Wissensentdeckung in Datenbanken

Katharina Morik und Claus Weihs

Fakultäten Informatik und Statistik

Technische Universität Dortmund

Bekannte Anwendungen

- Google ordnet die Suchergebnisse nach der Anzahl der auf sie verweisenden Hyperlinks an.
- Amazon empfiehlt einem Kunden, der A gekauft hat, das Produkt B, weil alle (viele) Kunden, die A kauften, auch B kauften.
- Der Markt wird beobachtet: wie äußern sich Verbraucher im WWW über ein Produkt? (Sentiment Analysis)
- Versicherungen bewerten ihre Produkte nach den Schadensfällen.
- Verkaufszahlen werden vorhergesagt.
- Daten physikalischer Vorgänge werden analysiert, z.B. Terrabytes von Messungen der Astrophysik.

Interesse an Anwendungen

- Business Reporting soll automatisiert werden. On-line Analytical Processing beantwortet nur einfache Fragen. Zusätzlich sollen Vorhersagen getroffen werden.
- Wissenschaftliche Daten sind so umfangreich, dass Menschen sie nicht mehr analysieren können, um Gesetzmäßigkeiten zu entdecken.
- Geräte sollen besser gesteuert werden, indem aus den log-Dateien gelernt wird.
- Das Internet soll nicht nur gesamte Dokumente liefern, sondern Fragen beantworten.
- Multimedia-Daten sollen personalisiert strukturiert und gezielter zugreifbar sein.

CRISP-DM: CRoss Industry Standard Process for Data Mining

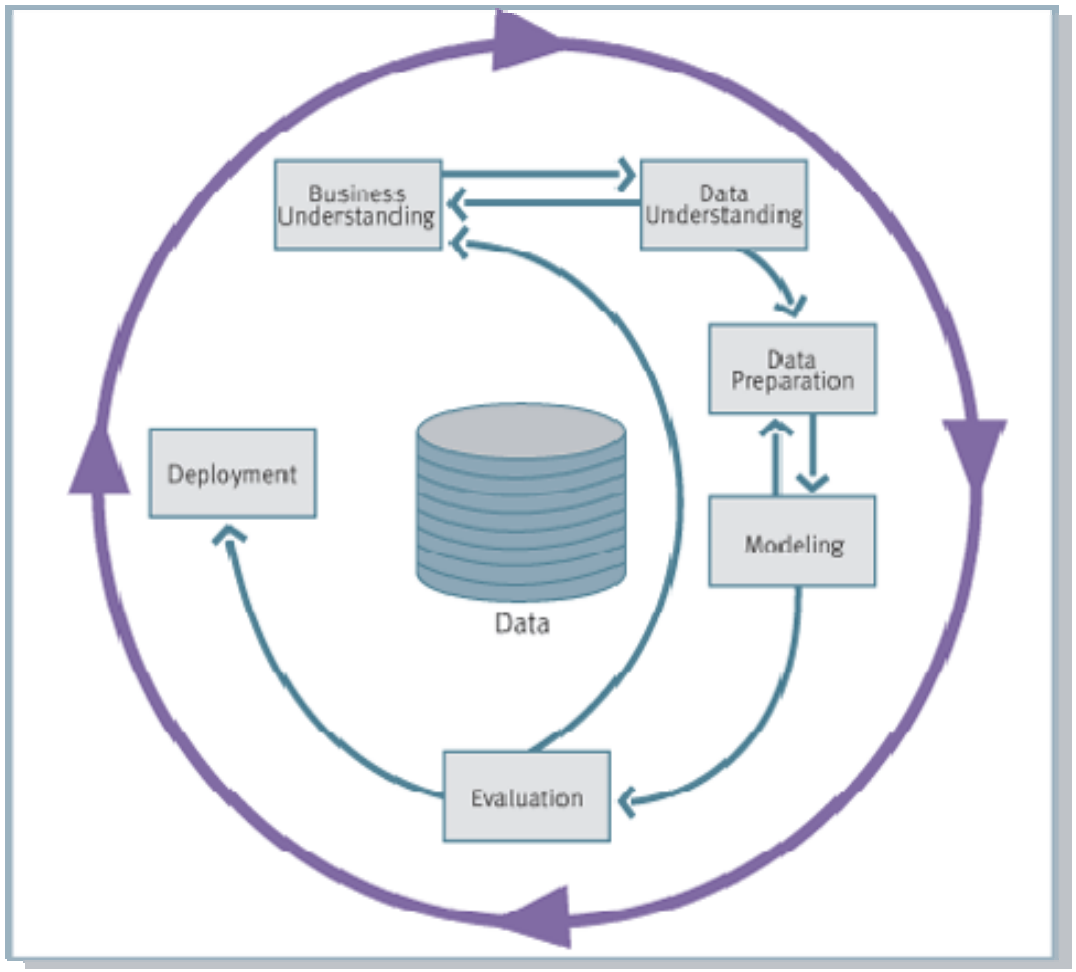
(http://www.crisp-dm.org)

