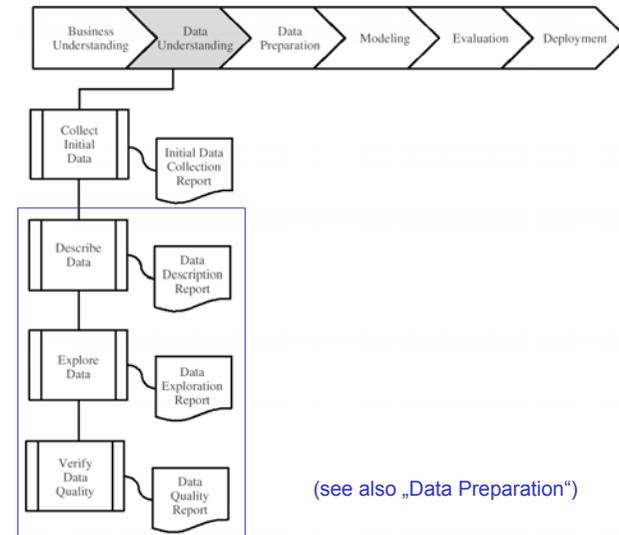


## Kapitel III

# Vertrautmachen mit Daten

## III Vertrautmachen mit Daten



## III Vertrautmachen mit Daten

### III.1 OLAP

#### III.1.1 Einführung in OLAP

Wie gesehen, gibt es große Unterschiede zwischen operativen Systemen und dem DWh

Entsprechend gibt es fundamentale Unterschiede auch zwischen den jeweiligen Zugriffsarten auf diese Datenquellen:

- **OLAP = On-Line Analytical Processing** benutzt DWh
- **OLTP = On-Line Transaction Processing** benutzt operative Systeme

## III.1.1 Einführung in OLAP

### OLTP

- hohe Zahl **kurzer**, atomarer, isolierter, wiederkehrender Transaktionen
  - z.B. Konto-Update, Flugbuchung, Telefon-Gespräch
- Transaktionen benötigen detaillierte, aktuelle Daten
- Daten werden (oft tupelweise) gelesen und relativ **häufig aktualisiert**
- Transaktionen dienen dem **Tagesgeschäft** und haben relativ hohe Ansprüche an die Bearbeitungsgeschwindigkeit

### III.1.1 Einführung in OLAP

#### Definition von OLAP:

- **OLAP Systeme**
  - dienen der **Entscheidungs-Unterstützung** oder
  - können in den Phasen „**Data Understanding**“ bzw. „**Data Preparation**“ im Rahmen des Data-Mining-Prozesses eingesetzt werden.
- **OLAP-Funktionen** erlauben
  - den schnellen, **interaktiven** Zugriff auf Unternehmensdaten
  - unter „beliebigen“ unternehmensrelevanten Blickwinkeln (**Dimensionen**)
  - auf verschiedenen **Aggregationsstufen**
  - mit verschiedenen Techniken der Visualisierung
- Hauptmerkmal ist die **multi-dimensionale** Sichtweise auf Daten mit flexiblen interaktiven Aggregations- bzw. Verfeinerungsfunktionen entlang einer oder mehrerer Dimensionen.

### III.1.1 Einführung in OLAP

#### Multi-Dimensionalität:

- Mehrdimensionale Sichtweise auf Daten ist sehr **natürlich**.
- Sichtweise der Analysten auf Unternehmen **ist** mehrdimensional.
  - ⇒ Konzeptuelles Datenmodell sollte mehrdimensional sein, damit Analysten leicht und intuitiv Zugang finden.
- **Beispiel:** *Verkaufszahlen* können nach unterschiedlichen Kriterien / Dimensionen aggregiert und analysiert werden.
  - nach **Produkt:** *Produkt, Produktkategorie, Industriezweig*
  - nach **Region:** *Filiale, Stadt, Bundesland*
  - nach **Zeit:** *Tag, Woche, Monat, Jahr*
  - nach verschiedenen Dimensionen des Käufers: **Alter, Geschlecht, Einkommen** des Käufers
  - und nach **beliebigen Kombinationen von Dimensionen**, z.B.
    - nach *Produktkategorie, Stadt und Monat*

### III.1.1 Einführung in OLAP

#### Kennzahlen:

- Die **Analyse-Gegenstände** von OLAP sind **numerische Werte**, typischerweise **Kennzahlen** genannt (oder auch Maße, Metriken oder Fakten).
  - **Beispiel:** *Verkaufszahlen, Umsatz, Gewinn, Lagerbestand,...*
- Diese numerischen Werte lassen sich auf verschiedene Weise verdichten, z.B.
  - Summenbildung
  - Mittelwertbildung
  - Minimum- oder Maximumbestimmung
- Die zulässige Art der Verdichtung hängt vom **Skalenniveau** der Kennzahl ab.

### III.1.1 Einführung in OLAP

#### Skalenniveaus

In der Statistik unterscheidet man die Attributausprägungen einer vorgegebenen Menge von Daten mittels Skalen mit unterschiedlichem Skalenniveau. Die wichtigsten Typen sind:

#### **Nominalskalierte Merkmale:**

Ausprägungen sind "Namen", keine Ordnung möglich  
→ keine Aggregation möglich

#### **Ordinalskalierte Merkmale:**

Ausprägungen können geordnet, aber Abstände nicht interpretiert werden.  
→ Median macht Sinn, Mittelwert z.B. nicht

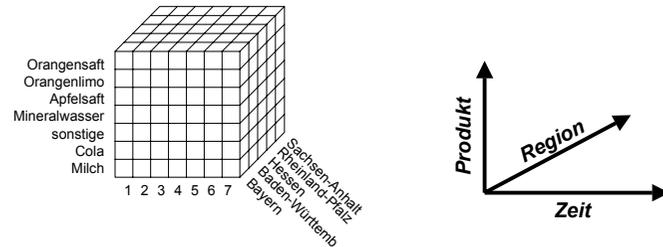
#### **Kardinalskalierte Merkmale:**

Ausprägungen sind Zahlen, Interpretation der Abstände möglich (metrisch)  
→ Mittelwertbildung, Standardabweichung etc. sinnvoll

### III.1.1 Einführung in OLAP

#### Dimensionen:

- Jede Kennzahl hängt von einer Menge von **Dimensionen** ab. Diese bilden den **Kontext der Kennzahlen**.
  - **Beispiel:** Die *Verkaufszahlen* (Kennzahl) hängen von den Dimensionen *Produkt*, *Region* und *Zeit* ab.
  - Die Dimensionen sind **orthogonal (unabhängig)**.
  - Sie definieren einen sog. **Hyper-Würfel (hyper cube)**.



