

# › Data Science Grundlagen

## Data Understanding

Prof. Dr. Carsten Lanquillon | Fakultät Wirtschaft und Verkehr | Wirtschaftsinformatik

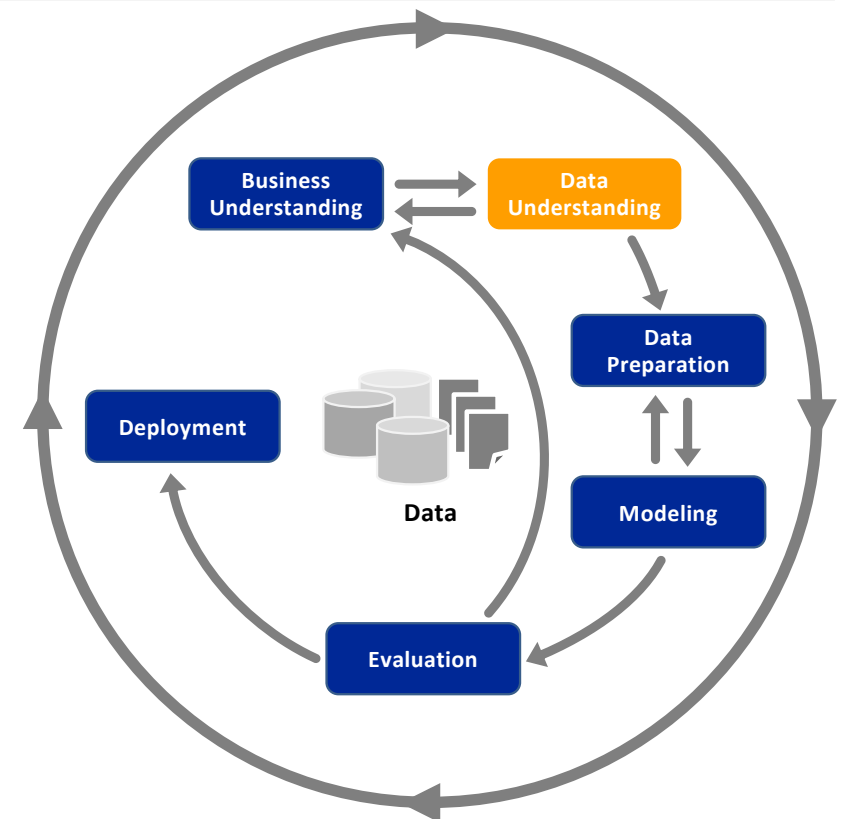
# CRISP-DM Phase 2: Data Understanding

## Aufgaben

- > Sammlung von Daten
- > Erkunden der Daten (Explorative Datenanalyse)
- > Bewertung der Datenqualität

## Erkenntnisse

- > Datenqualität ist sehr wichtig: *Garbage in, garbage out!*
- > Big Data als wesentlicher Faktor für den Erfolg
- > *The „unreasonable effectiveness of data“:*  
Mehr Daten schlagen den besseren Algorithmus!



# Überblick

---

- > Die Datenmatrix und andere Formen
- > Datentypen und Eigenschaften
- > Repräsentation von Daten
- > Die Daten erkunden und verstehen
  - Daten beschreiben und visualisieren
  - Bewertung der Datenqualität
  - Aussagekräftige Merkmale identifizieren

# Datenmatrix und Bezeichnungen

**Datenmatrix**

=

**Struktur Objekte x Merkmale**

Die meisten Data-Mining-Algorithmen erwarten eine Datenmatrix als Eingabe.

Die Datenmatrix (kurz Matrix) wird auch als Tabelle oder Datenmenge bzw. Datensatz (Data Set) bezeichnet.

