Data Mining und maschinelles Lernen

Einführung und Anwendung mit WEKA

Caren Brinckmann
16. August 2000

http://www.coli.uni-sb.de/~cabr/vortraege/ml.pdf

http://www.cs.waikato.ac.nz/ml/weka/

Inhalt

- Einführung: Data Mining
- ausführliches Beispiel
- Methoden maschinellen Lernens
- Eingabedaten
- Bewertung
- Anwendung mit WEKA

Einführung

- Data Mining ("*Datenbergbau*")
  - Problemlösung durch Analyse großer und komplexer Datenmengen (z.B. Korpora)
  - bedeutsame und aussagekräftige **Strukturen/Muster** und **Beziehungen** in den Datensätzen finden und beschreiben
  - Aufbau von Modellen
- Ziele
  - Erklärung der Daten, Wissenserwerb (welches sind die wichtigen Parameter?)
  - Vorhersagen für neue Daten
- Methoden: maschinelles Lernen

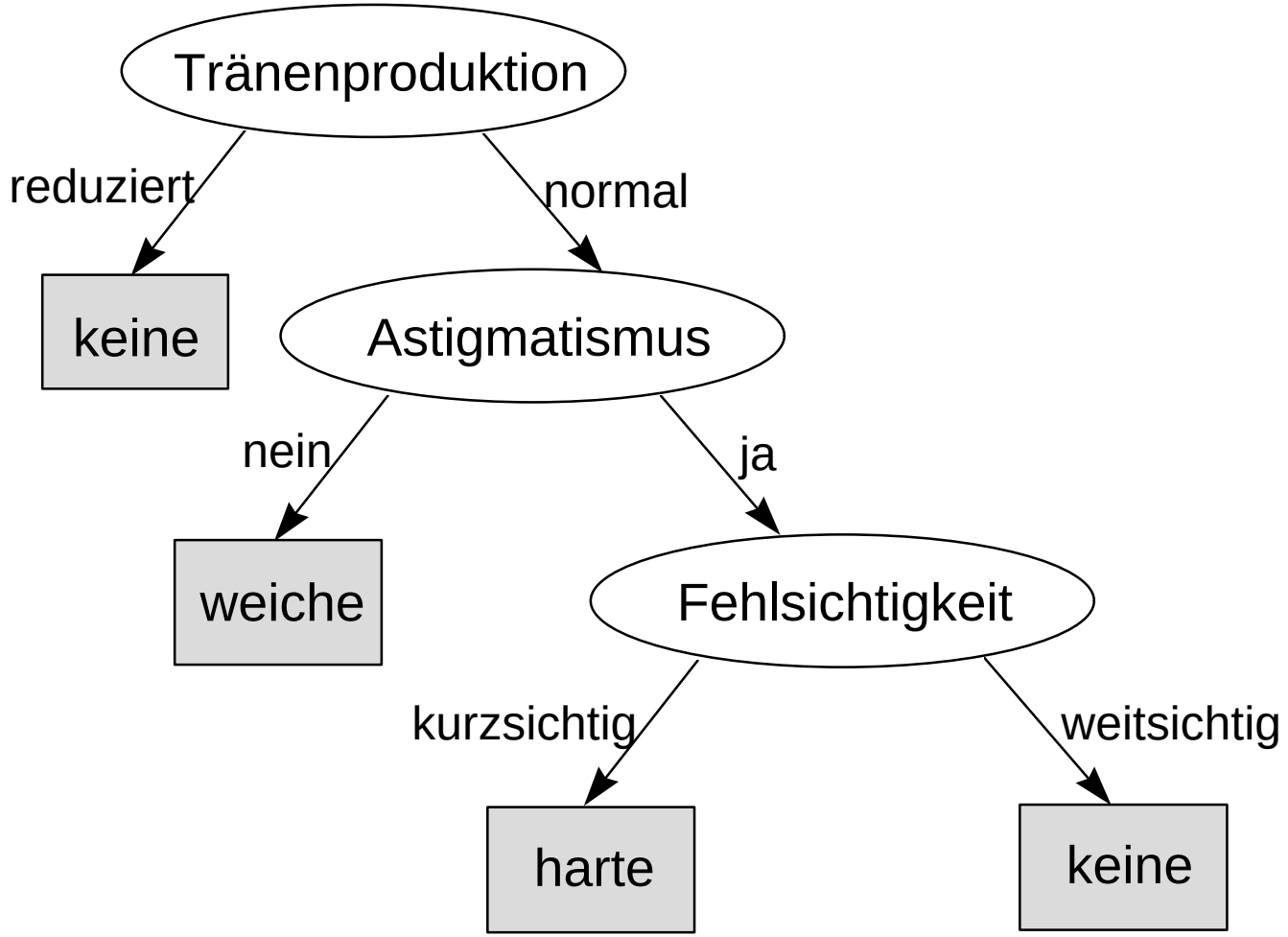
Beispiel: Kontaktlinsen-Empfehlung

Alter	Fehlsichtigkeit	Astigmatismus	Tränenproduktion	Kontaktlinsen
jung	kurzsichtig	nein	reduziert	keine
jung	kurzsichtig	nein	normal	weiche
jung	kurzsichtig	ja	reduziert	keine
jung	kurzsichtig	ja	normal	harte
jung	weitsichtig	nein	reduziert	keine
jung	weitsichtig	nein	normal	weiche
jung	weitsichtig	ja	reduziert	keine
jung	weitsichtig	ja	normal	harte
vor-alterssichtig	kurzsichtig	nein	reduziert	keine
vor-alterssichtig	kurzsichtig	nein	normal	weiche
vor-alterssichtig	kurzsichtig	ja	reduziert	keine
vor-alterssichtig	kurzsichtig	ja	normal	harte
vor-alterssichtig	weitsichtig	nein	reduziert	keine
vor-alterssichtig	weitsichtig	nein	normal	weiche
vor-alterssichtig	weitsichtig	ja	reduziert	keine
vor-alterssichtig	weitsichtig	ja	normal	keine
alterssichtig	kurzsichtig	nein	reduziert	keine
alterssichtig	kurzsichtig	nein	normal	keine
alterssichtig	kurzsichtig	ja	reduziert	keine
alterssichtig	kurzsichtig	ja	normal	harte
