

P  
A B C  
I4

## 1. Lagemaße

## 2. Informationsvisualisierung

## 3. Streumaße

```
In [1]: # Standard-Imports
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

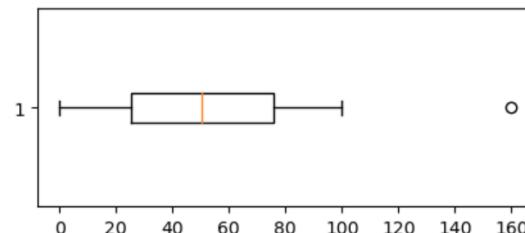
# darüberhinaus...
import seaborn as sns
```

## 1.1 Wiederholung: Boxplot

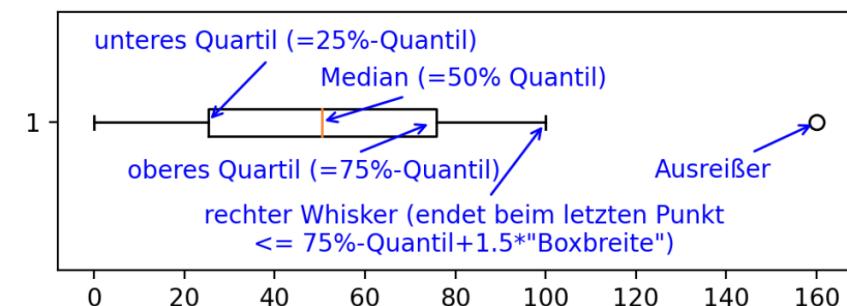
In [3]: # Punkte 0,1,...,100,160 (160 ist ein "Outlier")  
data = np.arange(0,102,1)  
data[-1] = 160  
data

Out[3]: array([ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12,  
13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25,  
26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38,  
39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51,  
52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64,  
65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77,  
78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90,  
91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 160])

In [86]: # Boxplot  
plt.boxplot ( data, vert=False );  
plt.gcf().set\_size\_inches(5,2);  
# Wieviel % der Daten Liegen in der Box?



Frage: Was bedeuten die jeweiligen Linien?



## 1.2 (empirisches) Quantil

Ein  $q$ -Quantil  $Q_q$  erfüllt die notwendige Eigenschaft, dass  $(q \cdot 100)\%$  der Daten kleiner gleich  $Q_q$  ist.

In [67]: `# die unteren 50% der Daten: 1, 2, ..., 50  
# die oberen 50% der Daten: 51, 52, ..., 100  
np.quantile ( range(1,101), q=0.5 )`

Out[67]: 50.5

In [68]: `# die unteren 50% der Daten: 0,1,...,49,(50)  
# die oberen 50% der Daten: (50),51,52,...,100  
np.quantile ( range(0,101), q=0.5 )`

Out[68]: 50.0

Es gibt verschiedene Möglichkeiten für eine formale Definition. Eine einfache Definition lautet:

(Man kann hier noch genauer werden: Wenn das gesuchte Quantil zwischen zwei Punkten liegt, aber näher am linken, dann sollte der Rückgabewert auch den linken Punkt höher gewichten. So ist es in numpy implementiert.)

$$Q_q(x) = \begin{cases} \frac{1}{2}(x_{nq} + x_{nq+1}), & \text{wenn } nq \text{ ganzzahlig ist,} \\ (x_{\lfloor nq+1 \rfloor}), & \text{wenn } nq \text{ nicht ganzzahlig ist.} \end{cases}$$

Hierbei sei der Datenvektor  $x = (x_1, \dots, x_n) \in \mathbb{R}^n$  der Größe nach sortiert, und  $\lfloor 3.14 \rfloor = 3$  bezeichne die Floor-Funktion.

## Verhalten bei Outliern:

In [21]: # Zahlen 0,1,...,99,100 (keine Outlier)  
data = np.arange(0,101)  
  
np.quantile(data,q=0.5),  
 np.quantile(data, q=0.99),  
 max (data)

Out[21]: (50.0, 99.0, 100)

In [22]: # Zahlen 0,1,...,99,10000 (ein Outlier)  
data = np.arange(0,100).tolist() + [10000]  
  
np.quantile(data, q=0.5),  
 np.quantile(data, q=0.99),  
 max(data)

Out[22]: (50.0, 99.0, 10000)

**Ergebnis:** Quantile sind robust gegenüber dem Vorliegen von Outliern.

**Best practice:** Zusätzlich zum Maximum z.B. das 99%-Quantil anschauen, zur Beschreibung des Wertebereichs der Daten.

## 1.3 Median

Der Median ist identisch zum 50%-Quantil der Daten: (wenn obige Definition des Quantils verwendet wird)

$$\text{Median}(x) = \begin{cases} \frac{1}{2}(x_{n/2} + x_{n/2+1}), & \text{wenn } n \text{ gerade ist,} \\ (x_{\lfloor n/2+1 \rfloor}), & \text{wenn } n \text{ ungerade ist.} \end{cases}$$

Hierbei sei der Datenvektor  $x = (x_1, \dots, x_n) \in \mathbb{R}^n$  der Größe nach sortiert,  
und  $\lfloor 3.14 \rfloor = 3$  bezeichne die Floor-Funktion.

Analog ist das Minimum identisch zum 0%-Quantil und das Maximum identisch zum 100%-Quantil.

**Frage:** Was ist der Median von  $x=[9,2,9,4]$  ?

In [2]: np.median ( [9,2,9,4] )

Out[2]: 6.5

## 1.4 Arithmetisches Mittelwert

Der (arithmetische) Mittelwert (engl.: *(arithmetic) mean*) eines Datenvektors  $x = (x_1, \dots, x_n) \in \mathbb{R}^n$  ist definiert via

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

**Frage:** Was ist der Mittelwert von  $x=[9,2,9,4]$ ?

In [3]: np.mean ( [9,2,9,4] )

Out[3]: 6.0

## Verhalten bei Outliern:

```
In [28]: # Zahlen 0,1,...,99,100 (keine Outlier)  
data = np.arange(0,101)  
  
np.median(data), np.mean(data)  
  
Out[28]: (50.0, 50.0)
```

```
In [29]: # Zahlen 0,1,...,99,10000 (ein Outlier)  
data = np.arange(0,100).tolist() + [10000]  
  
np.median(data), np.mean(data)  
  
Out[29]: (50.0, 148.01980198019803)
```

**Ergebnis:** Der Median ist robust gegenüber dem Vorliegen von Outliern, der Mittelwert nicht.

**Best practice:** Der Median ist eine gute Größe, die allgemeine Lage der Daten zu beschreiben (d.h. ein gutes *Lagemaß*).

**Frage:** Welche Gründe können Sie sich trotzdem für die Verwendung des Mittelwerts vorstellen?

## 1.5 Gesetz der großen Zahlen

Neben der größeren Akzeptanz ("Management-tauglich") des Mittelwerts weitere Begründung durch das G. d. gr. Z.

**Gesetz der großen Zahlen (Spezialfall):** Es seien  $X_1, X_2, \dots$  reellwertige Zufallszahlen (z.B. wiederholte Messungen):

$$X_i = \mu + \epsilon_i .$$

Hierbei modelliere  $\mu$  den wahren Wert und  $\epsilon_i$  ein zufälliges Rauschen, das während der Messung entsteht.

Dann gilt\*, dass der Mittelwert der ersten  $n$  Zahlen gegen den wahren Wert konvergiert:

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n X_i = \mu$$

\* unter technischen Annahmen, z.B.: Das Rauschen sei symmetrisch um 0 verteilt (d.h.  $\epsilon_1$  habe die gleiche Verteilung wie  $-\epsilon_1$ ); die einzelnen Messungen liefern unabhängige, aber identisch verteilte Ergebnisse; und es können keine beliebig hohen Rauschbeträge auftreten (d.h.  $\epsilon_1 \leq C$  für ein  $C > 0$ ).

**Interpretation:** Der arithmetische Mittelwert liefert auch für endliches  $n$  eine gute Beschreibung des "wahren Wertes".

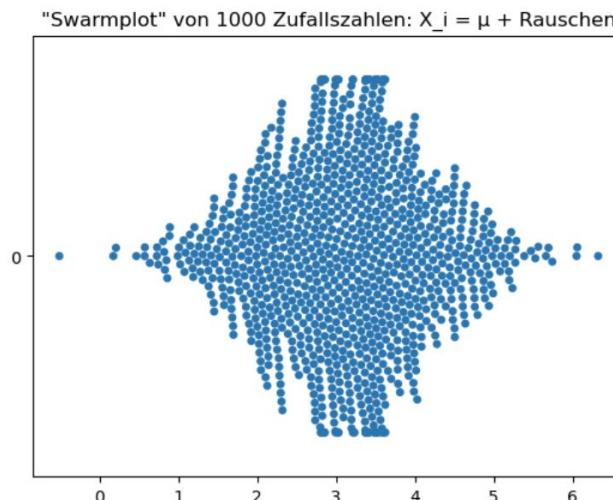
(Allgemein ist der "wahre Wert" der Erwartungswert  $\mathbb{E}[X]$  der Verteilung, die den Zufallszahlen  $X_1, \dots$  zugrunde liegt.)

## Gesetz der großen Zahlen im Beispiel:

In [95]:

```
# Zufallszahlen X1,X2,..., wobei X_i = μ + Rauschen
R = np.random.default_rng(42)
μ = 3.141592
x = R.normal(loc=μ, scale=1, size=1000)
sns.swarmplot(data=x, orient="h");
plt.title("Swarmplot" von 1000 Zufallszahlen: '+
'X_i = μ + Rauschen');

# hier wird ein Swarmplot verwendet; "geordnete"
# Alternative zum Stripplot, die sich visuell einer
# KDE annähert
```



In [50]:

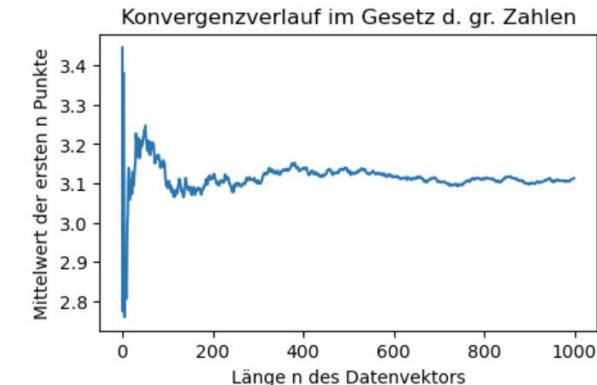
```
# Mittelwert der ersten n Zahlen
x[:10].mean(), x[:100].mean(), x[:1000].mean()
```

Out[50]:

```
(2.806020691594129, 3.0913223885161427, 3.11270044900405
33)
```

In [94]:

```
# Visualisierung des Konvergenzverlaufs
plt.plot(x.cumsum() / np.ones(1000).cumsum())
plt.xlabel("Länge n des Datenvektors")
plt.ylabel("Mittelwert der ersten n Punkte")
plt.title("Konvergenzverlauf im Gesetz d. gr. Zahlen");
plt.gcf().set_size_inches(5, 3);
```



