

# Klassifikation III

## Praktikum Data Warehousing und Data Mining

# Künstliche Neuronale Netze - Veranschaulichung

# Weitere Klassifikationstechniken

# Regelbasierte Klassifikatoren

- Klassifikation durch Regelsatz
  - Beispiel:
    1.  $\text{petalwidth} \leq 0.6$ : Iris-setosa
    2.  $\text{petalwidth} \leq 1.7$  AND  $\text{petallength} \leq 4.9$ : Iris-versicolor
    3. Sonst: Iris-virginica
- Übliches Vorgehen:
  - Entscheidungsbaum lernen
  - Deduktion der wichtigsten Regeln aus Baum
  - Nicht alle Tupel klassifiziert:
    - Default-Regel klassifiziert einige Tupel
    - Im Beispiel: Default-Regel: Iris-virginica
- Regelsätze oft einfacher als Entscheidungsbäume  
⇒ Generalisierung

## Assoziationsregeln zur Klassifikation - Beispiel

- Gegeben:  
Folgende Assoziationsregeln
  - Saft -> Cola; conf: 80%
  - Cola -> Saft; conf: 100%
  - Cola -> Bier; conf: 75%
  - Bier -> Cola; conf: 100%
- Vorhersageattribut:
  - Kauft Kunde Cola?
- Beispieletupel:
  - Kunde kauft Bier  
⇒ Kunde kauft Cola (4. Regel)

# Assoziationsregeln zur Klassifikation -Vorgehen

- Eine Regel passt:  
⇒ Klassifikation eindeutig (mit Konfidenz der Regel)
- Keine Regel passt:  
⇒ Mehrheits-Klasse bzw. unklassifiziert
- Mehrere Regeln passen:
  - Berücksichtigung der Regel mit höchster Konfidenz
    - Regel entscheidet
  - Berücksichtigung der  $k$  Regeln mit höchster Konfidenz (oder auch aller Regeln)
    - Häufigste auftretende Klasse
    - Klasse mit höchster durchschnittlicher Konfidenz der Regeln
  - ...
- Hinweis:  
Verfahren eignet sich auch für sequentielle Regeln.

