

# Zusammenfassung IVVA

## Contents

Informationsvisualisierung .....	6
Anscombe's Quartett .....	6
Amplify Cognition .....	6
Regel Nr1.....	7
Daten-Nutzer-Aufgaben.....	7
Visualisierungsdesign und Nutzung kann auf mehreren Ebenen falsch laufen .....	7
Informationsvisualisierungsprozess .....	8
Visuelle Abbildungen .....	9
Datentypen.....	9
Nominale Datentypen.....	9
Ordinale Datentypen .....	9
Quantitative Datentypen.....	9
Datenstrukturen .....	10
Tabelle (ohne Keys).....	10
Tabelle (mit Keys) .....	11
Zeitbezogene Daten.....	11
Ortsbezogene Daten.....	11
Bewegungsdaten.....	11
Graphen (Netzwerke).....	11
Hierarchien (Bäume) .....	11
Visuelle Strukturen .....	12
Markierungen .....	12
Channels .....	12
Überblick Visualisierung .....	14
Anzahl der Channel .....	15
Captions .....	15
Wahrnehmung .....	16
Sehzentrum .....	16
Wahrnehmungsmodell von Ware .....	17
Farbwahrnehmung und -modelle.....	17
HSV-Modell.....	18
Color-Mapping .....	18
CIE-Modell.....	18

Elementare visuelle Aufgaben .....	18
Suche .....	19
Queries.....	20
Eigenschaften visueller Channels .....	20
Einflussfaktoren .....	21
Position und Layout .....	22
Komplexe Visualisierungen .....	22
Interaktion .....	22
Bedingung und Interaktion.....	23
Interaktionsmodi (Spence) .....	23
Kontinuierliche Interaktion.....	23
Schrittweise Interaktion.....	23
Passive Interaktion .....	24
Gemischte Interaktion .....	24
Interaktionsdynamik.....	24
Antwortzeiten.....	24
Interaktionstechniken .....	24
Systemnahe Interaktionstechniken .....	24
Kategorien der Interaktion .....	25
Interaktionsdesign .....	26
Leitsätze .....	26
Prinzipien .....	26
Menschliche Reaktionszeit .....	27
Techniken.....	27
Mehrdimensionale Visualisierungen .....	27
nD-quantitative Techniken .....	27
nD-nominale Techniken .....	29
nD Techniken .....	30
Übersicht .....	30
Darstellung von Big Data – Ordnen-Methoden .....	30
Dimensionsreduktion .....	30
Feature Selektion .....	31
Dimensionsreduktionsverfahren .....	31
Übersicht .....	32
Visualisierung Zeitbasierter Daten .....	32
Periodische Zeitreihen .....	34

Diskrete Wertearchse .....	34
Übersicht .....	35
Graphen .....	35
Bäume .....	36
Graphendarstellung .....	37
Geobasierte Visualisierung .....	39
Karten .....	39
Visualisierung geobezogener Daten .....	39
Kartenprojektion.....	40
Verzerrte Darstellung geobasierter Daten .....	40
Visualisierung ortsbezogener Daten .....	40
Visualisierung nicht geobezogener Daten.....	41
Kartografische Abbildungen auf Marks und Channels .....	41
Glyphen .....	41
Spatio-Temporal Data.....	41
Datenvorverarbeitung.....	42
Methoden.....	42
Visual Analytics.....	43
Knowledge Discovery (KDD) .....	43
Vergleich zum Visualisierungsprozess .....	44
Visual Analytics Process Modell .....	45
Analytisches Schließen .....	45
Muster.....	46
Stärken und Schwächen von Mensch und Maschine .....	46
Analyse für die Visualisierung .....	47
Visualisierung mit automatischen Verfahren .....	47
Automatische Verfahren .....	48
Verbesserung der Modellierung durch Visualisierung .....	49
Von Daten zu Mustern.....	49
Clustering.....	50
Distanzmaße für Clusteringalgorithmen .....	50
Ähnlichkeit.....	50
Mustererkennung.....	50
Ähnlichkeit .....	50
Black-Box Integration .....	51
Visualisierung in Clustering und Dimensionsreduktion .....	51

Semi-Supervised Clustering .....	51
Visual Input Editing.....	52
White-Box Integration .....	52
Model-Data-Linking .....	52
Hierarchisches Clustering .....	53
DBScan.....	53
Dimensionsreduktion .....	53
Übersicht .....	54
Von Mustern zu Modellen.....	54
Data-Mining Aufgaben.....	54
Entscheidungsbäume .....	55
Qualitätsmaße .....	56
Integrationsvarianten .....	56
Overfitting.....	56
Visual Input Editing.....	57
Model Data Interaktion .....	57
Visual Model Verification .....	57
Klassifikation ohne Entscheidungsbäume.....	58
Support Vector Machines .....	58
K-Nearest-Neighbors .....	58
Vergleich.....	59
Bayes Klassifikation.....	59
Künstliche Neuronale Netze .....	59
Übersicht – Integrationsvarianten.....	60
Querverbindung.....	60
Direkte/ Indirekte Kopplung .....	61
Kognitive Psychologie .....	61
Aufmerksamkeit .....	62
Gedächtnis .....	63
Wissensrepräsentierung .....	63
Denken, Entscheiden, Urteilen .....	63
Abduktives Schließen .....	64
Entscheidungslehre.....	64
Problemlösen.....	65
Expertenfähigkeiten.....	65
Mentale Modelle .....	65

Interaktion .....	66
Evaluation .....	66
Testmethoden .....	67
Nested Model.....	68

## Informationsvisualisierung

Viele Aufgaben beinhalten Entscheidungen, viele Entscheidungen beruhen (im besten Fall) auf Daten und Informationen. Dies gilt im Besonderen für „objektive“ Entscheidungen, deren Auswirkung auch andere treffen.

**Definition:** „The use of computer-supported, interactive, visual representations of abstract data to amplify cognition“ (Card, Mackinlay, Shneiderman, 1999)

Von Daten zu Wissen



### Anscombe's Quartett

Abstrakte Daten wie Zahlen sind meist sprachliche Artefakte. Selbst wenn sie gelesen werden, werden sie danach im Sprachzentrum des Gehirns bewusst verarbeitet.

Informationen werden dabei nacheinander erfasst, und häufig in der Zeit strukturiert (d.h. zu einer „Story“).

Eine Visualisierung derselben Daten ist ein bildliches Artefakt. Dieses erlaubt uns, neue Information im Sehzentrum des Gehirns unbewusst zu verarbeiten.

Sehen strukturiert Informationen im Raum. Dies geschieht für das ganze Bild gleichzeitig. Erst beim Lesen der Informationen und der Strukturen (Punkte, Muster) müssen Bild und Sprache wieder verknüpft werden.

### Amplify Cognition

Information Erfassen		
Explain	Explore	Enjoy
Bekannte Informationen an andere vermitteln, unterstützt durch Visualisierung der Fakten/Daten.	Neue Informationen auf der Grundlage von Daten finden oder unsicheres Wissen bestätigen oder falsifizieren.	Zwanglose bzw. durch Neugier angetriebene Begegnung mit den Daten.
Nicht immer interaktiv	Sehr interaktiv, nicht immer ist ein Ziel bekannt	Selten ist eine Aufgabe bekannt
Information Produzieren		
Annotate	Record	Derive

## Regel Nr1

Visualisierungen sind Werkzeuge



Es gibt unzählige Designs, nach denen eine Visualisierung entworfen werden kann. Manche Designs sind gefährlich, die meisten helfen nicht wirklich, nur wenige sind gut. Man wähle ein gutes Design.

Die Wahl des Werkzeugs hängt von den Daten, der Aufgabe und den Nutzern ab. Visualisierungstechniken häufig nur für bestimmte Datentypen und Datenstrukturen geeignet. Visualisierungstechniken erleichtern nur bestimmte Aufgaben mit den Daten. Jedes Design basiert auf einer Priorisierung, Wichtigkeit der Daten, Aufgaben und Nutzern abwiegen.

- Have something to tell or to ask

### Daten-Nutzer-Aufgaben

Die Aufgaben unterscheiden sich in wesentlichen Details.

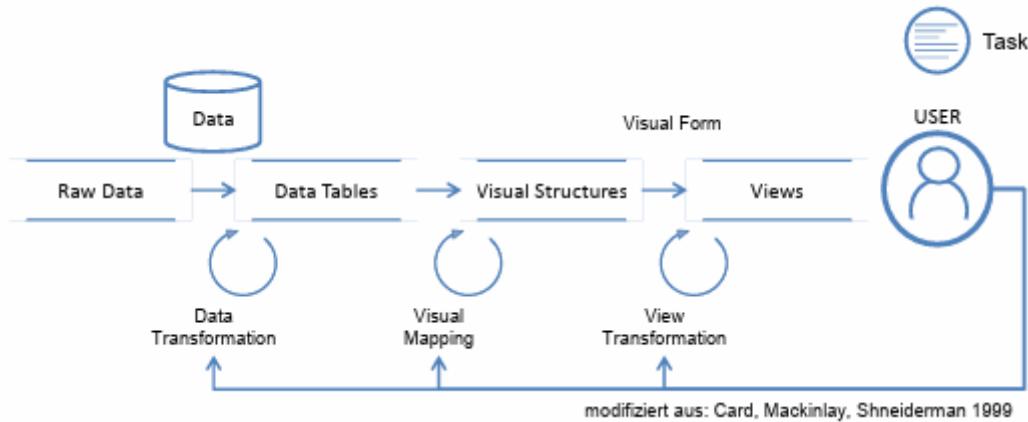
- Welche Information ist als bekannt vorausgesetzt?
- Welche Informationen werden gesucht?
- Was wird mit dieser neuen Information gemacht?

Das Design ihrer Visualisierung bestimmt mit, wie gut **bekannte Informationen zu finden** sind, und wie gut die **neuen Informationen zu lesen** sind. Im Kern besteht Ihre Aufgabe als Designer immer darin, diese beiden Aufgaben so einfach wie möglich zu machen.

### Visualisierungsdesign und Nutzung kann auf mehreren Ebenen falsch laufen

1. Design beeinträchtigt die Möglichkeiten der „Low-Level“ Wahrnehmung
2. Design beeinträchtigt die Mustererkennung
3. Design erschwert die Lesbarkeit der Charts („Visual Literacy“)
4. Schlussfolgerungen basieren auf falschen Annahmen

## Informationsvisualisierungsprozess



Das **Datenflussmodell** definiert Visualisierung als Transformationskette von Daten zum Bild. Es ist gleichzeitig ein Strukturmodell für den Aufbau von Visualisierungen und definiert, wie die Reaktion auf Änderungen (Daten oder Interaktion) organisiert werden kann.

**Datentransformation:** Rohdaten werden in Datentransformation zu **Datentabellen** umgewandelt. Dies ist notwendig, weil häufig das Format der Rohdaten, das Schema, die Struktur, oder die Werte selbst nicht zur Visualisierung passen. Und weil man Visualisierungen gerne für verschiedene Datenquellen wiederverwendet, passt man die Daten anstatt der Visualisierungen selbst an.

**Visuelle Abbildung:** Die Visuelle Abbildung ist die Komponente, die eine Visualisierung überhaupt erst zur Informationsvisualisierung macht. Sie beschreibt, welche Datenvariablen auf welche visuellen Strukturen abgebildet werden. Welche visuellen Strukturen zur Verfügung stehen, hängt teilweise vom Typ der Visualisierung ab.

**View Transformation:** Die View Transformation beschreibt welche Datenwerte der Datenvariable auf welche visuellen Variablenwerte der visuellen Struktur abgebildet werden sollen. Zu jeder Visuellen Struktur gibt es daher eine View Transformation. Erst die View Transformation macht die Visualisierung als View sichtbar.

**Achtung:** Die Visuelle Abbildung beschreibt, **dass** z.B. ein Alter auf eine Position abgebildet wird. Die View Transformation beschreibt, **welches** Alter auf **welche** Position abgebildet wird. Visuelle Strukturen sind daher (ironischerweise) eigentlich nicht sichtbar.

**Interaktion:** Interaktion ist technisch die Möglichkeit, die Transformationen und das Visual Mapping zu steuern. Der **Nutzer** soll die Sicht auf die Daten an neue Fragen/Aufgaben anpassen können, die evtl. erst nach Betrachten der ersten Visualisierung auftreten – ohne dass er warten muss, bis eine neue Visualisierung dafür gebaut ist. Der **Designer** löst mit Interaktion das Problem, dass er vielleicht nicht genau weiß, wer der Nutzer ist und was die Aufgabe ist. Der Visualisierungsprozess erlaubt dann Freiheitsgrade, die erst der Nutzer bestimmen und verändern kann. Aber: **Interaktion verschiebt Kosten** vom Designer (der sich eine genaue Anforderungsanalyse „spart“) zum Nutzer (der die Interaktion lernen muss).

# Visuelle Abbildungen

Die einfache Regel, dass eine Datenvariable einer visuellen Struktur entspricht, gilt häufig aber nicht immer. Die Position, auf die ein Datenwert abgebildet wird, ist manchmal nur von diesem einzelnen Wert abhängig oder von [allen] anderen Datenwerten abhängig.

## Datentypen

Datentypen unterschieden sich vor allem durch Operatoren, die auf Ihnen sinnvoll angewendet werden können.

Entscheidend ist die Bedeutung des Typs, nicht dessen Repräsentation oder Format. Bestimmte Eigenschaften visueller Strukturen werden bevorzugt erfasst.

Einfache Datentypen beziehen sich immer auf eine Datenvariable, Datenvariablen sind meist Eigenschaften von Objekten (hier genannt: Item). Datenstrukturen beschreiben Beziehungen zwischen Datenvariablen und Items. Eine Visualisierung kann Kombinationen von Datenvariablen, Items und Beziehungen abbilden.

## Nomiale Datentypen

Nomiale Datentypen sind eine Menge von Werten X, auf denen nur die beiden Operatoren

- Gleichheit (==)
- Ungleichheit (!=)

definiert sind.

Nomiale Daten sind kategorische Daten ohne natürliche Ordnung oder Rangfolge zwischen den Kategorien. Diese Art von Daten wird verwendet, um Dinge zu klassifizieren oder zu benennen.

Beispiele: Obstsorten, Tierarten, Branchen, Namen

## Ordinale Datentypen

Ordinalen Datentypen sind eine Menge von Werten X, auf denen die Operatoren

- Gleichheit (==)
- Ungleichheit (!=)
- Ordnung (<,>)

definiert sind.

Ordinalen Daten sind kategorische Daten, die eine natürliche Ordnung oder Rangfolge haben, aber der Abstand zwischen den Rängen ist nicht notwendigerweise gleich oder bekannt.

Beispiele: Schulnoten, Firmengröße, Ratings, Seriennummern

## Quantitative Datentypen

Quantitative Datentypen sind eine Menge von Werten X, auf denen die Operatoren

- Gleichheit (==)
- Ungleichheit (!=)
- Ordnung (<,>)

- Differenzen (-)
- Verhältnisse zwischen Differenzen (-/-)

definiert sind.

Quantitative Daten sind numerische Daten, die messbare Mengen darstellen und mathematische Operationen zulassen.

Beispiele: Temperaturwerte, Wochentage (diskret, zyklisch)

**Diskrete Quantitative Datentypen:** endliche, abzählbare Wertemenge

**Kontinuierliche Quantitative Datentypen:** unendliche, überabzählbare Wertemenge

**Intervallskala** Quantitative Datentypen: Verhältnisse (und Produkte) sind nicht sinnvoll zu berechnen, Skala hat keinen natürlichen Nullpunkt (Beispiel: Temperatur in Celsius).

**Verhältnisskala** Quantitative Datentypen: Verhältnisse sind sinnvoll zu berechnen, Skala hat natürlichen Nullpunkt (Beispiel: Temperatur in Kelvin)

**Lineare Skala** Quantitative Datentypen: Ordnungsrelation <> ist eine „Totalordnung“

**Zyklische Skala** Quantitative Datentypen: Ordnung/Differenzen müssen „künstlich“ definiert werden

## Datenstrukturen

Datentabellen sind eine sehr allgemeine Repräsentierung für sehr unterschiedliche Datenstrukturen. Visuelle Abbildung bildet mehrere Attribute auf mehrere visuellen Strukturen ab. Um die visuelle Abbildung geeignet wählen zu können, sollten Rolle und Datentyp der Spalten bekannt sein.

Spaltenattribute einer Tabelle können Keys oder Values sein. Die Datentypen der Keys (IDs, Namen, Zeitstempel) definieren die eigentliche Datenstruktur

- Keys: unabhängige Attribute der Datenstruktur
- Values: [von den Keys] abhängige Attribute der Datenstruktur
- Nur je eine Zeile darf die gleiche Kombination von Keys enthalten, da die Kombination von Keys identifiziert ein Item. Eine Tabelle definiert dann eine oder mehrere Abbildungen von Item auf Values.

Die im folgenden vorgestellten Datenstrukturen lassen sich mit einer Tabelle darstellen. Komplexere Datenstrukturen lassen sich durch mehr als eine Tabelle repräsentieren. (relationales Datenmodell einer Datenbank), dabei werden Tabellen verlinkt (Kantenliste + Knotenliste).

### Tabelle (ohne Keys)

Die Tabelle enthält keine Attribute als Keys, Items sind nur technisch Unterscheidbar (durch Zeilennummern). Typisch bei Daten, wo die Eigenschaften wichtiger sind als die Identifikation der Items und Notwendig, wo Items nicht identifiziert werden dürfen (Anonymisierung).

**Univariate** Daten beziehen sich auf Datensätze, die nur eine einzige Variable enthalten. Die Analyse konzentriert sich auf die



Beschreibung und das Verständnis der Eigenschaften dieser einen Variable.

**Bivariate** Daten beziehen sich auf Datensätze, die zwei Variablen enthalten. Die Analyse konzentriert sich auf die Beziehung oder den Zusammenhang zwischen diesen beiden Variablen.

**Multivariate** Daten umfassen Datensätze mit mehr als zwei Variablen. Die Analyse dieser Daten kann komplex sein und zielt darauf ab, Muster und Beziehungen zwischen mehreren Variablen gleichzeitig zu verstehen.

## Tabelle (mit Keys)

Die Tabelle enthält mindestens ein Attribut als Key. Dieser Key ist nominal.

## Zeitbezogene Daten

Mindestens ein Key hat einen Zeitstempel. Der Zeitstempel kann eine absolute oder relative Zeit sein.

## Ortsbezogene Daten

Mindestens ein Key definiert einen Ort.

Allgemein: Keys n-dimensionaler Datenvektor, der beliebige Art von Position/ Ort beschreiben kann.

Variante: „geographische Position“: quantitativ, mindestens zwei Keys definieren geographische Länge und Breite.

Variante: „Ortsname“: nominal, muss erst in geographische Position übersetzt werden, wenn eine Karte dargestellt werden soll.

## Bewegungsdaten

Kombination aus zeitbezogenen Daten und ortsbezogenen Daten.

Variante: Messungen an mehreren, aber festen Orten.

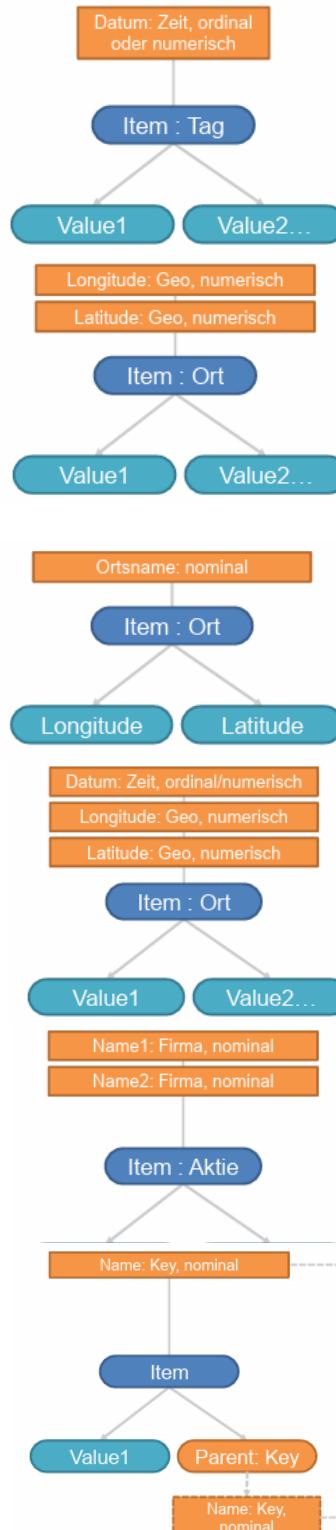
Variante: Ort der Messungen bewegt sich.

## Graphen (Netzwerke)

Mindestens zwei Keys sind vom gleichen kategorischen Datentyp. Die Keys bezeichnen eine Kante. In diesem Fall ist die Tabelle eigentlich die Kantenliste eines Graphens. Values sind dann Eigenschaften der Kante, nicht der Knoten.

## Hierarchien (Bäume)

Ein Valueattribut enthält einen (anderen) Key der gleichen Tabelle. Value definiert „ist Kind von“-Relation zwischen Items der Tabelle (Tabelle ist Knotenliste). Alternative Definition als „ist Elter von“-Relation (Tabelle ist Kantenliste).



Bäume sind mathematisch auch Graphen aber Visualisierungstechniken für Hierarchien sind im Allgemeinen keine Graphvisualisierungen. Hierarchievisualisierung nutzen die Richtung der Hierarchie.

## Visuelle Strukturen

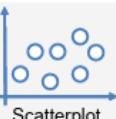
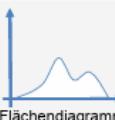
Jede neue Visualisierung beginnt als leeres Blatt. Der **Raum** ist bei weitem die wichtigste und vielseitigste visuelle Struktur. Position ohne weitere Angaben bezieht sich immer auf zwei unabhängige Variablen.

## Markierungen

Marks sind die geometrischen Elemente, aus denen die Visualisierung zusammengesetzt wird. Meistens repräsentieren sie jeweils ein Item oder eine Relation zwischen mehreren Items.

Markierungen können Punkte, Linien, Flächen oder Text sein.

In der Klausur KEIN Text als Mark!!! Ausnahme Beschriftung Legende, Achsen, Caption, Titel.

Beispiel	Abbildung	Beispiel	Abbildung	Beispiel	Abbildung
	Scatterplot 1 Item → 1 Punkt Metapher: Ort im Raum, Distanz		Matrix 2 Items → 1 Punkt Metapher: Abstraktion?		Attribut → Punkt Metapher: Abstraktion?  Entscheidungsbaum
	Parallel-Koordinaten 1 Item → 1 Linie Metapher: Verbindung		Node-Link Diagramm 2 Items → 1 Linie Metapher: Verbindung		Attribut → Linie Metapher: Kontinuierliche Änderung, Bewegung  Liniendiagramm
	Balkendiagramm 1 Item → 1 Fläche Metapher: Kategorien, Anhäufung		Venn Diagramm N Items → 1 Fläche Metapher: Zusammen gehören, enthalten sein, Kategorien, Menge		Attribut → Fläche Metapher: Kontinuierliche Änderung, Anhäufung  Flächendiagramm

Für die Auswahl der passenden Markierung gibt es die folgenden Kriterien:

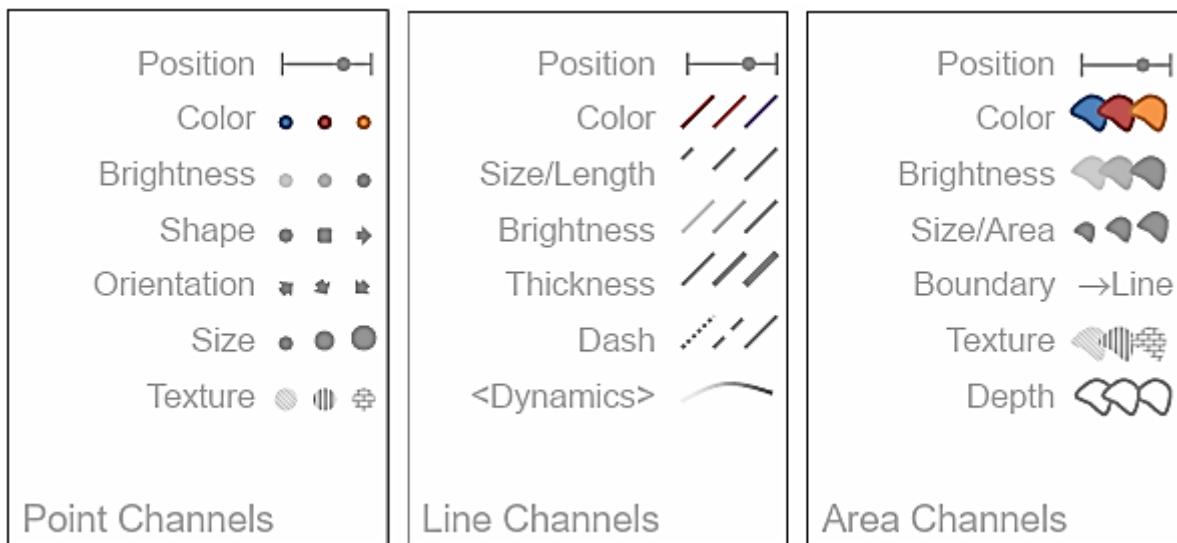
Was soll visuell repräsentiert werden?

- Items
- Paare
- Teilmengen
- Attribut

Welche Channels werden gebraucht?

## Channels

Die Flächengröße ist irrelevant. Die Markierungen sind „0D“ (Punkt), „1D“ (Linie), oder „2D“ (Fläche). Channels sind die visuellen Eigenschaften einer Markierung. Welche Channels eingesetzt werden können, hängt von einer Markierung ab.



Die visuelle Abbildung erhält im Normalfall die Beziehung zwischen Items und Merkmalen.



### Position

Die Position ist die einzige visuelle Struktur, welche notwendig genutzt werden muss, die für alle Datentypen eingesetzt werden kann und die alle visuellen Aufgaben (Suchen, Vergleichen, Ordnen) potenziell gut unterstützt. Die Position ist die visuelle Struktur mit der höchsten Anzahl unterscheidbarer Werte. Die Wahl der Datenvariablen für die Position ist die erste und wichtigste Entscheidung beim Design.

### Farbkanäle

Farbe ist vielseitig einsetzbar, Farbton, Helligkeit und Sättigung sind drei verschiedene Channel. Helligkeit und Sättigung wird praktische niemals gleichzeitig für zwei verschiedene Datenvariablen genutzt. Neben der Position sind die Farbkanäle die einzige visuelle Struktur, welche mit den kleinsten Markierungen (=Pixel) funktionieren.

Allerdings hängt die Nutzbarkeit von der richtigen Farbskala ab.