Motivation

- Zusammenarbeit von NCR, SPSS und DaimlerChrysler
- NCR: Mehrwert für Data Warehouse Kunden
- SPSS: Konzept für Data Mining Produkt 'Clementine'
- DaimlerChrysler: Praktische Erfahrung
- KEINE theoretische, akademische Entwicklung,
- SONDERN Entwicklung aus praktischer Erfahrung an realen Problemen.

+++ Eigene langjährige Erfahrung bei CIBA(-Geigy)

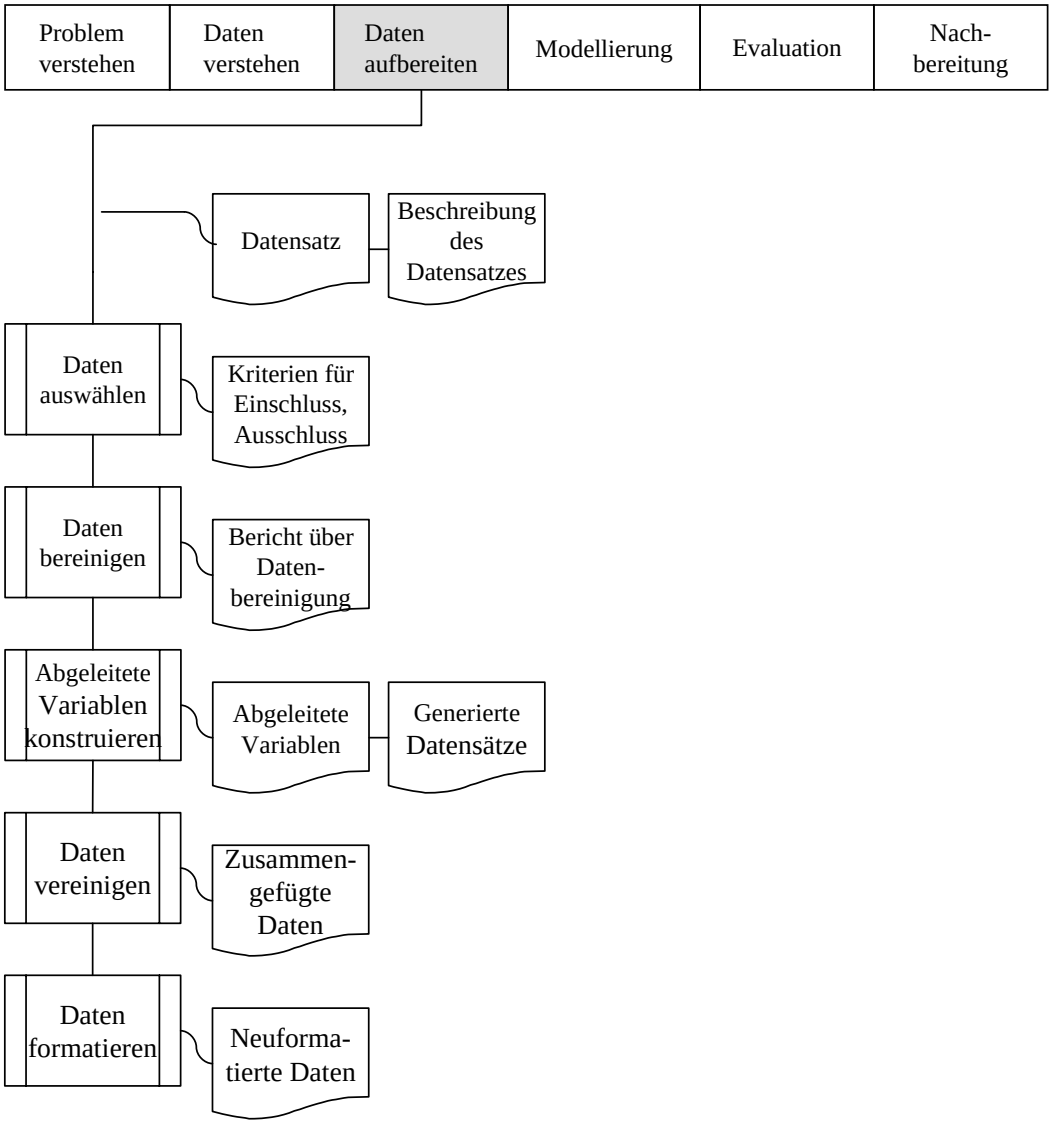
CRISP-DM: Übersicht



CRISP-DM: Schritte

- **Problem verstehen**: Analyseziele, Situationsbewertung, Datenanalyseziele, Projektplan
- **Daten verstehen**: Sammeln, beschreiben, untersuchen, Qualität von Rohdaten
- **Daten aufbereiten**: Ein- und Ausschluss, Bereinigung, Transformation von Variablen
- **Modellierung**: Methoden- und Testdesignwahl, Schätzung, Modellqualität
- **Evaluierung**: Modell akzeptieren, Prozess überprüfen, nächste Schritte
- **Nachbereitung**: Anwendungs- und Wartungsplan, Präsentation, Bericht

CRISP-DM: Datenvorbereitung



CRISP: Beispiel: Verteilungsstudien: Problemanalyse

- **Problem**: Charakterisierung der Verteilung von Medikamentenklassen im Körper
- **Studientyp**: Verteilung von 14C-markierten Substanzen in Ratten 5-6 Minuten nach intravenöser Injektion.
- **Rohdaten**: 26 Experimente, 20 Substanzen, 3-4 Ratten / Experiment, 85 Ratten, 24 Organe, 6 physico-chemische Eigenschaften: 3 Säure-Konstanten (pka), Lipophilität (log p), Molekulargewicht (WE), Wasserlöslichkeit (SO)
- **Datenanalyseziel**: Finden von typischen Verteilungsmustern für Medikamentenklassen auf der Basis von physico-chemischen Eigenschaften
- **Projektplan**: Deskription, Ersetzen fehlender Werte, Klassifikationregeln

