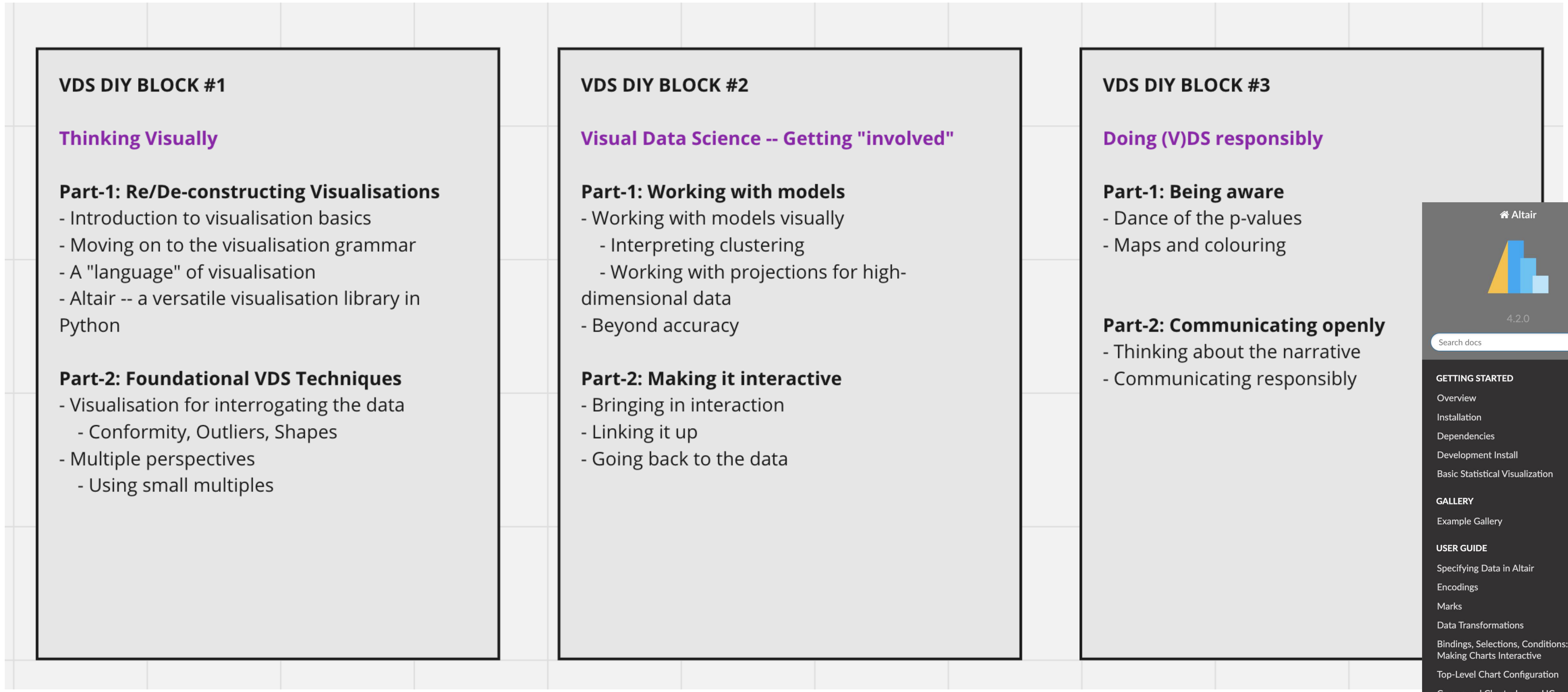
Doing Visual Data Science

Foundations, Techniques and Practice

Cagatay Turkay
Professor,
Centre for Interdisciplinary Methodologies
University of Warwick

University of Bergen, 10 August 2023

Next up .. VDS DIY



VDS DIY BLOCK #1

Thinking Visually

Part-1: Re/De-constructing Visualisations
- Introduction to visualisation basics
- Moving on to the visualisation grammar
- A "language" of visualisation
- Altair -- a versatile visualisation library in Python

Part-2: Foundational VDS Techniques
- Visualisation for interrogating the data
  - Conformity, Outliers, Shapes
- Multiple perspectives
  - Using small multiples

VDS DIY BLOCK #2

Visual Data Science -- Getting "involved"

Part-1: Working with models
- Working with models visually
  - Interpreting clustering
  - Working with projections for high-dimensional data
- Beyond accuracy

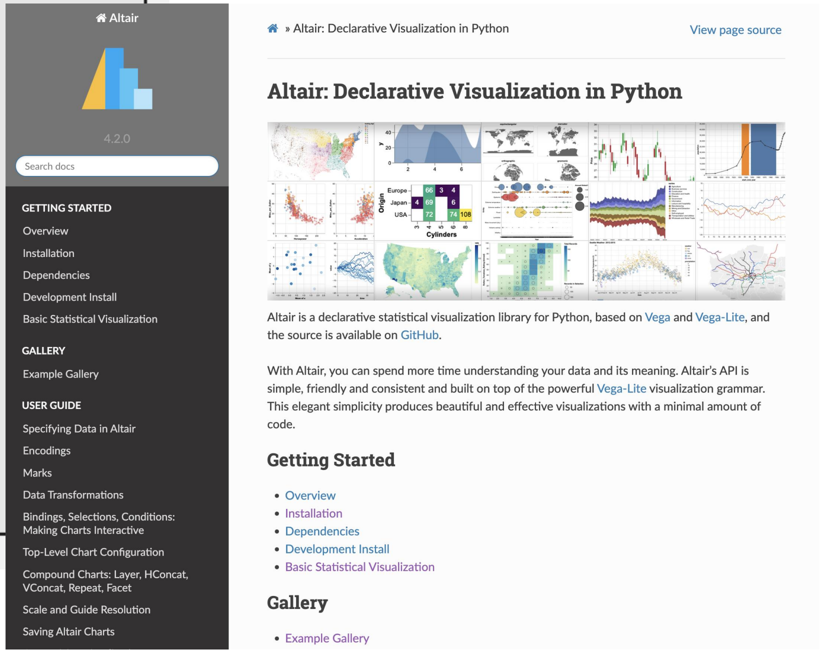
Part-2: Making it interactive
- Bringing in interaction
- Linking it up
- Going back to the data

VDS DIY BLOCK #3

Doing (V)DS responsibly

Part-1: Being aware
- Dance of the p-values
- Maps and colouring

Part-2: Communicating openly
- Thinking about the narrative
- Communicating responsibly



Altair

4.2.0

Search docs

GETTING STARTED
Overview
Installation
Dependencies
Development Install
Basic Statistical Visualization

GALLERY
Example Gallery

USER GUIDE
Specifying Data in Altair
Encodings
Marks
Data Transformations
Bindings, Selections, Conditions: Making Charts Interactive
Top-Level Chart Configuration
Compound Charts: Layer, HConcat, VConcat, Repeat, Facet
Scale and Guide Resolution
Saving Altair Charts

» Altair: Declarative Visualization in Python | View page source

Altair: Declarative Visualization in Python

Altair is a declarative statistical visualization library for Python, based on Vega and Vega-Lite, and the source is available on GitHub.

With Altair, you can spend more time understanding your data and its meaning. Altair's API is simple, friendly and consistent and built on top of the powerful Vega-Lite visualization grammar. This elegant simplicity produces beautiful and effective visualizations with a minimal amount of code.

Getting Started

- Overview
- Installation
- Dependencies
- Development Install
- Basic Statistical Visualization

Gallery

- Example Gallery

- Hands-on exercises to explore some of these practices/ideas
- Will use Altair in Jupyter Notebooks to visualise and bring interactivity

Doing Visual Data Science

Foundations, Techniques and Practice

DIY BLOCK #1 - Thinking Visually

Part-1: Re/De-constructing Visualisations

University of Bergen, 10 August 2023

BUILDING BLOCKS OF VISUALISATIONS

VISUAL ELEMENTS

Marks & Channels

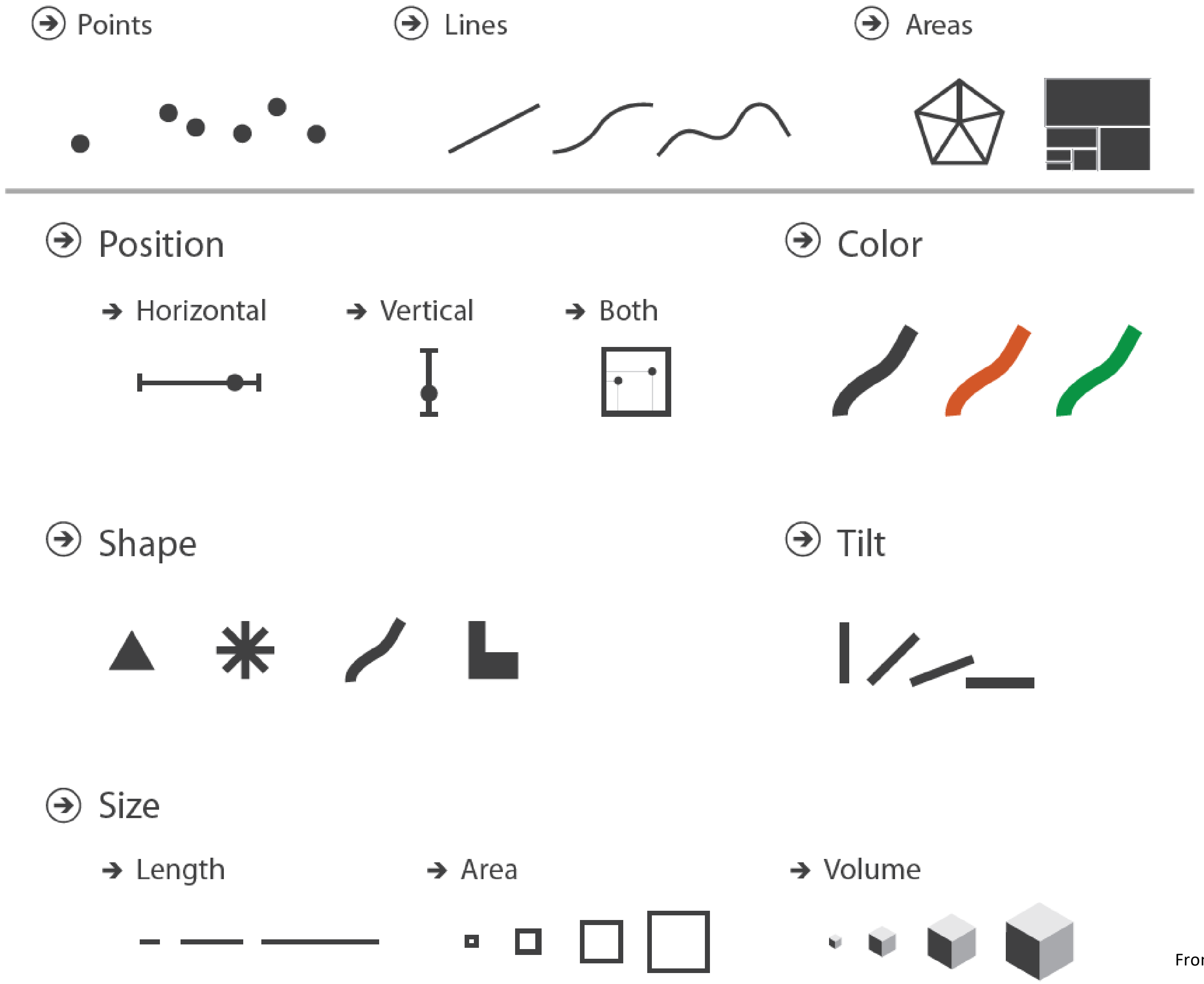
Visual Marks: represent **items (**or **links)**

