INF

FAKULTÄT FÜR
INFORMATIK

Classification Algorithms

Alexander Dockhorn

Organisatorisches

- Wöchentliche Termine
- Keine Ausarbeitungen im eigentlichen Sinne
  - Stattdessen Mitarbeit, Hausaufgaben, ordentliches Vorbereiten
- Anrechenbar als
  - Wissenschaftliches Seminar (Bachelor, 3 CP)
  - FIN SMK (Bachelor, 5 CP, mit Programmieraufgabe)
  - Informatikfach (Master, 6 CP mit Programmieraufgabe)
- Ablauf
  - 45 Minuten Vortrag/Diskussionsrunde, 45 Minuten Wettbewerb

Vorträge / Programmierprojekte

Vorträge

- 30 Minuten Vortrag im 2er Team
- 15 Minuten Diskussion
  Hier können alle anderen Punkte sammeln!

Programmierprojekte

- Implementation eines Klassifikationsalgorithmus, der NICHT im Seminar besprochen wird
- Paper suchen oder irgendetwas selber ausdenken
- Auswahl mit mir absprechen, Abgabetermin ist fix: 10.01.2016
- kurze Präsentation zum gewählten Algorithmus
  (10 Minuten + 5 Minuten Diskussion)
- Implementierungssprache: egal, Python bevorzugt,
  (noch) Solaris–kompatibel, Umstellung auf Linux erfolgt derzeit

Bewertung

Mitarbeit (1.5 CP)

- Jeder kann pro Woche zwei Punkte sammeln
- konstruktiver Beitrag zur Diskussion (Frage, Antwort, Kommentar, …)

Vortrag (1.5 CP)

- Einzelnoten für jeden Vortragenden
- Bonus für Interaktion, Einbinden der Zuhörer, Einsatz von Medien…

Programmierprojekt (2 bzw. 3 CP)

- Lauffähigkeit, rechtzeitige Abgabe, Dokumentation
- Vortrag

Klassifikation

Klassifikation

- Clustering / Klassifikation / Regression
  - Clustering: Keine Zugehörigkeitsinformationen
    - Strukturerkennung
  - Klassifikation: endlicher Wertebereich für Klassenlabel
    - Struktur unter Ausnutzung von mehr Informationen
  - Regression: unendlicher Wertebereich

Klassifikation

- In dieser Veranstaltung:
  - Feature Selection / Preprocessing
  - Überwachtes Lernen
    1. Case–based Reasoning, k Nearest Neighbours
    2. Decision Trees
    3. Bayes Classifier
    4. Linear / Quadratic Discriminant Analysis (LDA / QDA)
    5. Linear Learning Machines, Support Vector Machines
  - Teilüberwachtes Lernen
    1. Label Propagation
    2. Semi Supervised Support Vector Machines (S^3VM)
  - Active Learning
  - Ensemble Methoden

k Nearest Neighbours

k Nearest Neighbours

- Spezialfall: Case based Reasoning

- Beispielsituation: Gerichtsverhandlung
  - Der Richter kennt nur die folgenden beiden anderen Fälle aus der Vergangenheit:
    1. Die Angeklagte hat den Mann getötet und ist dafür zu lebenslanger Haft verurteilt worden.
    2. Die Angeklagte hat zweifelsfrei die Tat begangen und ist geständig, einer Frau in der Fußgängerzone die Handtasche aus der Hand gerissen zu haben. Dafür hat sie eine Freiheitsstrafe von 1 Jahr ohne Bewährung erhalten.

„Im Namen des Volkes…“

- Fallanalyse

	Fall 1	Fall 2
Täter:	Weiblich	Weiblich
Opfer:	Männlich	Weiblich
Straftat:	Mord / Totschlag	Raub
Haftstrafe:	Lebenslang	1 Jahr

„Im Namen des Volkes…“

- Fallanalyse

	Fall 1	Fall 2	Vorliegender Fall
Täter:	Weiblich	Weiblich	Männlich
Opfer:	Männlich	Weiblich	Weiblich
Straftat:	Mord / Totschlag	Raub	Diebstahl
Haftstrafe:	Lebenslang	1 Jahr	???

- Case-Based Reasoning
  - Suche den ähnlichsten Fall und treffe die gleiche Entscheidung

- Wie wird der Richter urteilen?

„Im Namen des Volkes…“

- Fallanalyse

	Fall 1	Fall 2	Vorliegender Fall
Täter:	Weiblich	Weiblich	Männlich
Opfer:	Männlich	Weiblich	Weiblich
Straftat:	Mord / Totschlag	Raub	Diebstahl
Haftstrafe:	Lebenslang	1 Jahr	???

