



Einführung

3 Tage



Einführung in Data Analytics

4 Tage



Datenauswertung in Excel

15 Tage



Präsentation & Kommunikation

5 Tage



Grundlagen der Python-Programmierung

20 Tage



Python für Datenanalysten

23 Tage



Datenbanken & SQL

30 Tage



Datenschnittstellen

10 Tage



Statistik-Grundlagen

15 Tage



Business Analytics

5 Tage



Datenvisualisierung mit Python

15 Tage

Power BI

20 Tage

KI & Machine Learning

10 Tage

Low-Code ML (IHK)

10 Tage

Arbeitsmarkttransfer und -vorbereitung

15 Tage

Abschluss-Projekt

40 Tage

Datenvisualisierung mit Python

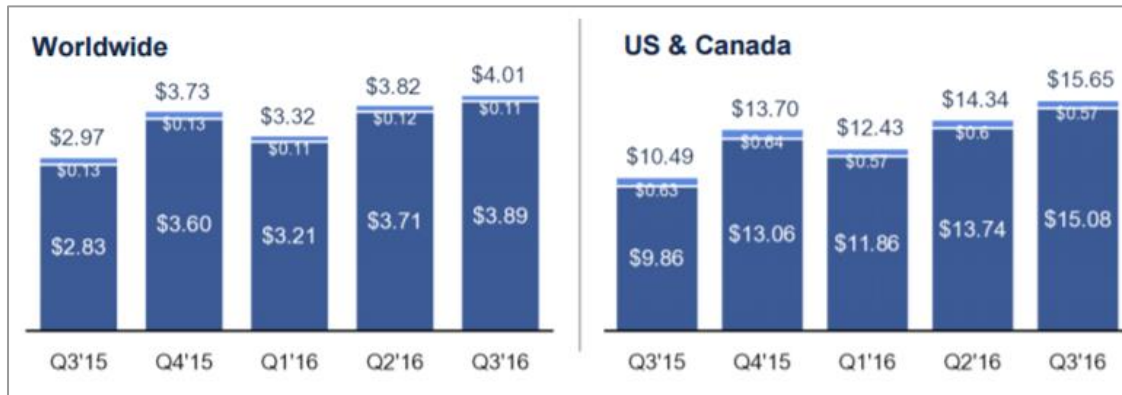
Vor Erstellung einer Präsentation ist ein gezeichnetes Storyboard mit den Überschriften bzw. Hauptinhalten hilfreich