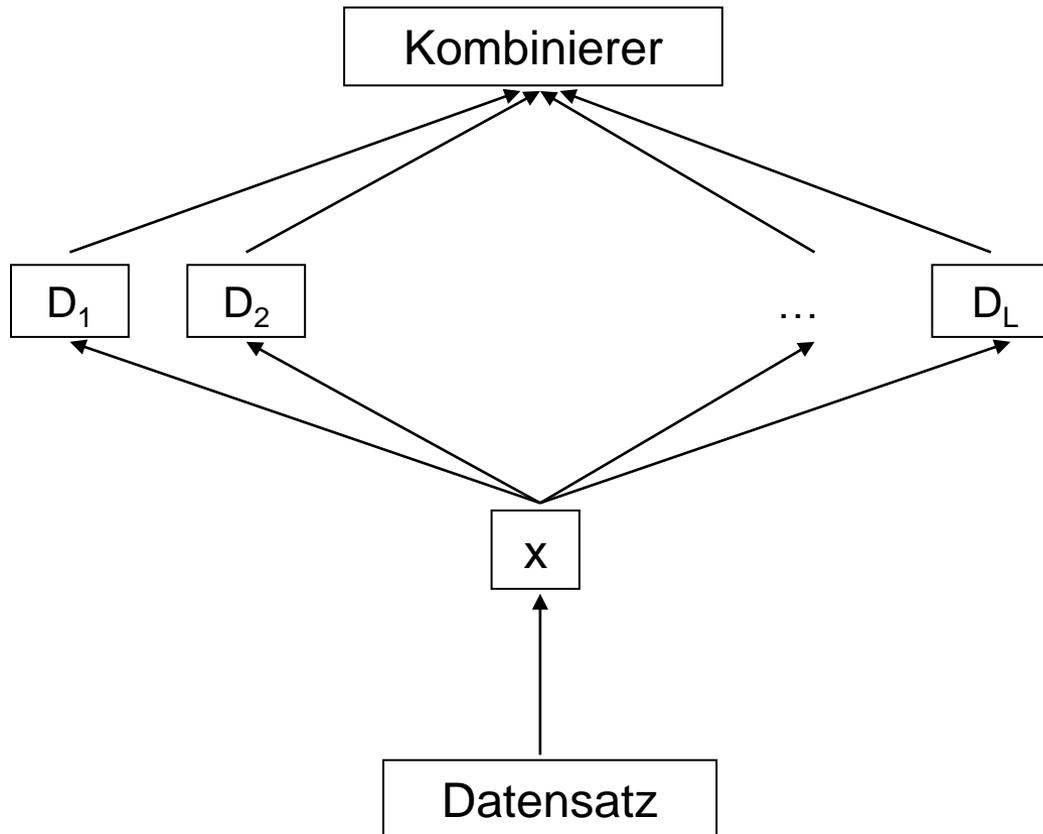
# Kombinierte Klassifikatoren

Combined Classifiers / Multiple  
Classifier System / Classifier Fusion /  
Ensemble Techniques / Committee of  
Machines

# Kombinierte Klassifikatoren - Motivation

- Im „banalen Leben“
  - Bei wichtiger Entscheidung
    - Konsultation mehrerer Experten
    - Beispiel: Ärzte vor kritischer OP, Freunde vor Pferdewette
  - Entscheidungsfindung
    - Mehrheit der Experten oder
    - Vertrauenswürdigste Experten
- Im Data Mining
  - Bei wichtiger Entscheidung
    - Mehrere Klassifikatoren
  - Entscheidungsfindung
    - Kombination der Klassifikatoren oder
    - Classifier Selection
- Ziel: Erhöhung der Accuracy / anderer Maße

# Kombinierte Klassifikatoren - Ansatzpunkte



**Kombinations-Ebene:**  
Einsatz verschiedener  
Kombinationstechniken

**Klassifikator-Ebene:**  
Einsatz verschiedener  
Klassifikatoren

**Feature-Ebene:**  
Einsatz verschiedener  
Feature-Mengen

**Daten-Ebene:**  
Einsatz verschiedener  
Teilmengen

# Daten-Ebene: Bagging & Boosting

- Ursprünglicher Datensatz  $D$ ,  $d = |D|$
- Bagging
  - Zufällige Auswahl von  $k$  Lerndatensätzen
    - Vorgehen: Ziehen mit Zurücklegen von  $d$  Tupeln
  - Lernen je eines Klassifikators pro Lerndatensatz
  - Resultierende  $k$  Klassifikatoren oft erstaunlich unterschiedlich
- Boosting
  - Ähnlich Bagging
  - Ausnahme  $(i+1)$ ter Klassifikator:  
Fokus auf falsch klassifizierte Tupel in  $(i)$ tem Klassifikator
- Optionaler Schritt
  - Evaluation aller  $k$  Klassifikatoren
  - Ergebnisse gewichtet (z.B. mit Accuracy)

# Feature-Ebene: Feature Selection

- Bekannt zur Dimensionsreduktion
- Bei Kombinierten Klassifikatoren:
  - Verschiedene Klassifikatoren durch verschiedene Attributmengen von verschiedenen Feature-Selection-Strategien
- Es ist nicht nur erfolgsversprechend, nur besonders „gute“ Attribute auszuwählen.
  - Verschiedene zufällige Teilmengen
  - Getrennt nach kategorischen/numerischen Attributen

# Klassifikator-Ebene

- Alternativen:
  - Einsatz eines Klassifikators mit verschiedenen Parametern, z.B. maximale Baumhöhe, ...
  - Verwendung verschiedener Klassifikatoren, z.B. Entscheidungsbaum, Neuronales Netzwerk, Naive Bayes, ...
  - Ein Klassifikator für jede Klasse (bei mehr als 2 Klassen)
- Ziel:
  - Klassifikatoren mit möglichst unterschiedlichen Ergebnissen

