tan, Steinbach, Kumar:
Introduction to Data Mining
(Pearson, 2006)

7.1 Bewertung von Clustern und Outliern

Cluster-Tendenz in einem Datensatz?

- Viele Verfahren finden in jedem Datensatz Cluster, egal ob welche da sind oder nicht.
- Test, ob Cluster in einem Datensatz sind:
  - Wende ein Cluster-Verfahren an
  - Teste ob wenigstens einige der gefundenen Cluster sinnvoll sind
- Problem: Negatives Ergebnis erlaubt keine Aussage
  - Es könnten Cluster vorhanden sein, die einem anderen Modell entsprechen (werden vom angewendeten Verfahren nicht gefunden)
- Test unabhängig vom Clustering: Gibt es im Datensatz überhaupt eine Tendenz zu clustern?

Hopkins-Statistik für Cluster-Tendenz



Datensatz
(n Punkte)

Zufallsauswahl aus Datensatz
(m Punkte) $m << n$

m gleichverteilte Punkte

w_i: Distanzen der ausgewählten Punkte zu ihrem nächsten Nachbarn im Originaldatensatz

u_i: Distanzen der gleichverteilten Punkte zu ihrem nächsten Nachbarn im Originaldatensatz

$$H = \frac{\sum_{i=1}^{m} u_i}{\sum_{i=1}^{m} u_i + \sum_{i=1}^{m} w_i}$$

$0 \leq H \leq 1$

H ≈ 0 : Daten sind sehr regulär (z.B. verteilt auf Gitter)
H ≈ 0,5 : Daten sind gleichverteilt
H ≈ 1 : Daten sind stark geclustert

7.1 Bewertung von Clustern und Outliern

Bewertung von Outliern:

starkes Outlier-Signal durch ein Outlier-Detection-Verfahren
$\Rightarrow$ das Objekt *könnte* ein Outlier sein

Entsprechende Unsicherheit in Outlier-Definitionen:

"an observation (or subset of observations) which appears to be inconsistent with the remainder of that set of data"

(Barnett, Lewis 1994)

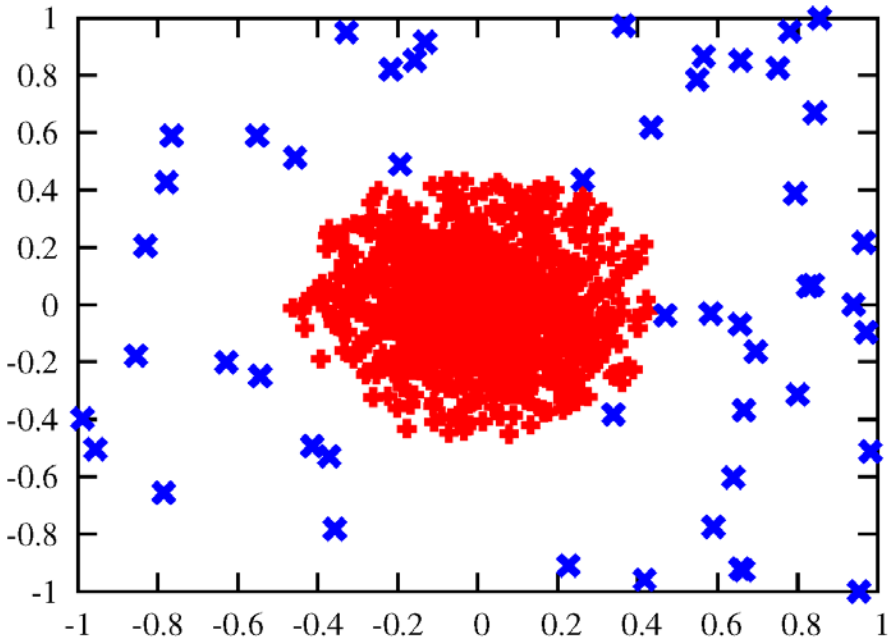
"an observation which deviates so much from the other observations as to arouse suspicions that it was generated by a different mechanism"

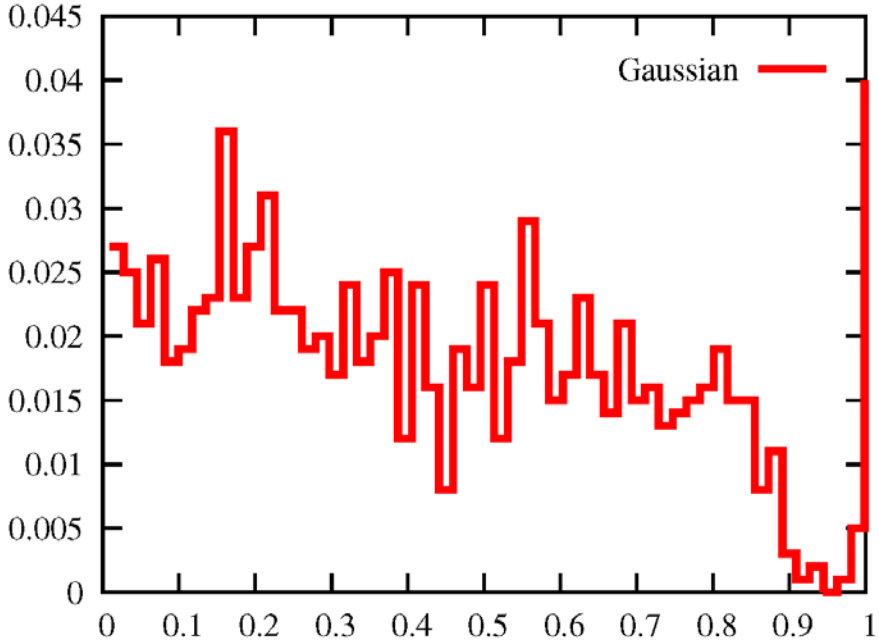
(Hawkins 1980)

Überprüfung (bzw. Entscheidung) erfordert Fachwissen über die Datengrundlage

7.1 Bewertung von Clustern und Outliern

Outlier Scores:

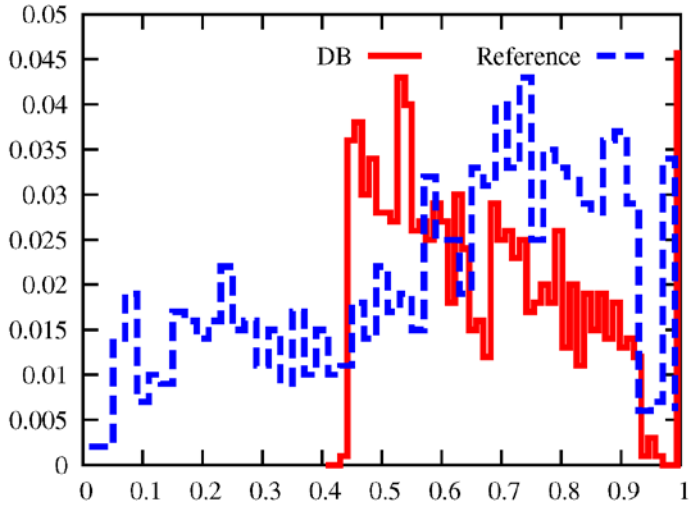


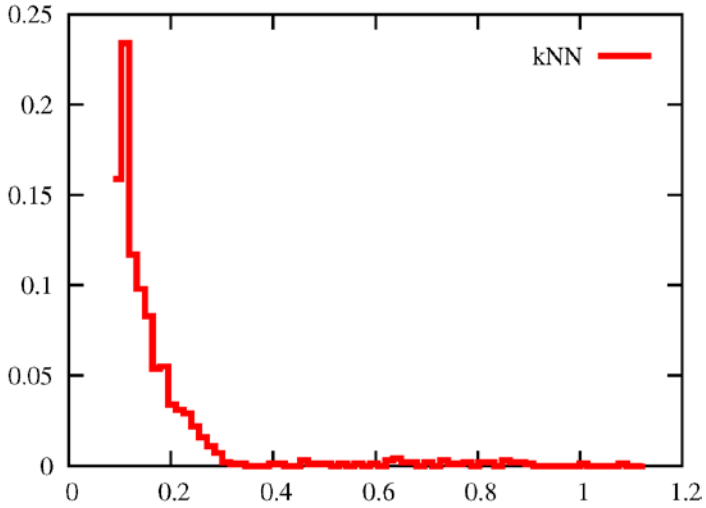


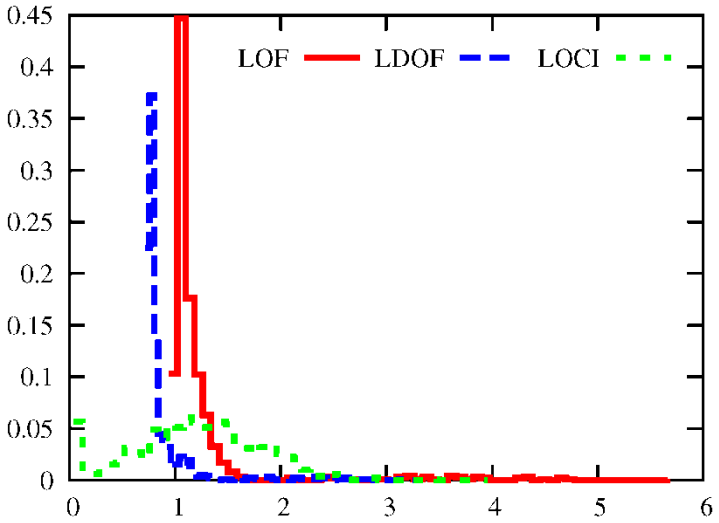
Hawkins (1980): "a sample containing outliers would show up such characteristics as large gaps between 'outlying' and 'inlying' observations and the deviation between outliers and the group of inliers, as measured on some suitably standardized scale"