Zeilen  
Objekte  
Instanzen  
Records  
Tupel  
Fälle  
Entitäten

Spalten, Attribute, Merkmale, Felder, Variablen

ID	Geschlecht	Alter	Erstattung	Einkommen	Betrug
1	männlich	45	ja	100 T	nein
2	weiblich	27	nein	80 T	nein
3	weiblich	51	nein	95 T	ja
4	männlich	32	nein	70 T	nein
5	männlich	42	ja	110 T	nein
6	männlich	37	ja	85 T	ja
7	weiblich	48	ja	105 T	nein
8	weiblich	39	nein	90 T	nein

# Daten verschiedener Art und Ausrichtung

---

## Zeilen-basiert (Records)

- > Datenmatrix
- > Transaktionen
- > Textdokumente

## Graph-basiert

- > Beziehungen in Sozialen Netzwerken
- > World Wide Web
- > Molekulare Strukturen

## Daten mit Abhängigkeiten

- > Zeitreihen
- > Sequenzen
- > Räumliche Daten
- > Geo-Daten
- > Bilder
- > Audio, Video

# Strukturiertheit

---

Strukturierte Daten

Halbstrukturierte Daten

Unstrukturierte Daten

# Von relationalen Tabellen zur Datenmatrix

## Kundendaten

- > Eine Zeile pro Kunde
- > “Snapshot” der Transaktionen beschreibt das Verhalten (viele Zeilen pro Kunde)
- > Aggregation auf Kundenebene notwendig

This column is an id field where the value is different in every column. It gets ignored for data mining purposes.

This column is from the customer information file.

This column is the target, what we want to predict.

2610000101	010377	14		A	19.1	14 Spring ...	TRUE
2610000102	103188	7		A	19.1	NULL	TRUE
2610000105	041598	1		B	21.2	71 W. 19 St.	FALSE
2610000171	040296	1		S	38.3	3562 Oak. ...	FALSE
2610000182	051990	22		C	56.1	9672 W. 142	FALSE
2610000183	111192	45		C	56.1	NULL	TRUE
2620000107	080891	6		A	19.1	P.O. Box 11	FALSE
2620000108	120398	3		D	10.0	560 Robson	TRUE
2620000220	022797	2		S	38.3	222 E. 11th	FALSE
2620000221	021797	3		A	19.1	10122 SW 9	FALSE
2620000230	060899	1		S	38.3	NULL	TRUE
2620000231	062099	10		S	38.3	RR 1729	TRUE
2620000300	032894	7		B	21.2	1920 S. 14th	FALSE

These rows have invalid customer ids, so they are ignored.

This column is summarized from transaction data.

This column is a text field with unique values. It gets ignored (although it may be used for some derived variables).

# Transaktionsdaten

- > Spezielle Form eines strukturierten Datensatzes
  - > Jede Zeile (Transaktion) besteht aus Itemmengen
  - > Speicherung in dieser Form widerspricht 1NF
- > Beispiel: Ausgewählte Artikel in einem Warenkorb

TID	Artikel (Items)
1	Brot, Milch, Saft
2	Bier, Brot
3	Bier, Milch, Saft, Windeln
4	Bier, Brot, Milch, Windeln
5	Milch, Saft, Windeln



# Text-Daten (Dokumente)

- > Einfacher Ansatz für die Verarbeitung beim Data Mining: **Bag-of-Words**
  - Jedes Dokument wird als Term-Vektor repräsentiert
  - Jede Komponente des Vektors steht für einen bestimmten Term (z.B. ein Wort)
  - Der Wert einer Komponente hängt u.a. von der Häufigkeit des Terms im Dokument ab
- > Erweiterungen berücksichtigen Zusammenhänge zwischen Wörtern: **Word Embeddings**

Giannis Antetokounmpo joined Kareem Abdul-Jabbar as the second Milwaukee Bucks player to win the Most Valuable Player award. The award was announced on Monday at a ceremony in Los Angeles.

At just 24 years old, Antetokounmpo became the third-youngest player to win MVP over the past 40 seasons, behind Derrick Rose and LeBron James. The native of Athens, Greece, is the fifth player born outside of the United States to win the award.