# Kombinations-Ebene: Strategien

- Problem:
  - Unterschiedliche Vorgehensweisen zur Wahl der Vorhersageklasse
- Alternativen
  - Majority Vote
    - Vorhersageklasse: Ergebnis der meisten Klassifikatoren
  - Weighted Majority Vote
    - Gewichtung mit Konfidenzwerten  
(z.B. von Entscheidungsbäumen, Nearest Neighbour)
  - Stacking
    - Ein weiterer Klassifikator zur Vorhersage der endgültigen Klasse
  - Scoring
    - Bei binären Entscheidungsproblemen, wenn Konfidenzen bekannt
    - $\text{score} = \text{confidence}$  if class=pos
    - $\text{score} = 1 - \text{confidence}$  if class=neg
    - Gesamt-Score: Mittel der Scores aller Klassifikatoren
    - Setzen eines Schwellwertes zur Klassifikation
  - Weitere Strategien in der Literatur...

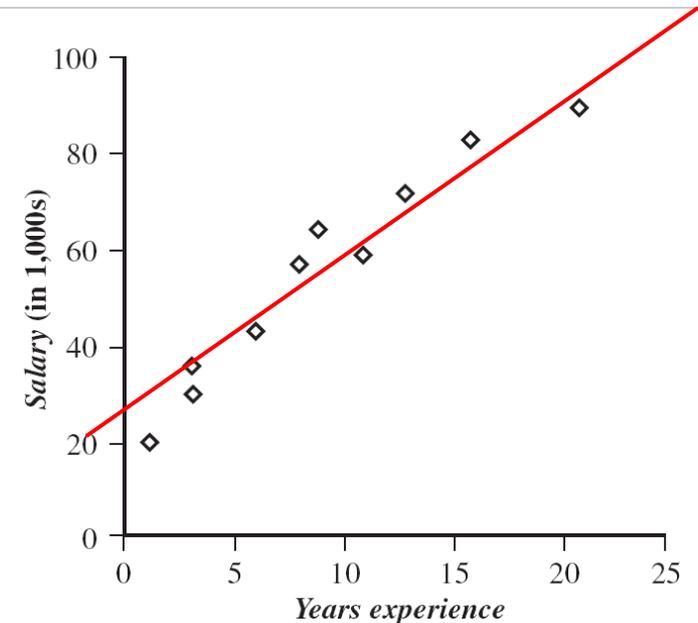
# Statistische Techniken zur Regression (Klassifikation durch Schwellwertsetzung)

# Regressionsprobleme

- Idee
  - Bestimmung eines unbekanntes *numerischen* Attributwertes (*ordinale* und *kategorische* (zumindest *binäre*) Vorhersagen durch Schwellwertsetzung)
  - Unter Benutzung beliebiger bekannter Attributwerte
- Beispiele:
  - Vorhersage von Kundenverhalten wie ‚Zeit bis Kündigung‘
  - Vorhersage von Kosten/Aufwand/Bedarf/Verkaufszahlen/...
  - Berechnung von diversen Scores/Wahrscheinlichkeiten
  - ...
- Klassifikation: Durch Schwellwertsetzung

# Einfache Lineare Regression

- Vorhersage der Zielvariable  $y$  durch eine Prediktorvariable  $x$
- Gegeben: Lerndatensatz  $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ ,  $n = |D|$
- Vermutung eines linearen Zusammenhangs
- Gesucht: Gerade  $y = w_0 + w_1 x$ 
  - Bestimmung von  $w_0$ ,  $w_1$  (Regressionskoeffizienten)



Lerndatensatz  $D$ :

$x$ years experience	$y$ salary (in \$1000s)
3	30
8	57
9	64
13	72
3	36
6	43
11	59
21	90
1	20
16	83