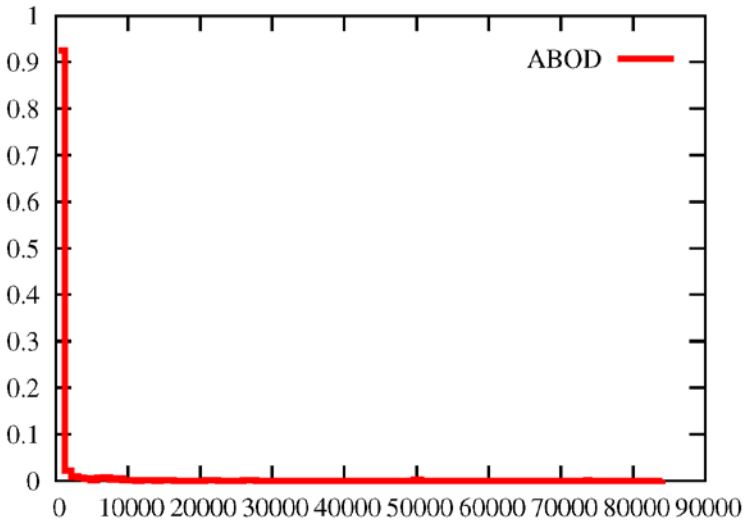
oft gutes Ranking, aber kein "*large gap*": keine klare Entscheidung Outlier/Inlier

7.1 Bewertung von Clustern und Outliern









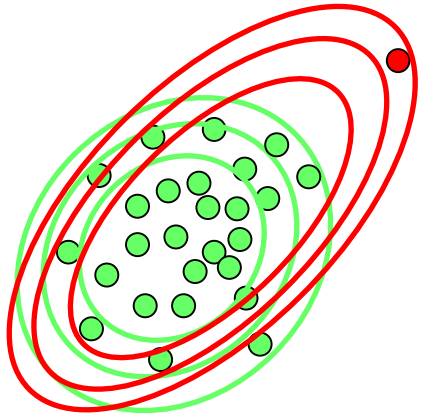
7.1 Bewertung von Clustern und Outliern

false positives und *false negatives* bei Outliern:

masking and *swamping* Effekt:

Outlier, die bei Modellbildung ja berücksichtigt werden, beeinflussen das Modell

- *masking*: das Modell wird so stark beeinflusst, dass der Outlier vom Modell erklärt wird, d.h., er wird maskiert
- *swamping*: durch die Modell-Verzerrung werden Inlier vom Modell nicht mehr gut erklärt, geraten in den Verdacht, Outlier zu sein

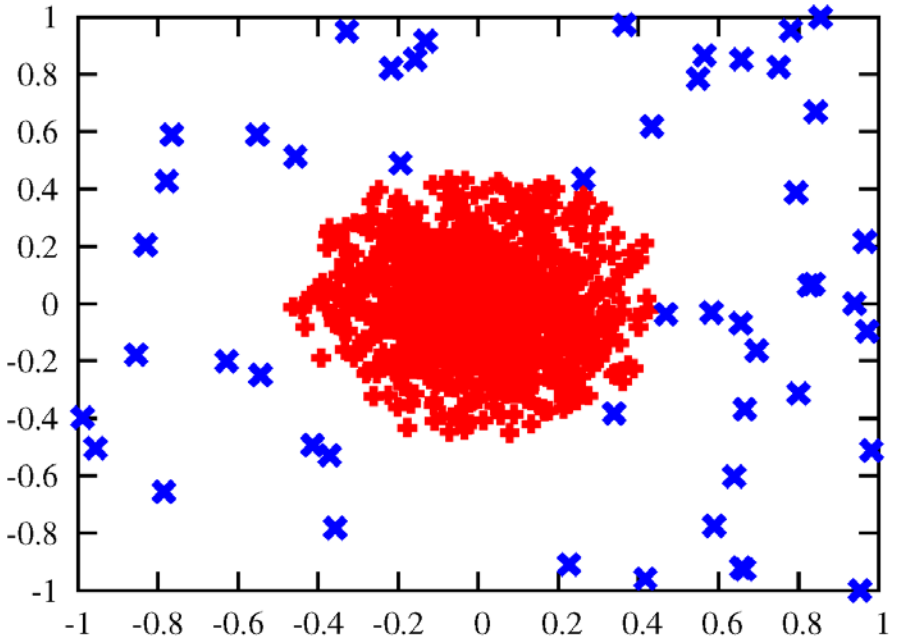


7.1 Bewertung von Clustern und Outliern

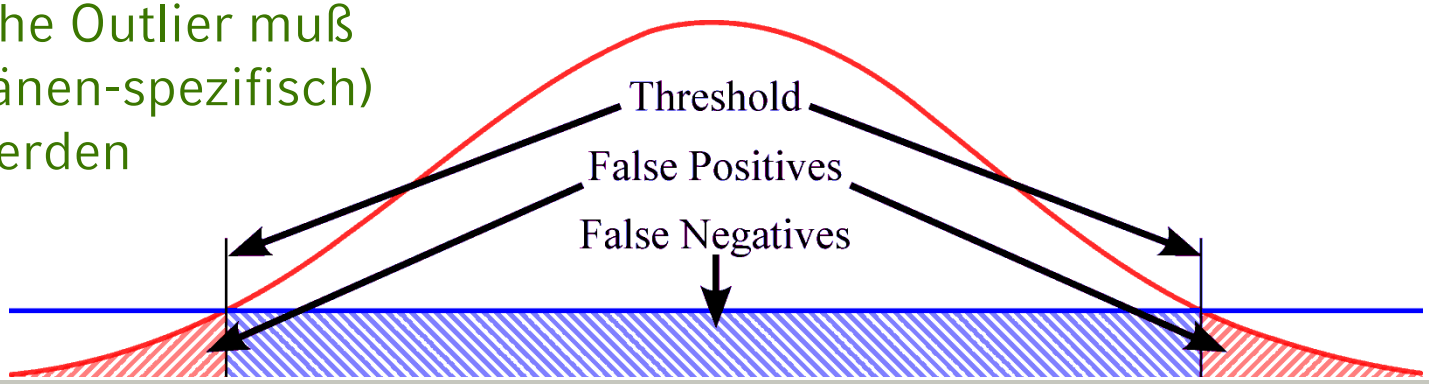
notwendige Niveaus von

false positives und *false negatives*:

- tatsächlich fremderzeugte Objekte bleiben unentdeckt, weil sie sehr gut zur Verteilung der normalen Objekte passen
- normale Objekte im Tail der "normalen" Verteilung müssen als Outlier erscheinen



→ über tatsächliche Outlier muß manuell (Domänen-spezifisch) entschieden werden



7.2 Evaluation von Cluster-Verfahren

Alternativen der Evaluation:

- "internal evaluation" (≈ unsupervised)
  - innere Sinnhaftigkeit (wie gut erklärt das gefundene Modell die Daten?)
  - Kohäsion/Separierung (Beispiele: TD^2, Silhouetten-Koeffizient)
  - Ähnlichkeitsmatrix (Korrelation, Visualisierung)
  - Voraussetzung: das Verfahren ist dem Problem grundsätzlich angemessen

7.2 Evaluation von Cluster-Verfahren

Alternativen der Evaluation:

- "external evaluation" (≈ supervised)
  - Überprüfung der Ergebnisse unabhängig vom Verfahren auf Daten mit bekannten Eigenschaften
  - Beispiel: Wiederentdecken von bekannten Klassen
  - Anwendung auf Daten aus dem gleichen Problembereich ist dann wahrscheinlich ähnlich sinnvoll
  - Problem: Entdeckung neuen Wissens wird bestraft

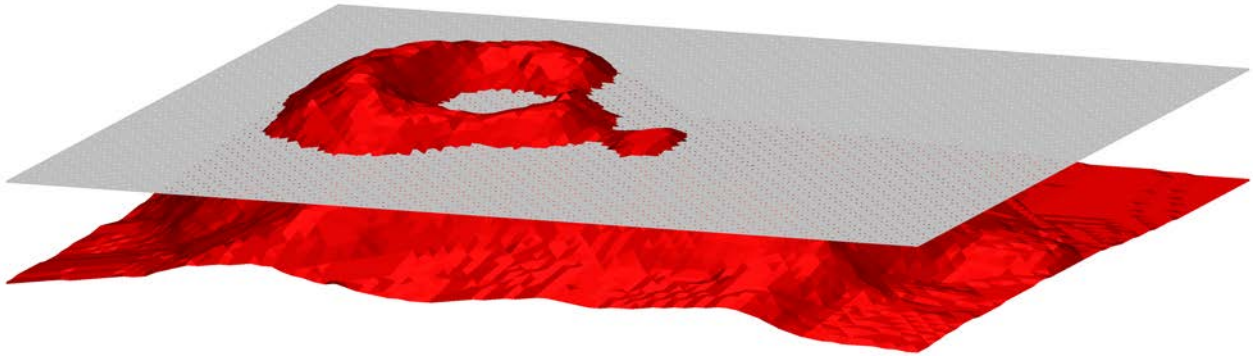
7.2 Evaluation von Cluster-Verfahren

Internal Evaluation – Grundproblem:

Angemessenheit des Algorithmus

Entscheidung *vor* der Anwendung nach grundlegenden Eigenschaften des Algorithmus und erwarteten Charakteristiken der Daten und Cluster:

- Typ des Clustering
  - z.B. biologische Taxonomie oder geologische Topographie: hierarchisch oder dichte-basiert
  - Clustering für Komprimierung: partitionierend



7.2 Evaluation von Cluster-Verfahren

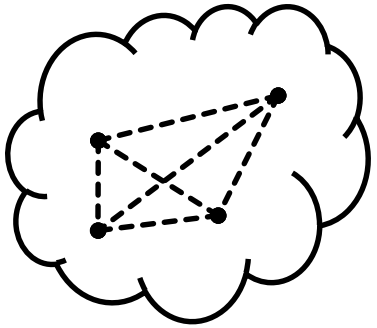
Angemessenheit des Algorithmus

- Charakteristiken des Datensatzes/der Attribute
  - z.B. k-means etc.: Mittelwert und Varianz müssen für die Daten sinnvoll berechnet und interpretiert werden können
  - andere (z.B. hierarchische Verfahren): die Natur der Daten ist weniger wichtig, solange eine Ähnlichkeitsmatrix erzeugt werden kann
- Noise/Outlier
  - EM/k-means: möglicherweise starke Modellverzerrung durch Ausreißer
  - dichte-basiert: stärkere Robustheit gegenüber Ausreißern

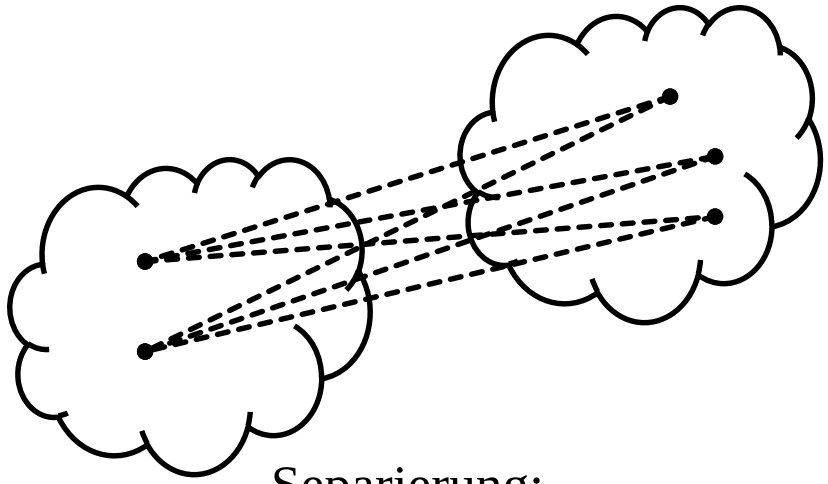
7.2 Evaluation von Cluster-Verfahren

Kohäsion und Separierung:

- Kohäsion: Wie stark hängt der Cluster zusammen?
- Separierung: Wie gut ist der Cluster von anderen Clustern getrennt?



Kohäsion:
kleine Distanzen innerhalb des Clusters



Separierung:
große Distanzen zwischen Clustern

- Validitätsmaß: geeignete Kombination von Kohäsion und Separierung

7.2 Evaluation von Cluster-Verfahren

Validitätsmaß für eine Menge von k Clustern, $C_1,\ldots,C_k$, Gewichtung w_i des Clusters C_i:

$$GesamtValidität = \sum_{i=1}^{k} w_i \cdot Validität(C_i)$$

Gewichtung z.B. nach Cluster-Größe

Mit einem gegebenen Maß der Nähe (proximity), z.B. Distanzfunktion, Ähnlichkeitsfunktion, ausgedrückt:

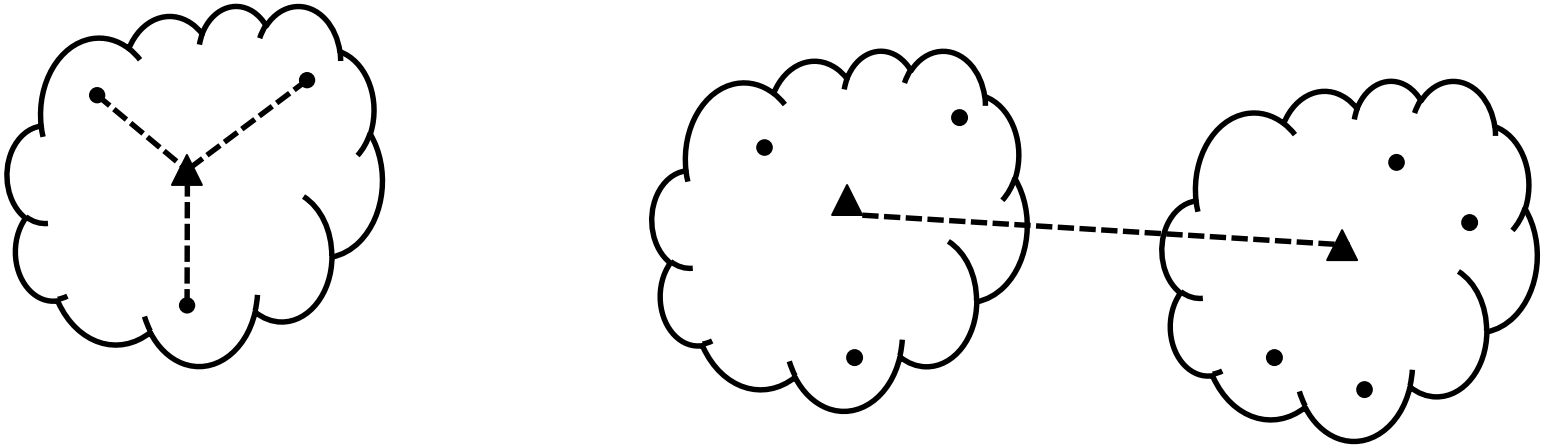
$$cohesion(C_i) = \sum_{\substack{x \in C_i \\ y \in C_i}} proximity(x, y)$$

$$separation(C_i, C_j) = \sum_{\substack{x \in C_i \\ y \in C_j}} proximity(x, y)$$

Vereinfachung für Prototypen ▲ c_i für Cluster C_i:

$$cohesion(C_i) = \sum_{x \in C_i} proximity(x, c_i)$$

$$separation(C_i, C_j) = proximity(c_i, c_j)$$



Beispiel: Kompaktheit (siehe Kap. 3)

- *Centroid* μ_C: Mittelwert aller Punkte im Cluster *C*
- *Maß für die Kosten* (Kompaktheit) *eines Clusters C*

$$TD^2(C) = \sum_{p \in C} dist(p, \mu_C)^2$$

- *Maß für die Kosten* (Kompaktheit) *eines Clustering*

$$TD^2 = \sum_{i=1}^{k} TD^2(C_i)$$

Beispiel: Silhouetten-Koeffizient (siehe Kap. 3)

Problem bei vielen Maßen: Abhängigkeit von der Anzahl der Cluster

- bei *k*-means und *k*-medoid: TD^2 und TD sinken monoton mit steigendem *k*
- bei EM: *E* steigt monoton mit steigendem *k*
- Silhouetten-Koeffizient
  - *a*(*o*): Abstand eines Objekts *o* zum Repräsentanten seines Clusters
  - *b*(*o*): Abstand zum Repräsentanten des „zweitnächsten" Clusters
  - Silhouette *s*(*o*) von *o*:

$$s(o) = \frac{b(o) - a(o)}{\max\{a(o), b(o)\}}$$

$-1 \leq s(o) \leq +1$

$s(o) \approx -1 / 0 / +1$: *schlecht / indifferent / gute Zuordnung*

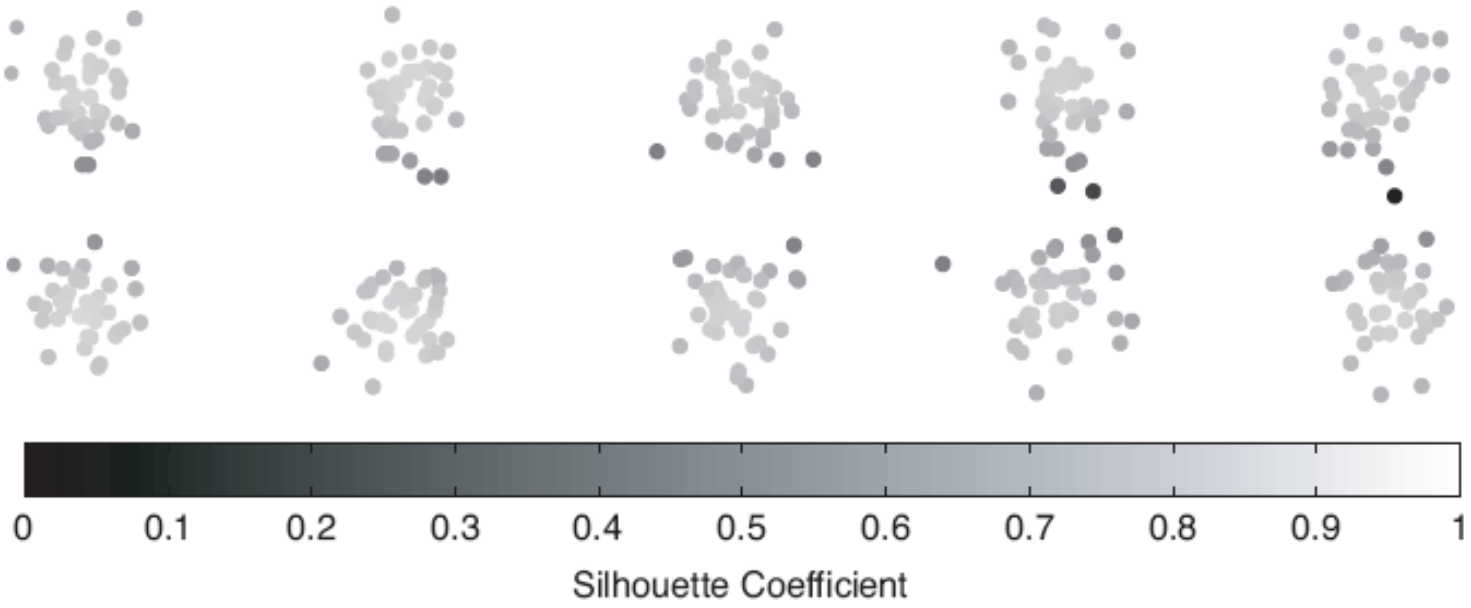
  - Silhouettenkoeffizient s_C eines Clusterings: durchschnittliche Silhouette aller Objekte
  - Interpretation des Silhouettenkoeffizients