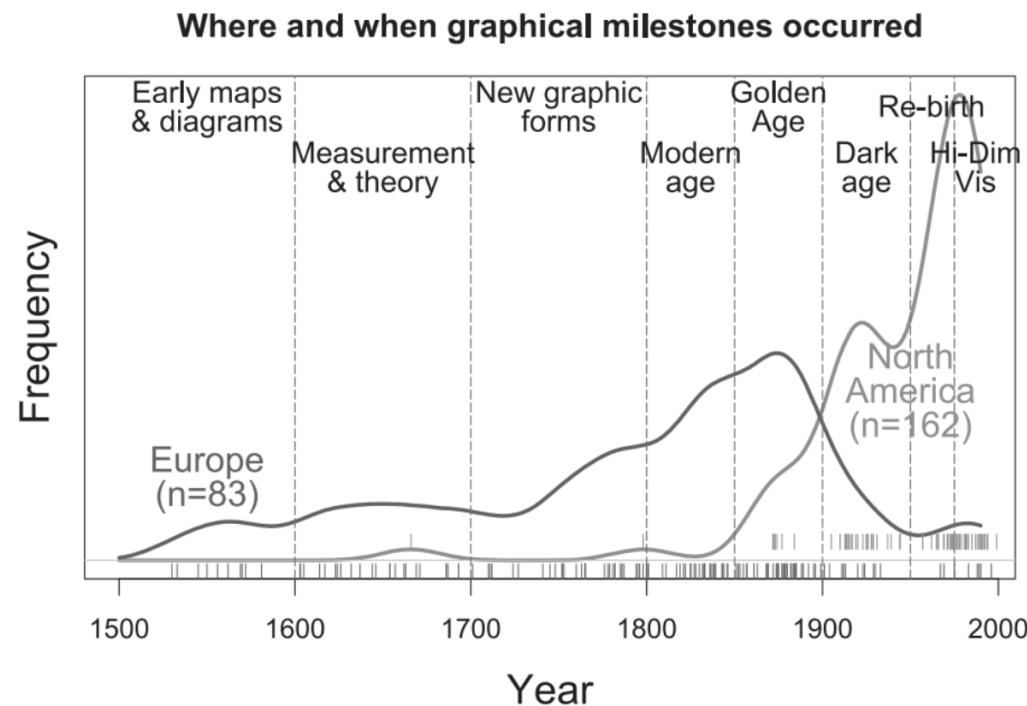
P  
A B C  
I4

**1. Lagemaße**

**2. Informationsvisualisierung**

**3. Streumaße**

## 2.1 Eine historische Einführung

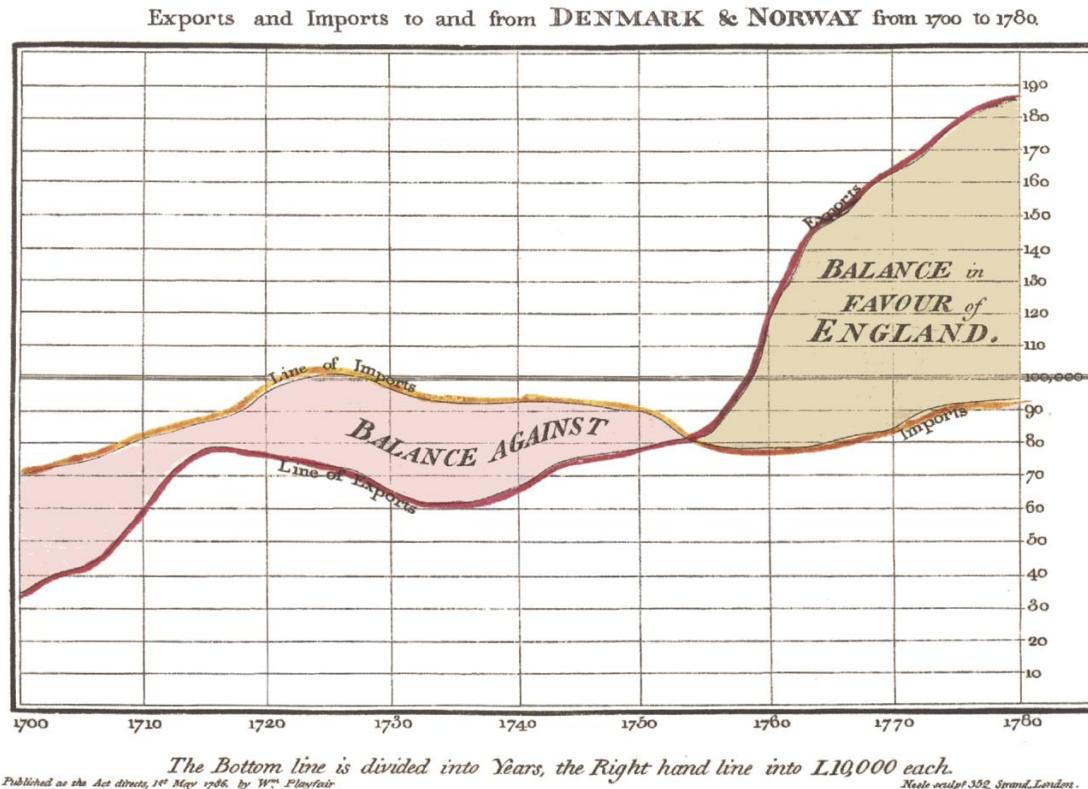


I.1 Time line of milestone events: Classified by place of development. Tick marks at the bottom show individual events. The smoothed curves plot their relative frequency, in Europe and North America. Source: © The Authors.

M. Friendly, H. Wainer: A History of Data Visualization and Graphic Communication, Harvard University Press, 2021

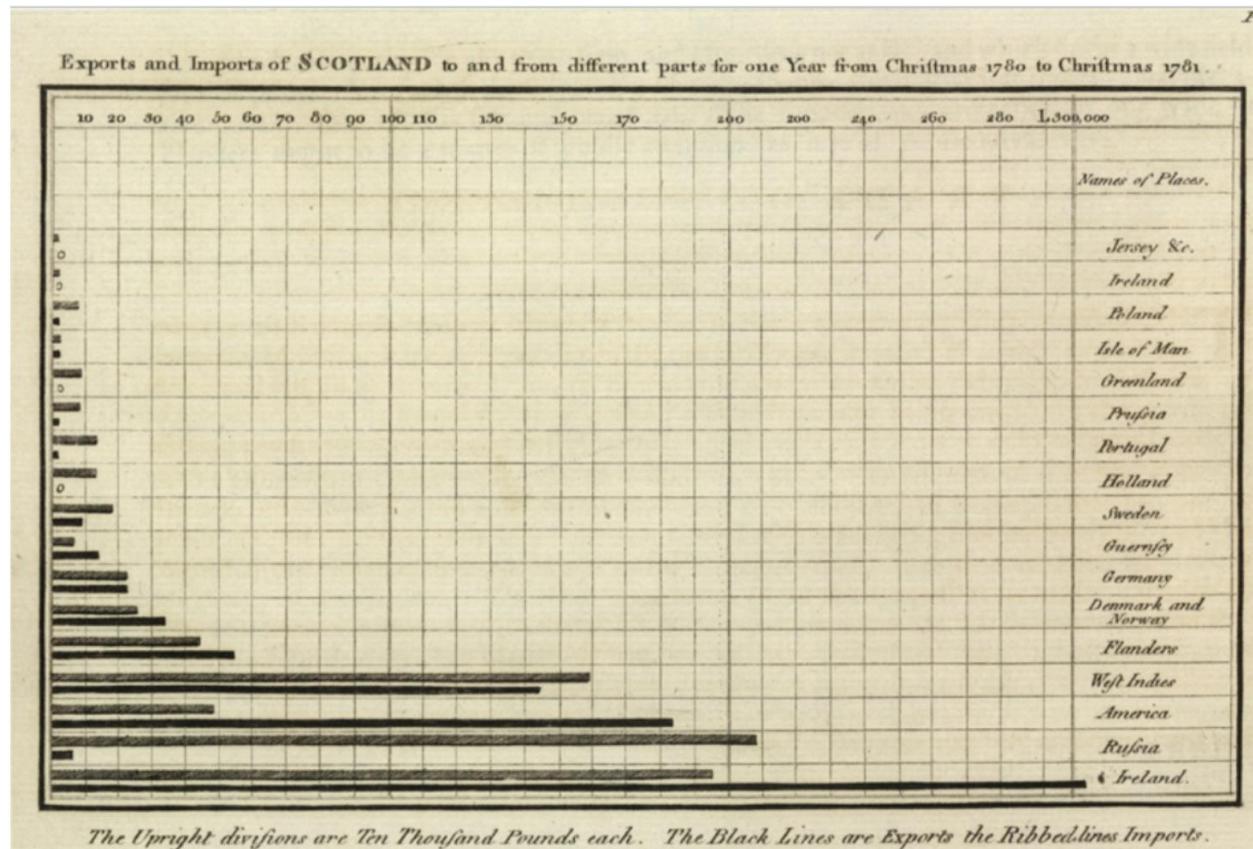
XXXXXXXXXX

## William Playfair (1759-1823): Ein Meilenstein in der grafischen Darstellung von Daten



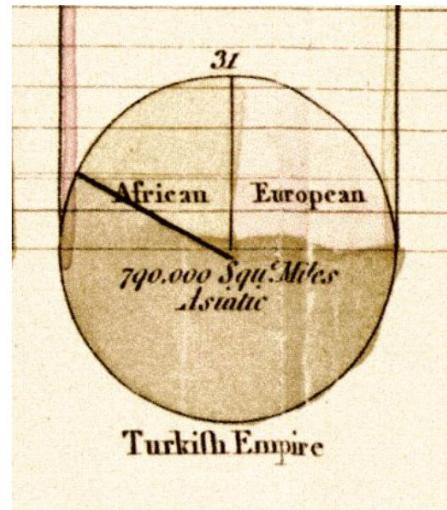
William Playfair: Commercial and Political Atlas, 1786 ([https://en.wikipedia.org/wiki/File:Playfair\\_TimeSeries-2.png](https://en.wikipedia.org/wiki/File:Playfair_TimeSeries-2.png))