Channels: change **appearance** based on **attribute**

Channel = Visual Variable

Basic visual elements

VISUAL MARKS
i.e., primitives

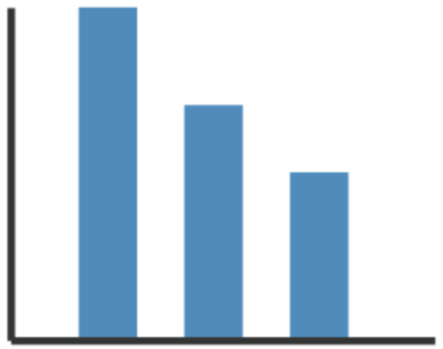


VISUAL CHANNELS
i.e., how they look

From Visualization Analysis & Design by T. Munzner, 2014

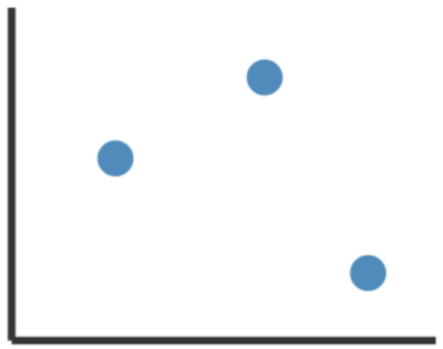
Visual encoding

Def. *Representing information visually as combination of marks and channels*



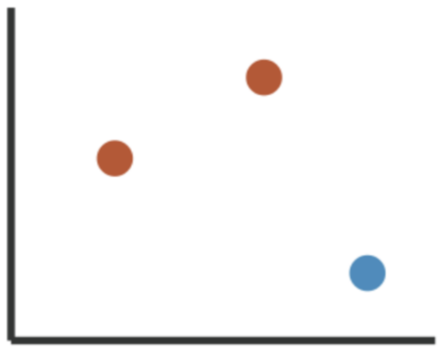
1:
vertical position

mark: line



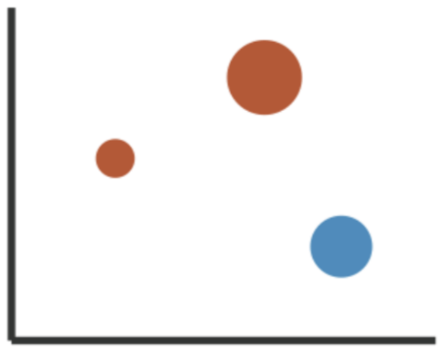
2:
vertical position
horizontal position

mark: point



3:
vertical position
horizontal position
color hue

mark: point



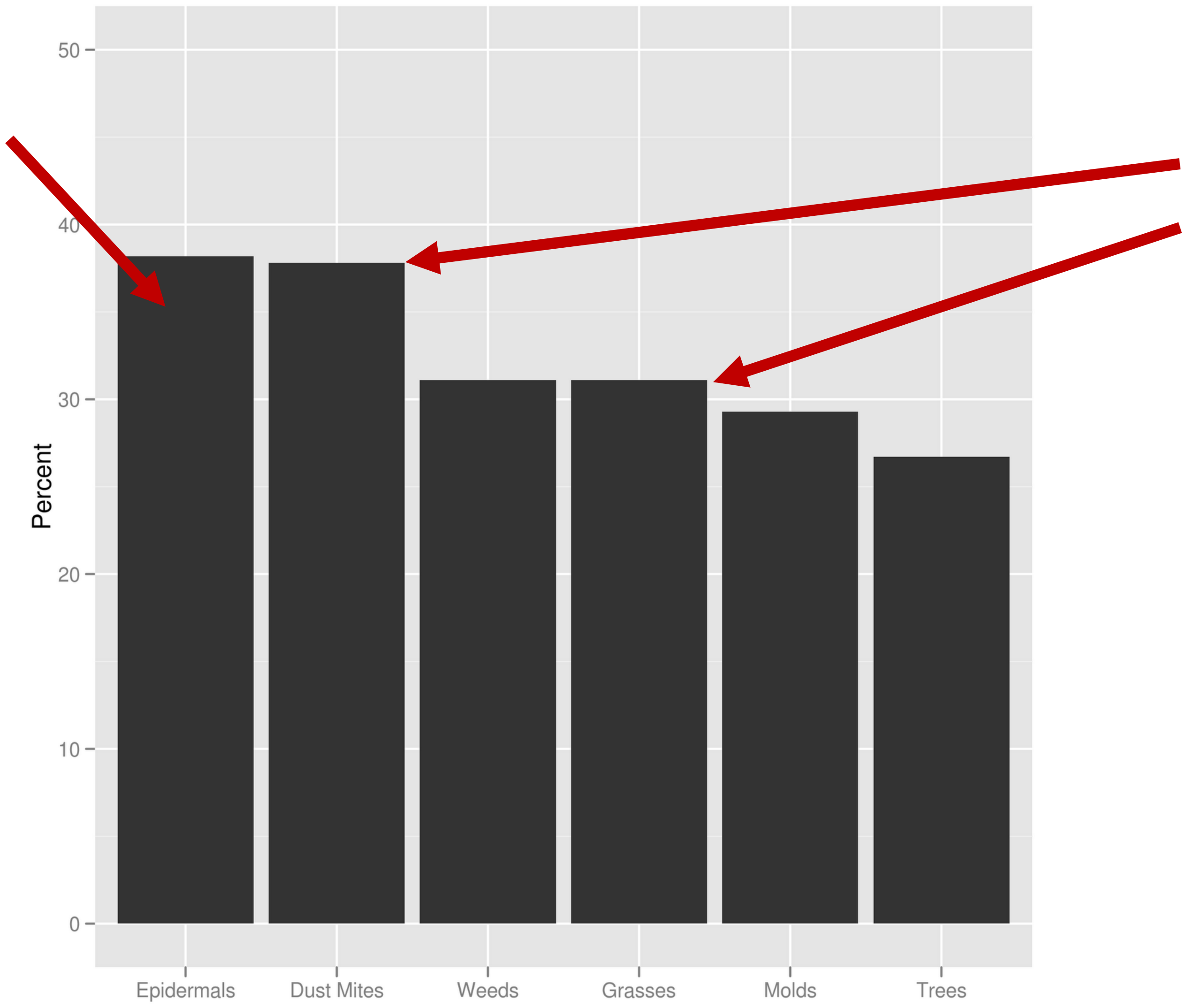
4:
vertical position
horizontal position
color hue
size (area)

mark: point

Basic visual elements in action

VISUAL MARK
lines (**thick** ones)

VISUAL CHANNEL
length (of bars)



http://rrubyperlundich.blogspot.co.uk/2012_09_01_archive.html

Basic visual elements in action

VISUAL CHANNELS

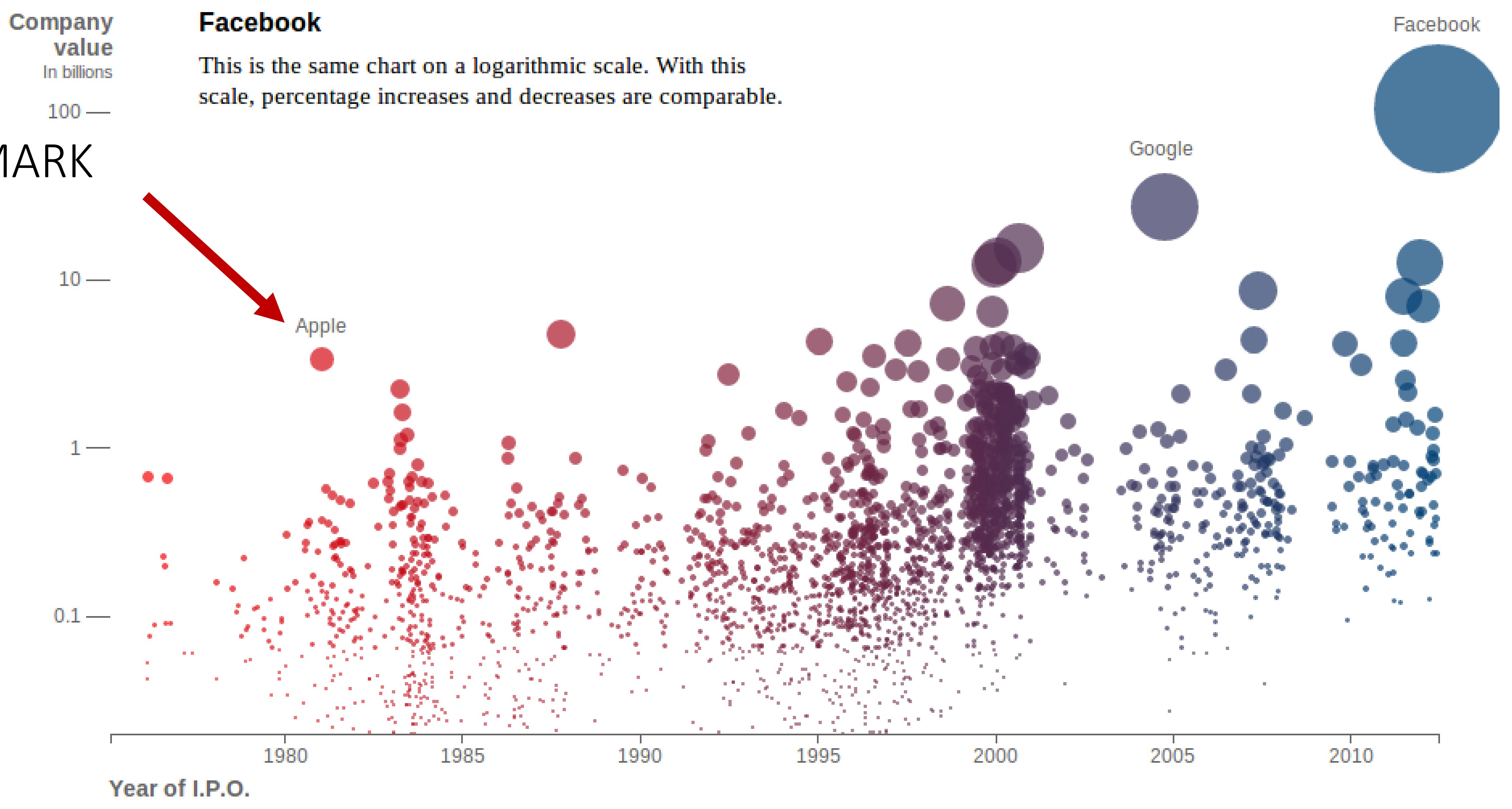
position x and y

colour

Area (redundant)

VISUAL MARK

points

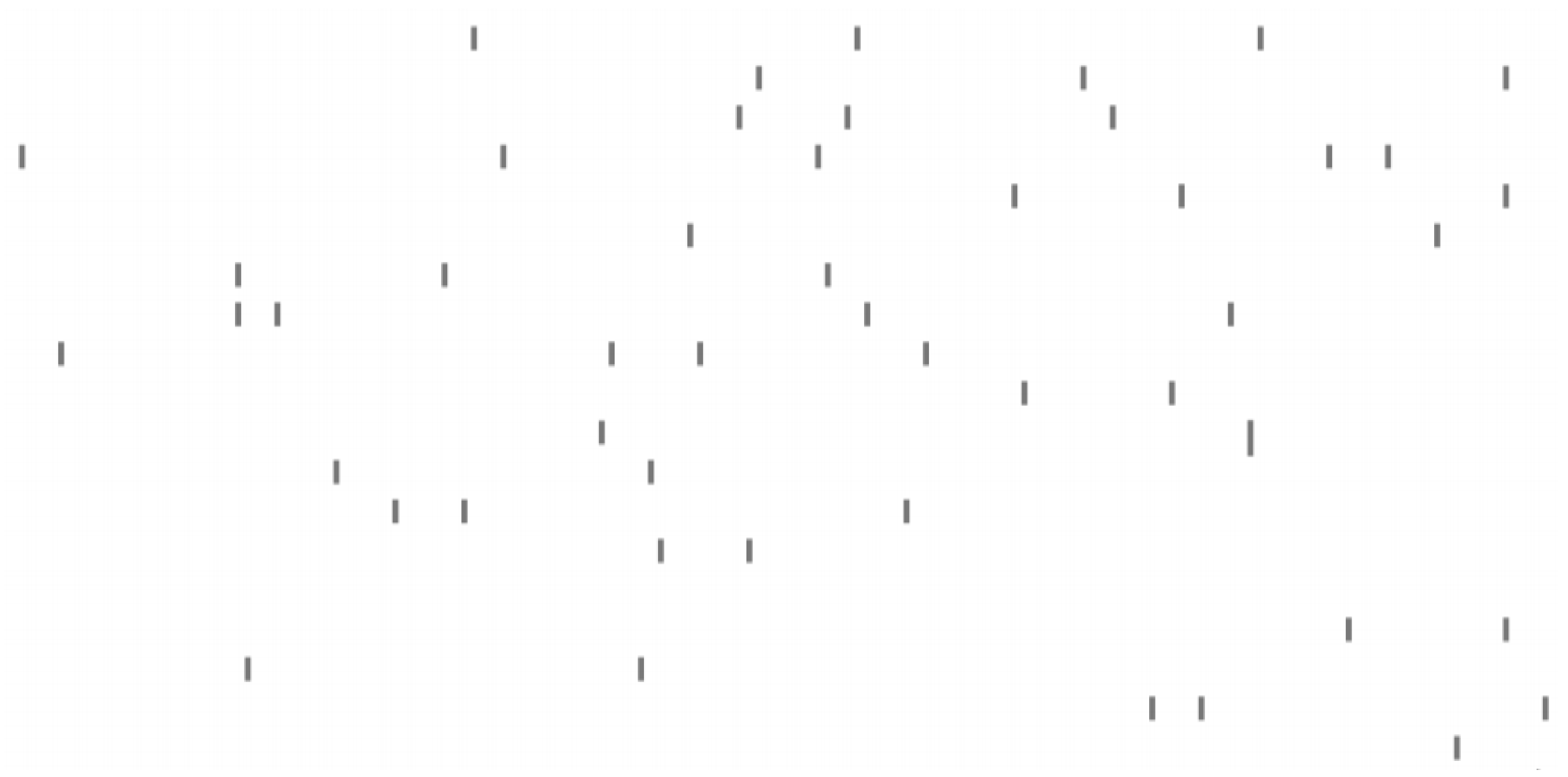


Facebook

This is the same chart on a logarithmic scale. With this scale, percentage increases and decreases are comparable.