### Länge

Neben der Position ist die Länge das einzige visuelle Attribut, welches numerische Größenverhältnisse getreu darstellen kann. (Parallele) Länge ist eventuell sogar besser für Größenverhältnisse geeignet als die Position.

Die Wahrnehmung von Längen ist besonders leicht beeinflussbar (numerischer Vergleich erfordert einfache Visualisierungen). Falls die Visualisierung also einen numerischen Vergleich unterstützen soll, muss man Länge und Position nutzen und alle anderen Channel und Markierungen weglassen.

## Größe

Größe und Fläche ist naheliegend für den qualitativen Vergleich von Größenordnungen. Wie gut Größen vergleichbar sind, hängt von den Größenverhältnissen ab. Größendifferenzen sind nicht gut vergleichbar.

Aber Größe kostet Platz.

## Form

Der Channel Form kann gut gelernt werden. Die Formen erhalten dabei eine Bedeutung und bekannte Formen können leicht wiedererkannt und besser gesehen werden. Potenziell könnte eine Visualisierung viele verschiedene Formen nutzen.

Der Nachteil von Formen ist, dass gelernte Formen Konventionen sind und nicht einfach umdefiniert werden können. Im Normalfall werden wenige neutrale Formen für nominale Datentypen in Visualisierungen verwendet.

## Orientierung

Die Orientierung kann nur mit bestimmten Merkmalen und Formen verwendet werden und kann ordinale und numerische Daten repräsentieren.

Ordinal sind abstrakte Richtungen und Numerische sind konkrete Richtungen.

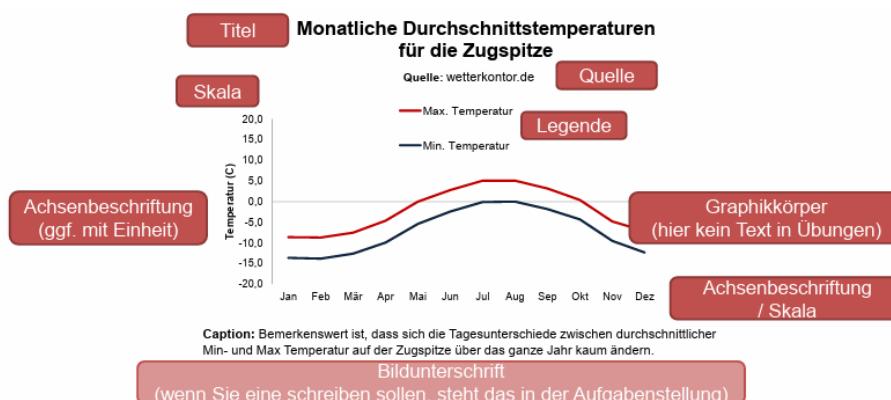
## Exoten

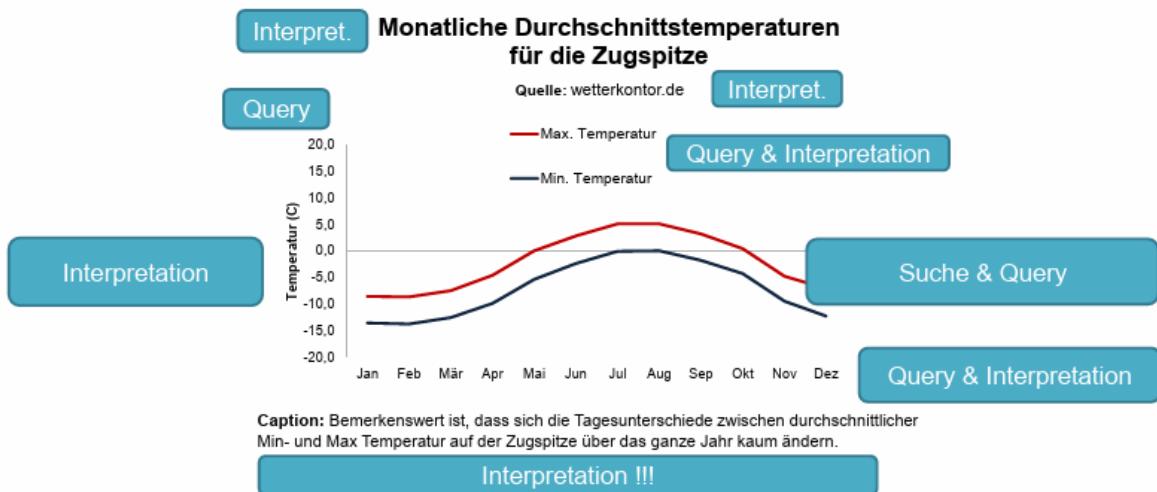
Tiefe/ Überdeckung, Schatten, Animation/ Bewegung, Schattierung, Textur und Unschärfe sollte vorsichtig verwendet werden, da diese zum Teil schwer visuell differenzierbar sind.

**In der Klausur KEINE Regenbogen-Farbskala verwenden!!!**

## Überblick Visualisierung

Überblick über notwendige Visualisierungsbestandteile.





## Anzahl der Channel

Explain	Explore	Enjoy
Generell wenige Channels	Eher wenige Channels	Nicht definiert
Sie wollen, dass ihre Visualisierung verständlich bleibt. Sie wollen auch, dass Sie noch verstehen, was eigentlich von anderen wahrgenommen wird.	Die Channels sind nicht gleichrangig. Wenn sie aber mit offenem Ergebnis explorieren wollen, sollten Sie vermeiden, dass dominante Channels die Wahrnehmung von Mustern verhindern.	Probieren Sie aus, was interessant ist und Ihnen Freude bereitet

	Nominal	Ordinal	Quantitativ	Spatial	Temporal
<b>Position</b>	+	+	+	+	+
<b>Länge</b>	-	+	+	?	?
<b>Größe</b>	-	+	o	-	-
<b>Farbsättigung</b>	-	+	o	-	-
<b>Textur</b>	+	+	-	-	-
<b>Farbton</b>	+	(-)	-	-	-
<b>Orientierung</b>	+	+	-	-	-
<b>Form</b>	+	-	-	-	-

## Captions

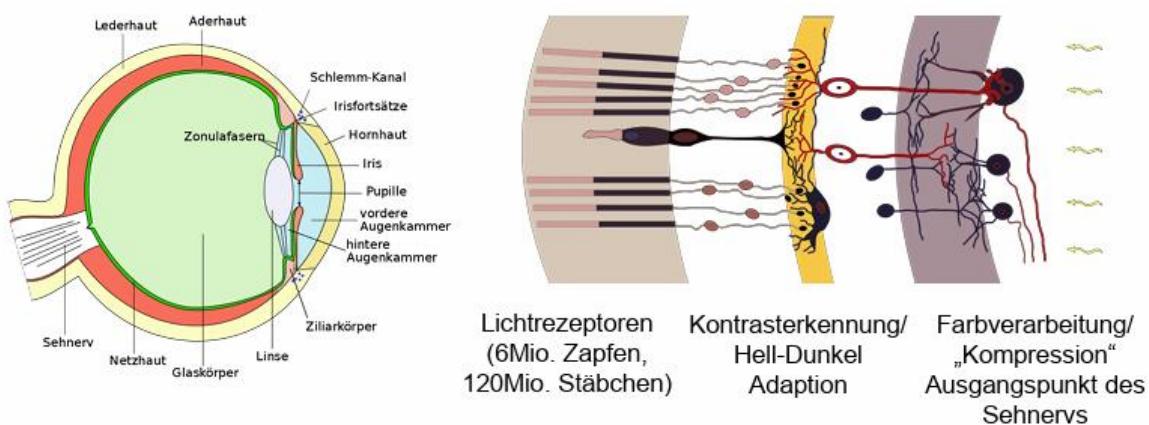
**Bildunterschrift:** Liefert eine kurze Interpretation/ Einordnung/ Fragestellung des Bildes und kann auf bestimmte Teile der Abbildung verweisen. Bild und Bildunterschrift sollten auch idealerweise ohne den Haupttext verständlich sein.

**Fließtext:** Verweist auf die Abbildung, sollte aber keine Bildbeschreibung der Abbildung sein, da Fließtext nicht immer neben dem Bild aufgefunden werden kann. Die Zentrale Aussage der Abbildung (Interpretation/ Einordnung) sollte man im Fließtext wiederfinden, die Abbildung sollte im Haupttext motiviert sein.

**Titel:** Der Titel ist keine Bildunterschrift, der Titel ist ein Name und dient primär zur Unterscheidung. Der Titel muss (im Gegensatz zur Bildunterschrift) nichts erklären, darf aber Interesse wecken. Die einfachsten Titel sind entsprechend Abbildung 1, 2, 3...

## Wahrnehmung

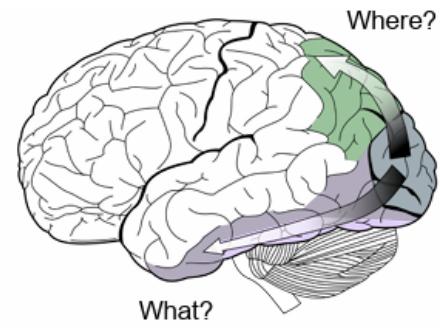
Visuelle Wahrnehmung beginnt im Auge (Netzhaut) und endet in den Hirnarealen für räumliche Orientierung und Handlungen und für die Erkennung von Objekten. Mehrstufiger Prozess bei den elementaren Informationen in komplexere transformiert werden. Dieser Prozess wird durch höhere kognitive Prozesse beeinflusst (Lernen, Erinnerung) und ist teilweise bewusst steuerbar (Aufmerksamkeit).



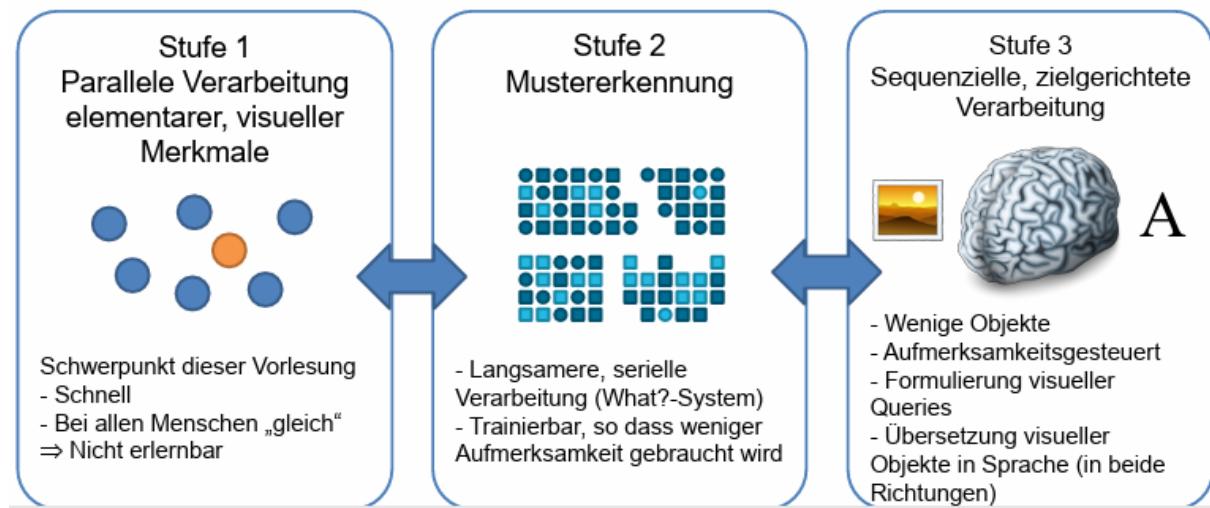
## Sehzentrum

Das Sehzentrum ist in mehrere Schichten (V1-V5) unterteilt, die unterschiedlichen Aufgaben erfüllen. Allgemein werden die verarbeiteten Informationen von V1 bis V5 stetig „komplexer/abstrakter“ und die Schichten sind in beide Richtungen verbunden. Die Funktion und Interaktion aller Schichten werden weiterhin erforscht.

	„What“-System	„Where“-System
Funktion	Erkennung	Lokalisierung
Erregung durch	Details	Bewegung
Speicherung	Längerfristig	Kurzfristig
Geschwindigkeit	Langsam	Schnell
Aufmerksamkeit	Bewußt	Vorbewußt



## Wahrnehmungsmodell von Ware



Wahrnehmung [des gleichen Bildes] liefert nicht zu jeder Zeit und/oder bei jedem Menschen das gleiche Ergebnis. Wahrnehmung wird durch Aufmerksamkeit, Erinnerung und andere höhere kognitive Prozesse gesteuert. Unter bestimmten Bedingungen kann die Wahrnehmung und Interpretation auch komplexester Muster durch Üben gelernt werden.

Ein absolut „richtiges“ oder „falsches“ Design kann es nur bezüglich gut verstandener, und bei jedem Menschen weitgehend gleich ablaufender Wahrnehmungsprozesse und kognitiver Prozesse geben. Ihre Aufgabe ist es nicht, ein „perfektes Design“ zu schaffen, sondern Designabwägungen zu machen, diese anwenden und begründen zu können.

## Farbwahrnehmung und -modelle

„Farbe“ kann durch drei (fast) unabhängige Größen definiert werden. Verschiedene Modelle definieren diese Größen unterschiedlich. Das RGB-Modell ist z.B. technisch relevant, und hat einen engen Bezug auch zu den Farbrezeptoren der Netzhaut.

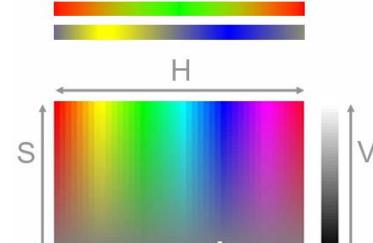
Die drei Farbrezeptoren sind genaugenommen:

- S-Zapfen (für „Short“, gemeint ist die Wellenlänge): maximale Empfindlichkeit bei 419 nm (blau)
- M-Zapfen („Medium“): maximale Empfindlichkeit bei 531 nm (grün)
- L-Zapfen („Long“): maximale Empfindlichkeit bei 558 nm (grün-gelb(!))
- Relatives Mengenverhältnis variiert zwischen verschiedenen Menschen (L:M:S ca. 30:60:10)

Noch in der Netzhaut werden die Informationen aus Rezeptoren S, M und L umgewandelt.



Andere „Bunte“ Farben entstehen als Kombination des Rot-Grün/Blau-Gelb Systems, dies erklärt, dass Helligkeit als unabhängige Farbqualität wahrgenommen wird aber nicht, warum die beiden bunten Kanäle nicht unabhängig wahrgenommen werden. Unbunte Farben liegen im neutralen Bereich der Blau-Gelb Skala.



## HSV-Modell

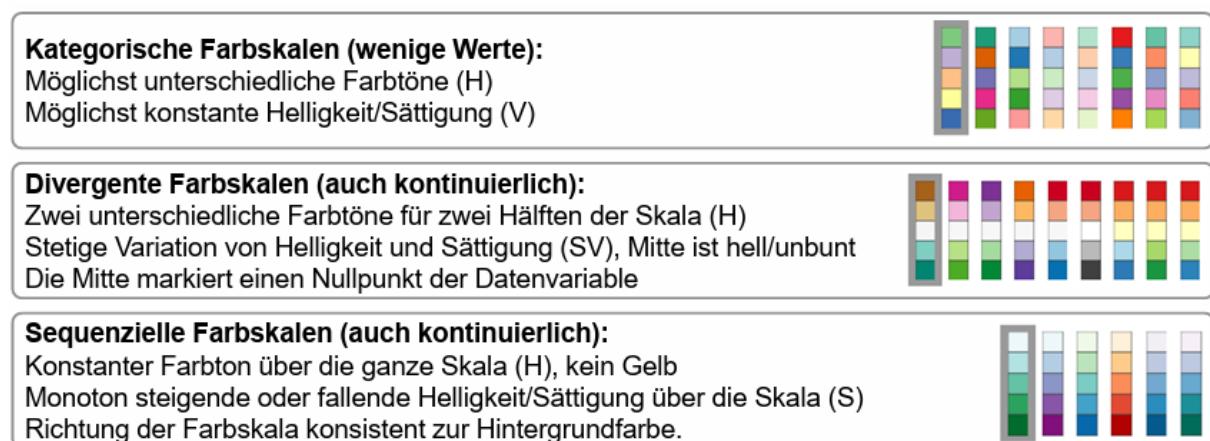
In der Wahrnehmung und Visualisierung relevant ist das HSV-Modell.

- H(ue), Farbton
- V(alue), Helligkeit
- S(aturation), Sättigung/“Buntheit“

Innerhalb einer Visualisierung wird häufig nur eine Datenvariable, maximal zwei Datenvariablen auf Farbe abgebildet.

## Color-Mapping

Color-Mapping ist die Zuweisung einer Skala von Datenwerten auf eine Skala von Farben, aber durch die drei Freiheitsgrade der Farbe und die Anzahl der Farben in der Farbskala gibt es viel mehr Möglichkeiten als bei den meisten anderen Channels. Die geeigneten Farbskalen beschränken die Freiheiten auf sinnvolle Weise.



## CIE-Modell

Das CIE-Modell ist eine Normfarbtafel (umgangssprachlich Schuhsohle) und erfasst die Gesamtheit wahrnehmbarer und darstellbarer Farben. Es stellt den Bezug zwischen physikalischen Eigenschaften und Farbwahrnehmung dar und modelliert additive Farbmischung.

Das CIE-Modell ist auch technisch relevant für den Vergleich des Farbraums eines Ausgabegeräts. Der Farbraum ist immer kleiner als der wahrgenommene Farbraum und bildet ein Dreieck zwischen den drei Grundfarben des Geräts.

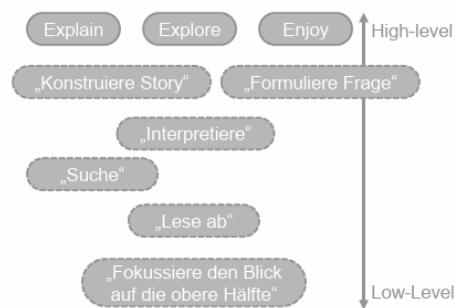
## Elementare visuelle Aufgaben

Aufgaben haben verschiedene Abstraktionsstufen.

- High-Level Aufgaben repräsentieren Ziele
- Low-Level Aufgaben beziehen sich auf elementare Handlungen

Eine Visualisierung ist gegeben falls nur ein Werkzeug für eine Teilaufgabe

Visualisierungsaufgaben sind nicht eindeutig definiert.

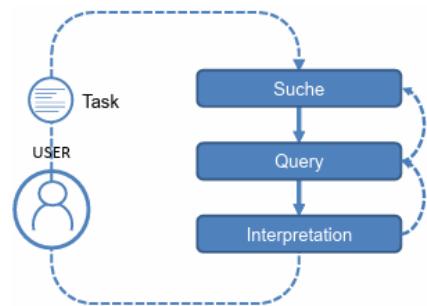


**Grundidee:** Was muss ein Mensch mit einer gegebenen Visualisierung, unabhängig von einer Aufgabe und der Visualisierungsart, immer machen?

**Schritt 1 Suchen:** Der Mensch muss die relevanten Informationen in der Visualisierung finden. Im Idealfall ist klar welche Informationen das sind und wo sie zu finden sind.

**Schritt 2 Query:** Die relevanten Informationen müssen aus der Visualisierung gelesen werden. „Lesen“ heißt hier: dass die Codierung als visuelle Struktur zurückübersetzt, wird in die Datenvariablen oder Items.

**Schritt 3 Interpretation:** Die gelesenen Informationen müssen im Zusammenhang mit der (Teil-) Aufgabe und auch dem Wissen des Nutzers interpretiert werden. Dies kann auch unabhängig von der Visualisierung geschehen.



Diese drei Schritte werden typischerweise nicht nur einmal, sondern mehrfach durchlaufen. Vor einer Interpretation können dabei auch mehrere Queries stehen. Vor jeder Query können mehrere Suchen ablaufen. Im Idealfall unterstützt eine Visualisierung möglichst alle Schritte. Im folgenden Beispiel „verfolgen“ wir die Aufmerksamkeit eines Anwenders, um zu bewerten, ob und warum eine Visualisierung „funktioniert“.

## Suche

Es ist beim Design generell hilfreich zu wissen, was man bei einem Anwender voraussetzen kann.

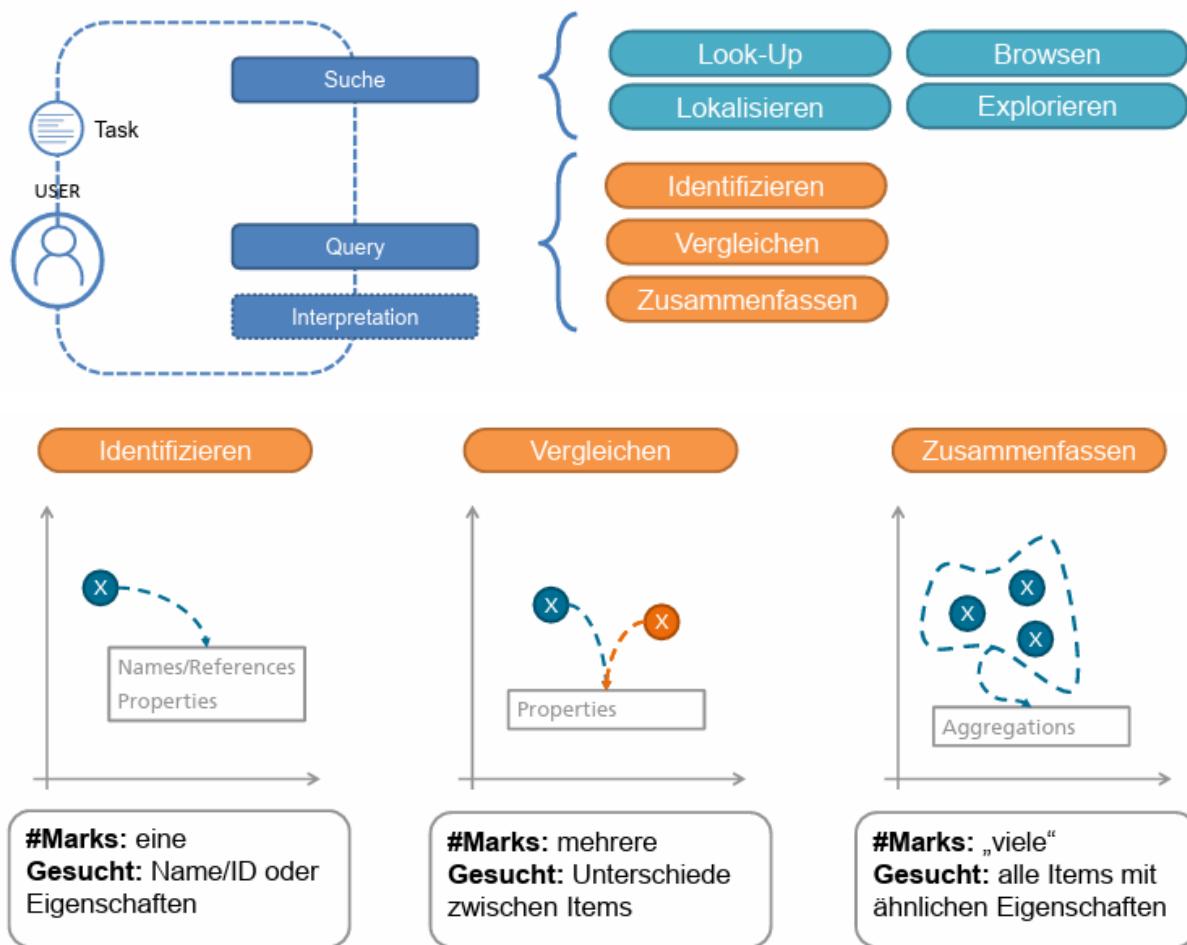
	Sie wissen, wonach Sie suchen	Sie kennen höchste Eigenschaften, von dem, was Sie suchen
Wie wissen, wo Sie etwas finden.	Look-Up	Browsen
Sie müssen erst herausfinden, wo Sie etwas finden.	Lokalisieren	Explorieren

Die vier verschiedenen Suchszenarien haben unterschiedlich hohe Anforderungen. Die Unterscheidung der Szenarien ist nicht nur eine Frage der Aufgabe, sondern die Vertrautheit eines Anwenders über die Inhalte der Visualisierung und die Vertrautheit mit der räumlichen Strukturierung der Visualisierung.

Bei „Browse“/„Explore“ sind Eigenschaften von dem, was gesucht wird, nicht zwingend gegeben, sondern müssen manchmal selbst erst gesucht werden. Ein Suchszenario ist nicht nur abhängig von der Aufgabe, sondern auch von dafür eingesetzter Visualisierung.

„Explore“ ist durch die wenigen bekannten Information das anspruchsvollste Szenario.

## Queries



## Eigenschaften visueller Channels

**Auswahl/ Hervorhebung:** Der Anwender soll in der Visualisierung ein bestimmtes Item finden. Welches Item sticht hier besonders hervor?

**Selektive** visuelle Channels helfen bei der **Lokalisierung** bestimmter Markierungen. Markierungen mit einer einzigartigen Ausprägung werden ohne Suche gefunden. Die meisten der „wichtigen“ Channels sind selektiv (Ausnahmefälle bei Form). Dabei ist die Wahrnehmung der Hervorhebung abhängig vom Kontrast und ein Channel für die Hervorhebung ist nicht für die Darstellung anderer Eigenschaften nutzbar. Die Nutzung mehrerer visueller Channels zur Hervorhebung verschiedener Markierungen ist ebenfalls nicht sinnvoll.

**Ordnung:** Sie wollen in der Visualisierung eine Ordnung sichtbar (und nutzbar) machen. Welche Markierung liegt jeweils „zwischen“ den anderen beiden?

**Ordinal** visuelle Channels helfen beim qualitativen **Vergleich** (also für ordinale Datenvariablen). Ausprägungen der visuellen Channels sind natürlich geordnet (also z.B. groß-mittel-klein). Die Channels müssen für den Vergleich nicht von einer Skala gelesen werden.

Sonderfälle:

- Position: die Nutzung von Position als ordinalem Channel dient nicht nur dem Vergleich, sondern vor allem der Strukturierung der Daten. Als Sortierung erleichtert sie jede **Suche**
- Orientierung: kann für zyklische Ordnungen oder für lineare Ordnungen genutzt werden

**Differenz:** Sie wollen in der Visualisierung z.B. Trends vergleichen, also numerische Änderungen über die Zeit. Wo ist die Differenz oben, wo ist sie unten größer?

**Quantitative** visuelle Channels helfen beim quantitativen **Vergleich**. Unterschiede von Ausprägungen der visuellen Channels können verglichen werden. Länge (und Position) sind die einzigen visuellen Channels bei denen das sicher so ist.