## William Playfair (1759-1823): Mutmaßlich Erfinder von Balken- und Piechart ...

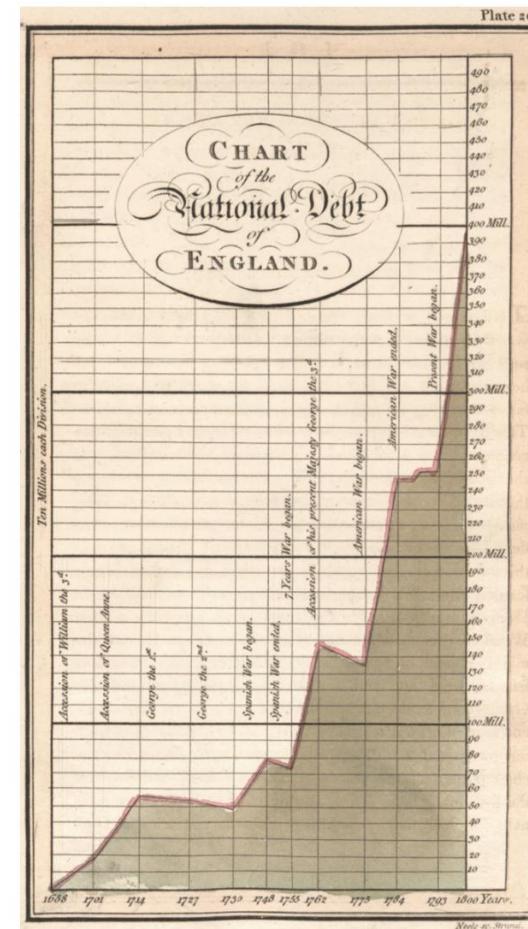


William Playfair: Commercial and Political Atlas, 1786 ([https://en.wikipedia.org/wiki/William\\_Playfair](https://en.wikipedia.org/wiki/William_Playfair))

... allerdings im Detail manchmal noch nicht ganz modern:

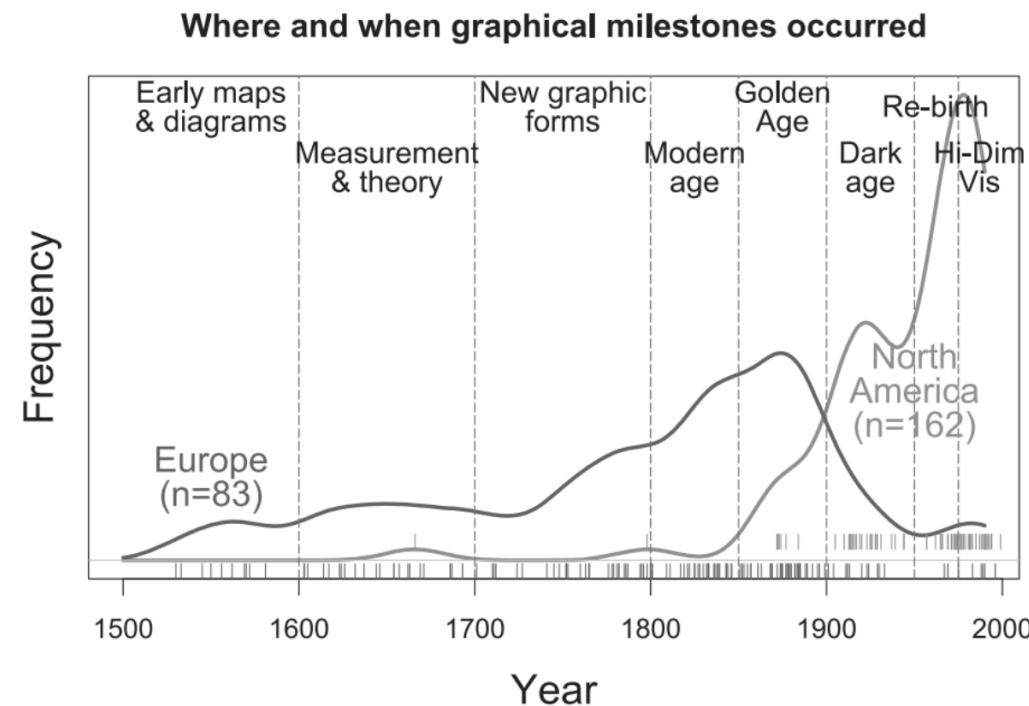


William Playfair: The Commercial and Political Atlas and Statistical Breviary, 1801 ([https://en.wikipedia.org/wiki/William\\_Playfair](https://en.wikipedia.org/wiki/William_Playfair))



(Jahre sind nicht äquidistant verteilt)

Wainer, H., Friendly, M. (2022). On the Origins of Data Visualization. In: Carriquiry, A.L., Tanur, J.M., Eddy, W.F. (eds) Statistics in the Public Interest. Springer Series in the Data Sciences. Springer, Cham. [https://doi.org/10.1007/978-3-030-75460-0\\_27](https://doi.org/10.1007/978-3-030-75460-0_27)



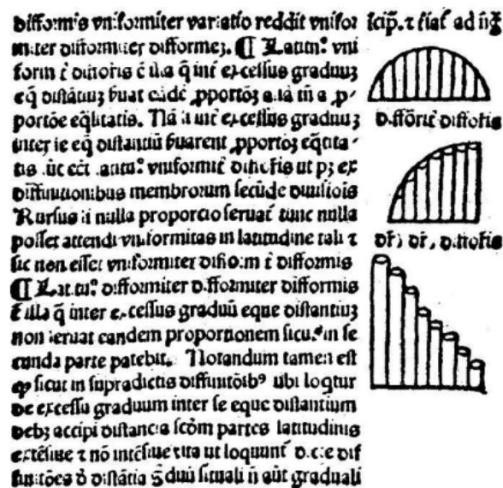
I.1 Time line of milestone events: Classified by place of development. Tick marks at the bottom show individual events. The smoothed curves plot their relative frequency, in Europe and North America. Source: © The Authors.

M. Friendly, H. Wainer: A History of Data Visualization and Graphic Communication, Harvard University Press, 2021

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

## Frühe Datenvisualisierungen:

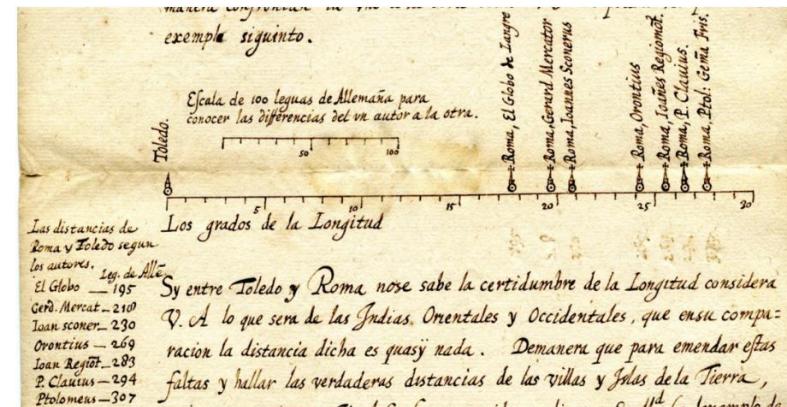
### "Oresme's Pipes" (~1360):



"If Oresme had data,  
we might have had  
statistical graphs  
400 years before Playfair"

(aus [2], Zitat aus Funkhouser, 1937)

### Michael Florent van Langren (1628):



### "The first (Known) Statistical Graph"

Friendly, Michael et al. "The First (Known) Statistical Graph: Michael Florent van Langren and the "Secret" of Longitude." The American Statistician 64 (2010): 174 - 184. <https://www.datavis.ca/papers/langren-TAS09154.pdf>

Der erste "Statistische Graph":

Michael Florent van Langren (1628):

Schätzung des Längengradunterschieds zwischen Toledo und Rom.



Figure 8. van Langren's 1644 graph, linearly rescaled and overlaid on a modern map of Europe. Toledo is located at lat/long ( $+39.86^{\circ}\text{N}$ ,  $-4.03^{\circ}\text{W}$ ). Rome is located at ( $+41.89^{\circ}\text{N}$ ,  $+12.5^{\circ}\text{W}$ ), both shown by markers on the map. This image makes clear what van Langren wished to communicate: the wide variability of the estimates, but also shows how far the estimates were biased.

Aus: Friendly, Michael et al. "The First (Known) Statistical Graph: Michael Florent van Langren and the "Secret" of Longitude." *The American Statistician* 64 (2010): 174 - 184. <https://www.datavis.ca/papers/langren-TAS09154.pdf>

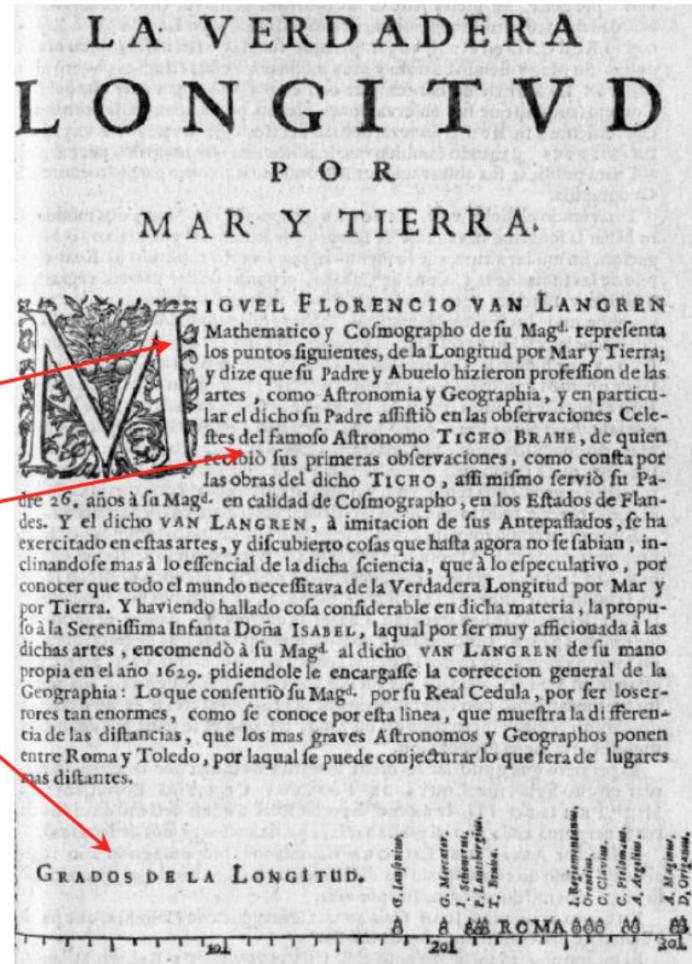
## What was he thinking?

