Informationssysteme: Neuere Konzepte – Teil II

Kapitel 4: KDD & Data Mining

Folien teilweise übernommen von Matthias Gimbel

Gliederung

Diese Vorlesung gliedert sich wie folgt:

- Einführung: Klassische Fragestellungen
- Data-Mining-Aufgaben
- Data-Mining-Verfahren
- KDD-Prozess

Informationssysteme: Neuere Konzepte – Teil II

Kapitel 4: KDD & Data Mining
- Einführung -

Motivation #1

- Automatisierte Erfassung und Erzeugung von Daten nimmt stetig zu.
- Dauerhafte Speicherung von Daten wird immer günstiger.

➔ riesige Datenfriedhöfe in Wissenschaft und Wirtschaft

➔ manuelle Sichtung unmöglich

➔ Vielzahl von Informationen über Sachverhalte enthalten

➔ oft nicht einmal Ansatzpunkt für konkrete Fragestellung klar

Motivation #2

Gibt es eine Möglichkeit, in solchen riesigen Datenmengen interessante Zusammenhänge automatisiert zu entdecken und zu validieren?

- vorher:
  - Standardisierte Berichte für bekannte Fragen
  - Ad-hoc-Anfragen (SQL) für konkrete Hypothesen
- KDD
  - Neue Entwicklungen an der Schnittstelle zwischen
    - Maschinellem Lernen
    - Datenbanktechnik
    - Statistik

Fragestellungen

- Beispiel: Einzelhandel
  - oft gemeinsam gekaufte Produkte
  - treue Kunden, Premium-Kunden und Schnäppchen-Jäger
  - Spezifische Interessensgruppen
  - Erfolg einer Marketing-Aktion
  - Absatzchancen neuer Produktsegmente
  - Cross-Selling (Partnerschaft mit anderen Anbietern)
  - Bestandsplanung (Wann kaufen Kunden wieviel wovon?)

Fragestellungen

- Beispiel: Telekommunikationsbranche
  - Auffinden von Kundengruppen, Gemeinsamkeiten zwischen Kunden
    - Familien, Wenigtelefonierer, Vielsurfer
  - Design von Spezialtarifen, Aktionsangeboten
  - Finden illegaler Nutzer
  - Kundenbindung - Customer Relationship Management (CRM)
    - Customer lifetime value
    - Rentabilität von Verlängerungsangeboten

Fragestellungen

- Beispiel: Banken
  - Finden von Kriterien für die Kreditwürdigkeit von Kunden
  - Prognose von Aktienkursen
- Beispiel: Wissenschaft
  - Wirksamkeit von Medikamenten
  - Zusammenhang von Umwelteinflüssen und Krankheiten
  - Finden von Genen in DNA-Strängen

Fragestellungen

- Beispiel: Web (Clickstream Analysis)
  - Identifikation von Web-Transaktionen
  - Häufigkeit des Seitenbesuchs
  - Verweildauer auf einer Seite
  - Häufige Navigationspfade durch Web-Site
  - Welche Faktoren führen zu Abbruch?
  - Welche Navigationspfade führen zu erfolgreichen Abschlüssen?

Definitionen

- *Knowledge Discovery in Databases* (KDD) is the non-trivial process of identifying **valid**, **novel**, **potentially useful**, and **ultimatively understandable** patterns in data.“

  (Fayyad, Piatetsky-Shapiro, Smyth, 1996)

- *Data Mining*: verschiedene Ansichten ....
  - Prozess der Wissensgewinnung insgesamt
  - spezielle Stufe dieses Prozesses: Anwendung der eigentlichen Lernverfahren

Definitionen

- **valid**
  gültig, statistisch gesichert
- **novel**
  neuartig, nicht schon bekannt
- **potentially useful**
  tatsächlich verwertbar
- **ultimatively understandable**
  für den menschlichen Betrachter interpretierbar und verständlich

- Patterns
  Muster, Zusammenhänge, Wissen

Informationssysteme: Neuere Konzepte – Teil II

Kapitel 4: KDD & Data Mining
- Data-Mining-Aufgaben -

Data-Mining-Aufgaben

- Fragestellungen führen aus Data-Mining-Sicht zu verschiedenen Aufgaben (Tasks) mit verschiedenen Eigenschaften:
  - Format der Wissensrepräsentation (am wichtigsten)
    - Regeln
    - Formeln / Modelle
    - Beispielinstanzen (Repräsentanten für Gruppe)
    - Kategorien: akzeptieren, ablehnen
  - Lernart
    - überwacht: Ergebniskategorien sind vorgegeben
    - unüberwacht: Ergebniskategorien werden vom Vefahren bestimmt

Data-Mining-Aufgaben

- Format der Ein-/Ausgabewerte
  - numerisch
    - kontinuierlich: 38,4°C
    - diskret: Noten von 1-6
  - textuell
    - nominal (beliebige Anzahl Ausprägungen)
    - kategorisch (begrenzte Anzahl Ausprägungen)
      - (rot, grün, blau)
    - linear (begrenzte Anzahl Ausprägungen, Ordnung zwischen den Ausprägungen)
      - (kalt, lauwarm, warm, heiß)

Klassifikation

- Ziel
  - Zuordnung von Datensätzen zu vorgegebenen Klassen
- Ausgangspunkt
  - Menge von Datensätzen mit Erfahrungswerten (also bereits klassifiziert)
- Lernproblem
  - Suche Klassifikator für neue Daten
- Beispiele
  - Kreditwürdigkeit:
    If Alter < 30 and Einkommen < 15000: kein Kredit
  - Wettervorhersage:
    If Temp = hot and Humidity > 85%: Gewitter

Klassifikation

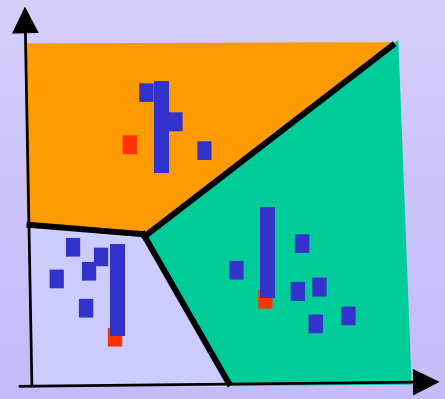
- Wissensrepräsentation
  - Regeln
- Eingabeformat
  - Attribute: beliebig
  - Klassen: diskret (Zahlen) oder Kategorien (Text)
- Lernart
  - überwacht (Klassen sind vorgegeben)

Segmentierung, Clustering

- Ziel
  - Einteilung der Daten in interessante und aussagekräftige Gruppen
- Ausgangspunkt
  - Menge von Datensätzen mit bestimmten Merkmalen
- Lernproblem
  - Gruppierung dieser Datensätze aufgrund ähnlicher Merkmale
- Beispiele
  - Identifikation von Kundengruppen
  - Zusammenfassen ähnlicher Fieberkurven

Segmentierung, Clustering

- Wissensrepräsentation
  - Beispielinstanzen oder abstrakte Kategorien (die der nachträglichen Interpretation bedürfen)
- Eingabeformat
  - entweder numerisch oder textuell linear (wegen notwendigem Abstandsmaß)
- Ausgabeformat
  - kategorisch
- Lernart
  - unüberwacht



Numerische Prognose

- Ziel
  - Vorhersage numerischer Werte
- Ausgangspunkt
  - Zeitreihe(n) von Werten
- Lernproblem
  - Modell, das den Wert eines bestimmten Merkmals zum (zukünftigen) Zeitpunkt x vorhersagt.
- Beispiele
  - Verkehrsprognose
  - Aktienkursprognose
  - Temperaturvorhersage
  - Voraussage kritischer Betriebszustände
  - Stromverbrauchsprognose