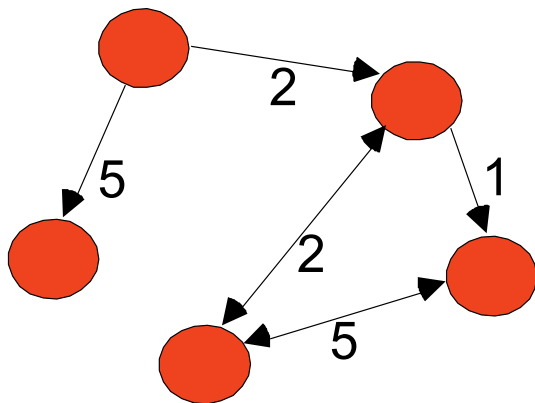
Antetokounmpo received 78 of the 101 possible first-place votes. Houston's James Harden came in second with the other 23 first-place votes. Every voter had either Antetokounmpo or Harden No. 1 or No. 2 on their ballot.



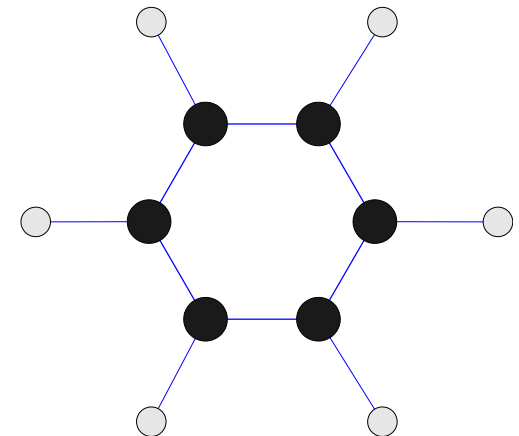
	team	coach	play	ball	score	game	win	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
⋮										
Document n	0	1	0	0	1	2	2	0	3	0

# Graph-Daten

Beispiele: Allgemeiner Graph, HTML-Seiten und Molekülstrukturen



```
<a href="papers/papers.html#bbbb">
Data Mining </a>
<li>
<a href="papers/papers.html#aaaa">
Graph Partitioning </a>
<li>
<a href="papers/papers.html#aaaa">
Parallel Solution of Sparse Linear System of Equations </a>
<li>
<a href="papers/papers.html#ffff">
N-Body Computation and Dense Linear System Solvers
```



# Bilder

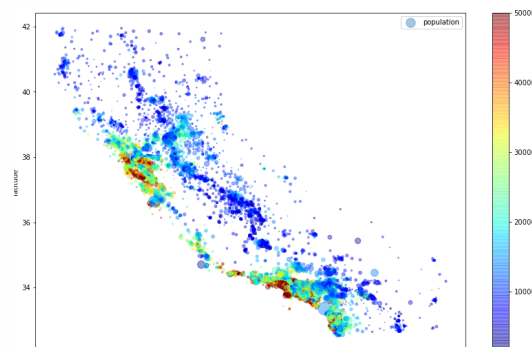
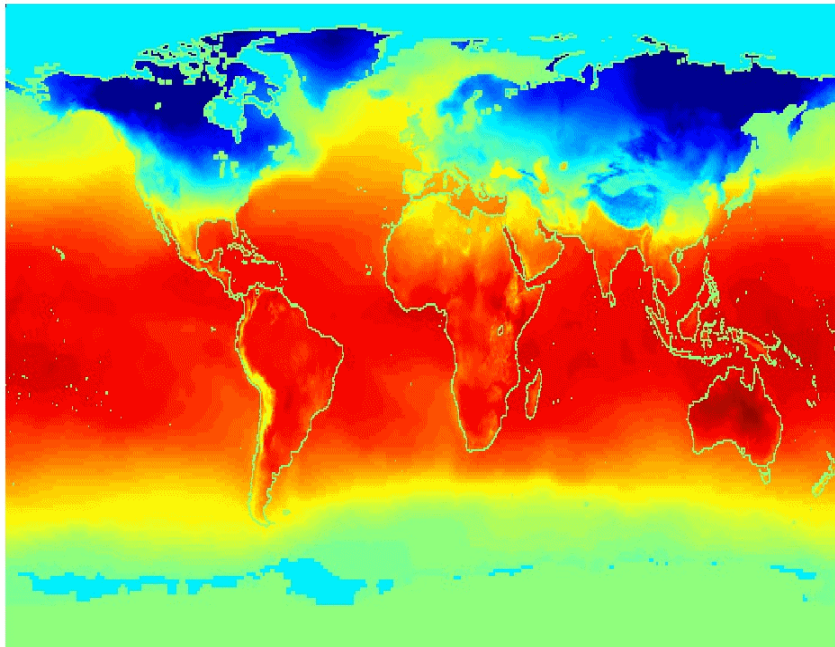
- > Bilder als Pixel-Matrix
- > Beachtung relevanter Nachbarschaftsinformationen von Punkten
- > Direkte Verwendung oft möglich (Datenmatrix wird zum Tensor)
- > Serialisierung der Pixelkodierung ergibt Datenmatrix
- > Invarianz bei Rotation, Skalierung und Verschiebung oft wichtig