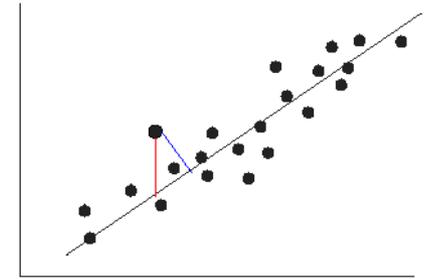
$$w_0 = 23,6$$

$$w_1 = 3,5$$

$$y = 23,6 + 3,5 x$$

# Berechnung der Regressionskoeffizienten

- Zunächst:
  - Bestimmung des Fehlers als Summe der quadratischen Abweichungen
  - $E = \sum_i (y_i - (w_0 + w_1 x_i))^2$
- Aus der notwendigen Bedingung für ein Minimum der Fehlfunktion lassen sich unter Verwendung von partiellen Ableitungen  $w_0$  und  $w_1$  berechnen:



y-Abstand euklidischer Abstand

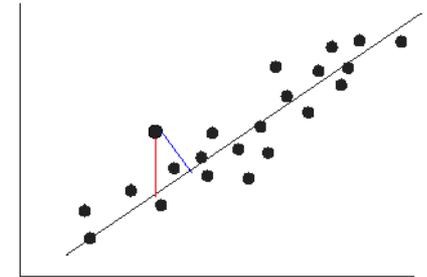
$$w_1 = \frac{\sum_{i=1}^{|D|} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{|D|} (x_i - \bar{x})^2} \quad w_0 = \bar{y} - w_1 \bar{x}$$

$\bar{x}, \bar{y}$ : Durchschnitt aller  $x_1, x_2, \dots, x_n$  bzw. aller  $y_1, y_2, \dots, y_n$

(Rechenbeispiel: S. Data-Mining-Buch von J. Han, M. Kamber)

# Lineare Regression – Fehlermaße

- Üblich ist:
  - Mittlerer quadratischer Abstand in  $y$ -Richtung
- Andere sinnvolle Fehlermaße:
  - Mittlerer absoluter Abstand in  $y$ -Richtung
  - Mittlerer euklidischer Abstand
  - Maximaler absoluter/quadratischer Abstand in  $y$ -Richtung
  - Maximaler euklidischer Abstand
- Diese Maße können jedoch nicht verwendet werden:
  - Betragsfunktion (absoluter Abstand) und Maximum sind nicht überall differenzierbar.
  - Die Ableitung beim euklidischen Abstand führt zu einem nichtlinearen Gleichungssystem; ist nicht analytisch lösbar.



$y$ -Abstand

euklidischer Abstand

# Multivariate Lineare Regression

- Typischerweise steht nicht nur eine Prediktorvariable  $x$  zur Verfügung, sondern mehrere ( $p$ ):

$$\text{Vektor } \mathbf{X}_i := x_{i,1}, x_{i,2}, \dots, x_{i,p}$$

- Lerndatensatz:  $D = \{(\mathbf{X}_1, y_1), (\mathbf{X}_2, y_2), \dots, (\mathbf{X}_n, y_n)\}$
- Hyper-Ebene:  $y = w_0 + w_1 x_1 + w_2 x_2 + \dots + w_p x_p$

- Die Methode zur Minimierung der Fehler-Quadrate kann übertragen werden:

Es entsteht ein lineares Gleichungssystem.

- Lösbar mit linearer Algebra (Matrizen).
- Lösung mit numerischen Methoden oft effizienter.

# Lineare Regression – Bewertung

- Eigenschaften
  - Aufwand zum Lösen der Gleichungen:  $O(p^3)$
  - Koeffizienten sind eventuell interpretierbar.
- Vorteile
  - Relativ simples Modell:  
 $p$ -dimensionale Hyperebene bei  $p$  Prediktorvariablen
  - Dient als Baseline zum Vergleich von Regressionstechniken.
- Nachteile
  - Gleichungen sind eventuell nicht lösbar, wenn Prediktorvariablen (nahezu) linear abhängig voneinander sind.
  - Alle Prediktorvariablen werden betrachtet, auch irrelevante.
  - Anfällig für Outlier. (Ignorieren von Datenpunkten gefährlich!)
  - Nicht alle Probleme sind linear...

