Digital Applications & Data Management

WS25/26

Dr. Jens Kohl

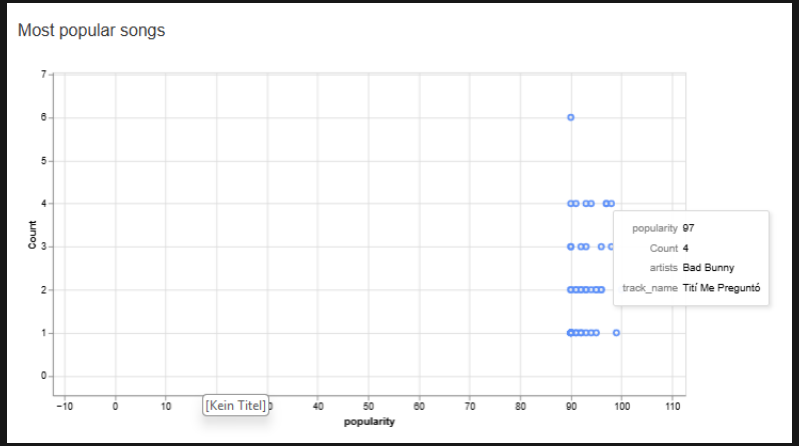
Roadmap Vorlesung

1. Einführung und Übersicht
2. Grundlagen Data Science
3. Vorgehen Data Science Use Case
4. Case Study Data Science
5. Grundlagen unüberwachtes Lernen
6. Grundlagen überwachtes Lernen (tabellarische Daten)
7. Case Study überwachtes Lernen (tabellarische Daten)
8. Grundlagen überwachtes Lernen (Bilddaten)
9. Case Study überwachtes Lernen und Transfer Learning (Bilddaten)
10. Grundlagen Generative AI
11. Generative AI mit Texten und Prompt Engineering
12. Agentic AI
13. Ausblick: Machine Learning in der Cloud und Reinforcement Learning

Vorlesung 3:
Vorgehen Data Science Use Case anhand Use Case Spotify

Was machen wir heute?

Motivation



Wir schauen uns anhand eines Fallbeispiels an, wie wir Erkenntnisse aus Daten gewinnen.
Das Fallbeispiel ist dabei komplexer als das vorige Beispiel: mehr Attribute, mehr Daten, ...

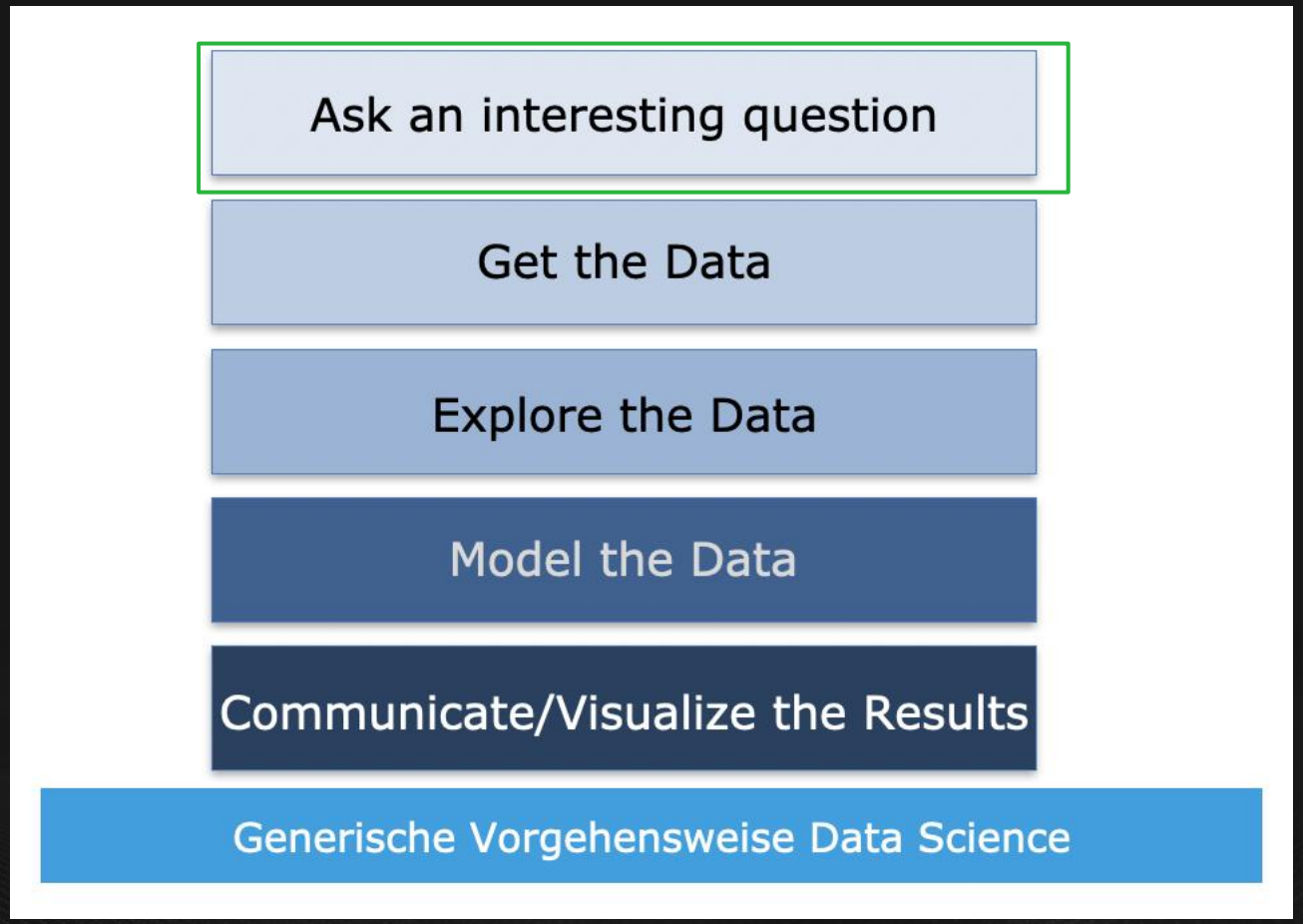
Fallbeispiel Spotify

Übersicht

- Datensatz: ~90 000 Lieder aus Spotify mit einigen Attributen
- In der Realität haben Sie oft mehr Daten und Eigenschaften („Features“) und vor allem Data- und Feature Engineering zu tun, um gute Ergebnisse zu erhalten.

Fallbeispiel Spotify

Vorgehen Data Science Use Case



Was sind die für den Geschäftszweck wichtigsten Informationen, die Sie aus dem Datensatz lernen können?

Quelle: Protopapas, Rader, Tanner, CS109 Data Science, 2020, Link, abgerufen am 15.09.2025

Fallbeispiel Spotify

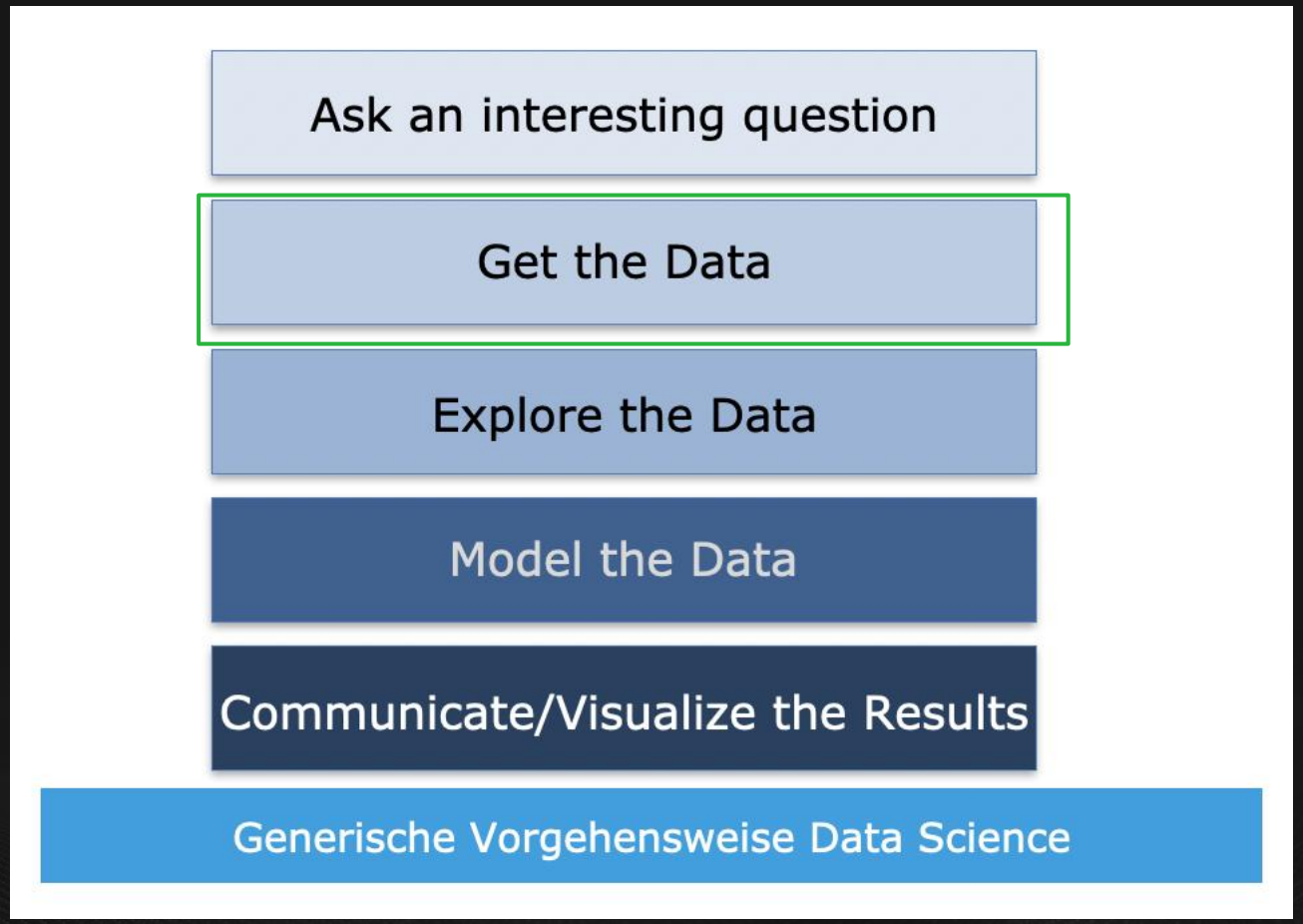
Übersicht

Für unseren Datensatz bieten sich beispielsweise folgende Fragen an:

- Was sind die populärsten Lieder? Was sind die unpopulärsten?
- Was für Eigenschaften haben sehr populäre/ nicht populäre Lieder?
- Welche dieser Eigenschaften haben den größten Einfluß auf Beliebtheit?

Angewandte Data Science

Vorgehen Data Science Use Case



Wie wurden die Daten generiert? Sind alle Daten relevant? Datenschutz?