- Es kann eine beliebige Zahl an Dimensionen geben (abhängig vom Zweck des OLAP-Systems und der enthaltenen Daten). In manchen Anwendungen treten bis zu 50 Dimensionen auf.

### III.1.1 Einführung in OLAP

#### Dimension Zeit:

- **Spezielle Dimension**, die in jedem OLAP-System existiert, ist die **Zeit**.
- Leistung eines Unternehmens wird immer anhand der Zeit bewertet:
  - aktueller Monat im Vergleich zu letztem Monat
  - aktueller Monat im Vergleich zum gleichen Monat des Vorjahres
- Dimension *Zeit* unterscheidet sich von allen anderen Dimensionen:
  - Zeit hat einen linearen Charakter:
    - Januar kommt vor Februar
  - Zeit hat Wiederholungscharakter: jeden Montag, werktags, ...
- OLAP-System muss Umgang mit der Dimension Zeit und den damit verbundenen Besonderheiten unterstützen.

#### Attribute und Attributelemente:

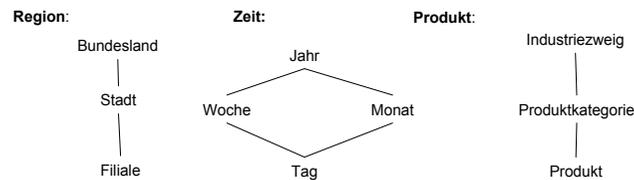
Jede Dimension ist durch eine **Menge von Attributen** charakterisiert.

- **Beispiel:** Die Dimension *Region* ist charakterisiert durch die Attribute *Filiale*, *Stadt* und *Bundesland*.

### III.1.1 Einführung in OLAP

#### Attribute und Attributelemente:

- Diese Attribute können **hierarchisch** angeordnet sein (Aggregationsstufen)
  - **Beispiel:**
    - Gesamtwert ergibt sich aus den Werten mehrerer *Bundesländer*.
    - Wert für ein *Bundesland* ergibt sich aus Werten mehrerer *Städte*.
    - Wert für eine Stadt ergibt sich aus Werten mehrerer *Filialen*.



### III.1.1 Einführung in OLAP

- Ein Pfad in einer solchen **Attribut-Hierarchie** (z.B. *Tag*, *Monat*, *Jahr*) wird auch **consolidation path** genannt.
- Jedes Attribut einer Dimension wird durch **Attributelemente** instanziiert.
  - **Beispiel:**
    - Das Attribut **Produkt** der Dimension *Produkt* hat die Attributelemente: *Coca-Cola*, *Pepsi-Cola*, *Afri-Cola*, ...
    - Das Attribut **Produktkategorie** hat die Attributelemente: *Orangensaft*, *Apfelsaft*, *Orangenlimo*, *Cola*, ...
    - Das Attribut **Industriezweig** hat die Attributelemente: *Lebensmittelindustrie*, *Textilindustrie*, *Schwerindustrie*, ...

### III.1.2 OLAP Funktionalität

#### III.1.2 OLAP Funktionalität

- Bei der Analyse können beliebige Aggregationsstufen visualisiert werden:

#### Drill-Down bzw. Roll-Up-Operationen

- Bedingungen an Dimensionen, Attribute und Attributelemente reduzieren Dimensionalität der visualisierten Daten:

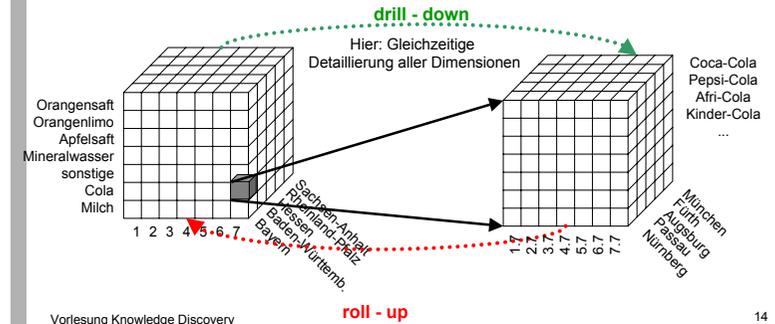
#### Slice & Dice - Operationen

- Analyse wird durch Vielzahl von **Visualisierungstechniken** unterstützt. Bedingungen werden **interaktiv** gewählt (Buttons, Menüs, *drag & drop*), so dass Analysten und Manager keine komplizierte Anfragesprache lernen müssen.

### III.1.2 OLAP-Funktionalität

#### Drill-Down und Roll-Up

- Entlang der Attribut-Hierarchien werden die Daten **verdichtet** bzw. wieder **detailliert** und sind so auf verschiedenen **Aggregationsstufen** für Analysen zugreifbar.
- Verdichtung/Detaillierung kann entlang einer, mehrerer oder aller Dimensionen geschehen - gleichzeitig oder in beliebiger Reihenfolge.



### III.1.2 OLAP-Funktionalität

#### Slice & Dice:

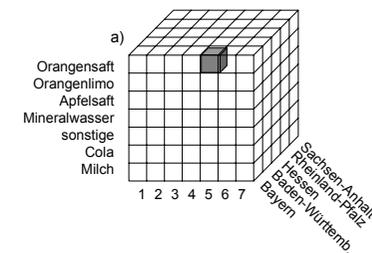
- Bei dieser Operation wird die **Dimensionalität** der visualisierten Daten **reduziert**.
- Zu einer Teilmenge der Dimensionen (sog. **page dimensions**) werden Bedingungen formuliert.
- Alle Daten in der resultierenden Tabelle genügen diesen Bedingungen.
- Die **page dimensions** tauchen in der neuen Tabelle nicht mehr explizit auf, sondern definieren implizit die Menge dargestellter Daten.

*Slice & Dice* entspricht dem Herausschneiden einer Scheibe (*slice*) aus dem Hyper-Würfel. Nur diese Scheibe wird weiterhin visualisiert.

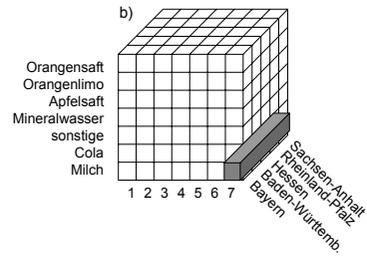
### III.1.2 OLAP-Funktionalität

#### Beispiele:

Lokation bestimmter atomarer und aggregierter Werte im Hyper-Würfel.