Numerische Prognose

- Wissensrepräsentation
  - Formel oder Modell
- Ein-/Ausgabeformat
  - numerisch
- Lernart
  - überwacht
    (Einstellen der Formel oder des Modells basiert meist auf Trainingsdaten aus der Vergangenheit)

Ähnlichkeitsanalyse

- Ziel
  - Suchen signifikanter Ähnlichkeiten zwischen Datensätzen oder Ereignissen
- Ausgangspunkt
  - Datenmenge
- Lernproblem
  - Finden von Zusammenhängen zwischen Merkmalen innerhalb oder zwischen Datensätzen
- Beispiele
  - Supermarkt:
    If Chips then Beer
  - Energieversorger:
    If temp > 80 then innerhalb 10 min Stromausfall

Ähnlichkeitsanalyse

- Wissensrepräsentation
  - Regeln
- Ein-/Ausgabeformat
  - numerisch: diskret
  - textuell: kategorisch
- Lernart
  - unüberwacht

Abweichungsanalyse

- Ziel
  - Suchen von Ausreißern in einem Datenbestand
- Ausgangspunkt
  - Datenbestand
- Lernproblem
  - Finde Datensätze, die erheblich vom Rest abweichen
- Beispiele
  - Identifikation von Kreditkartenbetrügern anhand der räumlichen Entfernung und des Betrags von Transaktionen
  - Entdeckung von Klausurmanipulationen ...

Abweichungsanalyse

- Wissensrepräsentation
  - Instanzbasiert
- Ein-/Ausgabeformat
  - numerisch: kontinuierlich oder diskret (Abstandsmaß muss definiert sein)
  - textuell: nicht nominal
- Lernart
  - unüberwacht

Informationssysteme: Neuere Konzepte – Teil II

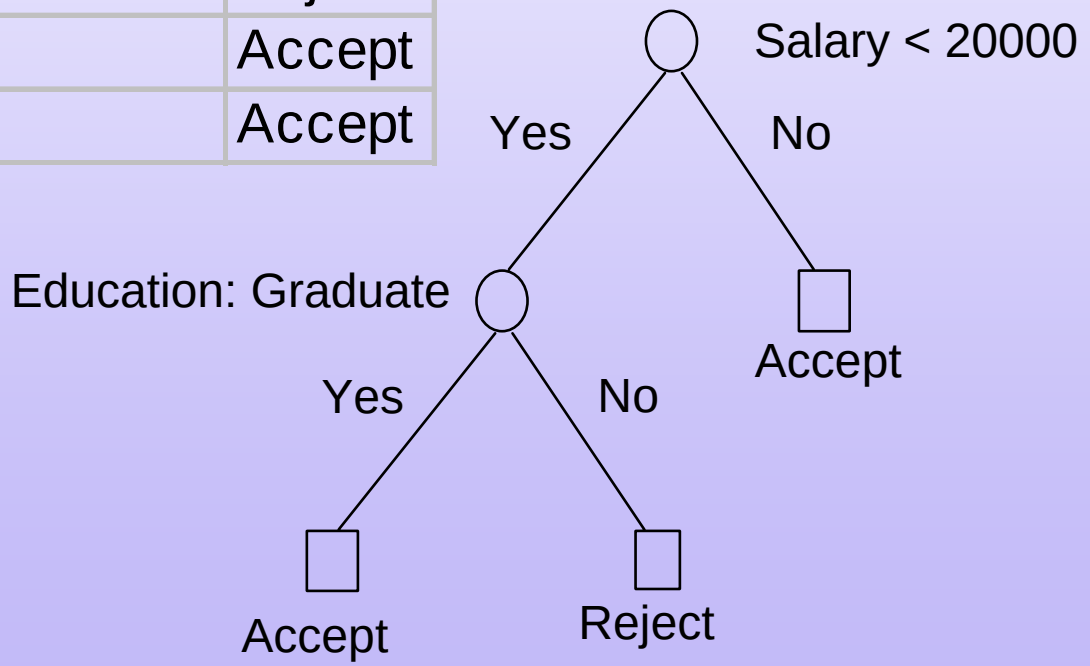
Kapitel 4: KDD & Data Mining
- Data-Mining-Verfahren -

Data-Mining-Verfahren

- Für jede Aufgabenstellung gibt es eine große Zahl von Verfahren
- Sie unterscheiden sich bezüglich
  - Komplexität
  - Qualität des eingesetzten Bewertungsmaßes
  - Restriktionen bezüglich Eingabedatentypen
  - Skalierbarkeit
- Wir werden jeweils einen wichtigen Vertreter herausgreifen

Entscheidungsbäume #1

Salary	Education	Label
10000	High school	Reject
40000	Under graduate	Accept
15000	Under graduate	Reject
75000	Graduate	Accept
18000	Graduate	Accept



Entscheidungsbäume #2

- Aufbau: Rekursives Verfahren
  - Initialisierung
    - Erstelle Wurzelknoten, betrachte alle Datensätze
  - In jedem Knoten:
    - Wähle das Attribut als Splitattribut, das bei einer entsprechenden Teilung der Daten in den entstehenden Teilmengen eine Klassifikation mit höchster Trefferquote erzeugt.
    - Unterteile die Daten in disjunkte Untermengen anhand des gewählten Splits; Erzeuge entsprechende Unterzweige.
    - Mache mit Unterknoten und ihren jeweiligen Untermengen genauso weiter, bis die Untermengen hinreichend homogen klassifiziert sind.

Entscheidungsbäume #3

- Spannende Frage: Wie wähle ich günstiges Splitattribut aus?
  - Verschiedene Bewertungsmaße für durchgeführte Splits bei unterschiedlichen Verfahren
  - Beispiel: *information gain* bei C4.5 (Programm/Algorithmus zur Ableitung von Entscheidungsbäumen), der auf der informationstheoretischen Entropie beruht:
    - Informationsgehalt einer Botschaft ist der negative Logarithmus ihrer Wahrscheinlichkeit
    - Informationsgehalt einer Klassifikation ist die Summe der negativen Logarithmen der Zugehörigkeitswahrscheinlichkeiten über alle Elemente
    - Maximiere *information gain* = Informationsgehalt vor Split – Informationsgehalt nach Split

Entscheidungsbäume #4

- Beobachtungen
  - Greedy-Algorithmus, d.h. initiale Attributauswahl ist entscheidend!
  - Widersprüche möglich, d.h. Klassifikation auf Blattebene immer nur mit p% Konfidenz
  - Komplexität: hoch
    (pro Knoten und getestetes Splitattribut ein Durchgang durch die Daten!)
  - Einsatzgebiet: Klassifikation

Klassifikationsregeln #1

Entscheidungsbaum liefert Regeln der Form

(A = 5) and (B < 3) => C

durch Greedy-Algorithmus.