Verteilungsstudien: Datenaufbereitung

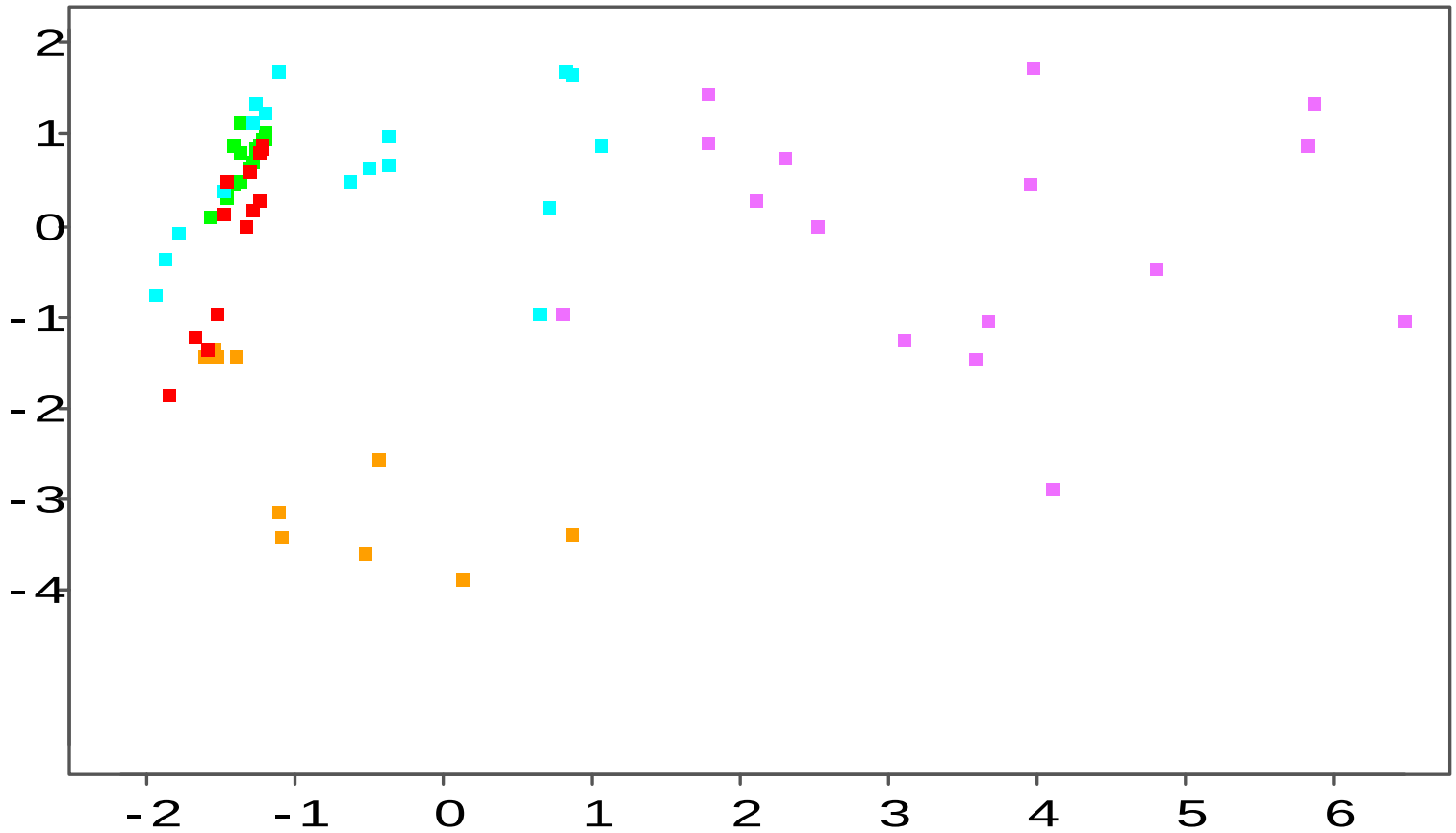
- **Univariate Analyse**: Viele fehlende Werte, insbesondere wegen nicht definierten Werten bei pka
- **Bivariate Analyse**: Niveau von 14C-Konzentrationen variiert mit Substanz-Dosen
- **Transformation 1**: Normalisierung der 14C-Werte mit Blut := 1
- **Transformation 2**: Bilden von physico-chemischen Klassen (z.B. Säuregehaltsklassen mit wohldefinierten pka-Werten)
- **Transformation 3**: Bilden von Therapieklassen für die Medikamente
- **Variablenselektion**: Weglassen der Verdauungsorgane $\longrightarrow$ 20 Organe
- **Beobachtungsselektion**: 78 vollständige Beobachtungen

Verteilungsstudien: Klassifikation

- **Therapieklassen**: Klassifikationsregel bei Linearer Diskriminanzanalyse mit 6 Organen
- **Therapieklassen**: Neuroleptica / Antidepressiva, Betablocker / Ca-Antagonisten, alle anderen Klassen zusammen sehr gut getrennt mit 96% Richtigkeit
- **Säuregehaltsklassen**: LDA mit 95% Richtigkeit
- **Säuregehaltsklassen**: RDA (Regularisierte DA) nach Box-Cox-Transformation mit 99% Richtigkeit
- **Säuregehaltsklassen**: Prognosefähigkeit genauso gut (Kreuzvalidierung)