Quelle: Protopapas, Rader, Tanner, CS109 Data Science, 2020, Link, abgerufen am 15.09.2025

Fallbeispiel Spotify

Übersicht Datensatz

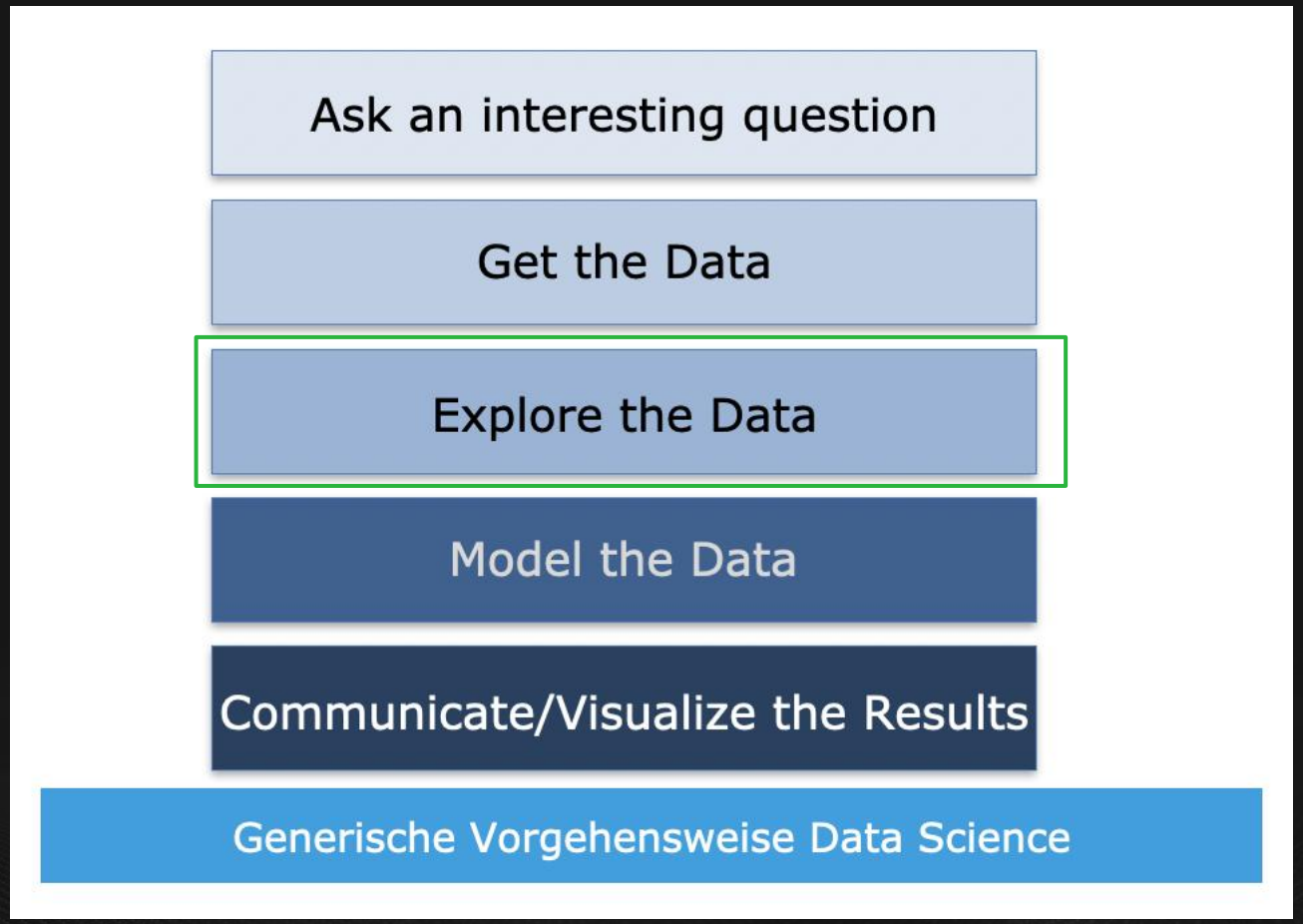
90’000 Zeilen mit folgenden Spalten:

- **track_id**: The Spotify ID for the
- track**artists**: The artists' names who performed the track. If there is more than one artist, they are separated by a ;
- **album_name**: The album name in which the track appears
- **track_name**: Name of the track
- **popularity**: **value between 0 and 100, with 100 being most popular**. Popularity is calculated by algorithm and is based, in the most part, on total number of plays track has had and how recent those plays are. Generally speaking, songs being played a lot now will have higher popularity than songs played a lot in the past. Duplicate tracks (e.g. the same track from a single and an album) are rated independently. Artist and album popularity is derived mathematically from track popularity.
- **duration_ms**: The track length in milliseconds
- **explicit**: true = yes it does; false = no it does not OR unknown
- **danceability**: how suitable track is for dancing based on combo of musical elements such as tempo, rhythm stability, beat strength, & overall regularity. Value of 0.0 is least and 1.0 is most danceable.
- **energy**: measure from 0.0 to 1.0 representing a perceptual measure of intensity & activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale
- **key**: The key the track is in. Integers map to pitches using standard Pitch Class notation. E.g. 0 = C, 1 = C♯/D♭, 2 = D, and so on. If no key was detected, the value is -1
- **loudness**: The overall loudness of a track in decibels (dB)
- **mode**: Mode indicates the modality (major or minor) of a track, the type of scale from which its melodic content is derived. Major is represented by 1 and minor is 0
- **speechiness**: detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks
- **acousticness**: confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic
- **instrumentalness**: Predicts whether a track contains no vocals. "Ooh" and "aah" sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly "vocal". The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content
- **liveness**: Detects presence of audience in recording. Higher liveness values represent increased probability that track was performed live. A value above 0.8 provides strong likelihood that the track is live
- **valence**: A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry)
- **tempo**: The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration
- **time_signature**: An estimated time signature. The time signature (meter) is a notational convention to specify how many beats are in each bar (or measure). The time signature ranges from 3 to 7 indicating time signatures of 3/4, to 7/4.
- **track_genre**: The genre in which the track belongs

Quelle: https://www.kaggle.com/datasets/maharshipandya/-spotify-tracks-dataset?resource=download

Fallbeispiel Spotify

Vorgehen Data Science Use Case



Erstes visuelles Verständnis? Gibt es Anomalien? Unplausible Werte? Sieht man erste Muster?

Quelle: Protopapas, Rader, Tanner, CS109 Data Science, 2020, Link, abgerufen am 15.09.2025

Fallbeispiel Spotify

Stochastik

Stochastik[1] besteht aus folgenden Teilgebieten:

- Wahrscheinlichkeitstheorie: mathematische Erfassung und Analyse zufälliger (nicht-deterministischer) Ereignisse [Backup]
- Mathematische Statistik[2]:
  - **Deskriptive Statistik: Daten durch Graphiken oder Tabellen visuell beschreiben.**
  - **Explorative Statistik[3]: Zusammenhänge/ Muster zwischen Daten finden und bewerten, Entdecken von Hypothesen**
  - Inferenzstatistik: aus einzelnen Eigenschaften einer Menge Eigenschaften über Gesamtmenge ableiten, Hypothesen testen

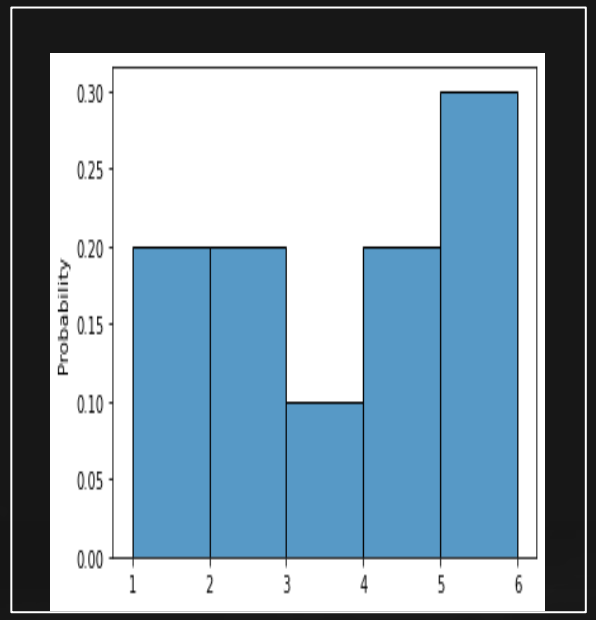
„Lies, damned lies, and statistics“
(Mark Twain)

1 Ratekunst, von στοχαστικὴ τέχνη
2 einordnen, von στατίζω
3 Begriff wurde geprägt von John Tukey 1977 in seinem Buch “Exploratory Data Analysis”

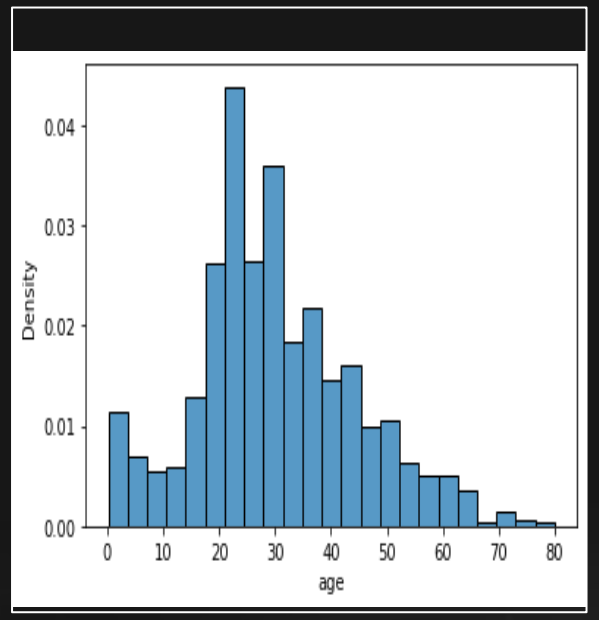
Fallbeispiel Spotify

Beispiele für deskriptive Statistik

Diskrete Verteilung

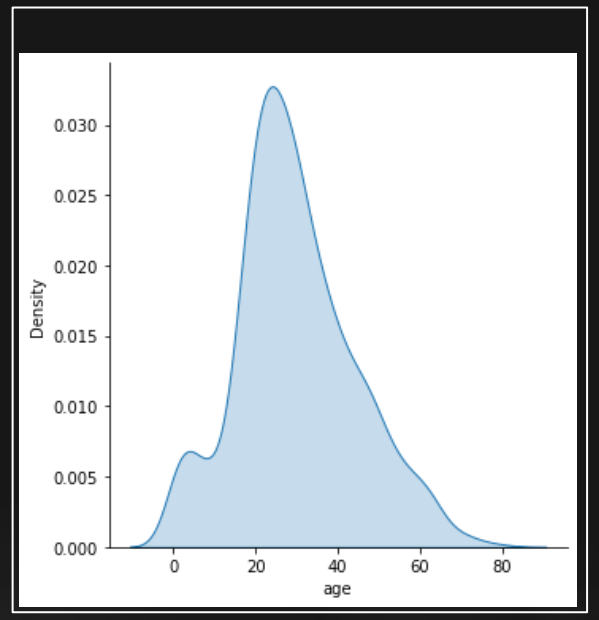


Rel. Häufigkeit Augen eines Würfels bei 10 Würfen.
Ergebnismenge = {1,...,6}

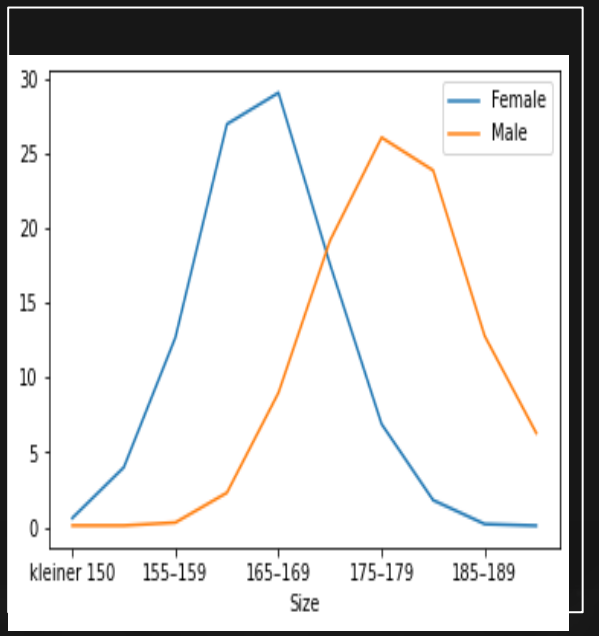


Diskretisierte Altersverteilung der Passagiere der Titanic.
Ergebnismenge in 23 „Körbe“

Kontinuierliche, stetige Verteilung



Kontinuierliche Altersverteilung der Passagiere der Titanic.
Ergebnismenge = $\mathbb{R}$



Größenverteilung Einwohner Deutschland in 2006[1]
Ergebnismenge = $\mathbb{R}$

Quellen: Bilder 1-3 eigene Darstellung, Bild 4 abgerufen unter Link

Fallbeispiel Spotify

Deskriptive Statistik: Übersicht wichtigste Parameter.