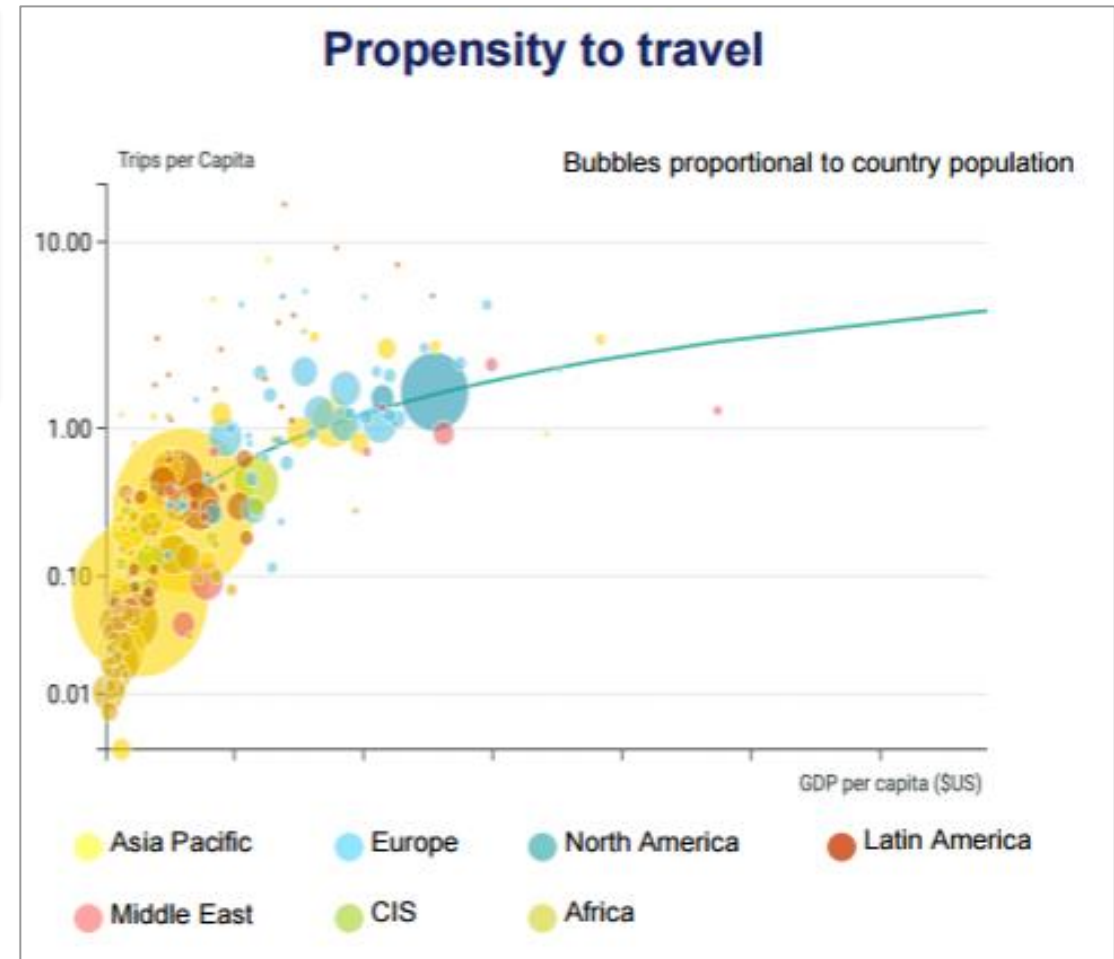
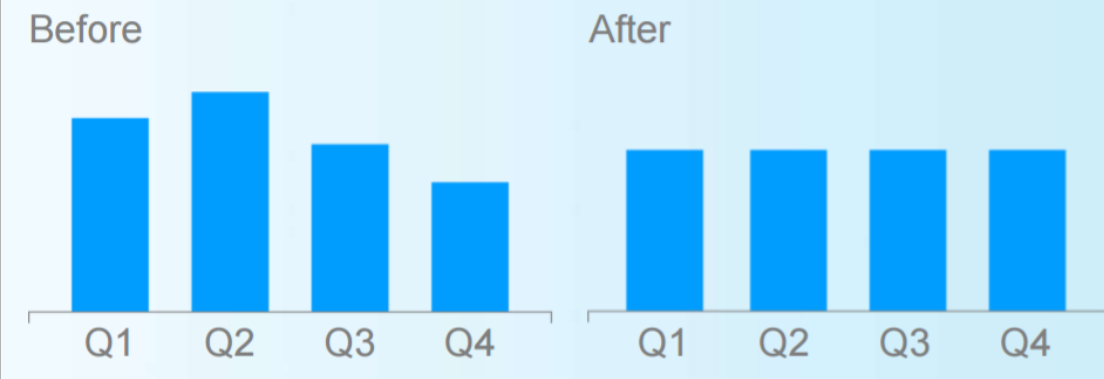
1