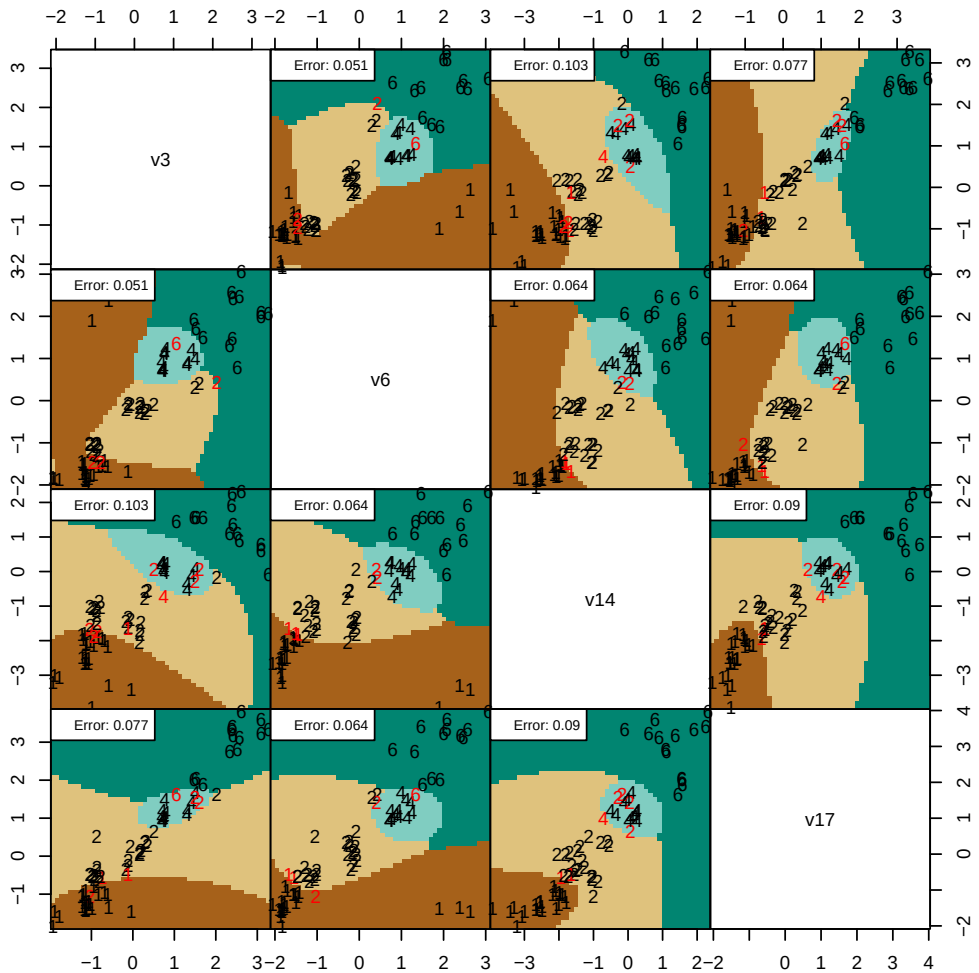
Verteilungsstudien: Therapieklassen

Trennung der Therapieklassen mit 2 Diskriminanzkomponenten aus 6 Organen



Verteilungsstudien: Säuregehaltsklassen

Fehlerraten bei 2-Organ-Kombinationen



Datenanalyse – generische Aufgabe

Population: Eine Menge von Objekten, um die es geht.

Merkmale: Eine Menge von Variablen (quantitativ oder qualitativ) beschreibt die Objekte.

Ausgabe: Ein quantitativer Wert (Messwert) oder ein qualitativer gehört zu jeder Beobachtung (Zielvariable).

Ein Lernverfahren findet eine Funktion, die Objekten einen Ausgabewert zuordnet. Oft **minimiert** die Funktion einen **Fehler**.

Modell: Das Lernergebnis (die gelernte Funktion) wird auch als *Modell* bezeichnet.