Lageparameter

- **Mean**: Mittelwert.
- **Median**: teilt Verteilung in 2 genau gleich große Hälften. Stabiler gegenüber Extremwerten als Mean.
- **Modus**: häufigster Wert der Verteilung.
- **Min**: kleinster Wert der Verteilung
- **Max**: größter Wert der Verteilung
- **P-Quantil**: Schwellenwert, der größer als p in % Elemente der Verteilung ist.

Streuungsparameter

- **Spannweite**: Abstand Min und Max-Wert
- **Varianz**: (quadratische) Abweichung Werte vom Mittelwert. Basis für Standardabweich.
- **Standardabweichung**: durchschnittliche Abweichung/Streuung Werte um Mittelwert.
- **Schiefe**: beschreibt Assymetrie Verteilung. Bei Rechtsschief sind häufiger Werte kleiner als Mittelwert, bei linksschief größer.
- **Wölbung**: Verteilungen mit geringer Wölbung streuen gleichmäßig; hohe W. bedeutet extremere, seltenere Ergebnisse.

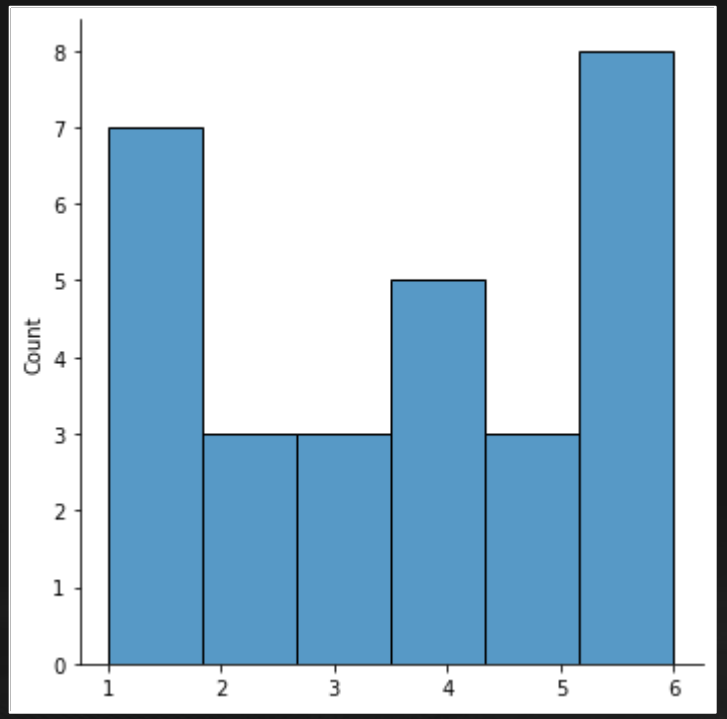
Zusammenhangsparam.

Spätere Vorlesung

Parameter ermöglichen eine komprimierte Erfassung einer Verteilung.

Fallbeispiel Spotify

Detaillierung Lageparameter



Mean = 3.62

Median: 4

Wird oft verwechselt!!!
Mean := Durchschnitt
Median := Wert, der Menge in genau 2 gleiche Hälften teilt

Modus: 6 ist häufigstes Ergebnis

Min: 1 ist niedrigster Ergebniswert

Max: 6 ist höchster Ergebniswert

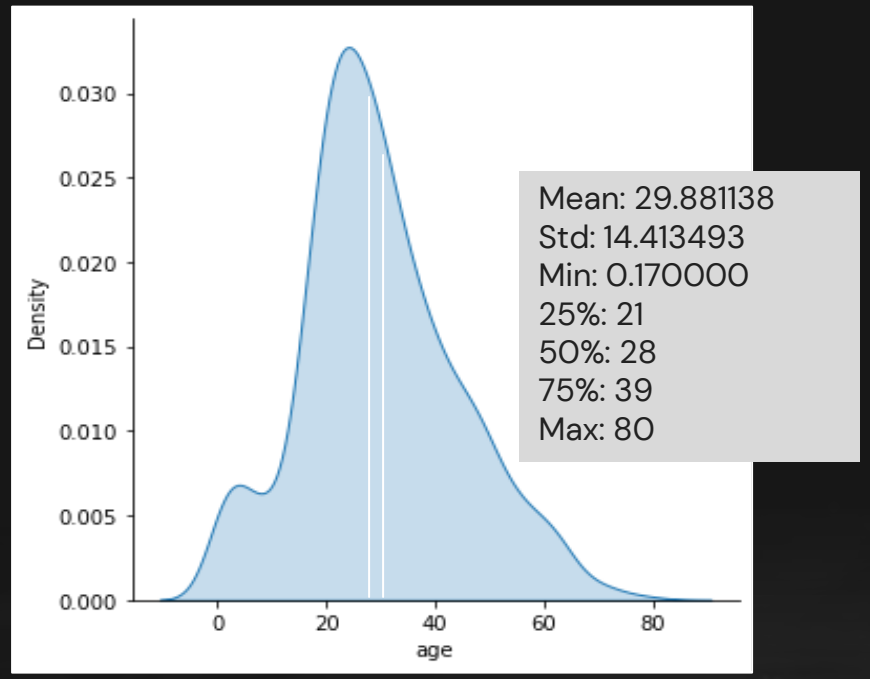
P-Quantil:

- 25% = 2 (7 von 30 Ergebnissen kleiner als 2)
- 50% = Median
- 75% = 6 (21 von 30 Ergebnissen kleiner als 6)

Parameter ermöglichen eine komprimierte Erfassung einer Verteilung.

Fallbeispiel Spotify

Detaillierung Lageparameter



Spannweite: 80 Jahre – 0,29 Jahre = 79,71 Jahre

Varianz: 207.55

Standardabweichung:14,41 → weite Streuung Alter

Schiefe: rechtsschief, da Median kleiner als Mean.

Mehr als 50% der Passagiere jünger als Durchschnittsalter.

Wölbung: geringe Wölbung, gleichmäßige Streuung.

Parameter ermöglichen eine komprimierte Erfassung einer Verteilung.

Quelle Bild: Visualisierung Ergebnisse Titanic-Notebook

Fallbeispiel Spotify

Deskriptive Statistik

```
# verwendeter Pompt: Verwende die describe Methode auf df_spotify.  Erkläre den Code, da ich ein Anfänger in Data Science und Python bin.
# Generiere eine Textzelle unter dieser Codezeile und erkläre dort die Ergebnisse von describe.
```

	Unnamed: 0	artists	album_name	track_name	popularity	duration_ms	explicit	danceability	energy	key	11
count	114000.000000	113999	113999	113999	114000.000000	1.140000e+05	114000	114000.000000	114000.000000	114000.000000	
unique	NaN	31437	46589	73608	NaN	NaN	2	NaN	NaN	NaN	
top	NaN	The Beatles	Alternative Christmas 2022	Run Rudolph Run	NaN	NaN	False	NaN	NaN	NaN	
freq	NaN	279	195	151	NaN	NaN	104253	NaN	NaN	NaN	
mean	56999.500000	NaN	NaN	NaN	33.238535	2.280292e+05	NaN	0.566800	0.641383	5.309140	
std	32909.109681	NaN	NaN	NaN	22.305078	1.072977e+05	NaN	0.173542	0.251529	3.559987	
min	0.000000	NaN	NaN	NaN	0.000000	0.000000e+00	NaN	0.000000	0.000000	0.000000	
25%	28499.750000	NaN	NaN	NaN	17.000000	1.740660e+05	NaN	0.456000	0.472000	2.000000	
50%	56999.500000	NaN	NaN	NaN	35.000000	2.129060e+05	NaN	0.580000	0.685000	5.000000	
75%	85499.250000	NaN	NaN	NaN	50.000000	2.615060e+05	NaN	0.695000	0.854000	8.000000	
max	113999.000000	NaN	NaN	NaN	100.000000	5.237295e+06	NaN	0.985000	1.000000	11.000000	

Next steps: Generate code with description | New interactive sheet

Fallbeispiel Spotify

Deskriptive Statistik

	Unnamed: 0	artists	album_name	track_name	popularity	duration_ms	explicit	danceability	energy	key	
count	114000.000000	113999	113999	113999	114000.000000	1.140000e+05	114000	114000.000000	114000.000000	114000.000000	11
unique	NaN	31437	46589	73608	NaN	NaN	2	NaN	NaN	NaN	
top	NaN	The Beatles	Alternative Christmas 2022	Run Rudolph Run	NaN	NaN	False	NaN	NaN	NaN	
freq	NaN	279	195	151	NaN	NaN	104253	NaN	NaN	NaN	
mean	56999.500000	NaN	NaN	NaN	33						
std	32909.109681	NaN	NaN	NaN	22						
min	0.000000	NaN	NaN	NaN	0						
25%	28499.750000	NaN	NaN	NaN	17						
50%	56999.500000	NaN	NaN	NaN	35						
75%	85499.250000	NaN	NaN	NaN	50						
max	113999.000000	NaN	NaN	NaN	100						

Next steps: Generate code with description | New interactive sheet

Wesentliche Ergebnisse der deskriptiven Analyse (df_spotify.describe(include='all')):
statistische Zusammenfassung der Spalten sowohl für numerische als auch für kategoriale Daten.
Für numerische Spalten (z.B. popularity, duration_ms, danceability, energy, loudness, tempo):

- **count**: Anzahl der vorhandenen (nicht fehlenden) Werte. Die meisten Spalten haben 114000 Einträge, was auf vollständige Daten hinweist, außer bei artists, album_name, track_name.
- **mean**: Durchschnittswert der Spalte. Z.B. liegt die durchschnittliche popularity bei ca. 33.24.
- **std**: Standardabweichung, Maß für die Streuung der Daten um den Mittelwert.
- **min/max**: Kleinster und größter Wert in der Spalte. Die popularity reicht von 0 bis 100.
- **25%, 50% (Median), 75%**: Quartile, die die Verteilung der Daten zeigen. Der Median der popularity ist 35. Der Wert für ein Quantil, z.B. 25% Quantil ist genau der Wert der größer ist als 25% der Datenmenge.

Für kategoriale Spalten (z.B. artists, album_name, track_name, explicit, track_genre):

- **count**: Anzahl der vorhandenen Werte.
- **unique**: Anzahl der verschiedenen (einzigartigen) Werte. Es gibt 114 einzigartige Musikgenres (track_genre).
- **top**: Der am häufigsten vorkommende Wert. Z.B. ist das häufigste Genre 'acoustic' oder artist „The Beatles"
- **freq**: Die Häufigkeit des am häufigsten vorkommenden Werts. Das Genre 'acoustic' kommt 1000 Mal vor.

Schneller Überblick über zentralen Tendenzen, Streuung und Verteilung der Daten sowie über Vielfalt in den kategorialen Spalten.