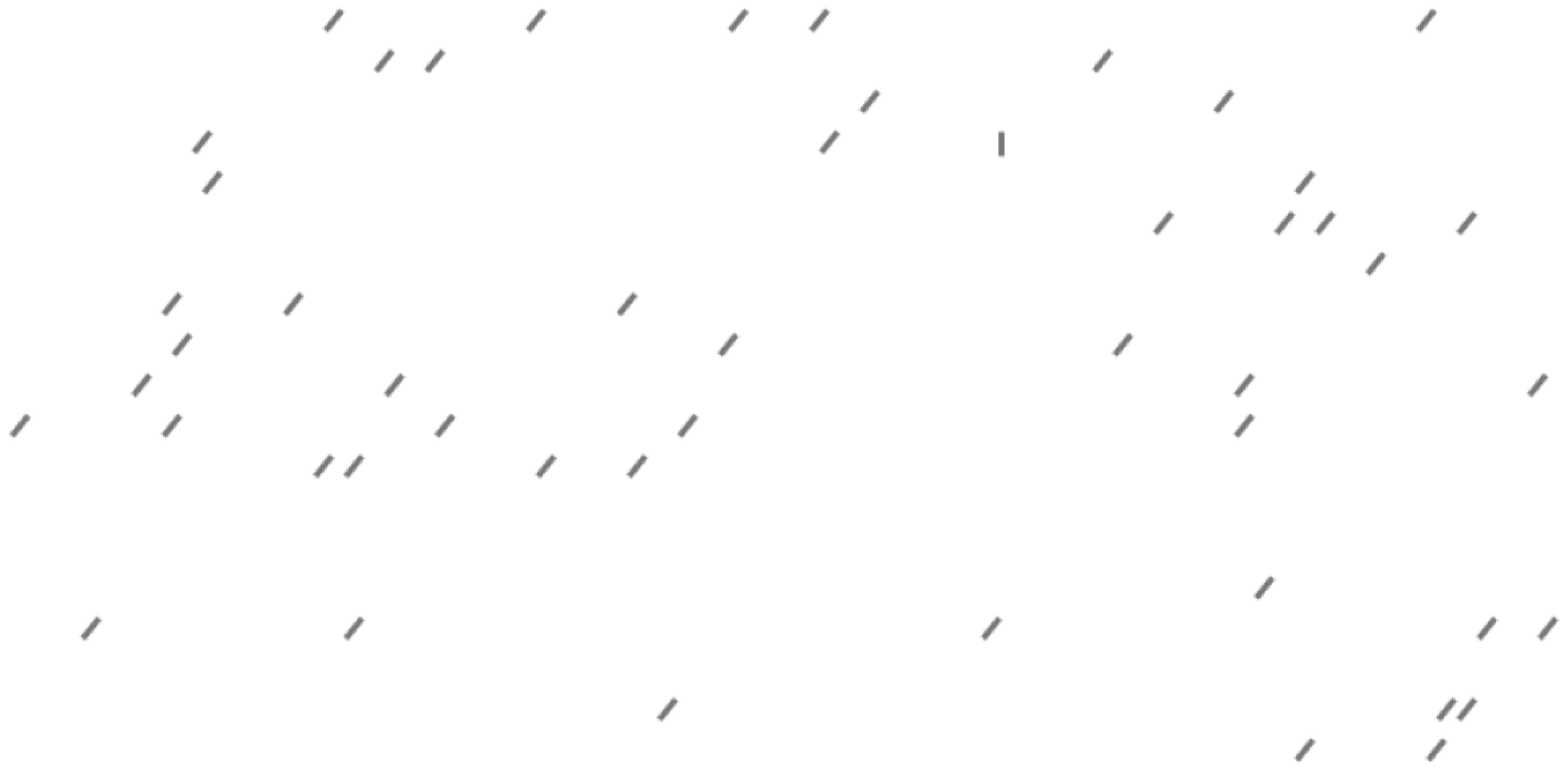
http://www.nytimes.com/interactive/2012/05/17/business/dealbook/how-the-facebook-offering-compares.html

FIND THE ODD ONE OUT!