Gute vs. schlechte Datenvisualisierung

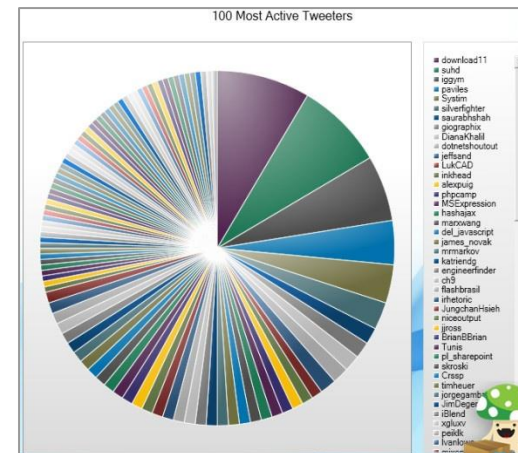
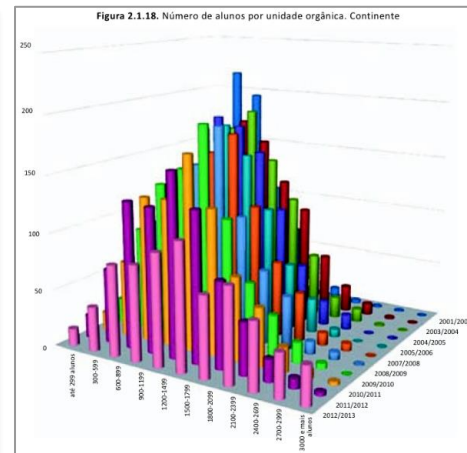
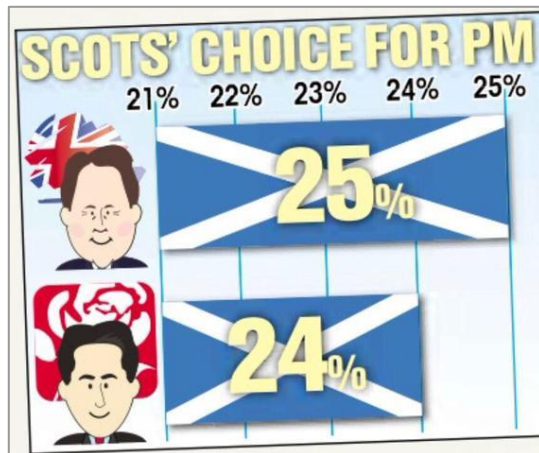
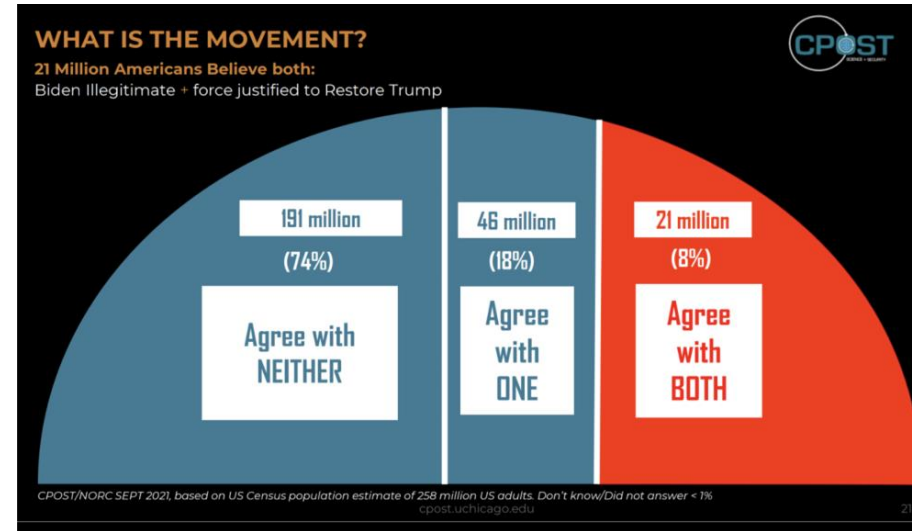
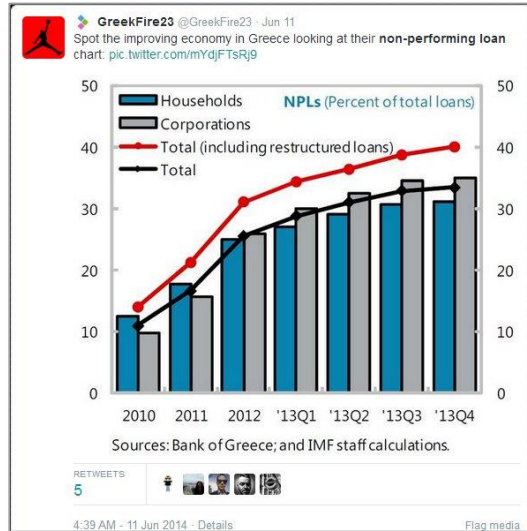
Sind diese Grafiken überzeugend?



Increasing Media Continuity



Sind diese Grafiken überzeugend?