Fallbeispiel Spotify

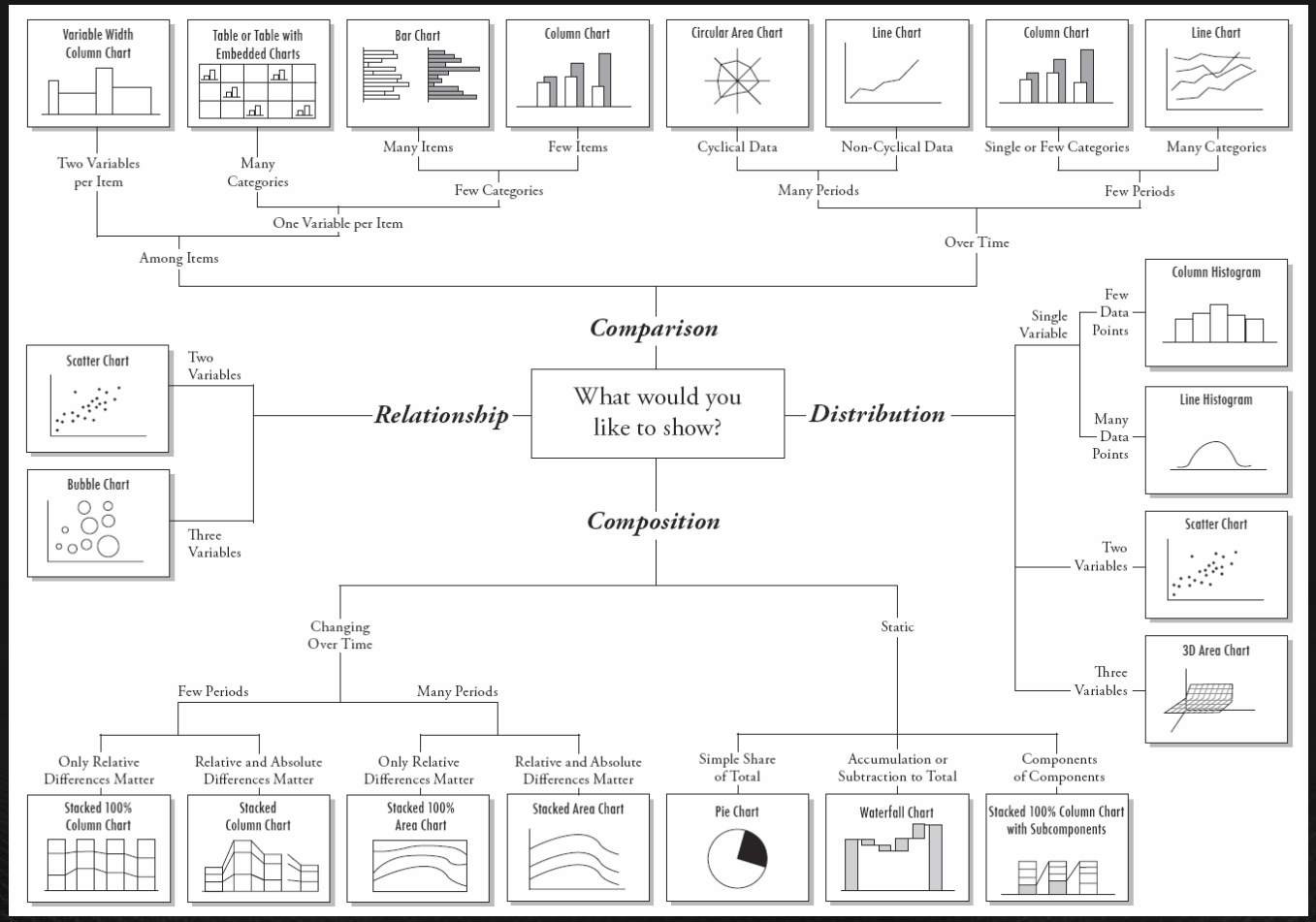
Explorative Statistik

- Daten aufbereiten und säubern:
  - Ersetzen von Nullwerten oder fehlende Werte (Data Imputation).
  - Entfernen von Duplikaten.
- Prüfen, ob Features relevant für die Hypothesen sind und ggf. Entfernen Features (Dimensionsreduktion).
- Entdecken von Ausreißern/ Anomalien in Features (Beispiel: Menschen mit Größe von 2,40 Meter oder mehr).
- Bauen neuer Features (z.B. Altersgruppen, ….)

- Entdecken von Mustern in Daten (Beispiel: gegenseitige Abhängigkeiten Features wie Einkommen und Wohnort).
- Bilden von Hypothesen (Beispiel: „In der 1. Klasse auf der Titanic war die Überlebenschance am höchsten“).

Fallbeispiel Spotify

Übersicht Plots/ Visualisierungen

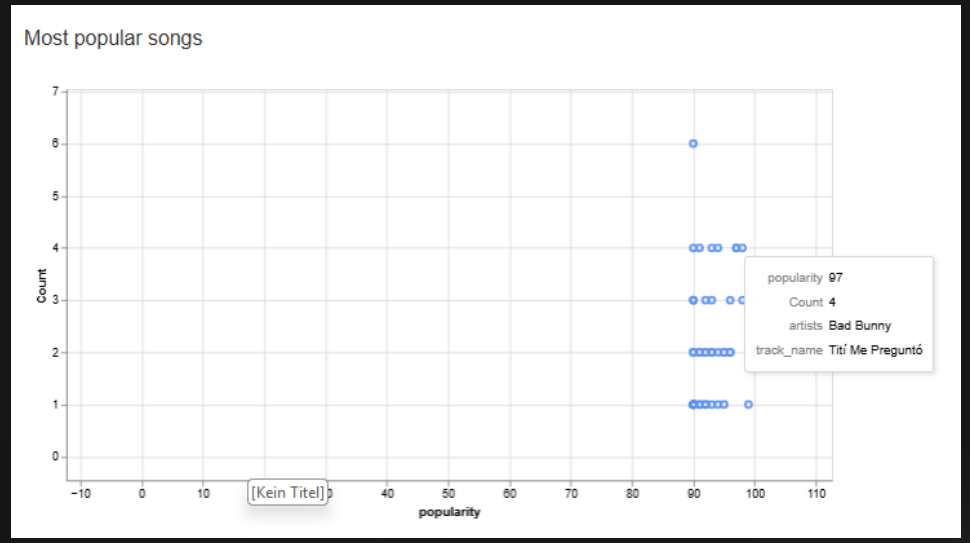


Quelle: Abela@Extreme Presentations, verfügbar unter: Link

Fallbeispiel Spotify

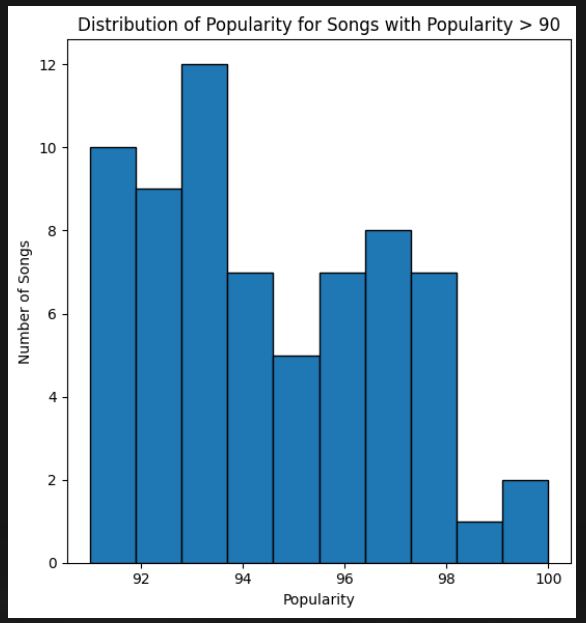
Explorative Statistik

PygWalker



Dokumentation: Link

MatPlotLib



Dokumentation: Link

PygWalker: schnelle Auswertungen, MatPlotLib hat größeren Funktionsumfang

Fallbeispiel Spotify

Data und Feature Engineering

- **Verbessern Lesbarkeit/ Erklärbarkeit**: sprechende Namen für Eigenschaften (**Features**) und Werte.
- **Imputation**: fehlende Werte löschen oder ersetzen (bspw. Mittelwert, definierter Wert, ….)
- **Typumwandlungen:** beim Einlesen der Daten werden numerische Features oft als Text erkannt.
- **Diskretisation**: Einteilen von Werten mit großem Wertebereich in Gruppen, bspw. Alter (Kind, Teenager, Erwachsene, …)
- **Categorization:** Werte mit beschränktem Wertebereich zusammenfassen (bspw. Farben, Wochentage, Geschlecht).
- **Outliers**: Werte, die sehr unterschiedlich zu restlichen Werten sind löschen oder per Standardwert ersetzen (bspw. Größe)
- **Normalisation/Scaling:** Werte innerhalb gewissen Wertebereichs bringen für Vermeiden Verzerrungen (bspw. Größe)
- **Feature Splitting**: Aufteilen Features für Infogewinn (bspw. Name in Vor-/Nachname, Adresse in Stadt und Straße)
- **One-hot-encoding**: Generieren neuer Features aus einem Feature (bspw. einzelne Tage statt Wochentage).
- Neue Features: Bauen neuer Features aus bestehenden oder Berechnungen (bspw. BMI aus Gewicht und Größe).
- **Feature removal**: Unwichtige Features löschen.

Data Engineering ist ein iterativer Prozess entlang des gesamten Use Cases und umfaßt oft 60 - 80% der Arbeit!

Fallbeispiel Spotify

Data und Feature Engineering

```
df_spotify.head(10)
```

	Unnamed: 0	artists	album_name	track_name	popularity	duration_ms	explicit	danceability	energy	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness	valence	tempo	time_signature	track_genre
0	0	Gen Hoshino	Comedy	Comedy	73	230666	False	0.676	0.4610	1	-6.746	0	0.1430	0.0322	0.000001	0.3580	0.7150	87.917	4	acoustic
1	1	Ben Woodward	Ghost (Acoustic)	Ghost - Acoustic	55	149610	False	0.420	0.1660	1	-17.235	1	0.0763	0.9240	0.000006	0.1010	0.2670	77.489	4	acoustic
2	2	Ingrid Michaelson;ZAYN	To Begin Again	To Begin Again	57	210826	False	0.438	0.3590	0	-9.734	1	0.0557	0.2100	0.000000	0.1170	0.1200	76.332	4	acoustic
3	3	Kina Grannis	Crazy Rich Asians (Original Motion Picture Sou...	Can't Help Falling In Love	71	201933	False	0.266	0.0596	0	-18.515	1	0.0363	0.9050	0.000071	0.1320	0.1430	181.740	3	acoustic
4	4	Chord Overstreet	Hold On	Hold On	82	198853	False	0.618	0.4430	2	-9.681	1	0.0526	0.4690	0.000000	0.0829	0.1670	119.949	4	acoustic
5	5	Tyrone Wells	Days I Will Remember	Days I Will Remember	58	214240	False	0.688	0.4810	6	-8.807	1	0.1050	0.2890	0.000000	0.1890	0.6660	98.017	4	acoustic
6	6	A Great Big World;Christina Aguilera	Is There Anybody Out There?	Say Something	74	229400	False	0.407	0.1470	2	-8.822	1	0.0355	0.8570	0.000003	0.0913	0.0765	141.284	3	acoustic
7	7	Jason Mraz	We Si																4	acoustic
8	8	Jason Mraz;Colbie Caillat	We Si																4	acoustic
9	9	Ross Copperman																	4	acoustic

- **Feature removal**: Löschen bspw. von Spalte Unnamed oder TrackID
- **Verbessern Lesbarkeit/ Erklärbarkeit**: sprechende Namen vorhanden?
- **Categorization:** Einsetzen von beschränkten Werten für Track_genre
- **One-hot-encoding**: Generieren neuer Features aus einem Feature (bspw. je 1 Spalte/ Feature je Wochentag).
- **Diskretisation**: Einteilen von Alter in Altergruppen wie Kind, Teenager, Erwachsene oder Einteilen Prozentwerte bei Popularity in Gruppen (0 – 25% := sehr schlecht, 25 – 50% := schlecht, ….)
- **Neue Features/ Feature splitting**: Aufbau neues Feature, bswp. Kombination key und mode zu C dur oder C moll.
- Mehrdeutigkeiten auflösen: ….
- **Imputation**: fehlende Werte löschen oder ersetzen.
- **Normalisation/Scaling:** Angleichen der Wertebereiche **von** Speechiness, Loudness und Tempo

Literatur

Statistik

- James, Witten, Hastie and Tibshirani: An Introduction to Statistical Learning (freies Ebuch unter: Link)
- Spiegelhalter: The Art of Statistics: Learning from data
- Witte: Statistics (10th Edition)
- Silver: The Signal and the noise
- Taleb: Black Swan
- Huff: How to Lie with Stastistics
- Wheelan: Naked statistics

Backup

WAHRSCHEINLICHKEITSTHEORIE

MOTIVATION WAHRSCHEINLICHKEITSRECHNUNG.