- Alternative: Direktes Absuchen des Rule Space
  - Vorgehensweise:
    - (1) Generiere zufällige Regeln aus den Attributwerten
    - (2) Prüfe Qualität der Regeln anhand der Daten
    - (3) Wähle die besten Regeln aus
  - Problem
  - Komplexität: sehr hoch (exponentieller Rule Space)

Klassifikationsregeln #2

- Regelwerk wird oft auch aus Entscheidungsbaum generiert wegen
  - besserer Strukturierung (keine Widersprüche)
  - Effizienz (kein exponentielles Absuchen des Regelraums)
- Oft besseres Endergebnis als Entscheidungsbaum wegen:
  - Flexiblerem Pruning (beliebige Einschränkungen weglassen) ➔ Korrektur der Greedy-Problematik
- Häufiges Vorgehen:
  - unterschiedliche Entscheidungsbäume generieren
  - häufig auftretende Regeln extrahieren

k-Nächster Nachbar

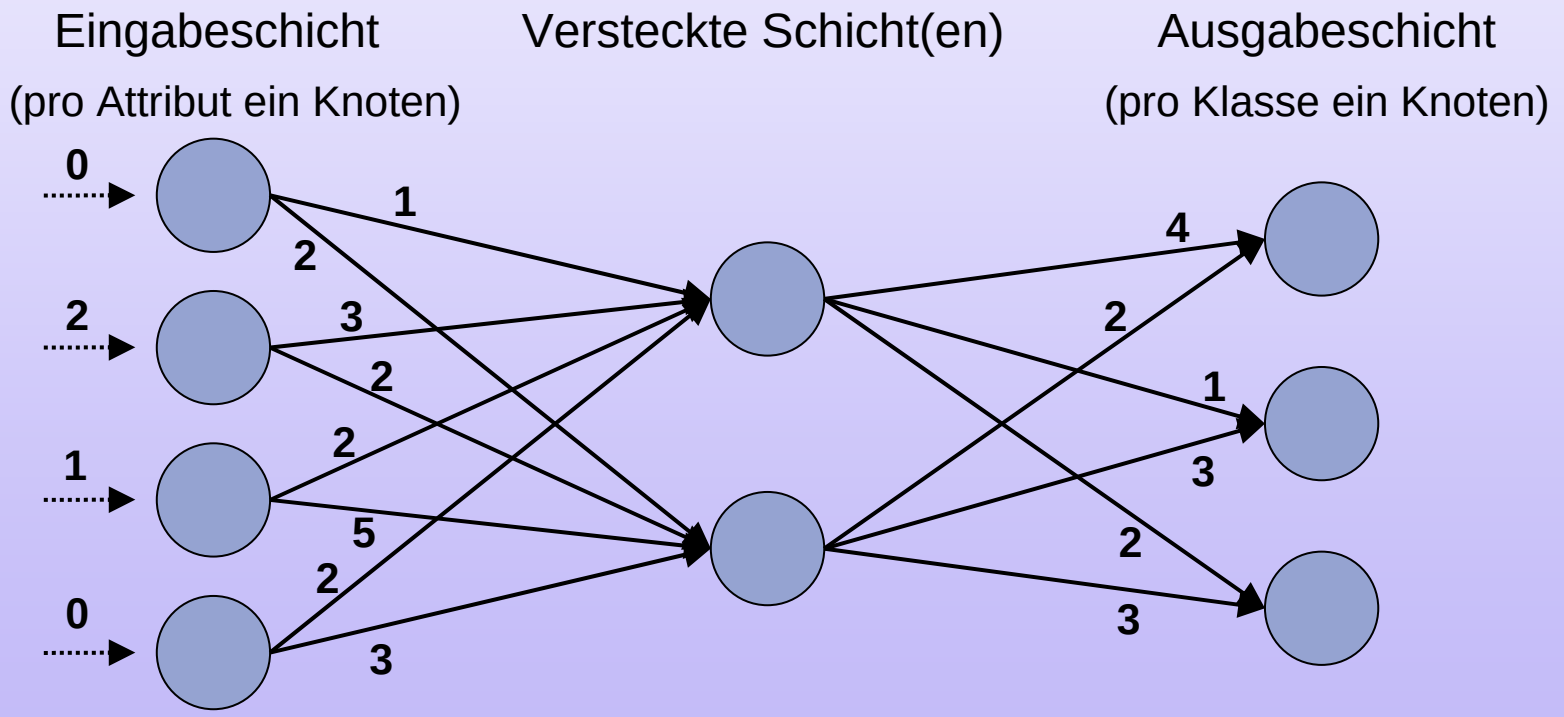
- Aufgabe: Finde zu neuem Tupel die Klasse
- Kein spezieller Lernschritt notwendig
- Feststellung der Klasse eines neuen Tupels:
  - 1.Schritt: Bestimmung der k Tupel, die dem neuen Tupel am nächsten sind
  - 2.Schritt: anhand der Klassen der k Tupel wird die neue Klasse festgelegt.
  - k: Anzahl der Tupel
- Einsatz: Klassifikation
- Problem: Metrik für Abstand benötigt
- Komplexität: hoch

Neuronale Netze #1

- Meist Feed-Forward-Netze:
  - Neuronen in Schichten
  - **Verarbeitung pro Knoten in 2 Schritten:**
    - (1) Gewichtung der Eingabewerte und Addition zu einem Wert
    - (2) Übergabe des Wertes an Schwellwertfunktion
  - Lernprozess
    - Auswertung des Fehlers
    - Anpassung der Netzparameter, meist Rückrechnung des Fehlers durch Backpropagation-Lernregel