Notation

ExampleSet

Meta Data View | Data View | Plot View

ExampleSet (14 examples, 1 special attribute, 4 regular attributes)

row no.	Play	Outlook	Temperat...	Humidity	Wind
1	no	sunny	85	85	false
2	no	sunny	80	90	true
3	yes	overcast	83	78	false
4	yes	rain	70	96	false
5	yes	rain	68	80	false
6	no	rain	65	70	true
7	yes	overcast	64	65	true
8	no	sunny	72	95	false
9	yes	sunny	69	70	false
10	yes	rain	75	80	false
11	yes	sunny	75	70	true
12	yes	overcast	72	90	true
13	yes	overcast	81	75	false
14	no	rain	71	80	true

- Der Raum möglicher Beobachtungen wird als p-dimensionale Zufallsvariable X geschrieben.
- Jede Dimension der Beobachtungen wird als X_i notiert (Merkmal).
- Die einzelnen Beobachtungen werden als $\vec{x_1}, ..., \vec{x_N}$ notiert.
- Die Zufallsvariable Y ist die Ausgabe (Zielvariable).
- N Beobachtungen von Vektoren mit p Komponenten ergeben eine $N \times p$-Matrix.

Lernaufgabe Clustering

Gegeben

- eine Menge $\mathcal{T} = \{\vec{x_1}, ..., \vec{x_N}\} \subset X$ von Beobachtungen,
- eine Anzahl K zu findender Gruppen $C_1, ..., C_K$,
- eine Abstandsfunktion $d(\vec{x}, \vec{x'})$ und
- eine Qualitätsfunktion.

Finde

- Gruppen $C_1, ..., C_K$, so dass
- alle $\vec{x} \in X$ einer Gruppe zugeordnet sind und
- die Qualitätsfunktion optimiert wird: Der Abstand zwischen Beobachtungen der selben Gruppe soll minimal sein; der Abstand zwischen den Gruppen soll maximal sein.

Lernaufgabe Klassifikation

Gegeben

- Klassen Y, oft $y \in \{+1, -1\}$,
- eine Menge $\mathcal{T} = \{(\vec{x_1}, y_1), ..., (\vec{x_N}, y_N)\} \subset X \times Y$ von Beispielen,
- eine Qualitätsfunktion.

Finde

- eine Funktion $f : X \rightarrow Y$, die die Qualitätsfunktion optimiert.

Lernaufgabe Regression

Gegeben

- Zielwerte Y mit Werten $y \in \mathcal{R}$,
- eine Menge $\mathcal{T} = \{(\vec{x_1}, y_1), ..., (\vec{x_N}, y_N)\} \subset X \times Y$ von Beispielen,
- eine Qualitätsfunktion.

Finde

- eine Funktion $f : X \rightarrow Y$, die die Qualitätsfunktion optimiert.

Funktionsapproximation

Wir schätzen die wahre, den Beispielen unterliegende Funktion. Gegeben

- eine Menge von Beispielen $\mathcal{T} = \{(\vec{x_1}, y_1), ..., (\vec{x_N}, y_N)\} \subset X \times Y$,
- eine Klasse zulässiger Funktionen f_θ (Hypothesensprache),
- eine Qualitätsfunktion,
- eine feste, unbekannte Wahrscheinlichkeitsverteilung $P(X)$.

Finde

- eine Funktion $f_\theta : X \rightarrow Y$, die die Qualitätsfunktion optimiert.

Zur Erinnerung: Verteilung

Wird in der nächsten Vorlesung wiederholt!!

Wenn wir die Verteilung kennen, können wir eine gute Prognose machen!

- Wenn wir wissen, dass $p_i = 0,01$ ist, dann ist es nicht so schlimm, wenn wir uns bei x_i irren – wir irren uns dann selten.
- Wenn wir wissen, dass $P(Y = +1) = 0,99$ ist, dann sagen wir immer $+1$ voraus und sind in 99% der Fälle richtig. Wir haben nur ein Risiko von 1%, uns zu irren.

Qualitätsfunktion – Fehlerfunktion

Fehlerrisiko:

$$R(Y, f(X)) = \sum_{i=1}^{N} Q(y_i, \vec{x_i}) p(\vec{x_i}) \qquad (1)$$

wobei $p(\vec{x_i})$ die Wahrscheinlichkeit ist, dass das Beispiel $\vec{x_i}$ aus X gezogen wird.