- Eine Münze wird geworfen: Welche Seite zeigt nach oben?
- Familie will Mitte August Grillen bei Sonnenschein. Kann sie die Wetterdaten der letzten Jahre nutzen, für ein gutes Datum?
- Roulette-Spielen in einer Spielbank: auf was sollte ich setzen?

Wahrscheinlichkeitsrechnung bietet uns mathematische Methoden für die Beantwortung solcher Fragestellungen.

GRUNDBEGRIFFE WAHRSCHEINLICHKEITSRECHNUNG AM FALLBEISPIEL MÜNZWURF.

Ergebnismenge: Menge aller möglichen Ergebnisse, z.B. **Ω = {Kopf, Zahl}.**
Eine **endliche Menge** wird als **diskret** bezeichnet**,** eine **nicht abzählbare Menge** als **kontinuierlich** (beispielsweise Zeit)**.**

Ereignis: auftretendes Element oder Teilmenge aus der Ergebnismenge, z.B. E:= Kopf geworfen

Definition **relative Häufigkeit von E**: $\frac{\text{relative Häufigkeit Ereignis E}}{\text{Anzahl aller Ereignisse}}$

Wir setzen die relative Häufigkeit Ereignis E gleich der Wahrscheinlichkeit E[1]. Dann können wir folgende Regeln definieren:

1. Pr[gesamte Ergebnismenge Ω] = 1
2. Pr[leere Menge ∅] = 0
3. 0 ≤ Pr[Ereignis E] ≤ 1
4. Pr[$\overline{\text{Ereignis E}}$] = 1 - Pr[Ereignis E] (Gegenwahrscheinlichkeit)
5. Pr[A∩ B] = Anzahl der gemeinsamen eingetretenen Ereignisse A und B
6. Pr[Ereignis A ∪ Ereignis B] = Pr[A] + Pr[B] – Pr[A∩ B] (A oder B trat auf)

Mit diesen 6 Regeln können wir diskrete und kontinuierliche Wahrscheinlichkeiten berechnen

1 Begründung hierfür erfolgt im späteren Verlauf dieser Vorlesung

EINFÜHRUNG WAHRSCHEINLICHKEITSRECHNUNG: BEDINGTE WAHRSCHEINLICHKEIT.

Die Wahrscheinlichkeit eines Ereignisses A kann sich ändern, wenn wir wissen, daß ein anderes Ereignis B schon geschah.

$$\mathbf{Pr}[\mathbf{A} \mid \mathbf{B}] = \frac{\mathbf{Pr}[\mathrm{A} \cap \mathrm{B}]}{\mathbf{Pr}[\boldsymbol{B}]}$$

Sprich: Wahrscheinlichkeit von A gegeben Evidenz B

Der Wert von Pr[B] „normalisiert" Pr[A|B], das heißt er passt die Wahrscheinlichkeit von A an die von B an.

Beispiele:

- Wie hoch ist die Wahrscheinlichkeit daß mindestens eine 3 gewürfelt wurde, falls eine ungerade Zahl gewürfelt wurde?
  Menge A = {3,4,5,6}, Menge B = {1,3,5}. Schnittmenge A und B = {3,5}. ➔ Pr[A|B] = $^{2/6}/_{3/6} = \frac{2}{3}$ = 66%

- Titanic: Wie hoch ist die Chance, daß ein Passagier Mann ist und überlebt?
  Anzahl überlebender Männer = 161, Anzahl männliche Passagiere = 843 ➔ Pr[A|B] = 161/843 = 19%

Die bedingte Wahrscheinlichkeit hilft bei der Untersuchung, wie stark ein Ereignis Einfluß auf ein anderes hat.

EINFÜHRUNG WAHRSCHEINLICHKEITSRECHNUNG: UNABHÄNGIGE VS. ABHÄNGIGE EREIGNISSE

Zwei (oder mehr) Ereignisse A, B sind statistisch unabhängig, falls ein Eintreten von A ein Eintreten von B nicht beeinflußt

$$\mathbf{Pr}[A \cap B] = \mathbf{Pr}[A] * \mathbf{Pr}[B]$$
$$\mathbf{Pr}[A \mid B] = \mathbf{Pr}[A]$$

Sprich: Evidenz von B ändert nicht die Wahrscheinlichkeit von A

Beispiele:

- In einer Schublade sind 5 paar schwarze Socken und 4 Paar weiße Socken. Sie ziehen 2 Paar Socken
  a. mit Zurücklegen in die Schublade (ordentlich!). Unabhängig?
  b. Ohne Zurücklegen und auf den Boden. Unabhängig?
- Titanic
  a. Überlebensrate Mann und seine Passagierklasse.
  b. Überlebenschance eines Passagiers und die Anzahl der Musiker in der Bordkapelle.

Prüfen Sie immer, ob Ereignisse voneinander abhängig sind (Correlation does not imply causation!!)

EINFÜHRUNG WAHRSCHEINLICHKEITSRECHNUNG: ZUFALLSVARIABLEN (RANDOM VARIABLE).

Zufallsvariablen ermöglichen, Ereignisse zu quantifizieren auch ohne Kenntnisse der gesamten Verteilung.

Beispiele:

- Eine Münze wird 3 mal geworfen. Y bezeichnet die Anzahl der Würfe mit Ergebnis „Kopf".
- Wir stehen an der Autobahn A9 und machen eine Verkehrszählung der LKW.
- Wir wählen zufällige Passagiere der Titanic und zählen mit X die Anzahl der Frauen.

Zufallsvariablen ermöglichen dann die Berechnungen der Wahrscheinlichkeit, bspw. höchstens 2 mal Kopf in 3 Würfen:
$Pr[X \leq 2] = Pr[X=0 \text{ Kopf geworfen}] + Pr[X=1 \text{ Kopf geworfen}] + Pr[X=2 \text{ Kopf geworfen}] = 1/8 + 3/8 + 3/8 = 7/8$

Der **Erwartungswert** definiert das Ergebnis, das die Zufallsvariable im Mittel (nach vielen Durchführungen) annimmt.

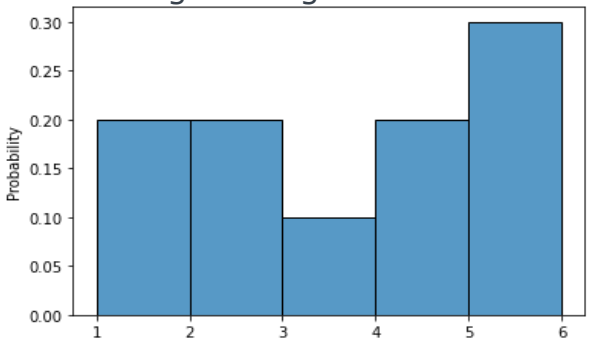
Die **Varianz** definiert die Streuung der Zufallsvariablen um den Erwartungswert (mehr dazu im nächsten Kapitel).

Wichtig ist beim Einsatz von Zufallsvariablen genügend oft zu messen („Sampling")!

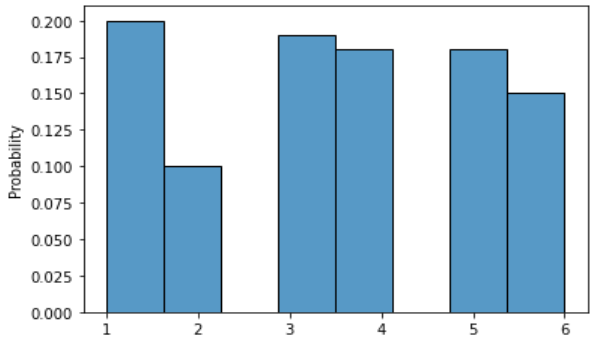
EINFÜHRUNG WAHRSCHEINLICHKEITSRECHNUNG: WAS IST GENÜGEND OFT MESSEN- ODER DAS GESETZ DER GROßEN ZAHLEN.

Wir messen mit den Zufallsvariablen X_1, …, X_6 wie oft bei einem Würfel Auge 1,…,6 gewürfelt wird.
Dabei interessiert uns, wie sich die relative Häufigkeit über die Anzahl der Würfe ändert.

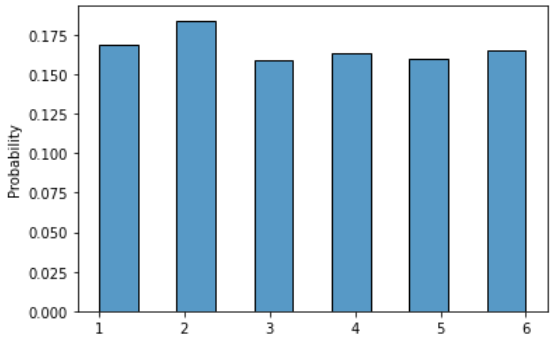
Rel. Häufigkeit Augen bei 10 Würfen



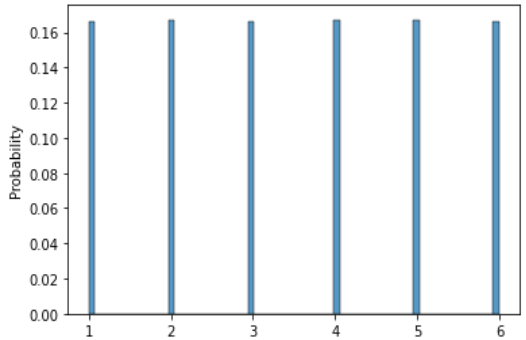
100 Würfen



1000 Würfen



500‘000 Würfen

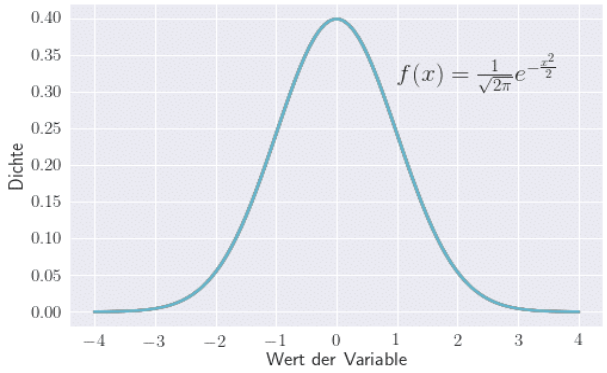


Gesetz der großen Zahlen: die relative Häufigkeit eines Ereignisses E nähert sich für hinreichend viele Wiederholungen seiner Wahrscheinlichkeit an.

Die Ergebnisse von Zufallsvariablen sind **nur dann** belastbar,
falls sie einer genügend großen Menge an Versuchen zugrunde liegen!!!

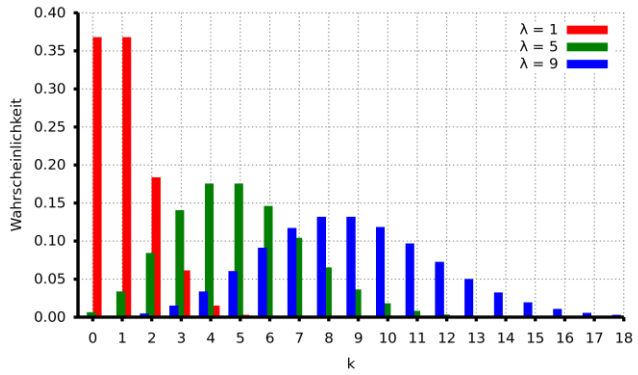
Quelle Bilder: Visualisierung Ergebnisse eines eigenen Python-Skripts

ÜBERSICHT WICHTIGER VERTEILUNGEN.