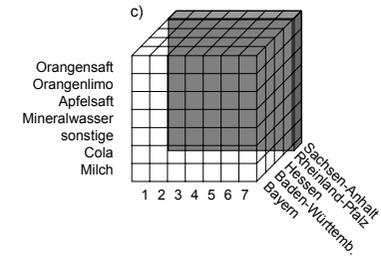


### III.1.2 OLAP-Funktionalität



b) Verkaufszahlen für Milch in ganz Süddeutschland im Juli

### III.1.2 OLAP-Funktionalität



c) Verkaufszahlen insgesamt für Sachsen-Anhalt

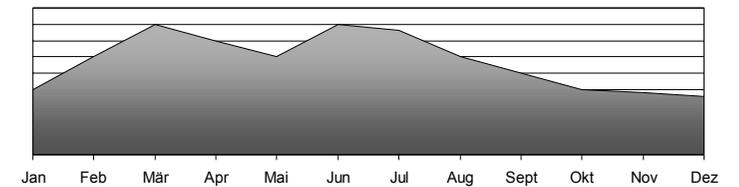
⇒ Aggregation der Verkaufszahlen über alle Monate **und** alle Produkte

### III.1.2 OLAP-Funktionalität

- Analyse bezieht sich nur selten auf einen Wert:
  - sondern auf eine Folge von Werten  
⇒ Entwicklungen und **Trends** erkennbar (d)
  - oder auf eine Menge von Werten  
⇒ Vergleiche verschiedener Werte ermöglicht (e)

### III.1.2 OLAP-Funktionalität

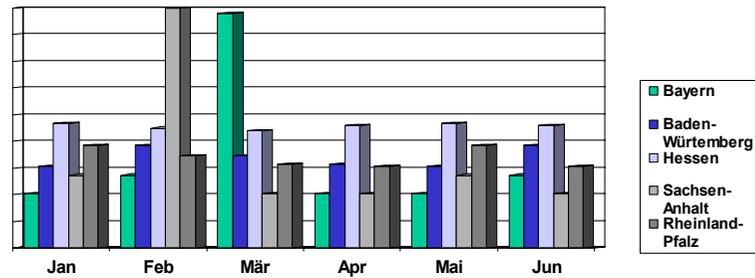
d) Entwicklung der Verkaufszahlen für Apfelsaft in Baden-Württemberg im letzten Jahr.



page dimensions: Produkt = Apfelsaft, Region = Baden-Württemberg

### III.1.2 OLAP-Funktionalität

e) Vergleich der Verkaufszahlen für Apfelsaft in den Regionen Deutschlands für das erste Halbjahr



page dimensions: Produkt = Apfelsaft

### III.1.3 Mehrdimensionales Datenmodell

#### III.1.3 Mehrdimensionales Datenmodell

Der beste Weg um zu einem OLAP-fähigen DWh zu kommen:

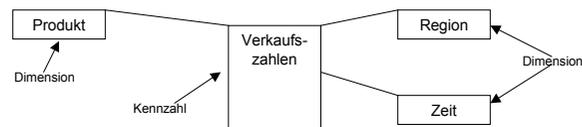
1. Erstellen eines **mehrdimensionalen** konzeptuellen Datenmodells.
2. Ableiten eines **relationalen** logischen Datenmodells.
  - Relationale DBS bilden die Implementierungsebene des DWh

#### Stern-Schema: (star schema)

- mehrdimensionales Datenmodell durch **Stern-Schema** realisierbar.
- Konstrukt eines Stern-Schemas:
  - **Kennzahlen:** Gegenstände der Analyse: Verkaufszahlen
  - **Dimensionen** definieren den Kontext der Kennzahlen: Produkt, Region, Zeit

### III.1.3 Mehrdimensionales Datenmodell

#### Beispiel:



### III.1.3 Mehrdimensionales Datenmodell

#### Vorteile des Stern-Schemas gegenüber herkömmlichen relationalen Schemata:

- Schema-Entwurf entspricht der **natürlichen Sichtweise** der Benutzer
  - Daten können in einer für Analysen adäquaten Weise zugegriffen werden.
- **Erweiterungen und Änderungen** am Schema sind leicht zu realisieren.
- **Beziehungen** zwischen den Tabellen sind **vordefiniert**
  - Join-Operationen können durch entsprechende Zugriffspfade unterstützt werden
  - Schnelle Antwortzeiten sind möglich
- Stern-Schema kann leicht in relationales DB-Schema umgesetzt werden.

### III.1.3 Mehrdimensionales Datenmodell

- Umsetzung des Stern-Schemas in relationale Tabellen:
  - **Kennzahlentabelle (major table):** Die Gegenstände der Analyse (Kennzahlen) werden in dieser Tabelle gesichert
  - **Nebentabelle (minor tables):** Jede Dimension wird zu einer eigenen Relation / Tabelle.

#### Kennzahlentabelle:

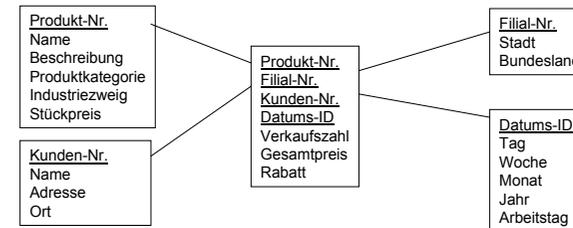
- Jedes **Tupel der Kennzahlentabelle** besteht aus
  - einem Zeiger für jede Dimensionstabelle (Fremdschlüssel), die den Kontext eindeutig definieren und
  - den numerischen Werten (**Daten**) für den jeweiligen Kontext.
- Sie enthält die eigentlichen Geschäftsdaten, die analysiert werden sollen.
- Die Kennzahlentabelle kann sehr viele Zeilen enthalten (Millionen).
- Der Schlüssel der Kennzahlentabelle wird durch die Gesamtheit der Dimensionszeiger gebildet

### III.1.3 Mehrdimensionales Datenmodell

#### Dimensionstabelle:

- Jede **Dimensionstabelle** enthält
  - einen eindeutigen Schlüssel (z.B. Produktnummer) und
  - beschreibende Daten der Dimension (**Attribute**).
- Dimensionstabellen sind deutlich kleiner als die Kennzahlentabelle.
- Zusammenhang zur Kennzahlentabelle über Schlüssel/Fremdschlüssel-Relation