### The first graph in context

From van Langren (1644), *The Truth about Longitude for Sea and Land.*

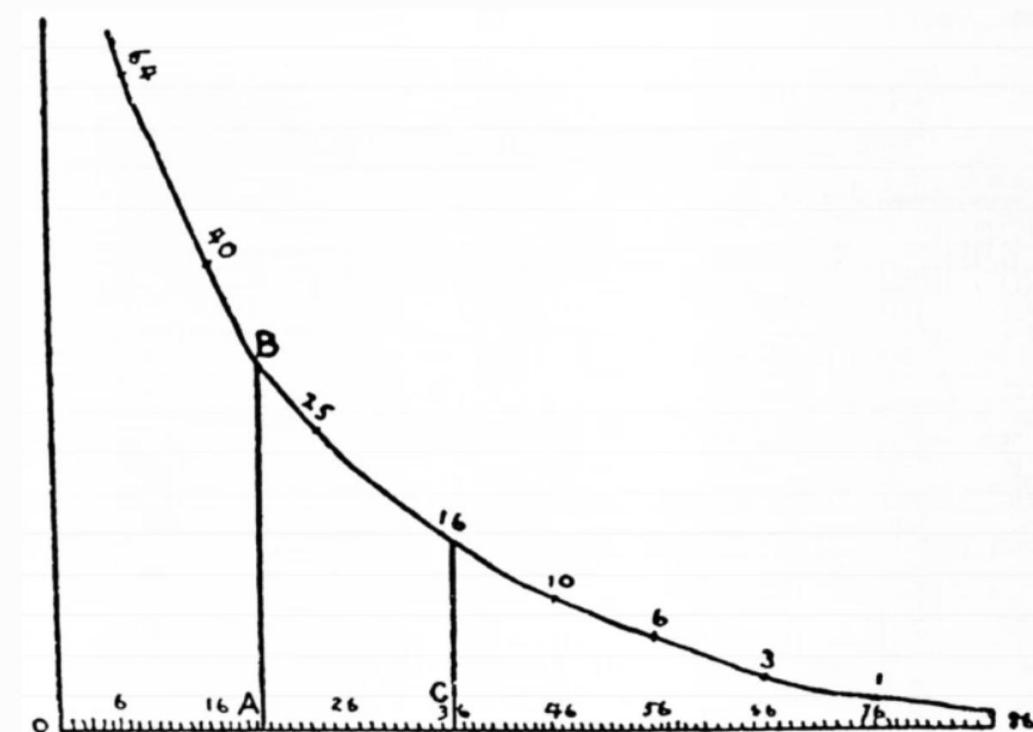
#### Patronage:

- **Credentials:** I am your chief mathematician & cosmographer
- **Problem:** Navigation at sea is most important problem for you to prosper. Many others have studied this, without success.
- **Demonstration:** I show the great errors from all previous scholars.
- **Supplication:** I have a solution, if you will grant me the magnificent awards you have given to others, less worthy than I am.



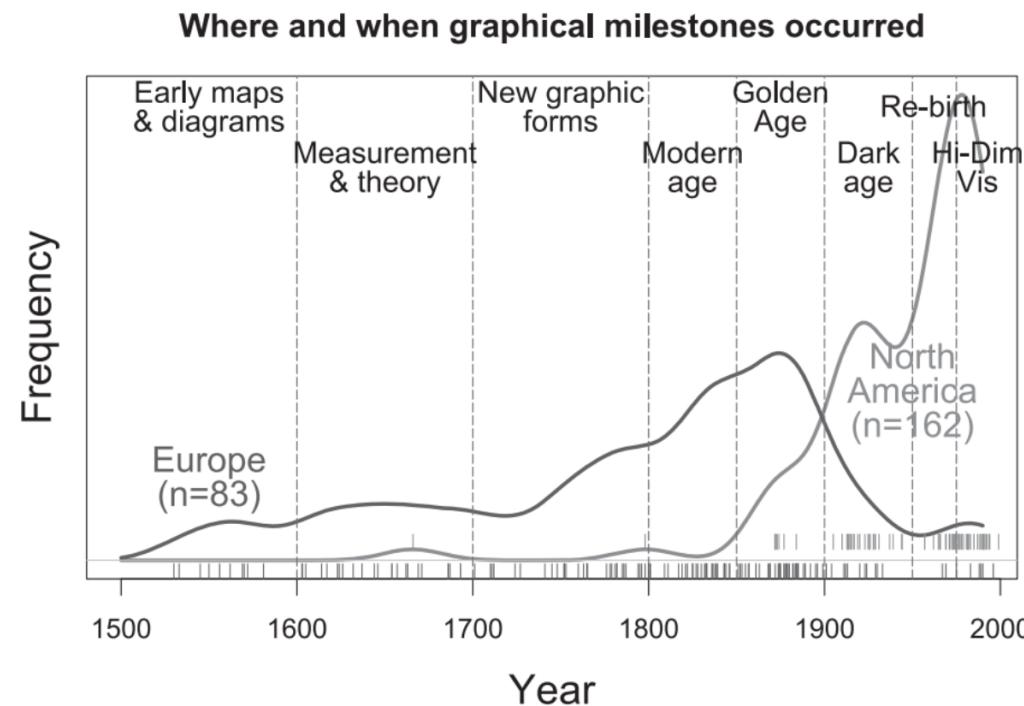
aus [2]

## Christiaan Huygens (1629-1695): Erste quantitative Visualisierung



Christian Huygens's 1669 curve showing how many people out of a 100 survive between the ages of infancy and 86. (The data are taken from John Graunt's *Natural and Political Observations on the Bills of Mortality, 1662*)

Wainer, H., Friendly, M. (2022). On the Origins of Data Visualization. In: Carriquiry, A.L., Tanur, J.M., Eddy, W.F. (eds) Statistics in the Public Interest. Springer Series in the Data Sciences. Springer, Cham. [https://doi.org/10.1007/978-3-030-75460-0\\_27](https://doi.org/10.1007/978-3-030-75460-0_27)

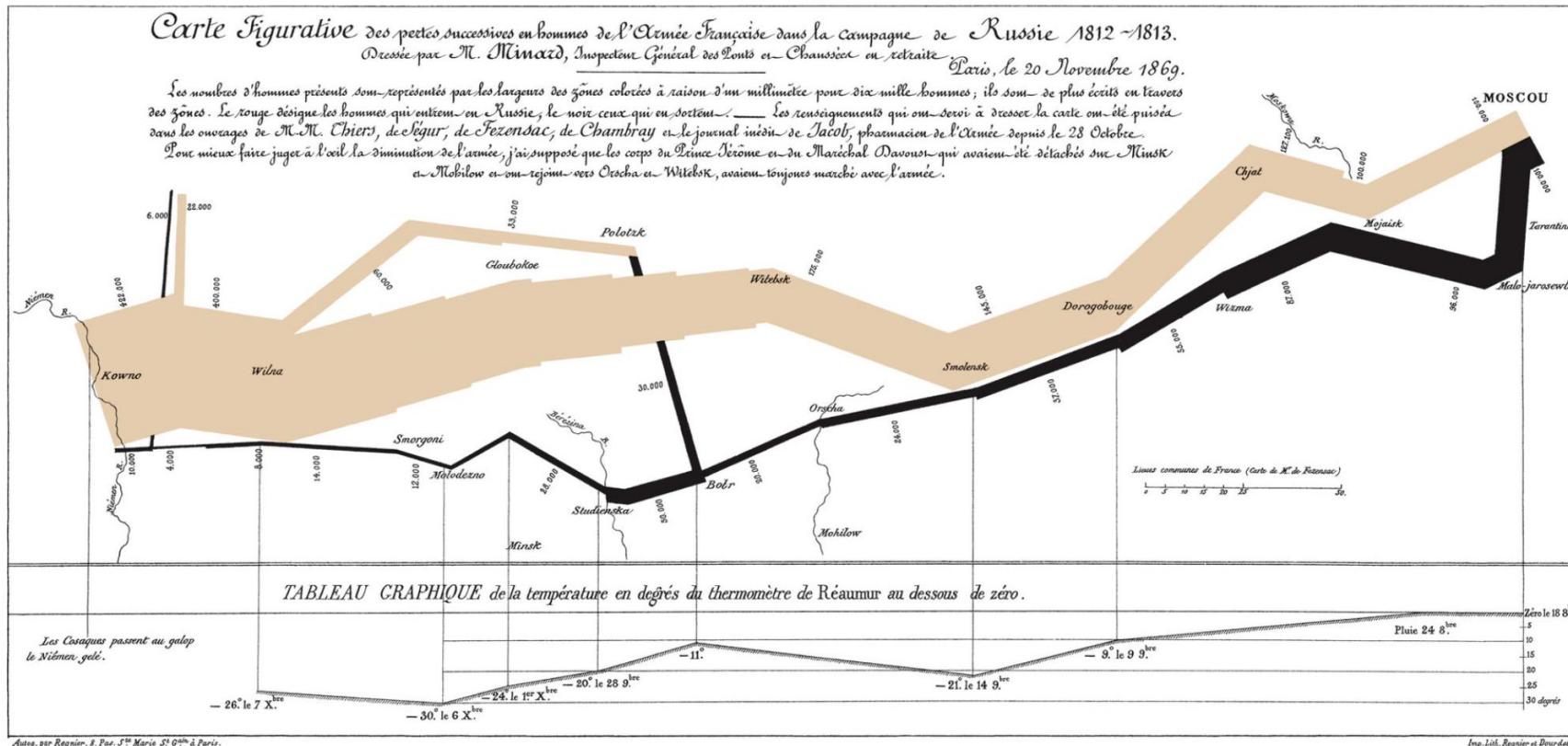


I.1 Time line of milestone events: Classified by place of development. Tick marks at the bottom show individual events. The smoothed curves plot their relative frequency, in Europe and North America. Source: © The Authors.

M. Friendly, H. Wainer: A History of Data Visualization and Graphic Communication, Harvard University Press, 2021

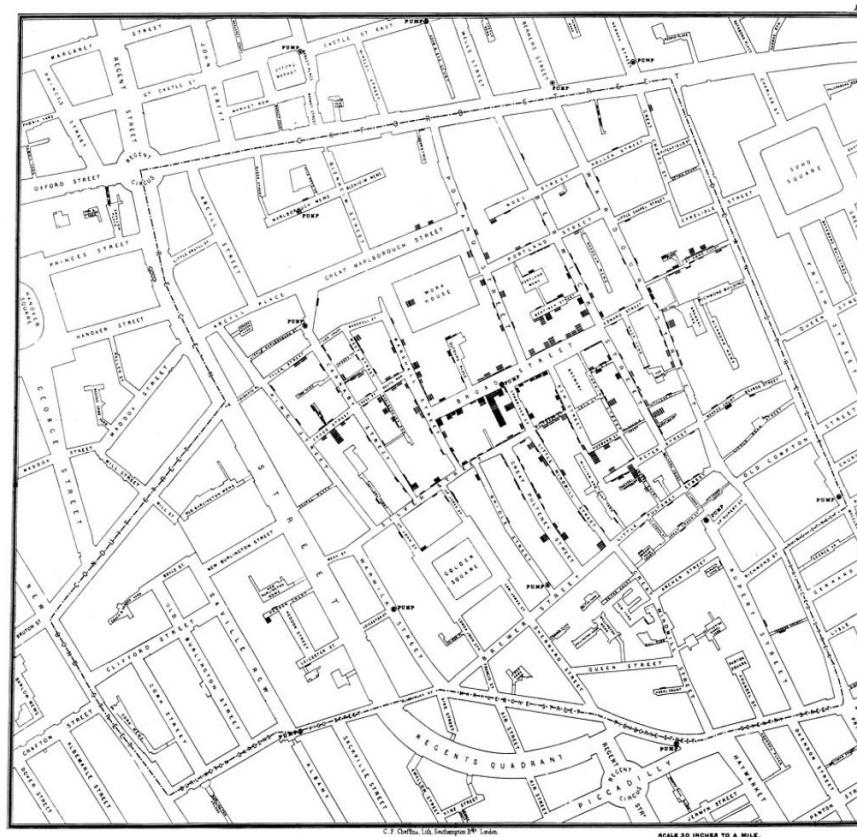
XXXXXXXXXXXXXXXXXXXXXX

## Charles Minard (1781-1870): "the best statistical graphic ever produced" (E. Tufte)



[https://en.wikipedia.org/wiki/Charles\\_Joseph\\_Minard](https://en.wikipedia.org/wiki/Charles_Joseph_Minard)