$s_C > 0{,}7$: starke Struktur

$s_C > 0{,}5$: brauchbare Struktur

7.2 Evaluation von Cluster-Verfahren

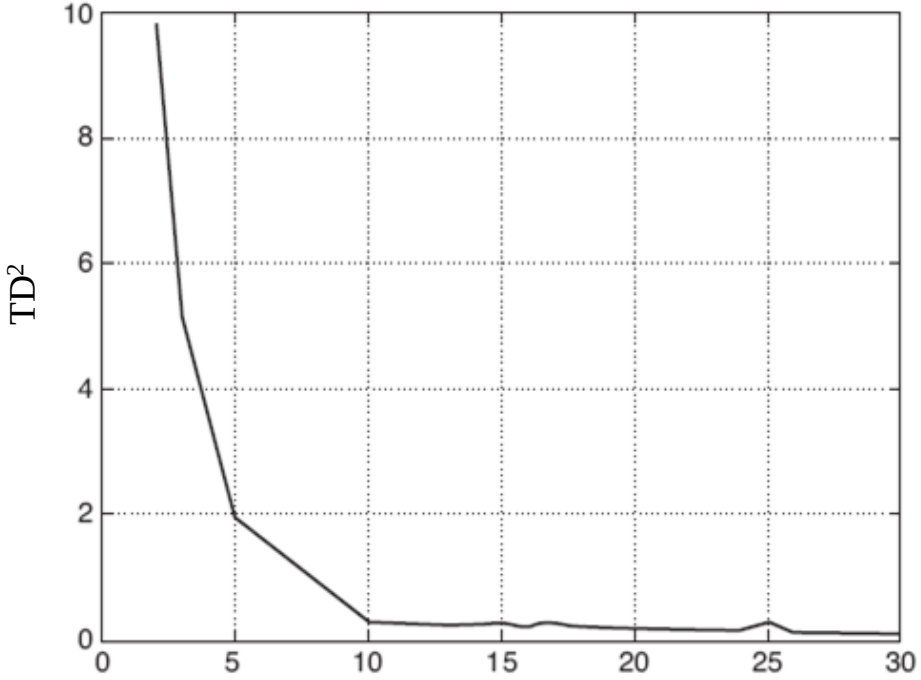
Silhouetten-Koeffizient für Punkte in zehn Clustern

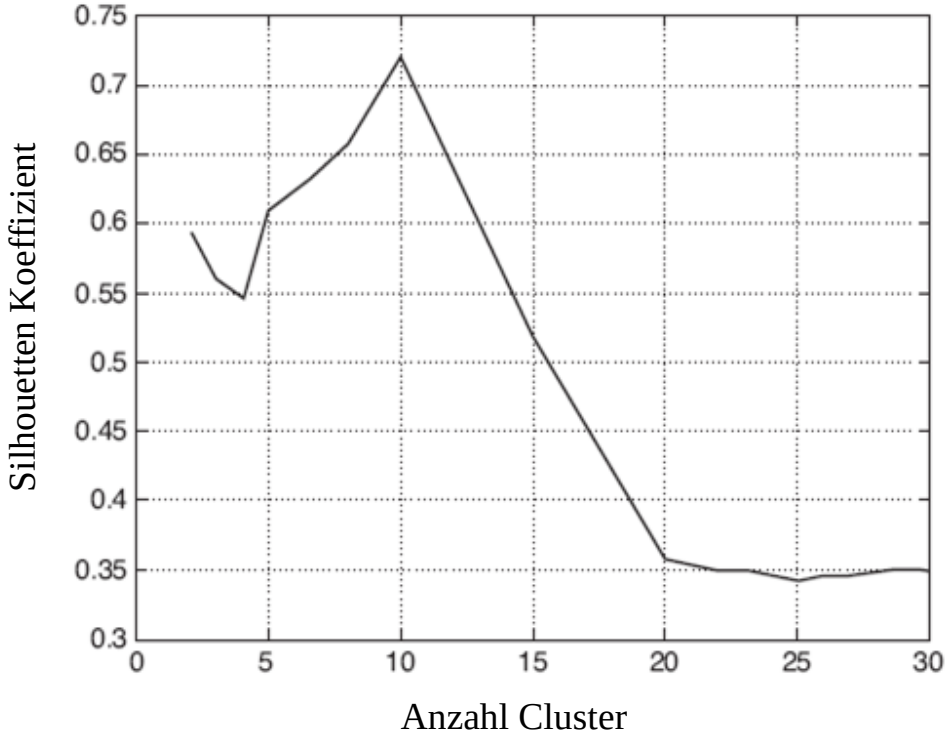


aus: Tan, Steinbach, Kumar: Introduction to Data Mining (Pearson, 2006)

7.2 Evaluation von Cluster-Verfahren

Vergleich TD^2 – durchschn. Silhouetten-Koeffizient für diesen Datensatz





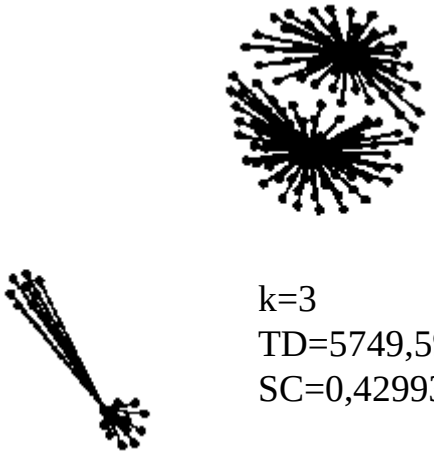
nach: Tan, Steinbach, Kumar: Introduction to Data Mining (Pearson, 2006)