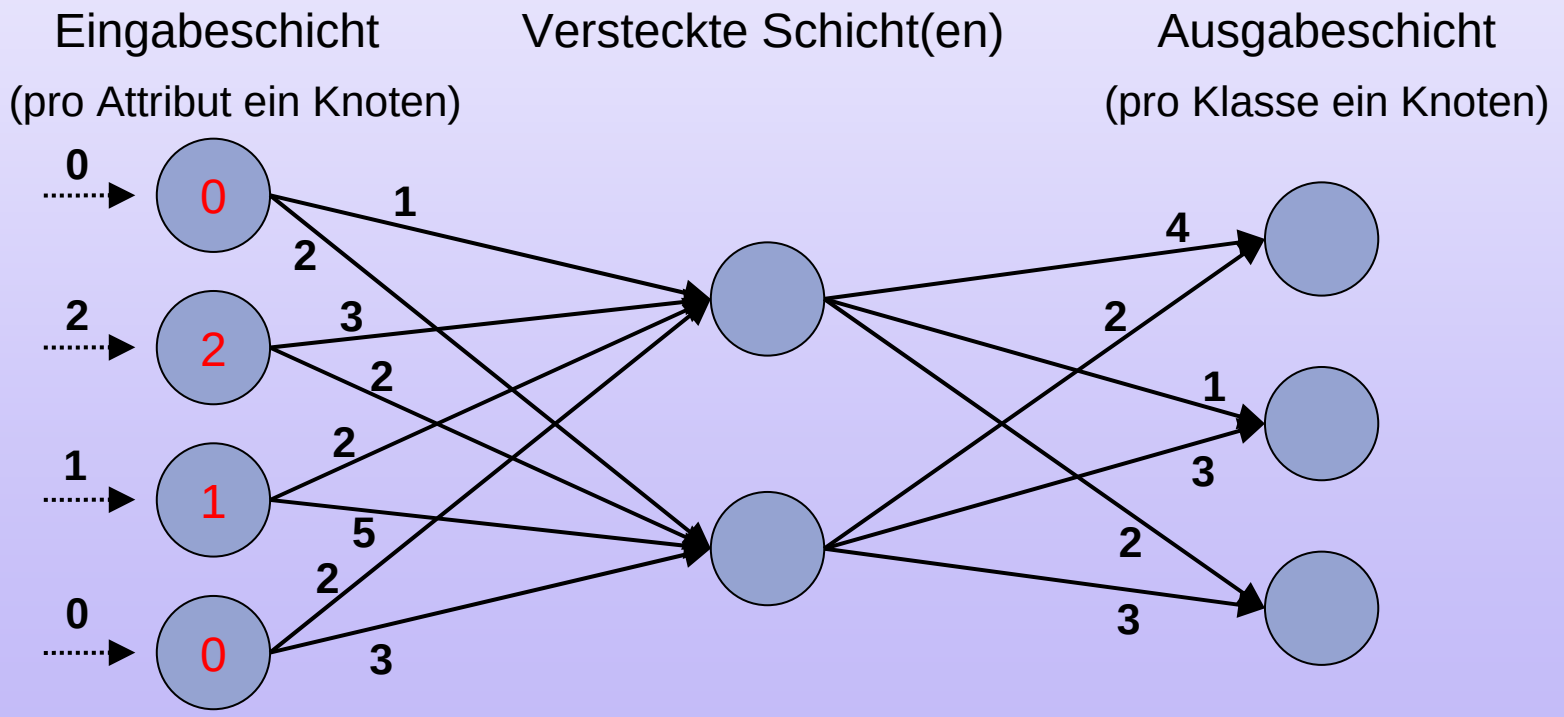
Neuronale Netze #2

Schwellwertfunktion s(x)=x



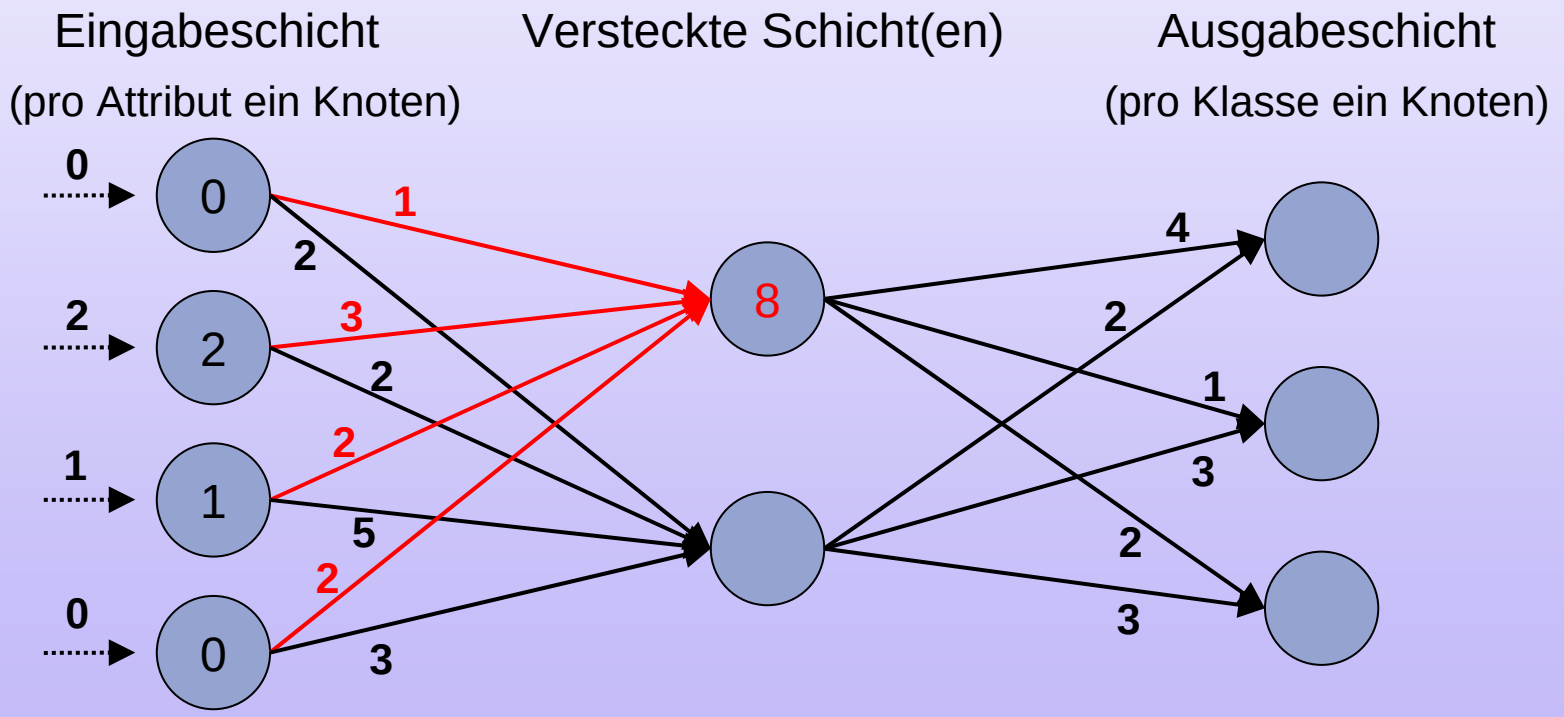
Neuronale Netze #2

Schwellwertfunktion s(x)=x



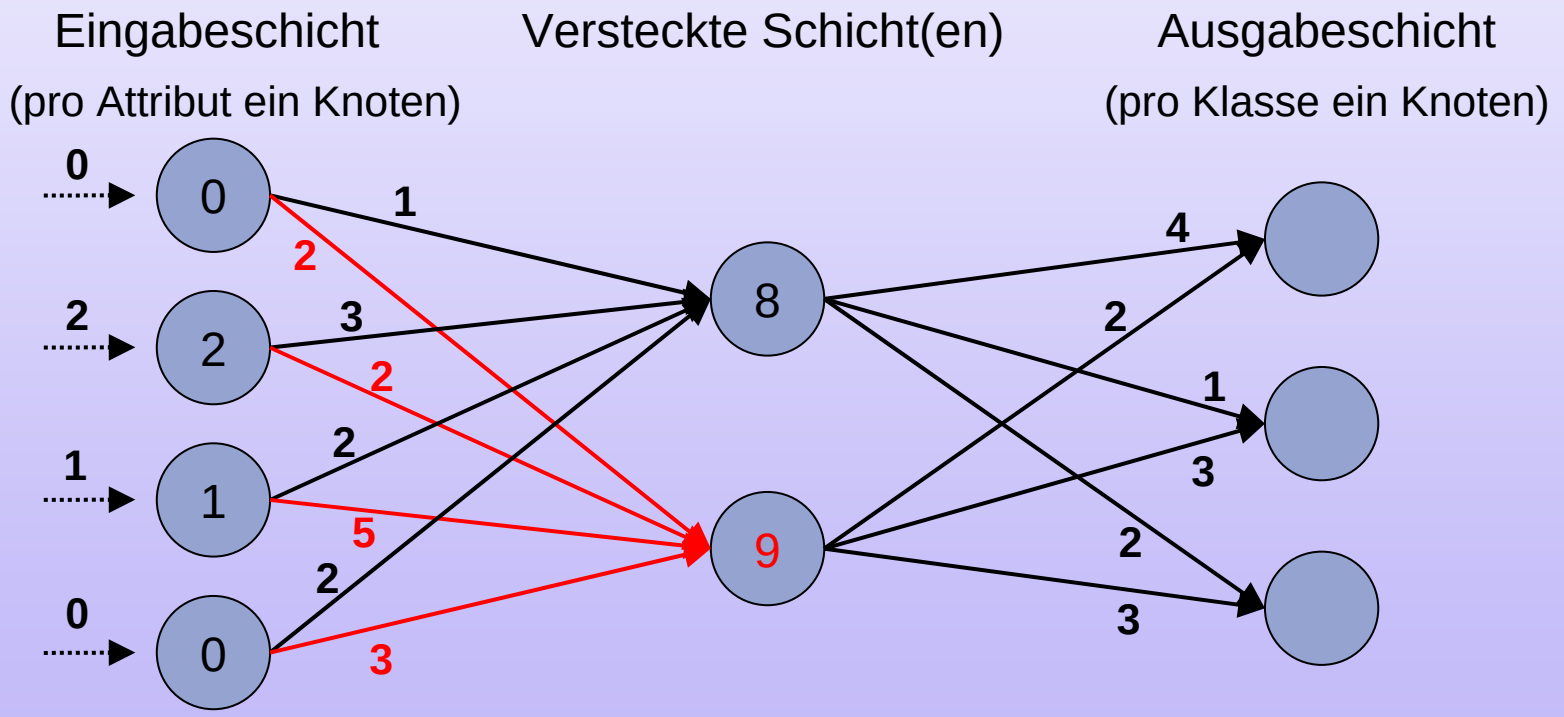
Neuronale Netze #2

Schwellwertfunktion s(x)=x



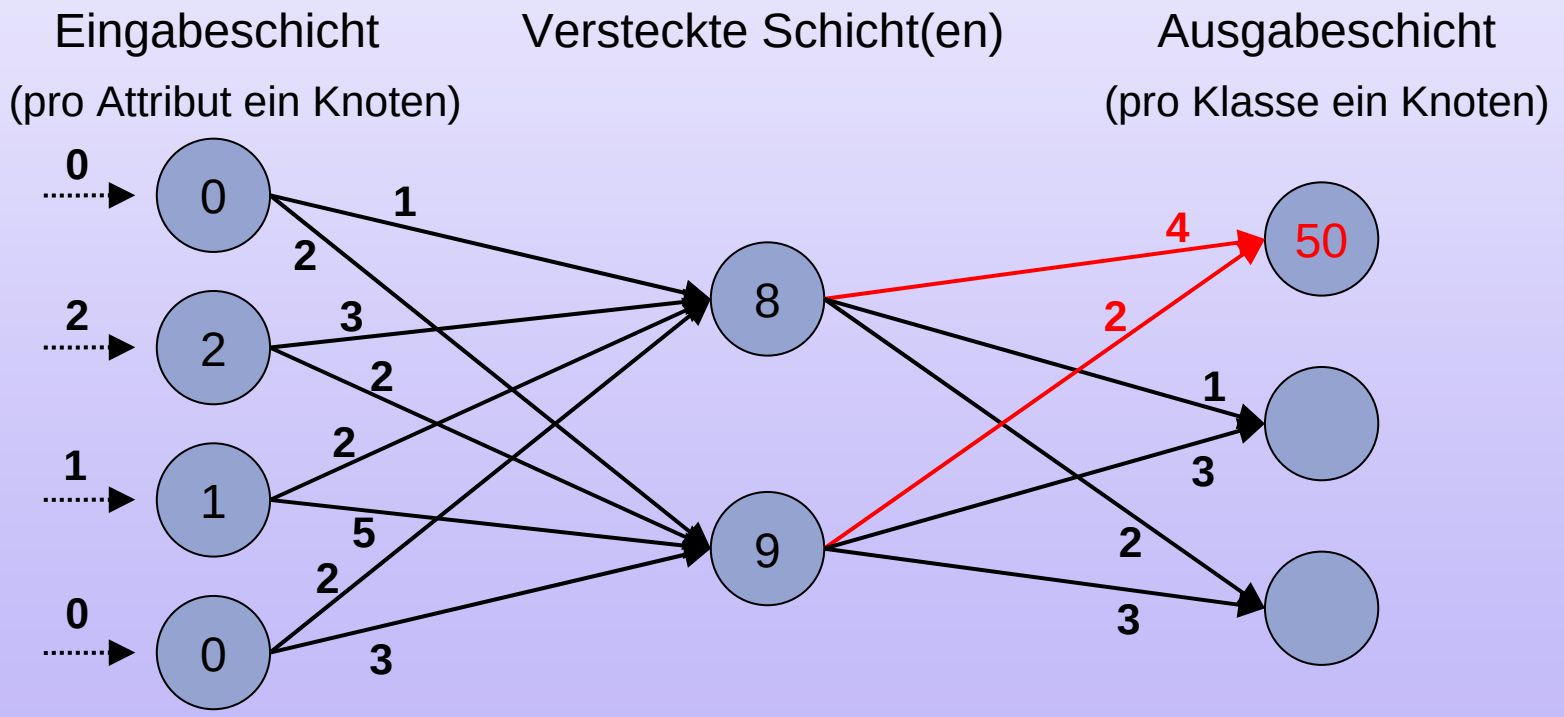
Neuronale Netze #2

Schwellwertfunktion s(x)=x



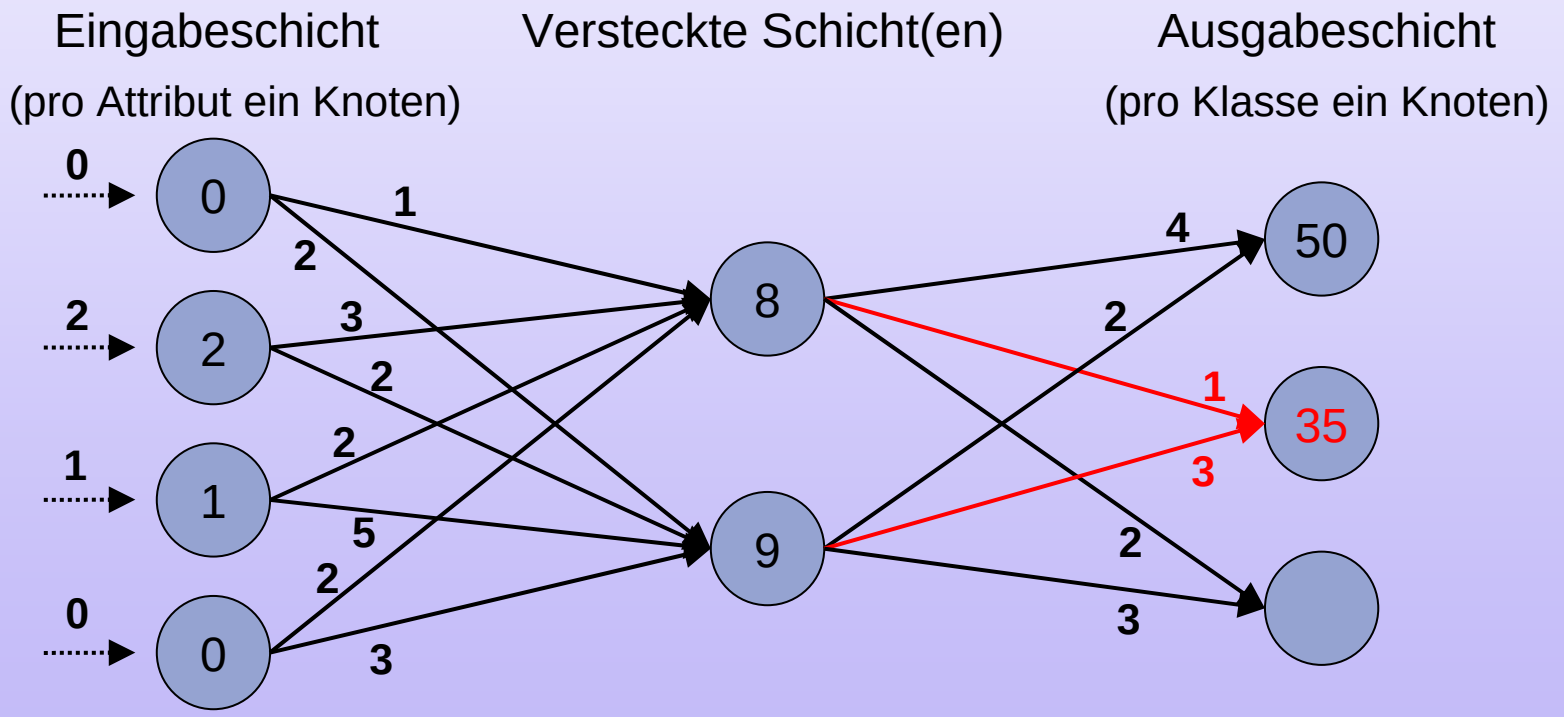
Neuronale Netze #2

Schwellwertfunktion s(x)=x



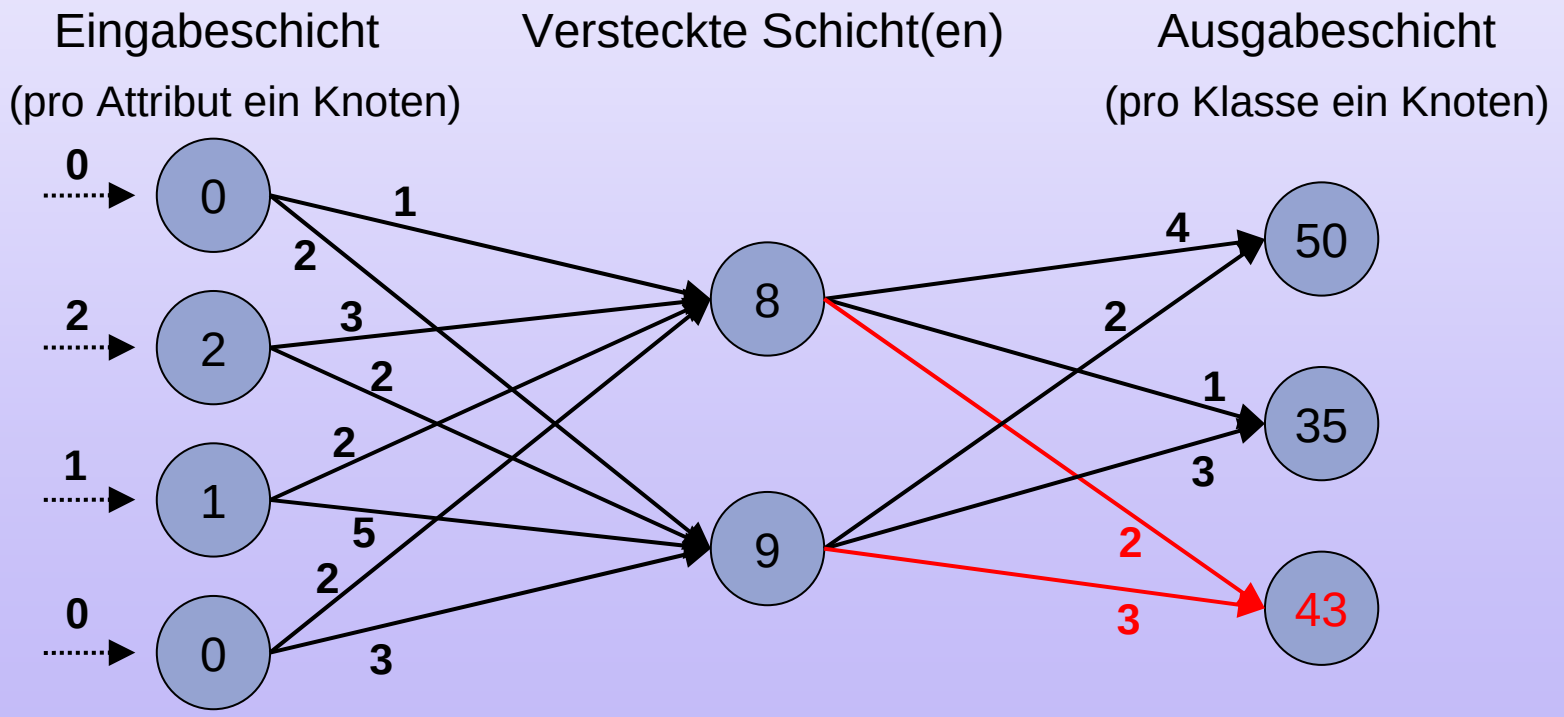
Neuronale Netze #2

Schwellwertfunktion s(x)=x



Neuronale Netze #2

Schwellwertfunktion s(x)=x



Neuronale Netze #3

- Einsatz:
  - Klassifikation: jeder Ausgang repräsentiert eine Klasse
  - Numerische Prognose: ein Ausgang liefert den Prognosewert
  - Eingangscodierung analog: numerisch oder pro Eingang ein Wert