Voraussetzungen:

- Skalen sind linear – logarithmische Skalen eignen sich nur für den qualitativen Vergleich
- Skalen beinhalten den Nullpunkt

**Zusammenfassen:** Sie wollen in der Visualisierung ähnliche Items zu Gruppen zusammenfassen. Wo sehen Sie welche Gruppierung (schneller, deutlicher)?

**Assoziative** visuelle Channels helfen dabei, ähnliche Items als Gruppen wahrzunehmen.

Explain: „Ähnlichkeit“ kann durch eine Abbildung von kategorischen Datenvariablen vorgegeben werden (explizit als Gleichheit/Ungleichheit).

Explore/Enjoy: „Ähnlichkeit“ kann auch eine kombinierte Wahrnehmung mehrerer visueller Channels sein („Muster“).

Die Wahrnehmung verschiedener Muster kann kollidieren. Meistens kann man die Wahrnehmung zwischen Mustern bewusst umschalten.

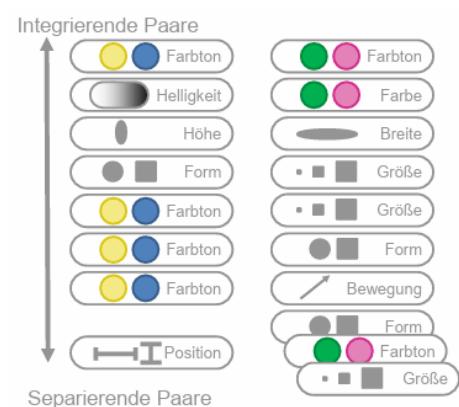
	Gruppierung	Selektion / Hervorheben	Vergleich-Anordnung	Vergleich-Quantitäten	#unterscheidbare Werte (ca.)
<b>Position</b>	+	+	+	+	display size
<b>Länge</b>	-	(+)	+	+	5-15
<b>Größe</b>	-	+	+	-	5-15
<b>Farbsättigung</b>	-	+	+	-	5-7
<b>Textur</b>	+	+	+/-	-	5-7
<b>Farnton</b>	+	+	-	-	7-8
<b>Orientierung</b>	+	+	○	-	4-6
<b>Form</b>	+	○	-	-	5-7 „neutrale“

Mehr dazu in der nächsten Vorlesung

## Einflussfaktoren

Die Wahrnehmung von Unterschieden in fast allen visuellen Channels ist immer abhängig von Verhältnis zwischen Kontrast und Entfernung. (Kontrast ist hier nicht nur Farbkontrast).

Priorisieren Sie, welche Unterschiede wichtig sind, und unbedingt wahrnehmbar sein sollten. Die wichtigsten Unterschiede sollten räumlich nah beieinander liegen, aber ansonsten einen hohen Kontrast haben.



Die **perzeptuelle** Länge beschreibt die Anzahl unterscheidbarer Werte für jeden Channel, auch bei schwierigeren Bedingungen. Das sind meist nicht viele. Bei der Darstellung/Unterscheidung von Kategorien ist das besonders relevant.

**Separierende** Paare erlauben mehr Datenvariablen in der gleichen Visualisierung.

**Integrierende** Paare erlauben mehr unterscheidbare Werte für eine Datenvariable.

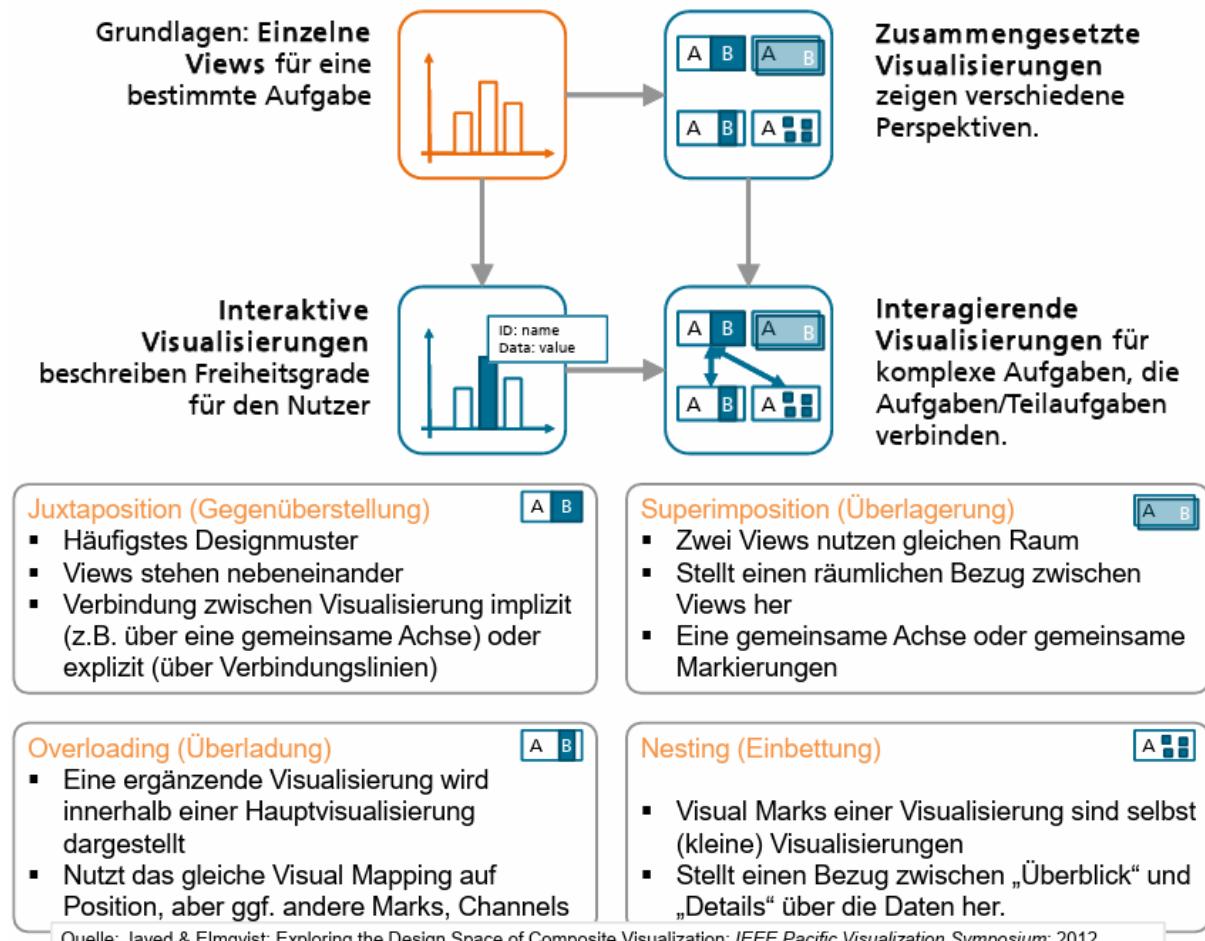
## Position und Layout



Jede Visualisierung gibt dem Raum eine Struktur.

Sobald diese Struktur vertraut ist, wird die Suche erleichtert. Eine einmal „gelesene“ Struktur kann auf benachbarte Visualisierungen angewandt werden (z.B. gemeinsame Achsen oder Legenden).

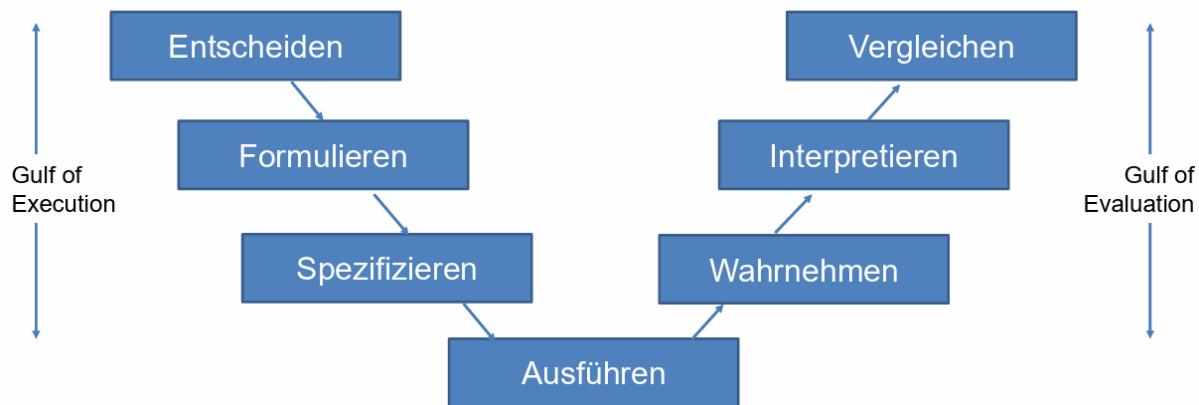
## Komplexe Visualisierungen



## Interaktion

Wozu Interaktion? Antizipation aller Fragestellungen der Nutzer unmöglich, alles auf einen Blick zeigen ist häufig nicht möglich, nutzergetriebene Veränderung der Darstellung, Exploration und neue Einblicke in die Daten.

## Bedingung und Interaktion



Bei den Übergängen zwischen den Stufen bereitet immer viele Probleme. Es sollte klar sein, welche Funktionen/ Interaktionen mit dem System durchgeführt werden können und der Zustand des Systems sollte leicht erkennbar sein.

Die **Benutzerschnittstelle** (eigentlich: Benutzungsschnittstelle) ist die Stelle oder Handlung, mit der ein Mensch mit einer Maschine in Kontakt tritt.

Eine **computergestützte** Benutzungsschnittstelle (nach DIN EN ISO 9241-110) beinhaltet alle Bestandteile eines interaktiven Systems (Software oder Hardware), die Informationen und Steuerelemente zur Verfügung stellen, die für den Benutzer notwendig sind, um eine bestimmte Arbeitsaufgabe mit dem interaktiven System zu erledigen.

## Interaktionsmodi (Spence)

### Kontinuierliche Interaktion

Kontinuierliche Veränderung in der Visualisierung, performante Reaktion auf die Aktion des Nutzers. Dabei müssen Change Blindness und peripheres Sehen beachtet werden.

### Schrittweise Interaktion

Interaktion entlang verschiedener Schritte im Rahmen der Aufgabenerfüllung und Entscheidungsfindung. Navigation von einem Schritt zum nächsten.

**Sensitivity**, im Sinne der Aufnahme von Signalen in der Umgebung/der Darstellung.

- Bei Spence motiviert durch Sozialwissenschaft, bei der „Sensitivity“ die zwischenmenschliche Aufmerksamkeit bezeichnet, die uns Menschen befähigt zu entscheiden, wie man auf bestimmte Signale reagieren kann oder soll
- Sensitivity bezogen auf Movement (SM): Welche Bewegungen kann ich ausführen? Wohin kann ich mich bewegen?
- Sensitivity bezogen auf Interaktion (SI): Welche Handlung/Interaktion ist dafür nötig?

**Affordance**: Aufforderungscharakter von Objekten

**Residue**: Hinweis auf weiter entfernte Informationen

## Passive Interaktion

Nutzer verbringen auch bei interaktiven Visualisierungen einen Großteil ihrer Zeit mit Augenbewegungen, Wahrnehmung und kognitiven Verarbeitungsprozessen. Passive Interaktion beinhaltet auch sich ändernde Repräsentierungen, aus denen die Benutzer wichtige Informationen ziehen.

## Gemischte Interaktion

Kombinationen aus kontinuierlicher, schrittweiser und passiver Interaktion.

## Interaktionsdynamik

Siehe Kapitel Mentale Modelle

## Antwortzeiten

- Animation, fließende Bewegungen, kontinuierliche Interaktion: 0,1s
- Reaktion des Systems auf Benutzeraktion: 1,0s
- Akzeptable Antwortzeit auf kompliziertere Anfrage: 10s (5s-30s)
- Visuelle Hinweise auf Verzögerungen > 1s durch Sanduhr, Fortschrittsbalken, o.ä.

## Interaktionstechniken

Interaktionstechniken bezeichnen die Möglichkeiten des Nutzers, direkt oder indirekt Datenrepräsentierungen zu manipulieren und zu interpretieren.

Taxonomien der Informationsvisualisierung mit Bezug zu Interaktion:

- Systemnahe Interaktionstechniken
- Dimensionen der Interaktionstechniken Spence (2007): Interaktionsmodi (kontinuierlich, schrittweise, passiv, zusammengesetzt)
- Interaktionsoperatoren Ward and Yang (2004): Interaktionsräume (Screen-space, data value-space, attribute space, etc.)
- Benutzeraufgaben Amar, Eagan, Stasko (2005): Werte abrufen, Filtern, Extremum finden, Anomalien finden, Korrelieren, etc.
- Kategorien der Interaktion

## Systemnahe Interaktionstechniken

### Selektion:

Ziel ist es, ein bestimmtes Objekt zu identifizieren oder eine Teilmenge zu definieren.

Feedback durch Markierungen oder eines ausgezeichneten visuellen Attributs für Teilmenge.

Fitts Law:  $Selektionszeit = a + b \cdot \log_2 \left( \frac{D}{W} + 1 \right)$ , mit Distanz zum Zentrum des Ziels (D), Größe/Ausdehnung des Ziels (W) und den empirischen determinierten Konstanten in ms (a,b).

### Navigation:

- Tastatur (1D-relativ)
- Maus/Touch (2D-absolut)
- Space-Mouse (6D-relativ)
- Trackingsysteme (6D-relativ/absolut)

- Ausnutzung zweihändiger Steuerung
- Sprache
- Kamerametaphern
  - World-in-Hand (6D-Handle)
  - Eyeball.in-Hand (6D-Handle)
  - Gehende Kamera (3D-Tracking)
  - Fliegende Kamera (6D-Handle)
- Zoom + Pan
  - Geometrischer Zoom
  - Semantischer Zoom (zusätzliche Informationen)
- Unterstützung durch Visualisierung
  - Lokale Orientierungshilfe
  - Globale Orientierungshilfe

### **Shneidermans Mantra:**

1. Überblick über alle Daten
2. Zoom und Filter
3. Details auf Anfrage

Alternative für visuelle Suche und große Datenmengen:

1. Suche
2. Zeige den Kontext
3. Expandiere auf Anfrage

### **Fokus+Kontext (Fisheye-View, Magic Lens):**

Variation der perspektivischen Transformation, schafft Raum abhängig vom Benutzerfokus.  
 Lokale Erhöhung des Detaillierungsgrads und effektiv kombinierbar mit LOD-Methoden.  
 Variation visueller Attribute (Magical Lens), beeinflusst Wahrnehmung räumlicher Beziehungen und erhält Korrespondenz (Kontext).

### **Überblick+Detail:**

Hierarchische Kopplung zweier/ mehrerer Visualisierungen, lokale Erhöhung des Detaillierungsgrades und räumliche Beziehungen bleiben erhalten aber Verdeckung und Korrespondenz können zu Problemen führen.

### **Brushing & Linking:**

Anwendbar insbesondere in Multiple Coordinated Views (verschiedene Perspektiven auf verschiedene Informationen). Auswahl (Brushing) von Elementen in einem View, um korrespondierende Daten (Linking) in den anderen Views hervorzuheben.

## **Kategorien der Interaktion**

Kategorien nach Yi et al. mit dem Ziel, die Verbindung zwischen Interaktionstechnik und Benutzeraufgaben zu stärken.

- Selektion: Markiere etwas als interessant
- Exploration: Zeige mir etwas anderes
- Rekonfiguration: Zeige mir eine andere Zusammenstellung
- Encodierung: Zeige mir eine andere Repräsentierung

- Abstrahieren/Spezialisieren: Zeige mir mehr oder weniger Details
- Filtern: Zeige mir etwas unter Bedingungen
- Bezug: Zeige mir Beziehungen der Elemente

## Interaktionsdesign

### Leitsätze

**Navigation:** Standardisierung von Arbeitsabläufen, klare Einbettung des Ziels bei eingebetteten Links (Residue), eindeutige und sprechende Überschriften, Optionsfelder für ausschließliche Auswahl, Entwurf von Seiten, die sich ohne Weiteres drucken lassen, Nutzung von Thumbnails als Vorschau auf größere Bilder.

**Organisation der Anzeige:** Konsistenz der Datenanzeige, effiziente Informationsaufnahme durch den Benutzer, minimale Gedächtnisbelastung, Flexibilität und Individualisierbarkeit, Anzeige nur von hilfreichen Informationen/ Daten, Graphische Darstellung anstelle von Text und Zahlen, Design der Anzeige in schwarz-weiß mit schrittweiser Einführung von Farben, wo hilfreich für Arbeitsaufgabe, involviere den Nutzer beim Design.

**Erzeugung von Aufmerksamkeit:** Intensive Farben für wichtige Aspekte reservieren, Markierung, Größe, maximal 3 Fonts, Vorsichtig mit blinkenden Elementen, nur in limitierten Bereichen, maximal 4 Standardfarben, weitere für gelegentliche Nutzung, Audio.

**Unterstützung der Dateneingabe:** Konsistenz aller Dateneingabe-Transaktionen, Minimierung der Input-Aktionen durch den Nutzer, Minimierung der Gedächtnisbelastung, Flexibilität bei der Dateneingabe.

### Prinzipien

#### Ermittlung des fachlichen Niveaus:

- **Anfänger** (allgemein bzgl. computergestützten Benutzungsschnittstellen) und Erstnutzer (einer spezifischen Software) benötigen intensive Hilfedialoge
- Sachkundige, **gelegentliche Nutzer** benötigen konsistente Abläufe, wiederkehrende Lösungsmuster, sinnvolle, aber konzise Meldungen
- **Experten** (Power User) verlangen schnelle Antwortzeiten, Feedback im Hintergrund, Shortcuts, Abkürzungen oder andere beschleunigte Dialoge

#### Ermittlung der Arbeitsaufgaben:

Häufigkeit von Tasks als wichtiger Maßstab

- Häufige Arbeitsaufgaben (z.B. Tastenkombination für Cut & Paste)
- Weniger häufige Aufgaben (z.B. Auswahl in Menüleiste zur Einrichtung eines Druckjobs)
- Seltene Aufgaben (z.B. mehrere Menüselektionen oder das Ausfüllen eines Formulars zur Änderung der TCP/IP-Protokollparameter)

#### Wähle einen Interaktionsstil:

Direkte Manipulation

- Kommandozeile
- Eingabeformular
- Menüauswahl
- Direkte Manipulation
- Spracherkennung

#### Die 8 goldenen Regeln der Gestaltung:

1. Konsistenz, wo immer möglich (Abläufe, Farben, Layout, Fonts, Menüs, etc.)
2. Möglichst universelle Benutzbarkeit (Anfänger bis Experten, Altersgruppen, Behinderungen, technische Vorlieben)
3. Informatives Feedback für jede Benutzeraktion
4. Design von Dialogen, die eine Gruppe von Aktionen zum Abschluss führen
5. Verhinderung von Fehlern
6. Einfaches Rückgängigmachen von Aktionen
7. Unterstützung eines Gefühls der Kontrolle über jeden Aspekt der Benutzungsschnittstelle
8. Reduzierung der Gedächtnisbelastung (Daumenregel: 7±2 Informationsbrocken können gleichzeitig im Arbeitsgedächtnis gehalten werden)

## Menschliche Reaktionszeit

Hick-Hyman-Gesetz:  $Reaktionszeit = a + b \cdot \log_2(C)$ , mit Anzahl der Auswahlmöglichkeiten ( $C$ ) und empirischen Konstanten  $a, b$ .

Unter optimalen Bedingungen ( $C=1$ ) 160ms + Antworthandlung,  $C=8$  480ms + Antwort

Schnellere Antwortzeit, wenn Menschen gelegentlich Fehler machen dürfen (Fehlertoleranz)

# Techniken

## Mehrdimensionale Visualisierungen

Fast alle diese Techniken sind nicht dafür geeignet (oder gedacht) irgendwo einzelne Datenwerte abzulesen. (Sie nehmen ja auch keinen Globus her, um eine Straße zu finden). Stattdessen dienen Sie in erster Linie dazu Werteverteilungen sichtbar zu machen, um Muster, Ausreißer oder Abhängigkeiten zwischen den Daten zu finden. Das Ablesen (wenn notwendig) kann im Einzelfall durch Interaktion (z.B. Tooltips, oder Details per Mausklick) ermöglicht werden.

### nD-quantitative Techniken

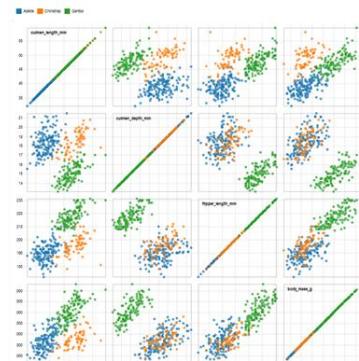
#### Scatterplot Matrix

Eine Scatterplot Matrix ist ein nested Layout, bei dem Attributnamen auf Positionen abgebildet werden (Spalten und Zeilen der Matrix) und die Attributwerte werden auf Positionen abgebildet (Koordinaten innerhalb der Spalten und Zeilen). Alle Dimensionen sind hier gleichberechtigt.

Scatterplot Matrizen sind für 4-8 Dimensionen geeignet und ist die getreueste Darstellung paarweiser Abhängigkeiten.

Weitgehend unabhängig von der Anordnung der Spalten und Zeilen und kaum Verzerrung der Wahrnehmung.

Aber skaliert schlecht für viele Datenpunkte und zeigt nur paarweise Relationen.



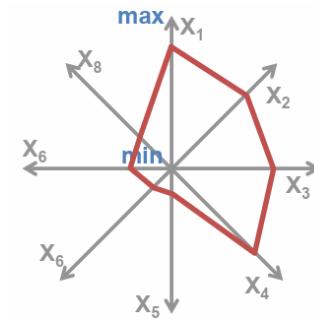
#### Starplot

Auch bekannt als Netzdiagramm, Radar-Charts, Spider Chart oder Polar Chart. Die Idee ist hierbei mehrere Achsen in einer sternförmigen Anordnung zu zeigen.

Jede Richtung definiert eine Skala ( $360^\circ$  werden gleichmäßig aufgeteilt), auf jeder dieser Skalen wird ein Datenpunkt abgetragen und diese Punkte werden miteinander verbunden.

Man kann entweder einen Plot für mehrere Datensätze verwenden, dies sorgt für einen einfachen Vergleich aber ist nur für wenige unterschiedliche Daten geeignet oder man verwendet einen Plot pro Datensatz, dadurch hat man weniger Overplotting aber der Vergleich ist schwierig.

Small Multiple Starplots sind je ein Plot für je einen Datensatz. Eine geeignete Anordnung hilft bei Small Multiple Visualisierung.



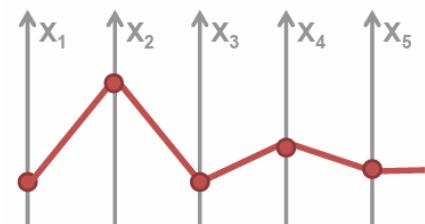
Starplots sind für 4-15 Dimensionen geeignet und können leicht über Formenwahrnehmung verglichen werden. Die Korrelationen von benachbarten Achsen sind gut sichtbar, Flächen sind leichter zu erkennen als Linien.

Aber Starplots sind nur für wenige Daten geeignet und abhängig von der Anordnung der Achsen.

### Parallele Koordinaten

Parallele Koordinaten verwenden ein ähnliches Prinzip wie Starplots nur dass hier die Koordinatenachsen parallel angeordnet sind.

Jede der parallelen Achsen definiert eine Skala und auf jeder dieser Skalen wird ein Datenpunkt abgetragen und diese Punkte werden miteinander verbunden.



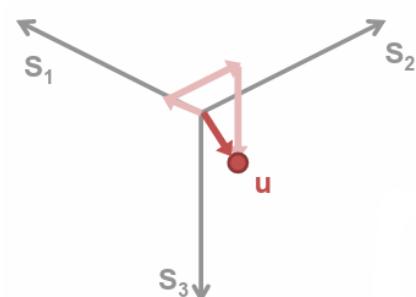
Parallele Koordinaten sind weit verbreitet und verwenden eher selten Small Multiples. Häufig gibt es bei parallelen Koordinaten Overplotting.

Das Layout unterstützt beliebig viele Achsen und Korrelationen lassen sich leicht erkennen aber Linienzüge können in Kreuzungen nicht verfolgt werden, außerdem beeinflusst die Achsenordnung und -richtung die Wahrnehmung, Korrelationen sind nur in direkt benachbarten Achsen erkennbar.

Viele Probleme von Parallelen Koordinaten können behoben werden durch Interaktion, z.B. durch Filter für Datenkategorien, Range Filter pro Achse, Achsen umsortieren/invertieren und linken von Gruppenliste und Sampleliste.

### RadViz

RadViz verwendet ein radiales Layout, ähnlich zum Starplot, hierbei ziehen hohe Werte an einer Achse einen Punkt zum Achsenende.



Datenpunkte werden erstellt anhand der gewichteten Summe der Achsenvektoren, Datenwerte müssen vorher min-max normiert werden.

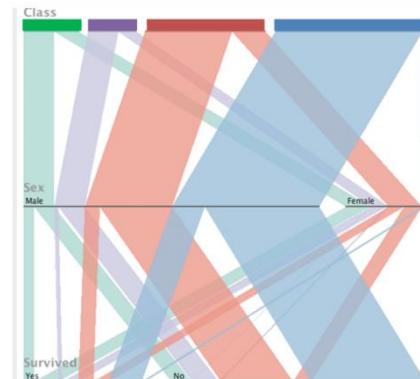
RadViz ist platzsparend und zeigt gut relative Datenverhältnisse aber die Achsenanordnung bestimmt die Position und die Punktposition ist durch die Normierung nicht eindeutig.

## nD-nominale Techniken

### Parallel Sets

Parallel Sets eignen sich für mehrere kategorische Variablen, eine wählbare Variable bestimmt zusätzlich die Farbe.

Bei Parallel Sets werden die Achsen parallel angeordnet, die Kategorien werden entlang der Achsen platziert. Die Anzahl der Items in einer Kategorie bestimmt die Breite dieser Kategorie. Die Breite und Reihenfolge bestimmen dann die Position. Alle Achsen mit zwei gemeinsamen Kategorien benachbarter Achsen werden als Trapez dargestellt, die Items werden zu einer Menge abstrahiert. Ein Attribut bestimmt die Farbe.