7.2 Evaluation von Cluster-Verfahren

TD, Silhouetten-Koeffizient: Beispiele

k=2
TD=6472,75
SC=0,506355

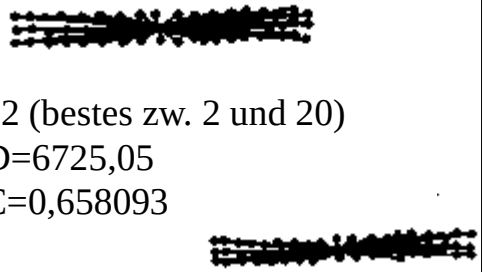
k=3
TD=5117,11
SC=0,364401



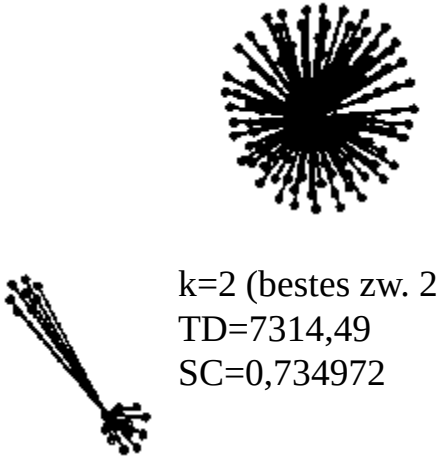
k=3
TD=5749,59
SC=0,429931

k=4
TD=3455,42
SC=0,552083

k=9 (bestes zwischen 2 und 20)
TD=2613,09
SC=0,395389



k=2 (bestes zw. 2 und 20)
TD=6725,05
SC=0,658093

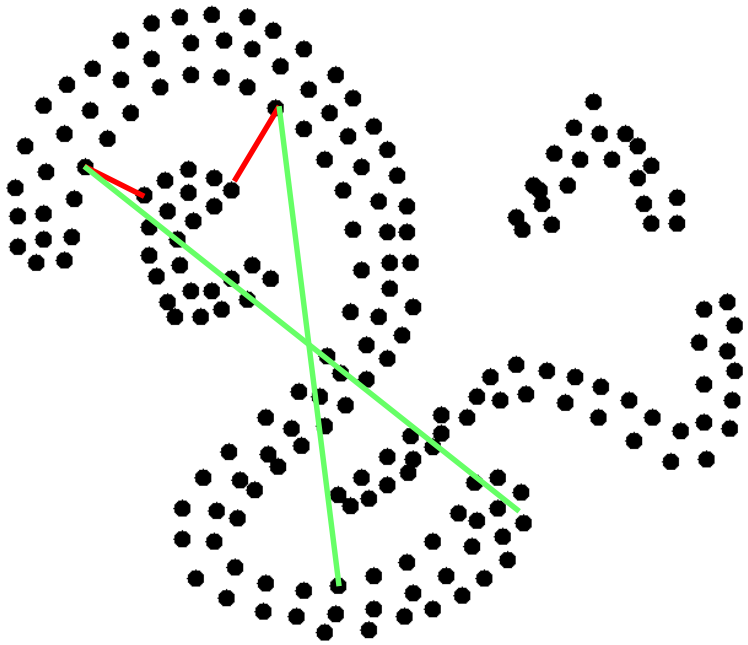


k=2 (bestes zw. 2 und 20)
TD=7314,49
SC=0,734972

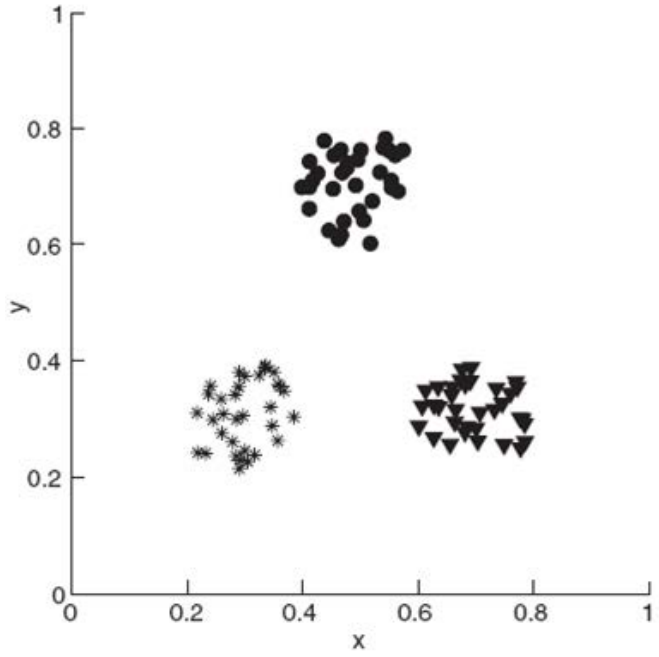
7.2 Evaluation von Cluster-Verfahren

Kohäsion und Separierung:

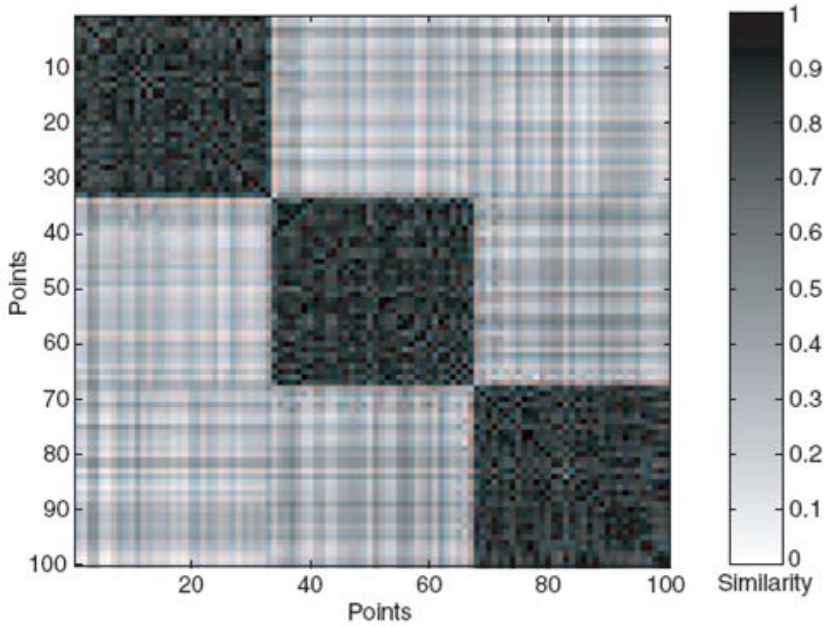
- Geeignet für globuläre Cluster, nicht geeignet für langgestreckte Cluster



Bewertung der Ähnlichkeitsmatrix



Datensatz
(gut separierbare Cluster)

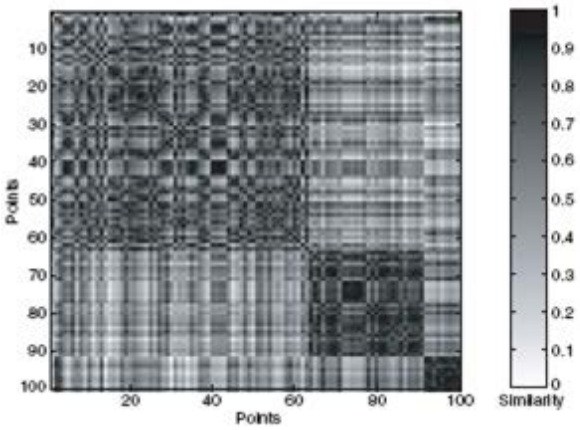


Ähnlichkeitsmatrix
(sortiert nach k-means Cluster-Labeln)

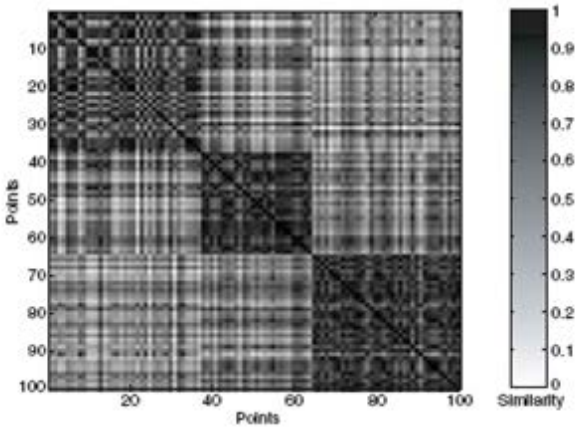
nach: Tan, Steinbach, Kumar: Introduction to Data Mining (Pearson, 2006)

7.2 Evaluation von Cluster-Verfahren

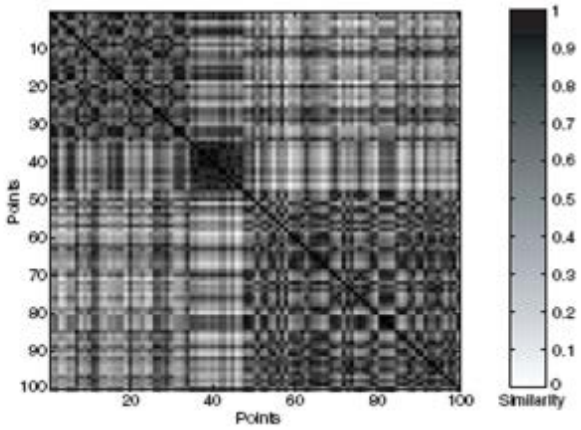
Ähnlichkeitsmatrizen für anderen Datensatz



DBSCAN



k-means



complete link

nach: Tan, Steinbach, Kumar: Introduction to Data Mining (Pearson, 2006)

7.2 Evaluation von Cluster-Verfahren

Internal Evaluation – Probleme:

- Angemessenheit des Clusterverfahrens für die gegebenen Daten
- Determinismus?
- Bestimmung von k?
- Vergleich verschiedener Verfahren gegeneinander?
- Zusammenhang: Zielfunktion des Cluster-Verfahrens – Bewertungsfunktion des Evaluations-Verfahrens

7.2 Evaluation von Cluster-Verfahren

External evaluation:

- Abbildung zwischen Clustering-Ergebnis und vorgegebenen Clustern (=Klassen) (sog. *ground truth* oder *gold standard*)
  - warum nicht einfach *confusion matrix*?
  - zwei grundlegende Ansätze:
    - Mapping von Objektmengen
    - Vergleich von Objektpaaren ("pair counting")
- Bewertung der Entsprechung zwischen vorgegebenen und gefundenen Teilmengen
  - bei Mapping von Objektmengen: informationstheoretische Maße
  - bei Vergleich von Objektpaaren: viele Maße, z.B. auch klassifikationstypische Maße (F-measure etc.)

7.2 Evaluation von Cluster-Verfahren

Mapping von Objekt-Mengen:

- N Objekte
- Clustering U mit R Clustern $U_1,\ldots,U_R$
- Clustering V mit C Clustern $V_1,\ldots,V_C$
- n_{ij}: Anzahl der Elemente in $U_i \cap V_j$
- *C×R contingency table*:

Informations-verlust?

$$\left[n_{ij}\right]_{j=1\ldots C}^{i=1\ldots R}$$

$U \setminus V$	V_1	V_2	$\ldots$	V_C	Summen
U_1	n_{11}	n_{12}	$\cdots$	n_{1C}	a_1
U_2	n_{21}	n_{22}	$\cdots$	n_{2C}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
U_R	n_{R1}	n_{R2}	$\cdots$	n_{RC}	a_R
Summen	b_1	b_2	$\cdots$	b_C	$\sum_{ij} n_{ij} = N$

?