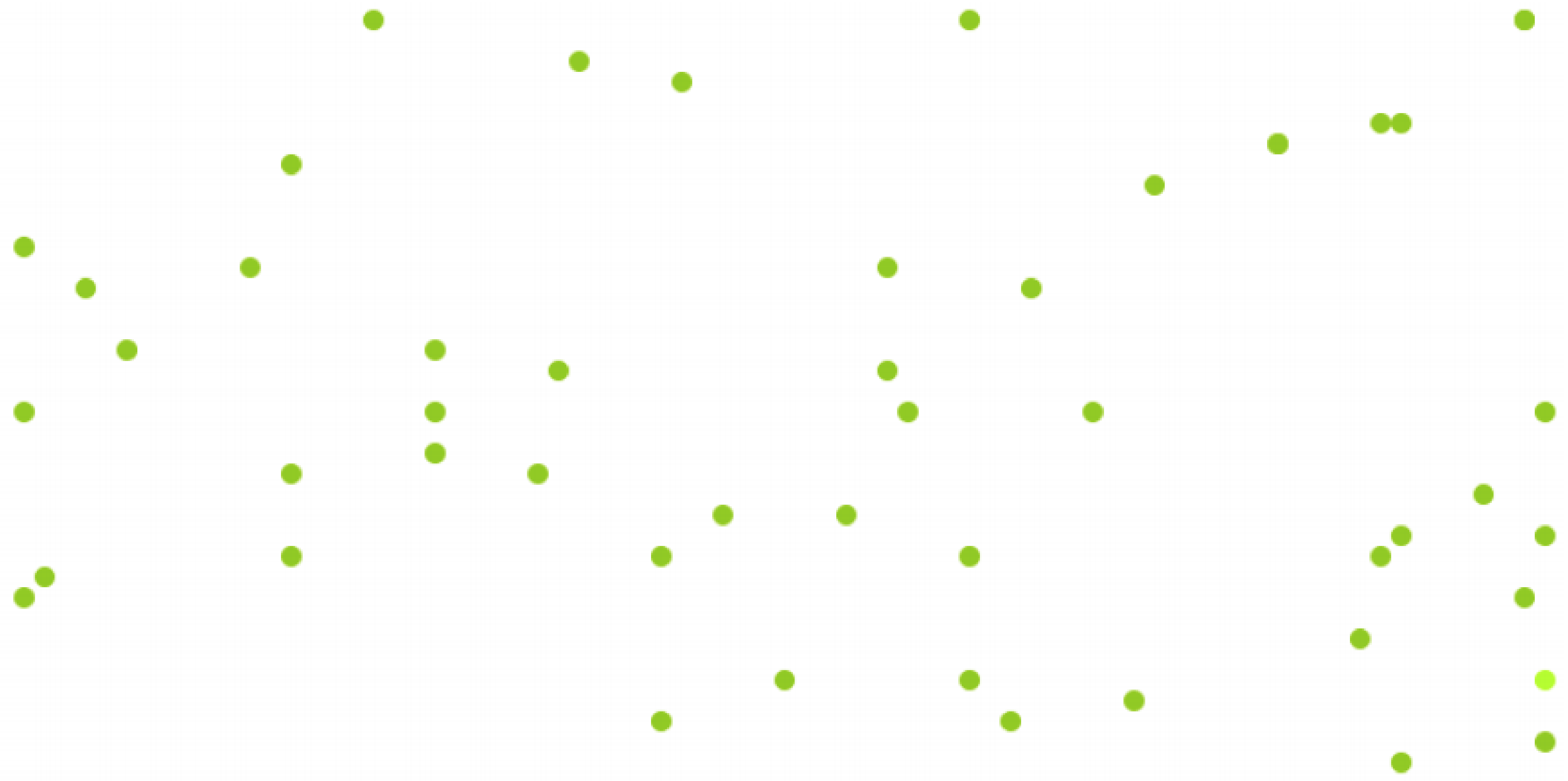


Length

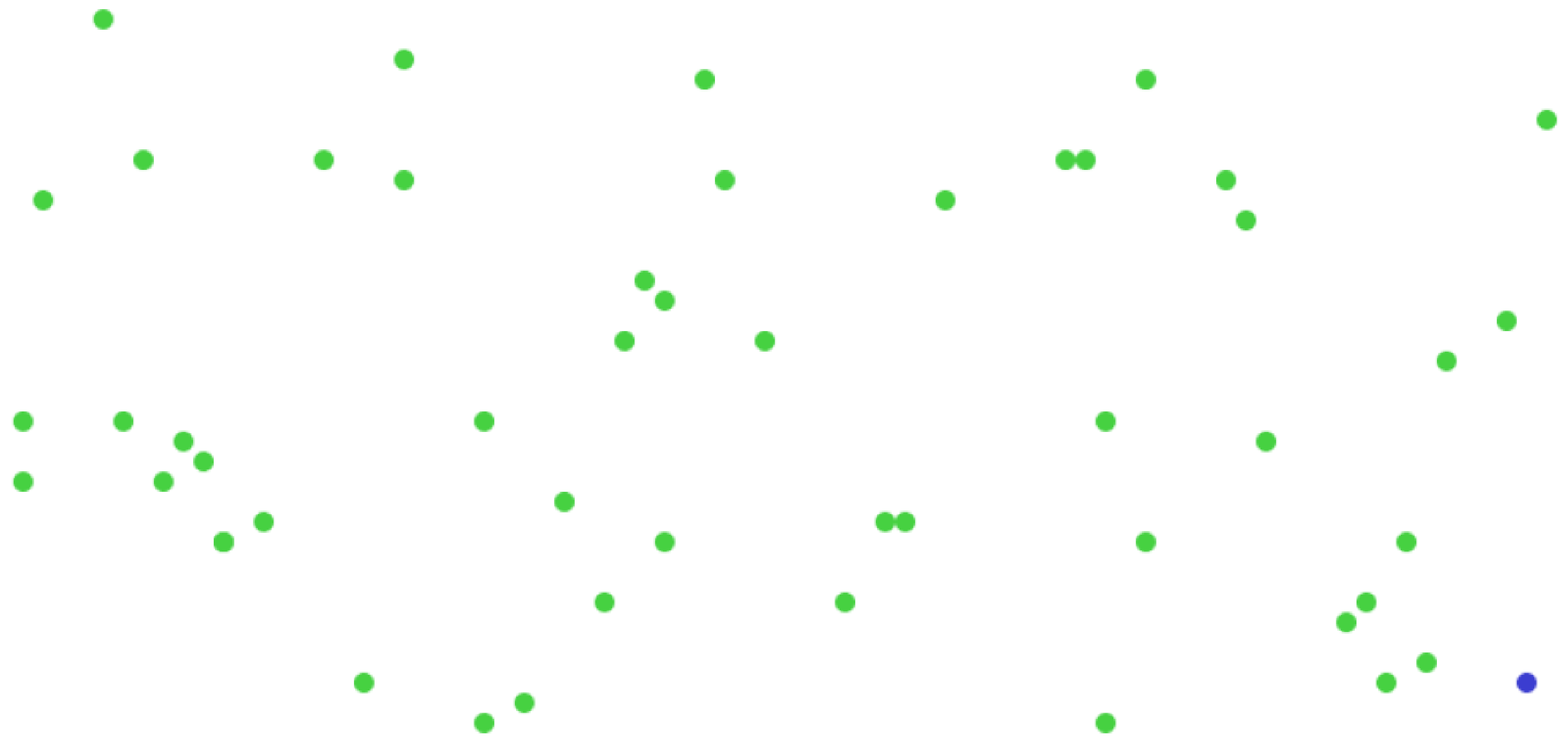
Size



Orientation

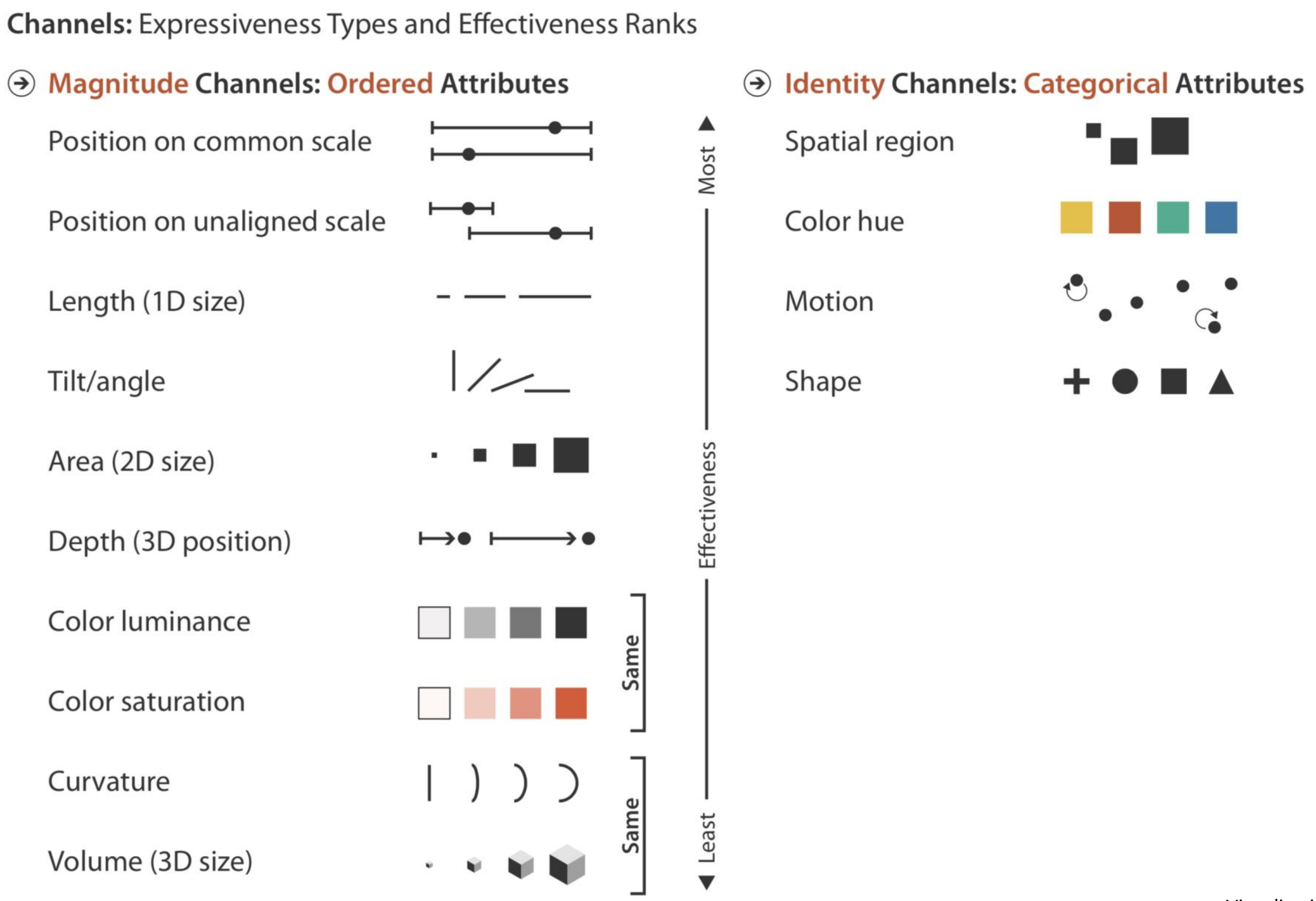


Lightness



Hue

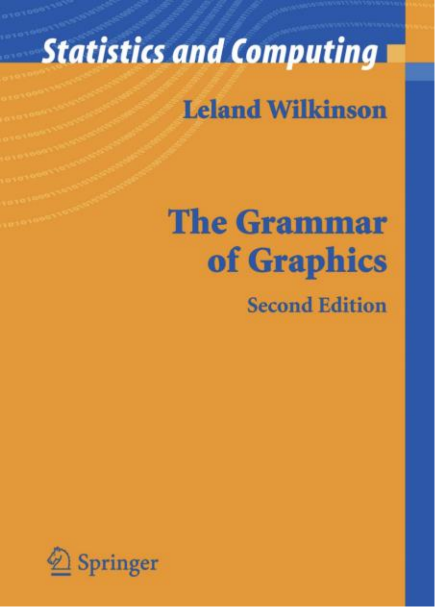
Visual channels don't work equally well!



MOST EFFECTIVE

LEAST EFFECTIVE

Visualization Analysis & Design by T. Munzner, 2014



Statistics and Computing

Leland Wilkinson

The Grammar of Graphics

Second Edition

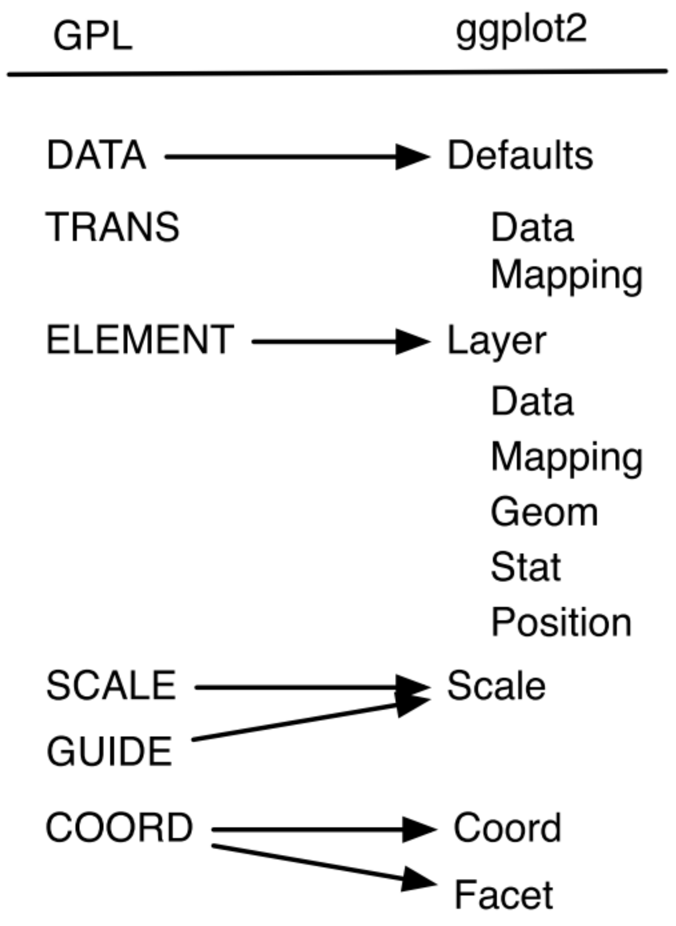
Springer

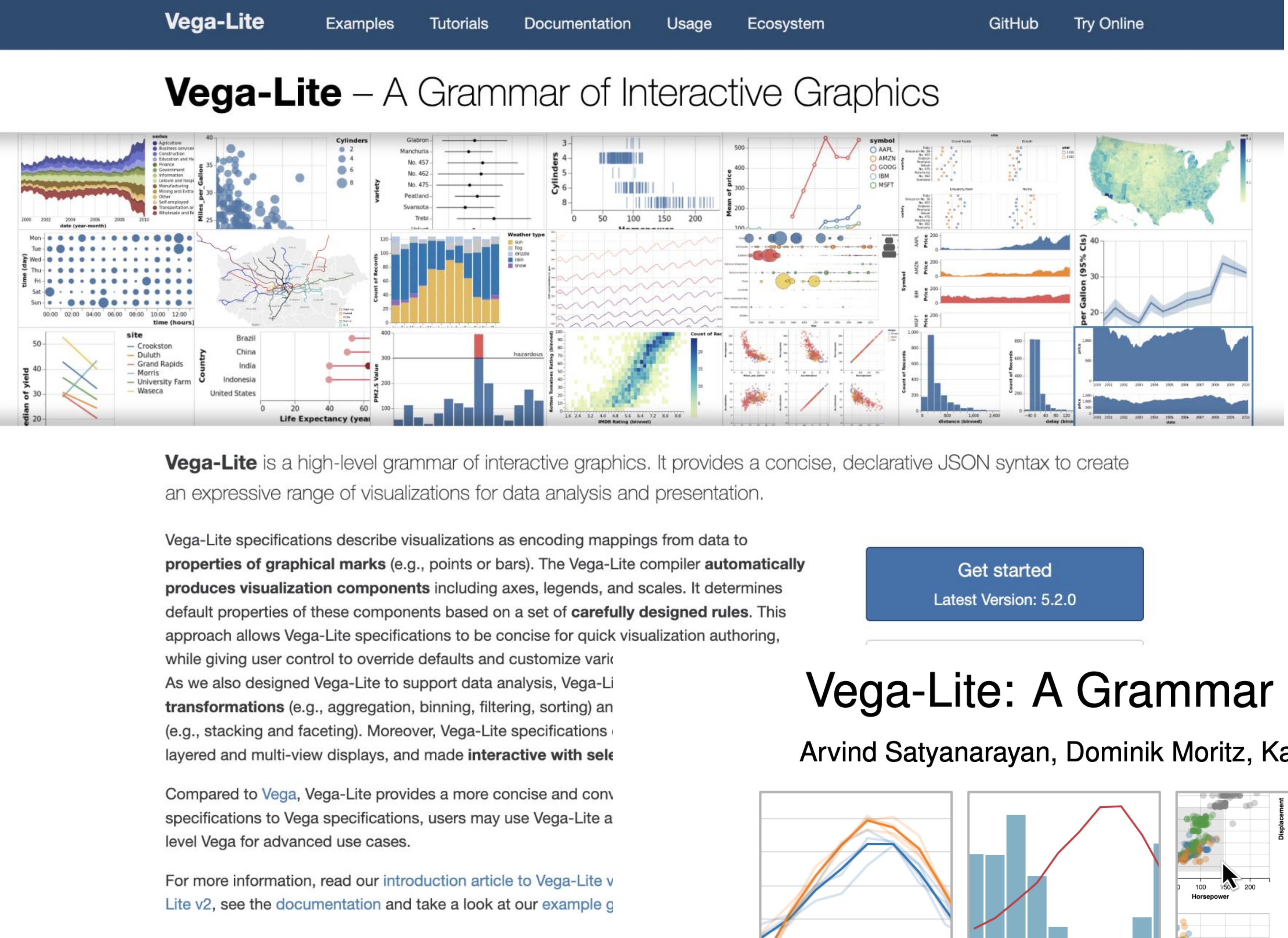
A Layered Grammar of Graphics

Hadley WICKHAM

Statistical graphic specifications are expressed in six statements:

1) DATA: a set of data operations that create variables from datasets,
2) TRANS: variable transformations (*e.g., rank*),
3) SCALE: scale transformations (*e.g., log*),
4) COORD: a coordinate system (*e.g., polar*),
5) ELEMENT: graphs (*e.g., points*) and their aesthetic attributes (*e.g., color*),
6) GUIDE: one or more guides (*axes*, *legends*, etc.).





Vega-Lite – A Grammar of Interactive Graphics

Vega-Lite is a high-level grammar of interactive graphics. It provides a concise, declarative JSON syntax to create an expressive range of visualizations for data analysis and presentation.

Vega-Lite specifications describe visualizations as encoding mappings from data to **properties of graphical marks** (e.g., points or bars). The Vega-Lite compiler **automatically produces visualization components** including axes, legends, and scales. It determines default properties of these components based on a set of **carefully designed rules**. This approach allows Vega-Lite specifications to be concise for quick visualization authoring, while giving user control to override defaults and customize vari
As we also designed Vega-Lite to support data analysis, Vega-Li
transformations (e.g., aggregation, binning, filtering, sorting) an
(e.g., stacking and faceting). Moreover, Vega-Lite specifications
layered and multi-view displays, and made **interactive with sele**

Compared to Vega, Vega-Lite provides a more concise and conv
specifications to Vega specifications, users may use Vega-Lite a
level Vega for advanced use cases.

For more information, read our introduction article to Vega-Lite v
Lite v2, see the documentation and take a look at our example g

Vega-Lite: A Grammar of Interactive Graphics

Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer

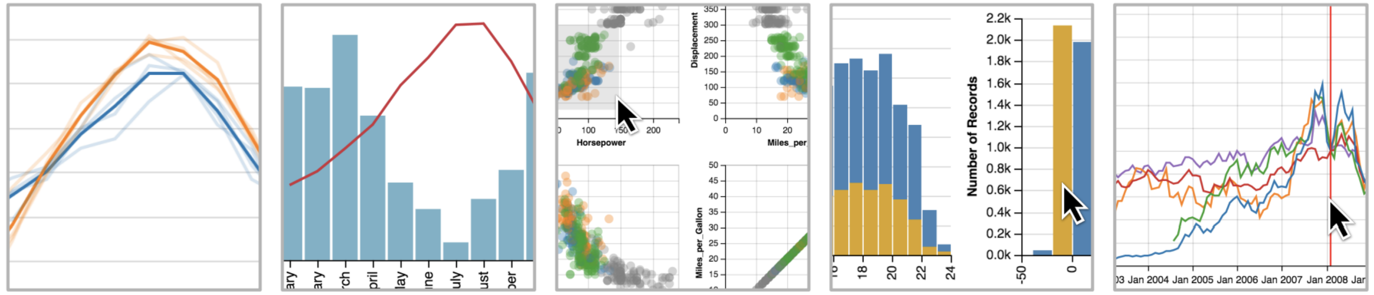
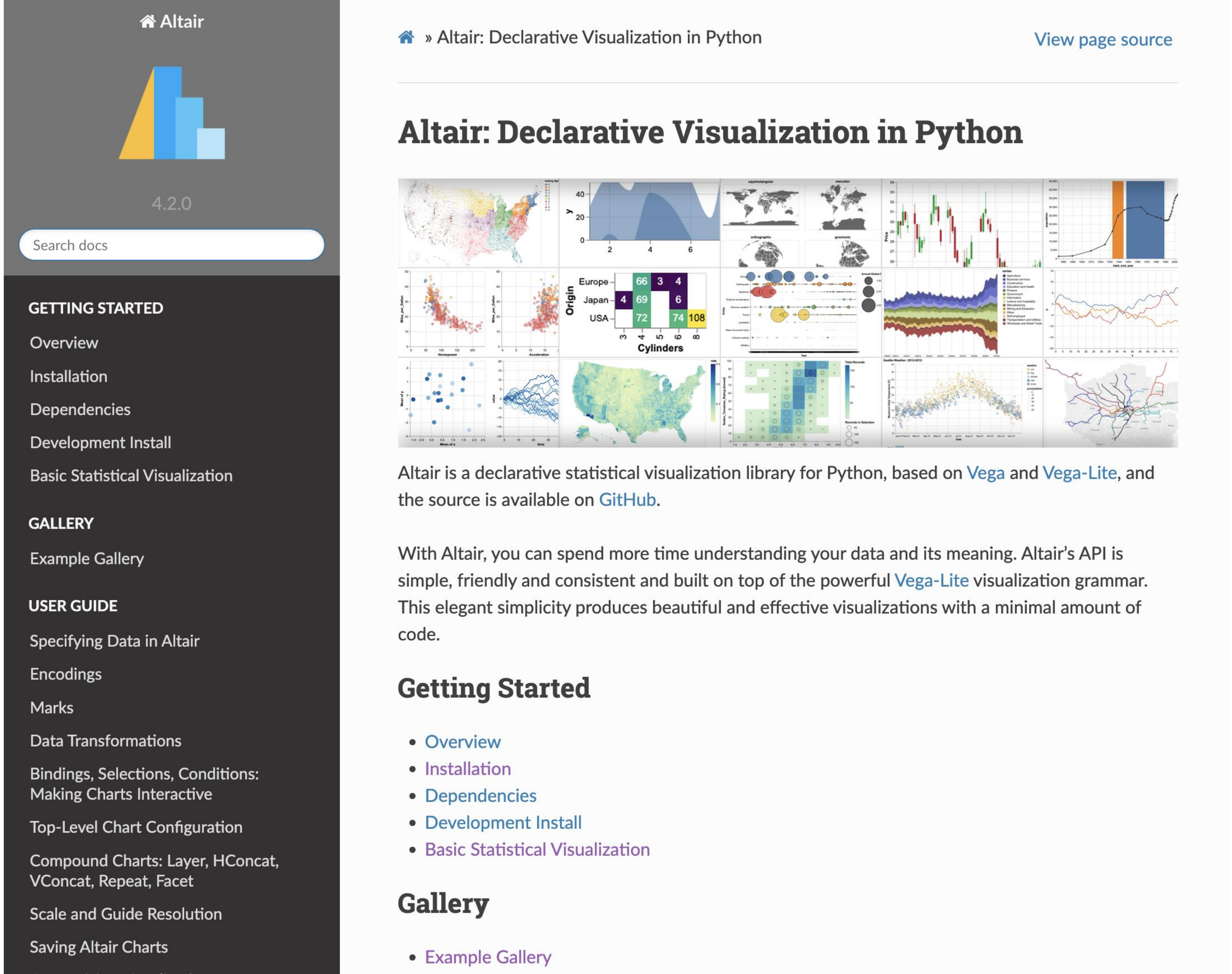


Fig. 1. Example visualizations authored with Vega-Lite. From left-to-right: layered line chart combining raw and average values, dual-axis layered bar and line chart, brushing and linking in a scatterplot matrix, layered cross-filtering, and an interactive index chart.