John Snow (1813-1858): "erste Root Cause Analyse" (Ursachensuche bei einem Cholera-Ausbruch 1854 in London)



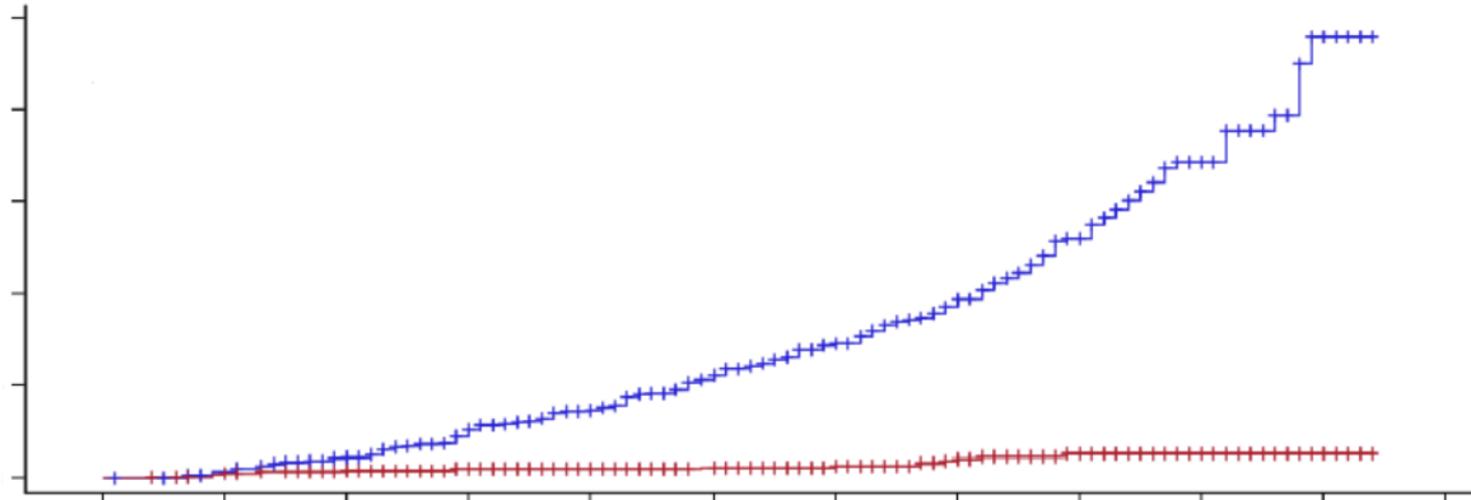
[https://en.wikipedia.org/wiki/John\\_Snow](https://en.wikipedia.org/wiki/John_Snow)



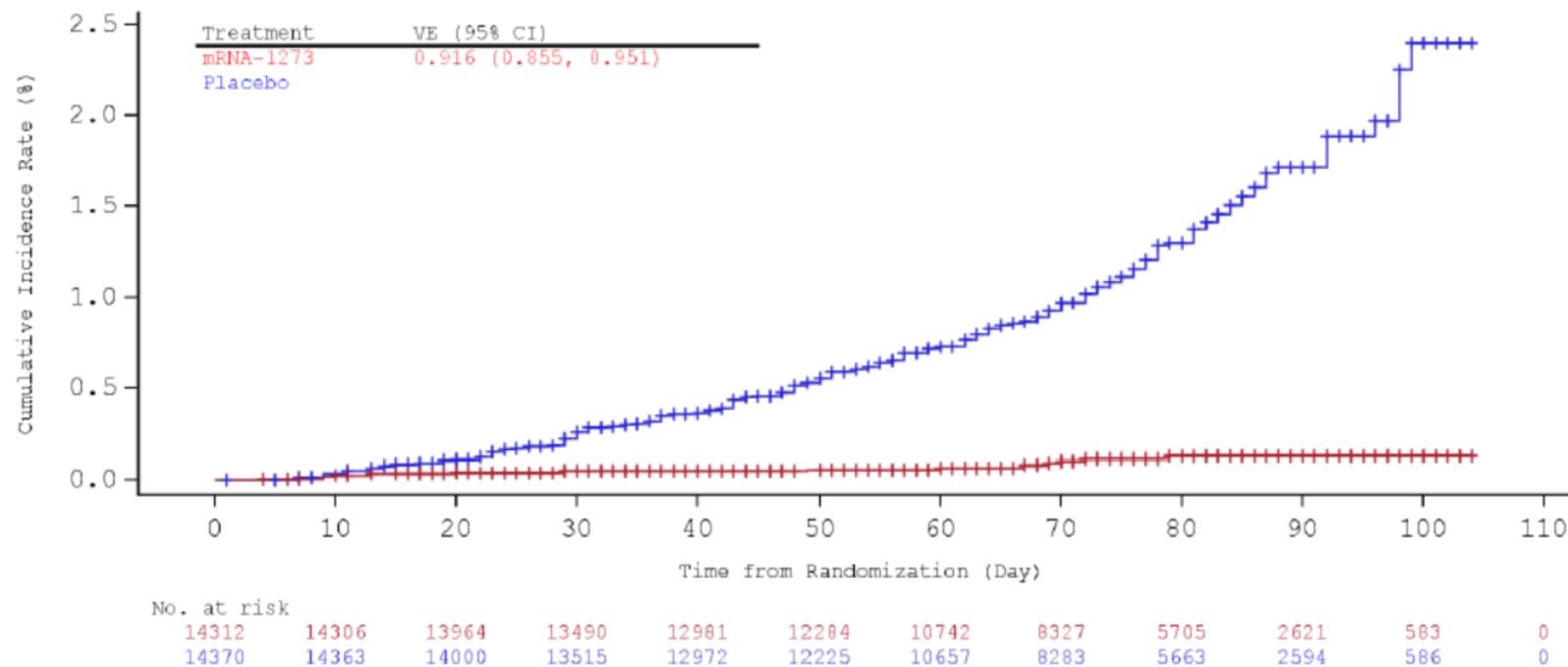


Image from "Toronto Flight Lines" (<http://www.biodiaspora.com/>) created by Bio.Diaspora 2012

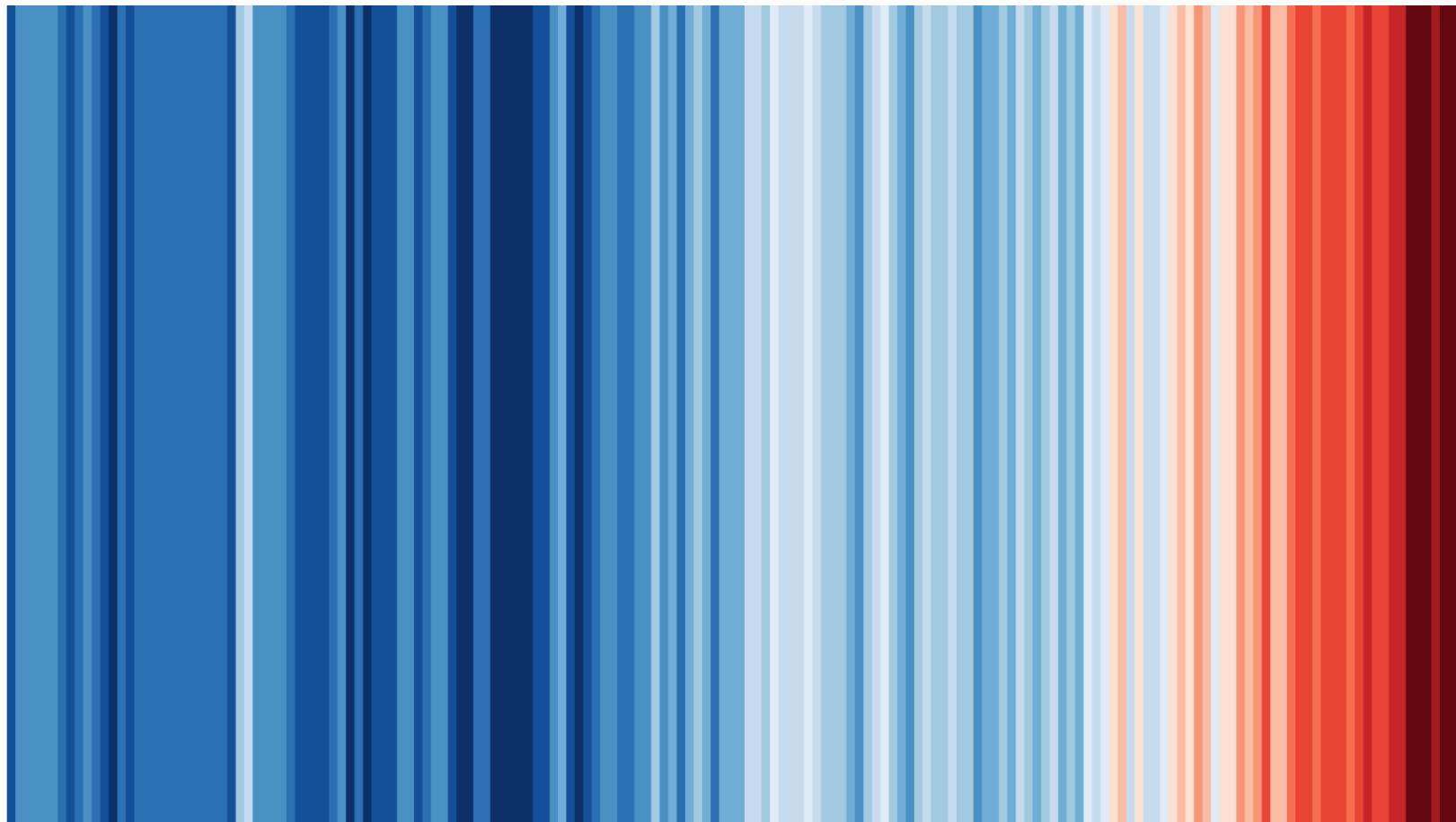
Aus: Andy Kirk. Data Visualization: A Successful Design Process. Packt Publishing, 2012. S. 43