7.2 Evaluation von Cluster-Verfahren

Mengen-Mapping (Beispiel):

- Datensatz *D*: {1,2,3,4,5,6}
- Clustering *U*: {1,2,3}, {4,5}, {6}
- Clustering *V*: {1,2,4}, {3,5,6}

$U \setminus V$	{1,2,4}	{3,5,6}	Summen
{1,2,3}	2	1	3
{4,5}	1	1	2
{6}	0	1	1
Summen	3	3	6

7.2 Evaluation von Cluster-Verfahren

Pair-Counting:

- N Objekte
- Clustering U mit R Clustern $U_1,\ldots,U_R$
- Clustering V mit C Clustern $V_1,\ldots,V_C$
  - a: Anzahl Objektpaare, die in U und V im selben Cluster sind
  - b: Anzahl Objektpaare, die in U im selben, aber in V in verschiedenen Clustern sind
  - c: Anzahl Objektpaare, die in U in verschiedenen aber in V im selben Cluster sind
  - d: Anzahl Objektpaare, die in U und in V zu verschiedenen Clustern gehören
- 2×2 *contingency table*:

$U \setminus V$	Paare im selben Cluster	Paare in versch. Clustern
Paare im selben Cluster	a	b
Paare in versch. Clustern	c	d

- Informationsverlust? Disjunktivität der Cluster?

$$a+b+c+d=\frac{N(N-1)}{2}$$

Pair-Counting (Beispiel):

- Datensatz D: {1, 2, 3, 4, 5, 6}
- Clustering U: {1, 2, 3}, {4, 5}, {6}
- Clustering V: {1, 2, 4}, {3, 5, 6}
- Paare in D: {(1,2), (1,3), (1,4), (1,5), (1,6), (2,3), (2,4), (2,5), (2,6), (3,4), (3,5), (3,6), (4,5), (4,6), (5,6)}
- Paare in U: {(1,2), (1,3), (2,3), (4,5)}
- Paare in V: {(1,2), (1,4), (2,4), (3,5), (3,6), (5,6)}
- a = |Paare $U \cap$ Paare V| = |{(1,2)}| = 1
- b = |Paare $U \setminus$ Paare V| = |{(1,3), (2,3), (4,5)}| = 3
- c = |Paare $V \setminus$ Paare U| = |{(1,4), (2,4), (3,5), (3,6), (5,6)}| = 5
- d = |Paare $D \setminus$ (Paare $U \cup$ Paare V)| = |{(1,5), (1,6), (2,5), (2,6), (3,4), (4,6)}| = 6

$U \setminus V$	Paare in V	nicht Paare in V
Paare in U	1	3
nicht Paare in U	5	6

7.2 Evaluation von Cluster-Verfahren

Bewertung eines Mengen-Mappings

- Precision/Recall?
  - Richtung der Abbildung?
  - Abdeckung?
- Entropie
- Mutual Information
- Normalized Mutual Information
- …

$U \setminus V$	V_1	V_2	V_3
U_1	10	0	0
U_2	12	1	3
U_3	8	5	7
U_4	25	8	8
U_5	15	7	7
U_6	20	0	0

7.2 Evaluation von Cluster-Verfahren

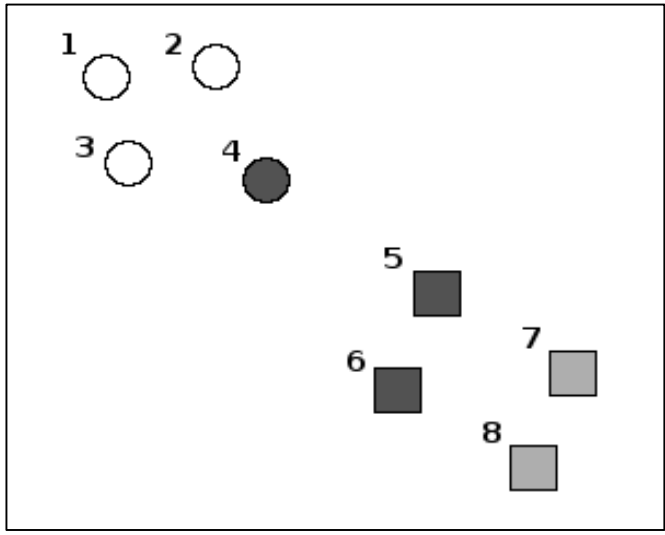
Bewertung einer Pair-Counting-Matrix

- Precision, Recall, F-measure: wie Klassifikation
- Rand-Index
- Adjusted Rand Index (Hubert&Arabie)
- Jaccard-Index
- ...

7.2 Evaluation von Cluster-Verfahren

Rand Index:

$$RI = \frac{a+d}{a+b+c+d}$$



a: #Paare in gleicher Klasse und gleichem Cluster

b: #Paare in gleicher Klasse aber versch. Clustern

c: #Paare in versch. Klassen aber gleichem Cluster

d: #Paare in versch. Klassen und versch. Cluster

Beispiel:

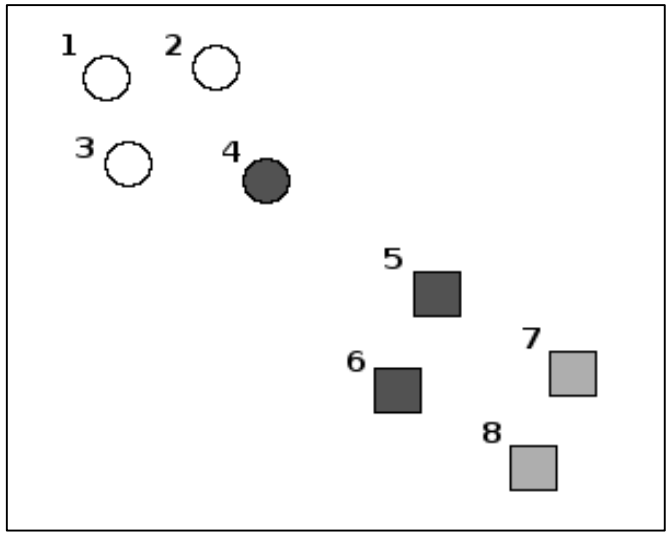
- 2 Klassen (Kreise und Quadrate)
- 3 Cluster (Schwarz, Weiß, Grau)

a = 5; b = 7; c = 2; d = 14

RI = 5+14/(5+7+2+14) = 0.6785

Jaccard Coefficient

$$Jc = \frac{a}{a+b+c}$$



a: #Paare in gleicher Klasse und gleichem Cluster

b: #Paare in gleicher Klasse aber versch. Clustern

c: #Paare in versch. Klassen aber gleichem Cluster

d: #Paare in versch. Klassen und versch. Cluster

Beispiel:

- 2 Klassen (Kreise und Quadrate)
- 3 Cluster (Schwarz, Weiß, Grau)

a = 5; b = 7; c = 2

Jc = 5/(5+7+2) = 0.3571

Adjusted Rand Index

Rand und Jaccard berücksichtigen nicht, welche Qualität durch zufällige Lösungen bereits erreicht werden kann.

Erwartungswert ist nicht 0, wenn zwei zufällige Partitionen verglichen werden

Adjustment for chance:

$$\text{Adjusted\_Criterion} = \frac{\text{Criterion} - E\{\text{Criterion}\}}{\text{Max\_Criterion} - E\{\text{Criterion}\}}$$

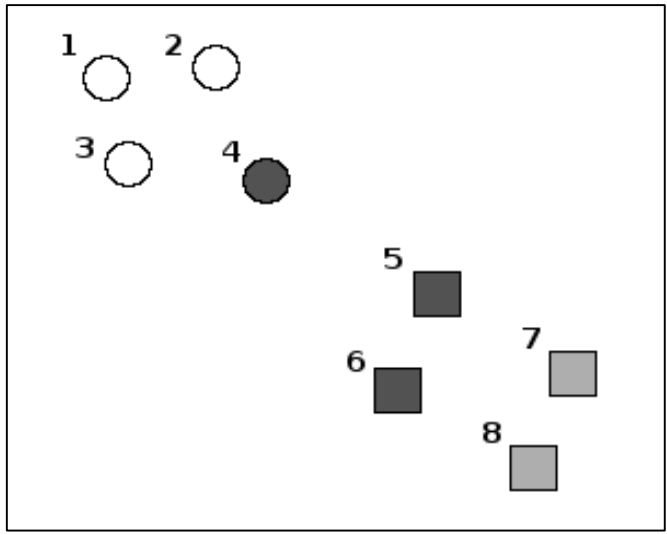
- Hubert & Arabie (1985) bestimmten analytisch den Erwartungswert für den Rand Index und schlugen den **Adjusted Rand Index** (ARI) vor

Adjusted Rand Index

ARI kann geschrieben werden als:

$$ARI = \frac{a - \frac{(a+c)(a+b)}{M}}{\frac{(a+c)+(a+b)}{2} - \frac{(a+c)(a+b)}{M}}$$

mit M = a + b + c + d



Beispiel:

- 2 Klassen (Kreise und Quadrate)
- 3 Cluster (Schwarz, Weiß, Grau)

a = 5; b = 7; c = 2; d = 14; M = 28

ARI = $\frac{5-\frac{7*9}{28}}{\frac{7+12}{2}-\frac{7*9}{28}}$ = **0.3793**

7.2 Evaluation von Cluster-Verfahren

Was sagt das Wiederauffinden der Klassen aus?