- Case-Based Reasoning
  - Suche den ähnlichsten Fall und treffe die gleiche Entscheidung
  - Hier: vorliegender Fall stimmt mit Fall 1 in nur einem Merkmal überein, mit Fall 2 in zwei Merkmalen
  - Ergebnis:

„Im Namen des Volkes…“

- Fallanalyse

	Fall 1	Fall 2	Vorliegender Fall
Täter:	Weiblich	Weiblich	Männlich
Opfer:	Männlich	Weiblich	Weiblich
Straftat:	Mord / Totschlag	Raub	Diebstahl
Haftstrafe:	Lebenslang	1 Jahr	???

- Case-Based Reasoning
  - Suche den ähnlichsten Fall und treffe die gleiche Entscheidung
  - Hier: vorliegender Fall stimmt mit Fall 1 in nur einem Merkmal überein, mit Fall 2 in zwei Merkmalen
  - Ergebnis: 1 Jahr Haftstrafe

k Nearest Neighbour

- Verwendung von nur einem Beispiel
  - Case–based Reasoning ist kNN mit k=1
- Mehr ähnliche Fälle können das Ergebnis absichern
  - Wenn der Richter mehr Fälle gekannt hätte und einbezogen hätte, wäre der Jugendliche glimpflicher davon gekommen.
- Mehrheitsentscheid
  - Entweder alle gleichberechtig oder gewichtet nach Abstand
- Wie lässt sich Ähnlichkeit mathematisch messen?
  - für Vektorräume: z.B. euklidischer Abstand

Abstandsmaße

- euklidischer Abstand

$$d(\vec{x},\vec{y}) = \sqrt[2]{\sum_i (x_i - y_i)^2}$$

- Manhatten Abstand (City–Block Distanz)

$$d(\vec{x},\vec{y}) = \sum_i |x_i - y_i|$$

- Minkowski Abstand*

$$d(\vec{x},\vec{y};p) = \sqrt[p]{\sum_i (x_i - y_i)^p}$$

Abstandsmaße

- Maximum Norm (Chebyshev Metrik)

$$d(\vec{x},\vec{y};\infty) = \lim_{p\to\infty} \sqrt[p]{\sum_i (x_i - y_i)^p} = \max_i |x_i - y_i|$$

- Hamming Abstand

$$d(\vec{x},\vec{y};0) = \lim_{p\to 0} \sqrt[p]{\sum_i (x_i - y_i)^p}$$

  - Entspricht der Anzahl der Komponenten, in denen sich x und y unterscheiden

Visualisierung von Abstandsmaßen

- Kreise
  - Menge von Punkten, die den gleichen Abstand zum Mittelpunkt haben

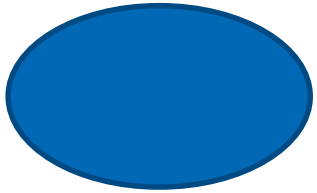
Gewichtete Abstandsmaße*

- Allgemeine Form

$$d(\vec{x}, \vec{y}; p) = \sqrt[p]{\sum_i w_i (x_i - y_i)^p}$$

Jede Dimension bekommt ein eigenes Gewicht

- Kreisbeispiel für euklidischen Abstand mit $\vec{w} = (1, 2)$



Abstand in y–Richtung zählt doppelt, also muss man nur halb so weit vom Mittelpunkt weg sein

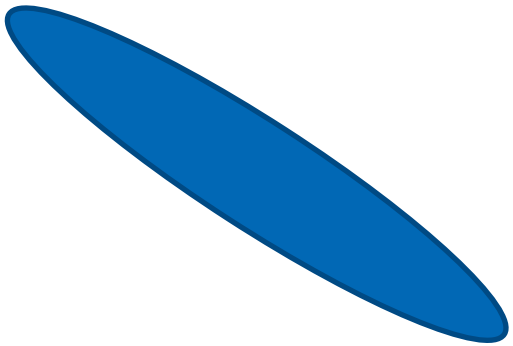
Gewichtete Abstandsmaße

- Nachteil einfacher Gewichte
  - Ellipsoide sind immer Achsenparallel
  - D.h. Abhängigkeiten von unterschiedlichen Attributen werden ignoriert

- Lösung: Mahalanobis–Distanz*

$$d(\vec{x},\vec{y}) = (\vec{x}-\vec{y})^T \Sigma^{-1} (\vec{x}-\vec{y})$$

  - Σ ist Kovarianzmatrix
  - alle paarweisen Kovarianzen werden berücksichtigt



Andere Abstandsmaße

- Standardisierter euklidischer Abstand (steuclidean)
  - Egalisiert unterschiedliche Skalierungen in den Dimensionen
  - Vor der Abstandsberechnung werden alle Dimensionen umgerechnet auf das Intervall [0,1]

- Quadratischer euklidischer Abstand (sqeuclidean)
  - In vielen Berechnungen braucht man nicht den Abstand selbst sondern d^2
  - Wurzelziehen entfällt einfach