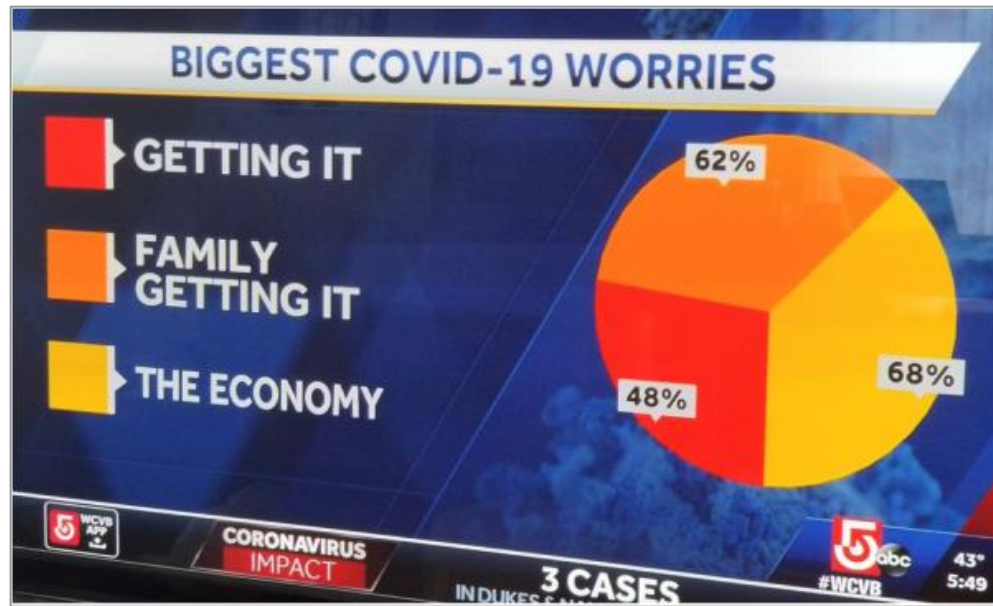


- <https://viz.wtf/>
- <https://www.consultantsmind.com/2017/04/10/bad-charts>
- <https://www.quora.com/What-are-examples-of-bad-data-visualization-thats-misleading-and-confusing>
- <https://skepchick.org/2015/05/bad-chart-thursday-uk-elections-edition>
- <https://www.businessinsider.com/the-27-worst-charts-of-all-time-2013-6>
- <https://junkcharts.typepad.com>