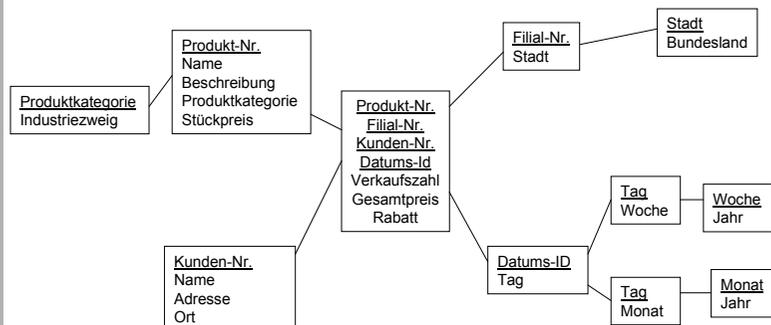
#### Beispiel: Tabellen abgeleitet aus einem Stern-Schema:



### III.1.3 Mehrdimensionales Datenmodell

#### Schneeflocken-Schema:

- Stern-Schema repräsentiert die Attribut-Hierarchien in den Dimensionen nicht explizit.
- Explizite Hierarchie kann durch sog. **Schneeflocken-Schemata (Snowflake Schema)** erreicht werden.
- **Beispiel:** Schneeflocken-Schema



### III.1.3 Mehrdimensionales Datenmodell

#### **MOLAP: Multidimensional On-Line Analytical Processing**

Spezifische Produkte für OLAP, die auf einer eigenen, proprietären mehrdimensionalen Datenbank beruhen.

Intern beruht die Datenbank auf einer Zell-Struktur, bei der jede Zelle entlang jeder Dimension identifiziert werden kann.

#### **ROLAP: Relational On-Line Analytical Processing**

Produkte, die eine multidimensionale Analyse auf einer relationalen Datenbank ermöglichen.

Sie speichern eine Menge von Beziehungen, die logisch einen mehrdimensionalen Würfel darstellen, aber physikalisch als relationale Daten abgelegt werden.

## Kapitel III.2 Visualisierung großer Datenmengen

### III.2 Visualisierung von großen Datenmengen

(Keim/Kriegel 1996)  
(Keim 1997)

#### III.2.1 Einführung

– Visualisierung kann verwendet werden für

- **explorative Datenanalyse:**
  - Ausgangspunkt: Datenbestand
  - Ziel: **datengetriebene** Bildung von **Hypothesen** durch interaktive Suche nach Strukturen / Abhängigkeiten
- **bestätigende Analyse:**
  - Ausgangspunkt: Hypothesen und Datenbestand
  - Ziel: Visualisierung, die vorgegebene Hypothesen bestätigt

## III.2 Visualisierung großer Datenmengen

- **Präsentation:**
  - Ausgangspunkt: bestätigte Zusammenhänge und Datenbestand
  - Ziel: Visualisierung der Zusammenhänge durch geeignete Visualisierungstechnik

– nachfolgend wird Aspekt der **explorativen Datenanalyse** betrachtet:

- **Vertrautmachen** mit Daten und Erkennen von Strukturen ist Voraussetzung für **Data Preparation-Phase** und für Auswahl / Anwendung von geeigneten Data Mining Algorithmen
- typische Hypothesen: funktionale Abhängigkeiten  
Datencluster

## III.2 Visualisierung großer Datenmengen

– Visualisierung kann **interaktiv** durchgeführt werden:

- Kombination menschlicher Wahrnehmungsfähigkeiten mit hoher Leistungsfähigkeit heutiger Rechner

– Visualisierungstechniken können in verschiedene Klassen eingeteilt werden:

- **Pixel-orientierte** Techniken
- **Geometrische** Techniken
- **Icon-basierte** Techniken
- Hierarchische Techniken
- Graph-basierte Techniken

## III.2.2 Pixel-orientierte Techniken

### III.2.2 Pixel-orientierte Techniken

– **Idee:**

- jeder **Attributwert** eines n-stelligen Datentupels wird als **ein farbiges Pixel** repräsentiert
- die **m Werte** eines Datentupels werden auf **m separate** Windows verteilt
- in jedem Window werden die Attributwerte eines Datentupels an **derselben** Stelle angezeigt

– Technik erlaubt die Visualisierung **sehr großer** Datenmengen

– Visualisierung kann

- **Anfrage-unabhängig** sein  
Datenbestand muß natürliche Ordnung haben (z.B. Zeit)
- **Anfrage-abhängig** sein

### III.2.2 Pixel-orientierte Techniken

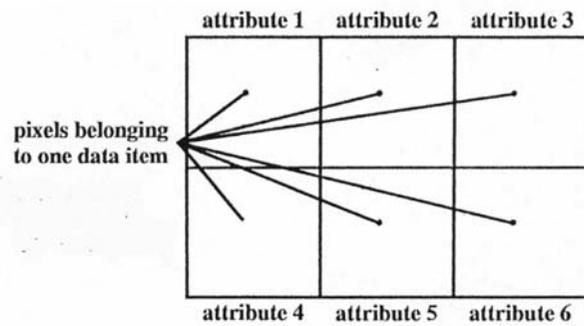
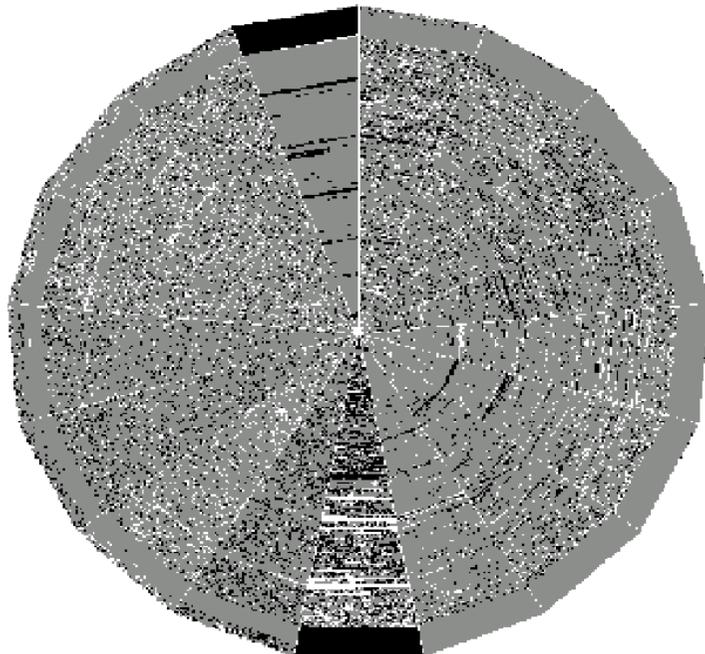
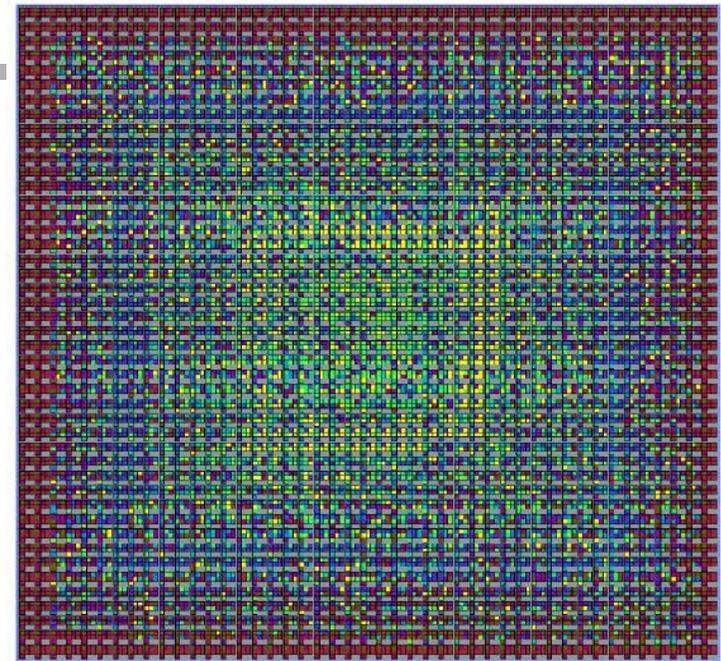