# Nichtlineare Regression

- Einige nichtlineare Probleme können als polynomielle Funktion modelliert werden. -> KNIME
- Polynomielle (u.a.) Funktionen können in lineare Regressionsmodelle transformiert werden, z.B.:

$$y = w_0 + w_1 x + w_2 x^2 + w_3 x^3$$

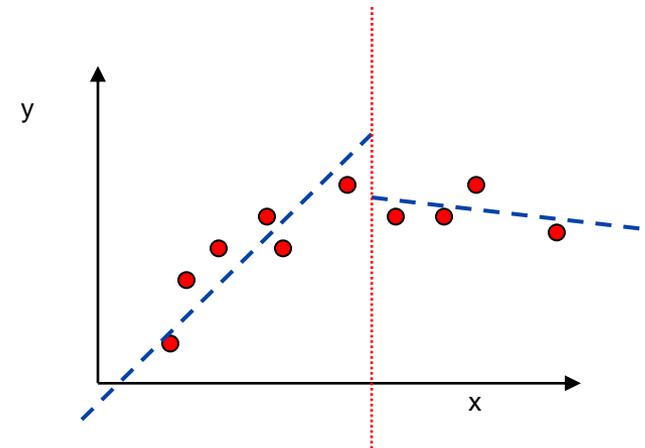
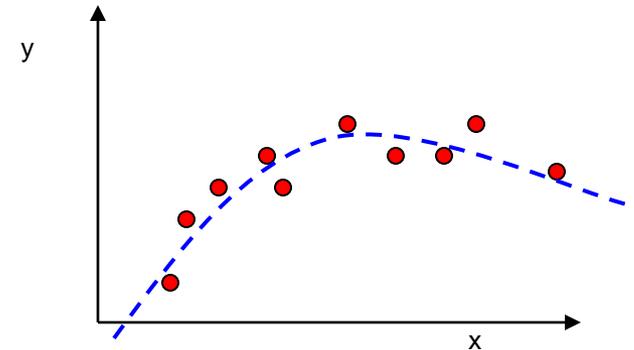
wird gemappt auf:

$$y = w_0 + w_1 x + w_2 x_2 + w_3 x_3$$

- Die Methode zur Minimierung der Fehler-Quadrate mit Ableitungstechniken kann prinzipiell auf beliebige Funktionen übertragen werden.
  - Eventuell sehr hoher Rechenaufwand, aber oft nicht lösbar.
  - Hintergrundwissen kann helfen, einen Term zu finden.