- Komplexität: recht hoch
- Problem: trainiertes Modell für den Menschen schwer (nicht) interpretierbar

Assoziationsregeln #1

- Gegeben
  - einige Mengen von Objekten {a,b}, {c,a}, {a,c,e}, …
- Ziel
  - Suche Assoziationen der Form: X-> Y
    - Wenn X in Menge, dann auch Y in Menge X={x1,x2,..,xn}, Y={y1,..,yn}
- Einsatz
  - Ähnlichkeitsanalyse
- (Klassisches) Beispiel
  - Wenn ein Kunde Pampers und Babynahrung kauft, dann kauft er auch Bier.

Assoziationsregeln #2

- Regel muss eine minimale Konfidenz (confidence) besitzen:
  - 1 & 2 ➔ 3 hat 90% Konfidenz, wenn ein Kunde, der 1 und 2 kauft, in 90% aller Fälle auch 3 kauft.

$$\frac{\text{Anzahl der Kassenbons mit Produkten 1 und 2 und 3}}{\text{Anzahl der Kassenbons mit Produkten 1 und 2}}$$

- Regel muss eine minimale Unterstützung (support) besitzen:
  - 1 & 2 => 3 sollte für einen minimalen prozentualen Anteil der Käufe gelten, um einen Geschäftswert zu haben.

$$\frac{\text{Anzahl der Kassenbons mit Produkten 1 und 2 und 3}}{\text{Anzahl aller Kassenbons}}$$

Assoziationsregeln #3

- Apriori-Algorithmus
  - Finde alle Mengen an Artikeln, die eine bestimmte minimale Unterstützung (support) haben:
    - Starte mit Mengen der Größe 1
    - Kombiniere in jedem Schritt zu Mengen der Größe n+1
    - Apriori-Regel: JEDE Untermenge einer Menge mit minimaler Unterstützung muss selbst minimale Unterstützung haben ➔ Einschränkung der Kandidatenmenge
  - Für alle Kombinationen aus den oben gefundenen Mengen:
    - Generierung aller Regeln
    - Bestimmung Konfidenz