Abbildung 1: Pixel-basierte Darstellung von Datentupel (Keim/Kriegel 1996)



### III.2.2 Pixel-orientierte Techniken

#### – Anfrage-abhängige Darstellung (query-dependent visualisation technique)

- visualisiert wird **Distanz** ( $d_1, \dots, d_m$ ) zwischen Anfrage  $q$  ( $q_1, \dots, q_m$ ) und Datentupel ( $a_1, \dots, a_m$ )
- Anfrage  $q$  kann **verallgemeinert** werden zu Anfrage, die für die verschiedenen Attribute nicht einzelne Werte, sondern **Intervalle** spezifiziert

⇒ Anfrage definiert Region im  $m$ -dimensionalen Raum der Attributwerte

### III.2.2 Pixel-orientierte Techniken

- Distanztupel wird um (m+1)-ten Wert erweitert, der Gesamtdistanz des Datentupels zur Anfrage beschreibt; i.a. ist **Gesamtdistanz** gewichtete Summe der Einzeldistanzen:

$$d_{m+1} = \sum_{i=1}^m w_i d_i \quad , w_i \geq 0$$

- Distanztupel werden nach **Gesamtdistanzwert**  $d_{m+1}$  **sortiert**
- Datentupel, die die Anfrage **erfüllen** (Distanztupel haben Werte 0), werden im **Zentrum** des Windows visualisiert; die zu den anderen Datentupeln gehörigen Distanztupel spiralförmig um diesen Mittelpunkt

### III.2.2 Pixel-orientierte Techniken

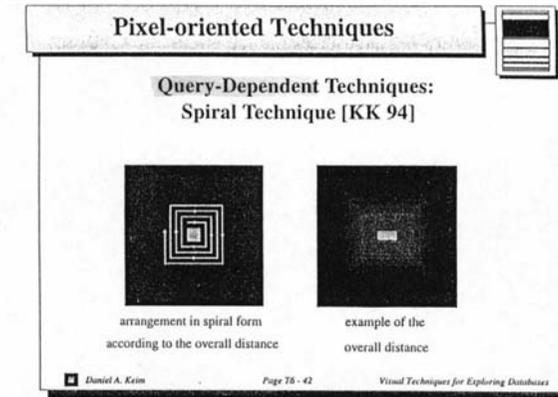


Abbildung 2: Anfrage-abhängige Visualisierung (Keim 1997)

### III.2.2 Pixel-orientierte Techniken

#### – Achsentechnik (axes techniques)

- **zwei** der  $m$  **Attribute** werden den beiden **Achsen** eines jeden Window zugeordnet
- negative bzw. positive Distanzwerte teilen Window in **4 Quadranten** auf
- Achsentechnik visualisiert die „**Richtung**“, in der die Datentupel von der Anfrage abweichen (in Bezug auf die zwei ausgewählten Attribute)

### III.2.2 Pixel-orientierte Techniken

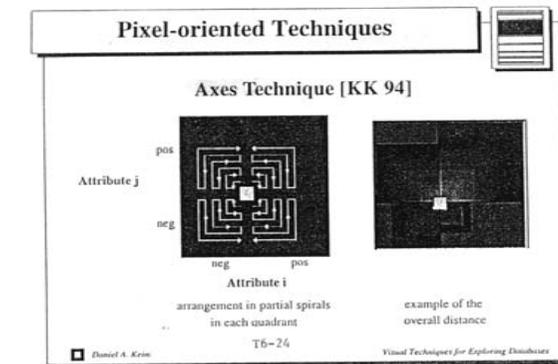


Abbildung 3: Achsentechnik (Keim 1997)

### III.2.2 Pixel-orientierte Techniken

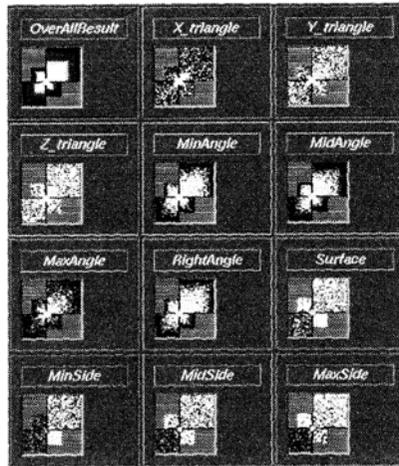
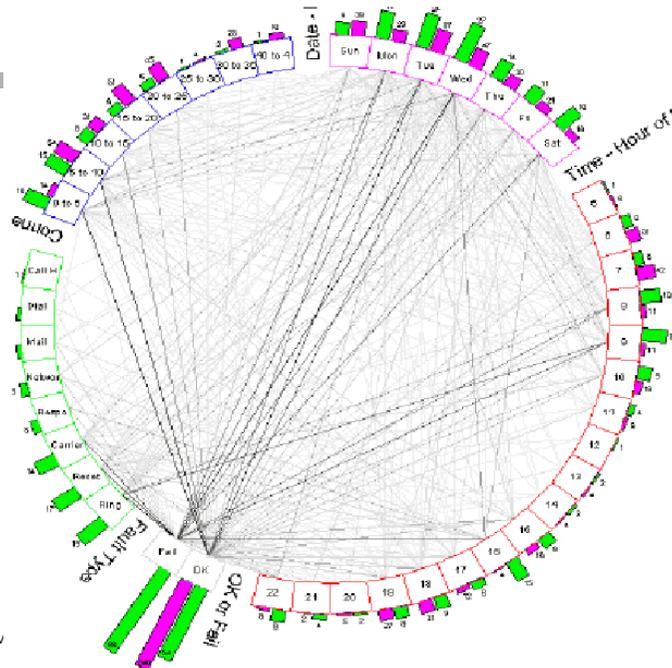