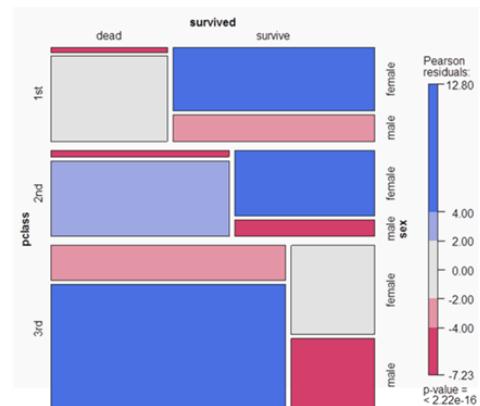
Parallel Sets sind gut für viele Daten und Dimensionen geeignet, aber sie sind abhängig von der Anordnung und der Reihenfolge der Kategorien. Overplotting kann in zwei Varianten auftreten, häufige Kategorien dominieren seltene und ist abhängig von der Reihenfolge, in der die Kategorien gezeichnet werden.

Auch hier kann Interaktion einige Schwächen ausgleichen durch z.B. verändern der Achsen- oder Kategorienreihenfolge, hervorheben von interessanten Teilmengen und Auswahl des Attributs, dass die Farbe bestimmt.

### Mosaic-Plot

Der Mosaic-Plot basiert auf der rekursiven Teilung von Flächen. Bei jeder Teilung wird ein anderes Attribut gewählt und die Richtung gewechselt. Die Teilung entspricht der Häufigkeit. Die Teilung ist möglich bis zur Bildschirmauflösung.

Der Mosaic-Plot verwendet effizient den Raum und skaliert für beliebig viele Daten. Die Abhängigkeit zwischen den ersten beiden Attributen ist immer gut sichtbar, die Abhängigkeiten zwischen der n. und n-1. Variable ist schwieriger darzustellen.

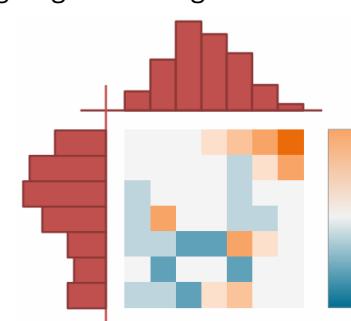


Der Mosaic-Plot ist schwer lesbar bei mehreren Variablen und allgemein schlecht lesbar und vergleichbar. Nullwerte brauchen eine extra Behandlung.

### KV-Map

Die KV-Map (Karnaugh-Veitch) wird mit mehreren nominalen Attributen (<20) verwendet. Auch hier wird eine rekursive Unterteilung eingesetzt. Aber die Unterteilung ist gleichmäßig und unabhängig von der Häufigkeit der Kategorien, stattdessen wird die Häufigkeit auf die Farbe abgebildet.

Die rekursive Unterteilung beginnt mit einem Attribut, dieses wird äquidistant unterteilt. Eine KV-Map mit zwei Attributen ist eine Matrix, in den Zellen der Matrix werden die Anzahl der Items gezählt, auf die beide Attribute zutreffen. Die Farben sind nicht die einfachen Häufigkeiten, sondern zeigen an, ob die beiden Attribute statistisch unabhängig sind. Weitere Dimensionen werden als



zusätzliche gleichmäßiger Unterteilung hinzugefügt, so lassen sich schnell mehrdimensionale Abhängigkeiten als Muster erkennen. Die Balken der Häufigkeiten sind nicht Teil der KV-Map.

KV-Maps sind gut geeignet für bis zu 10 Attributen und zeigen als einzige die Abhängigkeiten zwischen allen Attributen und nutzt die Fähigkeit, Frequenzen und Wiederholungen wahrzunehmen.

Die Mustererkennung in KV-Maps muss erst gelernt werden und die Muster sind nicht interpretierbar.

## nD Techniken

### Tabellarische Visualisierungen

Allgemeines Layout mit Spalten für Attribute und Zeilen für Items aber Werte werden graphisch und nicht durch Text dargestellt. Die Werte einer Zelle können verschieden dargestellt werden, z.B. durch Farbe oder Balkenlänge.

Tabellen sind eine vertraute Darstellung und sind für viele Attribute geeignet (als Heatmap), im Extremfall sogar ein Datensatz pro Pixel. Die Attribute können nominal oder quantitativ sein. Die Sortierbarkeit zeigt die Korrelation über viele Attribute. Tabellen sind aber auf mehrere hunderte Datensätze beschränkt.

## Übersicht

Name	Datentypen der Dimensionen	#Dimensionen	#Datensätze	Anordnung ohne Einfluss?	Lesbarkeit [mit Interaktion]	Aufgabe
Scatterplot-matrix	Quantitativ	4-8...	100-10000	++	Gut [Sehr Gut]	Paarweise Korrelationen finden und klassifizieren
Starplot	Quantitativ [Nominal*]	4-15...	10-100	---	Mittel [Gut]	Vergleich in vielen Dimensionen
Parallele Koordinaten	Quantitativ [Nominal*]	Ca. 30 (Bildbreite)	100-1000	--	Schlecht [Gut bis Sehr Gut]	Paarweise Korrelationen finden, Suche in nD
RadViz	Quantitativ	4-15	10-1000	---	Schlecht [Eher Schlecht]	Lineare Abhängigkeiten finden
Parallel Sets	Nominal [Quantitativ*]	Ca. 30 (Bildbreite)	„unendlich“	--	Mittel [Gut bis Sehr gut]	Paarweise Korrelationen , finden, Suche in nD
Mosaic Plot	Nominal [Quantitativ*]	4-6	„unendlich“	--	Schlecht [Mittel]	Häufigkeiten vergleichen, Paarweise Abhängigkeiten
KVMap	Nominal [Quantitativ*]	4-10	„unendlich“	-	Extrem Schlecht [Mittel → VA]	Korrelationen in nD finden, Muster in nD finden
Tabellen	alle	30+	100-1000	+	Gut [Sehr Gut]	Vergleichen, Korrelationen finden

## Darstellung von Big Data – Ordnen-Methoden

Der Begriff Big Data ist immer abhängig vom Problem.

### Dimensionsreduktion

Die Dimensionsreduktion ist ein Grundproblem in der Visualisierung, da der Bildschirm nur zwei Dimensionen hat und mehr Dimensionen nur bedingt wahrnehmbar und verständlich sind. Mehr Dimensionen benötigen mehr Berechnungen und redundante Dimensionen verzerrten die Ergebnisse.

**Curse of Dimensionality:** Je mehr Eigenschaften für zwei Items bekannt sind, desto wahrscheinlicher unterscheiden sie sich auch in irgendwas.

**Definition:** Gegeben seien Datensätze  $X$  mit ( $n > 2$ ) Attributen  $x = (x_1, x_2, x_3, x_4, \dots, x_n) \in X$ ) und eine Funktion  $dist_n$ , die einen Abstand/ Unterschiede zwischen zwei Datensätzen messen kann. Eine Dimensionsreduktion sucht allgemein eine Abbildung  $red: X \rightarrow \mathbb{R}^2$ , sodass gilt für alle Paare von Datensätzen  $x, y \in X$ .  $dist_n(x, y)$  verhält sich wie  $dist_2(red(x), red(y))$ .  $dist_2$  ist der euklidische Abstand, aber  $dist_n$  muss nicht der euklidische Abstand sein. Eine Abbildung  $red$  erhält also Unterschiede und Gemeinsamkeiten in den Daten. Ein großer Abstand vor der Reduktion, soll also einem großen Abstand nach der Reduktion entsprechen.

Dimensionsreduktion ist nicht einfach, dann in  $n$  Dimensionen  $n+1$  Objekte gleich weit voneinander entfernt sein können. Bei weniger Dimensionen hat man weniger Optionen, Unterschiede zu beschreiben.

## Feature Selektion

Definition: Gegeben seien Datensätze  $X$  mit ( $n > 2$ ) Attributen  $x = (x_1, x_2, x_3, x_4, \dots, x_n) \in X$ ). Eine Feature Selektion wählt aus diesen  $n$  Attributen  $m$  „relevante“ Attribute aus ( $m < n$ ). Allgemein lässt sich Relevanz so definieren: Wenn man vor der Selektion mit den Vektoren  $x$  eine Aussage über eine weitere Eigenschaft des Items machen kann (z.B. als Vorhersage einer unbekannten Eigenschaft), dann soll die gleiche Aussage auch mit  $s$  möglich sein.

## Dimensionsreduktionsverfahren

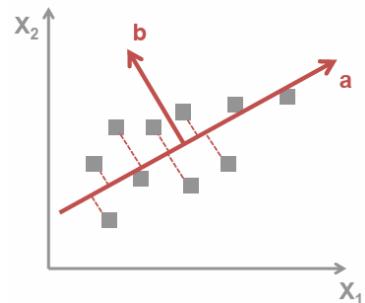
### PCA (Hauptkomponentenanalyse)

PCA eignet sich gut für numerische Daten mit linearer Abhängigkeit, wenn keine weiteren Informationen vorhanden sind. Andere Datensätze brauchen andere Methoden. PCA basiert auf der Varianzmaximierung und ist eine gängige Methode.

Es wird vorausgesetzt, dass die Achsen im ursprungsraum  $X$  orthogonal zueinanderstehen. Die neuen Dimensionen sind Linearkombinationen aller anderen:

- 1. Achse  $a: a_1 \cdot x_1 + a_2 \cdot x_2 + \dots + a_n \cdot x_n$
- 2. Achse  $b: b_1 \cdot x_1 + b_2 \cdot x_2 + \dots + b_n \cdot x_n$
- N. Achse  $w: w_1 \cdot x_1 + w_2 \cdot x_2 + \dots + w_n \cdot x_n$

An der 1. Achse wird die größte Varianz maximiert, an der 2. Achse die zweitgrößte Varianz usw.



Bei der Visualisierung werden normalerweise die ersten Hauptkomponenten (Da diese den Großteil der Variation beinhalten) verwendet. Informationsverlust pro fehlender Hauptkomponente kann berechnet werden, es handelt sich um eine verlustbehaftete Kompression.

### LDA (Linear Discriminant Analysis)

LDA benötigt gelabelte Daten und Sucht neue Achsen, um die Labels gut zu unterscheiden. Die neuen Achsen sind wieder lineare Kombinationen der ursprünglichen Achsen.

Gesucht werden Achsen, die das Fischer-Kriterium erfüllen: Mittelwerte der Klassen sind maximal weit auseinander und Varianz innerhalb der Klassen ist möglichst klein.

$$J(w) = \frac{|\widetilde{\mu}_1 - \widetilde{\mu}_2|}{\widetilde{s}_1^2 + \widetilde{s}_2^2} = \frac{\text{Distanz der Zentren}}{\text{Streuung}}$$

## MDS (Multidimensional Scaling)

Daten werden im niedrigdimensionalen Raum so geordnet, dass ihre Distanzen aus dem Originalraum im Zielraum gut abgebildet sind.

$$\min \sum (d(p_i, p_j) - d(p'_i, p'_j))^2$$

Funktioniert auch für Daten, von denen wir nur die Entfernung wissen, modifiziert auch für nominale Daten möglich. Diese, nicht-lineare Transformation ist ein Optimierungsproblem und hat eine iterative Lösung. Es gibt verschiedene Varianten von Zielfunktionen.

## SOM (Self-Organizing Map)

Zeitserien repräsentieren hier beliebige hochdimensionale Daten. Wie können diese Daten auf zwei Dimensionen arrangiert werden? Ähnliche Datensätze sollen auf ähnliche Positionen.

1. Jede Zelle wird mit einer zufälligen Zeitserie initialisiert (einem „Prototypen“). Dies entspricht dem Zentroiden beim k-Means.
2. Jede Zeitserie, wird in jene Zelle mit dem jeweils ähnlichsten Prototyp zugeordnet.
3. Die Prototypen in einer Zelle und ihre direkten Nachbarn (blau) werden verändert, so dass die den Originaldaten der Zelle ähnlicher werden. Die fernen Nachbarn (rot) werden verändert, so dass sie den Originaldaten unähnlicher werden.
4. Schritte 2 und 3 werden wiederholt, bis sich nichts mehr ändert. (Die Größe der Nachbarschaft in Schritt 3 wird dabei kontinuierlich kleiner.)

SOM als lernendes neuronales Netz, das sich an die Datentopologie (Anordnung der Daten im Originalraum) anpasst, dabei sind sich benachbarte Knoten ähnlich und Objekte werden durch eine nicht lineare Transformation in Knoten gruppiert.

## Übersicht

Verfahren	Besonderheiten/Stärken	Schwächen
PCA	Lineares Verfahren. Erhält die Varianz der Datenverteilung im Originalraum. Einfach, geschlossene Lösung.	Kann nicht-lineare Zusammenhänge nicht abbilden. 
LDA	Lineares Verfahren, benötigt gelabelte Daten, Sucht Achsen, die Labels möglichst gut trennen	Kann nicht-lineare Zusammenhänge nicht abbilden.
MDS	Nicht-Lineares Verfahren, Benötigt nur Distanzen, keine Punkte.	Lösung wird nur approximiert.
SOM	Nicht-Lineares Verfahren. Auch für die Gruppierung der Daten geeignet.	Lösung wird nur approximiert. Relativ viele Parameter

In keinem der Verfahren haben die neu erzeugten Achsen eine Bedeutung, außer dass sie die originalen Achsen kombinieren.

## Visualisierung Zeitbasierter Daten

Nicht alle Channel sind für zeitbezogene Daten geeignet und die Wahl der Markierung bestimmt die Interpretation.

### Filtern

Wie in vielen anderen Visualisierungen ist auch bei Zeitreihen Overplotting ein Problem. Hierfür kann man nach Verlauf filtern. Durch interaktives Brushing wird eine Teilmenge der Zeitreihen definiert.

### Heatmaps

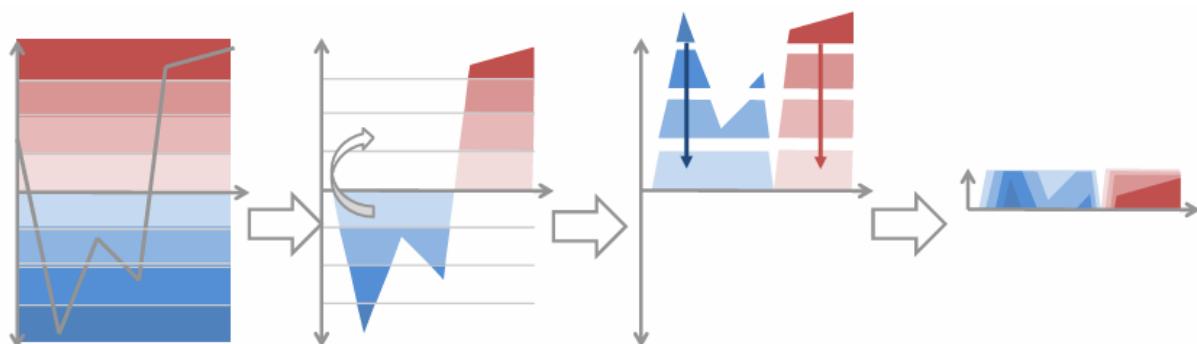
Eine weitere Methode, um gegen Overplotting vorzugehen sind Heatmaps. Dabei ersetzt man die Höhe durch Farbe und erhält so eine Value Heatmap. Daten sind nicht aggregiert, die Zeitreihe wird durch wenige Pixel Höhe dargestellt. Dadurch erhält man eine gute Vergleichbarkeit entlang der Zeitachse aber weniger gute Vergleichbarkeit auf der Werteachse.

Heatmaps können auch mit Aggregation verwendet werden, hier wird pro Pixel eine normalisierte Liniendichte bestimmt, so wird auf wenige Details der Zeitserien reduziert.

### Horizonplots

Horizonplots sind eine der neueren Visualisierungsmethoden, dabei wird ein Kompromiss zwischen einem Liniendiagramm und einer Heatmap gebildet. Horizonplots sind gut geeignet, wenn wenig Höhe für die Werteskala zur Verfügung steht. Werte werden nicht auf einer Farbe, sondern auf einen Farbverlauf abgebildet.

Jedem Wertebereich wird ein Farbband zugeordnet (divergierende Farbskala). Die Farbbänder werden an der Kurve abgeschnitten und der negative Bereich wird auf die positiven gespiegelt. Alle Farbbänder werden übereinander auf die Nulllinie gelegt.



Horizonplots sind eine platzsparende Technik für die Darstellung von Verläufen und Trends und sind besser auf der Werteachse vergleichbar als Heatmaps. Horizonplots skalieren auf etwa 50-100 parallele Zeitreihen. Aber es ist schwer den genauen Wert an einer Stelle abzulesen, vor allem wenn die y-Achse mehrfach belegt ist.

### Small Multiples

Small Multiples sind eine weitere Variante, um gegen Overplotting vorzugehen. Dabei werden viele, gleiche Visualisierungen mit jeweils verschiedenen Datensätzen. Darunter leidet die Vergleichbarkeit, nur nahe benachbarte Zeitreihen gut vergleichbar, der vertikale und horizontale Vergleich ist verschieden gut möglich.

Auch Small Multiples können mit Aggregation verwendet werden. Dabei werden Gruppen von ähnlichen Zeitreihen gebildet. Dies erfordert Gruppierung/ Clustering der Datensätze nach ähnlichen Zeitreihen, was ein aufwendiger Prozess ist. Small Multiples skaliert durch Gruppierung und Aggregation auf deutlich mehr Daten als die anderen Ansätze.

## Periodische Zeitreihen

Viele natürliche Abläufe wiederholen sich, hier möchte man die Vergleichbarkeit über mehrere Zyklen hinweg betrachten. Es ist notwendig, die Periodenlänge und die sich ändernde Frequenz zu bestimmen.

### Spiral Layouts

Bei Spiral Layouts wird die Zeitachse aufgerollt. Polarkoordinaten erzeugen dabei eine Spirale.

$$x = \text{rad} \cdot \frac{t}{\text{cyc}} \cdot \cos\left(360^\circ \cdot \frac{t}{\text{cyc}}\right)$$

$$y = \text{rad} \cdot \frac{t}{\text{cyc}} \cdot \sin\left(360^\circ \cdot \frac{t}{\text{cyc}}\right)$$

Die Zykluslänge  $\text{cyc}$  ist potenziell variabel, der Zyklus lässt sich als Muster erkennen.

Spiral Layouts sind kompakt aber leiden unter Verzerrung (Zeitachse wird nach außen breiter) und funktioniert nur bedingt als Liniengrafik.

### Matrixlayout

Bei matrixbasierten Layouts wird die Zeitachse gestapelt. Die View Transformation basiert auf Division mit Rest.

$$x = t \bmod \text{cyc}$$

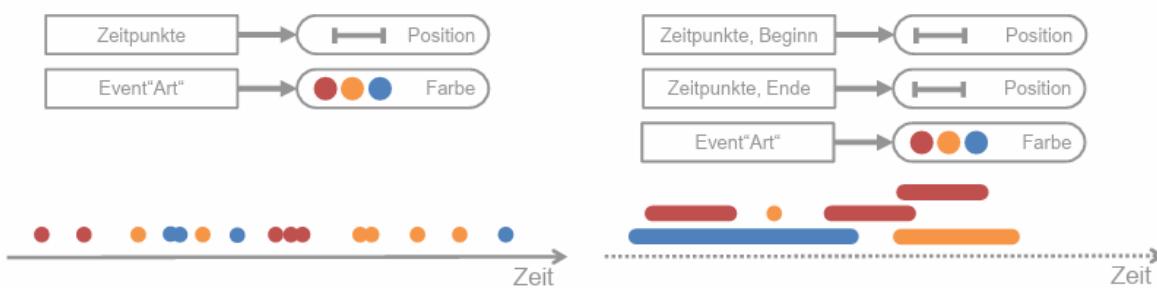
$$y = t \div \text{cyc}$$

Dieser Ansatz funktioniert auch bei Liniendiagrammen und ebenfalls gilt, die Zykluslänge  $\text{cyc}$  ist potenziell variabel, der Zyklus lässt sich als Muster erkennen.

## Diskrete Wertearchse

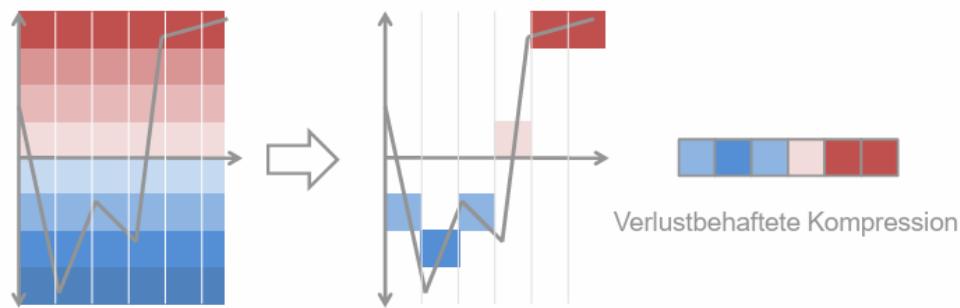
Ereignisse haben einen Zeitpunkt, aber nicht unbedingt ein quantitatives Attribut, die y-Achse ist nicht als kontinuierliche Skala verwendbar.

Ist die Dauer zwischen bestimmten Events wichtiger als die Zeitpunkte, kann man die Events als **Gantt-Chart** darstellen.

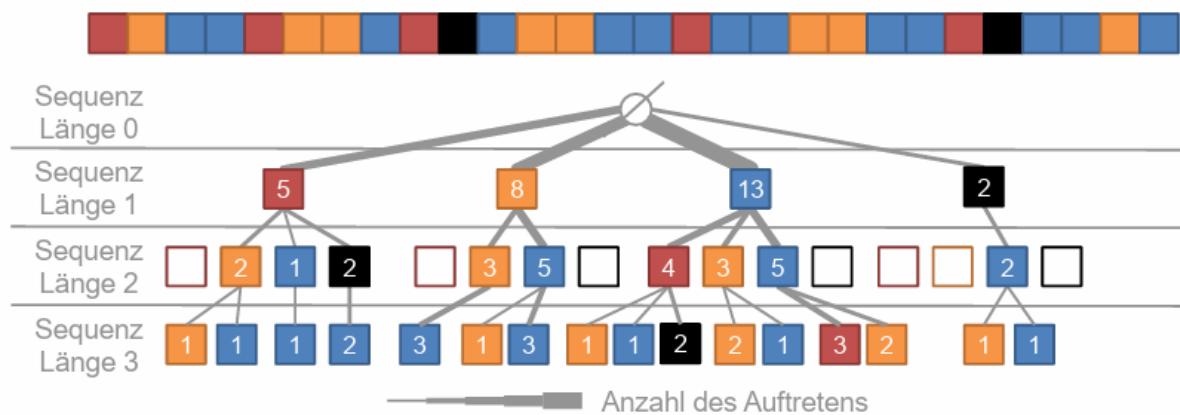


## Sequenzbaum

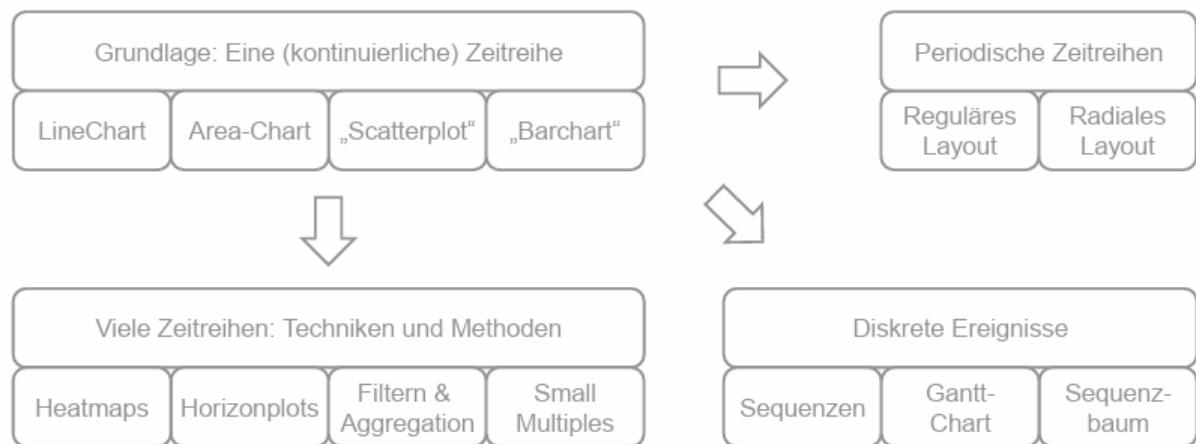
Beim Sequenzbaum wird eine Zeitserie in wiederkehrende Muster zerlegt. Der Sequenzbaum basiert auf Eventfolgen oder diskretisierten Werten.



Im Sequenzbaum zählt man die Länge der Sequenzen einer bestimmten Länge. Für Länge 1 ist das die Statistik der Häufigkeit von Events. Für Länge 2 zählt man entsprechend wie häufig ein Eventtyp auf einen bestimmten anderen folgt. Interessant sind die Sequenzen, die über-/unterdurchschnittlich häufig auftauchen. Die häufigsten Sequenzen werden dabei als Breite der Linie dargestellt.



## Übersicht



## Graphen

**Definition:** Ein Graph  $G = (E, V)$  besteht aus einer Menge von Knoten  $V$  und einer Menge von Kanten  $E \subset V \times V$ .  $V$  ist eine endliche Menge und  $E$  beschreibt eine allgemeine Relation auf  $V$ . Der Graph ist mindestens durch eine Kantenliste definiert. Die Kantenliste kann auch Attribute der Kanten enthalten.

## Bäume

Definition: Eine Folge von Knoten ist ein Weg, wenn aufeinanderfolgende Knoten durch Kanten verbunden sind. Ein Graph ist zusammenhängend, wenn zwischen je zwei Knoten ein Weg existiert. Ein Graph ist gerichtet, wenn die Kanten nicht-symmetrisch sind,  $(v_1, v_2) \in E \neq (v_2, v_1) \in E$ . Ein Graph ist zyklusfrei, wenn es keinen Weg gibt, der einen Knoten mehrfach erreicht. Ein Baum ist ein zyklusfreier, zusammenhängender Graph.

Die Eigenschaften beschreiben, ob der Graph eine Art von Ordnung beschreibt. (Da gibt es verschiedene). Mit einer Ordnung lassen sich diese Graphen deutlich einfacher darstellen.

### Node-Link Diagramm

Mit der Ordnungsrelation kann eine der beiden Koordinaten definiert werden, Kanten werden explizit dargestellt. Ein Kreuzungsfreies Layout ist immer möglich.