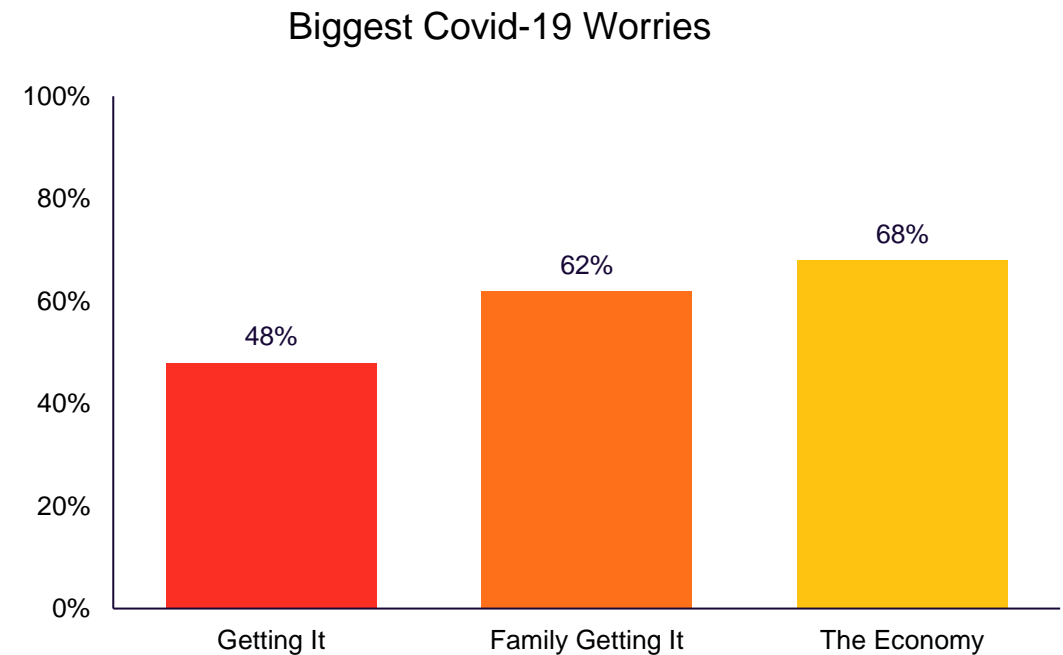
Gute Charts

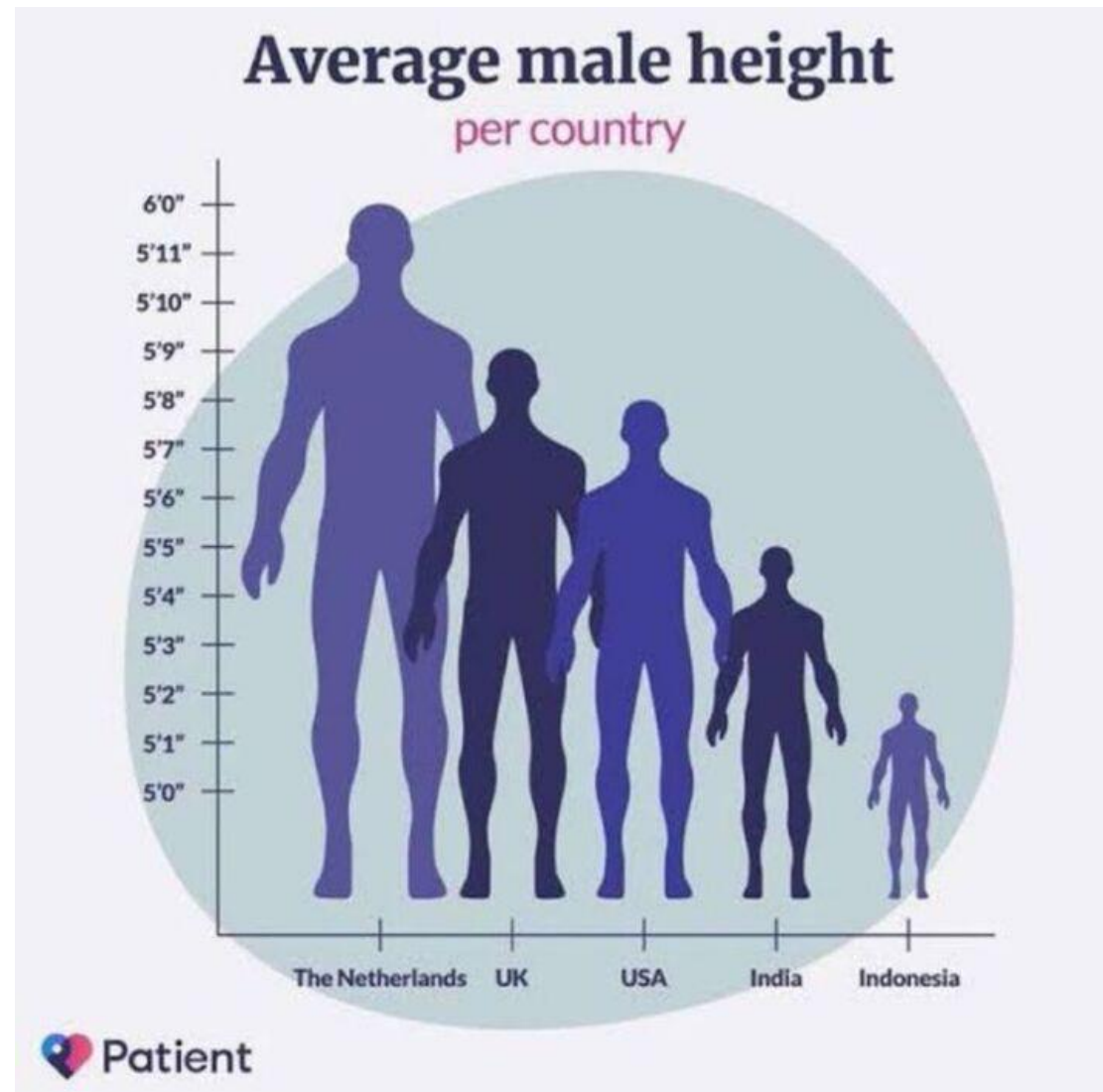
- Die Grafik passt in den Kontext und unterstützt die Aussage
- Auswahl des richtigen Charttyps