Abbildung 4: Partitioning a molecule into regions by using properties of the Triangulation with Axes Technique (Keim/Kriegel 1996)

### III.2.3 Geometrische Techniken

#### III.2.3 Geometrische Techniken

- Projektion multidimensionaler Datenbestände auf 2-dimensionale Darstellungen
- es existiert eine Vielzahl von Techniken (z.B. Hauptkomponentenanalyse, Faktoranalyse)



### III.2.3 Geometrische Techniken

#### Parallele Koordinatentechnik

(parallel coordinate visualization technique)

- Idee:
  - für n-dimensionale Datentupel werden n **äquidistante Achsen** verwendet (1 Achse pro Attribut)
  - jede Achse wird entsprechend dem Wertebereich des zugehörigen Attributs **skaliert**
  - Datentupel wird als **Polygon** visualisiert (Schnittpunkt mit Achse i repräsentiert Attributwert  $a_i$ )

### III.2.3 Geometrische Techniken

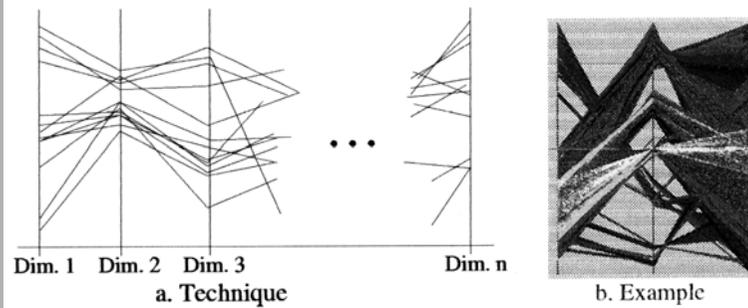


Abbildung 5: Parallele Koordinatentechnik (Keim/Kriegel 1996)

### III.2.3 Geometrische Techniken

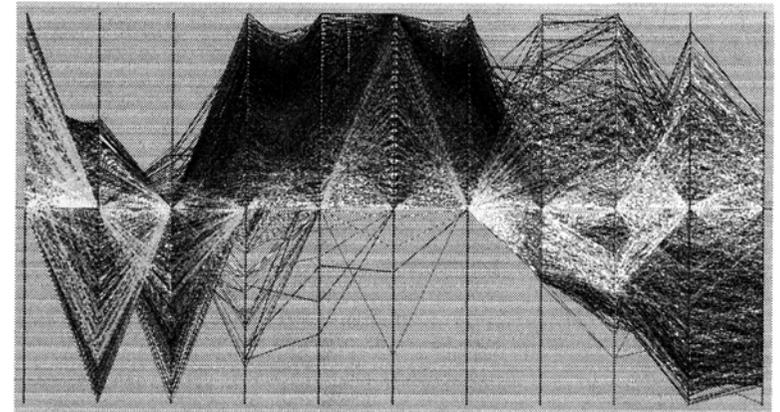


Abbildung 6: Example for the Parallel Coordinate Visualization of the Molecule Surface Data (Keim/Kriegel 1996)

### III.2.3 Geometrische Techniken

- Technik erlaubt die Visualisierung von **kleinen** Datenmengen (ca. 1000 Datenelemente)
- Technik erlaubt gute Visualisierung der „**Richtung**“, in der die Datentupel von der Anfrage abweichen - in Bezug auf **jedes** Attribut
- „Ausreißer“ (**hot spots**) sind unmittelbar sichtbar
- **funktionale Abhängigkeiten** zwischen Attributen sind gut erkennbar

### III.2.4 Icon-basierte Techniken

#### III.2.4 Icon-basierte Techniken

- multidimensionale Daten werden auf **Icons** abgebildet
- **Gestalt** des Icon repräsentiert Wert der Attribute
- **zwei** der Attribute werden den **Achsen** der Projektionsebene (d.h. des Windows) zugeordnet
- im folgenden betrachtet: **stick figure technique**
- **Idee:**
  - verbleibende (n-2) Attribute werden den **Winkeln** und den **Längen** der Figurelemente zugeordnet
  - sofern die Datenelemente hinreichend dicht beieinander liegen, entstehen **Muster**, die die Charakteristik der Daten widerspiegeln

### III.2.4 Icon-basierte Techniken

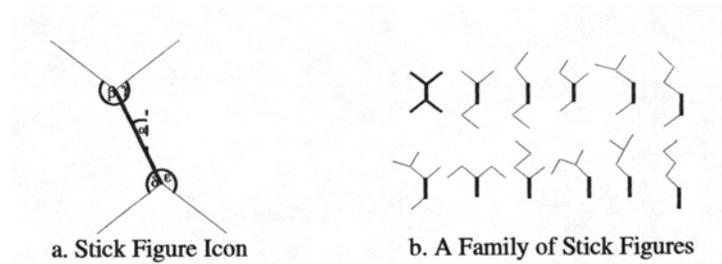


Abbildung 7: Stick Figure Visualization Technique (Keim 1997)

### III.2.4 Icon-basierte Techniken



Abbildung 8: Stick Figure Visualization of Census Data (Keim/Kriegel 1996)

### III.2.4 Icon-basierte Techniken

- **Komplexität** der Figuren **beschränkt** Anzahl der Dimensionen, die gut visualisiert werden können
- sofern die Anzahl der Datenelemente **zu groß** ist, sind Strukturen in den Daten kaum erkennbar
- Auswahl der Attribute, die den Projektionsachsen zugeordnet werden, ist sehr wichtig
  - sofern z.B. **Cluster-Attribute** als Achsen ausgewählt werden, sind Cluster durch Häufung bestimmter Elemente in gewissen Regionen erkennbar