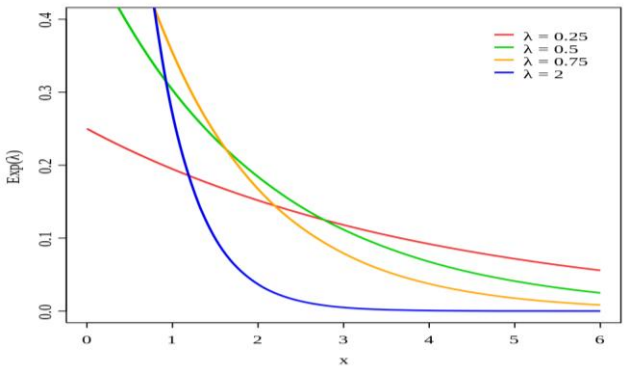
Normalverteilung:
Modellierung vieler natürlicher und statistischer Prozesse.

Beispiele:
- Größe Bevölkerung
- Prüfungsergebnisse
- Prozessqualität in einer Fabrik.



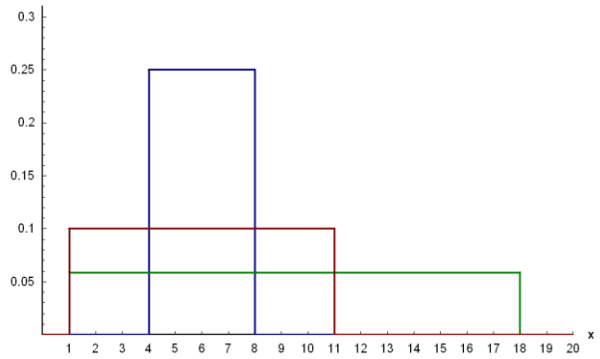
Poisson-Verteilung:
Modellierung Ereignisse, die bei konstanter mittlerer Rate unabhängig voneinander in einem festen Zeitintervall oder räumlichen Gebiet eintritt.

Beispiele:
- Hotline-Anrufe je Stunde
- Website-Ausfälle je Stunde



Expontentialverteilung:
Modellierung von Zeitintervallen.

Beispiele:
- Zeit bis Ausfall eines Geräts
- Wartezeit in Hotline



Gleichverteilung:
jeder Wert ist gleich wahrscheinlich (konstanter y-Wert).

Beispiele:
- Wurf einer idealen Münze oder Würfel

MATPLOTLIBS

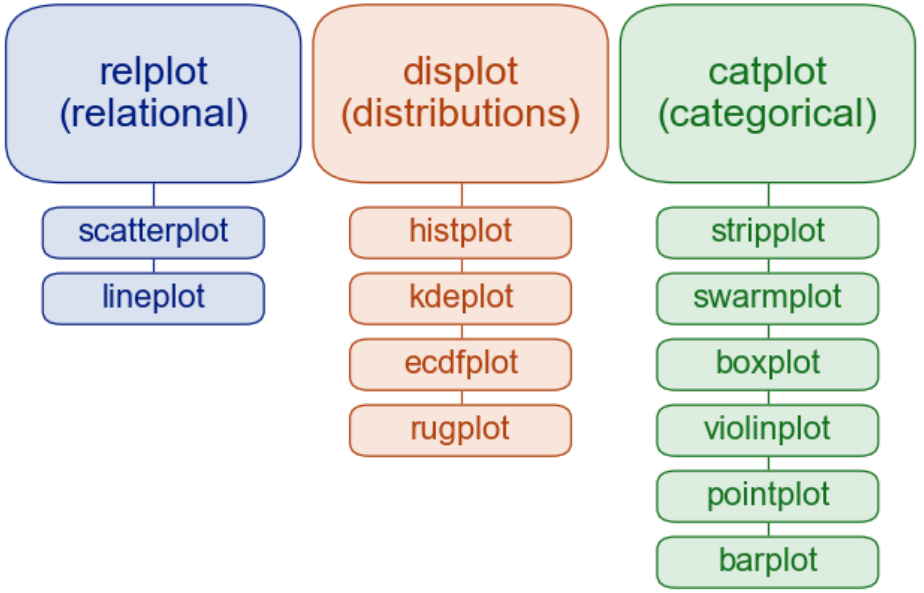
VISUALISIERUNG DATEN: ÜBERSICHT.

https://seaborn.pydata.org/tutorial.html

Was für Features werden geplottet?

- Zahlen
  - diskrete Werte: abzählbare Werte wie Ganzzahlen.
  - kontinuierliche Werte: nicht abzählbare, sehr viele unterschiedliche Werte wie reelle Zahlen.
- kategorische Variablen: Variablen mit einem Wert aus einer definierten Menge (bspw. Farben: rot, grün, ...).

Was für Plots gibt es?

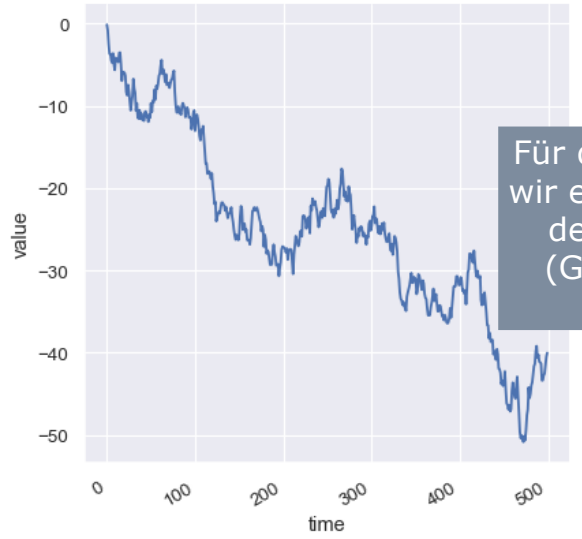


Die verschiedenen Plots unterscheiden sich, der Programmieraufruf ist aber prinzipiell gleich.

Quelle: Seaborn Dokumentation, verfügbar unter: Link

VISUALISIERUNG DATEN: RELATIONAL PLOTS.

Line Plots

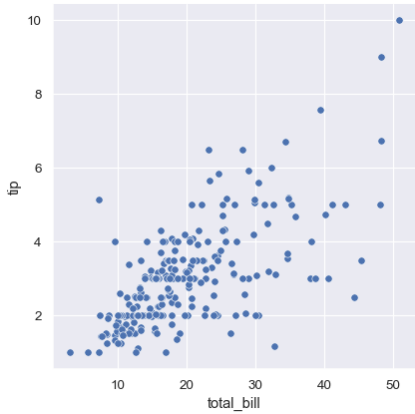


Für die x- und y-Achse nehmen wir ein Feature des Datensatzes der bei Data angegeben ist (Groß- und Kleinschreibung Feature beachten!)

```
sns.relplot(x="time",
            y="value",
            kind="line",
            data=df)
```

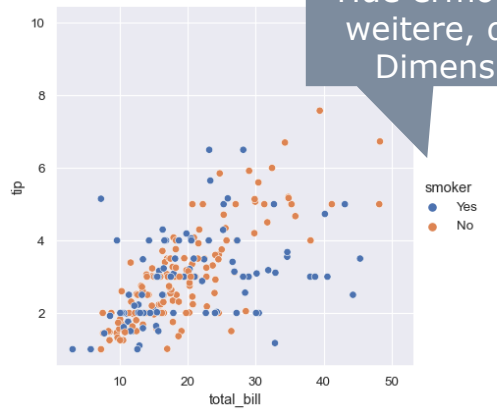
Ziel: Visualisierung von Änderungen über Zeit

Scatter-Plots




```
sns.relplot(x="total_bill",
            y="tip",
            data=tips)
```

Hue ermöglicht weitere, dritte Dimension



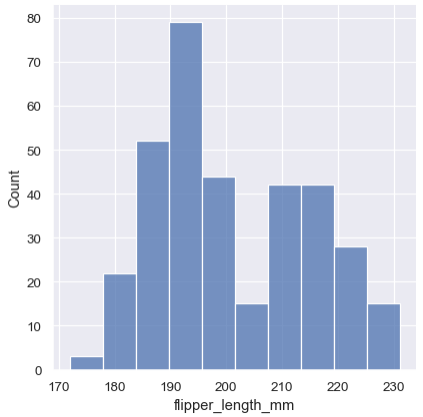

```
sns.relplot(x="total_bill",
            y="tip",
            hue="smoker",
            data=tips)
```

Ziel: Entdecken von Beziehungen zwischen 2 Features

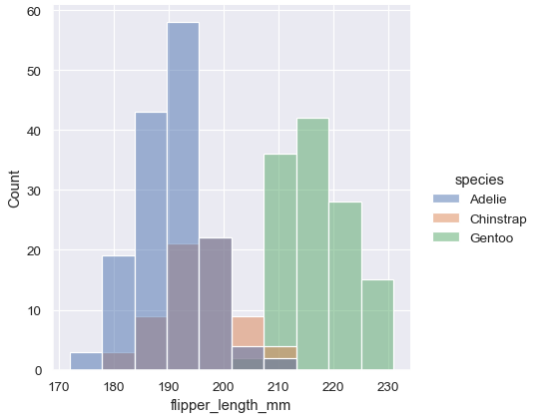
Quelle: Seaborn Dokumentation, verfügbar unter: Link

VISUALISIERUNG DATEN: VERTEILUNGEN.

Histogram (Histplot)




```
sns.displot(penguins,
x="flipper_length_mm")
```



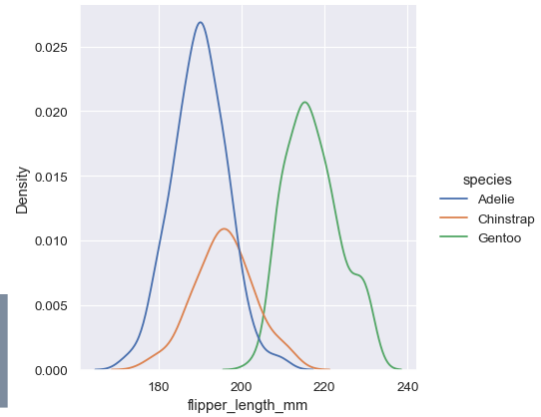

```
sns.displot(penguins,
            x="flipper_length_mm",
            hue="species")
```

Ziel: Entdecken Anomalien, Tendenzen, Verteilungen.
Aber: keine Visualisierung für kontinuierliche Features!

KDEPlot (Kernel density estimation)

Zusätzlicher Parameter Kind bei KDE

```
sns.displot(penguins,
            x="flipper_length_mm",
            kind="kde")
```



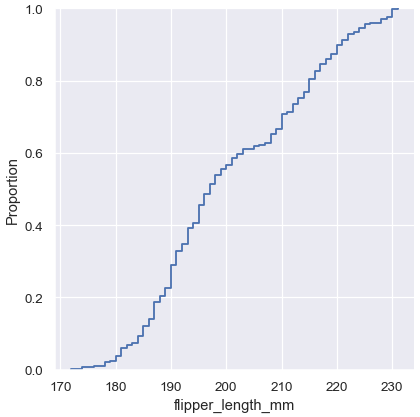

```
sns.displot(penguins,
            x="flipper_length",
            hue="species",
            kind="kde")
```

Ziel: Histogram für kontinuierliche Features.
Aber: Interpolation Zwischenwerte, kann falsch sein!