Mittlerer Quadratischer Fehler:

$$MSE(Y, f(X)) = \frac{1}{N} \sum_{i=1}^{N} (y_i - f(\vec{x_i}))^2 \qquad (2)$$

Mittlerer 0-1-Verlust: $Q(Y, f(X)) = \frac{1}{N} \sum_{i=1}^{N} Q(\vec{x_i}, f)$, wobei

$$Q(y_i, f(\vec{x_i})) = \begin{cases} 0, \mathit{falls} & f(\vec{x_i}) = y \\ 1, \mathit{falls} & f(\vec{x_i}) \neq y \end{cases}$$

Problem

- Wir haben nur eine endliche Menge von Beispielen. Alle Funktionen, deren Werte durch die Beispiele verlaufen, haben einen kleinen Fehler.
- Wir wollen aber für **alle** Beobachtungen das richtige y voraussagen. Dann sind nicht mehr alle Funktionen, die auf die Beispiele gepasst haben, gut.
- Wir kennen nicht die wahre Verteilung der Beispiele.
- Wie beurteilen wir da die Qualität unseres Lernergebnisses?

Lern- und Testmenge

Wir teilen die Daten, die wir haben, auf:

Lernmenge: Einen Teil der Daten übergeben wir unserem Lernalgorithmus. Daraus lernt er seine Funktion $f(x) = \hat{y}$.

Testmenge: Bei den restlichen Daten vergleichen wir $\hat{y}$ mit y.

Aufteilung in Lern- und Testmenge

- Vielleicht haben wir zufällig aus lauter Ausnahmen gelernt und testen dann an den normalen Fällen. Um das zu vermeiden, verändern wir die Aufteilung mehrfach.

  leave-one-out: Der Algorithmus lernt aus $N - 1$ Beispielen und testet auf dem ausgelassenen. Dies wird N mal gemacht, die Fehler addiert.

- Aus Zeitgründen wollen wir den Algorithmus nicht zu oft anwenden.

  Kreuzvalidierung: Die Lernmenge wird zufällig in n Mengen aufgeteilt. Der Algorithmus lernt aus $n - 1$ Mengen und testet auf der ausgelassenen Menge. Dies wird n mal gemacht.

Kreuzvalidierung

- Man teile alle verfügbaren Beispiele in n Mengen auf. z.B. $n = 10$.
- Für i=1 bis i=n:
  - Wähle die i-te Menge als Testmenge,
  - die restlichen $n - 1$ Mengen als Lernmenge.
  - Messe die Qualität auf der Testmenge.
- Bilde das Mittel der gemessenen Qualität über allen n Lernläufen. Das Ergebnis gibt die Qualität des Lernergebnisses an.

Fragestellungen des maschinellen Lernens / Statistik

- Welche Zusicherungen kann ich meinen Kunden geben? (Fehlerschranken)
- Wieviele Beispiele brauche ich?
- Welche Eigenschaften sollen die Beispiele haben, um gut vorherzusagen und wie finde (erzeuge) ich sie?
- Welche Modellklasse soll ich wählen?
- Welcher Algorithmus wird mit vielen Beispielen und vielen Dimensionen in kurzer Zeit fertig?

Was wissen Sie jetzt?

- Sie haben das CRISP kennengelernt, das den gesamten Ablauf der Wissensentdeckung beschreibt.
- Als Aufgaben der Modellbildung haben Sie **Clustering, Klassifikation, Regression** gesehen.
- Sie wissen, was die **Kreuzvalidierung** ist.

Was wissen Sie noch nicht?

- Es gibt viele verschiedene **Modellklassen**. Damit werden die Lernaufgaben spezialisiert.
- Es gibt unterschiedliche **Qualitätsfunktionen**. Damit werden die Lernaufgaben als Optimierungsaufgaben definiert.

Themen

- statistische Grundbegriffe
- lineare Modelle
- Klassifikation
- Entscheidungsbäume
- Datengenerierung: Versuchsplanung, Stichproben
- Stützvektormethode (SVM) und strukturelle Risikominimierung
- stetige Modelle
- Zeitreihen
- Clustering
- Finden häufiger Mengen