### III.2.4 Icon-basierte Techniken

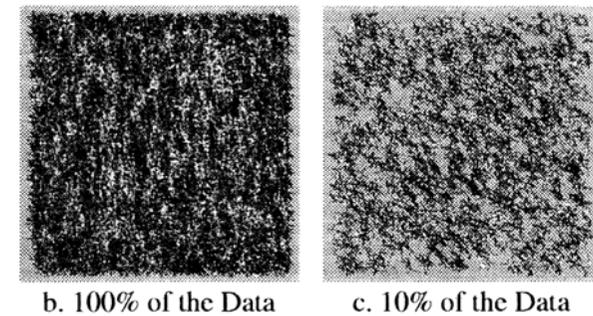


Abbildung 9: Four-dimensional Clusters in Six-dimensional Data (Keim/Kriegel 1996)

### III.3 Data Characterization Tool (DCT)

#### III.3 Data Characterization Tool (DCT)

(Engels et al. 1997; Engels/Theusinger 1998)

##### III.3.1 Introduction

- data characteristics may later on be exploited for
  - altering the **dimensionality** of the data
    - attribute generation
    - attribute filtering
    - attribute transformation
  - altering the **quantity** of the data
    - selecting learning examples
    - balancing learning examples
- DCT offers **statistical** and **information theoretical** measures

### III.3.2 Simple Characteristics

#### III.3.2 Simple Characteristics

- a collection of simple data characteristics provides **first insights** into the structure of the available data:
  - number of learning examples
  - number of classes
  - number of examples per class
  - number of attributes
  - number of numeric attributes
  - number of symbolic attributes

### III.3.3 Statistical Measures for Attributes

#### III.3.3 Statistical Measures for Attributes

- applicable for **numerical** attributes
- no NULL values
  - presume data cleaning
- collection of measures
  - (i) **location** parameters, among others
    - **arithmetic mean**
    - **$\alpha$ -trimmed mean**  
(cut away the  $2 \cdot \alpha$  percent extreme values of an attribute)
    - **median**

### III.3.3 Statistical Measures for Attributes

- location parameters provide insight into the existence of **extreme values**:
  - e.g. if arithmetic mean and  $\alpha$ -trimmed mean **differ** considerably, attribute contains extreme values
  - extreme values may be **interesting** or **disturbing** depending on the problem definition at hand
    - e.g. find unexpected values  
(e.g. fraud detection)

### III.3.3 Statistical Measures for Attributes

(ii) dispersion parameters, among others

- **standard deviation**
  - sensitive to extreme values
- **median deviation**
  - robust with respect to extreme values
- **quartiles distance**
  - range of the middle 50% of the data
  - robust with respect to extreme values

### III.3.4 Statistical Measures for Relationships between Classes / Attributes

#### III.3.4 Statistical Measures for Relationships between Classes / Attributes

- The analysis of the relationships between the classes/attributes provides further insights into the structure of the data set
- **Discriminant Analysis [Diskriminanzanalyse]** is an appropriate approach for performing such an analysis (if its assumptions are fulfilled)
- It indicates complexity of a **classification task**:  
How difficult/promising is it to discriminate between different classes?

### III.3.4 Statistical Measures for Relationships between Classes / Attributes

• covariance matrices and eigenvalues as determined in discriminant analysis provide estimations for the complexity of the classification problem, e.g.

- **Wilks Lambda:**
  - measures **class differences**:
    - value near 1: no good distinction between classes
    - value near 0: good distinction between classes
  - ⇒ **good classifier** can be learned

### III.3.5 Information Theoretical Measures

#### III.3.5 Information Theoretical Measures

- applicable for **symbolic** attributes
- measures are applicable for **single** attributes
- collection of measures are offered, among others

(i) **attribute entropy H(B):**

- given an attribute  $B$  with  $K$  different values  $b_1, \dots, b_K$   
 $p_i$ : probability of value  $b_i$

$$H(B) = - \sum_{i=1}^K p_i \log_2(p_i)$$

- interpret as number of yes/no-questions that are needed to determine a specific value  $b_i$
- $H(B)$  is maximum if all values have the same probability
- (see also description of C4.5 in Part V)

### III.3.5 Information Theoretical Measures

#### (ii) class entropy H(C):

- entropy of the **target attribute** (attribute used for the classification of the learning examples)
- indicates **complexity** of classification task: minimal number of yes/no-questions that are needed to determine the class membership of a learning example

#### (iii) joint entropy:

- entropy of a **combination** of attributes
- combines any **given** attribute with **target** attribute
  - indicates relative importance of selected attribute with respect to classification task

### III.3.5 Information Theoretical Measures

#### (iv) Equivalent Number of Attributes

- estimates the **number of attributes** that are needed to determine the value of the target attribute  $c$ , i.e. to determine the **class membership** of a learning example

$$EN.attr = \frac{H(c)}{\bar{I}_{gain}(c, A)} \quad \text{with}$$

$$\bar{I}_{gain}(c, A) = \frac{1}{S} \sum_{i=1}^S I_{gain}(c, A_i) = \frac{1}{S} \sum_{i=1}^S (H(c) - H(c|A_i))$$

- $\bar{I}_{gain}(c, A)$  indicates how much information - in the average - all attributes together provide about class membership (see also description of C4.5).
- If the number of relevant attributes that are provided by the data set is larger than the value of EN.attr, there exists a good chance to learn a good classification tree.

### III.3.5 Information Theoretical Measures

#### (v) Gini-Index

- similar to 'Information Gain', however based on a different interpretation of entropy
- small values indicate that the attributes do not contribute a lot of information about class membership

### III.3.6 Examples

#### III.3.6 Examples

##### a) Data set "Segment"

- classify pictures into 7 classes
- 19 numerical attributes
  - 3 attributes have constant values, are eliminated
- consider **statistical measures** for 2 attributes

	Mean	$\alpha$ -trimmed Mean	Std. Dev.
vegde-sd:	5,709	1,25	44,837
hegde-sd:	8,244	1,683	58,799
⋮	⋮	⋮	⋮

- measures indicate that attributes suffer from **extreme values**

### III.3.6 Examples

- **Multiple Correlation Coefficients** indicate that several attributes are (perfectly) correlated

Var:	reg.col	reg.row	vdg-mn	vgd-sd	hdg-mn	hdg-sd	intsy-mn	rwred-mn
MCC:	0,185	0,717	0,726	0,794	0,760	0,809	1,000	1,000
Var:	rwblue-mn	rwgrn-mn	xred-mn	xblue-mn	xgrn-mn	value-mn	sat.-mn	hue-mn
MCC:	1,000	1,000	1,000	1,000	1,000	1,000	0,774	0,94

Table 2. Multiple Correlation Coefficients for the sixteen variables of the segment dataset