Assoziationsregeln #4

- Beispiel: minimaler Support = 0,7

Datenbank

TID	Itemset
1	{1,3,4}
2	{2,3,5}
3	{1,2,3,5}
4	{2,5}

Assoziationsregeln #4

- Beispiel: minimaler Support = 0,7

Datenbank

TID	Itemset
1	{1,3,4}
2	{2,3,5}
3	{1,2,3,5}
4	{2,5}

1. Scan

Itemset	Support
{1}	0,5
{2}	0,75
{3}	0,75
{4}	0,25
{5}	0,75

Assoziationsregeln #4

- Beispiel: minimaler Support = 0,7

Datenbank

TID	Itemset
1	{1,3,4}
2	{2,3,5}
3	{1,2,3,5}
4	{2,5}

1. Scan

Itemset	Support
{1}	0,5
{2}	0,75
{3}	0,75
{4}	0,25
{5}	0,75

Häufige

Itemset	Support
{2}	0,75
{3}	0,75
{5}	0,75

Assoziationsregeln #4

- Beispiel: minimaler Support = 0,7

Datenbank

TID	Itemset
1	{1,3,4}
2	{2,3,5}
3	{1,2,3,5}
4	{2,5}

1. Scan

Itemset	Support
{1}	0,5
{2}	0,75
{3}	0,75
{4}	0,25
{5}	0,75

Häufige

Itemset	Support
{2}	0,75
{3}	0,75
{5}	0,75

Kombinieren

Itemset
{2,3}
{2,5}
{3,5}

Assoziationsregeln #4

- Beispiel: minimaler Support = 0,7

Datenbank

TID	Itemset
1	{1,3,4}
2	{2,3,5}
3	{1,2,3,5}
4	{2,5}

1. Scan

Itemset	Support
{1}	0,5
{2}	0,75
{3}	0,75
{4}	0,25
{5}	0,75

Häufige

Itemset	Support
{2}	0,75
{3}	0,75
{5}	0,75

Kombinieren

Itemset
{2,3}
{2,5}
{3,5}

2. Scan

Itemset	Support
{2,3}	0,5
{2,5}	0,75
{3,5}	0,5

Assoziationsregeln #4

- Beispiel: minimaler Support = 0,7

Datenbank

TID	Itemset
1	{1,3,4}
2	{2,3,5}
3	{1,2,3,5}
4	{2,5}

1. Scan

Itemset	Support
{1}	0,5
{2}	0,75
{3}	0,75
{4}	0,25
{5}	0,75

Häufige

Itemset	Support
{2}	0,75
{3}	0,75
{5}	0,75

Kombinieren

Itemset
{2,3}
{2,5}
{3,5}

2. Scan

Itemset	Support
{2,3}	0,5
{2,5}	0,75
{3,5}	0,5

Häufige

Itemset	Support
{2,5}	0,75

Assoziationsregeln #5

- itemsets aus 3-Elementen?
  - ➔Gibt's keine mehr - Warum?
- Regeln aus häufigen Itemsets bauen
- In unserem Fall:
  - 2 -> 5 mit Konfidenz 1
  - 5 -> 2 mit Konfidenz 1
- Im Allgemeinen
  - aus allen denkbaren Teilmengen eines Itemsets Regeln bauen

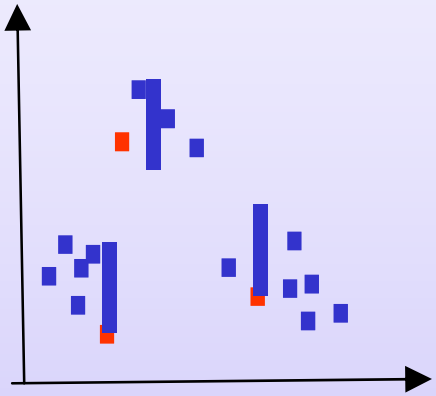
k-means-Clustering #1

- Aufgabe: Zusammenfassung beliebig-dimensionaler Daten zu Gruppen
- Algorithmus
  - Initialisierung
    - (1): aus Datenbestand werden k Tupel als Repräsentanten von Clustern gewählt
  - Iteration:
    - (2): Zuordnung der Tupel zu dem ähnlichsten Repräsentanten
    - (3): Anpassung der Repräsentanten
- Problem: Festlegung von k = #Cluster!

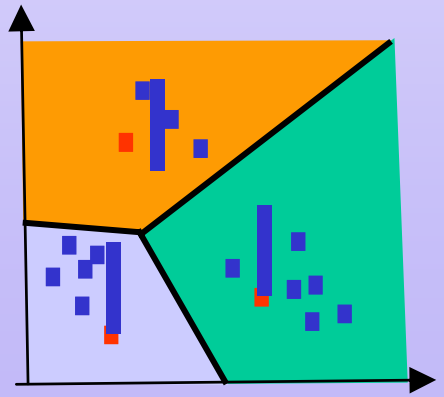
k-means-Clustering #2

k=3

1. Auswahl der Repräsentanten

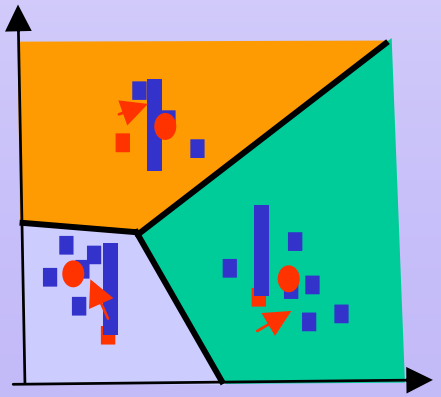


2. Zuordnung der Objekte



Iteration

3. Anpassung der Repräsentanten



Genetische Algorithmen #1

- eigentlich ein Optimierungsverfahren
- basiert auf Populationen von Individuen
- Optimierung der Population durch genetische Operatoren:
  - Selektion
    (Auswahl)
  - Mutation
    (Änderung eines oder mehrerer Merkmale)
  - Crossover
    (Kombination von Merkmalen verschiedener Individuen)

Genetische Algorithmen #2

- Grundlage
  - Optimierungskriterium (fitness)
- Einsatz
  - Klassifikation
  - Numerische Prognose

Ausreißeranalyse

- basiert auf statistischer Verteilung (➔Verteilung schätzen) oder bestimmtem Distanzmaß
- Beispiel
  - Distance-based Outliers
    - Tupel x ist ein (p, D)-Outlier, wenn in einem Abstand von D weniger als (1-p)% der Nachbarn liegen
- Problem
  - Abbildung nominaler Werte (wegen Distanzmaß)

Zeitreihenanalyse

- Analyse von Zeitreihen, also Folgen numerischer und nominaler Werte mit Zeitstempel
- 2 Stoßrichtungen und Einsatzbereiche
  - Mustererkennung (Clustering, Merkmalsextraktion), also Ähnlichkeitssuche auf Zeitreihen, Finden von Ereignissen
  - Generierung von Regeln
    - Wenn Kunde A und B kauft, dann innerhalb 3 Wochen auch C.
- spannend: Kombination aus beidem!

Text Mining

- Anwendung intelligenter Verfahren auf Texte
  - Zusammenfassen ähnlicher Dokumente
  - Klassifikation von Dokumenten
- Voraussetzung: Umwandlung von Texten in verwertbare Form, Merkmalsextraktion
  - Überführung in Tokens ohne Leerzeichen und Kleinschreibung
  - Entfernen von Stopwords (und, in ,...)
  - Stemming: Überführung in die Grundform (lesend, gelesen => lesen, ...)