Vorlesungen und Übungen 1

Vorlesung			Ausgabe	Abgabe	Übung		
14.4.	Übersicht, Einführung, Gruppen	Mo,We	Blatt 1 (Statistik)			Marco &	
16.4.	Statistik (E-Wert, Var., Cov., Dichte,	We			17.4.	Julia	Software-Einführung
21.4.	Lineare Modelle 1	Mo	Blatt 2 (lineare Modelle)	Blatt 1			
23.4.	Lineare Modelle 2	Mo			24.4.	Julia	Blatt 1
28.4.	Lineare Modelle 3	Mo	Blatt 3 (lineare Modelle)	Blatt 2			
30.4.	Klassifikation 1	We			1.5.	**Feiertag**	
05.5.	Klassifikation 2	We	Blatt 4 (Klassifikation)	Blatt 3			
07.5.	Klassifikation 3	We			8.5.	Marco	Blatt 2 und 3
12.5.	Versuchplanung, Stichproben	We	Blatt 5 (VP, Stichproben)	Blatt 4			
14.5.	Bäume	Mo			16.5.	Julia	Blatt 4
19.5.	random forests, bagging, boosting	We	Blatt 6 (Bäume, random forests)	Blatt 5			
21.5.	**Feiertag**				22.5.	Julia	Blatt 5
26.5.	SVM 1	Mo	Blatt 7 (SVMs)	Blatt 6			
28.5.	SVM 2	Mo			29.5.	Marco	Blatt 6
Pfingsten							

Vorlesungen und Übungen 2

09.6.	SVM 3	Mo	Blatt 8 (SVMs)	Blatt 7			
11.6.	**Feiertag**				12.6.	Marco	Blatt 7
16.6.	SVM 4	Mo	Blatt 9 (SVMs)	Blatt 8			
18.6.	SVM 5	Mo			19.6.	Marco	Blatt 8
23.6.	stetige Modelle 1	We	Blatt 10 (stetige Modelle)	Blatt 9			
25.6.	stetige Modelle 2	We			26.6.	Marco	Blatt 9
30.6.	stetige Modelle 3	We	Blatt 11 (stetige Modelle & Zeitreihen)	Blatt 10			
02.7.	Zeitreihen 1	We			3.7.	Julia	Blatt 10
07.7.	Zeitreihen 2	We	Blatt 12 (Zeitreihen & MCMC)	Blatt 11			
09.7.	MCMC	We			10.7.	Julia	Blatt 11
14.7.	Clustern	Mo	Blatt 13	Blatt 12			
16.7.	APRIORI	Mo			17.7.	Julia	Blatt 12
21.7.	FPGrowth	Mo		Blatt 13			
23.7.	Zusammenfassung, Rückblick	Mo,We			24.7.	Marco	Blatt 13

Übungen

Julia Schiffner und Marco Stolpe betreuen die Übungen und stehen auch für Fragen zur Verfügung.
Wir verwenden das System RapidMiner und können damit

- (fast) alle Vorverarbeitungsschritte und
- Verfahren und
- Validierungen der Ergebnisse durchführen.

Außerdem verwenden wir R, das Funktionen anbietet für

- (fast) alle Vorverarbeitungsschritte und
- Verfahren und
- Validierungsmethoden.

Wofür bekommen Sie einen Schein?

- Kommen Sie in jede Vorlesung – dann können Sie auch das Tempo bestimmen und Fragen stellen.
- Gehen Sie in die Übungsgruppe! Sie dürfen nur max. 2 mal unentschuldigt fehlen.
- Lösen Sie jede Übungsaufgabe:
  - Werden 50% der Punkte erreicht,
  - höchstens 3 Blätter nicht abgegeben und
  - mindestens eine Aufgabe in der Übung vorgerechnet

  bekommt man einen Schein.
- Nutzen Sie die Vorlesung/Übung zur Vorbereitung auf eine Fachprüfung!

Wir sehen uns...

In der ersten Übung werden RapidMiner und R vorgestellt. Sie findet statt:
Am Freitag 17.4.2009 In GB IV (Campus Süd) Raum 113
Gruppeneinteilung JETZT!

Literatur

Trevor Hastie, Robert Tibshirani, and Jerome Friedman.
The Elements of Statistical Learning: Data Mining, Inference, and Prediction.
Springer series in statistics. Springer, New York, USA, 2001.

Gerald Teschl and Susanne Teschl.
Mathematik für Informatiker.
Springer, 2006.