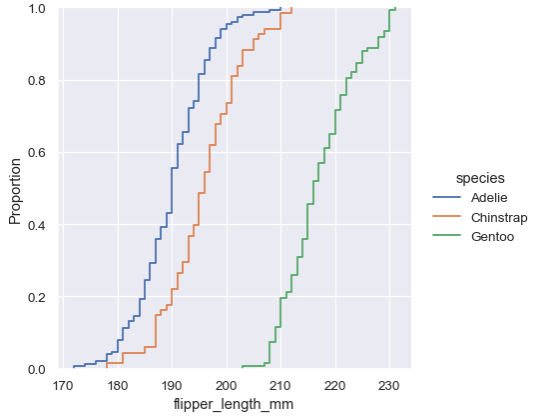
Quelle: Seaborn Dokumentation, verfügbar unter: Link

VISUALISIERUNG DATEN: VERTEILUNGEN.

Empirical cumulative distributions (ECDF)



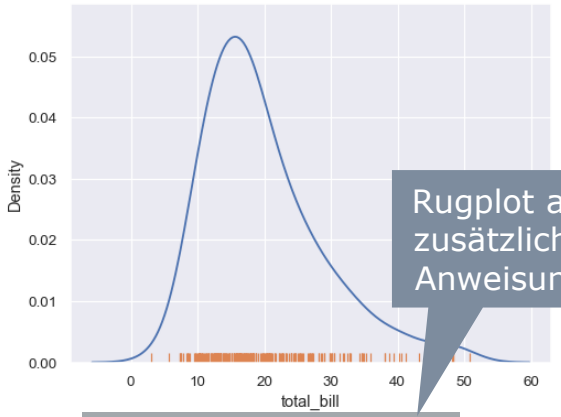

```
sns.displot(penguins,
x="flipper_length_mm",
kind="ecdf")
```




```
sns.displot(penguins,
x="flipper_length_mm",
hue="species",
kind="ecdf")
```

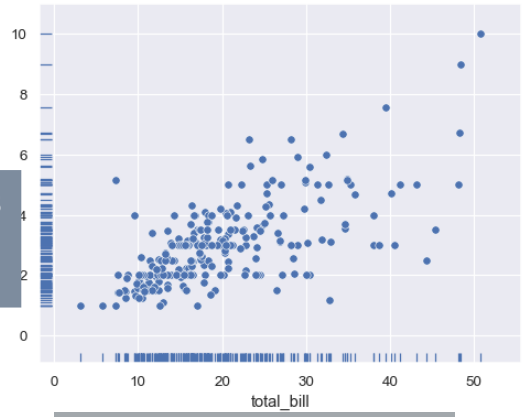
Abbilden jedes Wertes in Plot (Treppenfunktion).
Aber: weniger intuitiv.

Rugplots



Rugplot als zusätzliche Anweisung

```
sns.kdeplot(data=tips,
x="total_bill")
sns.rugplot(data=tips,
x="total_bill")
```



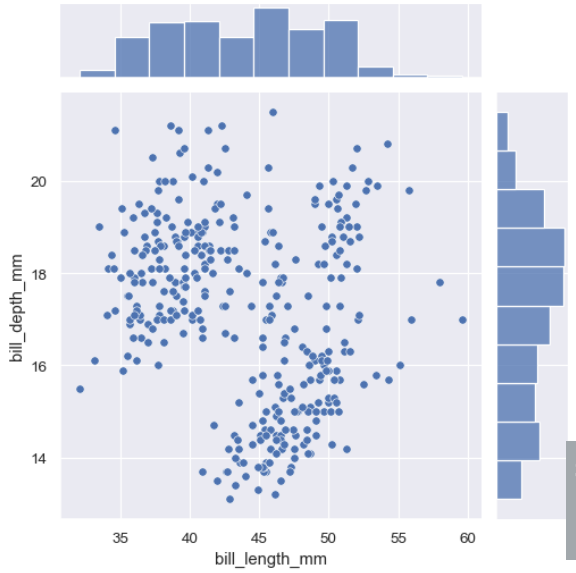

```
sns.scatterplot(data=tips,
x="total_bill", y="tip")
sns.rugplot(data=tips,
x="total_bill", y="tip")
```

Ziel: Aufzeigen Verteilung einer Variablen als zusätzliches Element in einem Plot. Aber: wird wenig genutzt

Quelle: Seaborn Dokumentation, verfügbar unter: Link

VISUALISIERUNG DATEN: WEITERE DISTRIBUTION PLOTS.

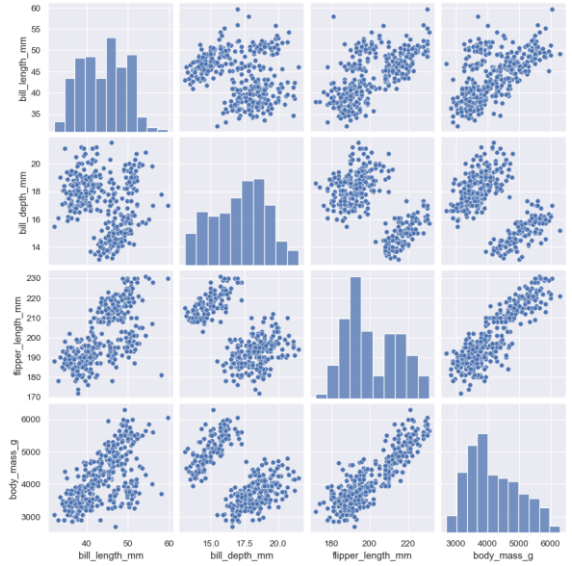
Joint-Plot




```
sns.jointplot(data=penguins,
              x="bill_length_mm",
              y="bill_depth_mm")
```

Ziel: Kombination von 2 verschiedenen Plots für Erkennen der Verteilung von Variablen und Beziehungen

Pairplot



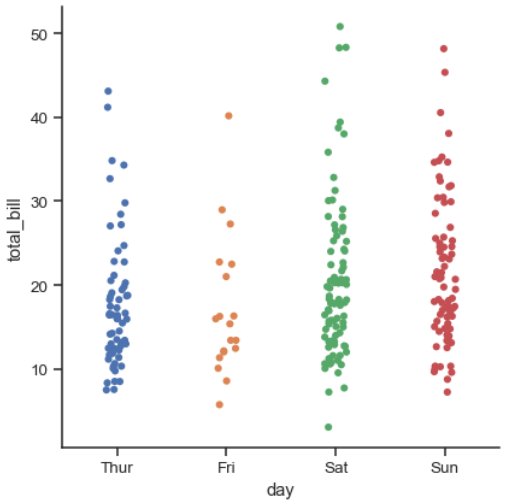

```
sns.pairplot(penguins)
```

Ziel: Entdecken von Beziehungen der Features zueinander

Quelle: Seaborn Dokumentation, verfügbar unter: Link

VISUALISIERUNG DATEN: KATEGORISCHE PLOTS.

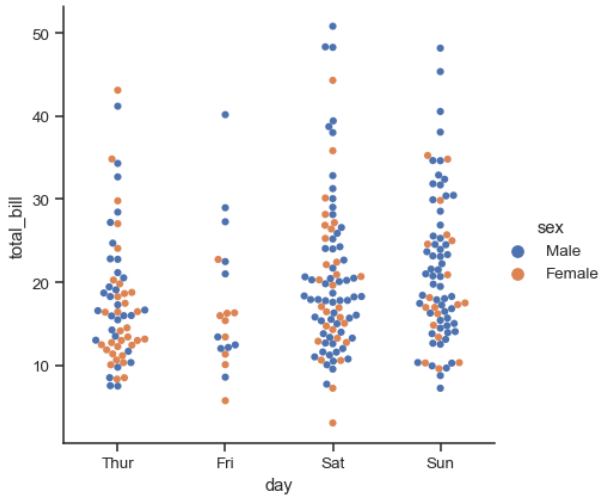
Kategorischer Scatterplot (Stripplot)




```
sns.catplot(x="day",
            y="total_bill",
            data=tips)
```

Ziel: Scatterplot für kategorische Variablen
Aber: eingeschränkte Sicht, da Punkte überlappen.

Kategorischer Scatterplot (Swarmplot)



Zusätzlicher Parameter Kind für Swarmplot

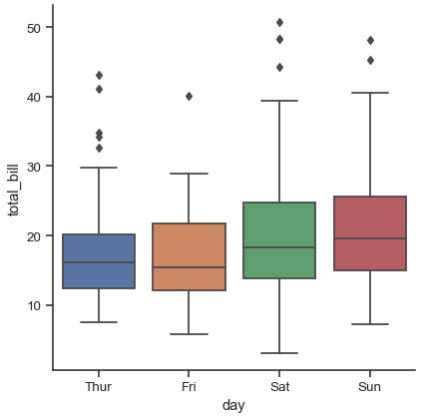
```
sns.catplot(x="day",
            y="total_bill",
            hue="sex",
            kind="swarm"
            data=tips)
```

Ziel: Verbessern Sichtbarkeit bei überlappenden Werten.
Aber: nur für kleine Datensätze.

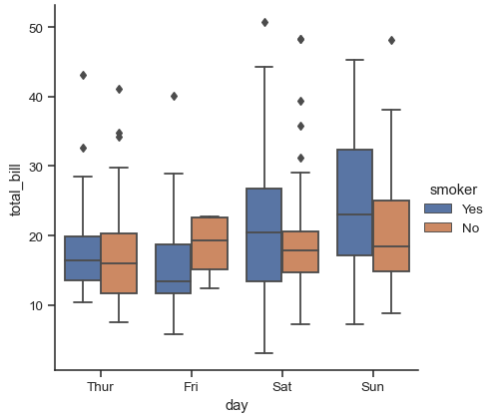
Quelle: Seaborn Dokumentation, verfügbar unter: Link

VISUALISIERUNG DATEN: KATEGORISCHE PLOTS.

Kategorischer Verteilungsplot (Boxplot)



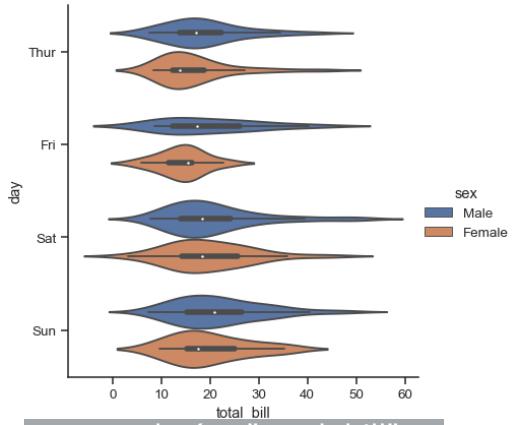

```
sns.catplot(x="day",
            y="total_bill",
            kind="box",
            data=tips)
```



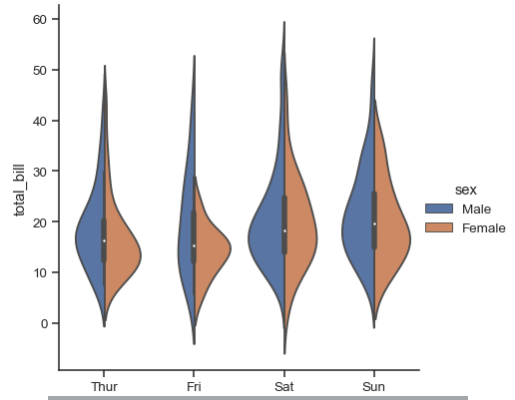

```
sns.catplot(x="day",
            y="total_bill",
            hue="smoker",
            kind="box",
            data=tips)
```

Ziel: Entdecken Anomalien, Tendenzen, Verteilungen.
Aber: keine Visualisierung für kontinuierliche Features!