- remove these attributes
  - considerable **reduction of dimensionality**
- very **small value of Wilks Lambda** indicates that there exists a high discrimination power within the data set
  - constructed classifier has very **high accuracy** (>90%)

### III.3.6 Examples

#### b) Data set "Post-Operative"

- classify patients in one of three classes: „Intensive Care“, „Normal Care“ or „Send them home“
- 7 symbolic attributes
  - **EN-attrib** has **high value**, since
    - class entropy  $H(C)=0.98$
    - mean information gain  $\bar{I}_{gain}(c, A) = 0.018$
 ⇒ attributes do not provide a lot of information, many of them are needed
  - **Gini-Index**  $\leq 0.021$  provides the same insight as EN.attr
- learning a classifier is **not promising**

### III.3.7 Remarks

#### Remarks

- DCT measures can be used to get various insights into the structure of the data set as well as into the possibility to learn a good classifier (DCT is oriented towards classification tasks)
  - is the available data set suitable for solving the KDD-task?
- Of course, the top-down business problem analysis **interacts** closely with the bottom-up analysis of the available data set(s)
  - **data set** and **problem** at hand have to fit together
- A complete DCT analysis involves a lot of effort
  - **balance** effort spent and insights gained

### III.4 Examples of flawed datasets

- **Classic flawed big data sets**
  - the Literary Digest Poll of 1936
  - the Lanarkshire Milk Experiment of 1930
- **Typical modern flawed big data sets**
  - voluntary surveys by magazines
  - customer data bases ignoring competitor data

Such flaws may be discovered using the techniques described before and/or by checking with common sense/background knowledge!

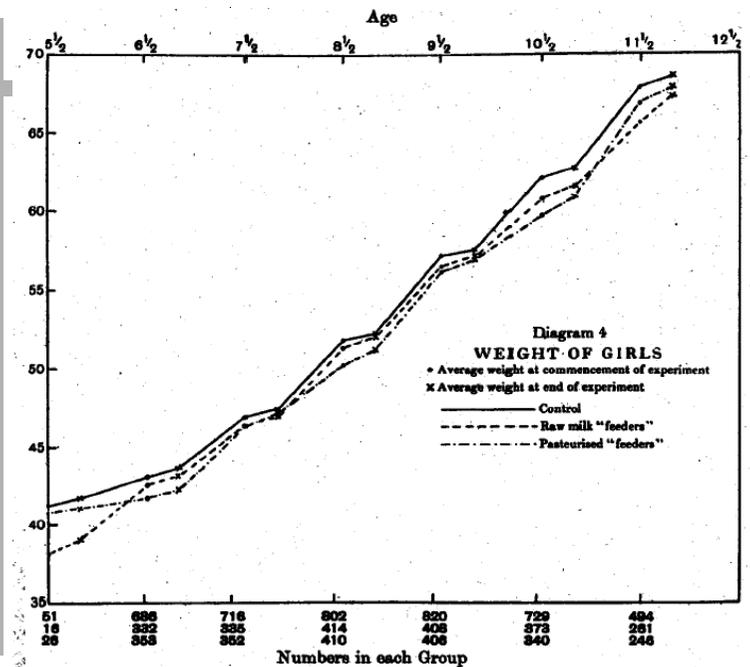
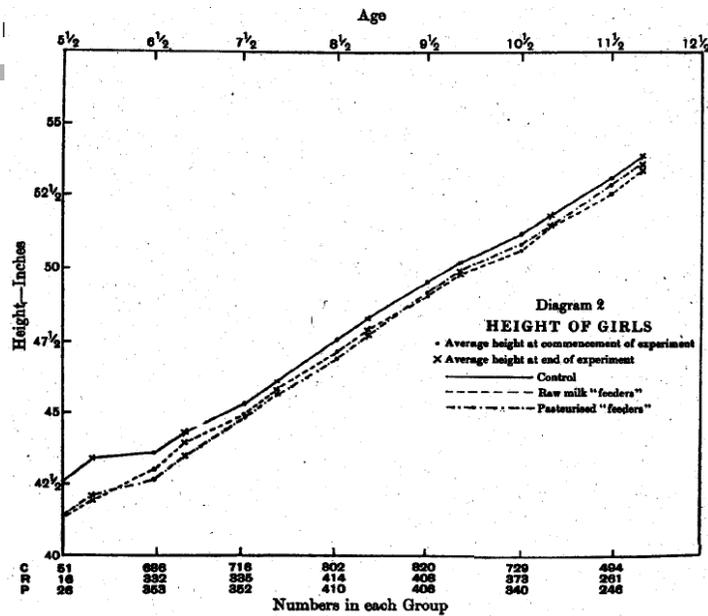
### III.4.2 Lanarkshire Milk Data

#### Poll for the US elections 1936

- 10 million car owners and telephone subscribers mailed
  - 2.376 million responded:
  - 57% for Alf Landon, 43% for Franklin D. Roosevelt
- Gallup polled 50000, and predicted FDR to win
- Result: **FDR 62%, Alf Landon 38%**
- Reasons: Biased sample, voluntary response

### III.4.2 Lanarkshire Milk Experiment of 1930

- For four months from February to June 1930, in the Scottish county of Lanarkshire 20,000 children, aged between 5 and 12 years, from 67 schools took part in an experiment:
  - 5,000 got raw milk
  - 5,000 got pasteurised milk
  - 10,000 got no milk
- Did milk help growing and, if so, which kind was better?



### III.4.2 Lanarkshire Milk Experiment of 1930

#### Problems with the experiment

- No school got both types of milk
- Allocation by ballot or alphabetically BUT then the teachers could reallocate "to obtain a more level selection"
- Weighed in February (with heavier clothes) and in June (with lighter clothes)
- Controls were analysed as one group

W.S. Gosset pointed out that a study of the identical twins amongst the group could have been much better controlled and would have given much more reliable results.

→ Small, well-planned studies are often better than large, hard to control ones.

### III.4.3 Data Quality

#### Some aspects influencing the data quality:

- Quality of variables, of definition of variables, of measurement and recording of variables
- Quality of sampling definition, of sampling procedure (choosing, locating, enrolling)
- Quality of representation
- Quality of data checks and balances
- Quality of control of potential influences

### III.4.3 Data Quality

#### Data quality criteria at the Lanarkshire Milk Experiment :

- Weight the best measure of growth? Definition with clothes? Accuracy?
- Schools sampled? (Only big schools)
- Pupils chosen by teachers.
- Allocation to groups 'corrected' by teachers
- Dropouts? Illnesses?
- Height as a check on weight
- Getting milk at home