Gütekriterien für strukturiertes Layout (nicht alle sind immer erfüllbar)

- Keine Kreuzungen
- Alle Knoten einer Hierarchiestufe auf gleicher Höhe
- Möglichst schmal
- Elternknoten zentriert über ihren Kindknoten
- Symmetrien sollten erkennbar sein
- Ordnungserhaltend
- Linearzeit

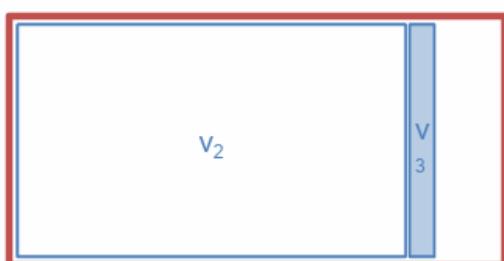
Optimieren eines Baums auf einem Gitter, verschieben von Knoten und vertauschen von Teilbäumen. Ein großes Problem ist dabei die Baubreite. Hiergegen kann man die Bäume radial anordnen. Die Hierarchieebenen bilden Kreisringe um den Wurzelknoten.

### Treemap

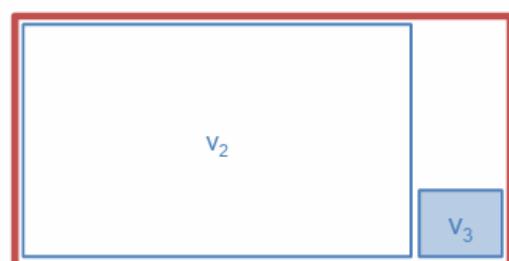
Bei Treemaps werden Beziehungen zwischen Knoten dargestellt als „enthalten sein in“. Wurzelknoten repräsentiert durch die ganze Fläche der Visualisierung. Kindknoten entstehen entsprechend durch Unterteilung.

Die Fläche eines Elternknotens wird rekursiv unterteilt und wird jeweils auf die Kindknoten aufgeteilt, Kanten werden nicht dargestellt. Die Größe der Flächen kann aus einer Eigenschaft der Knoten abgeleitet werden. Falls kein Größenwert gegeben ist, wird nach der Größe des Teilbaums aufgeteilt.

Bei squarified Treemaps wird sowohl horizontal als auch vertikal geteilt. Dadurch sind diese besser lesbar und vermeidet entartete Rechtecke, aber direkte Eltern-Kind-Beziehungen sind nicht mehr eindeutig.



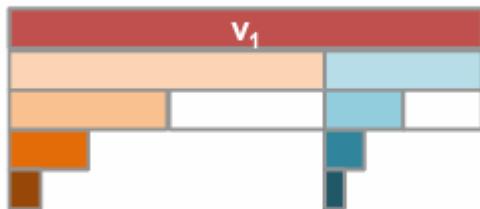
Normal Treemap: Unterteilung immer gleich



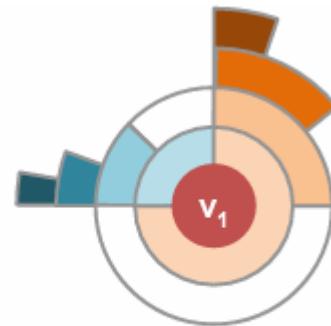
Squarified Treemap: Unterteilung wechselt

## Icicle Plot

Ein **Sunburst** ist ein Icicle Plot mit radialem Layout. Kanten werden nicht explizit dargestellt. Kindknoten stehen unter ihrem Elternknoten (Icicle Plot= oder an der Außenseite ihres Elternknotens (Sunburst)).



Icicle Plot



Sunburst

## Graphendarstellung

Node-Link Diagramme	Adjazenzmatrizen
<ul style="list-style-type: none"> <li>Pfade nachverfolgbar</li> <li>Gerichtete Kanten explizit</li> <li>Für theoretisch 10k Knoten</li> </ul>	<ul style="list-style-type: none"> <li>Keine Überlappung</li> <li>Dichte Graphen gut darstellbar</li> <li>Gerichtete Kanten -&gt; Asymmetrie</li> <li>Nur eine Layout-Dimension</li> </ul>
<ul style="list-style-type: none"> <li>Zwei Layout-Dimensionen</li> <li>Dichtbesetzte Graphen kaum lesbar</li> </ul>	<ul style="list-style-type: none"> <li>Verfolgen von Pfaden schwieriger</li> <li>Dünnbesetzte beeinflusst die Struktur</li> <li>Sortierung beeinflusst die Struktur</li> <li>Für hunderte Knoten</li> </ul>

### Kriterien für Node-Link Graphenlayout

- Unnötige Kantenüberschneidungen vermeiden
- Overplotting vermeiden: Knoten nicht übereinander
- Verbundene Knoten sollten nahe beieinander sein
- Stark verbundene Teilgraphen erkennbar
- Gegebener Platz soll genutzt werden
- Benachbarte Knoten im Durchschnitt gleich weit voneinander

Allgemeines Graphlayout ist vielleicht das schwierigste algorithmische Visualisierungsproblem. Verwandte Problemstellungen wie Clustering oder Dimensionsreduktion sind ähnlich schwierig, aber erscheinen nicht nur bei der Erstellung einer Visualisierung. Es handelt sich um ein allgemein ungelöstes Problem.

## Force-Directed Layout

Eines der ersten und bekanntesten Verfahren für diese Klasse von Verfahren ist das Fruchterman-Reingold Layout. Dieses Layout modelliert die Knoten und Kanten als Masse-Feder System. Ergänzung durch abstoßende Kräfte, unabhängig von den Kanten.

- Anziehende Kräfte nehmen linear mit dem Abstand zu
- Abstoßende Kräfte nehmen quadratisch mit dem Abstand ab

Das Layout entsteht durch Simulation der Kräfte und Bewegen der Knoten gemäß den Kräften, Knoten bewegen sich also nur während der Simulation. Dieses Layout ist geeignet für Graphen mit hunderten Knoten und mehreren Parametern.

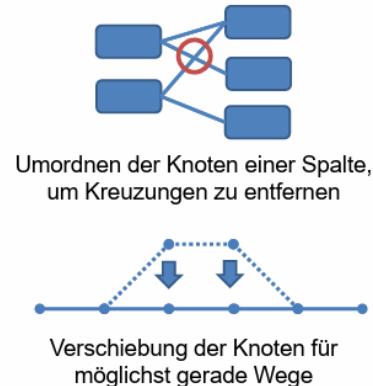
Es ist ein einfaches Layout und erlaubt Interaktion, Kanten und Knoten können Gewichte erhalten. Das Layout hat keine optimale Lösung und eine lange Laufzeit.

### Layer-Based Layout

Das Sugiyama Layout ist ein Layer-Based Layout. Layer-Based Layouts sind Layoutverfahren für gerichtete, azyklische Graphen. Es ist für allgemeine Graphen anwendbar, wenn man Schleifen entfernt, den Graphen künstlich richtet und Schleifen danach gesondert darstellt. Dabei gibt die Kantenrichtung eine Hauptrichtung vor und die Knoten werden in Spalten eingesortiert. Die Höhe eines Knotens in der Spalte ist zunächst unbestimmt.

Der Algorithmus beginnt am vorderen Ende, eine Iteration testet einfache Modifikationen am Graphen (Tauschen und verschieben von Knoten innerhalb einer Spalte). Jede Modifikation wird bewertet, falls keine Verbesserungen mehr möglich sind, wird die nächste Spalte betrachtet.

Auch dieses Verfahren ist einfach, strukturiert azyklische Graphen und definiert eine Hauptrichtung des Graphen. Graphen mit wenigen Zyklen müssen gesondert behandelt werden. Für Graphen mit vielen Zyklen sind diese Layouts nicht brauchbar.



### Constraint Based Layout

Das bekannteste Beispiel sind Metro-Maps, dabei liegen Knoten auf einem Gitter, Kanten verlaufen nur in bestimmten Winkeln. Kantenkreuzungen häufig in Knoten. Knotenform ist an die Kreuzungsart angepasst. Auch hier kommt ein ähnliches Optimierungsverfahren wie beim Sugiyama Layout zum Einsatz, allerdings mit den Kriterien Kantenwinkel, Kantenlänge, Kreuzungswinkel, Geradlinigkeit, Kantenreihenfolge pro Knoten, Getreue Wiedergabe einer Geoposition. Die Kantenpfade liefern zusätzliche Struktur.

Constrained-Based Layouts sind anwendbar auf allgemeine Graphen aber Knoten und Kantenverhältnis im Graphen sollte ausgewogen sein. Die Layouts sind visuell stark strukturiert und einfach zu lesen. Aber Pfade müssen gegeben falls definiert sein, es gibt nur eine lokale Lösung und teilweise komplexe Heuristiken.

### Edge-Bundling

Das Ziel des Edge-Bundlings ist die visuelle Komplexität von Graphenlayouts durch Bündeln zu reduzieren. Die Kanten werden nicht als kürzeste Wege gezeichnet, sondern in Kantengruppen zusammengelegt.



Direkter Weg (rot) wird ersetzt durch indirekten Weg (blau)

Eine Variante ist hierarchisches Edge-Bundling. Statt des direkten Wegs durchläuft eine Kante die Hierarchie bis zum gemeinsamen Vorfahren, dabei wird der Kantenzug über Splinekurven approximiert, dadurch durchlaufen mehrere Kanten den gleichen Pfad (werden gebündelt)

### Interaktion

Nach dem Prinzip Search, Show Context, Expand on Demand. Komplette Graphen sind für viele Aufgaben nicht notwendig. Die Darstellung sollte auf die wichtigen Knoten fokussiert sein (Zuletzt interagiert, deren Nachbarn). Vor dem Layout werden die Knoten des Graphen sinnvoll gefiltert. Das Layout wird auf die Aufgabe vereinfacht, Fokus auf eine im Detail erfassbare Menge, aber Verbindungen zu den nicht sichtbaren teilen ist sichtbar.

## Geobasierte Visualisierung

### Karten

Nutzt man Karten als Visualisierung, kann man mehr Vorkenntnisse eines Lesers vorraussetzen. Kenntnisse über die Orte und Topografie gibt es in der realen Welt in unterschiedlichen Detailgraden. Kenntnisse über häufig verwendete visuelle Metaphern in der Darstellung sind weit verbreitet, wie z.B. Straßenkarten (z.B. Rangordnung) Topografie der Karten (z.B. geographische Höhe → Höhentlinien) oder Politische Karten (Länder) Vorkenntnisse erlauben reichhaltige Assoziationen zum eigenen Weltwissen, dies erleichtert die Interpretation und (bei Ortskenntnis) die Suche. Aber Karten zeigen nicht immer Geografie, Kartenmetaphern werden erfolgreich auch für die Darstellung abstrakter Daten genutzt.

### Visualisierung geobezogener Daten

Der Ausgangspunkt sind geografische Orte gegeben durch Koordinaten oder Ortsnamen, diese können in Form von Punkten, Linien oder Flächen sein.

- |   |  |
|---|--|
| <ul style="list-style-type: none"> <li>▪ Darstellung und Vergleich von Distanzen</li> <li>▪ Abschätzung räumlicher Verteilungen und Flächen</li> <li>▪ Suche und Identifikation von geographischen Entitäten</li> </ul> | <ul style="list-style-type: none"> <li>▪ Fokus auf abstrakten Eigenschaften</li> <li>▪ „Debiasing“ von Flächen</li> <li>▪ Neudefinition von „Distanz“/„Ähnlichkeit“</li> </ul> |
|---|--|



Die bekannteste Form der Abbildung sind View-Transformationen auf 2D oder 3D Koordinaten.

Eine Globus Darstellung ist allgemein schwierig.

### Plattkarte

Koordinaten (Länge, Breite) werden direkt auf (x,y) abgebildet. Kreise werden im Norden und Süden auf Ellipsen abgebildet.

### Mercator Projektion

Die Erdkugel wird auf einen Zylinder abgebildet, die Länge auf X und  $\text{arctanh} \sin(B)$  wird auf Y abgebildet. Kreise werden im Norden und Süden aufn größere Kreise abgebildet.

### Winkel-Triple

Koordinaten werden direkt auf (x,y) abgebildet, Berechnung ist kompliziert, Alle Ellipsen in der Darstellung haben etwa gleiche Größe.

## Kartenprojektion

Um das Projektionsproblem zu umgehen, nimmt man an, dass die Erdoberfläche auf kleinen Gebieten flach ist und Fehler in der Projektion vernachlässigt werden können, wie z.B. Stadtpläne.

Eine getreue Abbildung der ganzen Erdoberfläche auf eine Kugel ist möglich, dies verlagert aber das Problem nur, die Projektion von 3D auf 2D bleibt schwierig. Eine getreue Abbildung auf 2D ist nur näherungsweise möglich. Man muss immer eine Eigenschaft festlegen, die erhalten bleiben soll, alle Eigenschaften zu erhalten ist sehr kompliziert. Die Eigenschaften sind die folgenden: Erhaltung der Nachbarschaft, Wiedererkennung der Orte, Flächentreue, Entfernungstreue, Winkeltreue.

### Erhaltung der Nachbarschaft

Das Erhalten der Nachbarschaft erleichtert die inkrementelle Suche (Navigation). Diese Navigation kann auf unterschiedlichen Detailstufen stattfinden.

Wiedererkennungswerte der Orte

Karten, die gelernten Konventionen entsprechen, erleichtern die Suche und Identifikation.

## Verzerrte Darstellung geobasierter Daten

Bei der verzerrten Darstellung wird die getreue Darstellung aufgegeben. Position wird auch durch andere Eigenschaften der Daten mit definiert, Verzerrung verändert meistens Winkel, Entfernung und Flächen. Verzerrung ändert selten Nachbarschaftsbeziehungen oder den Wiedererkennungswert. Zum Beispiel Metromap.

### Kartogramme

Kartogramme sind eine Klasse von sehr unterschiedlichen Techniken (stetige und diskrete Ansätze).

Stetige Kartogramme ist eine formerhaltende Verzerrung (Morphing). Ein Gitter mit Raumkoordinaten und Knotengewicht, das Gewicht drückt benachbarte Gitterknoten in die leereren Bereiche, diese können auch animiert werden.

Kartogramme können anstelle von geographischen Daten auch Ortspaare und Distanzen verwenden, um eine Karte zu erstellen (Dimensionsreduktion)

Diskrete Kartogramme haben keine Formerhaltung und nur mäßige Nachbarschaftserhaltung. Diskrete Kartogramme sind ein guter Kompromiss zwischen einer einfachen Grundform für z.B. Charts und einer groben Erhaltung der Nachbarschaft.

Die Verzerrung in Kartogrammen basiert auf zusätzlichen Daten. Kartogramme reduzieren auf die wesentlichen Aussagen, wobei geografische Vorteile von Karten erhalten bleiben.

## Visualisierung ortsbezogener Daten

Im Extremfall spielen geographische Position keine Rolle mehr für das Layout.

## Visualisierung nicht geobezogener Daten

Unterscheidend ist, dass Position nicht durch geographische Eigenschaften definiert wird. Herausforderung ist es hier, eine Position zu definieren.

Eine allgemeine Herausforderung ist die Spatialization, also das Abbilden von Eigenschaften der Daten auf zweidimensionale Koordinaten, hierfür können entweder die Dimensionsreduktionstechniken oder Graph-Layout-Algorithmen verwendet werden. Ein weiteres Problem ist, die relevanten Landmarken für die Orientierung auszuwählen. Karten liefern auch ein Bezugssystem für Punkte in einem Raum, wenn diese nicht geografisch sind.

Eine weitere Herausforderung ist die Konstruktion einer Topografie, die im Hintergrund dargestellt werden kann (oft, aber nicht immer), außerdem ist die Wahl geeigneter Metaphern, die Informationen und Beziehungen effektiv und konsistent(!) übersetzen wichtig.

## Kartografische Abbildungen auf Marks und Channels

Der Design-Space von Karten umfasst alle Markierungsarten und ist im Prinzip eine gemischte Visualisierung. Karten für die geografische Orientierung sind dabei häufig am komplexesten. Elektronische Versionen mischen Inhalte häufig über wählbare Layer.

### Glyphen

Glyphen sind ein kleines, unabhängiges visuelles Objekt. Sie zeigen (mehrere) Merkmale eines Items oder einer Menge von Items und haben eine eigenständige Position im Raum. Sie sind ein visuelles Zeichen, dass Eigenschaften anderer Zeichen nutzen kann.

Beim Design der Glyphen muss man einen Kompromiss zwischen der Komplexität und der erwarteten Anzahl der Glyphen finden. Wichtige Datenattribute sollten dominante visuelle Channel besetzten (Farbe, Form Größe, redundante Abbildung etc.). Glyphen sollten visuell einfacher sein, wenn Muster statt Werte relevant sind und wenn die Werteverteilung chaotisch sein können.

Es muss auch bei der Interaktion der visuellen Channel unterschieden werden, wie viele Werte sollen einzeln lesbar sein (separierende Channel) oder sollen viele Werte als Muster wahrgenommen werden (integrierte Channel).

Es sollte eine intuitive Abbildung gefunden werden, welche die Semantik in den Daten widerspiegelt. Also visuelle Eigenschaften, die leicht wiederzuerkennen oder zu vergleichen sind oder bekannte Darstellungen und Icons verwenden.

## Spatio-Temporal Data

Die Häufigste Interpretation von raum-zeitlichen Daten ist die Änderung von punktförmigen Orten über die Zeit. Die Bewegung ist meist stetig, man kann punktförmige Orte über die Zeit verbinden.

Die Trajektorie modelliert die Abhängigkeit des Raums von der Zeit in einer mathematischen Kurve.  $f: \mathbb{R} \rightarrow \mathbb{R}^{[2|3]}$ . Häufig unterscheidet man zwischen Start und Ziel und wählt Glyphen, um diese Orte und die Orte zwischen Start und Ziel zu markieren.

Es ist schwierig, Geschwindigkeit darzustellen z.B. über Samplingfrequenz, Länge, Form oder Farbe/ Helligkeit. Um viele Trajektorien darzustellen kann man diese aggregieren, filtern oder in 3D darstellen.

# Datenvorverarbeitung

Reale Daten sind oft unsauber, unvollständig, verrauscht, inkonsistent oder unglaubliche Ausreißer. Eine akkurate Datenbasis ist wichtig für verlässliche Analyseergebnisse, daher bereitet man Daten vor, um die Qualität sicherzustellen.

**Definition:** **Datenvorverarbeitung** umfasst diejenigen notwendigen **Datentransformationen**, die nicht vom Nutzer der Visualisierung während der Nutzung durchgeführt werden können oder sollen. Dies sind also Anpassungen, Bereinigungen, Strukturänderungen und Berechnungen, die durch den Entwickler (oder das Entwicklungsteam) im Vorfeld der Nutzung erfolgen müssen.

Die meiste Zeit wird mit Datenvorverarbeitung verbracht und nur ein kleinerer Teil mit Analyse und Visualisierung.

## Methoden

**Metadaten:** Metadaten sind immer hilfreich für die Datenvorverarbeitung, sie liefern Informationen zur Interpretation, Auflösung oder Referenzpunkte, Einheiten und wichtige Symbole und Schlüsselwörter.

**Statistik:** Die statistische Analyse wird zur Ausreißerkennung, Clusteranalyse, Korrelationsanalyse und für statistische Plots und Histogramme verwendet.

**Fehlende Werte:** Die folgenden Methoden sind möglich für den Umgang mit fehlenden Werten: Ignorieren, manuell Einfügen, Eliminieren des gesamten Datensatzes, Globale Konstante, Mittelwert, Wert basierend auf Ähnlichkeit. Die Methode sollte gewählt werden nach Typ und Semantik der Daten, Menge der fehlenden Werte und Expertise des Anwenders. Es sollten ebenfalls die Gründe für die fehlenden Werte ermitteln und Ersatzwerte markieren. Fehlerhafte Werte sind meistens schwer zu erkennen und durch den Menschen verursacht.

**Datenbereinigung:** Ausreißer (Outlier) sind Daten außerhalb des Datenspektrums, starke Ausreißer (Anomalies) sind Werte, welche für Experten ungewöhnlich sind. Man muss zwischen Fehlern, Anomalien und Ausreißern unterscheiden, aber die Abgrenzung zwischen Fehlern und Ausreißern ist nicht immer trivial. Ausreißer können durch die Visualisierung oder durch statistische Datenauswertung erkannt werden.

**Skalierung:** Datenskalierung auf einen bestimmten Datenintervall, hierzu gibt es viele verschiedene Varianten.

Min-Max Normalisierung  $f: \mathbb{R}^+ \rightarrow [0; 1]$

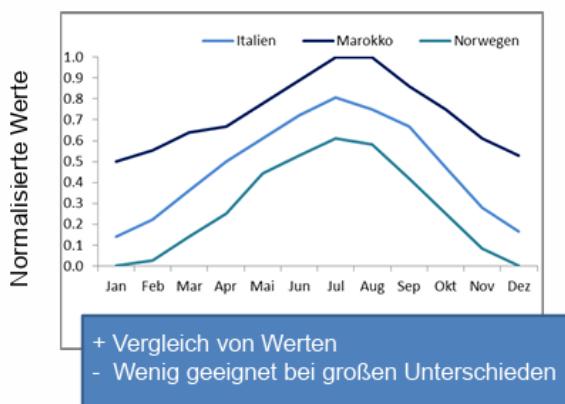
- Linear:  $f_{lin}(v) = \frac{v - min}{max - min}$
- Logarithmisch:  $f_{log}(v) = \frac{\ln v - \ln min}{\ln max - \ln min}$
- Quadratisch:  $f_{sq}(v) = \left( \frac{v - min}{max - min} \right)^2$
- Wurzel:  $f_{sqrt}(v) = \sqrt{\frac{v - min}{max - min}}$

Min-Max Normalisierung  $f: \mathbb{R}^+ \rightarrow [-1; 1]$

- Linear:  $f(v) = f_{lin}(v) \cdot 2 - 1$

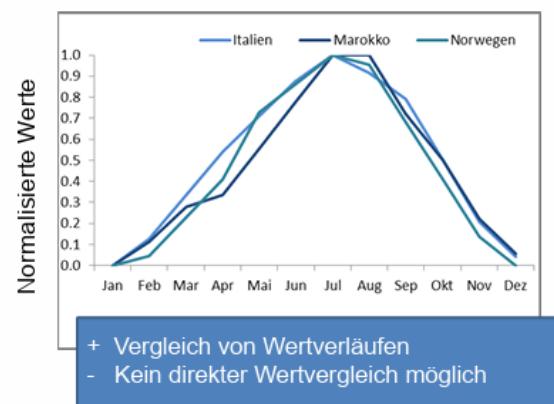
### Globale Normalisierung

Min und Max für alle Daten



### Lokale Normalisierung

Min und Max pro Datensatz



**Diskretisierung:** Betrifft hier die Vorverarbeitung kontinuierlicher Datenmengen oder Dimensionen. Extraktion einer diskreten Teilmenge als Approximation (mit Genauigkeitsverlust). Das Ziel ist den Berechnungsaufwand und den Speicherplatz zu begrenzen

**Sampling:** Sampling ist eine Methode zur Reduktion von Datenmengen. In der Visualisierung häufig zufällige Auswahlverfahren für repräsentative Stichproben/Samples zur Reduktion der zu visualisierenden Daten.

**Segmentierung:** Einteilung der Daten in zusammenhängende Abschnitte/ Regionen, die jeweils zu einer Kategorie gehören, dies ist nicht immer eindeutig möglich.

**Untermengen:** Reduktion der Datenmenge auf Untermengen, Definition über Filter und andere Einschränkungen.

**Datenintegration:** Fusion verschiedener Datenquellen durch Angleichung von Schema, Qualität etc.

## Visual Analytics

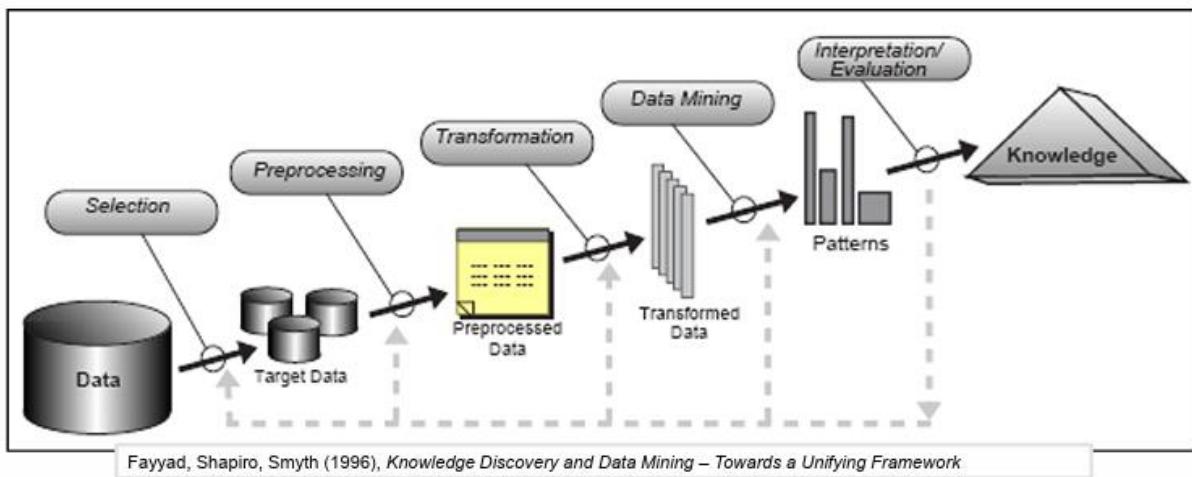
**Definition:** Visual Analytics ist die Kombination automatischer Analysetechniken mit interaktiven Visualisierungen, um große und komplexe Daten zu verstehen und auf deren Basis Entscheidungen treffen zu können.

**Definition:** Visual Analytics ist die Wissenschaft des analytischen Schließens, unterstützt durch interaktive, visuelle Schnittstellen.

**Informationsvisualisierung:** Nutzung von computerunterstützten, interaktiven, visuellen Repräsentierungen abstrakter Daten mit dem Ziel, das Erkenntnisvermögen zu verbessern.

## Knowledge Discovery (KDD)

**Knowledge Discovery in Databases (KDD):** Nicht-trivialer Prozess für die Suche nach validen, neuen, potenziell relevanten und nutzbaren Mustern in Datenbeständen.