Altair

4.2.0

Search docs

GETTING STARTED

- Overview
- Installation
- Dependencies
- Development Install
- Basic Statistical Visualization

GALLERY

- Example Gallery

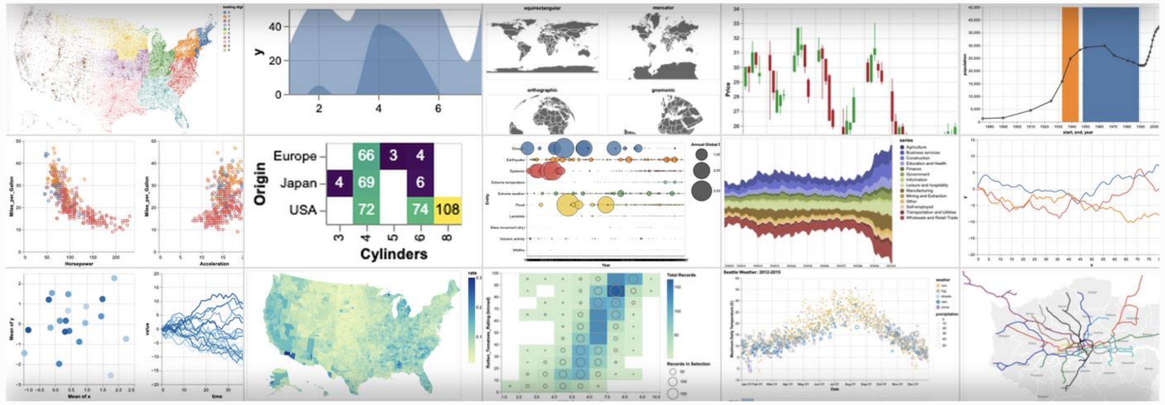
USER GUIDE

- Specifying Data in Altair
- Encodings
- Marks
- Data Transformations
- Bindings, Selections, Conditions: Making Charts Interactive
- Top-Level Chart Configuration
- Compound Charts: Layer, HConcat, VConcat, Repeat, Facet
- Scale and Guide Resolution
- Saving Altair Charts

» Altair: Declarative Visualization in Python

View page source

Altair: Declarative Visualization in Python



Altair is a declarative statistical visualization library for Python, based on Vega and Vega-Lite, and the source is available on GitHub.

With Altair, you can spend more time understanding your data and its meaning. Altair's API is simple, friendly and consistent and built on top of the powerful Vega-Lite visualization grammar. This elegant simplicity produces beautiful and effective visualizations with a minimal amount of code.

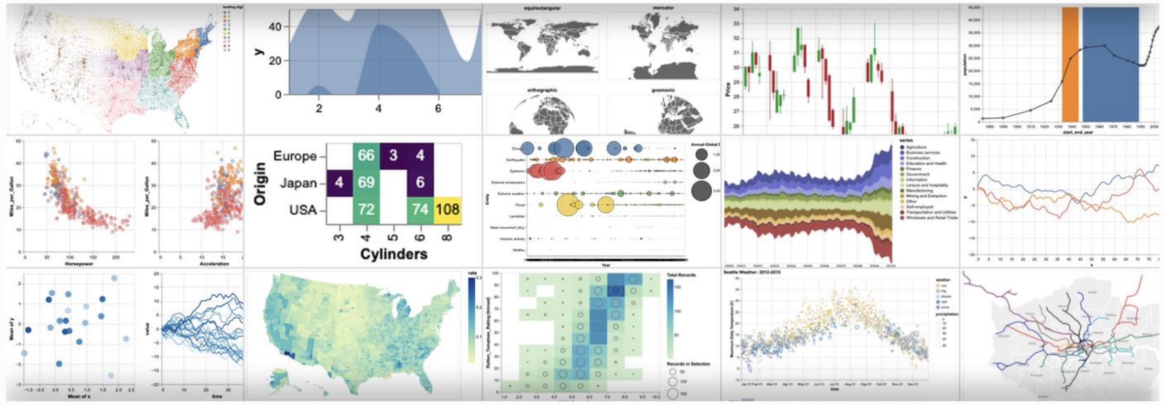
Getting Started

- Overview
- Installation
- Dependencies
- Development Install
- Basic Statistical Visualization

Gallery

- Example Gallery

Altair: Declarative Visualization in Python



Altair is a declarative statistical visualization library for Python, based on Vega and Vega-Lite, and the source is available on GitHub.

With Altair, you can spend more time understanding your data and its meaning. Altair's API is simple, friendly and consistent and built on top of the powerful Vega-Lite visualization grammar. This elegant simplicity produces beautiful and effective visualizations with a minimal amount of code.

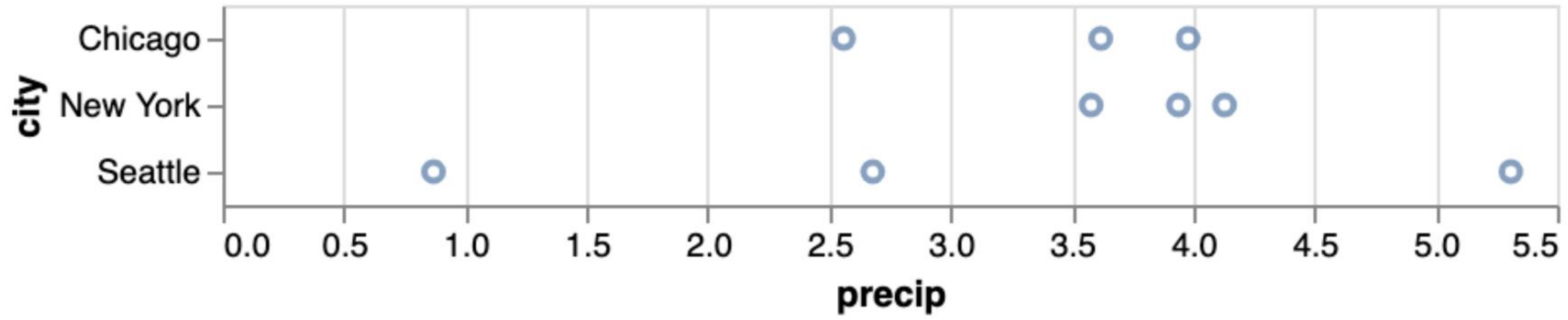
Getting Started

- Overview
- Installation
- Dependencies
- Development Install
- Basic Statistical Visualization

Gallery

- Example Gallery

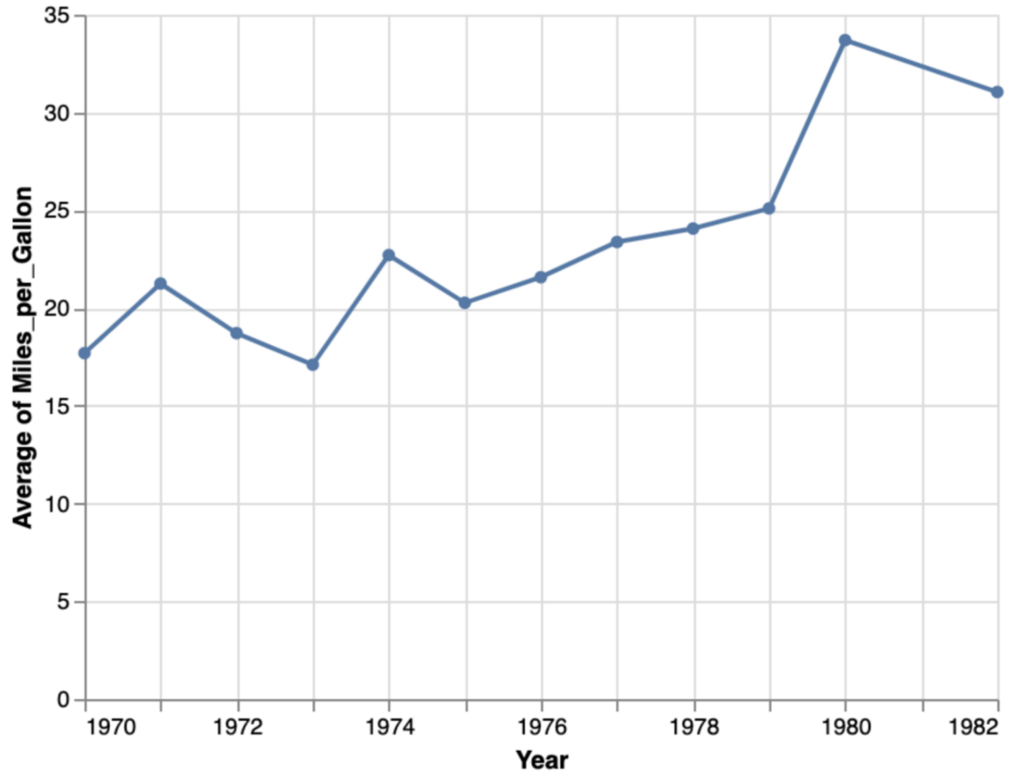
```python
alt.Chart(df).mark_point().encode(
    alt.X('precip'),
    alt.Y('city')
)
```



```python
line = alt.Chart(cars).mark_line().encode(
    alt.X('Year'),
    alt.Y('average(Miles_per_Gallon)')
)

point = alt.Chart(cars).mark_circle().encode(
    alt.X('Year'),
    alt.Y('average(Miles_per_Gallon)')
)

line + point
```



Doing Visual Data Science

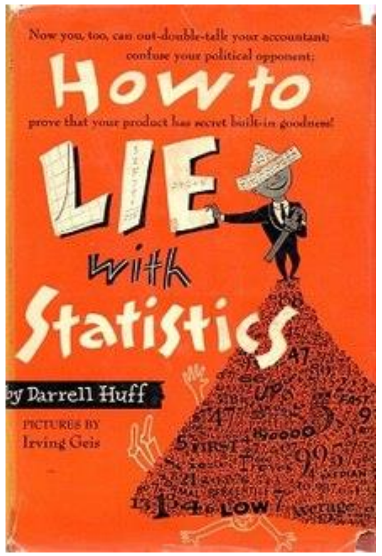
Foundations, Techniques and Practice

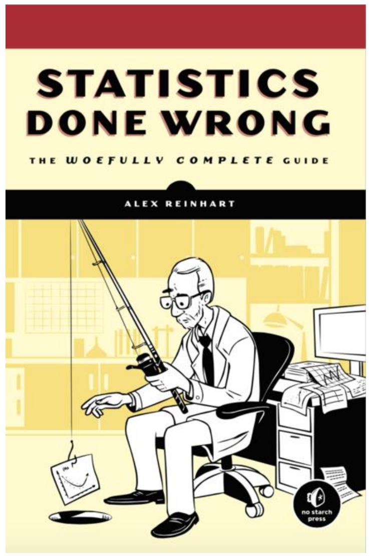
DIY BLOCK #3 - Doing (V)DS responsibly

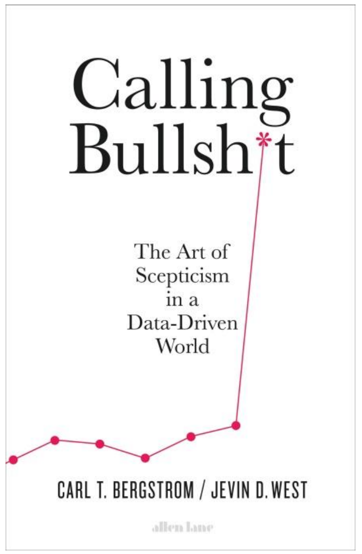
University of Bergen, 10 August 2023

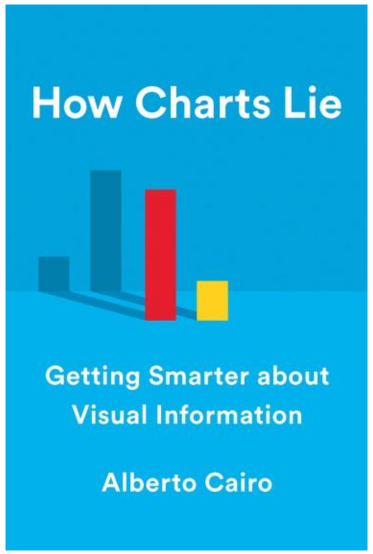
"There are three kinds of lies: lies, damned lies, and statistics."