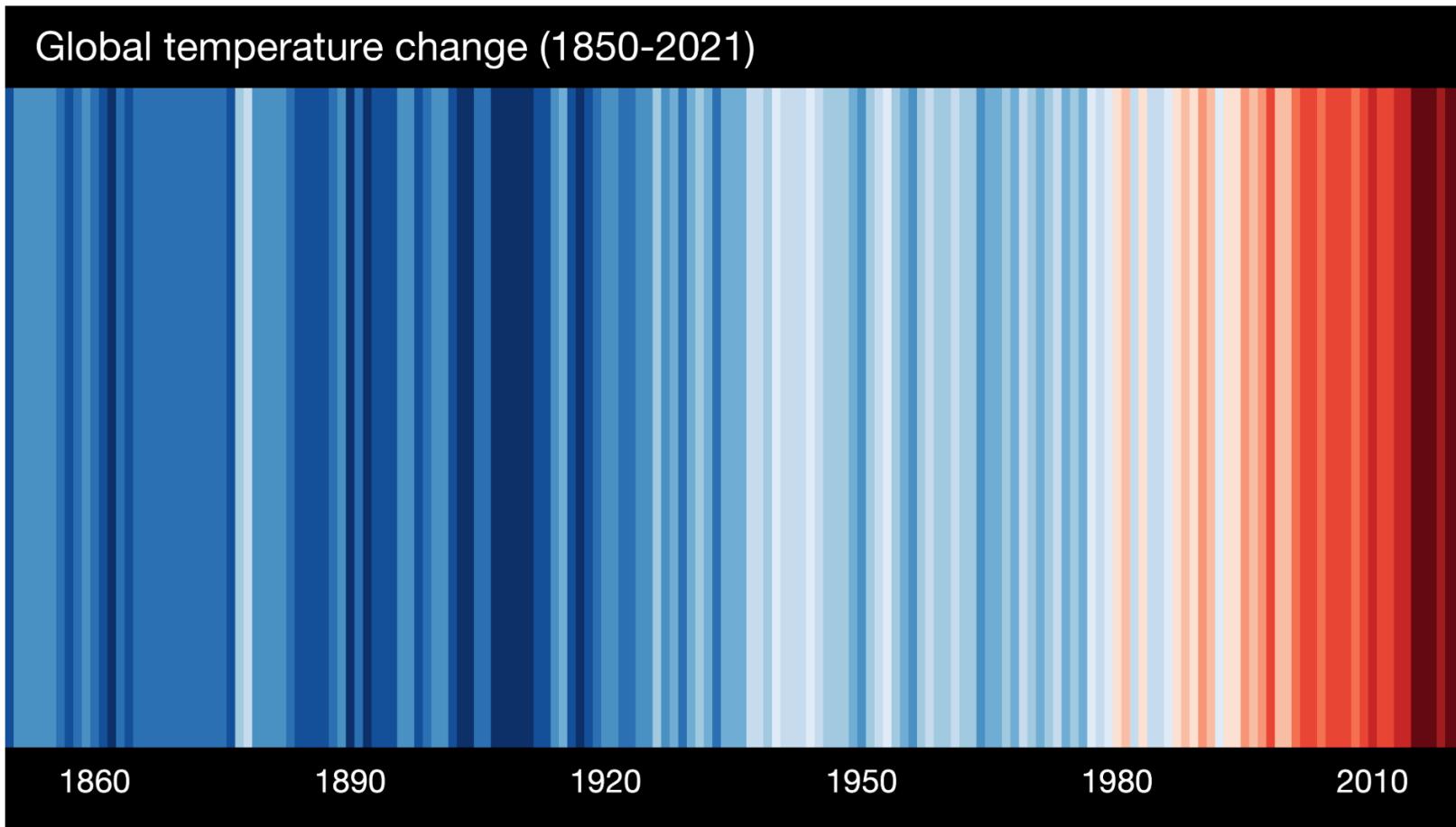


**Figure 2. Cumulative Incidence Curves for the First COVID-19 Occurrence After Randomization, mITT Set**



Moderna-Zulassungsstudie ihres COVID-19-Impfstoffs: <https://www.fda.gov/media/144434/download>, S. 28





Ed Hawkins, University of Reading: <https://showyourstripes.info/l/globe>

### Gute Quellen:

- [1] M. Friendly, H. Wainer: A History of Data Visualization and Graphic Communication, Harvard University Press, 2021
- [2] M. Friendly: A gleam in the mind's eye. Stories and lessond from the history of data visualization. (Talk given 2021)  
<https://www.datavis.ca/papers/SSC2021-talk.pdf>

## 2.2 Theoretische Grundlagen

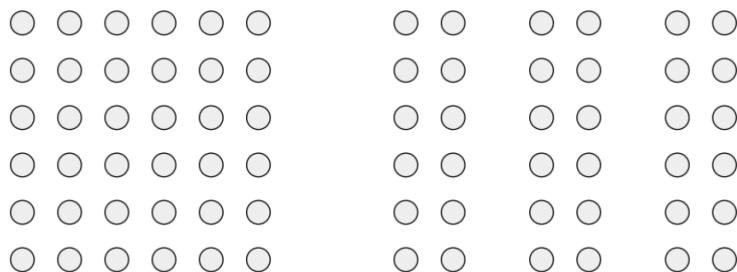
Einige Meilensteine:

- Max Wertheimer, Kurt Koffka, Wolfgang Kohler (ab 1922): Gestaltpsychologie (*Gestalt Laws*)
- Jaques Bertin: Autor von *Sémiologie graphique / Semiology of Graphics* (1967 / 1983)
- John Tukey: Autor von *Exploratory Data Analysis*, 1977 (Erfinder des Worts "Bit")
- Edward Tufte: Autor von z.B. *The Visual Display of Quantitative Information*, 1982 (Erfinder des Worts "Chartjunk")
- Leland Wilkinson: Autor von *The Grammar of Graphics*, 1999 (Grundlage von modernen Grafikpaketen wie ggplot2)

*Most principles of design should be greeted with some skepticism... we may come to see only through the lenses of word authority rather than with our own eyes.*

E. Tufte: The Visual Display of Quantitative Information

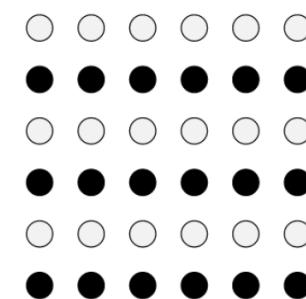
### Gestalt Laws: Gesetz der Nähe



[https://de.wikipedia.org/wiki/Datei:Gestalt\\_proximity.svg](https://de.wikipedia.org/wiki/Datei:Gestalt_proximity.svg)

Elemente mit geringen Abständen zueinander werden als zusammengehörig wahrgenommen.

### Gestalt Laws: Gesetz der Ähnlichkeit



[https://de.wikipedia.org/wiki/Datei:Gestalt\\_similarity.svg](https://de.wikipedia.org/wiki/Datei:Gestalt_similarity.svg)

Einander ähnliche Elemente werden eher als zusammengehörig erlebt als einander unähnliche.

## Aufgabe: Welches Gesetz ist rechts verletzt?

### Gestalt Laws: Gesetz der Nähe



[https://de.wikipedia.org/wiki/Datei:Gestalt\\_proximity.svg](https://de.wikipedia.org/wiki/Datei:Gestalt_proximity.svg)

Elemente mit geringen Abständen zueinander werden als zusammengehörig wahrgenommen.

### Gestalt Laws: Gesetz der Nähe

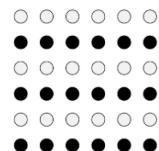


[https://de.wikipedia.org/wiki/Datei:Gestalt\\_proximity.svg](https://de.wikipedia.org/wiki/Datei:Gestalt_proximity.svg)

Elemente mit geringen Abständen zueinander werden als zusammengehörig wahrgenommen.

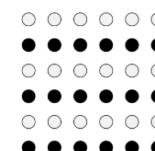
### Gestalt Laws: Gesetz der Ähnlichkeit

#### Gestalt Laws: Gesetz der Ähnlichkeit



[https://de.wikipedia.org/wiki/Datei:Gestalt\\_similarity.svg](https://de.wikipedia.org/wiki/Datei:Gestalt_similarity.svg)

Einander ähnliche Elemente werden eher als zusammengehörig erlebt als einander unähnliche.



[https://de.wikipedia.org/wiki/Datei:Gestalt\\_similarity.svg](https://de.wikipedia.org/wiki/Datei:Gestalt_similarity.svg)

Einander ähnliche Elemente werden eher als zusammengehörig erlebt als einander unähnliche.

## Aufgabe: Welches Gesetz ist rechts verletzt?

### Gestalt Laws: Gesetz der Nähe



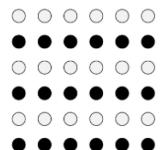
[https://de.wikipedia.org/wiki/Datei:Gestalt\\_proximity.svg](https://de.wikipedia.org/wiki/Datei:Gestalt_proximity.svg)  
Elemente mit geringen Abständen  
zueinander werden als  
zusammengehörig wahrgenommen.

### Gestalt Laws: Gesetz der Nähe



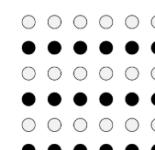
[https://de.wikipedia.org/wiki/Datei:Gestalt\\_proximity.svg](https://de.wikipedia.org/wiki/Datei:Gestalt_proximity.svg)  
Elemente mit geringen Abständen  
zueinander werden als  
zusammengehörig wahrgenommen.

### Gestalt Laws: Gesetz der Ähnlichkeit



[https://de.wikipedia.org/wiki/Datei:Gestalt\\_similarity.svg](https://de.wikipedia.org/wiki/Datei:Gestalt_similarity.svg)  
Einander ähnliche Elemente werden  
eher als zusammengehörig erlebt  
als einander unähnliche.

### Gestalt Laws: Gesetz der Ähnlichkeit

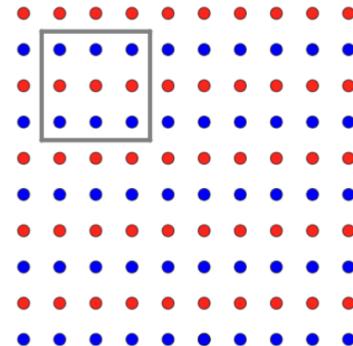


[https://de.wikipedia.org/wiki/Datei:Gestalt\\_similarity.svg](https://de.wikipedia.org/wiki/Datei:Gestalt_similarity.svg)  
Einander ähnliche Elemente werden  
eher als zusammengehörig erlebt  
als einander unähnliche.