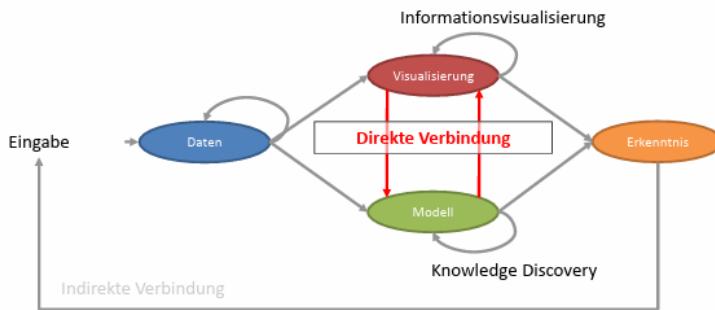
1. **Kennenlernen der Anwendungsdomäne:** Kennenlernen der Daten und der Fragestellung, Beurteilung der Relevanz von Daten, Zwischenergebnissen und der Analyse. Übersetzung der Fragestellung von der Anwendung in die Analyse.
2. **Auswahl der Daten:** Geschieht typischerweise mit dem Anwender nach den Kriterien Relevant, Repräsentativität, Vollständigkeit, Verfügbarkeit, Kosten.
3. **Datenvorverarbeitung (Wrangling):** Bewerten der Datenqualität, Datenintegration, Datenbereinigung und ggf. Merkmalsextraktion.
4. **Bestimmung der Data-Mining Aufgabe:** Aufgaben wie Clustering, Klassifikation und Regression bestimmen aus der Sprache des Anwenders in das technische Vokabular
5. **Auswahl des Verfahrens für die Aufgabe:** Für jede Aufgabe stehen hunderte Verfahren zur Verfügung
6. **Datentransformation und Reduktion:** Anpassung der Daten an die Anforderungen des Data-Mining Verfahren. Erhalt der relevanten Information ändert gegeben falls das Format und das Schema. Gemeinsam mit der Vorverarbeitung der mit Abstand aufwändigste Schritt.
7. **Data-Mining:** Anwendung des Verfahrens auf die gewählten Daten, gegeben falls Auswahl von Parametern und Qualitätskriterien. Ergebnis sind Muster und Modelle.
8. **Interpretation, Bewertung:** Sichtung und Test der Ergebnisse, Vermittlung der Ergebnisse an den Anwender und Ergebnis in den Kontext der Aufgabenstellung setzen.
9. **Nutzung:** Ergebnisse verwenden.

## Vergleich zum Visualisierungsprozess

Visualisierungsprozess	Knowledge Discovery
<ul style="list-style-type: none"> <li>• Datenflussmodelle</li> <li>• Prozessschritte laufen iterativ und interaktiv ab</li> <li>• Methoden und Techniken sind im Prinzip gleich</li> <li>• Interpretation und Evaluation durch Menschen</li> <li>• Ziel: Erkenntnisgewinn</li> </ul>	
<ul style="list-style-type: none"> <li>• Visual Mapping</li> <li>• View Transformation</li> <li>• „Bild“</li> <li>• Menschliche Wahrnehmung</li> </ul>	<ul style="list-style-type: none"> <li>• Data-Mining</li> <li>• Darstellung der Ergebnisse nicht definiert</li> </ul>

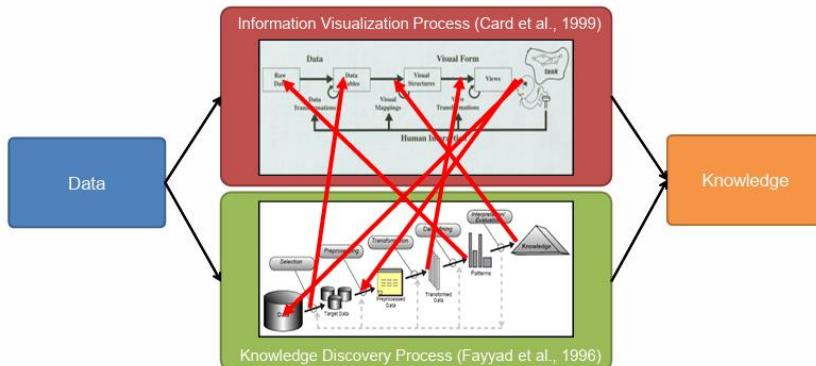
## Visual Analytics Process Modell

Die Visualisierung wird im KDD-Prozess für die Sichtungen von (Zwischen-)Ergebnissen genutzt. Visualisierungstechniken und Data-Mining Techniken werden dabei meist ohne Modifikationen nebeneinander eingesetzt. Anwender müssen zwischen der visuellen Darstellung und den automatischen Techniken übersetzen.



Direkte Verbindung zwischen beiden Tools, früher indirekte Verbindung als zwei unabhängige Tools und werden nur auf das aktuelle Teilproblem angewandt.

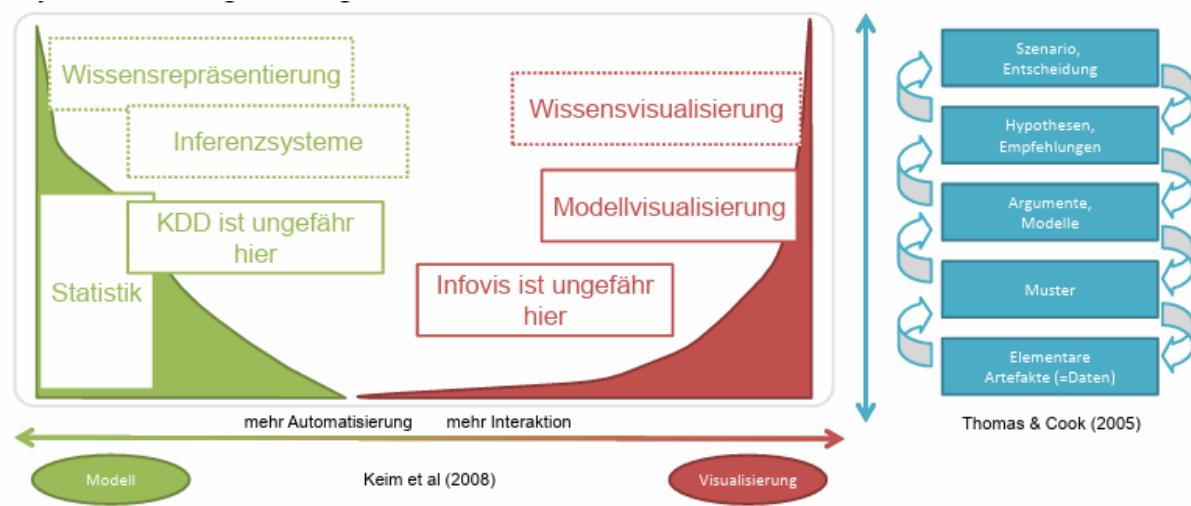
Die direkte Verbindung ist eine zentrale Idee von Visual Analytics. Techniken aus InfoVis und Knowledge Discovery wechselwirken sich während der Ausführung, z.B. kann Interaktion mit einer Visualisierung auch die Modellierung beeinflussen. Das folgende Diagramm zeigt die möglichen direkten Verbindungen



## Analytisches Schließen

Man kann allgemeine Fragen nicht auf der Grundlage einer einzelnen Beobachtung beantworten

Analytisches Schließen bedeutet das Herleiten von Entscheidungen aus empirischen Daten, Falsifikation/Bestätigung von Hypothesen und Entwicklung von Methoden für die Herleitung (Mining, Simulation, Präsentation...). Das allgemeine Ziel ist eine Nachvollziehbare Verknüpfung aller Artefakte.



## Muster

**KDD-Definition:** Ausdruck in einer formalen Sprache oder ein Modell, das eine Teilmenge aller Datensätze beschreibt.

**Definition:** Ein visuelles Muster ist ein subjektiver Sinneseindruck, der Informationen über mehrere zusammenfasst.

Muster sind eine nichtzufällige Teilmenge von Daten. Muster müssen sich aber nicht unbedingt wiederholen oder im Bild zusammenhängend sein. Ein Muster hat mehr als ein Element, und es enthält nicht alle Elemente.

Es gibt unterschiedliche Arten von **Modellen**, alle unterscheiden eine Menge X von einer Grundmenge Y. Beim KDD werden die Muster fast immer zusammen mit dem Modell erzeugt.

Ein **Beschreibungsmodell** für ein Muster  $X \subset Y$  ermöglicht es, X von  $Y \setminus X$  zu unterscheiden.

Ein **Prognosemodell** ermöglicht es, auch unbekannte  $z \notin Y$  entweder X oder  $Y \setminus X$  zuzuordnen.

### Muster in InfoVis

- Sinneseindruck
- $\Rightarrow$  Muster ohne Modell definiert
- Räumlich begrenzter Bereich des Sichtfelds
- Subjektiv
- Nicht formalisiert
- Nicht (direkt) kommunizierbar

### Muster im KDD

- Ausdruck in formaler Sprache
- $\Rightarrow$  Muster durch Modell definiert
- $\Rightarrow$  [Lesbar]
- Objektiv
- Reproduzierbar
- „nachweisbar nichtzufällig“

### Gemeinsame Eigenschaften

- Fassen mehrere Datensätze zu einem Artefakt (dem Muster) zusammen

## Stärken und Schwächen von Mensch und Maschine

Wenn eine Maschine Muster erkennen soll: Modellbeschreibung (a.k.a. Algorithmus) notwendig:

- (-) Suche oder Entwicklung des geeigneten Verfahrens schwierig
- (-) Frühe Eingrenzung der Suche auf wenige Varianten von Mustern
- (+) Ergebnis ist eine formale, nutzbare Beschreibung

Wenn ein Mensch Muster erkennen soll: Geeignete Visualisierung notwendig:

- (+) Muster können auch dann gefunden werden, ohne dass die vorher beschrieben werden müssen
- (+) Menschen können die Erkennung komplexester Muster lernen
- (-) Folgerung: Ergebnisse sind abhängig vom Anwender
- (-) Visuelle Muster sind nicht direkt nutzbar

#### Stärken des Menschen

- Flexible, robuste Wahrnehmung
- Mustererkennung unter schwierigsten Bedingungen
- Wahrnehmung ungewöhnlicher und unerwarteter Ereignisse
- Umgang mit Unsicherheit
- Kreativität
- Langzeitgedächtnis, Weltwissen
- Urteilskompetenz
- [Induktives Schließen]
- Abdiktives Schließen  
(vom Ergebnis auf eine Prämisse, Hypothesenbildung)

#### Stärken der Maschine

- Kontrollierbarkeit
- Wiederholbarkeit
- Großer Arbeitsspeicher
- Deduktives Schließen  
(von Prämisse und Regel auf das Ergebnis)
- Schnelligkeit
- Präzision
- Zuverlässigkeit
- Multitasking
- Ausdauer

#### Schwächen des Menschen

- Kleines Arbeitsgedächtnis
- Kognitiver „Flaschenhals“ - sequentielle, sprachliche Verarbeitung
- Umgang mit hochdimensionalen Daten
- Geringe Ausdauer
- Abhängigkeit von äußeren Einflüssen
- Nichtreproduzierbarkeit von Ergebnissen
- Abdiktives Schließen (Falsche Prämissen)

#### Schwächen der Maschine

- Umgang mit Rauschen in den Daten
- Umgang mit Mehrdeutigkeiten
- Umgang mit Datenunsicherheit
- Formale Beschreibung notwendig
- Sehr begrenzte Anpassung an neue Reize
- Umgang mit sehr hochdimensionalen Daten

Wenn hierzu noch tiefergehendes Verständnis notwendig ist, bitte VL-Foliensatz Nummer 9 ab Slide 63 beachten.

## Analyse für die Visualisierung

### Visualisierung mit automatischen Verfahren

Bei Visualisierungen ist der Bildschirmplatz begrenzt. Das menschliche Arbeitsgedächtnis ist begrenzt, dadurch können bei zu vielen Daten, die einzelnen Daten nicht zu Informationen verarbeitet werden. Um dies zu verbessern kann man die Datenmenge reduzieren durch Sampling, Aggregation/ Clustering und Filtern oder man „bereinigt“ die Visualisierung durch visuelle Aggregation, Hervorheben oder Layout. Um die Visualisierung weiter zu verbessern kann man die Verfahren Dimensionsreduktion, Feature Subset Selektion und Feature Sortierung verwenden. Ergebnisse einer Visualisierung sind häufig nur informell, dagegen kann man Historymechanismen und Organisationswerkzeuge einsetzen.

Der Mensch erkennt häufig Muster, wo keine sind, die visuelle Auffälligkeit entspricht nicht der statistischen Auffälligkeit. Interaktion führt dazu, dass erwartete Resultate hervorgehoben werden, verschiedene Visualisierungen schlechter vergleichbar sind und ist immer mit Lernaufwand verbunden. Mögliche Verbesserungen sind eine Automatische Steuerung oder Nutzerführung durch Analyse der Daten / Parametertest und Analyse der Interaktion

Praktisch jeder Teil des InfoVis-Modells ist Ansatzpunkt für automatische Verfahren, Inputseitig durch Datenverarbeitung und Outputseitig durch Interaktionsverarbeitung.

## Automatische Verfahren

**Sampling:** Beim Sampling wird die Anzahl der Datenelemente reduziert, die für eine Visualisierung noch sinnvoll sind. Dies geschieht typischerweise zufällig ist aber nicht immer ganz einfach.

**Aggregation:** Zusammenfassen von Datenelementen zu Mengen und Visualisierung dieser Mengen. Darstellung von dynamischen Änderungen eines Netzwerks durch Aggregation entlang der Zeitachse.

**Clustering:** Häufig als Vorverarbeitungsschritt, erzeugen eines Datenattributs ClusterID, diese ID wird wie normale Daten behandelt.

**Text/ Topic Mining:** Topics beschreiben Themenspektrum von Textdokumenten.

- Cluster von Dokumenten („in welchen Dokumenten wird Thema X behandelt“)
- Cluster von Wörtern („welche Terme tauchen beim Thema X immer wieder auf“)
- Extrem hochdimensional (#Wörter + #Docs)
- Multiclusering

**Filtering:** Degree-of-Interest (DOI) Berechnungen, bestimmen der Gewichtung für einzelne Daten, dabei beschreiben die Nutzerinteraktionen den Fokus, der Filter definiert dann einen Kontext um diesen Fokus.

**Dimensionsreduktion:** Suche nach neuen Dimensionen, die Informationen besser zusammenfassen. Kombination von fast immer numerischen Daten zu neuen Dimensionen.

- Lineare Projektion (Kamerametapher) nD->2D:  $y = ax_1 + bx_2 + cx_3, z = ox_1 + px_1 + qx_3$
- Lokal-lineare Projektion: Projektionsebene ist nicht flach
- Nicht-lineare Reduktion:  $y = \frac{ax_1}{b+x_2}, z = x_1^2 + x_2^2$

**Feature Subset Selektion:** Visualisierung kann nur n Features (=Attribute) darstellen. Suche nach Kombination von n existierenden Features, mit maximal viel Information. Suche Features, die statistisch unabhängig sind und entferne Duplikate durch Ähnlichkeitssuche.

**Feature Sorting:** Visualisierung ist (oft) abhängig von der Anordnung der Features, Mustererkennung ist Glückssache, heuristische Suche nach guten Anordnungen. Bei der Matrixsortierung kann man Quadrate auf der Hauptdiagonalen sichtbar machen durch Sortierung der Zeilen und Spalten.

**Ergebnismanagement:** Aus Interaktion werden relevante Muster identifiziert, diese Muster werden als Mengen abgespeichert und als neuer Datentyp interpretiert.

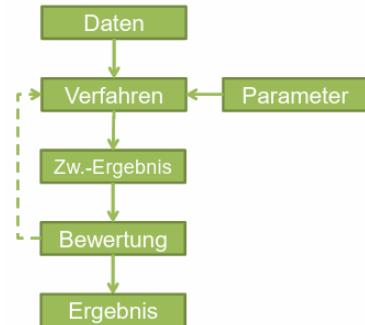
**History-Management:** Analyseergebnisse werden über längere Zeit aufgezeichnet und beinhalten Muster, Visualisierungseinstellungen und Notizen, dadurch wird die Historie sichtbar.

**Intelligent Visual Analytics Queries:** Suchen nach optimalen Parametern durch Ausprobieren viele Parameterkombinationen. Bei IVAQ werden aus Interaktionen automatisch relevante

Muster identifiziert und die Beziehung zwischen diesen charakterisiert /modelliert, daraus wird eine automatische Steuerung abgeleitet.

## Verbesserung der Modellierung durch Visualisierung

Automatische Verfahren funktionieren meist nach demselben Prinzip. Man sollte mehr als nur die Daten visualisieren. Das Ergebnis ist häufig ein Modell, welches Abhängigkeiten zwischen Variablen beschreibt, und dadurch gefundenes Wissen repräsentiert, diese sind aber oft komplex. Das Modell ist nicht sichtbar. Hierfür setzt man **Modellvisualisierung** ein um die Modelle sichtbar zu machen, dabei betrachtet man das Modell als Datensatz.



Das Ergebnis kann aber auch ein Muster sein. Hier muss man mithilfe der Visualisierung die Gemeinsamkeiten und Unterschiede finden. Die **Musterexploration** hilft hierbei. Die Muster können entweder als **Details** (Informationen über die Elemente werden dargestellt), **Abstraktion** (Muster wird als einziges Datenelement betrachtete) oder **Mengenvisualisierung** (Direkte Darstellung der Muster als Menge) dargestellt werden.

Für die Visualisierung von Modellen und Mustern lassen sich im Prinzip die gleichen Methoden und Kriterien nutzen wie bei der Visualisierung von Daten.

1. Gibt es evtl. eine „natürliche“ Repräsentierung für das Modell (z.B. als Baum, Graph, Tabelle, Vektoren)?
2. Wie stellt sich der Anwender das Modell vor?

## Von Daten zu Mustern

Clusteringverfahren berechnen eine automatische Gruppierung von großen Datenmengen nach ähnlichen Eigenschaften. Clustering wird fast immer ohne Vorwissen ausgeführt.

Dimensionsreduktion dient der Suche nach relevanten Eigenschaften in den Daten. Auch Dimensionsreduktion wird fast immer ohne Vorwissen ausgeführt.

Sei ein Datenobjekt ein Vektor  $\vec{v}$  von Attributen.

**Clustering** ordnet jedem Datenobjekt genau einen Cluster zu, so dass Objekte im gleichen Cluster so ähnlich wie möglich sind und Objekte in unterschiedlichen Clustern so unterschiedlich wie möglich sind. Zusammenfassen von Tabellenzeilen (Datenobjekten).

**Dimensionsreduktion**  $\rho$  ordnet jedem Datenobjekt einen kürzeren Vektor  $\rho(\vec{v})$  zu, so dass Ähnlichkeit und Unähnlichkeit zwischen Vektoren  $\vec{v}_1$  und  $\vec{v}_2$  in der Abbildung  $\rho(\vec{v}_1)$ ,  $\rho(\vec{v}_2)$  erhalten bleiben. Zusammenfassen von Tabellenspalten (Attribute).

Beide Verfahren sind sogenannte nicht-überwachte Verfahren. Die Ergebnisse der Abbildung hängen von der Verteilung der Daten ab und wird ohne Vorwissen ausgeführt. Beide Verfahren erhalten die Ähnlichkeit und berechnen meist auf numerischen Daten, aber Clustering liefert diskrete Ergebnisse und die Dimensionsreduktion liefert stetige Ergebnisse.

## Clustering

### Distanzmaße für Clusteringalgorithmen

- Manhattan-Distanz:  $d_1(a, b) = \sum_i |a_i - b_i|$
- Euklidische Distanz:  $d_2(a, b) = \sqrt{\sum_i (a_i - b_i)^2}$
- Maximum-Distanz:  $d_\infty(a, b) = \max_i (|a_i - b_i|)$

## Ähnlichkeit

Ähnlichkeit kann auch explizit für alle Punktepaare definiert werden, dabei erhält jedes Punktpaar einen Wert, so kann man eine Ähnlichkeitsmatrix aufstellen. Diese sind sehr frei definierbar und können leicht Vorwissen abbilden, flexibel modifiziert werden und die Daten brauchen keine Attribute. Aber die Ähnlichkeit muss keine Metrik sein und das Clustering kann möglicherweise vorgegeben sein.

## Mustererkennung

Muster sind die Grundlage für die Vorhersage und die Begriffsbildung/ Abstraktion.

Die **Vorhersage** ist die Fähigkeit, Gemeinsamkeiten zwischen vergangenen und zukünftigen möglichen Ereignissen herzustellen und unbekannte Eigenschaften aus Bekannten ableiten.

**Begriffsbildung** ist normalerweise implizit und eine Erweiterung/ Vereinfachung der Sprache. Außerdem hilft Begriffsbildung/ Abstraktion dabei Ordnung/ Strukturierung und Beziehungen/ Zusammenhänge festzustellen.

## Ähnlichkeit

Ähnlichkeit ist bei der Analyse nie eindeutig definiert. Es ist wichtig die passende Ähnlichkeitsmetrik zur Fragestellung zu finden. Mit Data-Mining kann man zwar automatisch Muster erkennen aber vor dem Data-Mining werden jedoch weitere automatische Verfahren auf die Daten angewendet. Alle Schritte der Analyse sind relevant für die Definition von Ähnlichkeit. Visuelle Analyse nicht nur bei dem eigentlichen Data-Mining, sondern bei allen Schritten, um sicherzustellen, dass relevante Informationen nicht verlorengehen und um mögliche Verfahren zu testen und zu verbessern. Je weniger Vorwissen für die Analyse gegeben ist, desto mehr Verfahren und Einstellungen müssen im Allgemeinen getestet werden. Die Integrationsvarianten sind nicht auf ein Verfahren beschränkt und die Verfahren sind nicht auf eine Integrationsvariante beschränkt.



Das Ziel ist eine Transformation in eine gute Repräsentierung der Daten ohne Verlust wichtiger Information.

Die Vorverarbeitung definiert Ähnlichkeit mit:

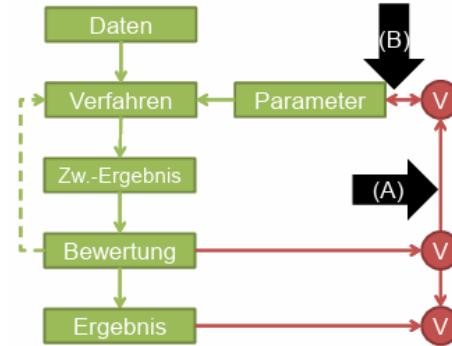
- Rauschentfernung macht Daten gleich, die vorher nicht gleich waren, Information über Ungleichheit geht ab hier verloren.
  - Log-Normalisierung verändert Verhältnisse zwischen Differenzen.

Ähnlichkeitsdefinition des Nutzers/Fachexperten ist „Ground Truth“. Sie ist die Grundlage für die Wahl von Verfahren/ Parameter entlang der Pipeline. Der Vergleich zwischen „Soll-Ist“ soll sichtbar werden.

# Black-Box Integration

Automatische Verfahren haben einige Schwächen, wie zum Beispiel, dass die Abhängigkeit der Ergebnisse von den Parametern häufig nicht vorhersehbar ist. Die Suche nach guten Parametern ist häufig ein Spiel mit Versuch und Irrtum, mehrere Parameter sind häufig voneinander abhängig. Die Optimierung nur eines Parameters genügt nicht.

Die Black-Box Integration hilft bei der Herstellung eines direkten visuellen Bezugs zwischen Parametern und Ergebnis oder Qualität. Die Parameterkonfigurationen werden interaktiv definiert. Das Verfahren wird ausgeführt und die verschiedene Parameterkonfiguration werden angezeigt (Ergebnisse oder Qualitätskriterien).



Bei der Black-Box Integration ist das Modell selbst nicht sichtbar und kann auf die Vorverarbeitung und DM-Verfahren angewendet werden. Die BBI zeigt, dass Visual mehrere Komponenten eines automatischen Verfahrens kann auch genutzt werden, um Werte zu verändern (B).

## Visualisierung in Clustering und Dimensionsreduktion

Durch Visualisierung kann ein Verfahren schrittweise überprüft werden: Geht entlang der Pipeline wichtige Information verloren? Sind, Merkmale, die hervorgehoben werden, brauchbar? Wie kann das Wissen des Fachexperten direkt in die Verfahren einfließen?

## Semi-Supervised Clustering

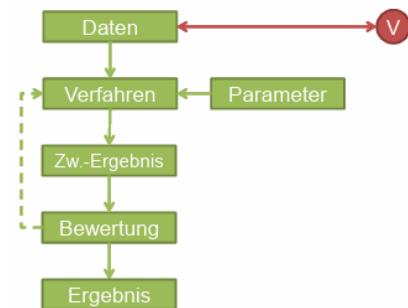
SSC ist ein Konzept aus dem Data-Mining und ist eine interaktive Modifikation der Qualitätsfunktion, bei der die Ähnlichkeit von Punktpaaren ( $dist(x_i, x_j)$ ) vorgegeben ist und die Distanz meistens Bestandteil der Qualitätsfunktion ist.

- Supervised: Alle Cluster sind bekannt
  - Partial Labelling: Clustering über Korrelation der Labeldichte und der Datendichte
  - Partial Constraints: bestimmter Anteil der Punktepaare definieren gleiche/ungleiche Cluster
  - Unsupervised: normales Clustering

## Visual Input Editing

Da Experten ihre Daten besser kennen als Softwareentwickler und dieses Wissen häufig schwer in das Verfahren integrierbar ist, verwendet man Visual Input Editing. Ohne Expertenwissen finden Verfahren nur einfache/bekannte oder unwichtige Zusammenhänge.

Beim Visual Input Editing bearbeitet man die Eingabedaten (z.B. Ähnlichkeit), dies ist sinnvoll, wenn die Daten bekanntes Wissen repräsentieren. Die Visualisierung muss die Anwendung des Wissens erleichtern und eine gewohnte Darstellung für die Experten bieten. Man kann dem Algorithmus auf drei Arten „zeigen“, wie er arbeiten soll: Urteilen, Ordnen, Sehen. Die Wahl der Visualisierung ist entscheidend, denn Daten müssen so dargestellt werden, dass Bewertung/ Interpretation/ Anordnung/ Vergleich durch Experten möglich ist. Die Wahl der Visualisierung ist entscheidet (Glyphen, DOI, Annotationen).



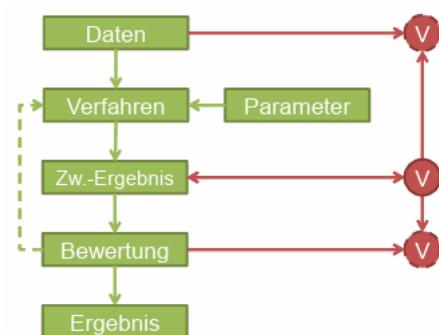
**Urteilen: Semi-Supervised Clustering:** Experte wird nach seinem Urteil bezüglich paarweiser Ähnlichkeit gefragt, nur wenige Paare werden dabei ausgewählt und dieses Verfahren funktioniert mit jeder vertrauten Visualisierung.

**Ordnen: Visuelles Ordnen:** Eine Abstandsfunktion bei nicht-numerischen Daten ist schwierig. Man verwendet die Sortierung/ Ordnung eines Experten, dieser ordnet Objekte per Drag & Drop an, der Abstand im Bild wird dann für die Abstandsfunktion im Clustering verwendet. Dieses Verfahren wird nur mit Samples gemacht und erfordert keinen Formalismus, der Algorithmus hat das Ziel, das Ordnungsschema des Experten zu reproduzieren.