Beispiel: Vivisimo.com

- Hierarchisches Clustering
  - Starte mit häufigen Wörtern
  - Fasse gemeinsam auftretende Basiscluster zu höheren Clustern zusammen
- Anwendung auf Suchmaschine
  - Weiterverarbeitung der Suchergebnisse durch hierarchisches Clustering

Beispiel: Vivisimo.com

Web Mining

- 3 Stoßrichtungen
  - Inhaltsanalyse: Web Content Mining
    - Text Mining auf Web-Daten
  - Strukturanalyse: Web Structure Mining
    - Schließen thematischer Zusammenhänge aus Linkstruktur des Web (Gruppen, die aufeinander verweisen, arbeiten am gleichen Thema, …)
  - Nutzungsanalyse: Web Usage Mining
    - Analyse von Weblogs, z.B. mittels Sequenzanalyse, um häufige Navigationsabfolgen zu finden

Informationssysteme: Neuere Konzepte – Teil II

Kapitel 4: KDD & Data Mining
- KDD-Prozess -

KDD-Prozess

- Anwendung einzelner Verfahren reicht nicht:
  - richtiges Verfahren muss erst gefunden werden
  - richtige Parameter müssen erst gefunden werden
  - spannender Ausschnitt aus dem Datenbestand muss erst gefunden werden
  - Daten müssen vorverarbeitet werden

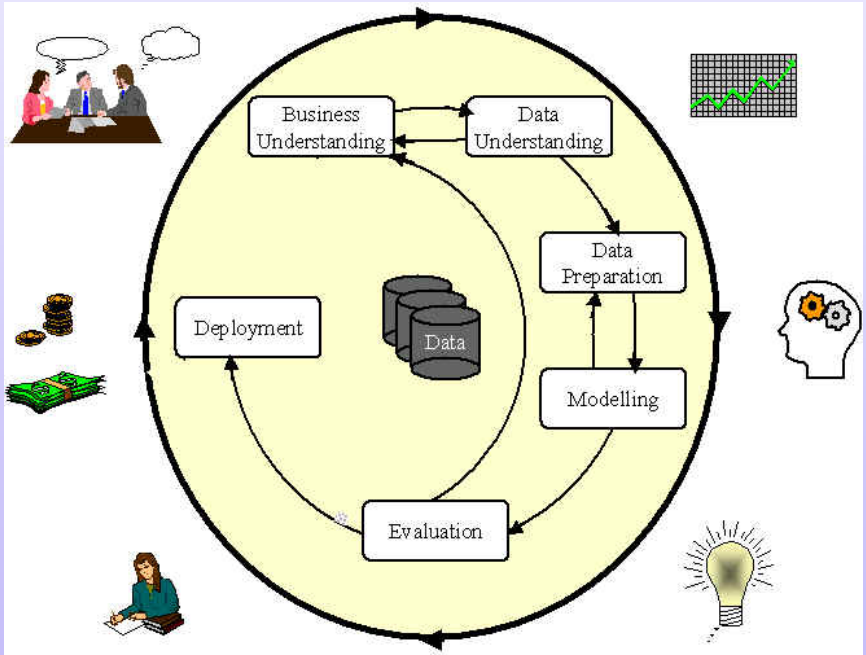
➔interaktiver, iterativer und explorativer Prozess!

KDD-Prozess

- CRISP-DM

  (»Cross Industry Standard Process for Data Mining«)

  - Zusammenschluss verschiedener Hersteller- und Anwenderfirmen
  - definiert allgemeines Prozessmodell



Business Understanding

- Identifiziere Geschäftsziele
- Analysiere Situation und Umfeld
- Formuliere Data-Mining-Ziele (und Erfolgskriterium!)
- Erstelle Projektplan
  - Zeitaufwand:
    - Data Understanding 20-30%
    - Data Preparation 50-70% (!)
    - Modeling + Evaluation 10-20%
    - Deployment 5-10%

Data Understanding

- Initiale Daten sammeln
  - Quellen identifizieren
  - Konsistenz der Quellen prüfen
- Daten beschreiben
  - Bedeutung und Verteilung der Attribute
- Daten erforschen
  - Visualisierung, interaktive Anfragen (OLAP, ...)
- Datenqualität sicherstellen
  - Missing Values ...

Data Preparation

- Selektieren
- Säubern
  - falsche und fehlende Werte ersetzen
- Vorverarbeiten
  - abgeleitete/aggregierte Attribute berechnen
  - numerische Attribute normieren
- Integrieren
  - Daten aus verschiedenen Quellen
  - semantische Ungleichheiten beachten
- Formatieren

Modeling

- Vefahren auswählen
- Trainings- und Testdaten separieren
- Modell bauen
  - Parameter geeignet einstellen
    (in der Regel mehrere Iterationen erforderlich)
- Ergebnis prüfen
  - an formulierten Zielen
  - an bereits bekanntem Wissen
  - gegenenfalls neue Parameter und nochmal bauen …

Evaluation & Deployment

- Evaluation
  - Messen an den Business Objectives
  - Fehler im Prozess identifizieren
- Deployment
  - Deployment-Plan
  - Wie lange soll das Modell genutzt werden?
  - Erfahrungen sammeln und dokumentieren

Umsetzung in DBMS

- Größtes Problem: Skalierbarkeit
- Ansätze
  - Verbindung von DB und DM
    - Query Shipping, also Abbildung in SQL
    - spezielle Operatoren im DBMS selbst
    - DM – Spracherweiterungen (DMQL, OLE DB für DM)
  - Materialisierung von Zwischenergebnissen
  - Parallelisierung
  - Interaktive Verfahren

Data-Mining-Werkzeuge

- Kriterien
  - Bedienungsfreundlichkeit (Statistiker, trainierter Anwender, Informatiker,...)
  - Skalierbarkeit (Client-Tool, Serverbasiert,...)
  - Umfang (kompletter KDD-Prozess, nur einzelne Algorithmen, statistische Pakete,....)
  - Preis!

Data-Mining-Werkzeuge

- IBM Intelligent Miner:
  - skalierbar (serverbasiert, läuft auch auf SP/2)
  - gute Datenbankanbindung
- SPSS Clementine:
  - sehr benutzerfreundlich
  - (inzwischen) serverbasiert
- SAS Enterprise Miner:
  - gute Anbindung an SAS-Statistikpaket

Data-Mining-Werkzeuge

	Berichts-wesen	Analyse	Planung	Data-Mining
Arcplan				
Ascential				
Brio	●	●		
Business Objects	●	◗		
Cognos	●	●	+	◗
Crystal Decisions	●	◗	◗	
Hyperion	◗	●	●	
IBM				●
Informatica	+	+		
Microsoft		◗		+
Microstrategy	●	●		
MIK		●		
MIS	◗	●	●	
NCR Teradata		◗		●
Oracle	◗	◗	●	+
SAP		◗	●	+
SAS	●	●	+	●

+ Neuentwicklung oder Zukauf in den vergangenen Monaten
● verfügbare Funktionalität
◗ eingeschränkte Funktionalität

Quelle: Carsten Bange, BARC, April 2003

COMPUTER ZEITUNG 19/2003/gk

Einordnung

	SQL-Anfrage	OLAP	Data Mining
Detaillierungsgrad	beliebig	aggregiert	oft auch Detaildaten
Problemanpassung	keine	allgemeine Werkzeuge	spezielle Algorithmen
Datenzugriff	Schlüssel+Attribute	Dimensionen	intern ausgewählt
Vorhehensweise	Top-down	Top-down	Bottom-up
Komplexität	gering	mittel	hoch

Literatur

- CRISP-Prozess
  - http://www.crisp-dm.org
- Data Mining
  - Ian H. Witten, Eibe Frank. Data Mining
- Software dazu
  - http://www.cs.waikato.ac.nz/ml/weka