Es gibt weitere *Gestalt Laws*, die auf die 20er-Jahre zurückgehen. Zusätzliche Gesetze von B. Palmer 1999, u.a.:

Stephen E. Palmer: Vision Science. MIT Press, Cambridge (USA) 1999

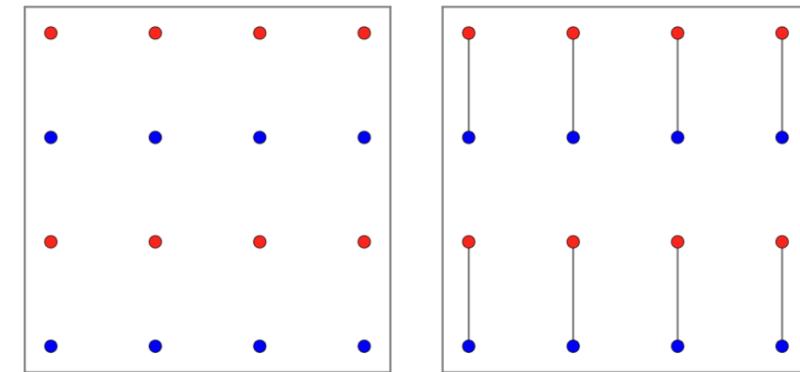
### Gesetz der gemeinsamen Region:



[https://de.wikipedia.org/wiki/Datei:Gestaltpsychologie\\_Gemeinsame\\_Region.svg](https://de.wikipedia.org/wiki/Datei:Gestaltpsychologie_Gemeinsame_Region.svg)

Elemente in abgegrenzten Gebieten werden als zusammengehörig empfunden.

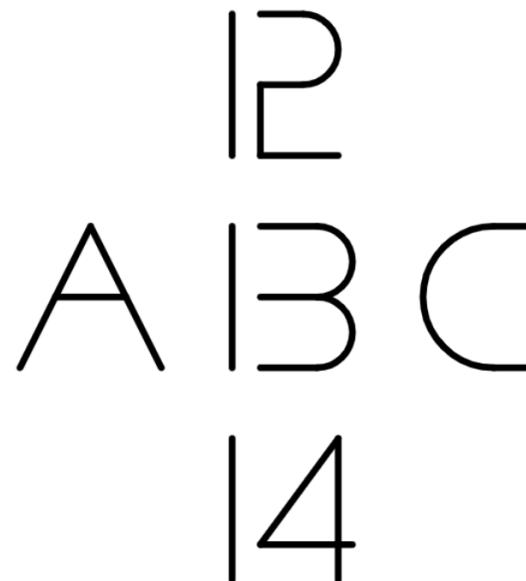
### Gesetz der verbundenen Elemente:



[https://de.wikipedia.org/wiki/Datei:Gestaltpsychologie\\_Verbundene\\_Elemente.svg](https://de.wikipedia.org/wiki/Datei:Gestaltpsychologie_Verbundene_Elemente.svg)

Verbundene Elemente werden als ein Objekt empfunden.

Im Rahmen der Gestaltpsychologie wird auch **Kontextsensitivität** genannt:  
Die Wahrnehmung einzelner Teile wird von der ganzheitlichen Wahrnehmung beeinflusst.



<https://upload.wikimedia.org/wikipedia/commons/6/65/ABC121314.svg>

Untersuchung von graphischen Elementen (nach MacKinlay, in der Tradition von Jaques Bertins):

	Quantitativ	Ordinal	Nominal
besser geeignet			
Position	••	Position	••
Länge	==	Farbdichte	•••
Winkel	<	Sättigung	•••
Neigung	/\	Farbe	•••
Fläche	••●	Länge	==
Farbdichte	•••	Winkel	<
Sättigung	•••	Neigung	/\
Farbe	•••	Fläche	••●
Form	•▲■	Form	•▲■
weniger geeignet			

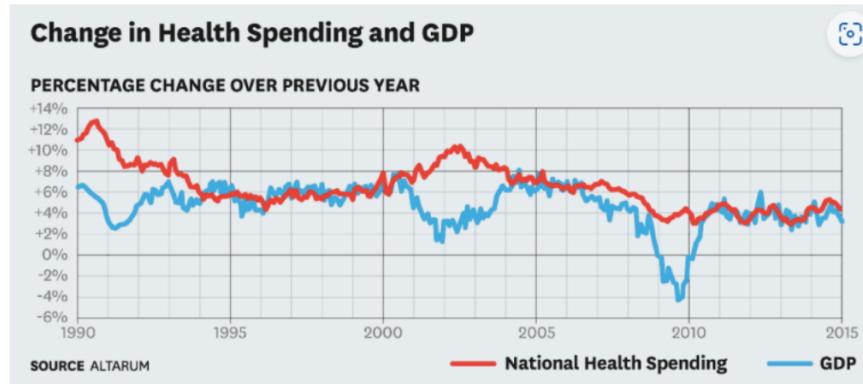
Aus: Nazemi, Kawa & Kaupp, Lukas & Burkhardt, Dirk & Below, Nicola. (2021). Datenvizualisierung.  
Adaptiert nach MacKinlay, Jock. 1986. „Automating the design of graphical presentations of relational information.“ ACM Trans. Graph. 5 (2): 110–41. doi:10.1145/22949.22950.

## 2.3 Vorher/Nachher-Vergleiche

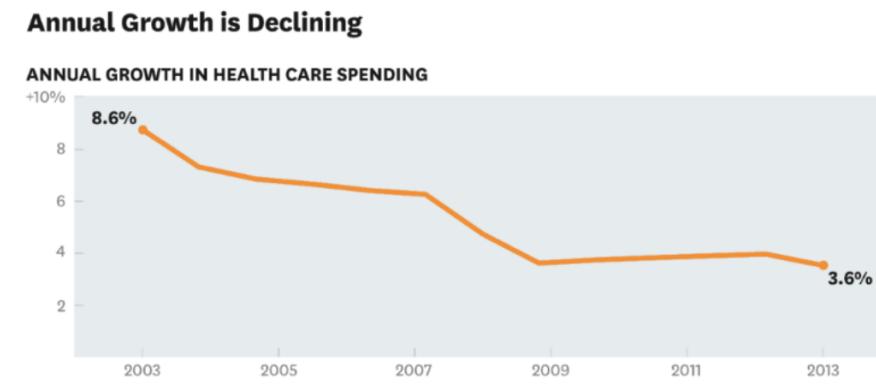
**Beispiel:** Sollten wir die Gesundheitsausgaben erhöhen?

Nach: Scott Berinato: Good Charts: The HBR Guide to Making Smarter, More Persuasive Data Visualizations. Harvard Business Press Books, 2016.  
Siehe auch: <https://hbr.org/2016/06/visualizations-that-really-work>

Erster Versuch:



Zweiter Versuch:



"Look at all the data I have and the work I've done!"

**Gut:** klare Message (in Bild und Text!)

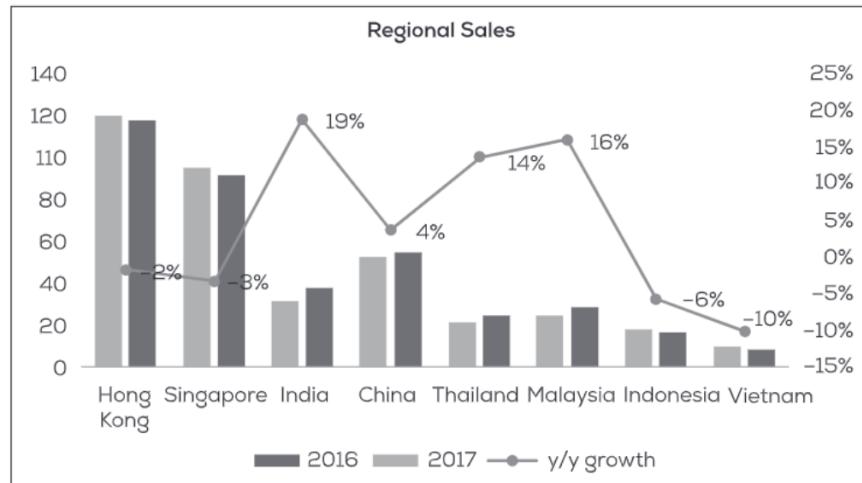
**Gut:** auf einen Blick intuitiv begreifbar

**Diskutabel:** der Ausschnitt ist suggestiv (=manipulativ)

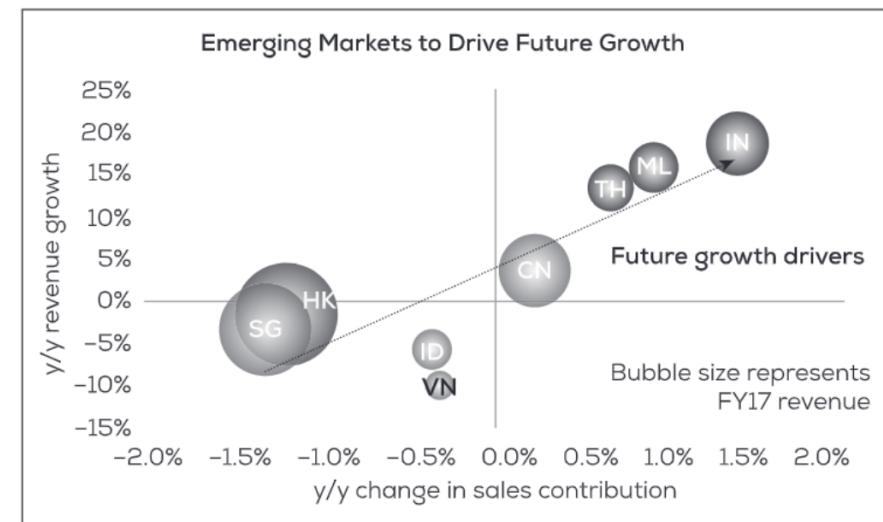
## Beispiel: Wo soll unsere Firma investieren?

Sejal Vora. The Power of Data Storytelling. Sage Publications Pvt. Ltd, 2019.

**Figure 2.3** What Most People Do?



**Figure 2.5** The Result When Data Storytelling Principles Are Applied



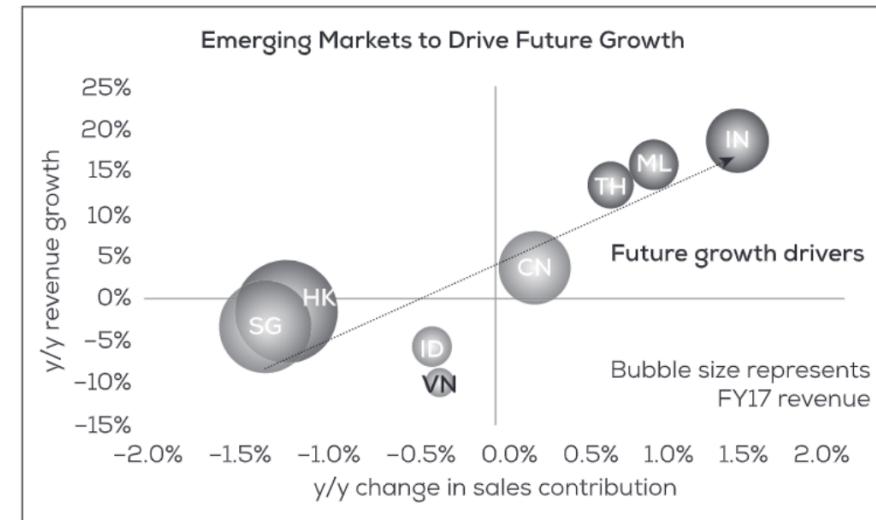
"unkommentierter" Überblick über die Daten.  
(Guter erster Schritt!)