- Weniger ist mehr (Verzicht auf 3D und überflüssigen "Chart-Junk")
- Keine Tricks wie abgeschnittene Skalen
- Titel und Achsen-Beschriftungen

Auswahl des passenden Charttyps



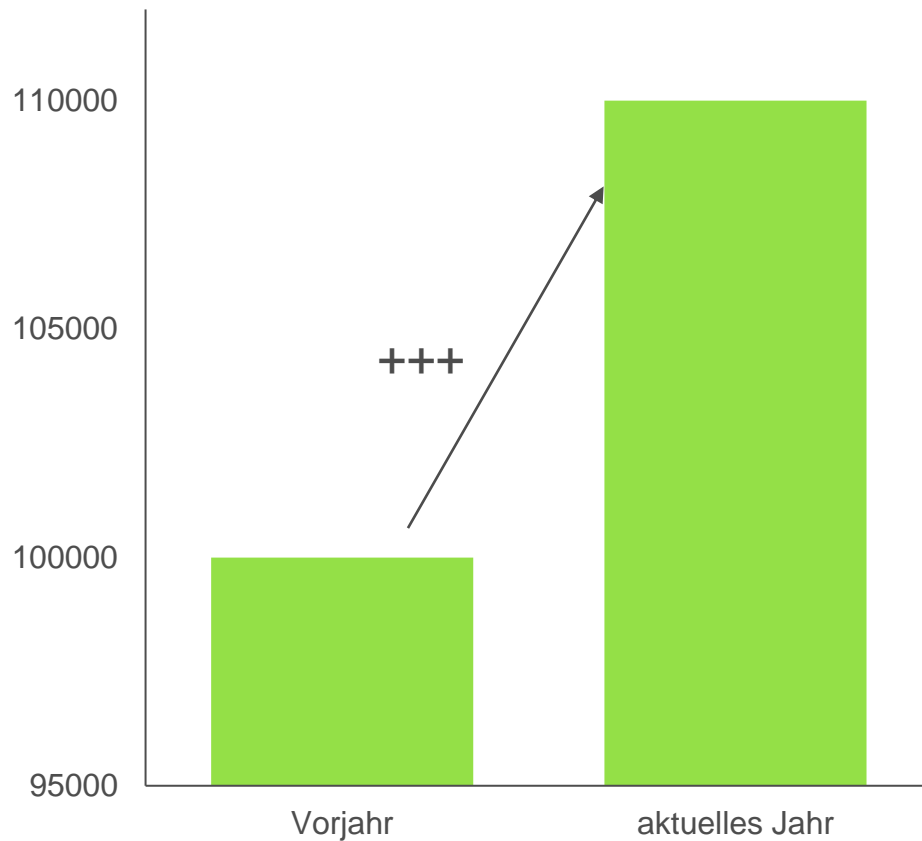
vs.