1 2 3 4 5 6 7 8 9 10

# Zeitabhängige und räumliche Daten

Jan



# Beispiele für menschenlesbare Dateiformate

---

## > CSV Comma-Seperated Values

```
id,title,description,price
1,shoes,red shoes,$70.00
2,hata black hat,$20.00
3,sweater,a wool sweater,$50.00
```

## > JSON JavaScript Object Notation

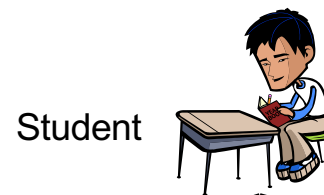
```
{"id":1, "name":"josh-shop", "listings":[1, 2, 3]}
{"id":2, "name":"provost", "listings":[4, 5, 6]}
```

## > XML eXtensible Markup Language

```
<listing id=1 title="shoes" price="$70.00">
  <description>red shoes</description>
</listing>
<listing id=2 title="hat" price="$20.00">
  <description>black hat</description>
</listing>
<listing id=3 title="sweater" price="$50.00">
  <description>a wool sweater</description>
</listing>
```

# Grundbegriffe aus der Statistik

## Merkmalsträger



## Merkmal

Geschlecht

Zufriedenheit mit  
Studiengangswahl

Alter  
[in Jahren]

## Merkmalsausprägung

  
männlich/  
weiblich

gar nicht  sehr  
zufrieden

0  
1  
2  
⋮

## Skalenniveau



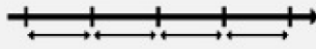
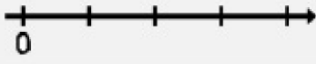
nominal

ordinal

metrisch

# Skalenniveaus: Übersicht

Das Skalenniveau eines Merkmals beschreibt die Art und den Gehalt der Informationen der Ausprägungen und die Vergleichsmöglichkeiten (Beziehungen) untereinander

Merkmalsart	Relation zwischen den Merkmalsausprägungen	Skalierung	Beispiele
<b>qualitatives Merkmal</b>	Verschiedenheit $x_i \neq x_j$	Nominalskala 	Familienstand, Geschlecht, Beruf, Postleitzahl
<b>komparatives Merkmal</b>	Rangfolge $x_i < x_j$	Ordinalskala 	Handelsklasse, Schulnoten, Rating-Urteile
<b>quantitatives Merkmal</b>	Abstände $(x_i - x_j)$ sinnvoll	Intervallskala 	Temperatur in [°C], Geburtsjahrgang
	Verhältnisse $(x_i : x_j)$ sinnvoll	Verhältnisskala 	Preis, Umsatz, Einkommen, Alter