**Sehen:** Die Ähnlichkeitsfunktion wird modifiziert durch die Trennung der Daten, die Distanz zwischen den Datenpunkten wird künstlich erhöht, aber dies kann zu Überanpassung führen.

## White-Box Integration

Eine automatische Optimierung ist schwierig und meist nur lokal optimal. Normalerweise erzeugt das Verfahren neue Ergebnisse automatisch. Bei der White-Box Integration werden (Zwischen-)Ergebnisse und Optimierungsschritte visualisiert und der Nutzer darf in die Optimierung eingreifen. White-Box bedeutet, dass man sieht, wie das Verfahren arbeitet, aber nur wenige Implementierungen erlauben eine detaillierte White-Box-Analyse.



## Model-Data-Linking

Das Ergebnis und die Daten sind eigentlich zwei unterschiedliche Repräsentierungen des gleichen Sachverhalts. Das Ergebnis ist unter anderem dann plausibel, wenn er Bezug zwischen Ergebnis und Daten hergestellt werden kann.

Model-Data-Linking stellt einen visuellen Bezug zwischen Daten und Ergebnis her. Die Voraussetzung hierfür ist, dass das Ergebnis nicht überall von allen Daten beeinflusst wird. Die Daten sind lokalisierbar im Modell und die Modellteile lokalisierbar in den Daten.

Model-Data-Linking nutzt Datenvisualisierung und Modellvisualisierung und verknüpft die Visualisierung der Daten durch räumliche Nähe oder Interaktion.

## Hierarchisches Clustering

Beim Clustering gibt es häufig mehrere gute Partitionen (je nach Verfahren, welches man wählt, k-Means oder DBScan). Generell gilt selten, dass die Datenverteilung überall nach dem gleichen Schema gilt. Die Idee von Hierarchischem Clustering ist das Berechnen der Cluster auf verschiedenen hierarchischen geordneten Detailstufen. Hierfür gibt es zwei Strategien.

### Partitionierendes hierarchisches Clustering

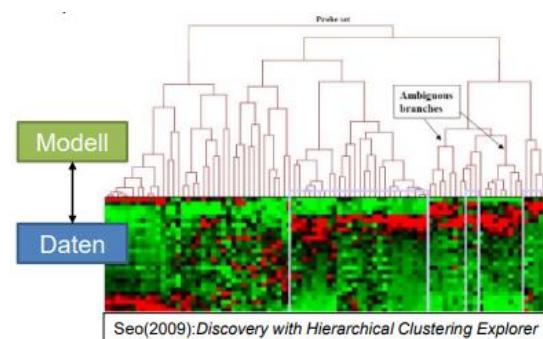
Man startet mit einem Cluster, welcher alle Datenobjekte enthält und unterteilt diesen rekursiv. Man sucht den jeweils schlechtesten Cluster und unterteilt diesen mit einem beliebigen Clustering Verfahren. So erhält man eine Hierarchie von Clustern, diese Hierarchie ist Teil des Modells.

### Agglomerates hierarchisches Clustering

Man startet mit jedem Datenobjekt als eigener Cluster und führt diese iterativ zusammen. Man sucht jeweils ein Paar, das am ähnlichsten ist, dieses Paar fügt man zu einem neuen Cluster zusammen. Die Hierarchie dieses Verfahrens ist immer binär.

### Dendrogrammvisualisierung

Ein Dendrogramm zeigt die Reihenfolge, in der Cluster zusammengefasst werden und zeigt den Informationsgewinn durch die Clustering Höhe. Es zeigt auch die Ähnlichkeit der Cluster. Die Anordnung der Hierarchie ist NP-hart.



### Pattern Exploration

Die Pattern Exploration kombiniert die Visualisierung von Modellen, Mustern und Daten. Korrespondenz durch identische visuelle Attribute (gleiche Position).

### DBScan

Beim density based Scan wachsen die Cluster entlang der Nachbarschaft. Man beginnt bei einem zufälligen Punkt, der noch keinen Cluster hat, und bestimmt die Nachbarn. Hat ein Punkt viele Nachbarn, genau dann liegt dieser Punkt dicht, Punkt und Nachbarn gehören zum gleichen Cluster. Hat ein Punkt wenig Nachbarn, aber wenigstens ein Nachbar liegt dicht, dann gehört der Punkt zu dessen Cluster (bei Mehrdeutigkeit zufällig gewählt). Hat ein Punkt keine dichten Nachbarn, dann ist er ein Ausreißer. Dann die nächste Iteration bei unbearbeiteten Nachbarn. Der DBScan liefert nur die Zuordnung der ClusterID und kein Modell. Die Parameter sind die Nachbarschaftsgröße und die minimale Anzahl einer dichten Nachbarschaft.

### Dimensionsreduktion

Die Beiden verfahren PCA und LDA wurden bereits vorgestellt aber werden hier nochmal vertieft.

Bei der Projektion auf zwei Dimensionen definiert man die Projektionsachsen  $\vec{a}$  und  $\vec{b}$ ,  $\vec{v} \rightarrow p(\vec{v}) = (\vec{a}\vec{v}, \vec{b}\vec{v})$ . Das Ergebnis sind die beiden Koordinaten.  $\vec{v}$  sind die Daten,  $\vec{a}$  und  $\vec{b}$

beschreiben das Modell bzw. das Zwischenergebnis. Und  $\rho(\vec{v})$  ist das Ergebnis. Wichtig ist, dass die Attribute des Datenobjekts numerisch sind.

Auch nicht-lineare Verfahren MDS und SOM wurden bereits vorgestellt und werden nochmal vertieft. Häufig entstehen bei Clustern komplexe Strukturen wie langgezogene Cluster. Beim Anwenden der White-Box Interaktion legt man Startpunkte für die Optimierung fest und die Karte ordnet sich um die vorgegebene Ordnung an. Es ist schwer zu erkennen, wenn eine Projektion aus einem hochdimensionalen Raum nicht funktioniert.

## Übersicht

Unterscheidung zwischen normaler Interaktion, die innerhalb der Visualisierung, die Vis-Pipeline (Data Transformation, Visual Mapping, View Transformation) ändert und Interaktion, die auch außerhalb der Visualisierung in andere Methoden eingreift.

Die Visualisierung ist idealerweise ein unabhängiges Stück Software, die Verbindung braucht eine ordentliche API, damit sie Analytics-Werbeversprechen von substanzialen Möglichkeiten für die visuelle Analyse besser unterscheiden können.

Siehe Übersicht – Integrationsvarianten

## Von Mustern zu Modellen

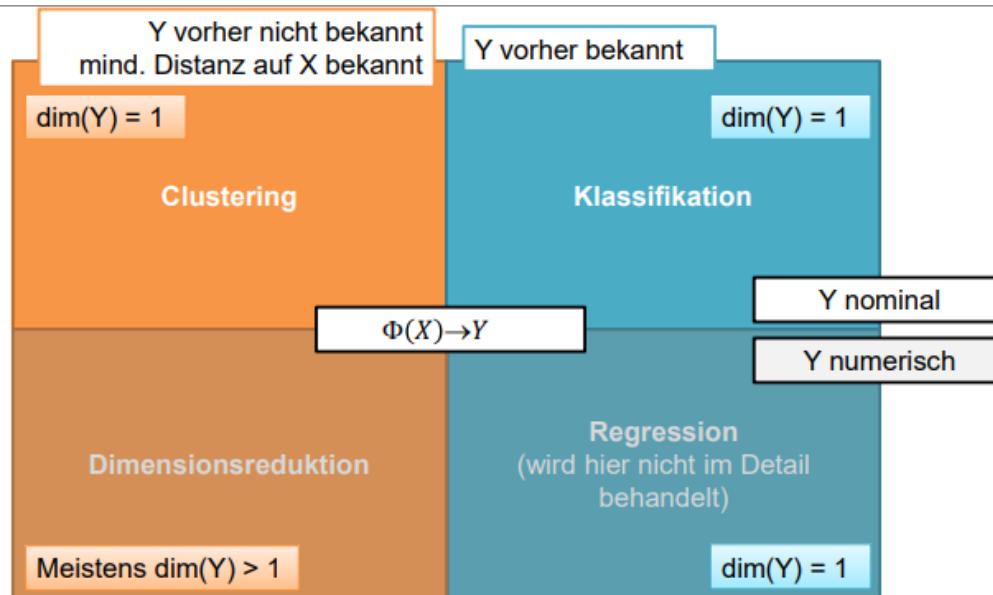
### Data-Mining Aufgaben

Clustering und Klassifikation lernen eine Funktion  $\Phi(X) \rightarrow Y$ .

Methoden des supervised Learning haben Datenobjekte  $d$ , die Informationen über  $X$  und  $Y$  enthalten. Bei der Klassifikation sind die Datenobjekte  $d \in X \times Y$  gegeben und  $Y$  ist nominal. Bei der Regression sind auch Datenobjekte  $d \in X \times Y$  gegeben, aber  $Y$  ist numerisch.  $\Phi$  ist häufig durch ein Modell repräsentiert.

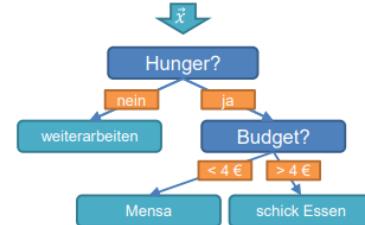
#### Qualitätskriterien:

- **Getreue Wiedergabe** der gewünschten Abbildung  $\Phi(X) \rightarrow Y$ , möglichst wenige Datenobjekte  $x$  sollen auf falschen Wert  $y$  abgebildet werden und Fehler sollten nicht systematisch sein.
- **Verallgemeinerbarkeit**, Klassifikationsmodell soll auch bei neuen/unbekannten Daten korrekte Ergebnisse liefern
- **Geringe Komplexität**, (Ockham's Razor) kurze, einfache Beschreibung und häufig notwendig für Verallgemeinerbarkeit, steht im Konflikt mit getreuer Wiedergabe aber Modell soll vom Menschen verstehbar sein.



## Entscheidungsbäume

Entscheidungsbäume sind ein hierarchisches Klassifikationsmodell. Die inneren Knoten repräsentieren Entscheidungen, meist anhand eines Attributs von x und Kanten repräsentieren Optionen. Die Klassifikation eines Datenobjekts x durchläuft genau einen Pfad zu einem Blattknoten. Dieser Blattknoten definiert das Ergebnis y des Klassifikators.



Entscheidungsbäume haben eine geringe Komplexität, mit einer geringen Tiefe lassen sich viele Fälle durch wenige Entscheidungen abdecken. Jede Entscheidung sollte Klassen möglichst gut trennen und im Baum nach Wichtigkeit geordnet sein.

Die Anzahl der Teilbäume/ Optionen pro Knoten ist variabel, pro Knoten wird nur ein Attribut abgedeckt und die Partitionierung des Attributs ist nicht immer notwendig. Das Bewertungskriterium für die Partitionierung ist die Trenngenaugkeit der Klassen. Das Abbruchkriterium ist die Genauigkeit des Baumes.

**Verfahren:** Gegeben ist eine Datentabelle mit  $i=1\dots n$  Attributen und  $m$  Datenobjekten. Alle Datenobjekte haben Labels Y.

- Der Baum wird rekursiv aufgebaut
- Jeder Knoten/ Teilbaum repräsentiert eine Teilmenge der Daten
- Für jeden Knoten berechne für jedes Attribut  $X_i$  die optimale Unterteilung der Teilmenge bezüglich der Klassen Y und wähle das Attribut mit der besten optimalen Unterteilung

**Visuelle Verfahren:** Visualisierung möglicher Splitpunkte über den Wertebereich eines Attributes. Für eine White-Box-Integration eine interaktive Definition von Splitpoints und Teilintervalle möglichst in unterschiedlichen Klassen. Für nominale Attribute ist die Berechnung von Splitpoints schwierig.

**Optimierung:** Der schwierigste Fall sind nominale Attribute, weil bei denen die Sortierung und die Ordnung der Werte egal ist, müssen alle Splitpoints und Umordnungen getestet werden. Datensätze mit gleichen kategorischen Werten gehören nicht zu einer Klasse, und es häufig

mehr als nur zwei Klassen gibt, muss der optimale Splitpoint auf der optimalen Umsortierung der Attributwerte auf dem optimalen Attribut gefunden werden. Bei ordinalen Daten entfällt die Umsortierung.

## Qualitätsmaße

Die Gruppen sollten homogen sein. Homogenität kann mit dem **Gini-Index** für jede Gruppe berechnet werden. Dabei beschreibt  $p_1$  den Anteil von  $y_1$  an einer Gruppe und  $p_2$  den Anteil von  $y_2$  an einer anderen Gruppe.

	<b>G<sub>1</sub></b>	<b>G<sub>2</sub></b>
<b>y<sub>1</sub></b>	0,4	0,9
<b>y<sub>2</sub></b>	0,6	0,1
Gini	$1 - (0,4^2 + 0,6^2) = 0,48$	$1 - (0,1^2 + 0,9^2) = 0,18$

G<sub>1</sub> und G<sub>2</sub> sind abhängig von der zu optimierenden Einteilung.

$$Gini(G) = 1 - \sum_i p_i^2$$

Der Index ist 0, wenn die Gruppe homogen ist und maximal, wenn die Klassen gleichverteilt sind. Die Qualität der Gruppierung ist bestimmt durch die Homogenität aller Gruppen.

$$Gini_{Ges} = w_1 Gini(G_1) + w_2 Gini(G_2)$$

Mit  $w_i$  der relativen Größe von  $G_i$ .

Ein weiteres Verfahren ist die **Entropie**

$$Entropy = - \sum_i p_i \log_2 p_i$$

Der Index ist 0, wenn die Gruppe homogen ist und maximal, wenn die Klassen gleichverteilt sind.

Die Homogenität der Klassen nach der Unterteilung ist aber nicht allein entscheidend, die Klassen könnten ja auch vor der Unterteilung schon ungleich verteilt sein.

Information Gain mit Entropy vorher und nacher:

$$IG = -Entropy_{before}(G) + \sum_i \frac{|G_i|}{|G|} Entropy(G_i)$$

## Integrationsvarianten

### Overfitting

Overfitting ist das Überanpassen an die Trainingsdaten. Die Trainingsdaten sind der Teil der Datensätze, mit denen das Modell berechnet wird. Die Testdaten sind der Teil der Datensätze, mit denen das Modell getestet wird. Die Trainings- und Testdaten werden vor der Modellierung getrennt.

Mit beliebig vielen Knoten kann der Entscheidungsbaum fast beliebig präzise werden und auf den gelernten Daten eine Genauigkeit von 100% erreichen. Der Baum ist aber nicht verallgemeinerbar, weil die Trainingsdaten eventuell nicht repräsentativ sind. Dies ist ein generelles Problem bei Klassifikationsverfahren.

Mit Pruning-Strategien lässt sich der Entscheidungsbaum vereinfachen, durch z.B. Abschneiden von Teilbäumen, bei denen der Information Gain zu gering ist.

Eine andere Strategie ist das Berechnen von mehreren Modellen auf unterschiedlichen Trainingsdaten, dann werden die Ergebnisse durch z.B. Voting gemittelt /Kreuzvalidation, Random Forest).

Der Grund für Overfitting ist, dass Rauschen im Datensatz mitmodelliert wird, das Verfahren kann zwischen Rauschen und Muster nicht unterscheiden. Der Mensch kann hier Rauschen und Muster manuell trennen, um Overfitting zu vermeiden.

## Visual Input Editing

Um Muster und Rauschen oder verschiedene Muster zu trennen, verwendet man Visual Input Editing. Dabei werden die Muster durch den Menschen markiert und die Labels werden durch neue ersetzt.

Man hat also eine Datenvisualisierung, welche die aktuellen existierenden Klassen zeigt. Der Nutzer verändert die Klassenlabel an den Originaldaten.

Das Label  $Y$  wird durch das Label  $\bar{Y}$  ersetzt, dabei beschreibt  $\bar{y}_1$  die vom Nutzer aktuell ausgewählten Daten und  $\bar{y}_2$  die nicht vom Nutzer ausgewählten Daten. Das neue Label kann durch Brushing innerhalb der Visualisierung definiert werden, durch nachzeichnen des Musters mit der Maus. Brushing ist unabhängig von der Visualisierung und dem Klassifikationsmodell. Das Modell wird dann auf den neuen Labeln berechnet, so wird das Rauschen ignoriert.

## Model Data Interaktion

Verfahren können nicht beliebige Muster gut modellieren. Es ist nicht sichtbar, ob das wahrgenommene /markierte Muster tatsächlich auch so modelliert werden. Eine Lösung ist die Model Data Interaktion, welche eine Erweiterung des Model-Data-Linking darstellt. Das Modell liefert Feedback über das erzeugte Muster. Das erzeugte und das originale Muster können visuell verglichen werden.

Der Anwender kann Muster selektieren. Aus den Mustern wird automatisch ein Modell erstellt, welches die Regel beschreibt, der die Selektion folgt. Im Feedback wird das Modell wieder auf die Originaldaten angewandt (man erhält neue Klassenlabel Muster und Nicht-Muster). Das modellierte Muster wird als Feedback gezeigt, optional kann das Feedback Daten ergänzen, die wahrscheinlich zum Muster gehören.

## Visual Model Verification

Die Bewertung kann sich nicht selbst bewerten und das Bewertungsschema ist für alle Daten und Klassen gleich. Systematische Fehler können so nicht erkannt werden und da alle Fehler gleich gewichtet sind, können sie schlecht lokalisiert werden.

Die Visual Model Verification bewertet die Klassifikation und die Klassifikationsfehler. Die entscheidende Kenngröße für Verifikation ist der Unterschied zwischen den modellierten und den tatsächlichen Daten  $diff(\Phi(X), Y)$ .

Beim Data-Mining werden alle Fehler zu einem Wert zusammengefasst, was ein gutes, allgemeines Qualitätskriterium ist aber schlecht für die Suche von Verbesserungsmöglichkeiten. Eine bessere Darstellung ist die **Confusion Matrix**, welche die modellierten mit den vorgegebenen Klassen vergleicht.

Die Confusion Matrix liefert Informationen darüber welche Klassen gut und welche weniger gut Unterschieden werden, zusätzlich können die Fehlklassifikationen mit unterschiedlichen Kosten verbunden werden.

Es gibt auch Verfahren, mit denen die Bewertung editiert wird. Die Kosten für Fehlklassifikation können durch Gewichte in der Confusion Matrix angegeben werden. Diese Gewichte können bei der Optimierung der Klassifikation einfließen. Die Optimierung vermeidet Fehlklassifikation mit hohen Kosten.

Die Differenz kann nominal, ordinal oder numerisch sein, `diff()` lässt sich nicht immer arithmetisch berechnen aber diese Differenz kann jedem Datenobjekt zugeordnet werden. Dabei verwendet man die Differenz als neues Attribut der Daten, das Attribut kann wie ein normales Datenattribut behandelt werden.

**Residual-Plots** stellen die Abweichung zwischen Prognose und Erwartung dar. Fehler werden lokal im Datensatz verortet und haben einen Fehlerwert pro Kategorie. Oft kann die Prognose nur für eine Klasse gut dargestellt werden.

Wenn die Klassifikationsfehler selbst ein Muster bilden, handelt es sich um einen systematischen Fehler.

Die Visual Model Verification kann bei der Klassifikation und beim Clustering eingesetzt werden. Das Ziel ist eine detaillierte Visualisierung der Klassifikationsfehler für die Diagnose. Eine Darstellung der Klassifikationsfehler abhängig von den Klassen Y, abhängig von den Daten X und innerhalb des Modells. Die meisten automatischen Verfahren nutzen Qualitätsmetriken, welche häufig Summen über Datenelemente, Klassen oder Elementpaare sind. Diese Summen können zerlegt werden, um detailliertere Informationen über die Fehler zu erzeugen.

## Klassifikation ohne Entscheidungsbäume

### Support Vector Machines

Bei Entscheidungsbäumen sind die Grenzen zwischen den Klassen immer achsenparallel, häufig gibt es Treppen und das Modell wirkt komplex.

Support Vector Machines suchen Ebenen die zwei Klassen trennt mit einem möglichst großen Abstand. Die Ebenengleichung (Hesse Form, Normalenvektor  $\vec{w}$ ) lautet:

$$y_i = \operatorname{sgn}(\vec{w}\vec{x} + \vec{b})$$

Es sind aber nicht alle Klassen durch eine Ebene trennbar. Der **Kernel-Trick** erhöht die Anzahl der Dimensionen, so ist die Punktmenge im hochdimensionalen Raum trennbar. Hierfür muss man eine grobe Vorstellung haben, wie die Klassen aussehen.

$$(x_1, x_2) \rightarrow \left( x_1, x_2, \frac{x_1^2}{x_2} \right)$$

Der Kernel-Trick ist auch bei anderen Verfahren sinnvoll, bei z.B. vermuteter nicht-linearer Abhängigkeit zwischen zwei oder mehr Variablen.

### K-Nearest-Neighbors

K-Nearest Neighbors ist ein Sample basiertes Klassifikationsverfahren. Samples sind Datenobjekte mit gegebenen Klassen, der Rest sind Datenobjekte ohne Klassen. 1-Nearest

Neigbor entspricht einem Schritt des k-Means verfahren, es wird aber eine Distanzfunktion benötigt. Wenn die Klassen Y für wenige Datenobjekte bekannt sind, nimmt man an, dass die Objekte in der Nähe wahrscheinlich ähnlich klassifiziert werden sollten. Man sucht die k nächsten Nachbarn, deren Klasse Y bekannt ist. Die Klasse wird z.B. durch Voting gewählt.

Cave-At 1.:

k-Means : Clusteringverfahren

k-Nearest-Neighbor : Klassifikationsverfahren

Cave-At 2.: „k“ ≠ „K“

das „k“ in „k-Means“ ist die Gesamtanzahl der Cluster.

das „k“ in „k-Nearest-Neighbor“ ist die Zahl der Nachbarn für das Voting

...nur falls im März oder so noch mal jemand danach fragt...

## Vergleich

Entscheidungsbäume	Support-Vector Machines	K-Nearest Neighbor
<ul style="list-style-type: none"> <li>Modellbasiert</li> <li>Verknüpfung einfacher Klassifikatoren zu einer Hierarchie</li> <li>Leicht interpretierbar</li> </ul>	<ul style="list-style-type: none"> <li>Modellbasiert</li> <li>Algebraisches/ Numerisches Verfahren, erfordert numerifizierte Daten</li> <li>Gute Trennung komplexer Klassen</li> </ul>	<ul style="list-style-type: none"> <li>Sample-basiert</li> <li>Sinnvoll, wenn Y nur für wenige Samples bekannt ist</li> <li>Braucht Abstände</li> </ul>

## Bayes Klassifikation

Bei der Bayes Klassifikation wendet man das Bayes-Theorem für bedingte Wahrscheinlichkeit an. Es wird die Klasse mit der größten Wahrscheinlichkeit gewählt.

Der Naive-Bayes betrachtet alle Attribute als unabhängig, die Einzelwahrscheinlichkeiten werden multipliziert.

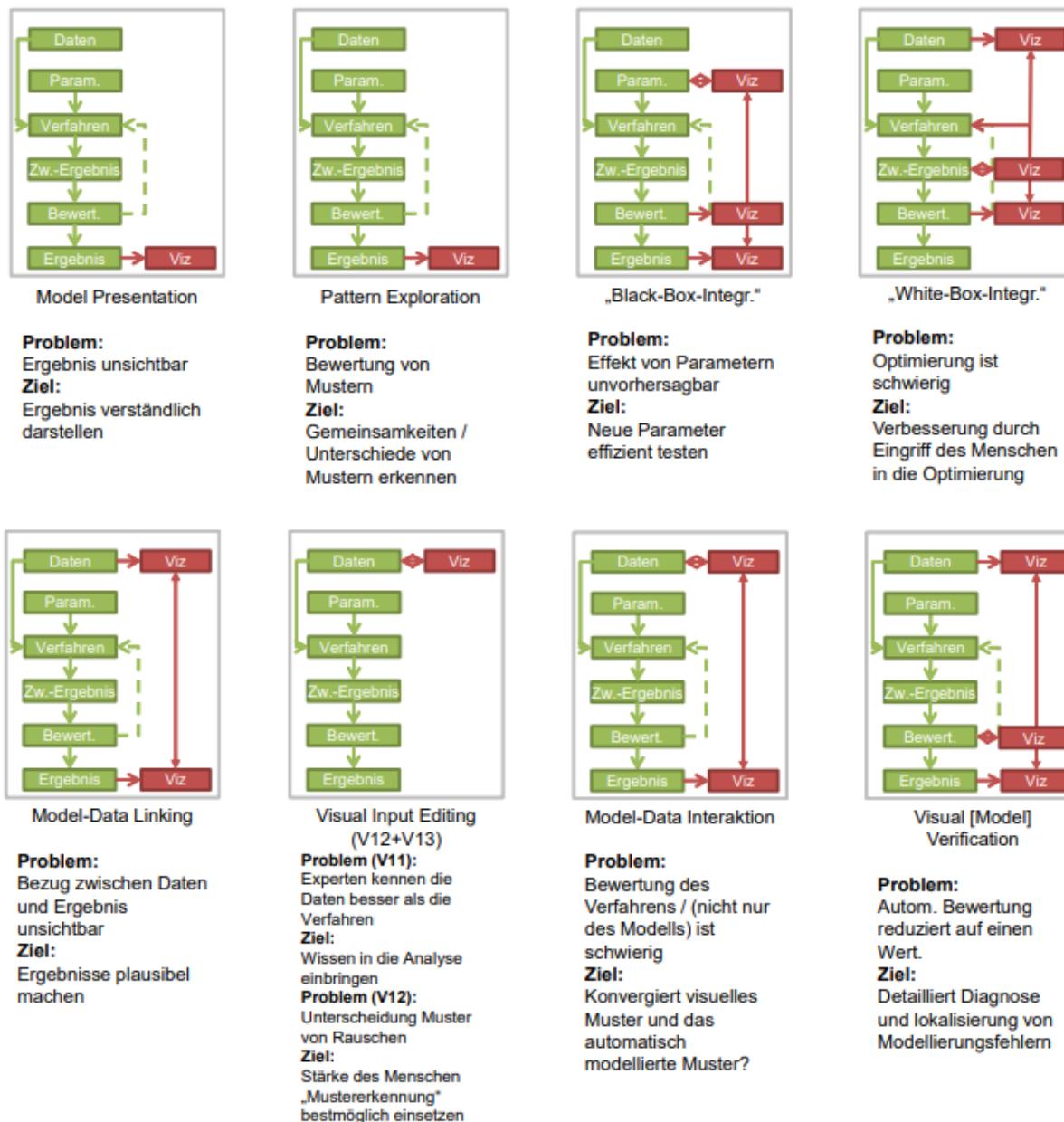
Die Bayes-Klassifikation ist nicht heuristisch, es muss kein guter Klassifikator gesucht werden, die Klassifikationsregel kann direkt berechnet werden. Eine White-Box Integration ist nicht möglich.