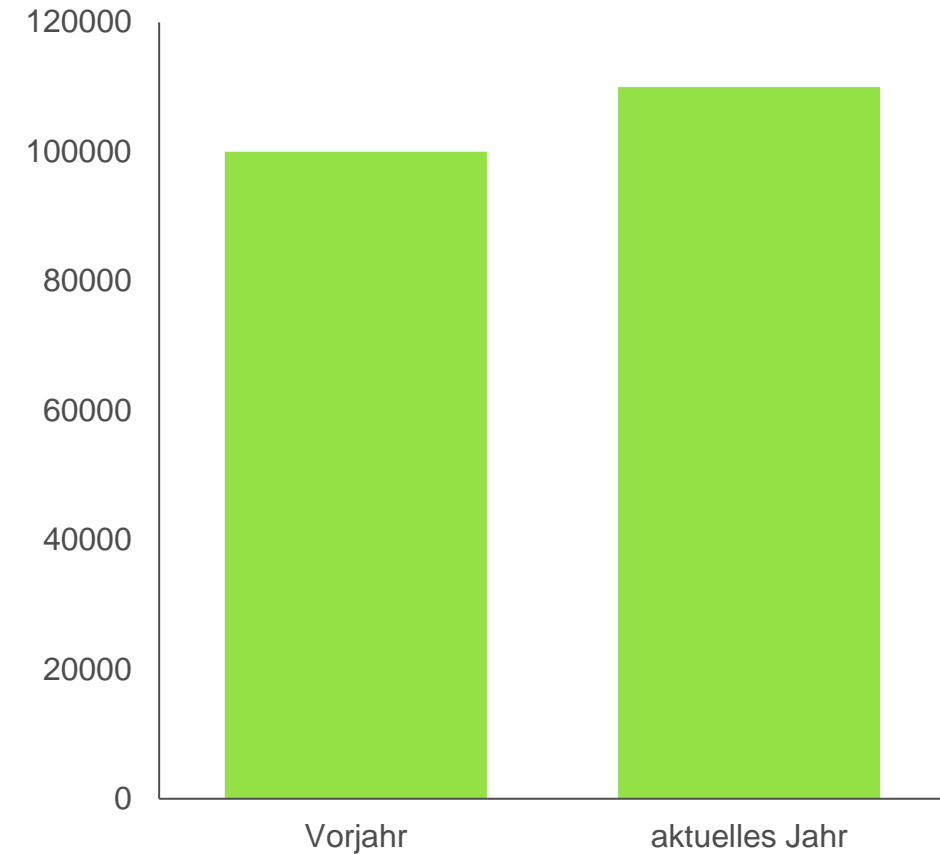


Beliebter Trick: Abgeschnittene Skalen

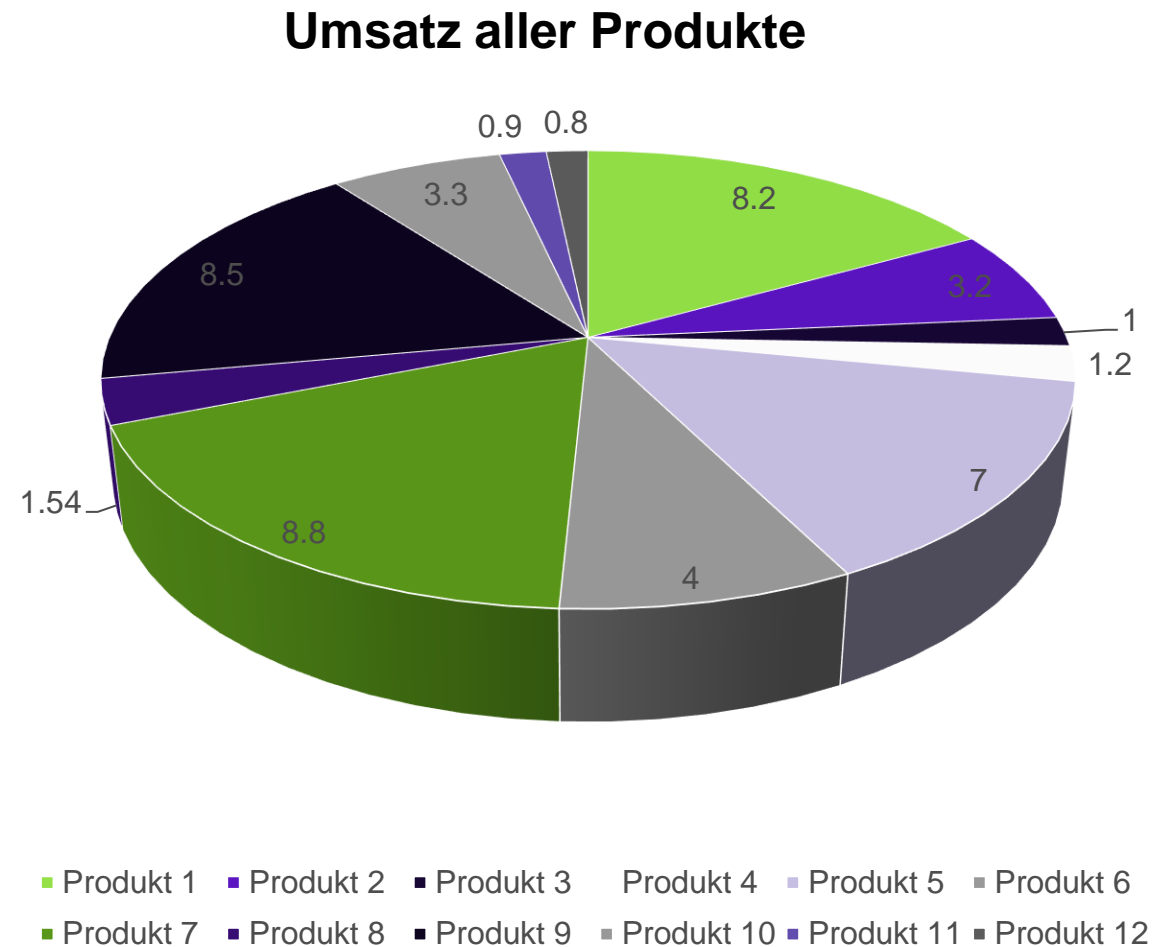
Spektakulärer Anstieg



Doch nicht so spektakulär

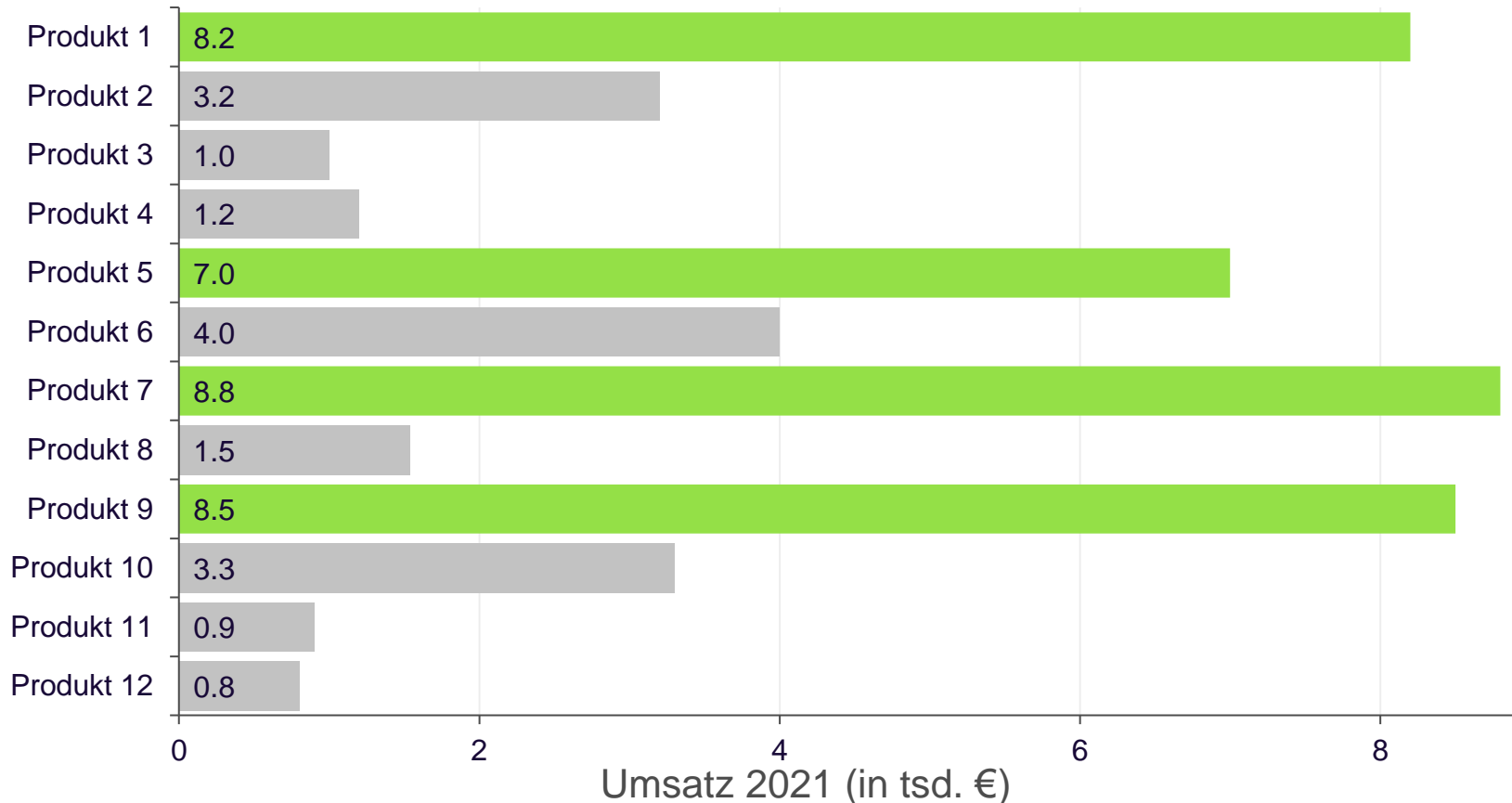


Was ist hier alles ungünstig?



Hier passen Aussage und Grafik zusammen

Wir sollten unsere Produktpalette verkleinern und uns auf die Umsatzbringer konzentrieren!

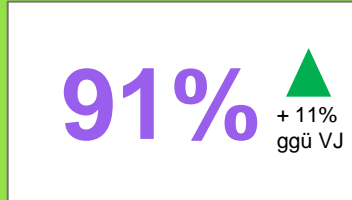


Produkte 1, 5, 7 und 9
machen 2/3 des
Umsatzes aus

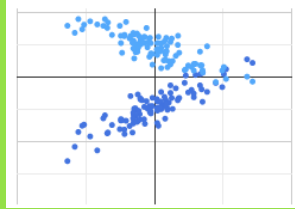
2

Visualisierungen / Charttypen

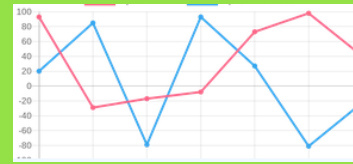
Text



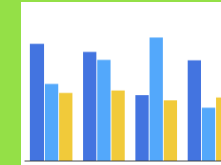
Punktwolke



Linienchart



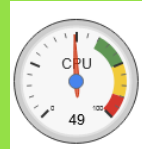
Säulen- diagramm



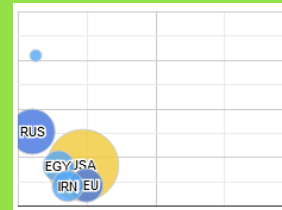
Tabelle

	Name	Salary	Full Time
1	Marie	\$24,700	✓
2	Albert	\$25,200	x
3	Enrico	\$25,700	✓
4	Lise	\$26,600	✓