**Gut:** Eine Message wird aktiv herausgearbeitet.  
**Diskutabel:** Die Grafik ist immer noch recht komplex.

Besser, wenn die Grafik noch erklärt wird:

- Hong Kong and Singapore continued to dominate regional sales; however, (...) [they] reduced their overall revenue contribution to 54% in 2017 (...).
- Emerging markets like India, Malaysia and Thailand on the other hand recorded a revenue growth between 14% and 19% (...) making them future growth drivers for the company's revenue generation.
- China maintained its significance with a marginal 4% revenue growth, while Indonesia and Vietnam performed poorly (...).

**Figure 2.5** The Result When Data Storytelling Principles Are Applied



### Zusammenfassung:

- Zielgruppe festlegen
- Kernaussage 1. für sich und 2. visuell herausarbeiten
- vereinfachen und Unnötiges weglassen
- Sehgewohnheiten (z.B. Gestalt Laws) beachten
- bei aller Vereinfachung: nicht lügen!

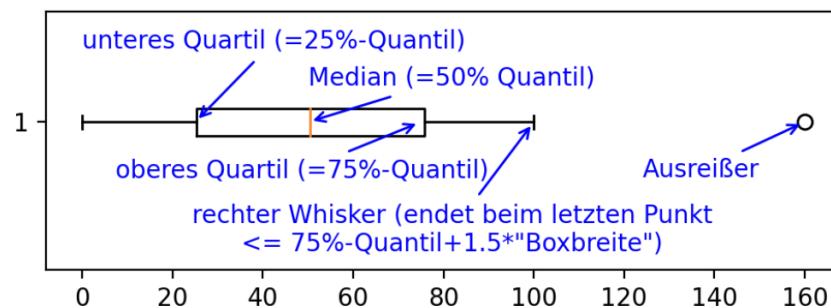
P  
A B C  
I4

## 1. Lagemaße

## 2. Informationsvisualisierung

## 3. Streumaße

### 3.1 Wiederholung: Inter Quartile Range (IQR)



Der **Inter Quartile Range (IQR)** für einen eindimensionalen Datenvektor  $x$  ist definiert als der Abstand zwischen dem oberen und dem unteren Quartil:

$$\text{IQR}(x) = Q_{0.75}(x) - Q_{0.25}(x)$$

Hierbei bezeichnet  $Q_q(x)$  das  $q$ -Quantil von  $x$ .

(Das Quartil wird in 25%-Schritten definiert: 1. Quartil = 25%-Quantil, 2. Quartil=50% Quantil, ...)

Der IQR ist robust gegenüber Outlier:

```
In [102]: ┌─ def iqr(x): return np.quantile(x,q=0.75)-np.quantile(x,q=0.25)      # Alternative: scipy.stats.iqr
          └─ iqr( np.arange(0,101) ), iqr( np.arange(0,100).tolist()+[10000] )
```

Out[102]: (50.0, 50.0)

**Best practice:** Der IQR beschreibt gut die "Größe der Punktwolke"  $x$  (d.h. ist ein gutes Streumaß).

## 3.2 (empirische) Varianz und Standardabweichung

Es sei ein Datenvektor  $x = (x_1, \dots, x_n) \in \mathbb{R}^n$  gegeben ( $n \geq 2$ ). Erinnerung:  $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$ .

- Die (empirische) **Varianz** von  $x$  ist definiert via

$$\text{var}(x) = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

(Die Normierung mit  $n-1$  ist weitgehend akzeptiert, vereinzelt findet man auch die ("nicht erwartungstreue") Normierung mit  $n$ .)

- Die (empirische) **Standardabweichung** von  $x$  ist definiert via

$$\text{sd}(x) = \sqrt{\text{var}(x)} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$$

```
In [115]: # Was ist hier passiert? x ist ein np.array
def var(x): return ( (x-np.mean(x))**2 ).sum() / ( len(x)-1 )

var ( np.arange(100) ), np.var ( np.arange(100) )
```

Out[115]: (841.666666666666, 833.25)

```
In [119]: # Standardmäßig teilt numpy durch n statt n-1 (Pandas folgt der (n-1)-Konvention)
var ( np.arange(100) ), np.var ( np.arange(100), ddof=1 ), pd.Series ( np.arange(100) ).var()
```

Out[119]: (841.666666666666, 841.666666666666, 841.666666666666)

## Verhalten bei Outliern:

```
In [124]: # Zahlen 0,1,...,99,100 (keine Outlier)  
data = np.arange(0,101)  
  
iqr(data), \  
np.std(data, ddof=1), \  
np.var(data, ddof=1)
```

Out[124]: (50.0, 29.300170647967224, 858.5)

```
In [125]: # Zahlen 0,1,...,99,10000 (ein Outlier)  
data = np.arange(0,100).tolist() + [10000]  
  
iqr(data), \  
np.std(data, ddof=1), \  
np.var(data, ddof=1)
```

Out[125]: (50.0, 990.5324525748566, 981154.5396039605)

## Ergebnisse:

- Die Standardabweichung sollte als Streumaß gegenüber der Varianz bevorzugt werden, da sie auf der gleichen Einheitenskala wie die Daten (und wie der Mittelwert) lebt.
- IQR ist robust gegenüber Outliern, Varianz und Standardabweichung nicht.

**Frage:** Welche Gründe fallen Ihnen ein, trotzdem die Standardabweichung zu benutzen?

### 3.3 Der Zentrale Grenzwertsatz

Neben der größeren Akzeptanz ("Management-tauglich") der Varianz: weitere Begründung durch den ZGWS.

**Zentraler Grenzwertsatz (Spezialfall):** Es seien  $X_1, X_2, \dots$  reellwertige Zufallszahlen (z.B. wiederholte Messungen):

$$X_i = \mu + \epsilon_i .$$

Hierbei modelliere  $\mu$  den wahren Wert und  $\epsilon_i$  ein zufälliges Rauschen. Die  $\epsilon_i$  haben Varianz  $\sigma^2$ .

Dann konvergiert\* die Verteilung der normierten summierten Abweichungen von  $\mu$  gegen eine Normalverteilung:

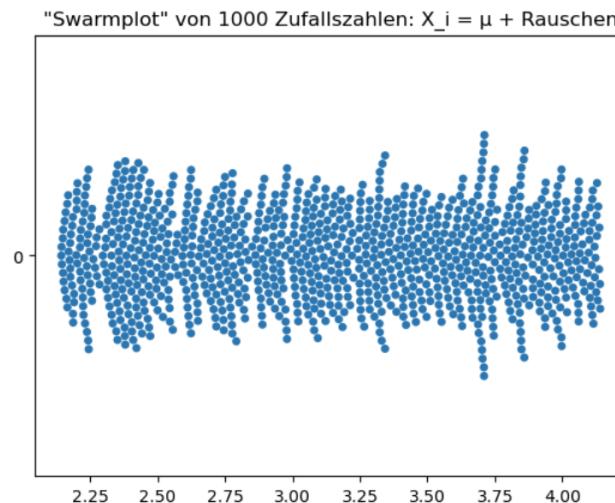
$$\frac{1}{\sqrt{n}} \sum_{i=1}^n (X_i - \mu) \xrightarrow{n \rightarrow \infty} \text{Normalverteilung } \mathcal{N}(0, \sigma^2)$$

\* unter technischen Annahmen, z.B.: Das Rauschen sei symmetrisch um 0 verteilt (d.h.  $\epsilon_1$  habe die gleiche Verteilung wie  $-\epsilon_1$ ); die einzelnen Messungen liefern unabhängige, aber identisch verteilte Ergebnisse; und es können keine beliebig hohen Rauschbeträge auftreten (d.h.  $\epsilon_1 \leq C$  für ein  $C > 0$ ).

**Interpretation:** Der Zentrale Grenzwertsatz liefert eine Begründung

- für die Wichtigkeit der Normalverteilung ("wenn man genügend viele zufällige Einflüsse aufsummiert, sieht es aus wie eine Normalverteilung, auch wenn die einzelnen Einflüsse nicht normalverteilt sind"),
- für die Standardabweichung bzw. die Varianz, da diese Größe im Limes  $n \rightarrow \infty$  erhalten bleibt.

In [163]: # Zufallszahlen  $X_1, X_2, \dots$ , wobei  $X_i = \mu + \text{Rauschen}$   
 $R = \text{np.random.default_rng(42)}$   
 $\mu = 3.141592$   
 $x = \mu + R.\text{uniform}(\text{low}=-1, \text{high}=1, \text{size}=1000)$   
 $\text{sns.swarmplot}(\text{data}=x, \text{orient}=\text{"h"})$   
 $\text{plt.title}(\text{"Swarmplot" von 1000 Zufallszahlen: } X_i = \mu + \text{Rauschen})$

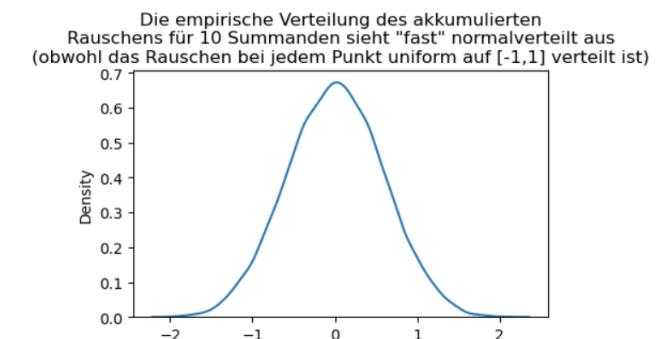


In [184]: # Betrachte den Ausdruck im ZGWS für  $n=10$ .  
# Dies ist EINE EINZIGE ZAHL, für die wir keine  
# Verteilungsaussage machen können.  
# Um eine Verteilungsaussage zu machen, müssen wir  
# das Experiment oft (hier 10 000x) wiederholen.

```
n=10
xs=[ mu + R.uniform(low=-1,high=1,size=1000)
     for repeat in range(10000) ]

zgws = [ (x-mu)[:n].sum() / np.sqrt(n) for x in xs ]

sns.kdeplot(zgws)
plt.title('Die empirische Verteilung des akkumulierten Rauschens')
plt.gcf().set_size_inches(5,3);
```



Der Zentrale Grenzwertsatz ist nicht klausurrelevant (die Aussage, dass die Varianz hier eine Rolle spielt, aber schon).

P  
A B C  
I4

1. Lagemaße

2. Informationsvisualisierung

3. Streumaße

**Vielen Dank für Ihre Aufmerksamkeit!**