Kategorischer Verteilungsplot (Violinplot)




```
sns.catplot(x="total_bill",
            y="day",
            hue="sex",
            kind="violin",
            data=tips)
```



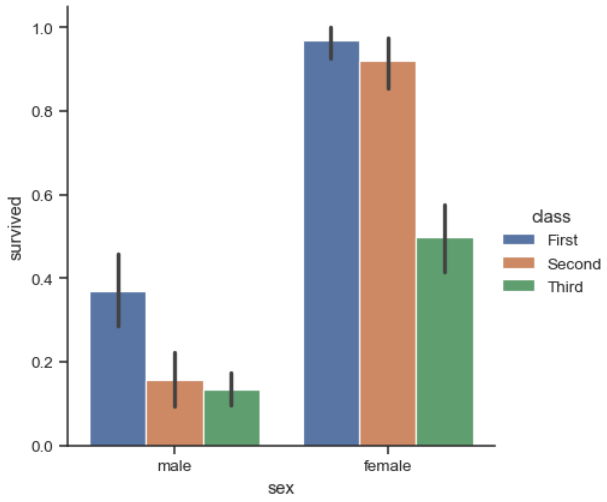

```
sns.catplot(x="day",
            y="total_bill",
            hue="sex",
            kind="violin",
            split=True,
            data=tips)
```

Ziel: Histogram für kontinuierliche Features.
Aber: Interpolation Zwischenwerte, kann falsch sein!

Quelle: Seaborn Dokumentation, verfügbar unter: Link

VISUALISIERUNG DATEN: KATEGORISCHE PLOTS.

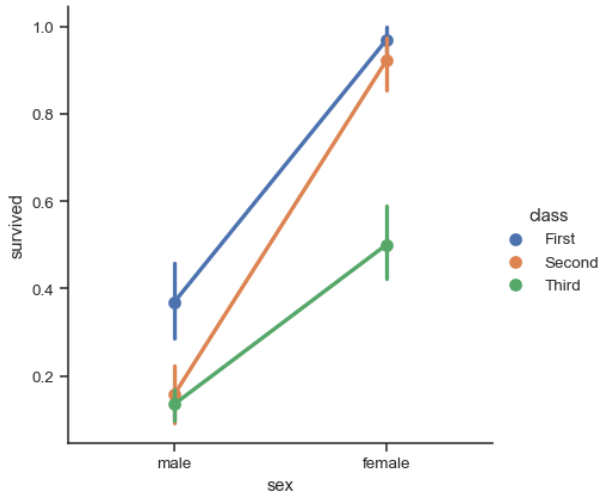
Statistische Abschätzung (Barplots)




```
sns.catplot(x="sex",
            y="survived",
            hue="class",
            kind="bar",
            data=titanic)
```

Ziel: Aufzeigen von Tendenzen.

Statistische Abschätzung (Pointplot)



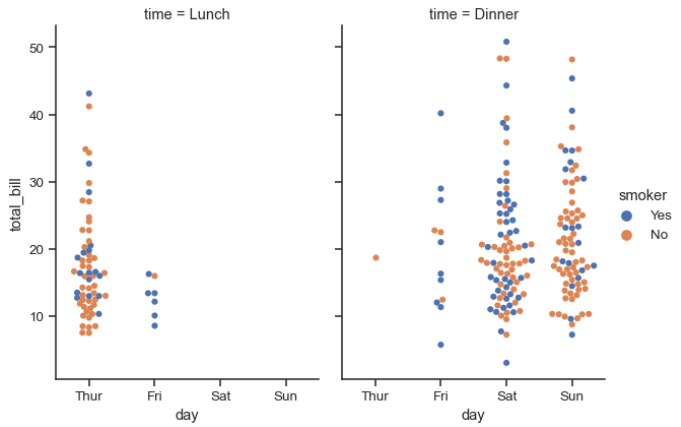

```
sns.catplot(x="sex",
            y="survived",
            hue="class",
            kind="point",
            data=titanic)
```

Ziel: Aufzeigen von Tendenzen

Quelle: Seaborn Dokumentation, verfügbar unter: Link

VISUALISIERUNG DATEN: WEITERE KATEGORISCHE PLOTS.

Visualisierung verschiedener Features.




```
sns.catplot(x="day",
            y="total_bill",
            hue="smoker",
            col="time",
            kind="swarm",
            data=tips)
```

Ziel: Aufzeigen von Tendenzen.

Quelle: Seaborn Dokumentation, verfügbar unter: Link

BAYES THEOREM UND BAYES`SCHE INFERENZ

BAYES THEOREM UND BAYES INFERENZ.

- Eine der wichtigsten und ältesten Methoden (1763!) für probabilistische Inferenz
- kann kontinuierliche und dynamische Aktualisierungen von Wahrscheinlichkeiten berechnen (d.h. über Verlauf der Zeit)
- Einsatzgebiete:
  - Autopilot Flugzeug oder autonomes Fahren: Fusion verschiedenster Sensoren für Lokalisieren inkl. Unschärfe.
  - Spam-Filter für Mails.
  - Expertensysteme inkl. Schlussfolgerung(en), bspw. für Medizin.

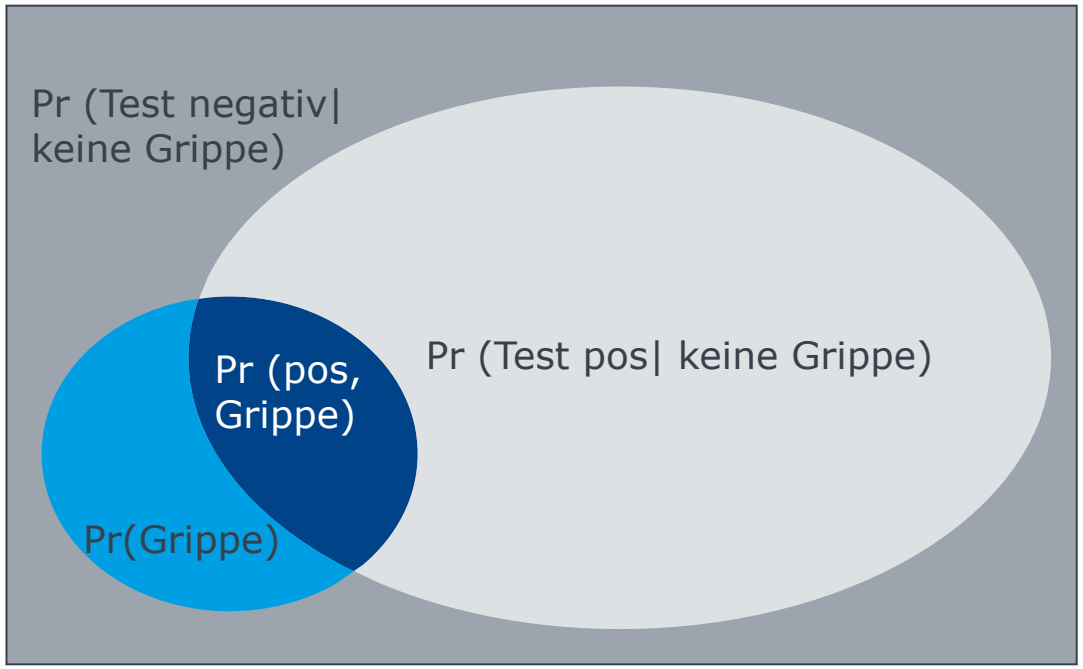
Bayes-Formel entsteht aus Umformungen der bedingten Wahrscheinlichkeit:

- Pr(A|B) := bedingte Wahrscheinlichkeit von A gegeben Evidenz B
- Pr(B|A) := bedingte Wahrscheinlichkeit von B gegeben Evidenz A
- Pr(A) := a priori (vorherige) Wahrscheinlichkeit von A
- Pr(B) := a priori Wahrscheinlichkeit von B

$$Pr(A|B) = \frac{Pr(B|A) * Pr(A)}{Pr(B)}$$

WIESO IST BAYES/ BAYES-INFERENZ SO WICHTIG? ODER: GRAPHISCHE VERANSCHAULICHUNG ANHAND DIAGNOSE.

Menge aller Menschen



Prior/ Vorher bekannte Wahrscheinlichkeiten:

- Prävalenz[1]: Häufigkeit Grippe Pr(Grippe)
- Sensitivität[2] Test: Pr(Test pos | Grippe)
- Spezifität[3] Test: Pr(Test neg | keine Grippe)

Uns interessieren die unbekannten Wahrscheinlichkeiten:

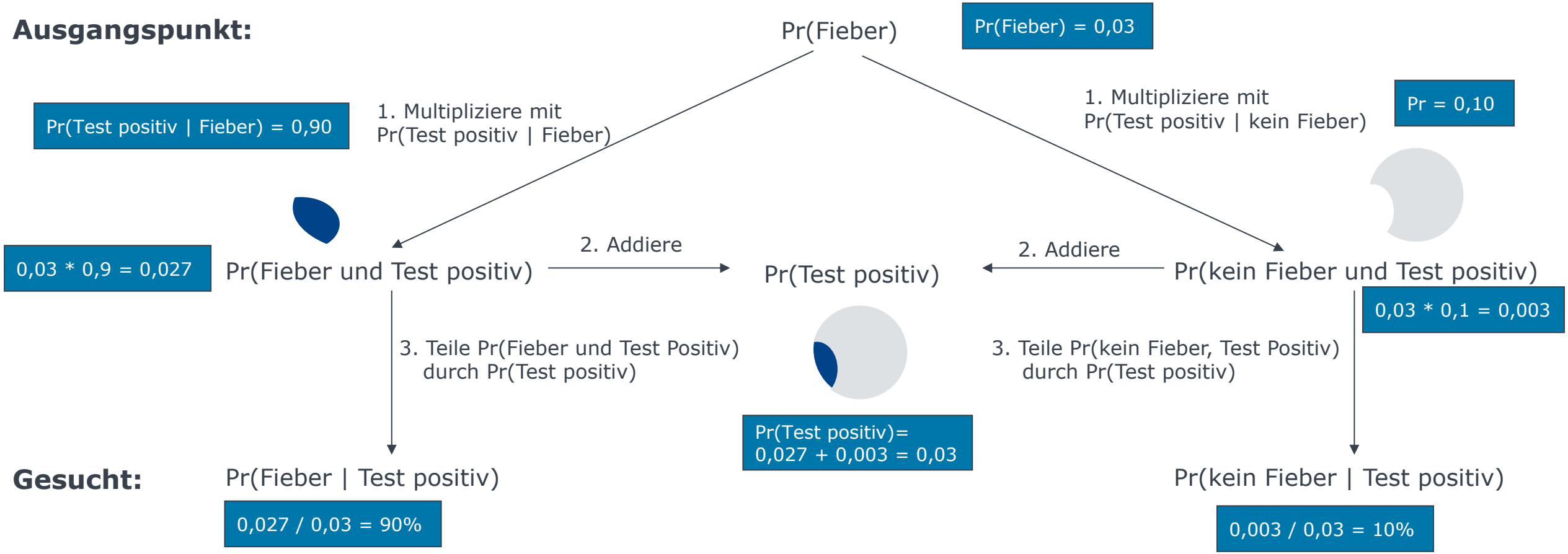
- Grippe bei positivem Test: Pr(Grippe | Test positiv)
- Fehlalarm Test: Pr(keine Grippe | Test positiv)

Beispiel Corona-Antigen-Tests[4]:
- Sensitivität ≥ 80%
- Spezifität ≥ 97%

1 Prävalenz: Wie viele Menschen haben die Krankheit (oft nur geschätzt!!!)
2 Sensitivität: Anteil an tatsächlich Kranken, bei denen auch eine Krankheit diagnostiziert wird
3 Spezifität: Anteil der Gesunden, bei denen tatsächlich keine Krankheit diagnostiziert wird
4 Quelle: Link

WIESO IST BAYES/ BAYES-INFERENZ SO WICHTIG? ALGORITHMUS BAYES INFERENZ.

Ausgangspunkt:



Nicht jede positiv getestete Person ist auch wirklich krank bei geringer Prävalenz!
Das ist wichtig und wird oft falsch verstanden: Link, Link (und viele weitere mehr ☹)