Eine weitere Form sind die Markov (Bayessche) Netze, die eine Verallgemeinerung der Bayes-Klassifikatoren darstellen. Die Abhängigkeiten werden als Graph dargestellt, dieser ist leicht verständlich aber das Erstellen von Modellen ist NP-schwer.

## Künstliche Neuronale Netze

Künstliche Neuronale Netze sind eine Verbindung einfacher Schaltelemente zu Netzwerken, die Topologie ist meist vorgegeben und die Eingabeschicht repräsentiert X und die Ausgabeschicht repräsentiert Y. Das Netzwerk lernt durch Änderung der Gewichte an den Kanten, so können komplexe Probleme approximiert werden aber das Modell ist nicht lesbar. White-Box Ansätze sind sehr anspruchsvoll.

## Übersicht – Integrationsvarianten



## Querverbindung

Gemeinsame Grundlage, statistische Maße für Abhängigkeit:

- Feature Subset Selektion und Entscheidungsbäume.

Gemeinsame Zielsetzung, reduktion der Komplexität:

- Dimension Reduktion und Feature Subset Selektion

Ähnliche Verfahren:

- Self-Organising Map, K-Means und k-Nearest Neigbor
- Dimensions Reduktion und Graph Layout Algorithmen

Iterative Optimierungstechniken: Alle Data-Mining Techniken

## Direkte/ Indirekte Kopplung

Die Integrationsstrategien sind ein Designmuster und keine Baupläne. Viele Verfahren enthalten mehr als eine dieser Strategien, manche Strategien überspannen mehrere Verfahren.

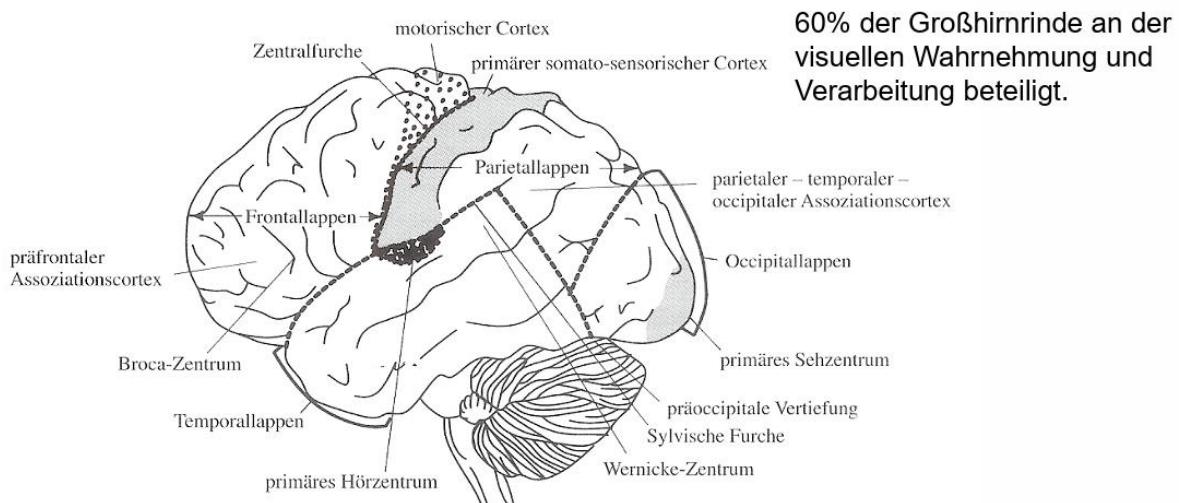
Beim Visual Analytics Process Modell werden die Informationsvisualisierung und die Knowledge Discovery gemeinsam eingesetzt und beschreibt die direkte Kopplung zwischen diesen beiden Technologien.

Jed Integrationsstrategie lässt sich mit direkter und indirekter Kopplung implementieren. Kopplung bedeutet, dass zwischen zwei Techniken ein Datenfluss oder -austausch stattfindet. Bei der indirekten Kopplung muss der Nutzer bei dieser Kopplung helfen. Im Idealfall muss der Nutzer nur mit der Visualisierung interagieren, der Datenfluss findet ohne menschliches Zutun sein Ziel.

## Kognitive Psychologie

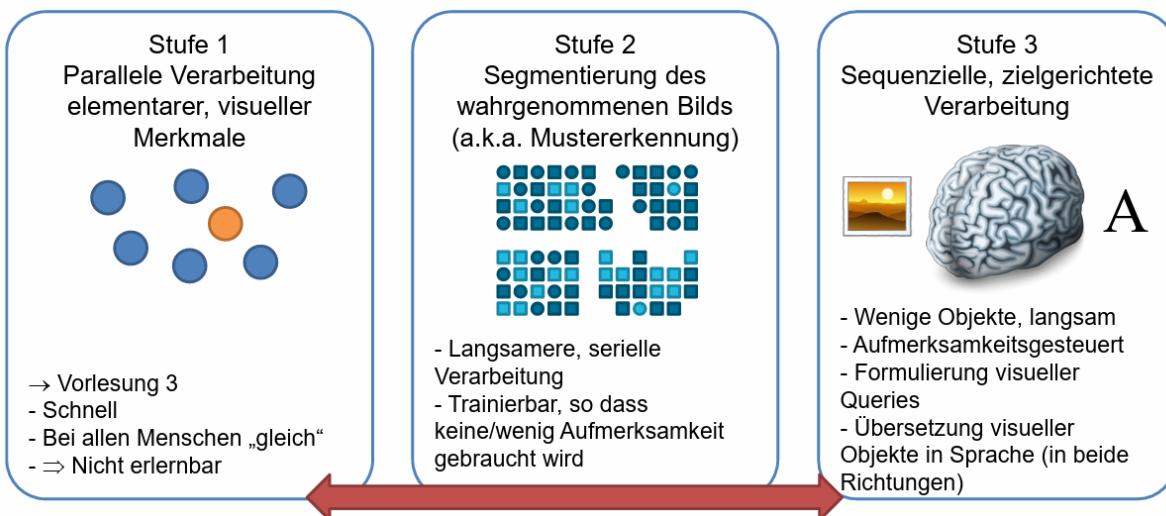
Große Intelligenzleistungen beruhen auf elementaren kognitiven Prozessen, die auf komplexe Weise zusammenwirken (Herbert Simon).

Das Ziel der kognitiven Psychologie ist das Verstehen von Funktionsweise, Stärken und Schwächen des menschlichen Nervensystems. Das Gehirn besteht aus einer Anzahl abgegrenzter Bereiche, die unterschiedlichen kognitiven Funktionen dienen.



**Abb. 1.6** Die seitliche Ansicht des cerebralen Cortex und seine wichtigsten Bestandteile (nach Kandel & Schwartz, 1984. Abdruck mit Genehmigung des Verlags. Copyright © 1984 by Elsevier Science Publishing Co., Inc.).

## Wahrnehmungsmodell nach Ware



## Aufmerksamkeit

Aufmerksamkeit ist die Selektivität der Wahrnehmung und hat drei **Aufgaben**:

- Planen/Kontrollieren (sich auf eine Handlung konzentrieren, willentlich/endogen)
- Überwachen (der eigenen Umwelt, automatisch/exogen)
- Selektieren (Trennung von relevanten und irrelevanten Informationen, ~exogen)

Aufmerksamkeit hat keinen uneingeschränkten **Parallelismus**:

- Parallel Wahrnehmung von Level 1-Informationen (nach Ware)
- Parallel Tätigkeiten mehrerer motorischer Systeme (Gehen, Mund bewegen)
- Schwieriger: Zwei Dinge mit einem motorischen System (nur ein System, um zwei Hände zu bewegen)
- Serieller Flaschenhals bei Informationsverarbeitung (unabhängig von Motorik; z.B. gleichzeitig zwei Zahlen addieren und multiplizieren)

Bandbreite der Sensorik:

1. Sehen: 1250 MB/s
2. Fühlen: 125 MB/s
3. Hören/Riechen: 12.5 MB/s
4. Schmecken

Aufmerksamkeit ist eine temporäre Verbindung verschiedener Wahrnehmungssysteme, motorischer Systeme und höheren, kognitiven Systemen (Gedächtnis, Planung, ...), selektierend für die jeweilige Anforderung der Informationsverarbeitung.

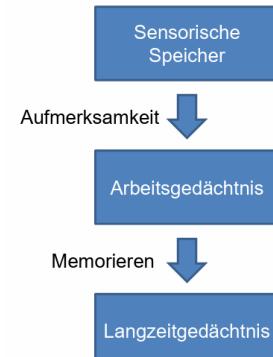
Die meisten Systeme sind jeweils für sich unabhängig, verschiedene Systeme können parallel an verschiedenen Aufgaben arbeiten, aber Systeme arbeiten für sich seriell, was zu Interferenz zwischen verschiedenen Aufgaben führen kann.

Die **visuelle Aufmerksamkeit** nach dem Wahrnehmungsmodell von Ware kann in Dorsale (willentliche) und ventrale (automatische) Wahrnehmung unterschieden werden. Die Aufmerksamkeit kann durch visuelle (und auditive) Hinweise gelenkt werden. Die Aufmerksamkeit beeinflusst welche Inhalte wahrgenommen werden. Reduktion des Suchaufwands, wenn Fokus des Nutzers bekannt.

Die **kognitive Aufmerksamkeit** ist das Denken als serieller Prozess, bei dem gegeben falls alternativen außerhalb des aktuellen Fokus ausgeblendet werden.

## Gedächtnis

Das Gedächtnis ist zentral für das Verständnis des Menschen und eine interne Repräsentierung. Das Gedächtnis ist aus dem sensorischen Speicher, dem Arbeitsgedächtnis und dem Langzeitgedächtnis aufgebaut. Die Prozesse im Gedächtnis sind die Lernphase (Enkodierphase), die Behaltensphase (Retentionsphase) und die Abrupphase (Testphase).



## Wissensrepräsentierung

Die Wissensrepräsentierung kann in **einen klassischen, ähnlichkeits- und theoriebasierten Ansatz** unterschieden werden.

Der **ähnlichkeitsbasierte Ansatz** hat unscharfe Grenzen von Kategorien und ist unterschieden in den Prototypansatz und den Exemplaransatz.

Der **theoriebasierte Ansatz** unterteilt in Kategorien und Attribute und stellt Beziehungen zwischen diesen her.

In Visual Analytics wird das Wissen des Nutzers betrachtet, dieses kann entweder **deklarativ** oder **nicht-deklarative** eingebunden werden.

**Semantisches Gedächtnis** (deklarativ): Anlehnung von visuellen Elementen, Metaphern etc. an die Erfahrungswelt der Benutzer.

**Episodisches Gedächtnis** (deklarativ): Fähigkeit zur Erzeugung mentaler Landkarten des Menschen sehr ausgeprägt. Diese Fähigkeit wird auf den Visualisierungsraum übertragen.

**Prozedurales Gedächtnis** (nicht-deklarativ): Anlehnung an existierende Bewegungsmuster -> Interaktion

**Perzeptuelles Wissen** (nicht-deklarativ): Schnellere Verarbeitung von Objekten, die schon einmal verarbeitet wurden -> konsistente Benutzerschnittstellen.

## Denken, Entscheiden, Urteilen

Logisches Denken ist ein zentraler Aspekt früher kognitiver Forschung.

Menschliches Denken ist nicht logisch im mathematischen Sinne. Folgende Ansätze gibt es:

- Deduktives Schließen (Vom Allgemeinen auf etwas Konkretes schließen)
  - Modus ponens (Wenn A, dann folgt B)
  - Modus tollens (Wenn nicht A dann folgt, nicht B)
- Abduktives Schließen
- Probabilistisch (A gilt daraus folgt, B wahrscheinlich)
- Erlaubnisbezogen (Wenn A, dann sollte B gelten)
- Naturalistic Decision Making (situative Entscheidungsfindung)

## Abduktives Schließen

Beispiel: Wenn ein Einbrecher im Haus ist, dann steht die Tür offen. Die Tür steht offen. Es folgt, dass ein Einbrecher im Haus ist.

- A-priori Wahrscheinlichkeit:  $P(H)=0,001$  (Hypothese H: "Einbruch")
- Bedingte Wahrscheinlichkeit:  $P(E|H) = 0,80$  bzw.  $P(E|\neg H) = 0,01$  (Ereignis E: "Tür steht offen")
- A-posteriori Wahrscheinlichkeit:  $P(H|E)$  ("Tür steht offen, also ist Einbrecher im Haus") = 0,74

$$P(H|E) = \frac{P(E|H) \cdot P(H)}{P(E|H) \cdot P(H) + P(E|\neg H) \cdot P(\neg H)}$$

Abduktives Schließen sucht die plausibelste Erklärung für beobachtete Phänomene und wird oft als Vermutungslogik betrachtet. Das Bayes-Theorem, ein normatives Modell, unterstützt abduktives Schließen, indem es die Revision von Wahrscheinlichkeiten auf Basis neuer Evidenz ermöglicht. Es berechnet die a-posteriori-Wahrscheinlichkeit einer Hypothese und kann zu Schlüssen führen, die von intuitiven Einschätzungen abweichen. Die Anwendung des Bayes-Theorems erfordert eine sorgfältige Berücksichtigung der a-priori Wahrscheinlichkeiten. Diese Methode kann jedoch komplex sein und steht oft im Kontrast zur menschlichen Neigung, Wahrscheinlichkeiten intuitiv und möglicherweise ungenau zu bewerten.

## Entscheidungslehre

Die Entscheidungslehre ist ein interdisziplinäres Thema und wird in drei Bereiche unterteilt:

- Normativ (volle, rationale Verantwortung bei den Entscheidenden)
- Deskriptiv (eingeschränkte Rationalität, Beobachtung realer Entscheidungen)
- Konstruktiv (ständige Anpassung von Heuristiken während des Entscheidungsprozesses)

Die Entscheidungsaufgaben bestehen aus:

- Entscheidungsaktivität (Auswahl/Selektion bzw. Finden von Alternativen)
- Alternativen (Anzahl, vorgegeben vs. unbekannt)
- Attribute (Eigenschaften der Alternativen, Kriterien)
- Gütemaße (subjektiver Eindruck, Effektivität, Domänenkriterien wie z.B. Wert des Portfolios)

Menschliche Fehler im Schlussfolgern entstehen oft durch Übersehen alternativer Prämissen; visuelle Hilfen können logische Fehler mindern und das Verständnis für Rahmungseffekte verbessern. Die Forschung zeigt, dass der Fokus der visuellen Analytik mehr auf analytischen als auf typischen Entscheidungsaufgaben liegt, wobei besseres Datenverständnis nicht immer zu besseren Entscheidungen führt. Eine visuelle Formulierung von Entscheidungsaufgaben könnte Abhilfe schaffen.

**Rahmungseffekte** (engl. "framing effects") beziehen sich auf die Veränderung von Entscheidungen oder Meinungen der Menschen, die dadurch entsteht, wie Informationen oder Optionen präsentiert werden. Diese Effekte zeigen, dass die Wahl der Worte, der Kontext und die Darstellung von Informationen einen signifikanten Einfluss darauf haben können, wie Menschen Informationen interpretieren und welche Entscheidungen sie treffen.

## Problemlösen

Problemlösen beginnt mit dem Verständnis der Differenz zwischen dem aktuellen Zustand und dem gewünschten Ziel, wobei das Denken als Simulation innerhalb eines mentalen Modells fungiert. Bei einfachen Problemen sind Anfangs- und Zielzustände klar, während bei komplexen Problemen der Zielzustand oft unbestimmt ist und bei sehr komplexen Problemen sogar die notwendigen Schritte (Operatoren) unbekannt sein können. Die Suche und Anwendung geeigneter Operatoren zur Überbrückung dieser Differenz ist ein Kernprozess des Problemlösens.

Visual Analytics unterstützt dieses Vorgehen, insbesondere bei explorativer Analyse und unklaren Zielzuständen, indem es das prozedurale Gedächtnis durch visuell-interaktive Operatoren, die kognitive Aktivitäten widerspiegeln, fördert. Dadurch wird die Identifikation von Lösungswegen in großen und unübersichtlichen Problemräumen erleichtert.

## Expertenfähigkeiten

Experten unterscheiden sich von Novizen durch ihre Fähigkeit, komplexe Probleme durch umfangreiches Üben und den Erwerb hochgradiger Kompetenzen zu lösen. Der Erwerb von Expertenfähigkeiten vollzieht sich in drei Phasen:

- Kognitive Phase: Erwerb von deklarativem Wissen und Verständnis grundlegender Konzepte.
- Assoziative Phase: Verknüpfung von Wissenselementen und Bildung von Produktionsregeln, die zum prozeduralen Wissen führen.
- Autonome Phase: Prozesse und Fertigkeiten werden zunehmend automatisiert und effizienter, erfordern weniger bewusste Aufmerksamkeit.

Experten zeichnen sich durch verbesserte Mustererkennung und schnelleren Zugriff auf relevantes Wissen im Langzeitgedächtnis aus. Dieses vertiefte Verständnis ist besonders relevant für die Gestaltung von Visualisierungen und Interaktionen im Bereich der Visual Analytics.

## Mentale Modelle

Visualisierungen sind als Externalisierung dann besonders effektiv, wenn sie nicht mental übersetzt werden müssen. Priming des mentalen Modells des Probanden durch Verwendung entsprechender Formulierungen in der Aufgabe.

### Charakteristika:

1. Mentale Modelle sind unvollständig, einfacher als formale Modelle und pragmatische Nutzung der kognitiven Ressourcen
2. Mentale Modelle haben unscharfe Grenzen
3. Mentale Modelle sind instabil, vergessen und Entwicklungsfähigkeit
4. Mentale Modelle sind unwissenschaftlich: Nutzungsmuster mit technischen Systemen werden beibehalten, auch wenn klar wird, dass diese unnötig / nicht-optimal sind
5. Mentale Modelle sind oft zusammenhanglos, Delegation von Wissen auf den Rechner

Das Mentale Modell ist eine vereinfachte, subjektive Repräsentierung von realen Prozessen und ist eine Vereinfachung zur Handhabung im begrenzten Arbeitsgedächtnis (Quantitative Beziehungen werden auf qualitative reduziert, Stichproben werden verkleinert). Auf bekannte Sachverhalte wird durch Analogiebildung zugegriffen.

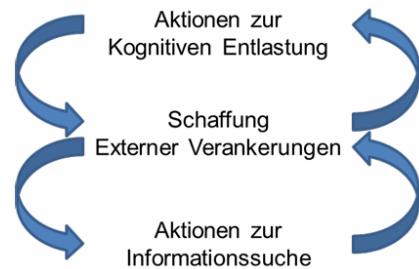
In der Visual Analytics geht es um die Untersuchung der Beziehung zwischen internen mentalen Modellen, die Personen im Kopf haben, und externen Repräsentationen wie graphischen Darstellungen. Interne Modelle sind abstrakte, mentale Konstrukte, die das Verständnis und die Erwartungen einer Person von der Welt widerspiegeln, während externe Modelle physische oder digitale Darstellungen dieser Verständnisse sind.

Bei der Interaktion mit graphischen Repräsentationen passen Menschen ihre internen Modelle an, basierend auf den Informationen, die durch die Visualisierungen vermittelt werden, was wiederum ihr Denken und Problemlösen unterstützt. Die Untersuchung in der Visual Analytics betrachtet interne und externe Repräsentationen nicht isoliert, sondern fokussiert auf deren Wechselwirkung und wie externe Visualisierungen die Bildung und Anpassung interner mentaler Modelle fördern.

## Interaktion

Kognitive Unterstützung durch Interaktion:

- Externe Verankerung: Analytisches Denken erfordert stabile Repräsentierungen, Verstehen des Bildes, Lokalisierung der Inhalte des Modells im Bild
  - Labels, Tick-Marks, Überschriften, Bildbeschreibung, Multiple-Linked Views, Model-Data-Linking, Visual Model Building
- Informationssuche: Simulation des mentalen Modells auf der Basis externer Repräsentierungen, Umstrukturierung des Modells, also suche bessere Repräsentierungen, Physische Aktion stabiler, präziser und billiger als mentale Transformationen
  - Interaktive Änderung von Visualisierungsparametern, Mappings, Transformationen, Multiple-Linked Views, Black-/White-Box Integration, automatische Steuerung
- Kognitive Entlastung: Konstruktion stabiler externer Repräsentierungen, Erstellung von Referenzen
  - Brushing, Notizen, Persistenz der erzeugten Modelle und der Visualisierungseinstellungen, Session-History



Implikation für das Design:

- Visualisierungsexperte: Hineinversetzen in das mentale Modell der Benutzer
- Benutzer: Verstehen des InfoVis-Designs und Feedback
- Konvergieren der mentalen Modelle
- Allgemein: Orientierung an kulturell-verankerten Modellen

Beziehung zwischen der Bedeutung einer Aussage innerhalb der Visualisierung und dem, was die Benutzer eigentlich ausdrücken möchten. Ein Design von InfoVis- und Visual Analytics-Systemen mit geringer semantischer Distanz erfordert ein intensives Verständnis der Problemdomäne und der Aufgaben der Benutzer.

## Evaluation

Die Evaluation hat Maßnahmen zur Prüfung der Effektivität, Effizienz und Nutzerzufriedenheit von Visual-Analytics- und Visualisierungstechniken in



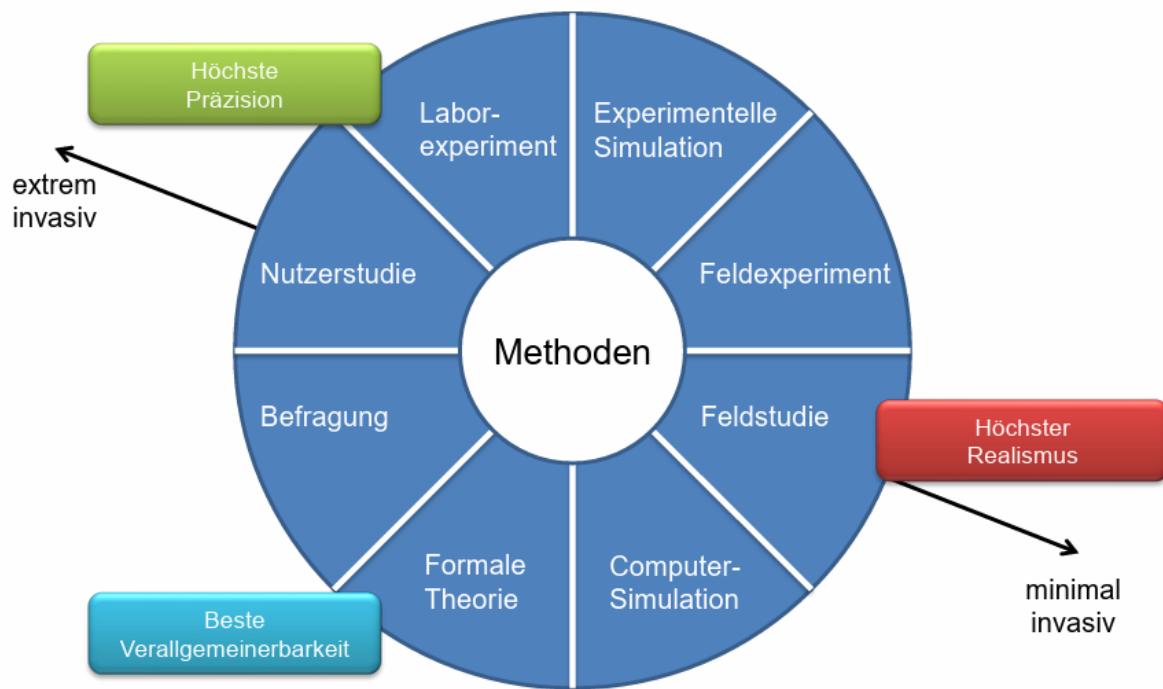
enger Verbindung mit kognitiven Ansätzen und Wahrnehmung. Dafür sind realistische Daten, Aufgaben und Szenarien nötig.

Die Visualisierung ist benutzendenorientiert und wird durch quantitative und qualitative Studien gemessen. Das Problem ist, dass jeder Mensch anders ist und Dinge anders wahrnimmt. Evaluationsergebnisse sind immer interpretationsbedürftig

Evaluierungsprozess:

- Was? ein interessanter Effekt, ein bestimmter Vorteil der Visualisierungstechnik
- Warum? Ziel der Evaluation
- Wann? Zeitpunkt im Entwicklungsprozess
- Wie? Evaluationsmethode oder -technik
- Wen? Auswahl der TeilnehmerInnen
- Wo? Ort der Studie
- Was? Ergebnisse der Studie
- Wer? Erwartungen an die Evaluatorin

## Testmethoden



Die Kriterien für Methoden (Verallgemeinerbarkeit, Präzision, Realismus) sind nicht alle gleichzeitig erfüllbar.

**Laborexperiment:** Stark kontrolliert, fokussierte Aufgabe, quantitativ, statistische Analyse, nicht realistisch.

**Nutzerstudie:** Probanden sind geladene Gäste, meist begrenzte Anzahl an Experten, geführte Evaluation, Beobachtung und Interview, Qualitativ und quantitativ, aber aufwendig.

**Feldstudie:** Keine Probanden, Aufgabe definiert Experte, kann langfristig sein mit einem offenen Protokoll, minimal invasiv, aber sehr zeitaufwändig.

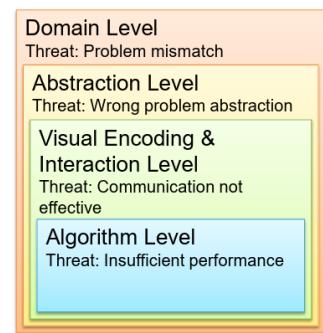
## Nested Model

Jede der Ebenen verlangt eigene Fragestellung und Methodiken. Die Fragestellungen bauen aufeinander auf.

Typische Probleme in der aktuellen Forschung sind falsche Fragestellungen, Vermischungen der einzelnen Stufen und unpassende Schlussfolgerungen zu den gestellten Fragen.

Starke Domänenunabhängigkeit verlangt Betrachtungen der kognitiven Psychologie.

Domänenabhängigkeit der Aufgabe bedingt domänenabhängige Evaluation. Evaluation muss für unterschiedliche Ebenen sehr unterschiedliche Methoden nutzen.



**High-Level-Ebene:** Einfluss der realen Welt erwünscht, wenig kontrollierbare Bedingungen

**Low-Level-Ebene:** Einfluss der realen Welt unerwünscht, wegen Präzision der Ergebnisse und Kontrollierbarkeit der Bedingungen.

## Task Levels