# Lokale Lineare Regression

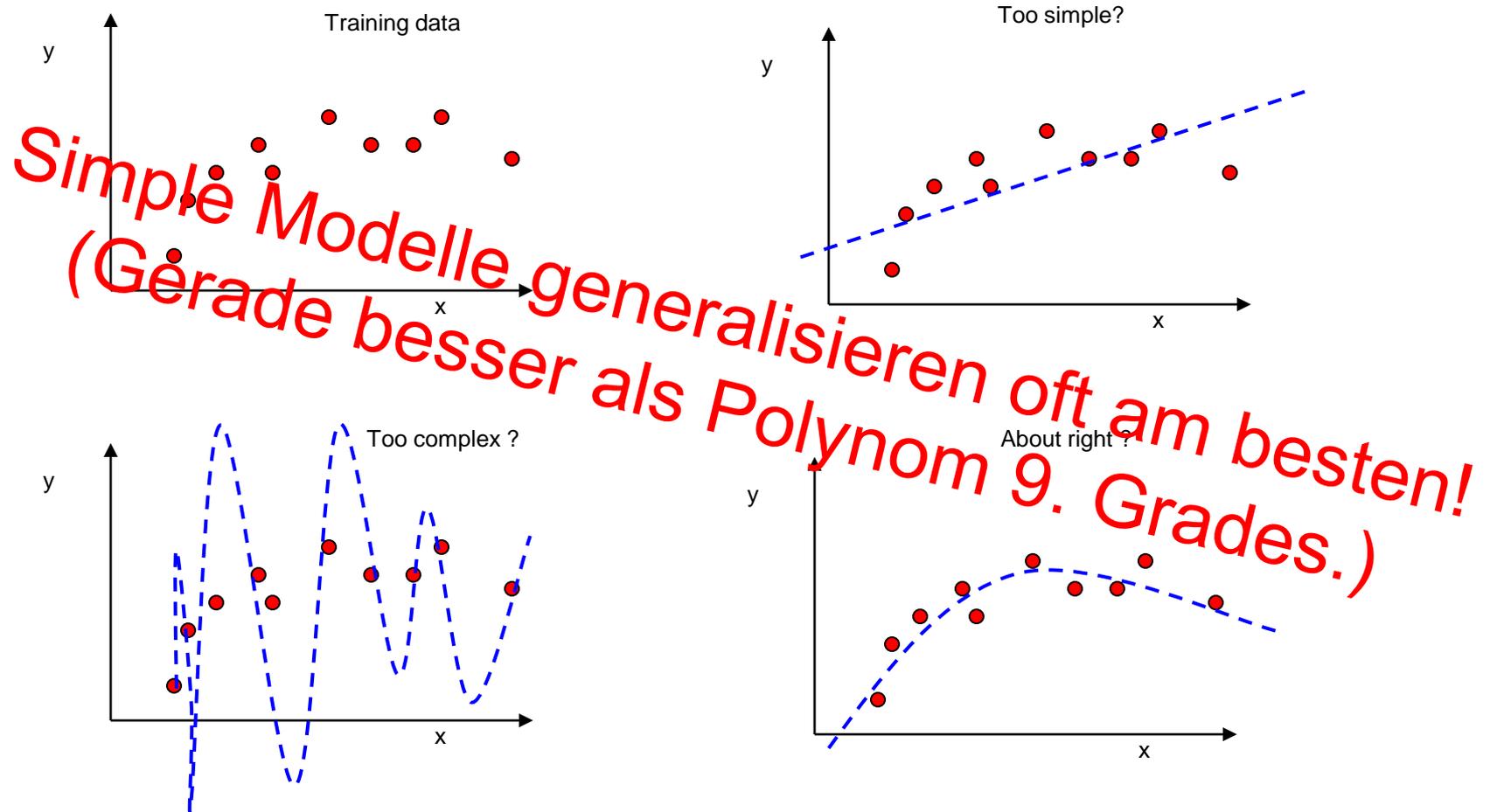
- Idee:  
Mehrere einfache Regressionsfunktionen (hier Geraden) für verschiedene Wertebereiche von  $x$
- Problem:  
Brüche an Wertebereichsgrenzen
- Eine Lösung:  
Splines „glätten“ die Übergänge
- Gut geeignet bei wenigen Prädiktorvariablen.
- Bestimmte Regressionsbauzme greifen die Idee „lokale Regression“ auf...



# Weitere nichtlineare Verfahren

- Oft ist eine numerische Parameter-Bestimmung aus partiellen Ableitungen nicht möglich:
  - Parameter gehen nichtlinear in die Regressionsfunktion ein.
  - Ein alternatives Fehlermaß wird verwendet.
- Lösungsansatz: „Systematisches Trial and Error“
  1. Aufstellen einer (beliebigen) Regressionsfunktion.
  2. Suche nach geeigneten Parametern:
    - Random Search
    - Hillclimbing
    - Varianten um lokale Minima zu verhindern
    - Genetische Algorithmen
    - ...

# Generalisierung vs. Overfitting



# Quellen

- J. Han und M. Kamber: „Data Mining: Concepts and Techniques“, Morgan Kaufmann, 2006.
- I.H. Witten und E. Frank: "Data Mining - Practical Machine Learning Tools and Techniques", Morgan Kaufmann, 2005.
- Hand, H. Mannila und P. Smyth: "Principles of Data Mining", MIT Press, 2001.
- T. M. Mitchell: „Machine Learning“, Mc Graw Hill, 1997.
- L. I. Kuncheva: „Combining Pattern Classifiers“, Wiley-Interscience, 2004.
- F. Klawonn: Folien zur Vorlesung „Data Mining“, 2006.
- C. Borgelt: Folien zur Vorlesung „Intelligent Data Analysis“, 2004.  
**Vorlesungsskript verfügbar (120 Seiten):** <http://fuzzy.cs.uni-magdeburg.de/studium/ida/txt/idascript.pdf>
- Pierre Geurts: Folien zur Vorlesung „Stochastic methods“.
- SPSS: Clementine 12.0 Algorithms Guide. 2007.