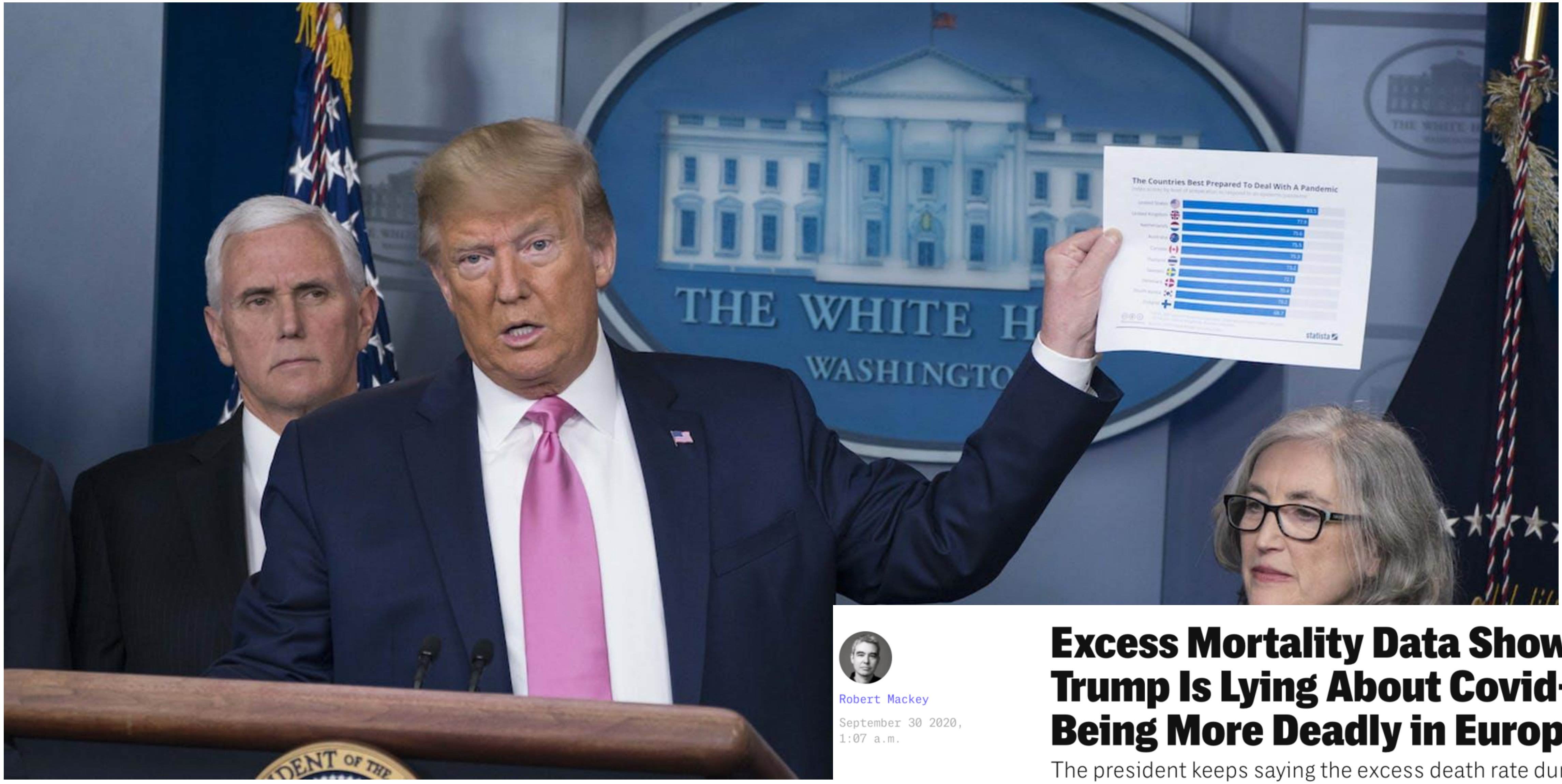
(late 1800s, source debated but popularised by Mark Twain, see: https://en.wikipedia.org/wiki/Lies,_damned_lies,_and_statistics)


Now you, too, can out-double-talk your accountant; confuse your political opponent;
How to
prove that your product has secret built-in goodness!
LIE
with
Statistics
by Darrell Huff
PICTURES BY
Irving Geis




Calling Bullsh*t
The Art of Scepticism in a Data-Driven World
CARL T. BERGSTROM / JEVIN D. WEST
allen lane





Robert Mackey

September 30 2020, 1:07 a.m.

Excess Mortality Data Shows Trump Is Lying About Covid-19 Being More Deadly in Europe

The president keeps saying the excess death rate during the pandemic is higher in Europe than in the U.S. That's not true.

https://theintercept.com/2020/09/29/excess-mortality-data-shows-trump-lying-covid-deadly-europe/

Truncated axes

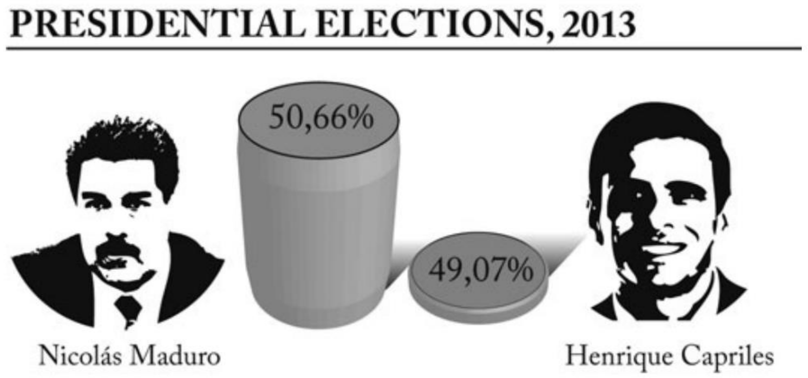


Fig. 5.3 Presidential election results in Venezuela, based on a graphic by Venezonala de Televisión. Notice the truncated *Y*-axis which greatly distorts the difference between the percentages of vote

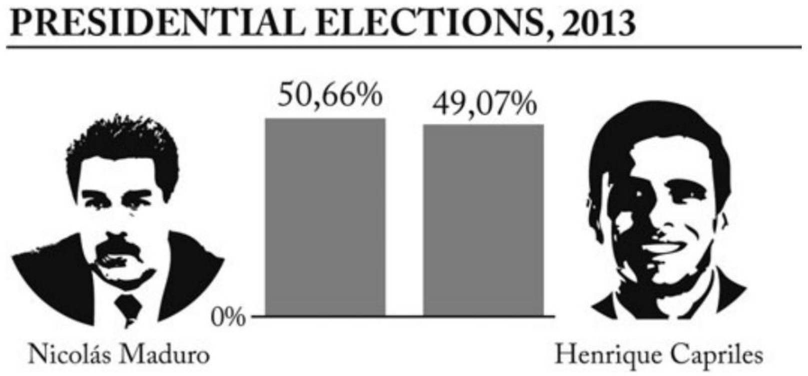
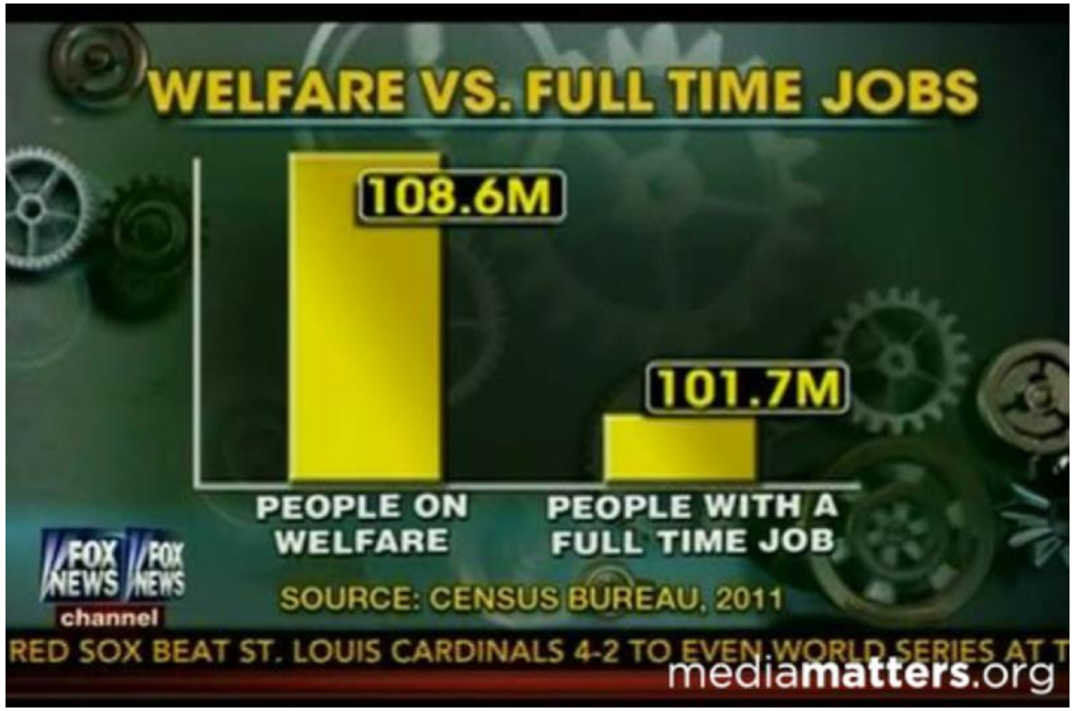


Fig. 5.4 An alternative version of the previous graphic in which a 0-baseline has been added, and the 3D effect has been removed

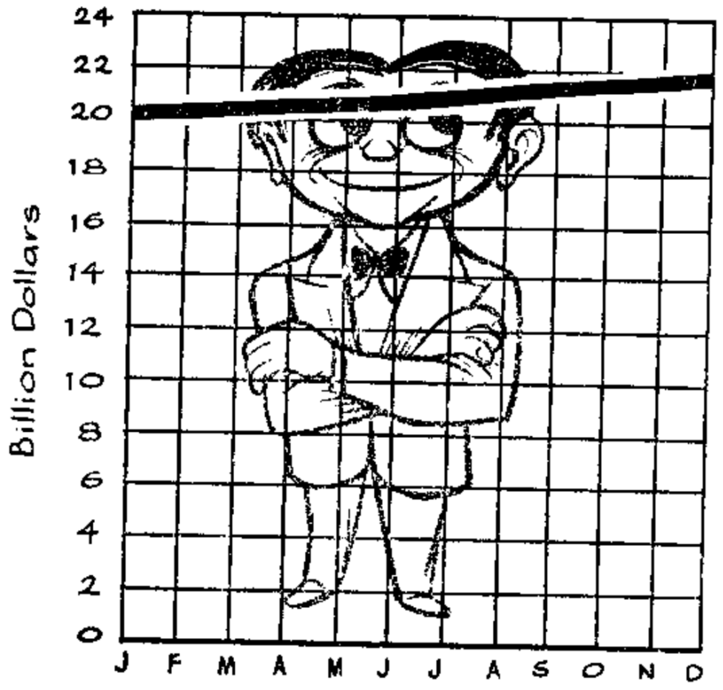


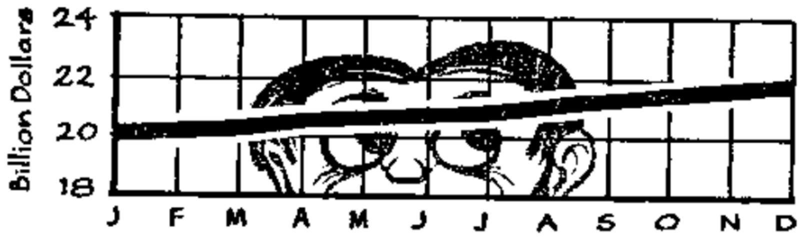
Chapter 5
Graphics Lies, Misleading Visuals

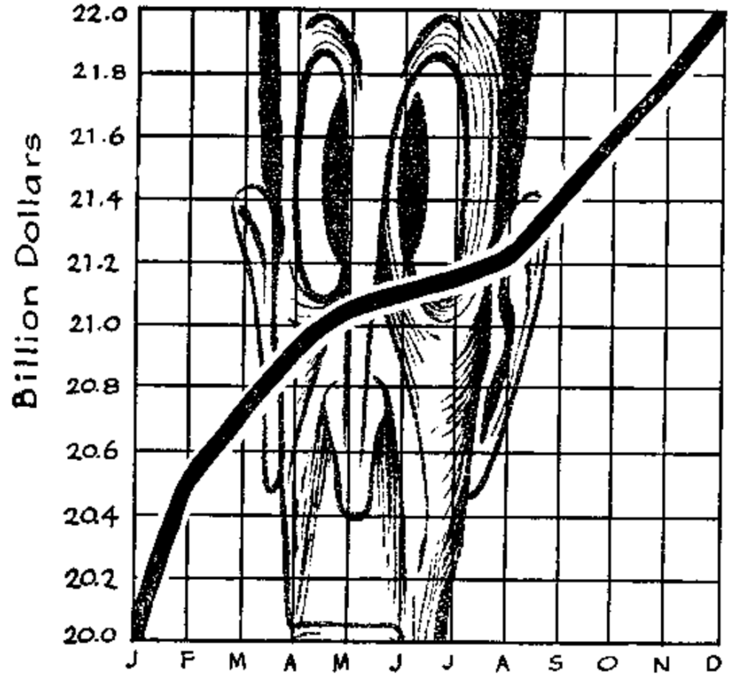
Reflections on the Challenges and Pitfalls of Evidence-Driven Visual Communication