- Ein Datensatz kann mehrere, unterschiedliche Konzeptebenen haben – welche Klassen werden wiedergefunden? *(Färber et al. 2010)*
- Beispiel: Amsterdam Library of Object Images (ALOI)
- 1000 Objekte
- gleiches Objekt = gleiche Klasse

7.2 Evaluation von Cluster-Verfahren

Feature für jedes Objekt:

- unterschiedliche Beleuchtungswinkel
- unterschiedliche Beleuchtungsfarben
- unterschiedliche Sichtwinkel (Drehwinkel des Objekts)

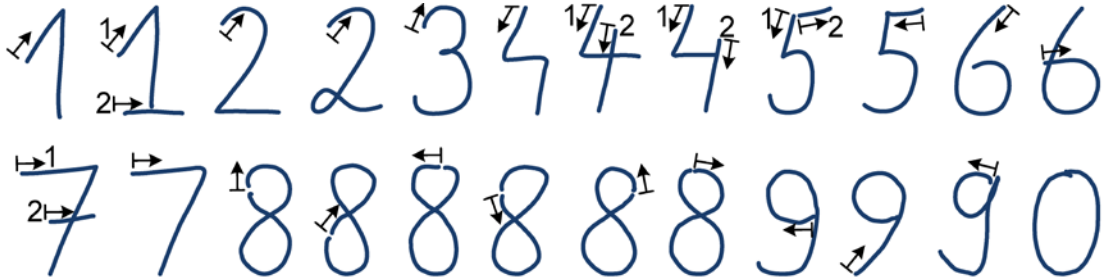
7.2 Evaluation von Cluster-Verfahren

mögliche Konzepte:

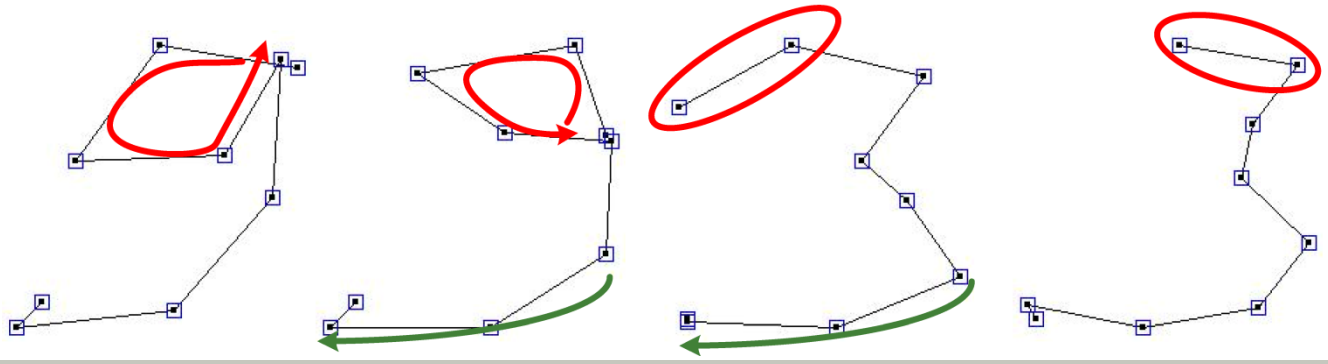
- gleiches Objekt in versch. Farbe
  (Form, Objekt-Art)
- verschiedene, ähnliche Objekte in gleicher Ansicht
- dominante Farbe und/oder Form
- ...

Beispiel Pendigits-Datensatz: handgeschriebene Ziffern

- Klassen: Ziffern 0-9
- Features: einzelne (Zeit-)Punkte im Lauf des Schriftzugs
- sinnvolle Konzepte unterscheiden sich von den vorgegebenen Klassen:
  - eine Ziffer zerfällt in Untergruppen unterschiedlicher Schreibweisen



  - verschiedene Ziffern haben große Gemeinsamkeiten



7.3 Evaluation von Outlier-Detection-Algorithmen

Bewertung von Outlier-Verfahren:

- Outlier vs. Inlier – ein Klassifikationsproblem?
- Class Imbalance: Die "Klasse" Outlier ist wesentlich kleiner, aber oft wichtiger als die "Klasse" Inlier – Probleme für:
  - Training
  - Bewertung
- Viele Outlier-Verfahren liefern keine Klassenentscheidung, sondern Outlier-*Scores* oder -*Factors* – ermöglicht ein Ranking der Objekte.
- Übliches Evaluationsschema für gerankte Ergebnisse: "*Receiver Operating Characteristic*" (ROC)

7.3 Evaluation von Outlier-Detection-Algorithmen

Receiver Operating Characteristic (ROC)

- Zwei-Klassen-Confusion-Matrix:
  - "Outlier" ist die zu entdeckende Klasse
  - mögliche Klassifikationsergebnisse
    - Outlier wird als Outlier erkannt: *true positive* (TP)
    - Outlier wird als Inlier klassifiziert: *false negative* (FN)
    - Inlier wird als Outlier klassifiziert: *false positive* (FP)
    - Inlier wird als Inlier erkannt: *true negative* (TN)

$C(o) \setminus K(o)$	Outlier	Inlier
Outlier	TP	FN
Inlier	FP	TN

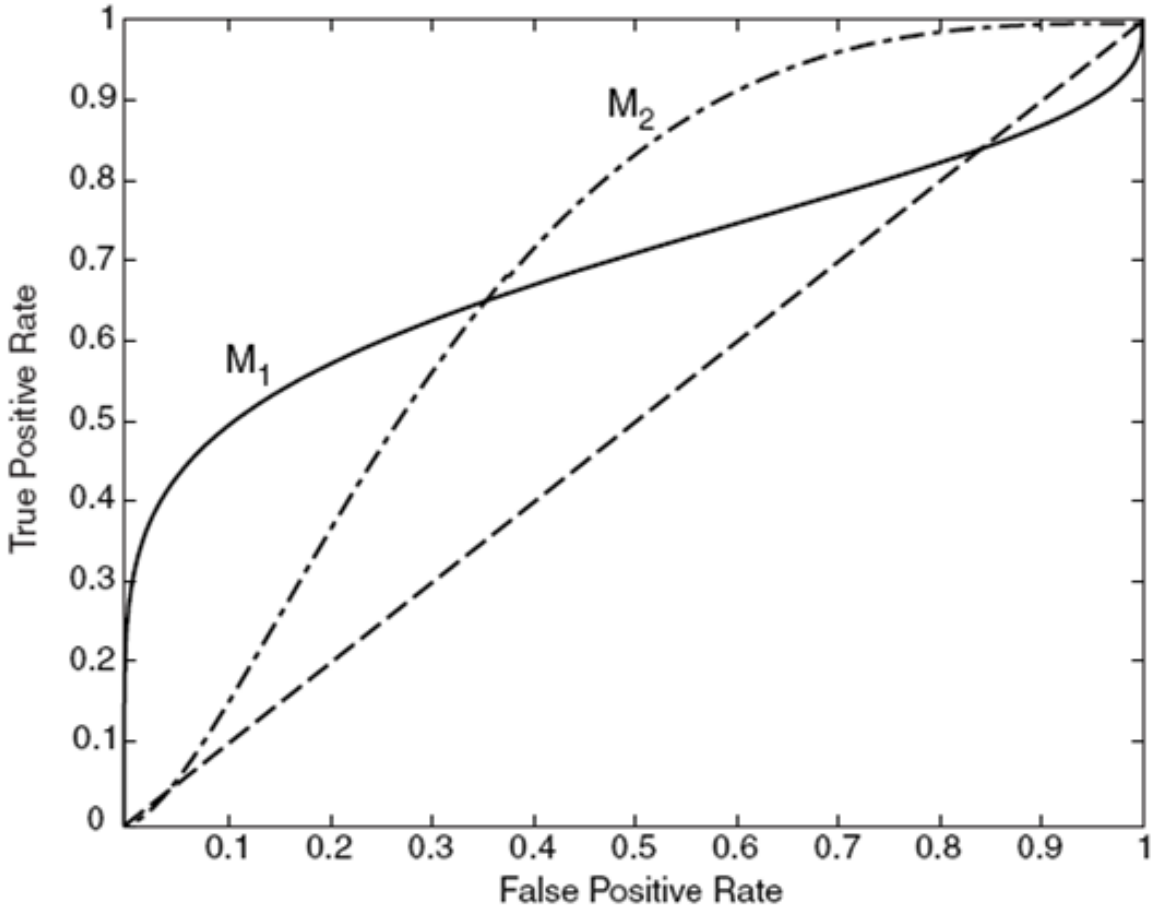
- Ranking: Kontinuum zwischen stärkster Outlier und schwächster Outlier (= stärkster Inlier)
- Bewerte Reihenfolge: TP sollten möglichst vor FP kommen
- ROC: Graphische Darstellung von TP-Rate vs. FP-Rate

Receiver Operating Characteristic (ROC)

- jedes TP im Ranking: Schritt nach oben
- jedes FP im Ranking: Schritt nach rechts
- zufälliges Ranking?
- Vergleich zweier Methoden: Fläche unter der ROC Kurve

  (ROC AUC)

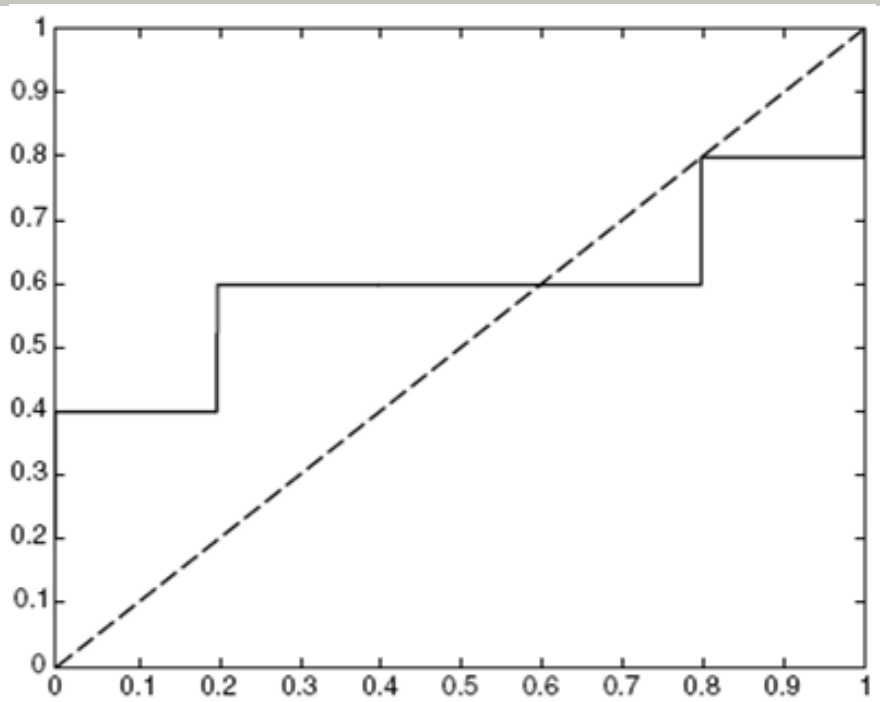
  $(0 \leq \text{ROC AUC} \leq 1)$



7.3 Evaluation von Outlier-Detection-Algorithmen

Beispiel:

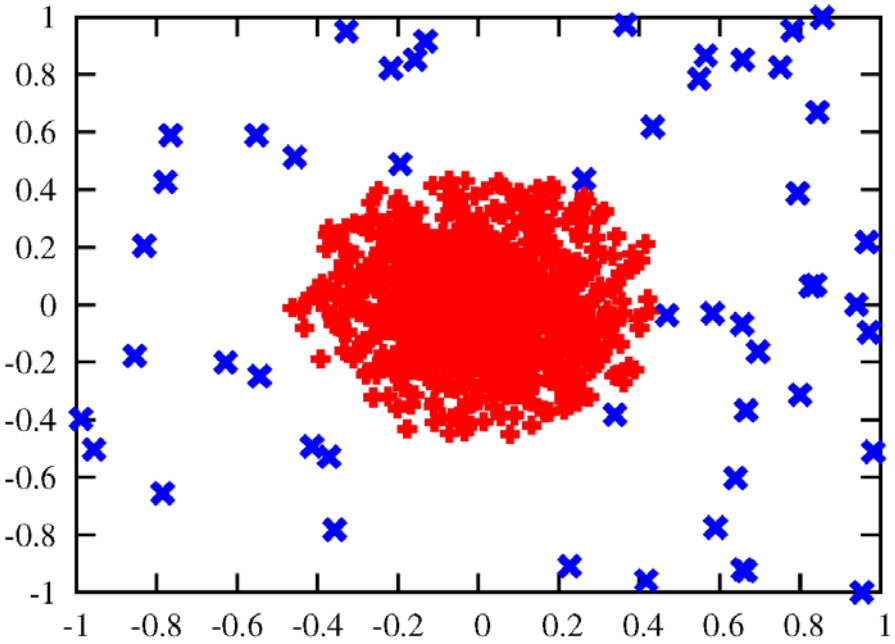
- 10 Objekte
- Klassen +/-
- Ranking:
  Klassenwahrscheinlichkeit +

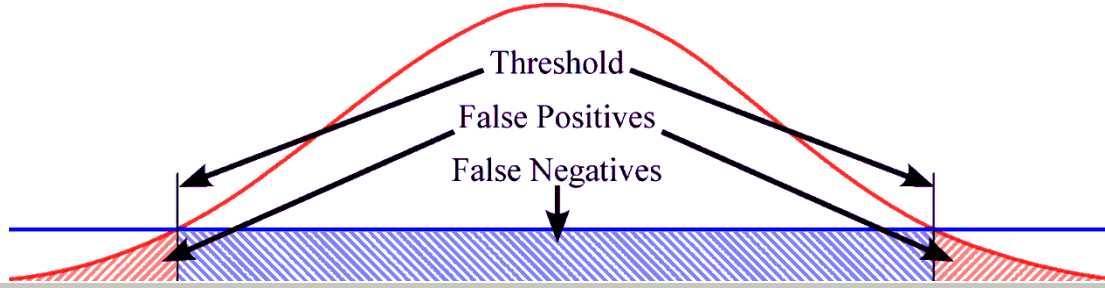


Class	+	−	+	−	−	−	+	−	+	+	
	0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
TP	5	4	4	3	3	3	3	2	2	1	0
FP	5	5	4	4	3	2	1	1	0	0	0
TN	0	0	1	1	2	3	4	4	5	5	5
FN	0	1	1	2	2	2	2	3	3	4	5
TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

Daten zur Evaluation von Outlier-Verfahren

- wie werden Outlier, die ein Verfahren finden soll, in Testdaten definiert?
- synthetische Daten
  - Design von "normaler" Verteilung (oder mehreren) und ungewöhnlicher, kleinerer Verteilung
  - auch nach sorgfältigem Design der Testdaten bleibt ein Mindestmaß an FP und FN





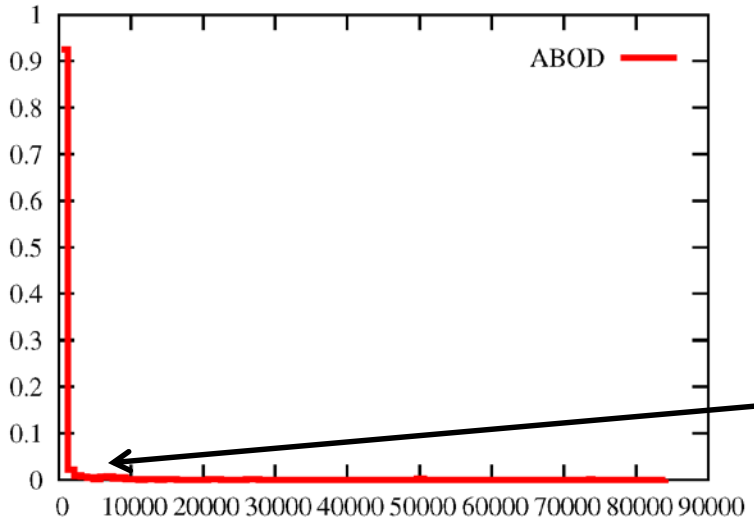
7.3 Evaluation von Outlier-Detection-Algorithmen

Daten zur Evaluation von Outlier-Verfahren

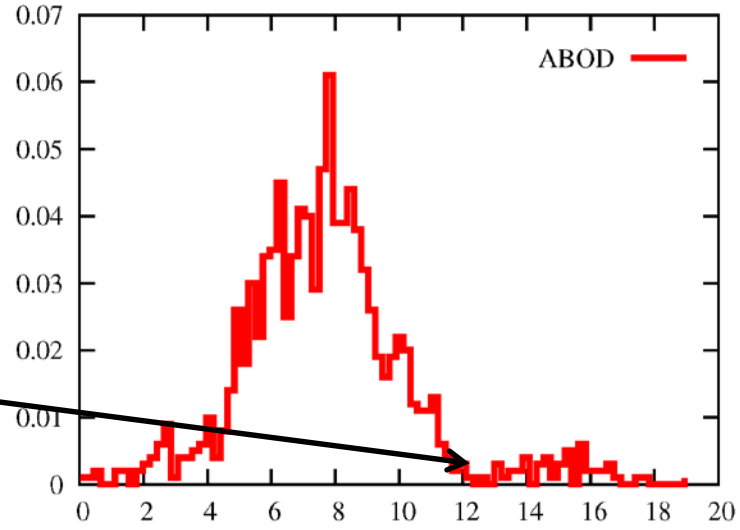
- wie werden Outlier, die ein Verfahren finden soll, in Testdaten definiert?
- reale Daten
  - kaum vordefinierte Outlier-Daten verfügbar
  - Klassifikationsprobleme, down-sampling einer Klasse als Outlier
  - die tatsächlichen Charakteristiken sind unbekannt, ein Mindestmaß an FP und FN kann ebenfalls nicht ausgeschlossen werden
- relativer Performanzvergleich verschiedener Verfahren?
  - verschiedene Verfahren finden verschiedene Outlier – Entsprechung zur Charakteristik der down-gesampelten Klasse oder Borderline-Punkten synthetischer Verteilungen oft unklar (insb. in hoch-dimensionalen Daten – keine visuelle Überprüfung möglich!)

Bewertung von Outlier-Verfahren:

- Outlier-Scores müssen sinnvoll interpretiert werden, um eine Entscheidung (Outlier/Inlier) zu treffen
- Ansatz: Abbildung auf "Outlier-Wahrscheinlichkeit" *(Kriegel et al. 2011)*
- Regularisierung/Normalisierung der Outlier-Scores auf 0…∞/0…1, möglichst Vergrößern des "Gap" zwischen Outliern und Inliern



Beispiel:
ABOD Scores
vor und *nach*
Regularisierung
–
Gap wird
prägnanter



7.3 Evaluation von Outlier-Detection-Algorithmen

- mögliche Auswertung mit Einbezug von Error-Kosten (übertragen aus Evaluationstechniken für Klassifikationsprobleme mit *imbalanced* Klassen):
  - Wie schlimm ist es, einen Outlier als Inlier zu klassifizieren (und umgekehrt)?
  - Kosten-Gewichtung durch Wahrscheinlichkeit, Outlier/Inlier zu sein

$$\text{Kosten} = \frac{1}{2}\sum_{x \in I} P(O \mid x) \cdot \frac{1}{|I|} + \frac{1}{2}\sum_{x \in O} P(I \mid x) \cdot \frac{1}{|O|}$$

Was haben Sie gelernt?

- Bewertung von Ergebnissen erfordert gründliche Analyse
- oft keine eindeutige, absolute Aussage möglich
- Vergleich von Ergebnissen relativ zueinander (besser/schlechter): erfordert Kriterium
  - Vergleich von Ergebnissen: Kriterium muss dem Problem angemessen sein
  - Vergleich von Verfahren: Kriterium darf nicht einzelne Verfahren systematisch bevorzugen
- interne vs. externe Evaluation
- interne Evaluations-Maße
  - Kohäsion, Separierung: Kompaktheit, Silhouette, Ähnlichkeitsmatrix
- externe Evaluations-Maße
  - mapping vs. pair counting
  - Precision, Recall, Rand-Index, Jaccard, ARI
  - Outlier detection: Receiver Operating Characteristic
- Problematik bei Verwendung von „ground truth"