# Die Skalenniveaus sind ordinalskaliert

---

- > Die verschiedenen Skalenniveaus stellen eine Hierarchie dar, die von niedrigsten Nominalskala bis zur Verhältnisskala reicht.
- > Merkmale, die auf einer hohen Skala gemessen wurden, können so transformiert werden, dass ihre Ausprägungen niedriger skaliert sind (**Skalentransformation**).
- > Methoden für ein niedriges Skalenniveau können unter Verlust an Information auch für ein höheres Skalenniveau verwendet werden. Die **Umkehrung** gilt allerdings **nicht**.
- > **Beispiel für Skalentransformationen:**
  - „Alter“ in Jahren anstatt des genauen Geburtsdatums
  - Abbildung auf Altersklassen wie *jung, mittel, alt*



# Beziehung zwischen Repräsentation und Skalenniveau

---

- > Zur Verarbeitung und Speicherung von Daten werden qualitative Merkmale oft codiert.
  - > Häufig wird dafür eine effiziente Codierung durch Zahlen verwendet.
  - > **Beispiel**  
Codierung für „**Familienstand**“: ledig → 0, verheiratet → 1, geschieden → 2, verwitwet → 3
  - > **Anmerkung**
    - > Nur aufgrund der Tatsache, dass Merkmalsausprägungen durch Zahlen dargestellt sind, kann nicht zwingend auf eine zu Grunde liegende Ordinalskala oder Kardinalskala geschlossen werden.
    - > Die Bestimmung des Skalenniveaus für ein Merkmal erfordert die Berücksichtigung inhaltlicher Aspekte.
- **Bestimmung und Festlegung der tatsächlichen Skalenniveaus aller verwendeten Merkmale im Datensatz ist Voraussetzung für valide und erfolgreiche Ergebnisse!**

# Weitere Eigenschaften von Daten

---

- > **Dimensionalität**
  - > Umgang mit vielen Dimensionen oft schwierig → „curse of dimensionality“
- > **Spärlichkeit (Sparseness)**
  - > Wie werden seltene Ereignisse gespeichert? → Speicherbedarf vs. einfacher Zugriff
- > **Granularität (Auflösung, Resolution)**
  - > Ergebnisse hängen stark von der Auflösung und dem Skalenniveau ab
- > **Schiefe der Klassenverteilung**
  - > Seltene und schiefe Klassen sind schwerer zu lernen

# Explorative Datenanalyse

---

## Daten beschreiben

1. Welche Datentypen existieren
2. Gibt es fehlende Daten
3. Können die Daten die Frage beantworten

## Daten erkunden

- Zielattribut festlegen
- Exploration einzelner Variablen
- Paarweise Erkundung
- Datenqualität
- Ausreißer identifizieren
- Korrelationen

## Daten visualisieren

- Visualisierungen zur Unterstützung des Verständnisses

# Schritte zum Datenverständnis

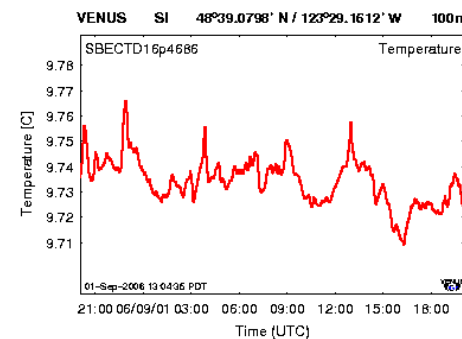
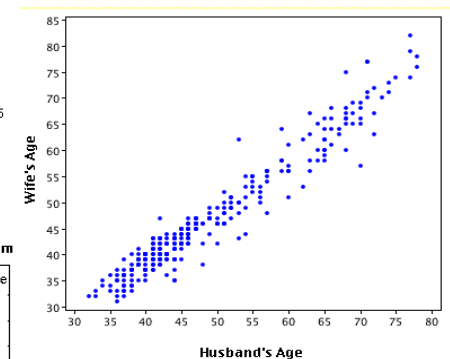
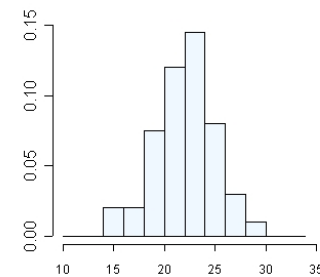
---