Alberto Cairo

Framing and the power of axes

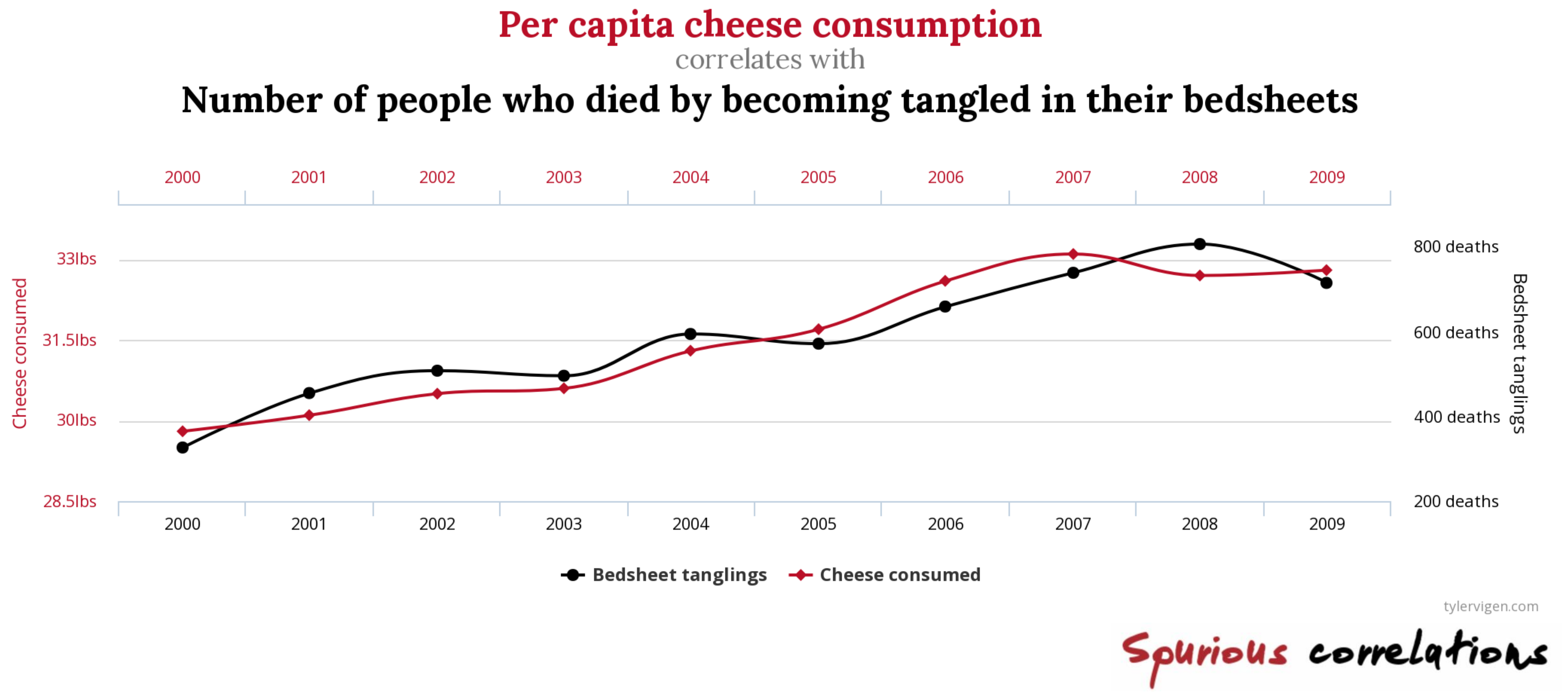




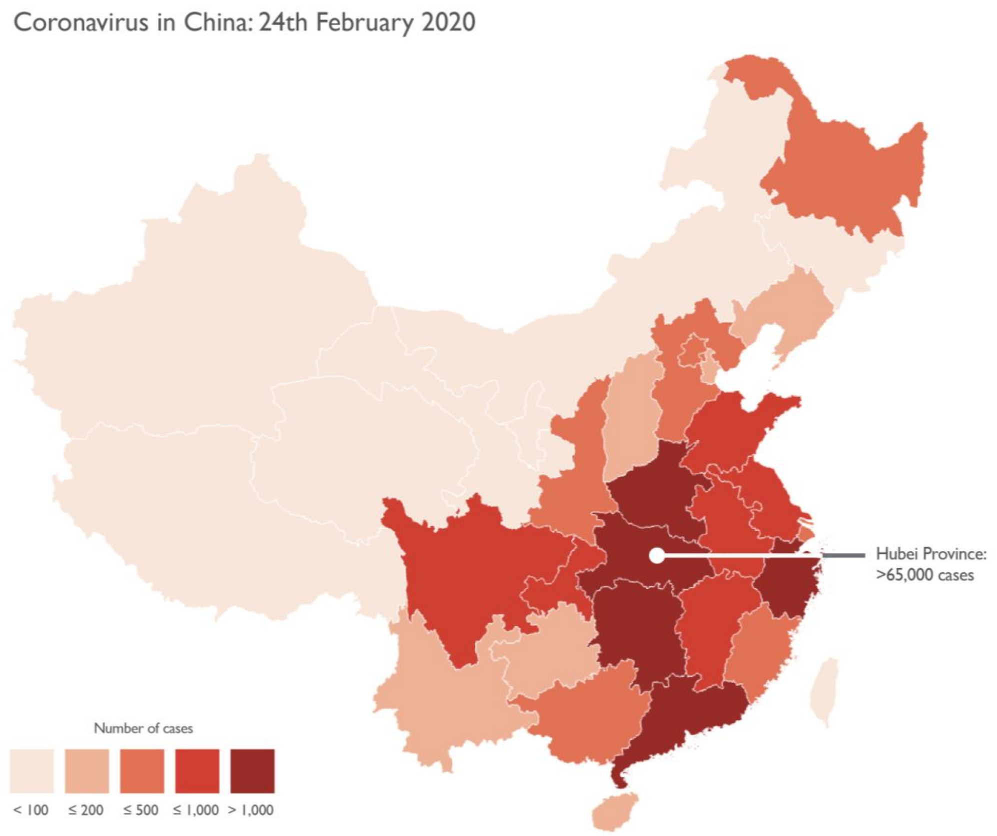


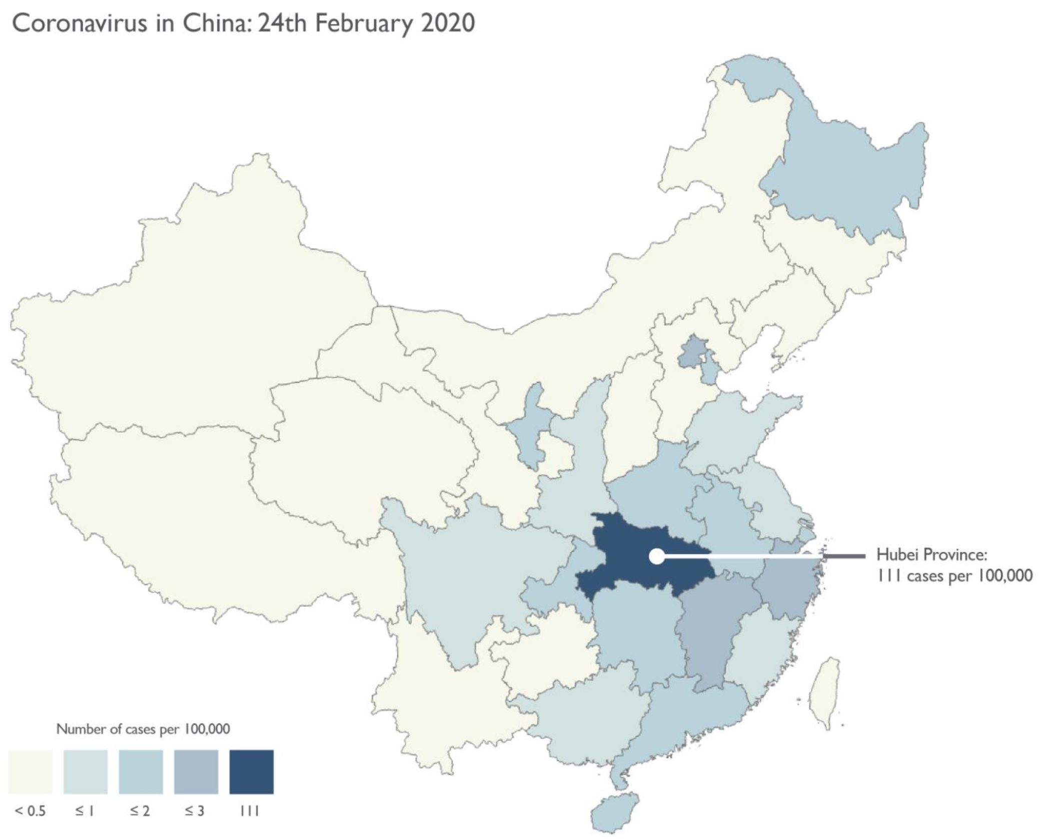
Figures from: Huff, D., 1993. *How to lie with statistics*. WW Norton & Company.