alterssichtig	weitsichtig	nein	reduziert	keine
alterssichtig	weitsichtig	nein	normal	weiche
alterssichtig	weitsichtig	ja	reduziert	keine
alterssichtig	weitsichtig	ja	normal	keine

Beispiel: Entscheidungstabelle

Astigmatismus	Tränenproduktion	Kontaktlinsen
ja	reduziert	keine
nein	reduziert	keine
ja	normal	harte
nein	normal	weiche

Beispiel: Entscheidungsbaum



Beispiel: Klassifikationsregeln

Wenn Tränenproduktion = reduziert:
    Kontaktlinsen = keine

Wenn Astigmatismus = nein:
    Kontaktlinsen = weiche

Wenn Fehlsichtigkeit = kurzsichtig:
    Kontaktlinsen = harte

Sonst: Kontaktlinsen = keine

Beispiel: Assoziationsregeln

Kontaktlinsen=weiche ==> Astigmatismus=nein
Kontaktlinsen=harte ==> Astigmatismus=ja
Kontaktlinsen=weiche ==> Tränenproduktion=normal
Kontaktlinsen=harte ==> Tränenproduktion=normal
Tränenproduktion=reduziert ==> Kontaktlinsen=keine
Alter=jung, Kontaktlinsen=keine ==> Tränenproduktion=reduziert
...

Methoden maschinellen Lernens

- Methoden, die **verständliche, strukturelle Beschreibungen** liefern (im WEKA toolkit implementiert)
- "Black box"-Methoden
  - Neuronale Netze
  - HMMs
  - genetische Algorithmen

Methoden im WEKA toolkit (1)

- überwacht (gelabelte Datensätze)
  - Klassifizierung: Bestimmung der Klasse/Kategorie (Beispiel Kontaktlinsen-Empfehlung)
    - Entscheidungstabellen (welche Attribute sind wichtig?)
    - Entscheidungsbäume (mit nominalen und numerischen Attributen)
    - Klassifikationsregeln
  - Numerische Vorhersage (numeric prediction): Bestimmung eines Zahlenwerts durch
    - Lineare Regression
    - Entscheidungsbäume
      - Regressionsbäume: Durchschnittswerte an den Blättern
      - Modellbäume: Regressionsgleichungen an den Blättern

Methoden im WEKA toolkit (2)