Gauge / Messinstrument



Bubble Chart

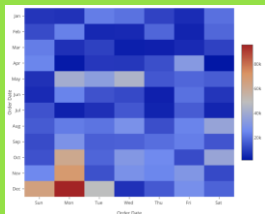


Balken- diagramm



... und
viele mehr

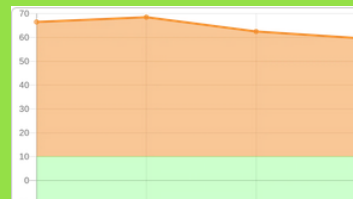
Heatmap



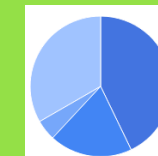
Geochart



Flächenchart



Torten-/ Donutchart



Für Python gibt es zahlreiche Grafikbibliotheken. Im Prinzip kann man die üblichen Grafiken mit all diesen Packages erstellen.

- **matplotlib**: Der Standard für Diagramme in Python (<https://www.python-graph-gallery.com>)
 - **pandas** bietet einige Diagrammtypen, die auf matplotlib basieren
 - **seaborn** setzt auf matplotlib auf und vereinfacht die Nutzung
- **plotly**: Für interaktive Charts, insbesondere zusammen mit dash, um Dashboards zu bauen
- **bokeh**: ebenfalls interaktive Charts.
- **plotnine**: Inspiriert von ggplot2 für R. Benutzt eine Grammar of Graphics
- **altair**: deklarative Grafikbibliothek, basiert auf Vega-Lite