Two axes charts and a false sense of association

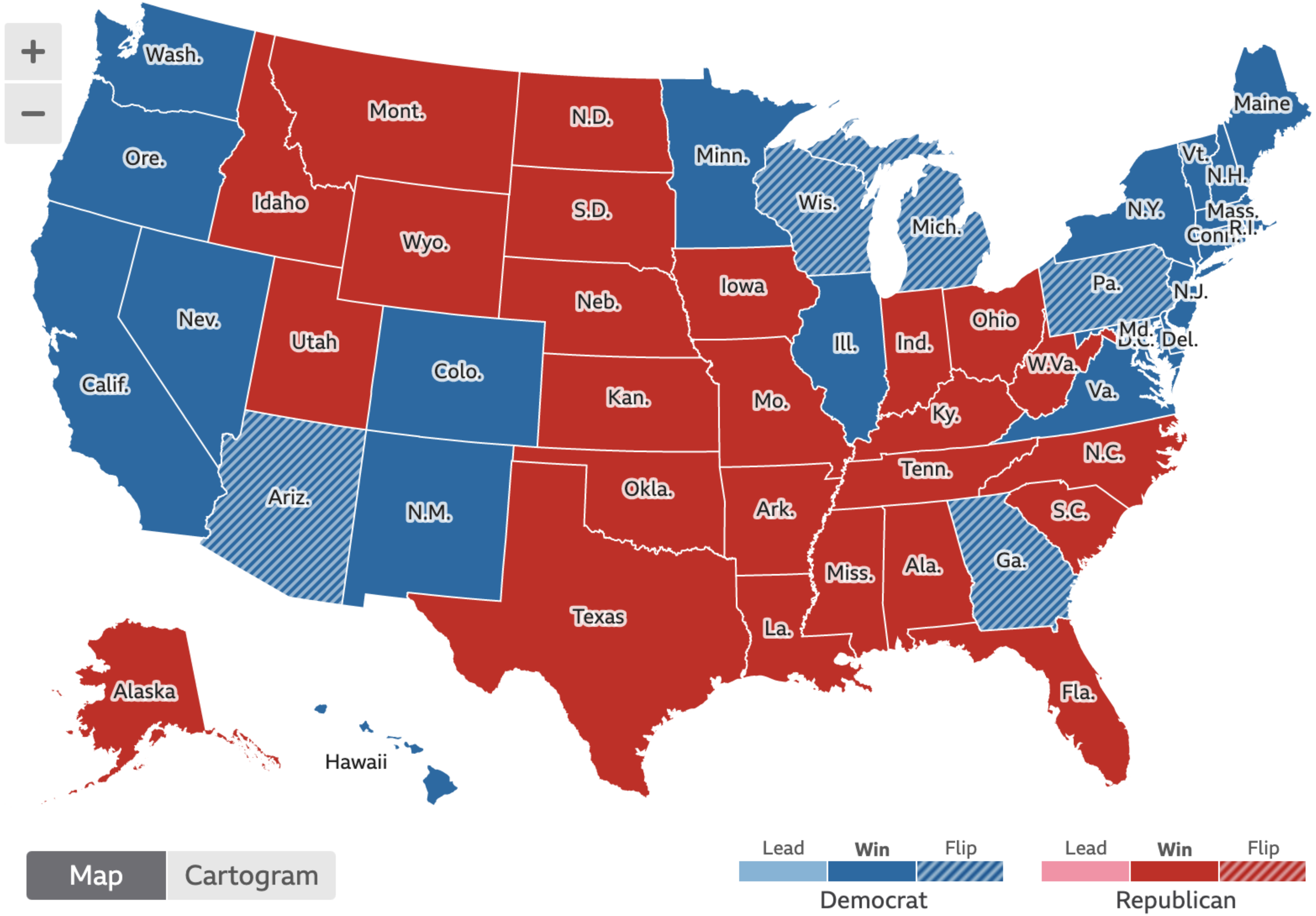


Absolute/ratios – unfair comparisons, data semantics

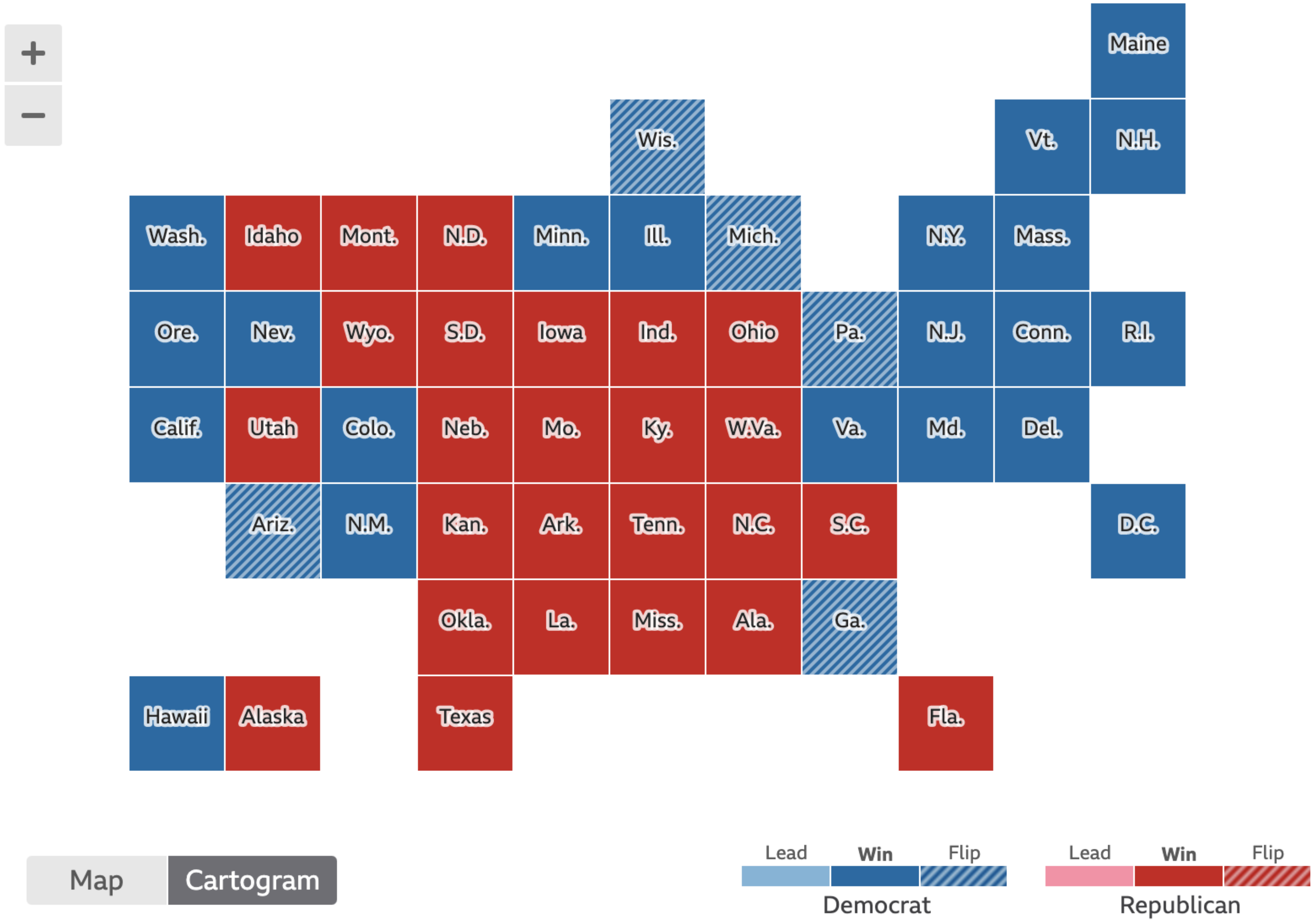



Coronavirus in China: 24th February 2020
Hubei Province:
111 cases per 100,000
Number of cases per 100,000
< 0.5
≤ 1
≤ 2
≤ 3
111

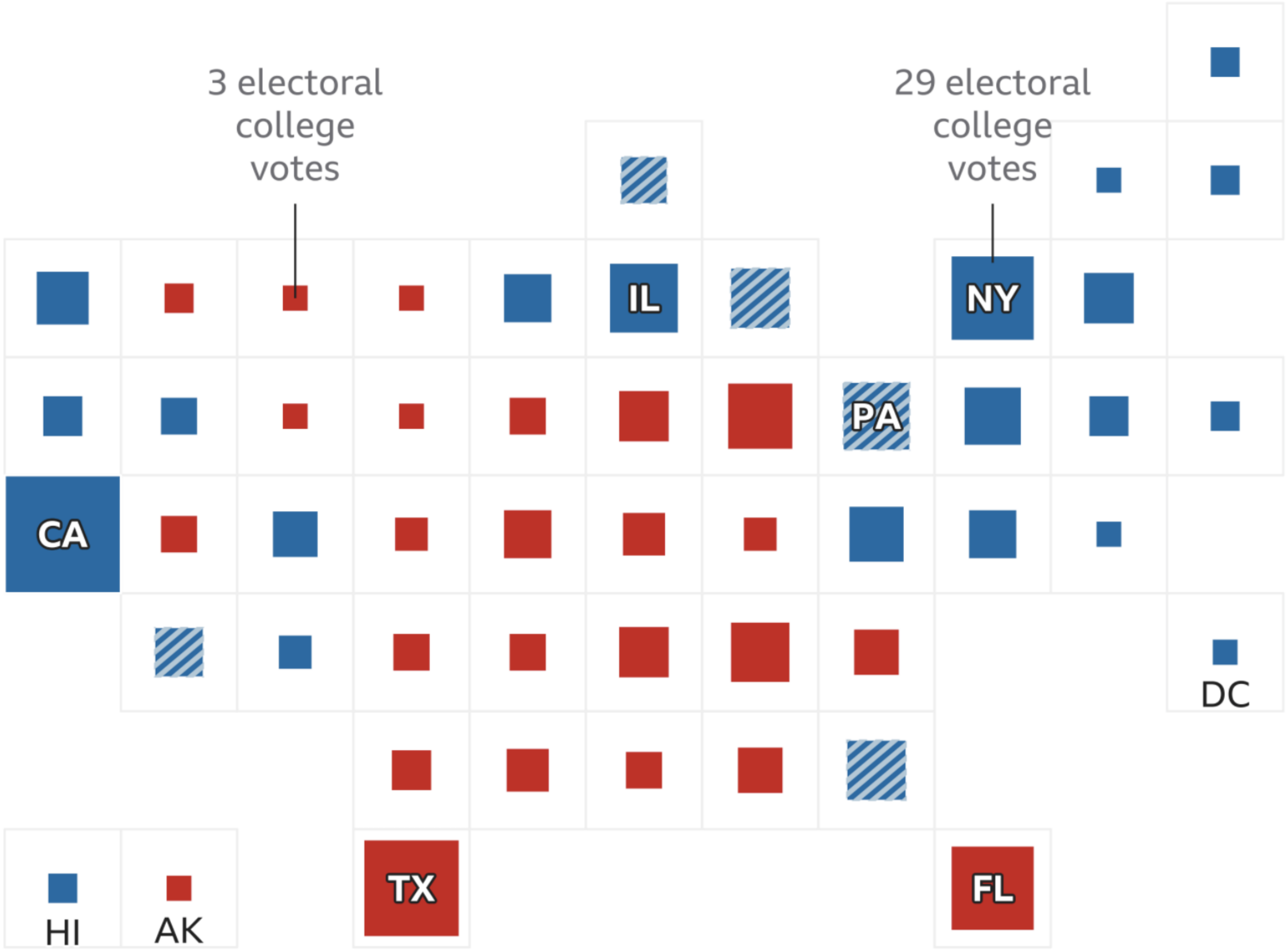
https://www.esri.com/arcgis-blog/products/product/mapping/mapping-coronavirus-responsibly/



https://www.bbc.co.uk/news/election-us-2020-54783016



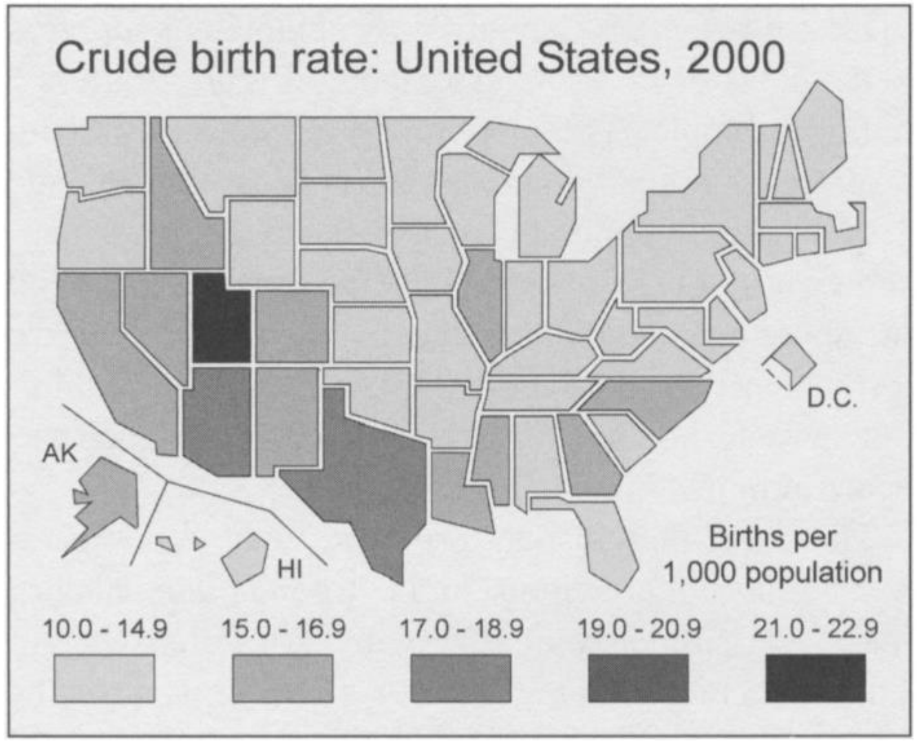
https://www.bbc.co.uk/news/election-us-2020-54783016

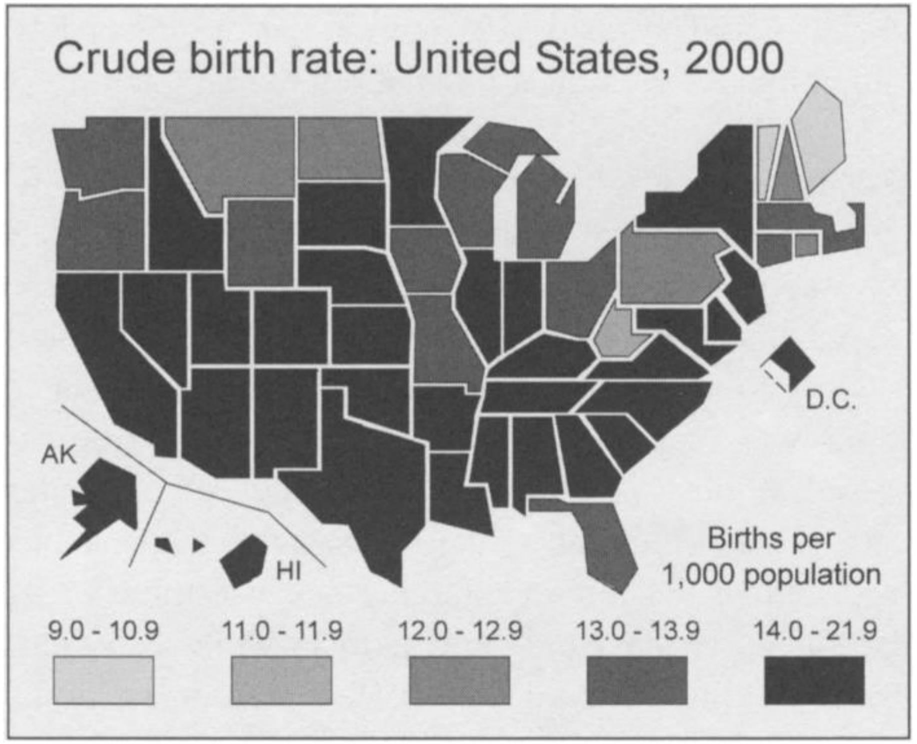


https://www.bbc.co.uk/news/election-us-2020-54783016

The first thing people do is looking at a visualisation

(not the legend)


Crude birth rate: United States, 2000
AK
HI
D.C.
Births per 1,000 population
10.0 - 14.9
15.0 - 16.9
17.0 - 18.9
19.0 - 20.9
21.0 - 22.9


Crude birth rate: United States, 2000
AK
HI
D.C.
Births per 1,000 population
9.0 - 10.9
11.0 - 11.9
12.0 - 12.9
13.0 - 13.9
14.0 - 21.9

Choropleth maps: “darker-means-more metaphor”

Source: Monmonier (2005)

Now some DIY …