- > Daten beschreiben: „Oberfläche“ Eigenschaften und mit Metadaten abgleichen
  - Verfügbare Merkmale und deren Bedeutung, ggf. Identifikation Zielgröße
  - Format (Repräsentation), Skalenniveau, Ausprägungen (Werte), Häufigkeiten
  - Beispiele anschauen
  
- > Daten erkunden
  - SQL-Abfragen, Berichte, Visualisierungen, Profiling, Verteilungen
  - Datenqualitätsprobleme identifizieren
  - Erste Erkenntnisse gewinnen (einfache Zusammenhänge)

# Datenvisualisierung

## Standarddiagramme

- > Histogramm (bevorzugt für metrische Merkmale)
- > Balkendiagramme (bevorzugt für qualitative Merkmale)
- > Kreisdiagramme (möglichst vermeiden)
- > Streudiagramme
- > Zeitreihen
- > Karten



# Datenqualität

---

- > Datenqualität bezieht sich auf Einsatzzweck: „fit for purpose“
  - Beim Data Mining wurden die Daten i.d.R. für einen anderen Einsatzzweck erhoben!
- > Datenqualität ist elementar für erfolgreiches Data Mining: „garbage in, garbage out“
  - Datenqualität sollte immer hinterfragt und kritisch geprüft werden!
- > Fragen
  - > Nach welchen Kriterien wird die Qualität von Daten beschrieben?
  - > Mit welchen Kennzahlen lassen sich die Datenqualitätskriterien messen?
  - > Wie lassen sich Datenqualitätsprobleme beheben?

# Typische Datenqualitätskriterien (Auswahl)

---

<b>Vollständigkeit</b>	Sind alle notwendigen Daten verfügbar und zugreifbar?
<b>Konformität</b>	Stimmen die Werte mit den erwarteten Wertebereichen und Formaten überein?
<b>Konsistenz</b>	Sind die Daten zwischen verschiedenen Systemen und Tabellen konsistent?
<b>Genauigkeit</b>	Decken sich die Daten mit den realen Objekten bzw. mit Daten einer verlässlichen Quelle?
<b>Eindeutigkeit</b>	Gibt es Duplikate?
<b>Integrität</b>	Sind die Beziehungen zwischen Objekten in den Daten konsistent?
<b>Rechtzeitigkeit</b>	Sind die Daten verfügbar, wenn sie benötigt werden?

# Typische Gründe für schlechte Datenqualität

---

- > Menschen (Fehler bei der Dateneingabe)
- > Architektur
- > Prozesse
- > Definitionen (Metadaten)
- > System-/ Datennutzung nicht im Einklang mit dem Entwurf/Konzept
- > Verfall der Genauigkeit oder Gültigkeit mit der Zeit



# Typische Datenqualitätsprobleme

---

Viele Datenqualitätskriterien lassen sich ohne Hintergrundwissen oder genaue Kenntnis der Objekte aus der realen Welt nicht messen.