- unüberwacht (ungelabelte Datensätze)
  - Assoziationsregeln: Beziehungen zwischen Attributen
    - *support*: korrekte Fälle / alle Fälle
    - *confidence*: korrekte Fälle / Anwendungsfälle
  - Clusteranalyse: Gruppierung von Fällen
    (möglicher zweiter Schritt: Klassifizierung)

Eingabe

- Menge von unabhängigen Fällen
- jeder Fall wird beschrieben durch seine Werte der vorher festgelegten Attribute
- Attribute: nominal (=kategorial) oder numerisch

- **Vorstrukturierung der Daten: 70% des Zeitaufwands!!**
  - Format
  - Richtigkeit
  - fehlende Werte
  - Redundanzen

Eingabe: ARFF-Format

```
@relation Kontaktlinsen

@attribute Alter              {jung, vor-alterssichtig, alterssichtig}
@attribute Fehlsichtigkeit    {kurzsichtig, weitsichtig}
@attribute Astigmatismus      {nein, ja}
@attribute Traenenproduktion  {reduziert, normal}
@attribute Kontaktlinsen      {weiche, harte, keine}

@data
% fehlende Werte werden durch ? ersetzt
jung,kurzsichtig,nein,reduziert,keine
jung,kurzsichtig,nein,normal,weiche
jung,kurzsichtig,ja,reduziert,keine
jung,kurzsichtig,ja,normal,harte
jung,weitsichtig,nein,reduziert,keine
jung,weitsichtig,nein,normal,weiche
jung,weitsichtig,ja,reduziert,keine
jung,weitsichtig,ja,normal,harte
vor-alterssichtig,kurzsichtig,nein,reduziert,keine
...
```

Bewertung der Lernmethoden

- große Datenmenge verfügbar
  ==> getrenntes Trainings- und Testmaterial
- knappe Datenmenge
  - Trainingsmaterial = Testmaterial: "resubstitution error"
  - "stratified n-fold cross-validation":
    - cross-validation: zufällige Aufteilung des Datenmaterials in voneinander getrenntes Trainings- und Testmaterial
    - stratification: jede Klasse ist im Trainings- und im Testmaterial proportional so häufig vertreten wie im ganzen Datenmaterial
    - n-fold: cross-validation wird n-mal wiederholt (mit jeweils einer anderen Aufteilung)

Anwendung mit WEKA

- .bash_profile (in Eurem Homeverzeichnis) ergänzen:

```
WEKAHOME=/proj/msegsynth/ml/weka302
CLASSPATH=$CLASSPATH:$WEKAHOME/weka.jar
export CLASSPATH
```

  und dann: `source .bash_profile`

- Dokumentation: $WEKAHOME/doc/packages.html
  $WEKAHOME/tutorial.pdf

- Beispieldaten: $WEKAHOME/data/

WEKA Befehle

- **`java`** <weka-Klasse> **`-t`** <ARFF-Datei mit Trainingsmaterial> **`>`** <Ausgabedatei>
- Beispiel Entscheidungsbaum:
  `java weka.classifiers.j48.J48 -t $WEKAHOME/data/book/contact-lenses.arff > contact-lenses_J48.out`
- weitere Optionen:
  -T <ARFF-Datei mit Testmaterial>
  -c <Nummer des Attributs, das klassifiziert werden soll>
  -x <Anzahl der Aufteilungen bei cross-validation>
  ... (**`java`** <weka-Klasse>)

Klassifikation und Assoziation mit WEKA

- **`weka.classifiers.ZeroR`**: Baseline-Klassifizierer
- **`weka.classifiers.DecisionTable -R`**: Entscheidungstabelle
- Entscheidungsbäume mit nur einer Ebene:
  - **`weka.classifiers.OneR`**: nur nominale Attribute
  - **`weka.classifiers.DecisionStump`**: binärer Baum
- **`weka.classifiers.j48.J48`**: Entscheidungsbaum
- **`weka.classifiers.j48.PART`**: Klassifikationsregeln
- **`weka.associations.Apriori -M`** <minimum support> **`-N`** <Anzahl Assoziationsregeln>

Numerische Vorhersage mit WEKA

- **`weka.classifiers.LinearRegression`**: lineare Regression
- **`weka.classifiers.m5.M5Prime -O r`**: Regressionsbaum
- **`weka.classifiers.m5.M5Prime`**: Modellbaum