Technische Aspekte, die auf Datenqualitätsprobleme hinweisen können

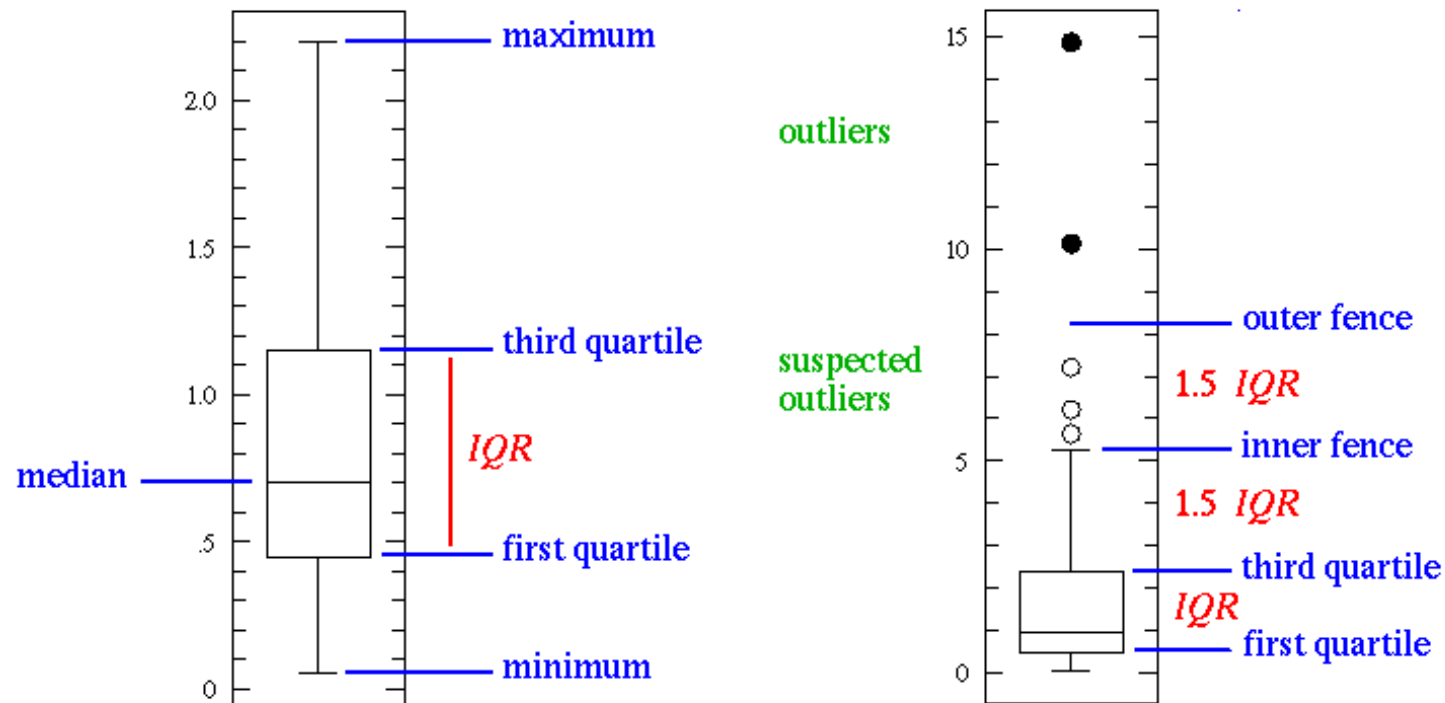
- > Merkmale sind veraltet oder redundant
- > Fehlende Werte
- > Ausreißer
- > Rauschen (Noise)
- > Werte decken sich nicht mit Vorgaben oder gesundem Menschenverstand

→ Datenbereinigung ist meist ein sehr aufwendig und langwierig (Schätzung: bis zu 80% der Zeit)

# Beispiele für Datenqualitätsprobleme

KID	PLZ	Geschlecht	Einkommen	Alter	Familienstand	Betrag
1001	10048	M	75000	C	M	5000
1002	J2S7K7	F	- 40000	40	W	4000
1003	90210		10000000	45	S	7000
1004	6269	M	50000	0	S	1000
1005	55101	F	99999		M	4000
1005	55101	F	65000	30	D	4000

# Box-Plots und Ausreißer nach Tukey



# Ausreißer bei Normalverteilungsannahme

---

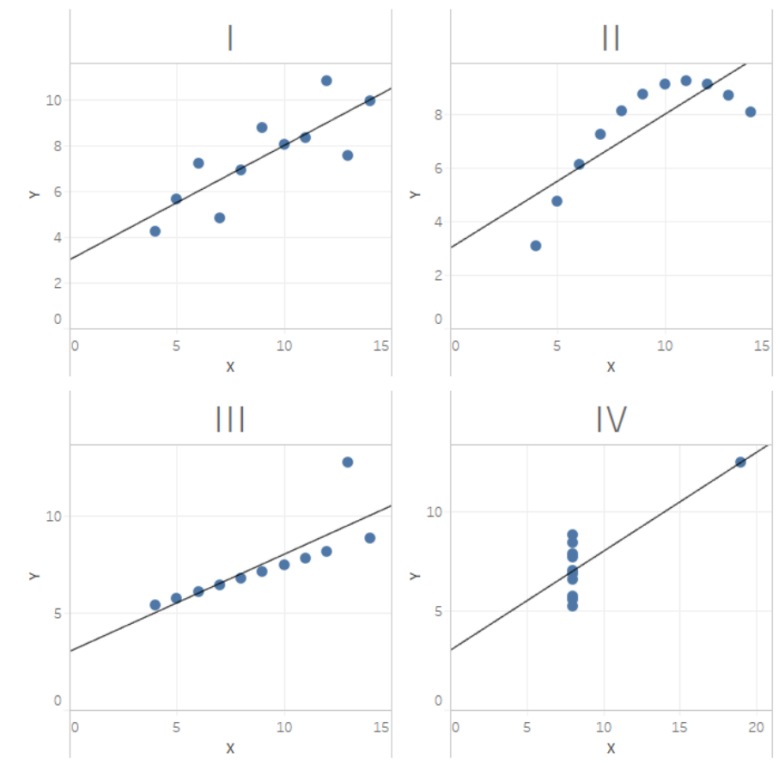
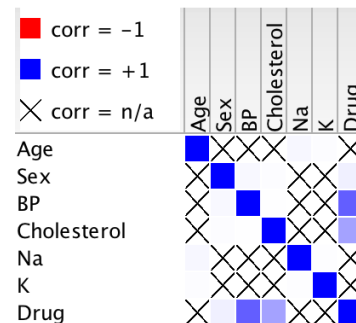
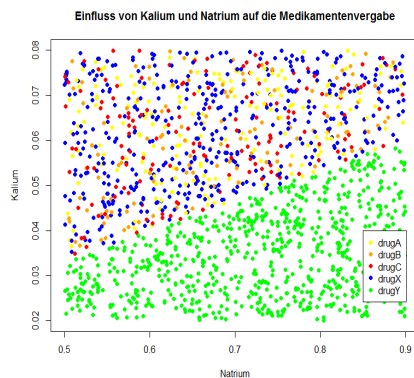
Arithmetischer Mittelwert  $\mu = \frac{1}{n} \sum_{i=1}^n x_i$       Stichprobenvarianz  $\sigma^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$

## **k·σ -Regel**

- > Untere Grenze:  $\mu - k \cdot \sigma$
- > Obere Grenze:  $\mu + k \cdot \sigma$
- > Beobachtungen außerhalb der Grenzen werden als Ausreißer betrachtet
- > Werte für k
  - k=3:  $\mu \pm 3 \cdot \sigma$  → 0,2% der Beobachtungen sind Ausreißer (bei Normalverteilungsannahme)
  - k=6:  $\mu \pm 6 \cdot \sigma$  → 0,0001% der Beobachtungen sind Ausreißer (bei Normalverteilungsannahme) → „six sigma“

# Aussagekräftige Merkmale identifizieren

- > Zusammenhänge zwischen beschreibenden Merkmalen und Zielgröße visualisieren (Zielgröße z.B. farblich kennzeichnen)
- > Zusammenhänge analytisch mit geeigneten Kennzahlen je nach Skalenniveau bewerten



# Zusammenfassung

---

- > Datenqualität bezieht sich auf Einsatzzweck: „fit for purpose“
- > Datenqualität ist immer wichtig (GIGO)
- > Datenqualitätsprobleme so früh wie möglich erkennen
- > Überblick mit Kennzahlen und Visualisierungen
- > Ausreißer und fehlende Werte beachten
- > Erwartete Abhängigkeiten und Zusammenhänge überprüfen
- > Explorative Datenanalyse